

# A311D2

## Datasheet

---

**Revision: 0.6**


**Release Date: 2021-11-30**

---

## Copyright

©2021 Amlogic. All rights reserved. No part of this document may be reproduced, transmitted, transcribed, or translated into any language in any form or by any means without the written permission of Amlogic.

## Trademarks

 , and other Amlogic icons are trademarks of Amlogic companies. All other trademarks and registered trademarks are property of their respective holders.

## Disclaimer

Amlogic may make improvements and/or changes in this document or in the product described in this document at any time.

This product is not intended for use in medical, life saving, or life sustaining applications.

Circuit diagrams and other information relating to products of Amlogic are included as a means of illustrating typical applications. Consequently, complete information sufficient for production design is not necessarily given. Amlogic makes no representations or warranties with respect to the accuracy or completeness of the contents presented in this document.

## Contact Information

- Website: [www.amlogic.com](http://www.amlogic.com)
- Pre-sales consultation: [contact@amlogic.com](mailto:contact@amlogic.com)
- Technical support: [support@amlogic.com](mailto:support@amlogic.com)

# Revision History

## Issue 0.6 (2021-11-30)

This is the 0.6 release.

Compared to last version, the following contents are changed.

Section	Change Description
3	Updated specs of memory and storage interface.
7.8.3.1	Updated the register description of PADCTRL_PIN_MUX_REGM.

## Issue 0.5 (2021-09-18)

This is the 0.5 release.

Compared to last version, the following contents are changed.

Section	Change Description
12.1.9	Updated the default value of DDR registers.
3	Updated specs of memory and storage interface.

## Issue 0.4 (2021-06-21)

This is the 0.4 release.

Compared to the previous version, the following content is changed.

Section	Change Description
5.5.1	Updated R <sub>PU/PD</sub> typ spec.
5.2	Updated AVDD18 and VDD18_AO min spec. Updated VDD_EE and VDD_GPU Typ. spec. Added a note about VDD_DDR.
5.6.7	Updated t <sub>RC</sub> max spec.
5.5.3	Added a note about VDDQ and VDDQLP.
5.3	Updated ripple voltage spec.

## Issue 0.3 (2021-04-08)

This is the 0.3 release.

Compared to the previous version, the following content is changed.

Section	Change Description
3	Updated AV1 video decoding spec.

## Issue 0.2 (2021-03-16)

This is the preliminary 0.2 release.

Compared to the previous version, the following topic is changed.

---

Section	Change Description
<a href="#">7.3.2</a>	Added a figure showing power on sequence.

Compared to the previous version, the following topics are added.

- [7.2.1](#)
- [7.2.2](#)
- [7.2.3](#)
- [7.2.4](#) and its child topics

### **Issue 0.1 (2021-01-29)**

This is the preliminary 0.1 release.

# Contents

---

Revision History .....	ii
<b>1 About This Document</b> .....	<b>1</b>
<b>2 General Description</b> .....	<b>2</b>
<b>3 Features Summary</b> .....	<b>3</b>
<b>4 Pinout Specification</b> .....	<b>8</b>
4.1 Pinout Diagram .....	8
4.2 Pin Order .....	8
4.3 Pin Description .....	18
4.4 Pin Multiplexing Tables .....	38
4.5 Signal Description.....	47
<b>5 Operating Conditions</b> .....	<b>57</b>
5.1 Absolute Maximum Ratings .....	57
5.2 Recommended Operating Conditions .....	57
5.3 Ripple Voltage Specifications .....	58
5.4 Thermal Resistance.....	59
5.5 DC Electrical Characteristics.....	59
5.5.1 Normal GPIO Specifications.....	60
5.5.2 Open Drain GPIO Specifications (For DIO_OD) .....	60
5.5.3 DDR4/LPDDR4/LPDDR4X SDRAM Specifications.....	61
5.6 Timing Information .....	62
5.6.1 I2C Timing Specification .....	63
5.6.2 EMMC/SD Timing Specification .....	64
5.6.3 NAND Timing Specification .....	68
5.6.4 SPICC Timing Specification .....	71
5.6.5 SPIFC Timing Specification.....	71
5.6.6 Ethernet Timing Specification.....	72
5.6.7 Audio Timing Specification .....	75
5.6.8 PDM Timing Specification .....	78
5.6.9 UART Timing Specification .....	79
5.7 Recommended Oscillator Electrical Characteristics.....	79
5.8 Power On Config .....	80
5.9 Power On Reset .....	80
5.10 Recommended Power On Sequence .....	81
5.11 Power Consumption.....	82
5.12 Storage and Baking Conditions .....	83
<b>6 Mechanical Dimensions</b> .....	<b>85</b>
<b>7 System</b> .....	<b>86</b>

7.1 Memory Map.....	86
7.2 Power Domain .....	90
7.2.1 A73 & A53 Power Domain.....	93
7.2.2 General Power Domain.....	94
7.2.3 Power Switch Sequence .....	95
7.2.4 Power Control.....	96
7.2.5 Register Description.....	99
7.3 System Booting .....	124
7.3.1 Overview .....	124
7.3.2 Power-on Flow Chart .....	124
7.4 CPU .....	125
7.4.1 Overview .....	125
7.4.2 Register Description.....	126
7.5 GPU .....	152
7.6 Clock .....	152
7.6.1 Overview .....	152
7.6.2 24MHz Crystal Oscillator Layout .....	154
7.6.3 Frequency Calculation .....	155
7.6.4 Clock Measure.....	168
7.6.5 Register Description.....	177
7.7 Reset.....	213
7.7.1 Overview .....	214
7.7.2 Register Description.....	215
7.8 GPIO .....	221
7.8.1 Overview .....	222
7.8.2 GPIO Multiplex Function .....	222
7.8.3 Register Description.....	229
7.9 Interrupt Control.....	236
7.9.1 Overview .....	236
7.9.2 Interrupt Source.....	237
7.9.3 Register Description.....	239
7.10 Timer .....	242
7.10.1 Overview .....	243
7.10.2 General-Purpose Timer.....	243
7.10.3 Watchdog Timer.....	243
7.10.4 Register Description.....	244
7.11 Crypto.....	247
7.11.1 Overview.....	248
7.11.2 Key Ladder .....	248
7.11.3 RNG .....	248

7.11.4 CTR_DRBG .....	249
7.11.5 EFUSE.....	249
7.12 MIPI_ISP TOP .....	249
7.12.1 MIPI_CSI_DPHY .....	249
7.12.2 MIPI_ISP_ADAPT.....	249
7.12.3 ISP_AHB_DMA .....	251
7.12.4 ISP_APB_DMA.....	251
7.12.5 Register Description.....	252
7.13 Mailbox.....	390
7.13.1 Overview .....	390
7.13.2 Register Description.....	390
<b>8 GE2D .....</b>	<b>402</b>
8.1 Overview .....	402
8.2 Register Description .....	404
<b>9 Video Path .....</b>	<b>434</b>
9.1 Video Input .....	434
9.1.1 Features .....	434
9.1.2 TS Input.....	435
9.1.3 Crypto Algorithm .....	436
9.1.4 TS Output .....	436
9.1.5 HDMIRX .....	441
9.1.6 CEC Registers .....	482
9.2 Video Output.....	507
9.2.1 Overview .....	507
9.2.2 VPU.....	507
9.2.3 Register Description.....	507
<b>10 Audio Path .....</b>	<b>1214</b>
10.1 Audio Clock Tree .....	1214
10.2 Audio Input .....	1217
10.2.1 Overview .....	1218
10.2.2 TDM Input Interface .....	1218
10.2.3 SPDIF Input Interface.....	1218
10.2.4 PDM .....	1219
10.2.5 eARC RX.....	1221
10.3 Audio Output.....	1245
10.3.1 Overview .....	1245
10.3.2 TDM Output Interface.....	1245
10.3.3 SPDIF Output Interface.....	1246
10.3.4 Audio REQ_FRDDR .....	1247
10.3.5 eARC TX .....	1249

10.4 DSP	1269
10.4.1 DSP_TOP	1270
10.4.2 DSP IRQ CTRL	1271
10.4.3 GPIO 32/QIF 32 Interface	1273
10.4.4 DSP Registers	1274
10.5 DDR Datapath	1291
10.5.1 Overview	1291
10.5.2 Audio TODDR	1291
10.5.3 Audio FRDDR	1291
10.6 Audio Loopback	1292
10.7 Audio Resample A/B	1293
10.8 Audio VAD	1294
10.9 Audio EQDRC	1295
10.10 Audio Locker	1296
10.11 Register Description	1298
10.11.1 CLK/RESET Registers	1298
10.11.2 OTHER Registers	1312
10.11.3 TODDR Registers	1319
10.11.4 FRDDR Registers	1326
10.11.5 DDR ARB Registers	1329
10.11.6 LoopBack Registers	1331
10.11.7 TDM Registers	1337
10.11.8 SPDIF Registers	1357
10.11.9 ResampleA Registers	1369
10.11.10 TOACODEC Registers	1374
10.11.11 TOHDMITX Registers	1377
10.11.12 FRHDMIRX	1379
10.11.13 TOVAD Registers	1379
10.11.14 PDM Registers	1379
10.11.15 LOCKER Registers	1383
10.11.16 VAD Registers	1387
10.11.17 EQDRC Registers	1395
10.11.18 EARC RX Registers	1417
<b>11 AFIFO</b>	<b>1439</b>
11.1 Register Description	1439
<b>12 Memory Interface</b>	<b>1444</b>
12.1 DDR	1444
12.1.1 Overview	1444
12.1.2 AXI to DDR Address Remapping	1445
12.1.3 DMC AXI Address Security Region Checking and Data Scrambling	1446



12.1.4 DMC Security Register Bit Assignment .....	1450
12.1.5 AXI0 CPU & GPU Virtual Channel .....	1453
12.1.6 DMC AXI Port QoS Control .....	1454
12.1.7 DMC TEST Module .....	1454
12.1.8 Clock, Refresh, Retraining and Frequency Change for 2 DMCs .....	1454
12.1.9 Register Description .....	1455
12.2 eMMC/SD .....	1533
12.2.1 Overview .....	1533
12.2.2 Pin Description .....	1533
12.2.3 eMMC/SD Mode .....	1534
12.2.4 Descriptor .....	1534
12.2.5 Register Description .....	1537
12.3 Serial Peripheral Interface Communication Controller .....	1545
12.3.1 Overview .....	1545
12.3.2 Features .....	1546
12.3.3 Function .....	1547
12.3.4 Register Description .....	1547
12.4 Serial Peripheral Interface Flash Controller .....	1554
12.4.1 Overview .....	1554
12.4.2 Features .....	1554
12.4.3 Register Description .....	1555
<b>13 I/O Interface</b> .....	<b>1565</b>
13.1 Universal Serial Bus .....	1565
13.1.1 Overview .....	1565
13.1.2 Features .....	1565
13.1.3 Register Description .....	1565
13.2 PCIE .....	1574
13.3 Ethernet .....	1574
13.3.1 Overview .....	1574
13.3.2 Ethernet Top Registers .....	1574
13.3.3 Ethernet MAC .....	1576
13.3.4 Ethernet PHY .....	1577
13.4 Inter-Integrated Circuit (I2C) .....	1582
13.4.1 Overview .....	1583
13.4.2 Features .....	1583
13.4.3 Register Description .....	1583
13.5 Universal Asynchronous Receiver And Transmitter .....	1593
13.5.1 Overview .....	1593
13.5.2 Features .....	1594
13.5.3 Functional Description .....	1595

---

13.5.4 Register Description.....	1595
13.6 Infrared Remote.....	1599
13.6.1 Overview .....	1599
13.6.2 IR Demodulation .....	1599
13.6.3 Legacy IR Control .....	1602
13.6.4 Multi Format IR Control .....	1603
13.6.5 Register Description.....	1604
13.7 Pulse-Width Modulation .....	1620
13.7.1 Overview .....	1620
13.7.2 Register Description.....	1622
13.8 SAR ADC .....	1625
13.8.1 Overview .....	1625
13.8.2 Register Description.....	1626
<b>14 System Interface</b> .....	<b>1636</b>
14.1 JTAG .....	1636
14.2 Temp Sensor .....	1636
14.2.1 Overview .....	1636
14.2.2 Register Description.....	1637
<b>15 Appendix</b> .....	<b>1644</b>
15.1 Net Length.....	1644

# 1 About This Document

---

This document is applicable for A311D2 series SoCs, please contact your Amlogic sales representative for details.

## 2 General Description

A311D2 is an advanced application processor designed for smart display in-vehicle-infotainment (IVI) applications. It integrates a powerful CPU/GPU subsystem, a secured 8K video CODEC engine with all major peripherals to form the ultimate high-performance smart display IVI chip.

The main system CPU is based on Big.LITTLE architecture which integrates a quad-core ARM Cortex-A73 CPU cluster and a quad-core Cortex-A53 cluster with unified L2 cache for each cluster to improve system performance. In addition, the CPU includes the NEON SIMD co-processor to improve software media processing capability.

The graphic subsystem consists of two graphic engines and a flexible video/graphic output pipeline. The ARM Mali-G52 MP8 (8EE) GPU handles all OpenGL ES 3.2, Vulkan 1.1 and OpenCL 2.0 graphic programs, while the 2.5D graphics processor handles additional scaling, alpha, rotation and color space conversion operations. Together, the CPU and GPU handle all operating system, network, user-interface and game related tasks.

Amlogic Video Engine (AVE-10) is a subsystem which uses dedicated hardware video decoders and encoders to offloads the CPUs from all video CODEC processing. AVE-10 is capable of decoding 8K4K resolution video within Trusted Video Path (TVP) for secured DRM applications. It supports all major video formats including MPEG-1/2/4, VC-1/WMV, AVS+, AVS2, MJPEG, H.264, H265-10, VP9-10, AV1 and also JPEG. The independent encoder can concurrently encode in JPEG up to 1080P at 60fps and H.265/H.264 up to 4K at 50fps.

The video/graphics output pipeline includes Dolby Vision <sup>Optional</sup>, HDR10+, HDR10, HLG and Technicolor Prime HDR processing, BT.2020/ BT.2100 processing, motion compensated and motion adaptive de-interlacer, flexible programmable super scalar, local dimming and many picture enhancement filters before passing the enhanced image to the video output ports. It supports 3 separate display controllers that can be flexibly configured to the V-by-One, LVDS, eDP, MIPI-DSI and HDMI output.

3 HDMI 2.1 receiver ports are available. The HDMI ports can receive up to 4K2K HDR video and support dynamic HDR, ALLM, QFT, QMS, VRR and HDCP 1.4/2.3.

The built-in three demux can process the video streams from the serial transport stream input interface, which can connect to external tuner/demodulator. DVB Common Descrambler 1.0 is supported in addition to DES, Triple DES (TDES/3DES) and AES streaming crypto formats. An integrated ISO7816 controller is included for interfacing to external smart card.

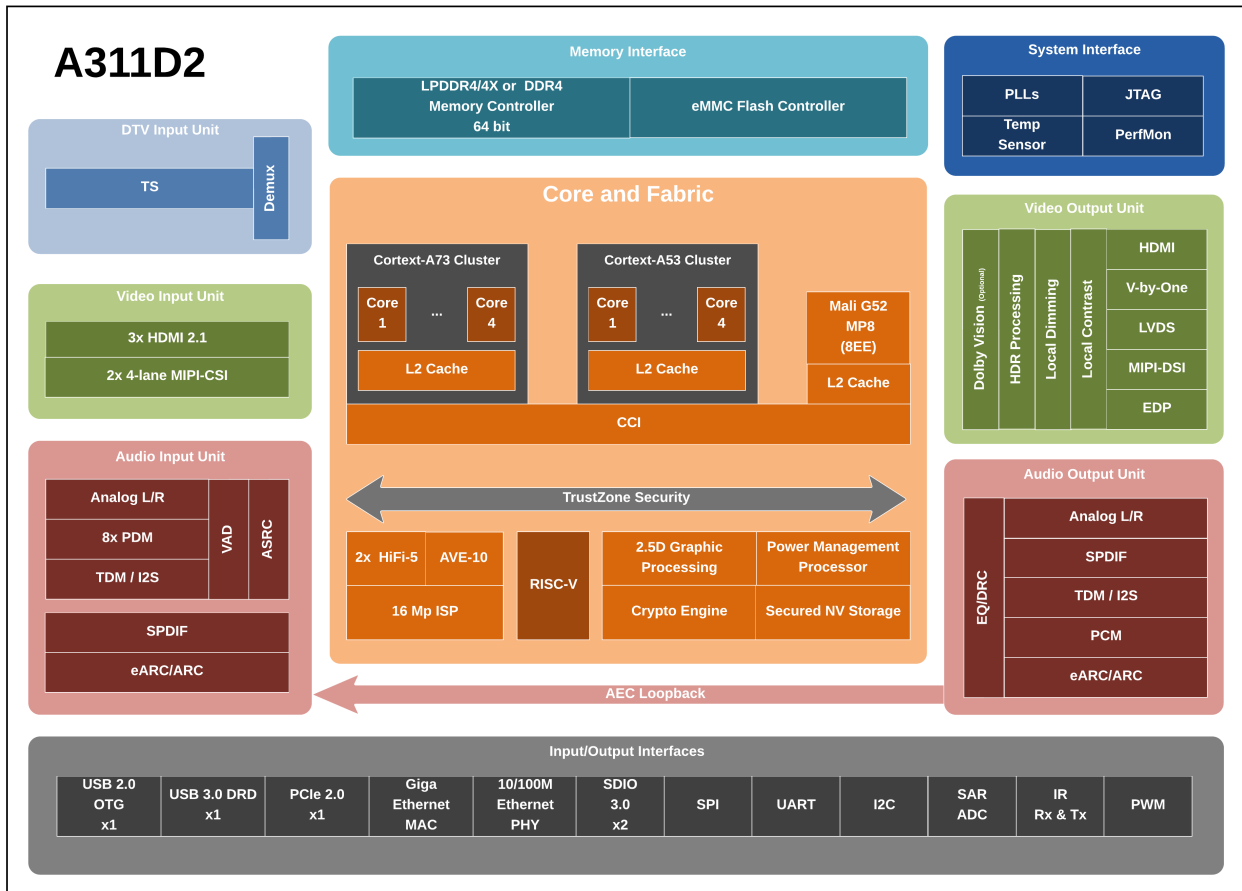
It is optimized for low power far-field voice application. Two dedicated HiFi 5 DSPs offload the main CPU for the top of the line audio front end and wake-word algorithms. It also has built-in Voice Activity Detection (VAD) module for ultra-low power operations during system standby and full digital MIC interface including PDM, TDM and I2S up to 8 channels are available.

The SoC integrates rich advanced network and peripheral interfaces, including a 10/100/1000M Ethernet MAC with RGMII, 10/100M Ethernet PHY, USB3.0 DRD, PCIe 2.0 and USB 2.0 high-speed port, SDIO 3.0 controller, eMMC 5.1 controller and multiple SDIO/SD card controllers, UART, I2C, high-speed SPI PWMs and a built-in IR blaster. The flexible and programmable QoS-based switch fabric and memory controller tie all the processing cores and peripherals together and connects to the DRAM memory bus.

Standard development environment utilizing SecureOS, Linux and GNU/GCC Android tool chain is supported. Please contact your AMLOGIC sales representative for more information.

# 3 Features Summary

Figure 3-1 Features Summary



TOB8T01

## CPU Sub-system

- Quad core ARM Cortex-A73 + Quad core Cortex-A53 CPU
- ARMv8.0 architecture with Neon extensions and cryptography extension
- Unified system L2 cache for each cluster
- Advanced TrustZone security system
- Application based traffic optimization using internal QoS-based switching fabrics
- CoreSight debugger support

## 3D Graphics Processing Unit

- ARM Mali-G52 MP8 (8EE) GPU
- 8-wide warps, 2x dual texture pipe, 8x 8-wide execution engines (EE)
- Concurrent multi-core processing
- OpenGL ES 3.2, Vulkan 1.1 and OpenCL 2.0 support

## 2.5D Graphics Processor

- Fast bitblt engine with dual inputs and single output
- Programmable raster operations (ROP)
- Programmable polyphase scaling filter

- Supports multiple video formats 4:2:0, 4:2:2 and 4:4:4 and multiple pixel formats (8/16/24/32 bits graphics layer)
- Fast color space conversion
- Advanced anti-flickering filter

### Crypto Engine

- AES block cipher with 128/256 bits keys, standard 16 bytes block size and streaming ECB, CBC and CTR modes
- DES/3DES block cipher with ECB and CBC modes supporting 64 bits key for DES and 192 bits key for 3DES
- Hardware key-ladder operation and DVB-CSA for transport stream encryption
- Built-in hardware True Random Number Generator (TRNG) and SHA-1/SHA-2 engine

### Video/Picture CODEC

- Amlogic Video Engine (AVE-10) with dedicated hardware decoders up to 8Kx4K@24fps and encoders up to 4Kx2K@50fps
- Support multi-video decoder up to 4Kx2K@60fps + 1x1080P@60fps
- Supports multiple "secured" video decoding sessions and simultaneous decoding and encoding
- Video/Picture Decoding
  - AV1 MP-10@5.1 up to 8Kx4K@24fps or 4Kx2K@60fps
  - H.265 HEVC MP-10@L5.1 up to 8Kx4K@24fps or 4Kx2K@60fps
  - VP9 Profile 2-10 up to 8Kx4K@24fps or 4Kx2K@60fps
  - AVS2 MP up to 4Kx2K@60fps
  - H.264 AVC HP@L5.1 up to 4Kx2K@30fps
  - MPEG-4 ASP@L5 up to 1080P@60fps (ISO-14496)
  - WMV/VC-1 SP/MP/AP up to 1080P@60fps
  - AVS-P16(AVS+) /AVS-P2 JiZhun Profile up to 1080P@60fps
  - MPEG-2 MP/HL up to 1080P@60fps (ISO-13818)
  - MPEG-1 MP/HL up to 1080P@60fps (ISO-11172)
  - Multiple language and multiple format sub-title video support
  - MJPEG and JPEG unlimited pixel resolution decoding (ISO/IEC-10918)
  - Supports JPEG thumbnail, scaling, rotation and transition effects
  - Supports \*.mkv,\*.wmv,\*.mpg, \*.mpeg, \*.dat, \*.avi, \*.mov, \*.iso, \*.mp4 and \*.jpg file formats
- Video/Picture Encoding
  - H.265 main-profile and H.264 high-profile video encoder up to 4K@50fps with low latency
  - Independent JPEG and H.264 encoder with configurable performance/bit-rate
  - JPEG image encoding up to 1080P at 60fps

### 11<sup>th</sup> Generation Advanced Amlogic TruLife Image Engine

- Supports Dolby Vision <sup>Optional</sup>, HDR10/10+, HLG, HLG, Prime HDR
- Full 12-bit internal data processing
- Motion compensated noise reduction and 3D digital noise reduction for random noise up to 4K resolution
- Block noise, mosquito noise, spatial noise and temporal noise reduction
- Motion compensated and motion adaptive de-interlacer

- Edge interpolation with low angle protection and processing
- 3:2/2:2 pulldown and Video on Film (VOF) detection and processing
- Smart sharpness with SuperScaler technology including de-contouring, de-ring, LTI, CTI, de-jaggy, peaking. Reproduce more high-frequency details while protecting flesh tones flexibly.
- Local contrast and dynamic non-Linear contrast for detail enhancement
- 3D LUTs with 17x17x17 nodes, provide 4913 different control points, which is competent for matching calibrated displays to a target color space
- High precision HSL color space based color management with low saturation protection, independent luma/hue/saturation adjustment to achieve blue/green extension, fresh tone correction, and wider gamut for video
- Video mixer: 4x1080p60 or 1x4Kp60 + 1x1080p60 graphic plane and 3x4Kp60 video plane
- Independent HDR re-mapping of video and graphic layer
- Local dimming control for high nits backlights

### Video Input/output Interface

- 3x HDMI 2.1 receiver and 1x HDMI 2.1 transmitter ports with EMP, ALLM, QFT, QMS, VRR, eARC, HDCP 1.4 /2.3, and up to 4Kx2K@60 max resolution
- 2x 8-lane V-by-One output with 1, 2, 4 regions supported, up to 4Kx2K 60Hz resolution
- LVDS output supporting up to 3x 1280x720 or 1x 1280x720 + 1x 1920x1080 resolution
- 2x 4-lane MIPI-DSI interface with panel calibration, resolution up to 2x 1920x1200 for each port
- 2x 4-lane eDP supporting panel up to 2560x1600 resolution
- Three independent Gamma table for LCD panel tuning
- Dithering logic for mapping to different LCD panel color depth

### Camera Interface

- MIPI-CSI camera interface with 2x4 lanes
- Supports RAW, YUV or RGB camera input formats

### ISP

- 16 Mp ISP with the resolution up to 4608x3456
- HDR sensor de-companding/ Dynamic Defect Pixel Correction/ Green channel equalization/ EW image flip
- 3A (AE/AWB/AF)
- Spatial/ Temporal/ Color Noise reduction
- Static White balance
- Lens shading correction
- Demosaic
- Chromatic aberration/ purple Fringing correction
- linear matrix color correction
- Gamma correction

### GDC (Graphic Distortion Correction)

- Supports single frame off line processing up to 8192x8192 video source
- Supports processing performance up to 8M@60fps for single mode
- Supports YUV semi-planer 420 format for both input and output
- Supports gray scale image processing
- Supports 360/ 180/ normal fish eye correction modes

- Supports top/ wall/ table installation mode

## Audio Processing and Input/Output

- HiFi 5 audio DSP <sup>Optional</sup> for highly optimized audio/voice processing
  - 64 KB Instruction Cache and 96 KB Data Cache
  - 32 MPU entries
  - MUL32, MUL16 and DIV32
  - HiFi 5 Single Precision Vector FP
  - HiFi 5 Half Precision Vector FP
  - HiFi 5 Neural Network Extension
- Supports MP3, AAC, WMA, RM, FLAC, Ogg, Dolby Audio <sup>Optional</sup>, DTS <sup>Optional</sup> and programmable with 7.1/5.1 down-mixing
- Low-power VAD and internal AEC loopback path
- 3 built-in TDM/I2S ports with TDM/PCM mode up to 192kHz x 32bits x 8ch or 48kHz x 32bits x 32ch and I2S mode up to 192kHz x 32bits x 8ch
- Digital microphone PDM voice input with programmable CIC, LPF and HPF, support up to 8 DMICs
- 1 L/R analog input channel and 1 L/R output channel
- Supports concurrent dual audio stereo channel output with combination of I2S+PCM
- Supports Audio EQ/DRC for audio speaker
- Supports dynamic EQ adjustment

## DTV Broadcasting Interface

- 3x Transport stream (TS) input interface with built-in demux processor for connecting to external digital TV tuner/demodulator
- Built-in PWM, I2C and SPI interfaces to control tuner and demodulator
- Integrated ISO 7816 smart card controller

## Memory and Storage Interface

- 64-bit DRAM memory interface with dual ranks, max 16GB total logic address space and max 8GB total DDR capacity physical address
- Compatible with JEDEC standard DDR4-3200 /LPDDR4/LPDDR4X-4266 SDRAM
- 4-bit SD card and 4-bit SDIO interface support up to SDR104/HS200 mode
- 8-bit eMMC memory interface support up to HS400 mode
- Built-in 8k bits OTP memory for secured key storage

## Network Interface

- IEEE 802.3 10/100/1000M Ethernet MAC with RGMII interface
- 10/100M Ethernet PHY interface
- WiFi/IEEE802.11 supporting via PCIe, USB or SDIO
- Bluetooth supporting via USB or UART
- Network interface optimized for mixed WIFI and BT traffic

## Integrated I/O Controllers and Interfaces

- One USB 2.0 high-speed USB I/O which supports USB OTG
- One USB 3.0 DRD port up to 5Gbps
- One PCIe 2.0 (Root Complex) port



- Multiple UARTs, I2Cs and PWMs
- SPI interface
- Programmable remote control input circuitry and IR-blaster output
- Built-in 10bit SAR ADC with 5 channels
- General Purpose IOs with built-in pull up and pull down

### **System, Peripherals and Misc. Interfaces**

- Integrated general purpose timers, counters, DMA controllers
- 24 MHz crystal input
- Embedded debug interface using ICE/JTAG

### **Power Management**

- Multiple internal power domains controlled by software
- Multiple sleep modes for CPU, system, DRAM, etc.
- Multiple internal PLLs to adjust the operating frequencies
- Multi-voltage I/O design for 1.8V and 3.3V

### **Security**

- Trustzone based Trusted Execution Environment (TEE)
- Secured boot, encrypted hardware self-setup OTP, encrypted DRAM with memory integrity checker, hardware key ladder and internal control buses and storage
- Separated secure/non-secure Entropy true RNG and video watermarking
- Pre-region/ID memory security control and electric fence
- Hardware based Trusted Video Path (TVP), and secured contents (needs SecureOS software)
- Secured IO and secured clock

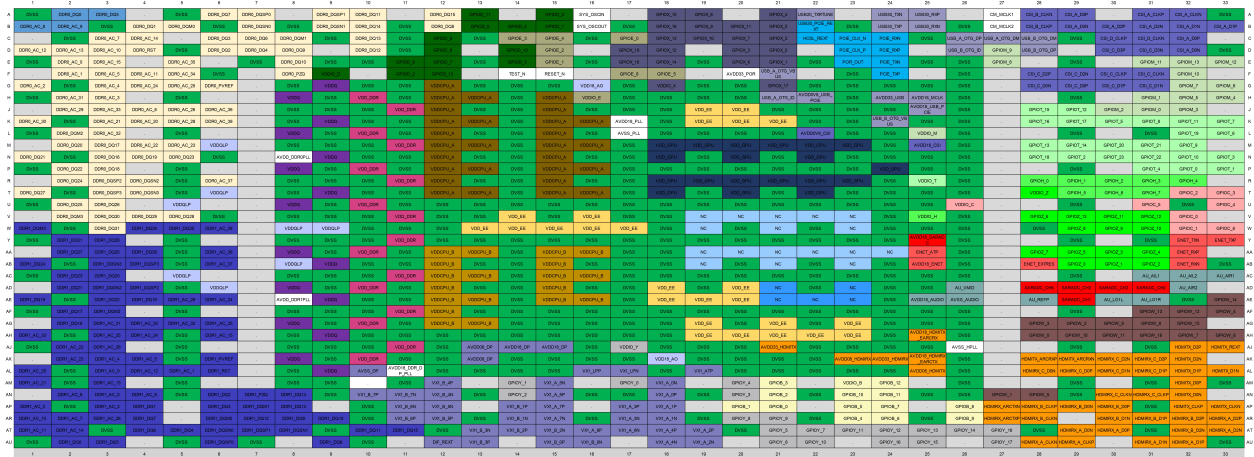
### **Package**

- FCBGA, 14.3 mm x 16.1 mm, 0.4 ball pitch, RoHS compliant

# 4 Pinout Specification

## 4.1 Pinout Diagram

Figure 4-1 Pinout Diagram (top view)



## 4.2 Pin Order

BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
A1	DVSS	A28	CSI_B_CLKN	B15	GPIOD_1
A2	DDR0_DQ0	A29	CSI_A_D3P	B16	SYS_OSCOUT
A3	DDR0_DQ5	A31	CSI_A_CLKP	B17	DVSS
A5	DVSS	A32	CSI_A_CLKN	B18	GPIOX_16
A6	DDR0_DQ7	A33	DVSS	B19	GPIOX_9
A7	DDR0_DQSP0	B1	DDR0_AC_8	B20	GPIOX_11
A9	DDR0_DQSP1	B2	DDR0_AC_9	B21	GPIOX_5
A10	DDR0_DQ11	B3	DVSS	B22	USB30_PCIE_REXT
A12	DDR0_DQ15	B4	DDR0_DQ1	B23	DVSS
A13	GPIOD_11	B5	DDR0_DQM0	B24	USB30_TXP
A15	GPIOD_0	B6	DVSS	B25	USB30_RXN
A16	SYS_OSCIN	B7	DDR0_DQSN0	B26	DVSS
A18	GPIOX_15	B8	DVSS	B27	CM_MCLK2
A19	GPIOX_8	B9	DDR0_DQSN1	B28	CSI_B_CLKP
A21	GPIOX_4	B10	DDR0_DQ14	B29	CSI_A_D3N
A22	USB20_TXRTUNE	B11	DVSS	B30	CSI_A_D2P
A24	USB30_TXN	B12	DDR0_DQ8	B31	CSI_A_D2N
A25	USB30_RXP	B13	GPIOD_5	B32	CSI_A_D1N
A27	CM_MCLK1	B14	GPIOD_4	B33	CSI_A_D1P

BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
C2	DVSS	D7	DDR0_DQ4	E19	DVSS
C3	DDR0_AC_7	D8	DDR0_DQ9	E20	GPIOX_6
C4	DDR0_AC_14	D10	DDR0_DQ12	E21	GPIOX_1
C6	DDR0_DQ3	D11	DVSS	E22	DVSS
C7	DDR0_DQ6	D12	GPIOD_6	E23	POR_OUT
C8	DDR0_DQM1	D14	GPIOD_10	E24	PCIE_TXN
C9	DVSS	D15	GPIOE_2	E25	DVSS
C10	DDR0_DQ13	D17	GPIOX_19	E26	DVSS
C11	DVSS	D18	GPIOX_12	E27	GPIOM_8
C12	GPIOD_9	D20	GPIOX_3	E29	DVSS
C13	DVSS	D21	GPIOX_0	E31	GPIOM_11
C14	GPIOE_3	D23	PCIE_CLK_P	E32	GPIOM_13
C15	GPIOE_4	D24	PCIE_RXP	E33	GPIOM_12
C16	DVSS	D26	USB_B_OTG_ID	F2	DDR0_AC_1
C17	GPIOE_0	D27	GPIOM_9	F3	DDR0_AC_5
C18	GPIOX_13	D28	USB_B_OTG_DP	F4	DDR0_AC_11
C19	GPIOX_10	D30	CSI_C_D3P	F5	DDR0_AC_34
C20	GPIOX_7	D31	CSI_C_D3N	F6	DVSS
C21	GPIOX_2	D32	CSI_A_D0N	F8	DDR0_PZQ
C22	HCSL_REXT	D33	DVSS	F9	VDDIO_D
C23	PCIE_CLK_N	E1	DVSS	F11	GPIOD_2
C24	PCIE_RXN	E2	DDR0_AC_0	F12	GPIOD_12
C25	DVSS	E3	DDR0_AC_15	F14	TEST_N
C26	USB_A_OTG_DP	E5	DDR0_AC_35	F15	RESET_N
C27	USB_A_OTG_DM	E7	DVSS	F17	GPIOE_6
C28	USB_B_OTG_DM	E8	DDR0_DQ10	F18	GPIOE_5
C29	DVSS	E9	DVSS	F20	AVDD33_POR
C30	CSI_D_CLKP	E10	DVSS	F21	USB_A_OTG_VBUS
C31	CSI_D_CLKN	E11	GPIOD_8	F23	DVSS
C32	CSI_A_D0P	E12	GPIOD_7	F24	PCIE_TXP
D1	DDR0_AC_12	E13	DVSS	F26	DVSS
D2	DDR0_AC_13	E14	GPIOD_3	F28	CSI_C_D2P
D3	DDR0_AC_10	E15	GPIOE_1	F29	CSI_C_D2N
D4	DDR0_RST	E16	DVSS	F30	CSI_C_CLKP
D5	DVSS	E17	GPIOX_18	F31	CSI_C_CLKN
D6	DDR0_DQ2	E18	GPIOX_14	F32	GPIOM_10

BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
G1	DDR0_AC_2	H10	VDD_DDR	J19	VDD_EE
G2	DVSS	H11	DVSS	J20	VDD_EE
G3	DDR0_AC_4	H12	VDDCPU_A	J21	DVSS
G4	DDR0_AC_24	H13	VDDCPU_A	J22	DVSS
G5	DDR0_AC_25	H14	DVSS	J23	DVSS
G6	DDR0_PVREF	H15	VDDCPU_A	J24	DVSS
G8	DVSS	H16	VDDIO_E	J25	AVDD18_USB_PCIE
G9	VDDQ	H17	DVSS	J26	DVSS
G10	DVSS	H18	DVSS	J28	GPIOT_15
G11	DVSS	H19	DVSS	J29	GPIOT_12
G12	VDDCPU_A	H20	DVSS	J30	GPIOM_2
G13	DVSS	H21	USB_A_OTG_ID	J31	GPIOM_0
G14	DVSS	H22	AVDD0V8_USB_PCIE	J32	GPIOM_3
G15	DVSS	H23	DVSS	K1	DDR0_AC_30
G16	VDD18_AO	H24	AVDD33_USB	K2	DVSS
G17	DVSS	H25	AVDD18_MCLK	K3	DDR0_AC_21
G18	VDDIO_X	H26	DVSS	K4	DDR0_AC_20
G19	DVSS	H29	DVSS	K5	DDR0_AC_28
G20	DVSS	H31	GPIOM_1	K6	DDR0_AC_38
G21	GPIOX_17	H32	GPIOM_5	K8	DVSS
G22	DVSS	H33	GPIOM_4	K9	VDDQ
G23	DVSS	J2	DDR0_AC_29	K10	DVSS
G24	DVSS	J3	DDR0_AC_33	K11	DVSS
G25	DVSS	J4	DDR0_AC_6	K12	VDDCPU_A
G26	DVSS	J5	DDR0_AC_26	K13	VDDCPU_A
G28	CSI_C_D0N	J6	DDR0_AC_36	K14	DVSS
G29	CSI_C_D0P	J8	DVSS	K15	VDDCPU_A
G30	CSI_C_D1P	J9	DVSS	K16	VDDCPU_A
G31	CSI_C_D1N	J10	DVSS	K17	AVDD18_PLL
G32	GPIOM_7	J11	VDD_DDR	K18	DVSS
G33	GPIOM_6	J12	VDDCPU_A	K19	VDD_EE
H1	DVSS	J13	DVSS	K20	VDD_EE
H2	DDR0_AC_31	J14	DVSS	K21	VDD_EE
H3	DDR0_AC_3	J15	VDDCPU_A	K22	DVSS
H5	DVSS	J16	DVSS	K23	DVSS
H8	VDDQ	J17	DVSS	K24	USB_B_OTG_VBUS
H9	DVSS	J18	DVSS	K25	DVSS

BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
K26	DVSS	M5	DDR0_AC_23	N12	VDDCPU_A
K28	GPIOT_16	M6	VDDQLP	N13	DVSS
K29	GPIOT_17	M8	DVSS	N14	DVSS
K30	GPIOT_5	M9	DVSS	N15	VDDCPU_A
K31	GPIOT_8	M10	DVSS	N16	DVSS
K32	GPIOT_11	M11	VDD_DDR	N17	DVSS
K33	GPIOT_7	M12	VDDCPU_A	N18	VDD_GPU
L1	DVSS	M13	VDDCPU_A	N19	DVSS
L2	DDR0_DQM2	M14	DVSS	N20	VDD_GPU
L3	DDR0_AC_32	M15	VDDCPU_A	N21	DVSS
L5	DVSS	M16	VDDCPU_A	N22	VDD_GPU
L8	VDDQ	M17	DVSS	N23	DVSS
L9	DVSS	M18	VDD_GPU	N24	DVSS
L10	VDD_DDR	M19	VDD_GPU	N25	DVSS
L11	DVSS	M20	VDD_GPU	N26	DVSS
L12	VDDCPU_A	M21	VDD_GPU	N28	GPIOT_18
L13	DVSS	M22	VDD_GPU	N29	GPIOT_2
L14	DVSS	M23	VDD_GPU	N30	GPIOT_23
L15	VDDCPU_A	M24	DVSS	N31	GPIOT_22
L16	DVSS	M25	AVDD18_CSI	N32	GPIOT_10
L17	AVSS_PLL	M26	DVSS	N33	GPIOT_3
L18	DVSS	M28	GPIOT_13	P1	DVSS
L19	DVSS	M29	GPIOT_14	P2	DDR0_DQ22
L20	DVSS	M30	GPIOT_20	P3	DDR0_DQ18
L21	DVSS	M31	GPIOT_21	P5	DVSS
L22	AVDD0V8_CSI	M32	GPIOT_9	P8	VDDQ
L23	DVSS	N1	DDR0_DQ21	P9	DVSS
L24	DVSS	N2	DVSS	P10	VDD_DDR
L25	VDDIO_M	N3	DDR0_DQ16	P11	DVSS
L26	DVSS	N4	DDR0_DQ19	P12	VDDCPU_A
L29	DVSS	N5	DDR0_DQ23	P13	VDDCPU_A
L31	DVSS	N6	DVSS	P14	DVSS
L32	GPIOT_19	N8	AVDD_DDR0PLL	P15	VDDCPU_A
L33	GPIOT_6	N9	VDDQ	P16	VDDCPU_A
M2	DDR0_DQ20	N10	DVSS	P17	DVSS
M3	DDR0_DQ17	N11	DVSS	P18	DVSS
M4	DDR0_AC_22			P19	DVSS

BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
P20	DVSS	R30	GPIOH_2	U5	VDDQLP
P21	DVSS	R31	GPIOH_3	U8	VDDQ
P22	DVSS	R32	GPIOH_4	U9	DVSS
P23	DVSS	T1	DDR0_DQ27	U10	VDD_DDR
P24	VDD_GPU	T2	DVSS	U11	DVSS
P25	DVSS	T3	DDR0_DQSP3	U12	DVSS
P26	DVSS	T4	DDR0_DQSN3	U13	DVSS
P29	DVSS	T5	DVSS	U14	DVSS
P31	GPLOT_4	T6	VDDQLP	U15	DVSS
P32	GPLOT_0	T8	DVSS	U16	DVSS
P33	GPLOT_1	T9	VDDQ	U17	DVSS
R2	DDR0_DQ24	T10	DVSS	U18	DVSS
R3	DDR0_DQSP2	T11	DVSS	U19	DVSS
R4	DDR0_DQSN2	T12	VDDCPU_A	U20	DVSS
R5	DVSS	T13	VDDCPU_A	U21	DVSS
R6	DDR0_AC_37	T14	DVSS	U22	DVSS
R8	DVSS	T15	VDDCPU_A	U23	DVSS
R9	DVSS	T16	VDDCPU_A	U24	DVSS
R10	DVSS	T17	DVSS	U25	DVSS
R11	VDD_DDR	T18	VDD_GPU	U26	VDDIO_C
R12	VDDCPU_A	T19	DVSS	U29	DVSS
R13	DVSS	T20	VDD_GPU	U31	GPIOC_5
R14	DVSS	T21	DVSS	U32	DVSS
R15	VDDCPU_A	T22	VDD_GPU	U33	GPIOC_4
R16	DVSS	T23	DVSS	V2	DDR0_DQM3
R17	DVSS	T24	DVSS	V3	DDR0_DQ30
R18	VDD_GPU	T25	DVSS	V4	DDR0_DQ29
R19	VDD_GPU	T26	DVSS	V5	DDR0_DQ28
R20	VDD_GPU	T28	VDDIO_Z	V6	DVSS
R21	VDD_GPU	T29	GPIOH_5	V8	DVSS
R22	VDD_GPU	T30	GPIOH_6	V9	DVSS
R23	VDD_GPU	T31	GPIOH_7	V10	DVSS
R24	DVSS	T32	GPIOC_2	V11	VDD_DDR
R25	VDDIO_T	T33	GPIOC_3	V12	DVSS
R26	DVSS	U1	DVSS	V13	DVSS
R28	GPIOH_0	U2	DDR0_DQ25	V14	VDD_EE
R29	GPIOH_1	U3	DDR0_DQ26	V15	DVSS
				V16	VDD_EE

BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
V17	DVSS	W24	DVSS	AA3	DDR1_DQ25
V18	DVSS	W25	DVSS	AA4	DDR1_DQ26
V19	NC	W26	DVSS	AA5	DVSS
V20	NC	W28	DVSS	AA6	DDR1_AC_36
V21	NC	W29	GPIOZ_8	AA8	VDDQ
V22	NC	W30	GPIOZ_9	AA9	DVSS
V23	NC	W31	GPIOZ_10	AA10	VDD_DDR
V24	DVSS	W32	GPIOC_1	AA11	DVSS
V25	VDDIO_H	W33	GPIOC_6	AA12	VDDCPU_B
V26	DVSS	Y1	DVSS	AA13	VDDCPU_B
V28	GPIOZ_6	Y2	DDR1_DQ31	AA14	DVSS
V29	GPIOZ_13	Y3	DDR1_DQ28	AA15	VDDCPU_B
V30	GPIOZ_11	Y5	DVSS	AA16	VDDCPU_B
V31	GPIOZ_12	Y8	DVSS	AA17	DVSS
V32	GPIOC_0	Y9	DVSS	AA18	DVSS
W1	DDR1_DQM3	Y10	DVSS	AA19	NC
W2	DVSS	Y11	VDD_DDR	AA20	NC
W3	DDR0_DQ31	Y12	DVSS	AA21	NC
W4	DDR1_DQ29	Y13	DVSS	AA22	NC
W5	DDR1_DQ30	Y14	DVSS	AA23	NC
W6	DDR1_AC_38	Y15	DVSS	AA24	NC
W8	VDDQLP	Y16	DVSS	AA25	ENET_ATP
W9	VDDQLP	Y17	DVSS	AA26	DVSS
W10	DVSS	Y18	DVSS	AA28	GPIOZ_7
W11	DVSS	Y19	DVSS	AA29	GPIOZ_5
W12	DVSS	Y20	DVSS	AA30	GPIOZ_3
W13	VDD_EE	Y21	DVSS	AA31	GPIOZ_4
W14	VDD_EE	Y22	DVSS	AA32	ENET_RXP
W15	VDD_EE	Y23	DVSS	AB1	DDR1_DQ24
W16	VDD_EE	Y24	DVSS	AB2	DVSS
W17	VDD_EE	Y25	AVDD18_SARADC	AB3	DDR1_DQSN3
W18	DVSS	Y26	DVSS	AB4	DDR1_DQSP3
W19	NC	Y29	DVSS	AB5	DVSS
W20	DVSS	Y31	DVSS	AB6	DDR1_AC_37
W21	NC	Y32	ENET_TXN	AB8	VDDQLP
W22	DVSS	Y33	ENET_TXP	AB9	VDDQ
W23	NC	AA2	DDR1_DQ27	AB10	DVSS

BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
AB11	DVSS	AC19	DVSS	AD29	SARADC_CH3
AB12	VDDCPU_B	AC20	DVSS	AD30	SARADC_CH2
AB13	DVSS	AC21	DVSS	AD31	SARADC_CH0
AB14	DVSS	AC22	DVSS	AD32	AU_AIR2
AB15	VDDCPU_B	AC23	DVSS	AE1	DDR1_DQ19
AB16	DVSS	AC24	DVSS	AE2	DVSS
AB17	DVSS	AC25	DVSS	AE3	DDR1_DQ22
AB18	DVSS	AC26	DVSS	AE4	DDR1_DQ16
AB19	NC	AC29	DVSS	AE5	DDR1_AC_29
AB20	DVSS	AC31	AU_AIL1	AE6	DDR1_AC_34
AB21	NC	AC32	AU_AIL2	AE8	AVDD_DDR1PLL
AB22	DVSS	AC33	AU_AIR1	AE9	VDDQ
AB23	NC	AD2	DDR1_DQ21	AE10	DVSS
AB24	DVSS	AD3	DDR1_DQSN2	AE11	DVSS
AB25	AVDD18_ENET	AD4	DDR1_DQSP2	AE12	VDDCPU_B
AB26	DVSS	AD5	DVSS	AE13	VDDCPU_B
AB28	ENET_EXTRES	AD6	VDDQLP	AE14	DVSS
AB29	GPIOZ_0	AD8	VDDQ	AE15	VDDCPU_B
AB30	GPIOZ_1	AD9	DVSS	AE16	VDDCPU_B
AB31	GPIOZ_2	AD10	VDD_DDR	AE17	DVSS
AB32	ENET_RXN	AD11	DVSS	AE18	VDD_EE
AB33	DVSS	AD12	VDDCPU_B	AE19	VDD_EE
AC1	DVSS	AD13	DVSS	AE20	VDD_EE
AC2	DDR1_DQ23	AD14	DVSS	AE21	NC
AC3	DDR1_DQ20	AD15	VDDCPU_B	AE22	NC
AC5	VDDQLP	AD16	DVSS	AE23	NC
AC8	DVSS	AD17	DVSS	AE24	DVSS
AC9	DVSS	AD18	VDD_EE	AE25	AVDD18_AUDIO
AC10	DVSS	AD19	DVSS	AE26	AVSS_AUDIO
AC11	VDD_DDR	AD20	VDD_EE	AE28	AU_REFP
AC12	VDDCPU_B	AD21	NC	AE29	SARADC_CH1
AC13	VDDCPU_B	AD22	DVSS	AE30	AU_LO1L
AC14	DVSS	AD23	NC	AE31	AU_LO1R
AC15	VDDCPU_B	AD24	DVSS	AE32	DVSS
AC16	VDDCPU_B	AD25	DVSS	AE33	GPIOW_14
AC17	DVSS	AD26	AU_VMID	AF1	DVSS
AC18	DVSS	AD28	SARADC_CH6		



BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
AF2	DDR1_DQ17	AG14	DVSS	AH21	VDD_EE
AF3	DDR1_DQM2	AG15	DVSS	AH22	VDD_EE
AF5	DVSS	AG16	DVSS	AH23	VDD_EE
AF8	DVSS	AG17	DVSS	AH24	DVSS
AF9	DVSS	AG18	DVSS	AH25	AVDD18_HDMITX_EARCRX
AF10	DVSS	AG19	VDD_EE	AH26	DVSS
AF11	VDD_DDR	AG20	DVSS	AH28	GPIOW_0
AF12	VDDCPU_B	AG21	VDD_EE	AH29	GPIOW_10
AF13	DVSS	AG22	DVSS	AH30	GPIOW_11
AF14	DVSS	AG23	VDD_EE	AH31	GPIOW_16
AF15	DVSS	AG24	DVSS	AH32	GPIOW_7
AF16	DVSS	AG25	DVSS	AH33	GPIOW_6
AF17	DVSS	AG26	DVSS	AJ1	DVSS
AF18	DVSS	AG28	GPIOW_3	AJ2	DDR1_AC_22
AF19	DVSS	AG29	GPIOW_2	AJ3	DDR1_AC_28
AF20	DVSS	AG30	GPIOW_9	AJ5	DVSS
AF21	DVSS	AG31	GPIOW_8	AJ8	DVSS
AF22	DVSS	AG32	GPIOW_15	AJ9	DVSS
AF23	DVSS	AH1	DDR1_AC_33	AJ10	DVSS
AF24	DVSS	AH2	DVSS	AJ11	VDD_DDR
AF25	DVSS	AH3	DDR1_AC_25	AJ12	DVSS
AF26	DVSS	AH4	DDR1_AC_24	AJ13	AVDD08_DP
AF29	DVSS	AH5	DVSS	AJ14	AVDD18_DP
AF31	GPIOW_13	AH6	DDR1_AC_15	AJ15	AVDD18_DP
AF32	GPIOW_12	AH8	DVSS	AJ16	DVSS
AF33	GPIOW_4	AH9	VDDQ	AJ17	VDDIO_Y
AG2	DDR1_DQ18	AH10	DVSS	AJ18	DVSS
AG3	DDR1_AC_31	AH11	DVSS	AJ19	DVSS
AG4	DDR1_AC_30	AH12	DVSS	AJ20	DVSS
AG5	DDR1_AC_32	AH13	DVSS	AJ21	AVDD33_HDMITX
AG6	DDR1_AC_35	AH14	DVSS	AJ22	DVSS
AG8	VDDQ	AH15	DVSS	AJ23	DVSS
AG9	DVSS	AH16	DVSS	AJ24	DVSS
AG10	VDD_DDR	AH17	DVSS	AJ25	DVSS
AG11	DVSS	AH18	DVSS	AJ26	AVSS_HPLL
AG12	VDDCPU_B	AH19	VDD_EE	AJ29	DVSS
AG13	VDDCPU_B	AH20	VDD_EE		

BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
AJ31	DVSS	AL5	DDR1_AC_1	AM17	GPIYOY_0
AJ32	HDMITX_D2P	AL6	DDR1_RST	AM18	VX1_A_0N
AJ33	HDMITX_REXT	AL8	DVSS	AM20	GPIYOY_4
AK2	DDR1_AC_23	AL9	VDDQ	AM21	GPIOB_3
AK3	DDR1_AC_4	AL10	AVSS_DP	AM23	VDDIO_B
AK4	DDR1_AC_5	AL11	AVDD18_DDR_DP_PLL	AM24	GPIOB_12
AK5	DVSS	AL12	DVSS	AM26	DVSS
AK6	DDR1_PVREF	AL13	DVSS	AM29	DVSS
AK8	VDDQ	AL14	DVSS	AM31	DVSS
AK9	DVSS	AL15	DVSS	AM32	HDMITX_D0P
AK10	VDD_DDR	AL16	VX1_LPP	AM33	DVSS
AK11	DVSS	AL17	VX1_LPN	AN2	DDR1_AC_8
AK12	DVSS	AL18	DVSS	AN3	DDR1_AC_0
AK13	AVDD08_DP	AL19	VX1_ATP	AN4	DDR1_AC_6
AK14	DVSS	AL20	DVSS	AN5	DVSS
AK15	DVSS	AL21	DVSS	AN6	DDR1_DQ2
AK16	DVSS	AL22	DVSS	AN7	DDR1_PZQ
AK17	DVSS	AL23	DVSS	AN8	DDR1_DQ14
AK18	VDD18_AO	AL24	DVSS	AN9	DVSS
AK19	DVSS	AL25	AVDD08_HDMITX	AN10	VX1_B_7P
AK20	DVSS	AL26	DVSS	AN11	VX1_B_7N
AK21	DVSS	AL28	HDMIRX_C_D0N	AN12	VX1_B_4N
AK22	DVSS	AL29	HDMIRX_C_D0P	AN13	DVSS
AK23	AVDD08_HDMIRX	AL30	HDMIRX_C_D1N	AN14	GPIYOY_2
AK24	AVDD33_HDMIRX	AL31	HDMIRX_C_D1P	AN15	VX1_A_5P
AK25	AVDD18_HDMIRX_EARCTX	AL32	HDMITX_D1P	AN16	DVSS
AK26	DVSS	AL33	HDMITX_D1N	AN17	DVSS
AK28	HDMITX_ARCRXP	AM1	DDR1_AC_21	AN18	VX1_A_0P
AK29	HDMITX_ARCRXN	AM2	DVSS	AN19	DVSS
AK30	HDMIRX_C_D2N	AM3	DDR1_AC_13	AN20	GPIYOY_3
AK31	HDMIRX_C_D2P	AM5	DVSS	AN21	GPIOB_2
AK32	HDMITX_D2N	AM8	DVSS	AN22	DVSS
AL1	DDR1_AC_20	AM9	DVSS	AN23	GPIOB_10
AL2	DVSS	AM11	DVSS	AN24	GPIOB_11
AL3	DDR1_AC_9	AM12	VX1_B_4P	AN25	DVSS
AL4	DDR1_AC_12	AM14	GPIYOY_1	AN26	DVSS
		AM15	VX1_A_5N		

BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
AN27	GPIOW_1	AR6	DDR1_DQM0	AT10	DDR1_DQ11
AN28	GPIOW_5	AR7	DDR1_DQ12	AT11	DDR1_DQ15
AN29	DVSS	AR8	DDR1_DQ9	AT12	DVSS
AN30	HDMIRX_C_CLKN	AR9	DDR1_DQ10	AT13	VX1_B_3N
AN31	HDMIRX_C_CLKP	AR10	DVSS	AT14	VX1_B_2P
AN32	HDMITX_D0N	AR11	VX1_B_6P	AT15	VX1_B_2N
AP1	DDR1_AC_2	AR12	VX1_B_5P	AT16	VX1_A_6P
AP2	DVSS	AR13	DVSS	AT17	VX1_A_6N
AP3	DDR1_AC_3	AR14	VX1_B_1N	AT18	VX1_A_4P
AP4	DDR1_DQ1	AR15	VX1_A_7N	AT19	VX1_A_2P
AP6	DDR1_DQ3	AR16	DVSS	AT20	DVSS
AP7	DDR1_DQM1	AR17	VX1_A_3N	AT21	GPIOW_5
AP8	DDR1_DQ13	AR18	VX1_A_1N	AT22	GPIOW_7
AP10	DVSS	AR19	DVSS	AT23	GPIOW_11
AP11	VX1_B_6N	AR20	GPIOW_8	AT24	GPIOW_12
AP12	VX1_B_5N	AR21	GPIOW_9	AT25	GPIOW_13
AP14	VX1_B_1P	AR22	DVSS	AT26	GPIOW_14
AP15	VX1_A_7P	AR23	GPIOB_4	AT27	GPIOW_18
AP17	VX1_A_3P	AR24	GPIOB_6	AT28	DVSS
AP18	VX1_A_1P	AR25	DVSS	AT29	HDMIRX_A_D0N
AP20	GPIOB_1	AR26	GPIOB_9	AT30	HDMIRX_A_D0P
AP21	GPIOB_0	AR27	HDMIRX_ARCTXP	AT31	DVSS
AP23	GPIOB_5	AR28	HDMIRX_B_CLKN	AT32	HDMIRX_B_D2N
AP24	GPIOB_7	AR30	HDMIRX_B_D1N	AT33	HDMIRX_A_D2N
AP26	GPIOB_8	AR31	HDMIRX_B_D1P	AU1	DVSS
AP27	HDMIRX_ARCTXN	AR32	HDMIRX_B_D2P	AU2	DDR1_DQ0
AP28	HDMIRX_B_CLKP	AR33	HDMIRX_A_D2P	AU3	DDR1_DQ5
AP29	HDMIRX_B_D0N	AT1	DDR1_AC_11	AU5	DVSS
AP30	HDMIRX_B_D0P	AT2	DDR1_AC_14	AU6	DDR1_DQSP0
AP31	DVSS	AT3	DVSS	AU7	DVSS
AP32	HDMITX_CLKP	AT4	DDR1_DQ6	AU9	DDR1_DQ8
AP33	HDMITX_CLKN	AT5	DDR1_DQ4	AU10	DVSS
AR1	DDR1_AC_10	AT6	DDR1_DQSN0	AU12	DIF_REXT
AR2	DDR1_AC_7	AT7	DDR1_DQSP1	AU13	VX1_B_3P
AR3	DDR1_AC_26	AT8	DDR1_DQSN1	AU15	VX1_B_0P
AR4	DDR1_DQ7	AT9	DVSS	AU16	VX1_B_0N

BALL #	NET NAME	BALL #	NET NAME	BALL #	NET NAME
AU18	VX1_A_4N	AU25	GPIOY_15	AU32	HDMIRX_A_D1P
AU19	VX1_A_2N	AU27	GPIOY_17	AU33	DVSS
AU21	GPIOY_6	AU28	HDMIRX_A_CLKN		
AU22	GPIOY_10	AU29	HDMIRX_A_CLKP		
AU24	GPIOY_16	AU31	HDMIRX_A_D1N		

## 4.3 Pin Description

The pin assignments are described in the following table.

**Table 4-1 Pin Assignments**

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
<b>Audio</b>					
AU_AIL1	AI	-	Audio line in, left channel of port 1	AVDD18_AUDIO	NC
AU_AIL2	AI	-	Audio line in, left channel of port 2	AVDD18_AUDIO	NC
AU_AIR1	AI	-	Audio line in, right channel of port 1	AVDD18_AUDIO	NC
AU_AIR2	AI	-	Audio line in, right channel of port 2	AVDD18_AUDIO	NC
AU_LO1L	AO	-	Audio line out, left channel of port 1	AVDD18_AUDIO	NC
AU_LO1R	AO	-	Audio line out, right channel of port 1	AVDD18_AUDIO	NC
AU_REFP	A	-	Audio ADC/DAC positive reference voltage	AVDD18_AUDIO	NC
AU_VMID	A	-	MID voltage of audio ADC/DAC	AVDD18_AUDIO	NC
AVSS_AUDIO	AP	-	Analog ground for Audio ADC/DAC	-	To VSS
AVDD18_AUDIO	AP	-	Analog 1.8V for Audio ADC/DAC	-	To 1.8V
<b>MIPI CSI</b>					
CSI_A_CLKN	AI	-	MIPI CSI CLK negative input for channel A	AVDD18_CSI	NC
CSI_A_CLKP	AI	-	MIPI CSI CLK positive input for channel A	AVDD18_CSI	NC
CSI_A_D0N	AIO	-	MIPI CSI channel A data 0 negative input	AVDD18_CSI	NC
CSI_A_D0P	AIO	-	MIPI CSI channel A data 0 positive input	AVDD18_CSI	NC
CSI_A_D1N	AIO	-	MIPI CSI channel A data 1 negative input	AVDD18_CSI	NC
CSI_A_D1P	AIO	-	MIPI CSI channel A data 1 positive input	AVDD18_CSI	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
CSI_A_D2N	AIO	-	MIPI CSI channel A data 2 or channel B data0 negative input	AVDD18_CSI	NC
CSI_A_D2P	AIO	-	MIPI CSI channel A data 2 or channel B data0 positive input	AVDD18_CSI	NC
CSI_A_D3N	AIO	-	MIPI CSI channel A data 3 or channel B data1 negative input	AVDD18_CSI	NC
CSI_A_D3P	AIO	-	MIPI CSI channel A data 3 or channel B data1 positive input	AVDD18_CSI	NC
CSI_B_CLKN	AI	-	MIPI CSI CLK negative input for channel B	AVDD18_CSI	NC
CSI_B_CLKP	AI	-	MIPI CSI CLK positive input for channel B	AVDD18_CSI	NC
CSI_C_CLKN	AI	-	MIPI CSI CLK negative input for channel C	AVDD18_CSI	NC
CSI_C_CLKP	AI	-	MIPI CSI CLK positive input for channel C	AVDD18_CSI	NC
CSI_C_D0N	AIO	-	MIPI CSI channel C data 0 negative input	AVDD18_CSI	NC
CSI_C_D0P	AIO	-	MIPI CSI channel C data 0 positive input	AVDD18_CSI	NC
CSI_C_D1N	AIO	-	MIPI CSI channel C data 1 negative input	AVDD18_CSI	NC
CSI_C_D1P	AIO	-	MIPI CSI channel C data 1 positive input	AVDD18_CSI	NC
CSI_C_D2N	AIO	-	MIPI CSI channel C data 2 or channel D data0 negative input	AVDD18_CSI	NC
CSI_C_D2P	AIO	-	MIPI CSI channel C data 2 or channel D data0 positive input	AVDD18_CSI	NC
CSI_C_D3N	AIO	-	MIPI CSI channel C data 3 or channel D data1 negative input	AVDD18_CSI	NC
CSI_C_D3P	AIO	-	MIPI CSI channel C data 3 or channel D data1 positive input	AVDD18_CSI	NC
CSI_D_CLKN	AI	-	MIPI CSI CLK negative input for channel D	AVDD18_CSI	NC
CSI_D_CLKP	AI	-	MIPI CSI CLK positive input for channel D	AVDD18_CSI	NC
AVDD0V8_CSI	AP	-	Analog Power supply 0.8V for MIPI CSI	-	To 0.8V
AVDD18_CSI	AP	-	Analog 1.8V power supply for MIPI CSI	-	To 1.8V
<b>DDR0</b>					
DDR0_AC_0	DO	-	DDR PHY0 address/command/control signal bit 0	VDDQ	NC
DDR0_AC_1	DO	-	DDR PHY0 address/command/control signal bit 1	VDDQ	NC
DDR0_AC_2	DO	-	DDR PHY0 address/command/control signal bit 2	VDDQ	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
DDR0_AC_3	DO	-	DDR PHY0 address/command/control signal bit 3	VDDQ	NC
DDR0_AC_4	DO	-	DDR PHY0 address/command/control signal bit 4	VDDQ	NC
DDR0_AC_5	DO	-	DDR PHY0 address/command/control signal bit 5	VDDQ	NC
DDR0_AC_6	DO	-	DDR PHY0 address/command/control signal bit 6	VDDQ	NC
DDR0_AC_7	DO	-	DDR PHY0 address/command/control signal bit 7	VDDQ	NC
DDR0_AC_8	DO	-	DDR PHY0 address/command/control signal bit 8	VDDQ	NC
DDR0_AC_9	DO	-	DDR PHY0 address/command/control signal bit 9	VDDQ	NC
DDR0_AC_10	DO	-	DDR PHY0 address/command/control signal bit 10	VDDQ	NC
DDR0_AC_11	DO	-	DDR PHY0 address/command/control signal bit 11	VDDQ	NC
DDR0_AC_12	DO	-	DDR PHY0 address/command/control signal bit 12	VDDQ	NC
DDR0_AC_13	DO	-	DDR PHY0 address/command/control signal bit 13	VDDQ	NC
DDR0_AC_14	DO	-	DDR PHY0 address/command/control signal bit 14	VDDQ	NC
DDR0_AC_15	DO	-	DDR PHY0 address/command/control signal bit 15	VDDQ	NC
DDR0_AC_20	DO	-	DDR PHY0 address/command/control signal bit 20	VDDQ	NC
DDR0_AC_21	DO	-	DDR PHY0 address/command/control signal bit 21	VDDQ	NC
DDR0_AC_22	DO	-	DDR PHY0 address/command/control signal bit 22	VDDQ	NC
DDR0_AC_23	DO	-	DDR PHY0 address/command/control signal bit 23	VDDQ	NC
DDR0_AC_24	DO	-	DDR PHY0 address/command/control signal bit 24	VDDQ	NC
DDR0_AC_25	DO	-	DDR PHY0 address/command/control signal bit 25	VDDQ	NC
DDR0_AC_26	DO	-	DDR PHY0 address/command/control signal bit 26	VDDQ	NC
DDR0_AC_28	DO	-	DDR PHY0 address/command/control signal bit 28	VDDQ	NC
DDR0_AC_29	DO	-	DDR PHY0 address/command/control signal bit 29	VDDQ	NC
DDR0_AC_30	DO	-	DDR PHY0 address/command/control signal bit 30	VDDQ	NC
DDR0_AC_31	DO	-	DDR PHY0 address/command/control signal bit 31	VDDQ	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
DDR0_AC_32	DO	-	DDR PHY0 address/command/control signal bit 32	VDDQ	NC
DDR0_AC_33	DO	-	DDR PHY0 address/command/control signal bit 33	VDDQ	NC
DDR0_AC_34	DO	-	DDR PHY0 address/command/control signal bit 34	VDDQ	NC
DDR0_AC_35	DO	-	DDR PHY0 address/command/control signal bit 35	VDDQ	NC
DDR0_AC_36	DO	-	DDR PHY0 address/command/control signal bit 36	VDDQ	NC
DDR0_AC_37	DO	-	DDR PHY0 address/command/control signal bit 37	VDDQ	NC
DDR0_AC_38	DO	-	DDR PHY0 address/command/control signal bit 38	VDDQ	NC
DDR0_DQ0	DIO	-	DRAM0 data bus bit 0	VDDQ	To DRAM0
DDR0_DQ1	DIO	-	DRAM0 data bus bit 1	VDDQ	To DRAM0
DDR0_DQ2	DIO	-	DRAM0 data bus bit 2	VDDQ	To DRAM0
DDR0_DQ3	DIO	-	DRAM0 data bus bit 3	VDDQ	To DRAM0
DDR0_DQ4	DIO	-	DRAM0 data bus bit 4	VDDQ	To DRAM0
DDR0_DQ5	DIO	-	DRAM0 data bus bit 5	VDDQ	To DRAM0
DDR0_DQ6	DIO	-	DRAM0 data bus bit 6	VDDQ	To DRAM0
DDR0_DQ7	DIO	-	DRAM0 data bus bit 7	VDDQ	To DRAM0
DDR0_DQ8	DIO	-	DRAM0 data bus bit 8	VDDQ	To DRAM0
DDR0_DQ9	DIO	-	DRAM0 data bus bit 9	VDDQ	To DRAM0
DDR0_DQ10	DIO	-	DRAM0 data bus bit 10	VDDQ	To DRAM0
DDR0_DQ11	DIO	-	DRAM0 data bus bit 11	VDDQ	To DRAM0
DDR0_DQ12	DIO	-	DRAM0 data bus bit 12	VDDQ	To DRAM0
DDR0_DQ13	DIO	-	DRAM0 data bus bit 13	VDDQ	To DRAM0
DDR0_DQ14	DIO	-	DRAM0 data bus bit 14	VDDQ	To DRAM0
DDR0_DQ15	DIO	-	DRAM0 data bus bit 15	VDDQ	To DRAM0
DDR0_DQ16	DIO	-	DRAM0 data bus bit 16	VDDQ	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
DDR0_DQ17	DIO	-	DRAM0 data bus bit 17	VDDQ	NC
DDR0_DQ18	DIO	-	DRAM0 data bus bit 18	VDDQ	NC
DDR0_DQ19	DIO	-	DRAM0 data bus bit 19	VDDQ	NC
DDR0_DQ20	DIO	-	DRAM0 data bus bit 20	VDDQ	NC
DDR0_DQ21	DIO	-	DRAM0 data bus bit 21	VDDQ	NC
DDR0_DQ22	DIO	-	DRAM0 data bus bit 22	VDDQ	NC
DDR0_DQ23	DIO	-	DRAM0 data bus bit 23	VDDQ	NC
DDR0_DQ24	DIO	-	DRAM0 data bus bit 24	VDDQ	NC
DDR0_DQ25	DIO	-	DRAM0 data bus bit 25	VDDQ	NC
DDR0_DQ26	DIO	-	DRAM0 data bus bit 26	VDDQ	NC
DDR0_DQ27	DIO	-	DRAM0 data bus bit 27	VDDQ	NC
DDR0_DQ28	DIO	-	DRAM0 data bus bit 28	VDDQ	NC
DDR0_DQ29	DIO	-	DRAM0 data bus bit 29	VDDQ	NC
DDR0_DQ30	DIO	-	DRAM0 data bus bit 30	VDDQ	NC
DDR0_DQ31	DIO	-	DRAM0 data bus bit 31	VDDQ	NC
DDR0_DQM0	DIO	-	DRAM0 data mask 0	VDDQ	To DRAM0
DDR0_DQM1	DIO	-	DRAM0 data mask 1	VDDQ	To DRAM0
DDR0_DQM2	DIO	-	DRAM0 data mask 2	VDDQ	NC
DDR0_DQM3	DIO	-	DRAM0 data mask 3	VDDQ	NC
DDR0_DQSN0	DIO	-	DRAM0 data strobe 0 complementary	VDDQ	To DRAM0
DDR0_DQSN1	DIO	-	DRAM0 data strobe 1 complementary	VDDQ	To DRAM0
DDR0_DQSN2	DIO	-	DRAM0 data strobe 2 complementary	VDDQ	NC
DDR0_DQSN3	DIO	-	DRAM0 data strobe 3 complementary	VDDQ	NC
DDR0_DQSP0	DIO	-	DRAM0 data strobe 0	VDDQ	To DRAM0



Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
DDR0_DQSP1	DIO	-	DRAM0 data strobe 1	VDDQ	To DRAM0
DDR0_DQSP2	DIO	-	DRAM0 data strobe 2	VDDQ	NC
DDR0_DQSP3	DIO	-	DRAM0 data strobe 3	VDDQ	NC
DDR0_PVREF			DRAM0 reference voltage	VDDQ	To GND by capacitor
DDR0_PZQ	A	-	DRAM0 reference pin for ZQ calibration	VDDQ	To GND by 240ohm
DDR0_RST	DO	-	DRAM0 DDR4/LPDDR4/LPDDR4X RSTn	VDDQ	NC
AVDD_DDR0PLL	P	-	Analog power supply for DDR0PLL	-	To DDR VDDQ
<b>DDR1</b>					
DDR1_AC_0	DO	-	DDR PHY1 address/command/control signal bit 0	VDDQ	NC
DDR1_AC_1	DO	-	DDR PHY1 address/command/control signal bit 1	VDDQ	NC
DDR1_AC_2	DO	-	DDR PHY1 address/command/control signal bit 2	VDDQ	NC
DDR1_AC_3	DO	-	DDR PHY1 address/command/control signal bit 3	VDDQ	NC
DDR1_AC_4	DO	-	DDR PHY1 address/command/control signal bit 4	VDDQ	NC
DDR1_AC_5	DO	-	DDR PHY1 address/command/control signal bit 5	VDDQ	NC
DDR1_AC_6	DO	-	DDR PHY1 address/command/control signal bit 6	VDDQ	NC
DDR1_AC_7	DO	-	DDR PHY1 address/command/control signal bit 7	VDDQ	NC
DDR1_AC_8	DO	-	DDR PHY1 address/command/control signal bit 8	VDDQ	NC
DDR1_AC_9	DO	-	DDR PHY1 address/command/control signal bit 9	VDDQ	NC
DDR1_AC_10	DO	-	DDR PHY1 address/command/control signal bit 10	VDDQ	NC
DDR1_AC_11	DO	-	DDR PHY1 address/command/control signal bit 11	VDDQ	NC
DDR1_AC_12	DO	-	DDR PHY1 address/command/control signal bit 12	VDDQ	NC
DDR1_AC_13	DO	-	DDR PHY1 address/command/control signal bit 13	VDDQ	NC
DDR1_AC_14	DO	-	DDR PHY1 address/command/control signal bit 14	VDDQ	NC
DDR1_AC_15	DO	-	DDR PHY1 address/command/control signal bit 15	VDDQ	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
DDR1_AC_20	DO	-	DDR PHY1 address/command/control signal bit 20	VDDQ	NC
DDR1_AC_21	DO	-	DDR PHY1 address/command/control signal bit 21	VDDQ	NC
DDR1_AC_22	DO	-	DDR PHY1 address/command/control signal bit 22	VDDQ	NC
DDR1_AC_23	DO	-	DDR PHY1 address/command/control signal bit 23	VDDQ	NC
DDR1_AC_24	DO	-	DDR PHY1 address/command/control signal bit 24	VDDQ	NC
DDR1_AC_25	DO	-	DDR PHY1 address/command/control signal bit 25	VDDQ	NC
DDR1_AC_26	DO	-	DDR PHY1 address/command/control signal bit 26	VDDQ	NC
DDR1_AC_28	DO	-	DDR PHY1 address/command/control signal bit 28	VDDQ	NC
DDR1_AC_29	DO	-	DDR PHY1 address/command/control signal bit 29	VDDQ	NC
DDR1_AC_30	DO	-	DDR PHY1 address/command/control signal bit 30	VDDQ	NC
DDR1_AC_31	DO	-	DDR PHY1 address/command/control signal bit 31	VDDQ	NC
DDR1_AC_32	DO	-	DDR PHY1 address/command/control signal bit 32	VDDQ	NC
DDR1_AC_33	DO	-	DDR PHY1 address/command/control signal bit 33	VDDQ	NC
DDR1_AC_34	DO	-	DDR PHY1 address/command/control signal bit 34	VDDQ	NC
DDR1_AC_35	DO	-	DDR PHY1 address/command/control signal bit 35	VDDQ	NC
DDR1_AC_36	DO	-	DDR PHY1 address/command/control signal bit 36	VDDQ	NC
DDR1_AC_37	DO	-	DDR PHY1 address/command/control signal bit 37	VDDQ	NC
DDR1_AC_38	DO	-	DDR PHY1 address/command/control signal bit 38	VDDQ	NC
DDR1_DQ0	DIO	-	DRAM1 data bus bit 0	VDDQ	To DRAM1
DDR1_DQ1	DIO	-	DRAM1 data bus bit 1	VDDQ	To DRAM1
DDR1_DQ2	DIO	-	DRAM1 data bus bit 2	VDDQ	To DRAM1
DDR1_DQ3	DIO	-	DRAM1 data bus bit 3	VDDQ	To DRAM1
DDR1_DQ4	DIO	-	DRAM1 data bus bit 4	VDDQ	To DRAM1
DDR1_DQ5	DIO	-	DRAM1 data bus bit 5	VDDQ	To DRAM1

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
DDR1_DQ6	DIO	-	DRAM1 data bus bit 6	VDDQ	To DRAM1
DDR1_DQ7	DIO	-	DRAM1 data bus bit 7	VDDQ	To DRAM1
DDR1_DQ8	DIO	-	DRAM1 data bus bit 8	VDDQ	To DRAM1
DDR1_DQ9	DIO	-	DRAM1 data bus bit 9	VDDQ	To DRAM1
DDR1_DQ10	DIO	-	DRAM1 data bus bit 10	VDDQ	To DRAM1
DDR1_DQ11	DIO	-	DRAM1 data bus bit 11	VDDQ	To DRAM1
DDR1_DQ12	DIO	-	DRAM1 data bus bit 12	VDDQ	To DRAM1
DDR1_DQ13	DIO	-	DRAM1 data bus bit 13	VDDQ	To DRAM1
DDR1_DQ14	DIO	-	DRAM1 data bus bit 14	VDDQ	To DRAM1
DDR1_DQ15	DIO	-	DRAM1 data bus bit 15	VDDQ	To DRAM1
DDR1_DQ16	DIO	-	DRAM1 data bus bit 16	VDDQ	NC
DDR1_DQ17	DIO	-	DRAM1 data bus bit 17	VDDQ	NC
DDR1_DQ18	DIO	-	DRAM1 data bus bit 18	VDDQ	NC
DDR1_DQ19	DIO	-	DRAM1 data bus bit 19	VDDQ	NC
DDR1_DQ20	DIO	-	DRAM1 data bus bit 20	VDDQ	NC
DDR1_DQ21	DIO	-	DRAM1 data bus bit 21	VDDQ	NC
DDR1_DQ22	DIO	-	DRAM1 data bus bit 22	VDDQ	NC
DDR1_DQ23	DIO	-	DRAM1 data bus bit 23	VDDQ	NC
DDR1_DQ24	DIO	-	DRAM1 data bus bit 24	VDDQ	NC
DDR1_DQ25	DIO	-	DRAM1 data bus bit 25	VDDQ	NC
DDR1_DQ26	DIO	-	DRAM1 data bus bit 26	VDDQ	NC
DDR1_DQ27	DIO	-	DRAM1 data bus bit 27	VDDQ	NC
DDR1_DQ28	DIO	-	DRAM1 data bus bit 28	VDDQ	NC
DDR1_DQ29	DIO	-	DRAM1 data bus bit 29	VDDQ	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
DDR1_DQ30	DIO	-	DRAM1 data bus bit 30	VDDQ	NC
DDR1_DQ31	DIO	-	DRAM1 data bus bit 31	VDDQ	NC
DDR1_DQM0	DIO	-	DRAM1 data mask 0	VDDQ	To DRAM1
DDR1_DQM1	DIO	-	DRAM1 data mask 1	VDDQ	To DRAM1
DDR1_DQM2	DIO	-	DRAM1 data mask 2	VDDQ	NC
DDR1_DQM3	DIO	-	DRAM1 data mask 3	VDDQ	NC
DDR1_DQSN0	DIO	-	DRAM1 data strobe 0 complementary	VDDQ	To DRAM1
DDR1_DQSN1	DIO	-	DRAM1 data strobe 1 complementary	VDDQ	To DRAM1
DDR1_DQSN2	DIO	-	DRAM1 data strobe 2 complementary	VDDQ	NC
DDR1_DQSN3	DIO	-	DRAM1 data strobe 3 complementary	VDDQ	NC
DDR1_DQSP0	DIO	-	DRAM1 data strobe 0	VDDQ	To DRAM1
DDR1_DQSP1	DIO	-	DRAM1 data strobe 1	VDDQ	To DRAM1
DDR1_DQSP2	DIO	-	DRAM1 data strobe 2	VDDQ	NC
DDR1_DQSP3	DIO	-	DRAM1 data strobe 3	VDDQ	NC
DDR1_PVREF	0	0	DRAM1 reference voltage	VDDQ	To GND by capacitor
DDR1_PZQ	A	-	DRAM1 reference pin for ZQ calibration	VDDQ	To GND by 240ohm
DDR1_RST	DO	-	DRAM1 DDR4/LPDDR4/LPDDR4X RSTn	VDDQ	NC
AVDD_DDR1PLL	P	-	Analog power supply for DDR1PLL	-	To DDR VDDQ
<b>Ethernet</b>					
ENET_ATP	AIO	-	Ethernet PHY analog test pin	AVDD18_ENET	NC
ENET_EXTRES	A	-	Ethernet PHY external resistor connection	AVDD18_ENET	NC
ENET_RXN	AIO	-	Ethernet PHY receive data negative input	AVDD18_ENET	NC
ENET_RXP	AIO	-	Ethernet PHY receive data positive input	AVDD18_ENET	NC
ENET_TXN	AIO	-	Ethernet PHY transmit data negative output	AVDD18_ENET	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
ENET_TXP	AIO	-	Ethernet PHY transmit data positive output	AVDD18_ENET	NC
AVDD18_ENET	AP	-	Analog 1.8V power supply for Ethernet module	-	To 1.8V
<b>GPIOB</b> - refer to <a href="#">Table 4-6</a> for functional multiplex information.					
GPIOB_0	DIO	Up	General purpose input/output bank B signal 0	VDDIO_B	NC
GPIOB_1	DIO	Up	General purpose input/output bank B signal 1	VDDIO_B	NC
GPIOB_2	DIO	Up	General purpose input/output bank B signal 2	VDDIO_B	NC
GPIOB_3	DIO	Up	General purpose input/output bank B signal 3	VDDIO_B	NC
GPIOB_4	DIO	Up	General purpose input/output bank B signal 4	VDDIO_B	NC
GPIOB_5	DIO	Up	General purpose input/output bank B signal 5	VDDIO_B	NC
GPIOB_6	DIO	Up	General purpose input/output bank B signal 6	VDDIO_B	NC
GPIOB_7	DIO	Up	General purpose input/output bank B signal 7	VDDIO_B	NC
GPIOB_8	DIO	Up	General purpose input/output bank B signal 8	VDDIO_B	NC
GPIOB_9	DIO	Down	General purpose input/output bank B signal 9	VDDIO_B	NC
GPIOB_10	DIO	Up	General purpose input/output bank B signal 10	VDDIO_B	NC
GPIOB_11	DIO	Down	General purpose input/output bank B signal 11	VDDIO_B	NC
GPIOB_12	DIO	Up	General purpose input/output bank B signal 12	VDDIO_B	NC
VDDIO_B	P	-	Power supply for GPIO bank B	-	To 1.8V or 3.3V
<b>GPIOC</b> - refer to <a href="#">Table 4-6</a> for functional multiplex information.					
GPIOC_0	DIO	Up	General purpose input/output bank C signal 0	VDDIO_C	NC
GPIOC_1	DIO	Up	General purpose input/output bank C signal 1	VDDIO_C	NC
GPIOC_2	DIO	Up	General purpose input/output bank C signal 2	VDDIO_C	NC
GPIOC_3	DIO	Up	General purpose input/output bank C signal 3	VDDIO_C	NC
GPIOC_4	DIO	Up	General purpose input/output bank C signal 4	VDDIO_C	NC
GPIOC_5	DIO	Up	General purpose input/output bank C signal 5	VDDIO_C	NC
GPIOC_6	DIO	Up	General purpose input/output bank C signal 6	VDDIO_C	NC
VDDIO_C	P	-	Power supply for GPIO bank C	-	To 1.8V or 3.3V
<b>GPIOX</b> - refer to <a href="#">Table 4-11</a> for functional multiplex information.					
GPIOX_0	DIO	Up	General purpose input/output bank X signal 0	VDDIO_X	NC
GPIOX_1	DIO	Up	General purpose input/output bank X signal 1	VDDIO_X	NC
GPIOX_2	DIO	Up	General purpose input/output bank X signal 2	VDDIO_X	NC
GPIOX_3	DIO	Up	General purpose input/output bank X signal 3	VDDIO_X	NC
GPIOX_4	DIO	Up	General purpose input/output bank X signal 4	VDDIO_X	NC
GPIOX_5	DIO	Up	General purpose input/output bank X signal 5	VDDIO_X	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
GPIOX_6	DIO	Down	General purpose input/output bank X signal 6	VDDIO_X	NC
GPIOX_7	DIO	Up	General purpose input/output bank X signal 7	VDDIO_X	NC
GPIOX_8	DIO	Up	General purpose input/output bank X signal 8	VDDIO_X	NC
GPIOX_9	DIO	Up	General purpose input/output bank X signal 9	VDDIO_X	NC
GPIOX_10	DIO	Up	General purpose input/output bank X signal 10	VDDIO_X	NC
GPIOX_11	DIO	Up	General purpose input/output bank X signal 11	VDDIO_X	NC
GPIOX_12	DIO	Up	General purpose input/output bank X signal 12	VDDIO_X	NC
GPIOX_13	DIO	Up	General purpose input/output bank X signal 13	VDDIO_X	NC
GPIOX_14	DIO	Up	General purpose input/output bank X signal 14	VDDIO_X	NC
GPIOX_15	DIO	Up	General purpose input/output bank X signal 15	VDDIO_X	NC
GPIOX_16	DIO	Up	General purpose input/output bank X signal 16	VDDIO_X	NC
GPIOX_17	DIO	Down	General purpose input/output bank X signal 17	VDDIO_X	NC
GPIOX_18	DIO	Up	General purpose input/output bank X signal 18	VDDIO_X	NC
GPIOX_19	DIO	HiZ	General purpose input/output bank X signal 19	VDDIO_X	NC
VDDIO_X	P	-	Power supply for GPIO bank X	-	To 1.8V or 3.3V
<b>GPIOW-</b> refer to <a href="#">Table 4-2</a> for functional multiplex information.					
GPIOW_0	OD_5V	HiZ	General purpose input/output bank W signal 0	-	NC
GPIOW_1	OD_5V (Input only)	HiZ	General purpose input bank W signal 1	-	NC
GPIOW_2	OD_5V	HiZ	General purpose input/output bank W signal 2	-	NC
GPIOW_3	OD_5V	HiZ	General purpose input/output bank W signal 3	-	NC
GPIOW_4	OD_5V	HiZ	General purpose input/output bank W signal 4	-	NC
GPIOW_5	OD_5V (Input only)	HiZ	General purpose input bank W signal 5	-	NC
GPIOW_6	OD_5V	HiZ	General purpose input/output bank W signal 6	-	NC
GPIOW_7	OD_5V	HiZ	General purpose input/output bank W signal 7	-	NC
GPIOW_8	OD_5V	HiZ	General purpose input/output bank W signal 8	-	NC
GPIOW_9	OD_5V (Input only)	HiZ	General purpose input bank W signal 9	-	NC
GPIOW_10	OD_5V	HiZ	General purpose input/output bank W signal 10	-	NC
GPIOW_11	OD_5V	HiZ	General purpose input/output bank W signal 11	-	NC
GPIOW_12	OD_5V	HiZ	General purpose input/output bank W signal 12	-	NC
GPIOW_13	OD_5V	HiZ	General purpose input/output bank W signal 13	-	NC
GPIOW_14	OD_5V	HiZ	General purpose input/output bank W signal 14	-	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
GPIOW_15	OD_5V	HiZ	General purpose input/output bank W signal 15	-	NC
GPIOW_16	OD_5V	HiZ	General purpose input/output bank W signal 16	-	NC
<b>GPIOD-</b> refer to <a href="#">Table 4-11</a> for functional multiplex information.					
GPIOD_0	DIO	Up	General purpose input/output bank D signal 0	VDDIO_D	NC
GPIOD_1	DIO	Up	General purpose input/output bank D signal 1	VDDIO_D	NC
GPIOD_2	DIO	Up	General purpose input/output bank D signal 2	VDDIO_D	NC
GPIOD_3	DIO	Up	General purpose input/output bank D signal 3	VDDIO_D	NC
GPIOD_4	DIO	Down	General purpose input/output bank D signal 4	VDDIO_D	NC
GPIOD_5	DIO	Up	General purpose input/output bank D signal 5	VDDIO_D	NC
GPIOD_6	DIO	Down	General purpose input/output bank D signal 6	VDDIO_D	NC
GPIOD_7	DIO	Down	General purpose input/output bank D signal 7	VDDIO_D	NC
GPIOD_8	DIO	Down	General purpose input/output bank D signal 8	VDDIO_D	NC
GPIOD_9	DIO	Down	General purpose input/output bank D signal 9	VDDIO_D	NC
GPIOD_10	DIO	Up	General purpose input/output bank D signal 10	VDDIO_D	NC
GPIOD_11	DIO	Down	General purpose input/output bank D signal 11	VDDIO_D	NC
GPIOD_12	DIO	HiZ	General purpose input/output bank D signal 12	VDDIO_D	NC
RESET_N	DIO	HiZ	System reset input		To RE-SET_N
TEST_N	DIO	Up	SOC test pin should be pulled up during normal power-on.		NC
VDDIO_D	P	-	Power supply for GPIO bank D	-	To 1.8V or 3.3V
<b>GPIOE-</b> refer to <a href="#">Table 4-11</a> for functional multiplex information.					
GPIOE_0	DIO	HiZ	General purpose input/output bank E signal 0	VDDIO_E	NC
GPIOE_1	DIO	HiZ	General purpose input/output bank E signal 1	VDDIO_E	NC
GPIOE_2	DIO	Down	General purpose input/output bank E signal 2	VDDIO_E	NC
GPIOE_3	DIO	HiZ	General purpose input/output bank E signal 3	VDDIO_E	NC
GPIOE_4	DIO	HiZ	General purpose input/output bank E signal 4	VDDIO_E	NC
GPIOE_5	DIO	HiZ	General purpose input/output bank E signal 5	VDDIO_E	NC
GPIOE_6	DIO	HiZ	General purpose input/output bank E signal 6	VDDIO_E	NC
VDDIO_E	P	-	Power supply for GPIO bank E	-	To 1.8V or 3.3V
<b>GPIOZ-</b> refer to <a href="#">Table 4-7</a> for functional multiplex information.					
GPIOZ_0	DIO	Up	General purpose input/output bank Z signal 0	VDDIO_Z	NC
GPIOZ_1	DIO	Up	General purpose input/output bank Z signal 1	VDDIO_Z	NC
GPIOZ_2	DIO	Up	General purpose input/output bank Z signal 2	VDDIO_Z	NC
GPIOZ_3	DIO	Up	General purpose input/output bank Z signal 3	VDDIO_Z	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
GPIOZ_4	DIO	Up	General purpose input/output bank Z signal 4	VDDIO_Z	NC
GPIOZ_5	DIO	Up	General purpose input/output bank Z signal 5	VDDIO_Z	NC
GPIOZ_6	DIO	Up	General purpose input/output bank Z signal 6	VDDIO_Z	NC
GPIOZ_7	DIO	Up	General purpose input/output bank Z signal 7	VDDIO_Z	NC
GPIOZ_8	DIO	Up	General purpose input/output bank Z signal 8	VDDIO_Z	NC
GPIOZ_9	DIO	Down	General purpose input/output bank Z signal 9	VDDIO_Z	NC
GPIOZ_10	DIO	Down	General purpose input/output bank Z signal 10	VDDIO_Z	NC
GPIOZ_11	DIO	Down	General purpose input/output bank Z signal 11	VDDIO_Z	NC
GPIOZ_12	DIO	Down	General purpose input/output bank Z signal 12	VDDIO_Z	NC
GPIOZ_13	DIO	Down	General purpose input/output bank Z signal 13	VDDIO_Z	NC
VDDIO_Z	P	-	Power supply for GPIO bank Z	-	To 1.8V or 3.3V
<b>GPLOT-</b> refer to <a href="#">Table 4-11</a> for functional multiplex information.					
GPLOT_0	DIO	Down	General purpose input/output bank T signal 0	VDDIO_T	NC
GPLOT_1	DIO	Down	General purpose input/output bank T signal 1	VDDIO_T	NC
GPLOT_2	DIO	Down	General purpose input/output bank T signal 2	VDDIO_T	NC
GPLOT_3	DIO	Down	General purpose input/output bank T signal 3	VDDIO_T	NC
GPLOT_4	DIO	Down	General purpose input/output bank T signal 4	VDDIO_T	NC
GPLOT_5	DIO	Down	General purpose input/output bank T signal 5	VDDIO_T	NC
GPLOT_6	DIO	Down	General purpose input/output bank T signal 6	VDDIO_T	NC
GPLOT_7	DIO	Down	General purpose input/output bank T signal 7	VDDIO_T	NC
GPLOT_8	DIO	Down	General purpose input/output bank T signal 8	VDDIO_T	NC
GPLOT_9	DIO	Down	General purpose input/output bank T signal 9	VDDIO_T	NC
GPLOT_10	DIO	Down	General purpose input/output bank T signal 10	VDDIO_T	NC
GPLOT_11	DIO	Down	General purpose input/output bank T signal 11	VDDIO_T	NC
GPLOT_12	DIO	Down	General purpose input/output bank T signal 12	VDDIO_T	NC
GPLOT_13	DIO	Down	General purpose input/output bank T signal 13	VDDIO_T	NC
GPLOT_14	DIO	Down	General purpose input/output bank T signal 14	VDDIO_T	NC
GPLOT_15	DIO	Down	General purpose input/output bank T signal 15	VDDIO_T	NC
GPLOT_16	DIO	Down	General purpose input/output bank T signal 16	VDDIO_T	NC
GPLOT_17	DIO	Down	General purpose input/output bank T signal 17	VDDIO_T	NC
GPLOT_18	DIO	Down	General purpose input/output bank T signal 18	VDDIO_T	NC
GPLOT_19	DIO	Down	General purpose input/output bank T signal 19	VDDIO_T	NC
GPLOT_20	DIO	Down	General purpose input/output bank T signal 20	VDDIO_T	NC
GPLOT_21	DIO	Down	General purpose input/output bank T signal 21	VDDIO_T	NC



Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
GPIOT_22	DIO	Down	General purpose input/output bank T signal 22	VDDIO_T	NC
GPIOT_23	DIO	Down	General purpose input/output bank T signal 23	VDDIO_T	NC
VDDIO_T	P	-	Power supply for GPIO bank T	-	To 1.8V or 3.3V
<b>GPIO M</b> - refer to <a href="#">Table 4-11</a> for functional multiplex information.					
GPIOM_0	DIO	Down	General purpose input/output bank M signal 0	VDDIO_M	NC
GPIOM_1	DIO	Down	General purpose input/output bank M signal 1	VDDIO_M	NC
GPIOM_2	DIO	Down	General purpose input/output bank M signal 2	VDDIO_M	NC
GPIOM_3	DIO	Down	General purpose input/output bank M signal 3	VDDIO_M	NC
GPIOM_4	DIO	Down	General purpose input/output bank M signal 4	VDDIO_M	NC
GPIOM_5	DIO	Down	General purpose input/output bank M signal 5	VDDIO_M	NC
GPIOM_6	DIO	Up	General purpose input/output bank M signal 6	VDDIO_M	NC
GPIOM_7	DIO	Up	General purpose input/output bank M signal 7	VDDIO_M	NC
GPIOM_8	DIO	Down	General purpose input/output bank M signal 8	VDDIO_M	NC
GPIOM_9	DIO	Down	General purpose input/output bank M signal 9	VDDIO_M	NC
GPIOM_10	DIO	Down	General purpose input/output bank M signal 10	VDDIO_M	NC
GPIOM_11	DIO	Down	General purpose input/output bank M signal 11	VDDIO_M	NC
GPIOM_12	DIO	Up	General purpose input/output bank M signal 12	VDDIO_M	NC
GPIOM_13	DIO	Up	General purpose input/output bank M signal 13	VDDIO_M	NC
VDDIO_M	P	-	Power supply for GPIO bank M	-	To 1.8V or 3.3V
<b>GPIO Y</b> - refer to <a href="#">Table 4-14</a> for functional multiplex information.					
GPIOY_0	DIO	Down	General purpose input/output bank Y signal 0	VDDIO_Y	NC
GPIOY_1	DIO	Down	General purpose input/output bank Y signal 1	VDDIO_Y	NC
GPIOY_2	DIO	Down	General purpose input/output bank Y signal 2	VDDIO_Y	NC
GPIOY_3	DIO	Up	General purpose input/output bank Y signal 3	VDDIO_Y	NC
GPIOY_4	DIO	Up	General purpose input/output bank Y signal 4	VDDIO_Y	NC
GPIOY_5	DIO	Up	General purpose input/output bank Y signal 5	VDDIO_Y	NC
GPIOY_6	DIO	Up	General purpose input/output bank Y signal 6	VDDIO_Y	NC
GPIOY_7	DIO	Down	General purpose input/output bank Y signal 7	VDDIO_Y	NC
GPIOY_8	DIO	Down	General purpose input/output bank Y signal 8	VDDIO_Y	NC
GPIOY_9	DIO	Down	General purpose input/output bank Y signal 9	VDDIO_Y	NC
GPIOY_10	DIO	Up	General purpose input/output bank Y signal 10	VDDIO_Y	NC
GPIOY_11	DIO	Up	General purpose input/output bank Y signal 11	VDDIO_Y	NC
GPIOY_12	DIO	Up	General purpose input/output bank Y signal 12	VDDIO_Y	NC
GPIOY_13	DIO	Up	General purpose input/output bank Y signal 13	VDDIO_Y	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
GPIOW_14	DIO	Down	General purpose input/output bank Y signal 14	VDDIO_Y	NC
GPIOW_15	DIO	Up	General purpose input/output bank Y signal 15	VDDIO_Y	NC
GPIOW_16	DIO	Down	General purpose input/output bank Y signal 16	VDDIO_Y	NC
GPIOW_17	DIO	Up	General purpose input/output bank Y signal 17	VDDIO_Y	NC
GPIOW_18	DIO	Up	General purpose input/output bank Y signal 18	VDDIO_Y	NC
VDDIO_Y	P	-	Power supply for GPIO bank Y	-	To 1.8V or 3.3V
<b>GPIOH-</b> refer to <a href="#">Table 4-14</a> for functional multiplex information.					
GPIOH_0	DIO	Up	General purpose input/output bank H signal 0	VDDIO_H	NC
GPIOH_1	DIO	HiZ	General purpose input/output bank H signal 1	VDDIO_H	NC
GPIOH_2	DIO	Down	General purpose input/output bank H signal 2	VDDIO_H	NC
GPIOH_3	DIO	Up	General purpose input/output bank H signal 3	VDDIO_H	NC
GPIOH_4	DIO	Down	General purpose input/output bank H signal 4	VDDIO_H	NC
GPIOH_5	DIO	Down	General purpose input/output bank H signal 5	VDDIO_H	NC
GPIOH_6	DIO	HiZ	General purpose input/output bank H signal 6	VDDIO_H	NC
GPIOH_7	DIO	HiZ	General purpose input/output bank H signal 7	VDDIO_H	NC
VDDIO_H	P	-	Power supply for GPIO bank H	-	To 1.8V or 3.3V
<b>HDMI RX</b>					
HDMIRX_A_CLKN	AI	-	HDMIRX Port A TMDS clock negative input	AVDD33_HDMIRX	NC
HDMIRX_A_CLKP	AI	-	HDMIRX Port A TMDS clock positive input	AVDD33_HDMIRX	NC
HDMIRX_A_D0N	AI	-	HDMIRX Port A TMDS data0 negative input	AVDD33_HDMIRX	NC
HDMIRX_A_D0P	AI	-	HDMIRX Port A TMDS data0 positive input	AVDD33_HDMIRX	NC
HDMIRX_A_D1N	AI	-	HDMIRX Port A TMDS data1 negative input	AVDD33_HDMIRX	NC
HDMIRX_A_D1P	AI	-	HDMIRX Port A TMDS data1 positive input	AVDD33_HDMIRX	NC
HDMIRX_A_D2N	AI	-	HDMIRX Port A TMDS data2 negative input	AVDD33_HDMIRX	NC
HDMIRX_A_D2P	AI	-	HDMIRX Port A TMDS data2 positive input	AVDD33_HDMIRX	NC
HDMIRX_ARCTXN	AO	-	Audio Return Channel output	AVDD18_HDMIRX_EARCTX	NC
HDMIRX_ARCTXP	AO	-	Audio Return Channel output	AVDD18_HDMIRX_EARCTX	NC
HDMIRX_B_CLKN	AI	-	HDMIRX Port B TMDS clock negative input	AVDD33_HDMIRX	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
HDMIRX_B_CLKP	AI	-	HDMIRX Port B TMDS clock positive input	AVDD33_HDMIRX	NC
HDMIRX_B_D0N	AI	-	HDMIRX Port B TMDS data0 negative input	AVDD33_HDMIRX	NC
HDMIRX_B_D0P	AI	-	HDMIRX Port B TMDS data0 positive input	AVDD33_HDMIRX	NC
HDMIRX_B_D1N	AI	-	HDMIRX Port B TMDS data1 negative input	AVDD33_HDMIRX	NC
HDMIRX_B_D1P	AI	-	HDMIRX Port B TMDS data1 positive input	AVDD33_HDMIRX	NC
HDMIRX_B_D2N	AI	-	HDMIRX Port B TMDS data2 negative input	AVDD33_HDMIRX	NC
HDMIRX_B_D2P	AI	-	HDMIRX Port B TMDS data2 positive input	AVDD33_HDMIRX	NC
HDMIRX_C_CLKN	AI	-	HDMIRX Port C TMDS clock negative input	AVDD33_HDMIRX	NC
HDMIRX_C_CLKP	AI	-	HDMIRX Port C TMDS clock positive input	AVDD33_HDMIRX	NC
HDMIRX_C_D0N	AI	-	HDMIRX Port C TMDS data0 negative input	AVDD33_HDMIRX	NC
HDMIRX_C_D0P	AI	-	HDMIRX Port C TMDS data0 positive input	AVDD33_HDMIRX	NC
HDMIRX_C_D1N	AI	-	HDMIRX Port C TMDS data1 negative input	AVDD33_HDMIRX	NC
HDMIRX_C_D1P	AI	-	HDMIRX Port C TMDS data1 positive input	AVDD33_HDMIRX	NC
HDMIRX_C_D2N	AI	-	HDMIRX Port C TMDS data2 negative input	AVDD33_HDMIRX	NC
HDMIRX_C_D2P	AI	-	HDMIRX Port C TMDS data2 positive input	AVDD33_HDMIRX	NC
AVDD18_HDMIRX_EARCTX	AP	-	Analog Power supply 1.8V for HDMIRX and eARCTX	-	To 1.8V
AVDD08_HDMIRX	AP	-	Analog Power supply 0.8V for HDMI RX	-	To 0.8V
AVDD33_HDMIRX	AP	-	Power supply for HDMI RX analog 3.3V	-	To 3.3V
<b>HDMI TX</b>					
HDMITX_ARCRXN	AO	-	Audio Return Channel iutput	AVDD18_HDMITX_EARCRX	NC
HDMITX_ARCRXP	AO	-	Audio Return Channel iutput	AVDD18_HDMITX_EARCRX	NC
HDMITX_CLKN	AO	-	HDMI TMDS clock positive output	3.3V	NC
HDMITX_CLKP	AO	-	HDMI TMDS clock negative output	3.3V	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
HDMITX_D0N	AO	-	HDMI TMDS data0 positive output	3.3V	NC
HDMITX_D0P	AO	-	HDMI TMDS data0 negative output	3.3V	NC
HDMITX_D1N	AO	-	HDMI TMDS data1 positive output	3.3V	NC
HDMITX_D1P	AO	-	HDMI TMDS data1 negative output	3.3V	NC
HDMITX_D2N	AO	-	HDMI TMDS data2 positive output	3.3V	NC
HDMITX_D2P	AO	-	HDMI TMDS data2 negative output	3.3V	NC
HDMITX_REXT	A	-	HDMI output strength setting resistor	AVDD18_HDMITX_EARCRX	NC
AVDD18_HDMITX_EARCRX	AP	-	Analog Power supply 1.8V for HDMITX and eARCRX	-	To 1.8V
AVDD08_HDMITX	AP	-	Analog Power supply 0.8V for HDMI TX	-	To 0.8V
AVDD33_HDMITX	AP	-	Pin for decap, Only decap is needed	-	NC
AVSS_HPLL	AP	-	Ground of HDMI PLL	-	To GND
<b>PCIe and USB</b>					
USB_A_OTG_DM	AIO	-	USB 2.0 Port A OTG negative data signal	AVDD33_USB	NC
USB_A_OTG_DP	AIO	-	USB 2.0 Port A OTG positive data signal	AVDD33_USB	NC
USB_A_OTG_ID	AI	-	USB 2.0 Port A ID detect signal, internal pull up to 1.8V	AVDD18_USB_PCIE	NC
USB_A_OTG_VBUS	AI	-	USB 2.0 Port A host cable power detection (1.8V input tolerance)	AVDD18_USB_PCIE	NC
USB_B_OTG_DM	AIO	-	USB 2.0 Port B OTG negative data signal	AVDD33_USB	NC
USB_B_OTG_DP	AIO	-	USB 2.0 Port B OTG positive data signal	AVDD33_USB	NC
USB_B_OTG_ID	AI	-	USB 2.0 Port B ID detect signal, internal pull up to 1.8V	AVDD18_USB_PCIE	NC
USB_B_OTG_VBUS	AI	-	USB 2.0 Port B host cable power detection (1.8V input tolerance)	AVDD18_USB_PCIE	NC
USB20_TXRTUNE	AIO	-	USB 2.0 PortA and Port B host output strength setting resistor	AVDD33_USB	NC
USB30_PCIE_REXT	AO	-	PCIe and USB3.0 port output strength setting resistor	AVDD18_USB_PCIE	NC
USB30_RXN	AI	-	USB3.0 port B input negative signal	AVDD18_USB_PCIE	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
USB30_RXP	AI	-	USB3.0 port B input positive signal	AVDD18_USB_PCIE	NC
USB30_TXN	AO	-	USB3.0 port B output negative signal	AVDD18_USB_PCIE	NC
USB30_TXP	AO	-	USB3.0 port B output positive signal	AVDD18_USB_PCIE	NC
PCIE_CLK_N	AO	-	PCIe reference clock negative signal	AVDD18_USB_PCIE	NC
PCIE_CLK_P	AO	-	PCIe reference clock positive signal	AVDD18_USB_PCIE	NC
PCIE_RXN	AI	-	PCIe input negative signal	AVDD18_USB_PCIE	NC
PCIE_RXP	AI	-	PCIe input positive signal	AVDD18_USB_PCIE	NC
PCIE_TXN	AO	-	PCIe output negative signal	AVDD18_USB_PCIE	NC
PCIE_TXP	AO	-	PCIe output positive signal	AVDD18_USB_PCIE	NC
HCSL_REXT	AIO	-	PCIe reference clk output strength setting resistor	AVDD18_USB_PCIE	NC
AVDD0V8_USB_PCIE	AP	-	Analog 0.8V power supply for USB and PCIe	-	To VDD_EE
AVDD18_USB_PCIE	AP	-	Analog 1.8V power supply for USB and PCIe	-	To 1.8V
AVDD33_USB	P	-	3.3V Power supply for USB	-	To 3.3V
<b>POR</b>					
POR_OUT	AO	Down	Power on reset output	AVDD33_POR	NC
AVDD33_POR	AP	-	Power supply for POR	-	NC
<b>SARADC</b>					
SARADC_CH0	AI	-	ADC channel 0 input	AVDD18_SARADC	NC
SARADC_CH1	AI	-	ADC channel 1 input	AVDD18_SARADC	NC
SARADC_CH2	AI	-	ADC channel 2 input	AVDD18_SARADC	NC
SARADC_CH3	AI	-	ADC channel 3 input	AVDD18_SARADC	NC
SARADC_CH6	AI	-	ADC channel 6 input	AVDD18_SARADC	NC
AVDD18_SARADC	P	-	Analog power supply for SARADC	-	To 1.8V
<b>System Clock &amp; PLL</b>					
SYS_OSCIN	AI	-	24MHz crystal oscillator input	VDD18_AO	To XTAL

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
SYS_OSCOUT	AO	-	24MHz crystal oscillator output	VDD18_AO	To XTAL
AVDD18_PLL	AP	-	Analog power of System PLL	-	To 1.8V
AVSS_PLL	AP	-	Ground of System PLL	-	To GND
<b>MCLK</b>					
CM_MCLK1	AO	-	MCLK 1 output for camera	AVDD18_MCLK	NC
CM_MCLK2	AO	-	MCLK 2 output for camera	AVDD18_MCLK	NC
AVDD18_MCLK	AP	-	Power supply 1.8V for MCLK	-	To 1.8V
<b>Digital Power</b>					
VDD_DDR	P	-	Core Power supply for DDR PHY	-	To VDD_EE
VDD_EE	P	-	Power supply for core logic	-	To VDD_EE
VDD_GPU	P	-	Power supply for GPU	-	To VDD_GPU
VDD18_AO	P	-	1.8V Power supply for Always On Domain	-	To VDD18_AO
VDDCPU_A	P	-	Power supply for CPU (Cortex A73)	-	To VDDCPU_A
VDDCPU_B	P	-	Power supply for CPU (Cortex A53)	-	To VDDCPU_B
VDDQ	P	-	DDR IO Power supply for DDR PHY	-	To VDDQ
VDDQLP	P	-	DDR IO Power supply for DDR PHY	-	To VDDQ
DVSS	P	-	Digital power ground	-	To GND
<b>Display interface</b>					
VX1_A_0N	AO	-	V-by-one A data0 or LVDS A data0 or MIPI DSI A data0 or eDP A Aux negative output	AVDD18_DP	NC
VX1_A_0P	AO	-	V-by-one A data0 or LVDS A data0 or MIPI DSI A data0 or eDP A Aux positive output	AVDD18_DP	NC
VX1_A_1N	AO	-	V-by-one A data1 or LVDS A data1 or MIPI DSI A data1 or eDP A data0 negative output	AVDD18_DP	NC
VX1_A_1P	AO	-	V-by-one A data1 or LVDS A data1 or MIPI DSI A data1 or eDP A data0 positive output	AVDD18_DP	NC
VX1_A_2N	AO	-	V-by-one A data2 or LVDS A data2 or MIPI DSI A clk or eDP A data1 negative output	AVDD18_DP	NC
VX1_A_2P	AO	-	V-by-one A data2 or LVDS A data2 or MIPI DSI A clk or eDP A data1 positive output	AVDD18_DP	NC
VX1_A_3N	AO	-	V-by-one A data3 or LVDS A clk or MIPI DSI A data2 or eDP A data2 negative output	AVDD18_DP	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
VX1_A_3P	AO	-	V-by-one A data3 or LVDS A clk or MIPI DSI A data2 or eDP A data2 positive output	AVDD18_DP	NC
VX1_A_4N	AO	-	V-by-one A data4 or LVDS A data3 or MIPI DSI A data3 or eDP A data3 negative output	AVDD18_DP	NC
VX1_A_4P	AO	-	V-by-one A data4 or LVDS A data3 or MIPI DSI A data3 or eDP A data3 positive output	AVDD18_DP	NC
VX1_A_5N	AO	-	V-by-one A data5 or LVDS B data0 negative output	AVDD18_DP	NC
VX1_A_5P	AO	-	V-by-one A data5 or LVDS B data0 positive output	AVDD18_DP	NC
VX1_A_6N	AO	-	V-by-one A data6 or LVDS B data1 negative output	AVDD18_DP	NC
VX1_A_6P	AO	-	V-by-one A data6 or LVDS B data1 positive output	AVDD18_DP	NC
VX1_A_7N	AO	-	V-by-one A data7 or LVDS B data2 negative output	AVDD18_DP	NC
VX1_A_7P	AO	-	V-by-one A data7 or LVDS B data2 positive output	AVDD18_DP	NC
VX1_ATP	AO	-	V-by-one B or LVDS or MIPI DSI or eDP test output	AVDD18_DP	NC
VX1_B_0N	AO	-	V-by-one B data0 or LVDS B clk or MIPI DSI B data0 or eDP B Aux negative output	AVDD18_DP	NC
VX1_B_0P	AO	-	V-by-one B data0 or LVDS B clk or MIPI DSI B data0 or eDP B Aux positive output	AVDD18_DP	NC
VX1_B_1N	AO	-	V-by-one B data1 or LVDS B data3 or MIPI DSI B data1 or eDP B data0 negative output	AVDD18_DP	NC
VX1_B_1P	AO	-	V-by-one B data1 or LVDS B data3 or MIPI DSI B data1 or eDP B data0 positive output	AVDD18_DP	NC
VX1_B_2N	AO	-	V-by-one B data2 or LVDS C data0 or MIPI DSI B clk or eDP B data1 negative output	AVDD18_DP	NC
VX1_B_2P	AO	-	V-by-one B data2 or LVDS C data0 or MIPI DSI B clk or eDP B data1 positive output	AVDD18_DP	NC
VX1_B_3N	AO	-	V-by-one B data3 or LVDS C data1 or MIPI DSI B data2 or eDP B data2 negative output	AVDD18_DP	NC
VX1_B_3P	AO	-	V-by-one B data3 or LVDS C data1 or MIPI DSI B data2 or eDP B data2 positive output	AVDD18_DP	NC
VX1_B_4N	AO	-	V-by-one B data4 or LVDS C data2 or MIPI DSI B data3 or eDP B data3 negative output	AVDD18_DP	NC
VX1_B_4P	AO	-	V-by-one B data4 or LVDS C data2 or MIPI DSI B data3 or eDP B data3 positive output	AVDD18_DP	NC
VX1_B_5N	AO	-	V-by-one B data5 or LVDS C clk negative output	AVDD18_DP	NC
VX1_B_5P	AO	-	V-by-one B data5 or LVDS C clk positive output	AVDD18_DP	NC
VX1_B_6N	AO	-	V-by-one B data6 or LVDS C data3 negative output	AVDD18_DP	NC

Net Name	Type	Default Pull Up/Dn	Description	Power Domain	If Unused
VX1_B_6P	AO	-	V-by-one B data6 or LVDS C data3 positive output	AVDD18_DP	NC
VX1_B_7N	AO	-	V-by-one B data7 negative output	AVDD18_DP	NC
VX1_B_7P	AO	-	V-by-one B data7 positive output	AVDD18_DP	NC
VX1_LPN	AO	-	V-by-one B or LVDS or MIPI DSI or eDP test output	AVDD18_DP	NC
VX1_LPP	AO	-	V-by-one B or LVDS or MIPI DSI or eDP test output	AVDD18_DP	NC
DIF_REXT	AO	-	eDP, MIPI DSI, LVDS and V-by-one reference resistor	AVDD18_DP	NC
AVDD08_DP	AP	-	Analog Power supply 0.8V for Display interface	-	To 0.8V
AVDD18_DP	AP	-	Analog Power supply 1.8V for display interface	-	To 1.8V
AVDD18_DDR_DP_PLL	AP	-	Analog Power supply 0.8V for DDR and display interface PLL	-	To 0.8V
AVSS_DP	AP	-	Analog ground for display interface	-	To VSS

#### Abbreviations:

- DI = Digital input pin
- DO = Digital output pin
- DIO = Digital input/output pin
- OD5V = 5V input tolerant open drain (OD) output pin, need external pull Up
- A = Analog setting or filtering pin
- AI = Analog input pin
- AO = Analog output pin
- AIO = Analog input/output pin
- P = Power pin
- AP = Analog power pin
- NC = No connection
- Up = Pull-Up
- Down = Pull-down
- COMB PIN = Combined Pin
- Z = High-Z

## 4.4 Pin Multiplexing Tables

Multiple usage pins are used to conserve pin consumption for different features. The device can be used in many different applications but each application will not utilize all the onchip features. As a result, some of the features share the same pin. Most of the multiple usage pins can be used as a GPIO pin as well.



**Table 4-2 GPIOW\_x Multi-Function Pin**

Pin Name	Func1	Func2	Func3
GPIOW_0	HDMIRX_A_HPD		
GPIOW_1	HDMIRX_A_5VDET		
GPIOW_2	HDMIRX_A_SDA	UART_AO_A_TX	HDMITX_SDA
GPIOW_3	HDMIRX_A_SCL	UART_AO_A_RX	HDMITX_SCL
GPIOW_4	HDMIRX_C_HPD		
GPIOW_5	HDMIRX_C_5VDET		
GPIOW_6	HDMIRX_C_SDA	UART_AO_A_TX	
GPIOW_7	HDMIRX_C_SCL	UART_AO_A_RX	
GPIOW_8	HDMIRX_B_HPD		
GPIOW_9	HDMIRX_B_5VDET		
GPIOW_10	HDMIRX_B_SDA	UART_AO_A_TX	
GPIOW_11	HDMIRX_B_SCL	UART_AO_A_RX	
GPIOW_12	CEC_A		
GPIOW_13	HDMITX_SDA		
GPIOW_14	HDMITX_SCL		
GPIOW_15	HDMITX_HPD_IN		
GPIOW_16	CEC_B		

**Table 4-3 GPIOD\_x Multi-Function Pin**

Pin Name	Func1	Func2	Func3	Func4
GPIOD_0	UART_AO_A_TX			
GPIOD_1	UART_AO_A_RX			
GPIOD_2	I2CM_AO_A_SCL	I2CS_AO_SCL	UART_AO_B_TX	
GPIOD_3	I2CM_AO_A_SDA	I2CS_AO_SDA	UART_AO_B_RX	
GPIOD_4	IR_OUT	RTC_CLK_IN	UART_AO_B_CTS	
GPIOD_5	IR_IN	PWM_AO_H		
GPIOD_6	JTAG_A_CLK	PWM_AO_C	PWM_AO_C_HIZ	IR_OUT
GPIOD_7	JTAG_A_TMS	PWM_AO_G	PWM_AO_G_HIZ	
GPIOD_8	JTAG_A_TDI	SPDIF_OUT		
GPIOD_9	JTAG_A_TDO	SPDIF_IN		
GPIOD_10	GEN_CLK_OUT	PWM_AO_H	UART_AO_B_RTS	
GPIOD_11	PWM_AO_G			
GPIOD_12	WD_RSTO			

**Table 4-4 GPIOE\_x Multi-Function Pin**

Pin Name	Func1	Func2	Func3
GPIOE_0	PWM_AO_A	I2CM_AO_A_SCL	
GPIOE_1	PWM_AO_B	I2CM_AO_A_SDA	
GPIOE_2	PWM_AO_C	CLK25M	
GPIOE_3	PWM_AO_D	I2CM_AO_B_SCL	
GPIOE_4	PWM_AO_E	I2CM_AO_B_SDA	CLK12M_24M
GPIOE_5	PWM_AO_F	RTC_CLK_OUT	
GPIOE_6	PWM_AO_G		

**Table 4-5 GPIOB\_x Multi-Function Pin**

Pin Name	Func1	Func2
GPIOB_0	EMMC_D0	
GPIOB_1	EMMC_D1	
GPIOB_2	EMMC_D2	
GPIOB_3	EMMC_D3	SPIF_HOLD
GPIOB_4	EMMC_D4	SPIF_MO
GPIOB_5	EMMC_D5	SPIF_MI
GPIOB_6	EMMC_D6	SPIF_CLK
GPIOB_7	EMMC_D7	SPIF_WP
GPIOB_8	EMMC_CLK	
GPIOB_9		
GPIOB_10	EMMC_CMD	
GPIOB_11	EMMC_DS	
GPIOB_12		SPIF_CS

**Table 4-6 GPIOC\_x Multi-Function Pin**

Pin Name	Func1	Func2	Func3
GPIOC_0	SDCARD_D0	JTAG_B_TDO	SPI_B_MOSI
GPIOC_1	SDCARD_D1	JTAG_B_TDI	SPI_B_MISO
GPIOC_2	SDCARD_D2	UART_AO_A_RX	SPI_B_SCLK
GPIOC_3	SDCARD_D3	UART_AO_A_TX	SPI_B_SS0
GPIOC_4	SDCARD_CLK	JTAG_B_CLK	
GPIOC_5	SDCARD_CMD	JTAG_B_TMS	
GPIOC_6	GEN_CLK_OUT		

**Table 4-7 GPIOZ\_x Multi-Function Pin**

Pin Name	Func1	Func2	Func3	Func4
GPIOZ_0	ETH_MDIO	ISO7816_CLK		SPI_E_MOSI
GPIOZ_1	ETH_MDC	ISO7816_DATA		SPI_E_MISO
GPIOZ_2	ETH_RGMII_RX_CLK	TSIN_B_VALID		SPI_E_SCLK
GPIOZ_3	ETH_RX_DV	TSIN_B_SOP		SPI_E_SS0
GPIOZ_4	ETH_RXD0	TSIN_B_DIN0		SPI_F_MOSI
GPIOZ_5	ETH_RXD1	TSIN_B_CLK		SPI_F_MISO
GPIOZ_6	ETH_RXD2_RGMII	TSIN_B_FAIL	TSIN_C_VALID	SPI_F_SCLK
GPIOZ_7	ETH_RXD3_RGMII	TSIN_B_DIN1	TSIN_C_SOP	SPI_F_SS0
GPIOZ_8	ETH_RGMII_TX_CLK	TSIN_B_DIN2	TSIN_C_DIN0	
GPIOZ_9	ETH_TXEN	TSIN_B_DIN3	TSIN_C_CLK	
GPIOZ_10	ETH_TXD0	TSIN_B_DIN4	TSIN_D_VALID	
GPIOZ_11	ETH_TXD1	TSIN_B_DIN5	TSIN_D_SOP	
GPIOZ_12	ETH_TXD2_RGMII	TSIN_B_DIN6	TSIN_D_DIN0	
GPIOZ_13	ETH_TXD3_RGMII	TSIN_B_DIN7	TSIN_D_CLK	

**Table 4-8 GPIOH\_x Multi-Function Pin**

Pin Name	Func1	Func2	Func3
GPIOH_0	MIC_MUTE_KEY		
GPIOH_1	Mic_Mute_LED	PWM_VS	
GPIOH_2	I2CM_D_SCL	UART_F_TX	PCIECK_REQN
GPIOH_3	I2CM_D_SDA	UART_F_RX	
GPIOH_4	I2CM_E_SCL	UART_F_CTS	
GPIOH_5	I2CM_E_SDA	UART_F_RTS	
GPIOH_6	ETH_LINK_LED	I2CM_A_SDA	
GPIOH_7	ETH_ACT_LED	I2CM_A_SCL	

**Table 4-9 GPIOM\_x Multi-Function Pin**

Pin Name	Func1	Func2
GPIOM_0	TDM_D12	PDM_DIN1
GPIOM_1	TDM_D13	PDM_DIN2
GPIOM_2	TDM_D14	PDM_DIN3
GPIOM_3	TDM_D15	PDM_DCLK
GPIOM_4	TDM_SCLK3	PDM_DIN0
GPIOM_5	TDM_FS3	PDM_DIN1

Pin Name	Func1	Func2
GPIOM_6	I2CM_D_SCL	
GPIOM_7	I2CM_D_SDA	
GPIOM_8	SPI_B_MOSI	UART_D_TX
GPIOM_9	SPI_B_MISO	UART_D_RX
GPIOM_10	SPI_B_SCLK	UART_D_CTS
GPIOM_11	SPI_B_SS0	UART_D_RTS
GPIOM_12	SPI_B_SS1	I2CM_C_SCL
GPIOM_13	SPI_B_SS2	I2CM_C_SDA

Table 4-10 GPIOX\_x Multi-Function Pin

Pin Name	Func1	Func2
GPIOX_0	SDIO_D0	
GPIOX_1	SDIO_D1	
GPIOX_2	SDIO_D2	
GPIOX_3	SDIO_D3	
GPIOX_4	SDIO_CLK	
GPIOX_5	SDIO_CMD	
GPIOX_6	PWM_B	
GPIOX_7	PWM_C	
GPIOX_8	TDM_D0	
GPIOX_9	TDM_D1	
GPIOX_10	TDM_FS0	
GPIOX_11	TDM_SCLK0	
GPIOX_12	UART_C_TX	
GPIOX_13	UART_C_RX	
GPIOX_14	UART_C_CTS	CLK12M_24M
GPIOX_15	UART_C_RTS	
GPIOX_16	PWM_A	
GPIOX_17	I2CM_C_SDA	
GPIOX_18	I2CM_C_SCL	
GPIOX_19	PWM_D	

**Table 4-11 GPIOT\_x Multi-Function Pin**

Pin Name	Func1	Func2	Func3
GPIOT_0	MCLK_1		
GPIOT_1	TDM_SCLK1		
GPIOT_2	TDM_FS1		
GPIOT_3	TDM_D2	SPDIF_IN	
GPIOT_4	TDM_D3	SPDIF_OUT	
GPIOT_5	TDM_D4	ISO7816_CLK	
GPIOT_6	TDM_D5	ISO7816_DATA	SPI_D_MOSI
GPIOT_7	TDM_D6	TSIN_A_SOP	SPI_D_MISO
GPIOT_8	TDM_D7	TSIN_A_DIN0	SPI_D_SCLK
GPIOT_9	TDM_D8	TSIN_A_CLK	SPI_D_SS0
GPIOT_10	TDM_D9	TSIN_A_VALID	
GPIOT_11	TDM_D10		
GPIOT_12	TDM_D11		
GPIOT_13	MCLK_2		
GPIOT_14	TDM_SCLK2		
GPIOT_15	TDM_FS2		
GPIOT_16	I2CM_B_SCL		
GPIOT_17	I2CM_B_SDA		
GPIOT_18	SPI_A_MOSI		
GPIOT_19	SPI_A_MISO		
GPIOT_20	SPI_A_SCLK	I2CM_A_SCL	
GPIOT_21	SPI_A_SS0	I2CM_A_SDA	
GPIOT_22	SPI_A_SS1	I2CM_C_SCL	
GPIOT_23	SPI_A_SS2	I2CM_C_SDA	

**Table 4-12 GPIOY\_x Multi-Function Pin**

Pin Name	Func1	Func2	Func3	Func4
GPIOY_0	SPI_C_MOSI			
GPIOY_1	SPI_C_MISO		PWM_E	
GPIOY_2	SPI_C_SCLK			
GPIOY_3	SPI_C_SS0			
GPIOY_4	SPI_C_SS1	TSIN_C_SOP	HSYNC	
GPIOY_5	SPI_C_SS2	TSIN_C_DIN0	VSYNC	
GPIOY_6	UART_E_TX	TSIN_C_CLK		

Pin Name	Func1	Func2	Func3	Func4
GPIOY_7	UART_E_RX	TSIN_C_VALID		
GPIOY_8	UART_E_CTS	TSIN_D_SOP	PWM_F	
GPIOY_9	UART_E_RTS	TSIN_D_DIN0	3D_SYNC_OUT	
GPIOY_10	UART_D_CTS	TSIN_D_CLK	VX1_A_HTPDN	eDP_A_HPD
GPIOY_11	UART_D_RTS	TSIN_D_VALID	VX1_B_HTPDN	eDP_B_HPD
GPIOY_12	UART_D_TX		VX1_A_LOCKN	
GPIOY_13	UART_D_RX		VX1_B_LOCKN	
GPIOY_14			PWM_VS	
GPIOY_15	I2CM_E_SCL			
GPIOY_16	I2CM_E_SDA			
GPIOY_17	I2CM_F_SCL			
GPIOY_18	I2CM_F_SDA	PCIECK_REQN		

Table 4-13 CSI Multi-Function Pin

Pin Name	Mux
CSI_B_D0N	CSI_A_D2N
CSI_B_D0P	CSI_A_D2P
CSI_B_D1N	CSI_A_D3N
CSI_B_D1P	CSI_A_D3P
CSI_D_D0N	CSI_C_D2N
CSI_D_D0P	CSI_C_D2P
CSI_D_D1N	CSI_C_D3N
CSI_D_D1P	CSI_C_D3P

Table 4-14 VX1 Multi-Function Pin

Pin Name	LVDS	EDP	DSI
VX1_A_0N	LVDS_A_0N	EDP_A_AUXN	DSI_A_D0N
VX1_A_0P	LVDS_A_0P	EDP_A_AUXP	DSI_A_D0P
VX1_A_1N	LVDS_A_1N	EDP_A_0N	DSI_A_D1N
VX1_A_1P	LVDS_A_1P	EDP_A_0P	DSI_A_D1P
VX1_A_2N	LVDS_A_2N	EDP_A_1N	DSI_A_CKN
VX1_A_2P	LVDS_A_2P	EDP_A_1P	DSI_A_CKP
VX1_A_3N	LVDS_A_CLKN	EDP_A_2N	DSI_A_D2N
VX1_A_3P	LVDS_A_CLKP	EDP_A_2P	DSI_A_D2P
VX1_A_4N	LVDS_A_3N	EDP_A_3N	DSI_A_D3N

Pin Name	LVDS	EDP	DSI
VX1_A_4P	LVDS_A_3P	EDP_A_3P	DSI_A_D3P
VX1_A_5N	LVDS_B_0N		
VX1_A_5P	LVDS_B_0P		
VX1_A_6N	LVDS_B_1N		
VX1_A_6P	LVDS_B_1P		
VX1_A_7N	LVDS_B_2N		
VX1_A_7P	LVDS_B_2P		
VX1_B_0N	LVDS_B_CLKN	EDP_B_AUXN	DSI_B_D0N
VX1_B_0P	LVDS_B_CLKP	EDP_B_AUXP	DSI_B_D0P
VX1_B_1N	LVDS_B_3N	EDP_B_0N	DSI_B_D1N
VX1_B_1P	LVDS_B_3P	EDP_B_0P	DSI_B_D1P
VX1_B_2N	LVDS_C_0N	EDP_B_1N	DSI_B_CKN
VX1_B_2P	LVDS_C_0P	EDP_B_1P	DSI_B_CKP
VX1_B_3N	LVDS_C_1N	EDP_B_2N	DSI_B_D2N
VX1_B_3P	LVDS_C_1P	EDP_B_2P	DSI_B_D2P
VX1_B_4N	LVDS_C_2N	EDP_B_3N	DSI_B_D3N
VX1_B_4P	LVDS_C_2P	EDP_B_3P	DSI_B_D3P
VX1_B_5N	LVDS_C_CLKN		
VX1_B_5P	LVDS_C_CLKP		
VX1_B_6N	LVDS_C_3N		
VX1_B_6P	LVDS_C_3P		
VX1_B_7N			
VX1_B_7P			

Table 4-15 DDR Multi-Function Pin

Pin Name	LPDDR4/LPDDR4X	DDR4
DDR0_AC_0	CKEA0	CKE0
DDR0_AC_1	CKEA1	CKE1
DDR0_AC_2	CSA0	CS_N0
DDR0_AC_3	CSA1	NC
DDR0_AC_4	CLKA_T	BG0
DDR0_AC_5	CLKA_C	A10
DDR0_AC_6	NC	A8
DDR0_AC_7	NC	A11
DDR0_AC_8	CAA2	WE_N

Pin Name	LPDDR4/LPDDR4X	DDR4
DDR0_AC_9	CAA3	BG1
DDR0_AC_10	CAA1	A12
DDR0_AC_11	CAA0	RAS_N
DDR0_AC_12	CAA5	A3
DDR0_AC_13	CAA4	CAS_N
DDR0_AC_14	NC	A7
DDR0_AC_15	NC	A13
DDR0_AC_20	CKEB0	CLK1_T
DDR0_AC_21	CKEB1	CLK1_C
DDR0_AC_22	CSB1	NC
DDR0_AC_23	CSB0	NC
DDR0_AC_24	CLKB_T	BA1
DDR0_AC_25	CLKB_C	A4
DDR0_AC_26	NC	BA0
DDR0_AC_28	CAB1	ACT_N
DDR0_AC_29	CAB3	A0
DDR0_AC_30	CAB5	A9
DDR0_AC_31	CAB2	A1
DDR0_AC_32	CAB4	NC
DDR0_AC_33	CAB0	A6
DDR0_AC_34	NC	A2
DDR0_AC_35	NC	A5
DDR0_AC_36	NC	ODT0
DDR0_AC_37	NC	ODT1
DDR0_AC_38	NC	CS_N1
DDR1_AC_0	CKEA0	CKE0
DDR1_AC_1	CKEA1	CKE1
DDR1_AC_2	CSA0	CS_N0
DDR1_AC_3	CSA1	NC
DDR1_AC_4	CLKA_T	BG0
DDR1_AC_5	CLKA_C	A10
DDR1_AC_6	NC	A8
DDR1_AC_7	NC	A11
DDR1_AC_8	CAA2	WE_N
DDR1_AC_9	CAA3	BG1



Pin Name	LPDDR4/LPDDR4X	DDR4
DDR1_AC_10	CAA1	A12
DDR1_AC_11	CAA0	RAS_N
DDR1_AC_12	CAA5	A3
DDR1_AC_13	CAA4	CAS_N
DDR1_AC_14	NC	A7
DDR1_AC_15	NC	A13
DDR1_AC_20	CKEB0	CLK1_T
DDR1_AC_21	CKEB1	CLK1_C
DDR1_AC_22	CSB1	NC
DDR1_AC_23	CSB0	NC
DDR1_AC_24	CLKB_T	BA1
DDR1_AC_25	CLKB_C	A4
DDR1_AC_26	NC	BA0
DDR1_AC_28	CAB1	ACT_N
DDR1_AC_29	CAB3	A0
DDR1_AC_30	CAB5	A9
DDR1_AC_31	CAB2	A1
DDR1_AC_32	CAB4	NC
DDR1_AC_33	CAB0	A6
DDR1_AC_34	NC	A2
DDR1_AC_35	NC	A5
DDR1_AC_36	NC	ODT0
DDR1_AC_37	NC	ODT1
DDR1_AC_38	NC	CS_N1

## 4.5 Signal Description

**Table 4-16 SD Card Interface Signal Description**

Signal Name	Type	Description
SDCARD_D0	DIO	SD Card data bus bit 0 signal
SDCARD_D1	DIO	SD Card data bus bit 1 signal
SDCARD_D2	DIO	SD Card data bus bit 2 signal
SDCARD_D3	DIO	SD Card data bus bit 3 signal
SDCARD_CLK	DO	SD Card clock signal
SDCARD_CMD	DIO	SD Card command signal

**Table 4-17 SDIO Interface Signal Description**

Signal Name	Type	Description
SDIO_D0	DIO	SDIO data bus bit 0 signal
SDIO_D1	DIO	SDIO data bus bit 1 signal
SDIO_D2	DIO	SDIO data bus bit 2 signal
SDIO_D3	DIO	SDIO data bus bit 3 signal
SDIO_CLK	DO	SDIO clock signal
SDIO_CMD	DIO	SDIO command signal

**Table 4-18 Clock Interface Signal Description**

Signal Name	Type	Description
CLK12M_24M	DO	12MHz/24MHZ clock output
CLK25M	DO	25MHz clock output

**Table 4-19 UART Interface Signal Description**

Signal Name	Type	Description
UART_D_TX	DO	UART Port D data output
UART_D_RX	DI	UART Port D data input
UART_D_CTS	DI	UART Port D Clear To Send Signal
UART_D_RTS	DO	UART Port D Ready To Send Signal
UART_E_TX	DO	UART Port E data output
UART_E_RX	DIO	UART Port E data input
UART_E_CTS	DIO	UART Port E Clear To Send Signal
UART_E_RTS	DIO	UART Port E Ready To Send Signal
UART_A_TX	DIO	UART Port A data output
UART_A_RX	DIO	UART Port A data input
UART_A_CTS	DIO	UART Port A Clear To Send Signal
UART_A_RTS	DIO	UART Port A Ready To Send Signal
UART_AO_A_TX	DIO	UART Port A data output in AO domain
UART_AO_A_RX	DIO	UART Port A data input in AO domain
UART_AO_B_TX	DIO	UART Port B data output in AO domain
UART_AO_B_RX	DIO	UART Port B data input in AO domain
UART_AO_B_CTS	DIO	UART Port B Clear To Send Signal in AO domain
UART_AO_B_RTS	DIO	UART Port B Ready To Send Signal in AO domain
UART_C_TX	DIO	UART Port C data output
UART_C_RX	DIO	UART Port C data input
UART_C_CTS	DIO	UART Port C Clear To Send Signal

Signal Name	Type	Description
UART_C_RTS	DIO	UART Port C Ready To Send Signal
UART_F_TX	DIO	UART Port F data output
UART_F_RX	DIO	UART Port F data input
UART_F_CTS	DIO	UART Port F Clear To Send Signal
UART_F_RTS	DIO	UART Port F Ready To Send Signal

**Table 4-20 ISO7816 Interface Signal Description**

Signal Name	Type	Description
ISO7816_DATA	DIO	ISO7816 data signal
ISO7816_CLK	DO	ISO7816 clock signal

**Table 4-21 TS In Interface Signal Description**

Signal Name	Type	Description
TSIN_A_DIN0	DI	Serial TS input port A data
TSIN_A_CLK	DI	TS input port A clock
TSIN_A_SOP	DI	TS input port A start of stream signal
TSIN_A_VALID	DI	TS input port A date valid signal
TSIN_B_DIN0	DI	Serial/Parallel TS input port B data 0
TSIN_B_DIN1	DI	Parallel TS input port B data 1
TSIN_B_DIN2	DI	Parallel TS input port B data 2
TSIN_B_DIN3	DI	Parallel TS input port B data 3
TSIN_B_DIN4	DI	Parallel TS input port B data 4
TSIN_B_DIN5	DI	Parallel TS input port B data 5
TSIN_B_DIN6	DI	Parallel TS input port B data 6
TSIN_B_DIN7	DI	Parallel TS input port B data 7
TSIN_B_FAIL	DI	TS input port B fail signal
TSIN_B_CLK	DI	TS input port B clock
TSIN_B_SOP	DI	TS input port B start of stream signal
TSIN_B_VALID	DI	TS input port B date valid signal
TSIN_C_DIN0	DI	Serial TS input port C data
TSIN_C_CLK	DI	TS input port C clock
TSIN_C_SOP	DI	TS input port C start of stream signal
TSIN_C_VALID	DI	TS input port C date valid signal
TSIN_D_DIN0	DI	Serial TS input port D data
TSIN_D_CLK	DI	TS input port D clock

Signal Name	Type	Description
TSIN_D_SOP	DI	TS input port D start of stream signal
TSIN_D_VALID	DI	TS input port D data valid signal

**Table 4-22 PWM Interface Signal Description**

Signal Name	Type	Description
PWM_A	DO	PWM channel A output signal
PWM_B	DO	PWM channel B output signal
PWM_C	DO	PWM channel C output signal
PWM_D	DO	PWM channel D output signal
PWM_E	DO	PWM channel E output signal
PWM_F	DO	PWM channel F output signal
PWM_AO_A	DO	PWM A output signal in Always On domain
PWM_AO_B	DO	PWM B output signal in Always On domain
PWM_AO_C / PWMAO_C_HI_Z	DO	PWM C output signal in Always On domain, or extended HiZ function of PWM_AO_C
PWM_AO_D	DO	PWM D output signal in Always On domain
PWM_AO_E	DO	PWM E output signal in Always On domain
PWM_AO_F	DO	PWM F output signal in Always On domain
PWM_AO_G / PWM_AO_G_HI_Z	DO	PWM G output signal in Always On domain, or extended HiZ function of PWM_AO_G
PWM_AO_H	DO	PWM H output signal in Always On domain
PWM_VS	DO	

**Table 4-23 I2C Interface Signal Description**

Signal Name	Type	Description
I2CM_E_SCL	DO	I2C bus group E clock output, master mode
I2CM_E_SDA	DIO	I2C bus group E data input/output, master mode
I2CM_A_SCL	DO	I2C bus group E clock output, master mode
I2CM_A_SDA	DIO	I2C bus group E data input/output, master mode
I2CM_B_SCL	DO	I2C bus group E clock output, master mode
I2CM_B_SDA	DIO	I2C bus group E data input/output, master mode
I2CM_C_SCL	DO	I2C bus group E clock output, master mode
I2CM_C_SDA	DIO	I2C bus group E data input/output, master mode
I2CM_D_SCL	DO	I2C bus group E clock output, master mode
I2CM_D_SDA	DIO	I2C bus group E data input/output, master mode
I2CM_AO_A_SCL	DIO	I2C bus group AO data input/output, master mode
I2CM_AO_A_SDA	DIO	I2C bus group AO data input/output, master mode

Signal Name	Type	Description
I2CM_AO_B_SCL	DIO	I2C bus group AO data input/output, master mode
I2CM_AO_B_SDA	DIO	I2C bus group AO data input/output, master mode
I2CS_AO_SCL	DI	I2C bus group AO data input, slave mode
I2CS_AO_SDA	DIO	I2C bus group AO data input, output slave mode

**Table 4-24 eMMC Interface Signal Description**

Signal Name	Type	Description
EMMC_D0	DIO	eMMC/NAND data bus bit 0 signal
EMMC_D1	DIO	eMMC/NAND data bus bit 1 signal
EMMC_D2	DIO	eMMC/NAND data bus bit 2 signal
EMMC_D3	DIO	eMMC/NAND data bus bit 3 signal
EMMC_D4	DIO	eMMC/NAND data bus bit 4 signal
EMMC_D5	DIO	eMMC/NAND data bus bit 5 signal
EMMC_D6	DIO	eMMC/NAND data bus bit 6 signal
EMMC_D7	DIO	eMMC/NAND data bus bit 7 signal
EMMC_CLK	DO	eMMC clock signal
EMMC_CMD	DIO	eMMC command signal
EMMC_DS	DI	eMMC data strobe

**Table 4-25 HDMI Interface Signal Description**

Signal Name	Type	Description
HDMIRX_A_HPD	OD_5V	HDMI RX port A hot plug in signal output
HDMIRX_A_5VDET	OD_5V	HDMI RX port A 5V power detection
HDMIRX_A_SDA	OD_5V	HDMI RX port A DDC_I2C interface data signal
HDMIRX_A_SCL	OD_5V	HDMI RX port A DDC_I2C interface clock signal
HDMIRX_B_HPD	OD_5V	HDMI RX port B hot plug in signal output
HDMIRX_B_5VDET	OD_5V	HDMI RX port B 5V power detection
HDMIRX_B_SDA	OD_5V	HDMI RX port B DDC_I2C interface data signal
HDMIRX_B_SCL	OD_5V	HDMI RX port B DDC_I2C interface clock signal
HDMIRX_C_HPD	OD_5V	HDMI RX port C hot plug in signal output
HDMIRX_C_5VDET	OD_5V	HDMI RX port C 5V power detection
HDMIRX_C_SDA	OD_5V	HDMI RX port C DDC_I2C interface data signal
HDMIRX_C_SCL	OD_5V	HDMI RX port C DDC_I2C interface clock signal
HDMITX_SDA	OD_5V	HDMI TX DDC_I2C interface data signal
HDMITX_SCL	OD_5V	HDMI TX DDC_I2C interface clock signal
HDMITX_HPD_IN	OD_5V	HDMI TX hot-plug in signal input

Signal Name	Type	Description
CEC_A	OD_5V	Customer Electronics Control signal
CEC_B	OD_5V	2nd pin of Customer Electronics Control signal

**Table 4-26 SPIF Interface Signal Description**

Signal Name	Type	Description
SPIF_CS	DO	SPIF chip select
SPIF_CLK	DO	SPIF Serial Clock
SPIF_MO	DIO	SPIF 1bit mode Output, 2/4 bit mode data I/O 0
SPIF_MI	DIO	SPIF 1bit mode Input, 2/4 bit mode data I/O 1
SPIF_WP	DIO	SPIF Write protection output, 4 bit mode data I/O 2
SPIF_HOLD	DIO	SPIF bus hold output, 4 bit mode data I/O 3

**Table 4-27 SPDIF Interface Signal Description**

Signal Name	Type	Description
SPDIF_IN	DI	SPDIF input signal
SPDIF_OUT	DO	SPDIF output signal

**Table 4-28 PCIE Interface Signal Description**

Signal Name	Type	Description
PCIECK_REQN	DI	PCIE clock request input

**Table 4-29 SPI Interface Signal Description**

Signal Name	Type	Description
SPI_A_MOSI	DIO	SPI master output, slave input A
SPI_A_MISO	DIO	SPI master input, slave output A
SPI_A_SCLK	DIO	SPI clock A
SPI_A_SS0	DIO	SPI slave select 0 A
SPI_A_SS1	DIO	SPI slave select 1 A
SPI_A_SS2	DIO	SPI slave select 2 A
SPI_B_MOSI	DIO	SPI master output, slave input B
SPI_B_MISO	DIO	SPI master input, slave output B
SPI_B_SCLK	DIO	SPI clock B
SPI_B_SS0	DIO	SPI slave select 0 B
SPI_B_SS1	DIO	SPI slave select 1 B
SPI_B_SS2	DIO	SPI slave select 2 B
SPI_C_MOSI	DIO	SPI master output, slave input C
SPI_C_MISO	DIO	SPI master input, slave output C

Signal Name	Type	Description
SPI_C_SCLK	DIO	SPI clock C
SPI_C_SS0	DIO	SPI slave select 0 C
SPI_C_SS1	DIO	SPI slave select 1 C
SPI_C_SS2	DIO	SPI slave select 2 C
SPI_D_MOSI	DIO	SPI master output, slave input D
SPI_D_MISO	DIO	SPI master input, slave output D
SPI_D_SCLK	DIO	SPI clock D
SPI_D_SS0	DIO	SPI slave select 0 D
SPI_E_MOSI	DIO	SPI master output, slave input E
SPI_E_MISO	DIO	SPI master input, slave output E
SPI_E_SCLK	DIO	SPI clock E
SPI_E_SS0	DIO	SPI slave select 0 E
SPI_F_MOSI	DIO	SPI master output, slave input F
SPI_F_MISO	DIO	SPI master input, slave output F
SPI_F_SCLK	DIO	SPI clock F
SPI_F_SS0	DIO	SPI slave select 0 F

Table 4-30 Remote Interface Signal Description

Signal Name	Type	Description
IR_IN	DI	IR remote control input
IR_OUT	DO	IR remote control output

Table 4-31 Time Division Multiplexing Signal Description

Signal Name	Type	Description
MCLK_1	DO	Master clock output 1, for I2S master mode
MCLK_2	DO	Master clock output 2, for I2S master mode
TDM_D0	DIO	Data input/output 0 of TDM
TDM_D1	DIO	Data input/output 1 of TDM
TDM_D2	DIO	Data input/output 2 of TDM
TDM_D3	DIO	Data input/output 3 of TDM
TDM_D4	DIO	Data input/output 4 of TDM
TDM_D5	DIO	Data input/output 5 of TDM
TDM_D6	DIO	Data input/output 6 of TDM
TDM_D7	DIO	Data input/output 7 of TDM
TDM_D8	DIO	Data input/output 8 of TDM
TDM_D9	DIO	Data input/output 9 of TDM

Signal Name	Type	Description
TDM_D10	DIO	Data input/output 10 of TDM
TDM_D11	DIO	Data input/output 11 of TDM
TDM_D12	DIO	Data input/output 12 of TDM
TDM_D13	DIO	Data input/output 13 of TDM
TDM_D14	DIO	Data input/output 14 of TDM
TDM_D15	DIO	Data input/output 15 of TDM
TDM_SCLK0	DIO	Bit clock output of TDM
TDM_FS0	DIO	Frame sync output of TDM (Word clock of I2S)
TDM_SCLK1	DIO	Bit clock output of TDM
TDM_FS1	DIO	Frame sync output of TDM (Word clock of I2S)
TDM_SCLK2	DIO	Bit clock output of TDM
TDM_FS2	DIO	Frame sync output of TDM (Word clock of I2S)
TDM_SCLK3	DIO	Bit clock output of TDM
TDM_FS3	DIO	Frame sync output of TDM (Word clock of I2S)

**Table 4-32 PDM Signal Description**

Signal Name	Type	Description
PDM_DIN0	DI	PDM input data 0 signal
PDM_DIN1	DI	PDM input data 1 signal
PDM_DIN2	DI	PDM input data 2 signal
PDM_DIN3	DI	PDM input data 3 signal
PDM_DCLK	DO	PDM output clock signal

**Table 4-33 JTAG Interface Signal Description**

Signal Name	Type	Description
JTAG_A_TDO	DO	JTAG data output channel A
JTAG_A_TDI	DI	JTAG data input channel A
JTAG_A_TMS	DI	JTAG Test mode select input channel A
JTAG_A_CLK	DO	JTAG Test clock input channel A
JTAG_B_TDO	DO	JTAG data output channel B
JTAG_B_TDI	DI	JTAG data input channel B
JTAG_B_TMS	DI	JTAG Test mode select input channel B
JTAG_B_CLK	DO	JTAG Test clock input channel B



**Table 4-34 Ethernet Interface Signal Description**

Signal Name	Type	Description
ETH_LINK_LED	DO	Ethernet link LED indicator
ETH_ACT_LED	DO	Ethernet active LED indicator
ETH_RGMII_RX_CLK	DI	Ethernet RGMII interface receive clock input
ETH_RGMII_TX_CLK	DIO	Ethernet RGMII transmit clock
ETH_TXEN	DIO	Ethernet RMII/RGMII Interface transmit enable
ETH_TXD3_RGMII	DIO	Ethernet RGMII interface transmit data 3
ETH_TXD2_RGMII	DIO	Ethernet RGMII interface transmit data 2
ETH_TXD1	DIO	Ethernet RMII/RGMII interface transmit data 1
ETH_TXD0	DIO	Ethernet RMII/RGMII interface transmit data 0
ETH_RX_DV	DI	Ethernet RMII/RGMII interface receive data valid signal
ETH_RXD3_RGMII	DI	Ethernet RGMII interface receive data 3
ETH_RXD2_RGMII	DI	Ethernet RGMII interface receive data 2
ETH_RXD1	DI	Ethernet RMII/RGMII interface receive data 1
ETH_RXD0	DI	Ethernet RMII/RGMII interface receive data 0
ETH_MDIO	DIO	Ethernet SMI interface management data input/output
ETH_MDC	DO	Ethernet SMI interface management clock

**Table 4-35 Other Signal Description**

Signal Name	Type	Description
RTC_CLK_OUT	DO	RTC32.768 KHz clock output
WD_RSTO	DO	WD_RSTO Watchdog output
MIC_MUTE_KEY	DI	MIC MUTE KEY detect signal
MIC_MUTE_LED	DIO	MIC MUTE state output signal
HSYNC	DIO	Horizontal synchronized signal
VSYNC	DIO	Vertical synchronized signal
3D_SYNC_OUT	DIO	3D sync signal for 3D Pnale
VX1_A_HTPDN	DI	VX1 channel A hot plug detect
VX1_B_HTPDN	DI	VX1 channel B hot plug detect
VX1_A_LOCKN	DI	VX1 channel A lock detect
VX1_B_LOCKN	DI	VX1 channel B lock detect
eDP_A_HPD	DI	eDP panel A hot plug detect
eDP_B_HPD	DI	eDP panel B hot plug detect

**Table 4-36 Other Signal Description**

Signal Name	Type	Description
GEN_CLK_OUT	DO	General clock output, for debug

## 5 Operating Conditions

### 5.1 Absolute Maximum Ratings

The table below gives the absolute maximum ratings. Exposure to stresses beyond those listed in this table may result in permanent device damage, unreliability or both.

**Table 5-1 Absolute Maximum Ratings**

Characteristic	Value	Unit
VDDCPU_A/B Supply Voltage	1.15	V
VDD_EE Supply Voltage	1.0	V
VDD_GPU Supply Voltage	1.0	V
VDD_DDR Supply Voltage	1.0	V
VDDQ Supply Voltage	1.4	V
VDDQLP Supply Voltage	1.4	V
AVDD_DDR0PLL	1.98	V
AVDD_DDR1PLL	1.98	V
1.8V Supply Voltage	1.98	V
3.3V Supply Voltage	3.63	V
Input voltage, $V_I$	-0.3 ~ VDDIO+0.3	V
Junction Temperature	125	°C

### 5.2 Recommended Operating Conditions

**Table 5-2 Recommended Operating Conditions**

Symbol	Parameter	Min.	Typ.	Max.	Unit
VDDCPU	Voltage for Cortex CPU	0.68 <sup>1</sup>	-	1.03 <sup>2</sup>	V
VDD_EE and other 0.8V domain	Voltage for core logic	0.77	0.83	0.9 <sup>2</sup>	V
VDD_GPU	Voltage for GPU	0.77	0.83	0.9 <sup>2</sup>	V
VDD_DDR <sup>6</sup>		0.77	0.80~0.84	0.9	
VDDQ	DDR IO Supply Voltage	1.06	-	1.26	V
VDDQLP <sup>5</sup>	DDR IO Supply Voltage	0.57	-	1.26	V
AVDD18	1.8V AVDD for HDMI, USB, SARADC, PCIe, AUDIO, MIPI_DSI, MIPI_CSI and ETHERNET phy.	1.74	1.80	1.89	V
VDD18_AO	1.8V VDD for XTAL, SARADC, efuse and IOVREF	1.74	1.80	1.89	V
AVDD_DDR0PLL and AVDD_DDR1PLL	Analog power supply for DDRPLL module	1.06	-	1.89	V
AVDD33	3.3V AVDD for USB and HDMI RX	3.15	3.3	3.45	V

Symbol	Parameter	Min.	Typ.	Max.	Unit
VDDIO	LV mode	1.71	1.80	1.89	V
	HV mode	3.0 <sup>3</sup>	3.3	3.45	V
T <sub>J</sub>	Operating Junction Temperature	0	-	105 <sup>4</sup>	°C
T <sub>A</sub>	Operating Ambient Temperature	0	-	70	°C



### Note

1. Minimal VDDCPU\_A/B voltage is for sleep mode while system runs at very low speed. Higher clock will need higher voltage. Considering the power supply may have 3% deviation, the minimal voltage in actual application should not be set to lower than min spec plus 0.02V.
2. Likewise, maximum VDDCPU\_A/B voltage in actual application should not be higher than max spec minus 0.03V. Voltage of VDDCPU\_A/B will affect CPU speed. Use lower voltage when CPU runs on lower speed to save power. Recommend to use +/-1.5% or higher precision DCDC.
3. GPIO cannot work if VDDIO voltage is out of the spec of LV / HV mode. GPIO output at HV mode will be weaker & max operating speed will be lower if VDDIO are design to 3.0V. Do not design VDDIO to lower than 3.0V in HV mode, recommend to use +/-1.5% or higher precision DCDC to supply power for VDDIO, actual voltage supplies to VDDIO (HV mode) should not be lower than 2.9V.
4. For operating temperature, good heat sink may be needed to guarantee T<sub>J</sub> < max spec.
5. When not in LPDDRx mode, the VDDQLP supply pin should be tied to the VDDQ supply.
6. The minimum and typical voltage of VDD\_DDR needs to be increased by 0.3V (Min 0.80V and Typ 0.83V) When the target rate is >3733Mbps for LPDDR4 and LPDDR4X mode.

## 5.3 Ripple Voltage Specifications

Table 5-3 Ripple Voltage Specifications

Power	Max Ripple	Unit	Test State
VDDCPU_A	40	+/-mV	Run APK StabilityTest
VDDCPU_B	40	+/-mV	Run APK StabilityTest
VDD_EE and other 0.8V domain	40	+/-mV	Play 4K video
VDD_GPU	70	+/-mV	Run APK Basemark ES 2.0 Taiji
DDR4 VDD_DDR	40	+/-mV	Kernel boot
LPDDR4/LPDDR4 VDD_DDR	32	+/-mV	Kernel boot
DDR4 VDDQ and AVDD_DDR0PLL, AVDD_DDR1PLL	60	+/-mV	Kernel boot
LPDDR4/LPDDR4X VDDQ and AVDD_DDR0PLL, AVDD_DDR1PLL	27.5	+/-mV	Kernel boot
LPDDR4 VDDQLP	45	+/-mV	Kernel boot
LPDDR4X VDDQLP	30	+/-mV	Kernel boot
AVDD18	30	+/-mV	Kernel boot

Power	Max Ripple	Unit	Test State
VDD18_AO_XTAL	30	+/-mV	Kernel boot
AVDD33	50	+/-mV	WIFI SCAN
VDDIO LV	60	+/-mV	Kernel boot
VDDIO HV	60	+/-mV	WIFI SCAN



### Note

The ripple specifications are for reference only. Customers should perform pressure / performance / reliability tests (high / low temperature tests, wet / heat tests, functional tests, etc.) on the product to confirm system stability.

## 5.4 Thermal Resistance

Jedec 2P2S board 101.5mm\*114.5mm, natural convection, ambient temperature 25°C.

Symbol	Parameter	Value(°C/ Watt)	Air Flow(m/s)
$\Theta_{JA}$	Package junction-to-ambiance thermal resistance in nature convection	12.6	0
$\Theta_{JB}$	Package junction-to-pcb thermal resistance in nature convection	4.0	0
$\Theta_{JC}$	Package junction-to-case thermal resistance in nature convection	0.1	0



### Note

- Due to the thinness of the SOC, DRAM or capacitors placed close to SOC may prevent heat-sink touching SOC top side. A special convex shape heatsink is recommended.
- For more information, check the following JEDEC standards:
  - JESD51-2A: Integrated Circuits Thermal Test Method Environmental Conditions - Natural Convection (Still Air)
  - JESD51-8: Integrated Circuit Thermal Test Method Environmental Conditions -Junction-to-Board
  - JESD51-12: Guidelines for Reporting and Using Electronic Package Thermal Information
- m/s = meters per second

## 5.5 DC Electrical Characteristics

## 5.5.1 Normal GPIO Specifications

Table 5-4 Normal GPIO Specifications (For DIO)

Symbol	Parameter	Min.	Typ.	Max.	Unit
$V_{IH}(VDDIO=3.3V)$	High-level input voltage	$IOVREF + 0.37$	-	$VDDIO + 0.3$	V
$V_{IL}(VDDIO=3.3V)$	Low-level input voltage	-0.3	-	$IOVREF - 0.23$	V
$V_{IH}(VDDIO=1.8V)$	High-level input voltage	$IOVREF/2 + 0.3$	-	$VDDIO + 0.3$	V
$V_{IL}(VDDIO=1.8V)$	Low-level input voltage	-0.3	-	$IOVREF/2 - 0.3$	V
$R_{PU/PD}$	Built-in pull up/down resistor	-	41K		ohm
$IoL/IoH(DS=0)^4$	GPIO driving capability	0.5	-	-	mA
$IoL/IoH(DS=1)$	GPIO driving capability	2.5	-	-	mA
$IoL/IoH(DS=2)$	GPIO driving capability	3	-	-	mA
$IoL/IoH(DS=3)$	GPIO driving capability	4 <sup>1</sup>	-	-	mA
VOH	Output high level with IoL/IoH loading	$VDDIO - 0.5$	-	-	V
VOL	Output low level with IoL/IoH loading	-	-	0.4	V



### Note

1. With Minimal IoL/IoH driving capability loading, IO is guaranteed to meet  $V_{ol} < 0.4V$  or  $V_{OH} > (VDDIO - 0.5V)$  spec.
2. Maximal GPIO loading is 6mA for application such as driving LED, which does not care about  $V_{ol}/V_{oh}$  spec. Please set DS=3 for such application.
3. VDD18\_AO supplies power to IOVREF.
4. Do not use this setting, it's too weak for most applications.

## 5.5.2 Open Drain GPIO Specifications (For DIO\_OD)

Symbol	Parameter	Min.	Typ.	Max.	Unit
$V_{IH}(OD5V)$	High-level input voltage	1.5		5.5	V
$V_{IL}(OD5V)$	Low-level input voltage	-0.3		0.8	V
$R_{PU/PD}$	No built-in pull up/down resistor on OD IO	-	-	-	ohm

Symbol	Parameter	Min.	Typ.	Max.	Unit
Io	OD IO driving low capability	4		6	mA
VOL	Output low level with min Io loading			0.4	V

 **Note**

1. With Minimal IoL driving capability loading, IO is guaranteed to meet Vol<0.4V spec
2. Maximal GPIO loading is 6mA for application such as driving LED, which does not care about Vol spec
3. The  $V_{iL}$  /  $V_{iH}$  of OD PAD is irrelevant to VDDIO voltage.
4. "OD 5V" means that in applications such as I2C, use a resistor greater than 1Kohm to pull it up to 5V. Do not connect the pad directly to the 5V power supply.

### 5.5.3 DDR4/LPDDR4/LPDDR4X SDRAM Specifications

**Table 5-5 Recommended Operating Conditions**

Symbol	Parameter	Min.	Typ.	Max.	Unit
VDDQ	PHY supply voltage (DDR4)	1.14	1.20	1.26	V
VDDQ <sup>1</sup>	PHY supply voltage (LPDDR4)	1.06	1.1	1.17	V
VDDQ <sup>1</sup>	PHY supply voltage (LPDDR4X)	1.06	1.1	1.17	V
VDDQLP	IO supply voltage (DDR4)	1.14	1.20	1.26	V
VDDQL-P <sup>2</sup>	IO supply voltage (LPDDR4)	1.06	1.1	1.17	V
VDDQL-P <sup>3</sup>	IO supply voltage (LPDDR4X)	0.57	0.6	0.65	V
Vref	Input reference supply voltage	0.49*VDDQ	0.5*VDDQ	0.51*VDDQ	V

 **Note**

1. The minimum and typical voltage of VDDQ needs to be increased by 0.3V(Min 1.09V and Typ 1.13V) When the target rate is >3733Mbps for LPDDR4 and LPDDR4X mode.
2. The minimum and typical voltage of VDDQLP needs to be increased by 0.3V(Min 1.09V and Typ 1.13V) When the target rate is >3733Mbps for LPDDR4 mode.
3. The minimum and typical voltage of VDDQLP needs to be increased by 0.3V(Min 0.60V and Typ 0.63V) When the target rate is >3733Mbps for LPDDR4X mode.

**Table 5-6 Address and command DC specifications – DDR4 Mode**

Symbol	Parameter	Min.	Typ.	Max.	Unit
VdIVW_total	Rx Mask voltage-p-p total			136	mv
VOH	DC output logic high	0.9*VDDQ			V
VOL	DC output logic low			0.1*VDDQ	V
RTT	Input termination resistance to VDDQ	200	240	280	ohm
		100	120	140	
		67	80	93	
		50	60	70	
		42	48	56	
		34	40	46	
		28	34	40	

**Table 5-7 Address and command DC Specifications – LPDDR4 Mode**

Symbol	Parameter	Min.	Typ.	Max.	Unit
VOH	DC output logic high	0.9*VDDQ	-	-	V
VOL	DC output logic low	-	-	0.1*VDDQ	V
RTT	Input termination resistance to VDDQ	216	240	264	ohm
		108	120	132	
		72	80	88	
		54	60	66	
		43.2	48	52.8	
		36	40	44	

**Table 5-8 Address and command DC Specifications – LPDDR4X Mode**

Symbol	Parameter	Min.	Typ.	Max.	Unit
VOH	DC output logic high	0.8*VDDQ	-	-	V
VOL	DC output logic low	-	-	0.2*VDDQ	V
RTT	Input termination resistance to VDDQ	216	240	264	ohm
		108	120	132	
		72	80	88	
		54	60	66	
		43.2	48	52.8	
		36	40	44	

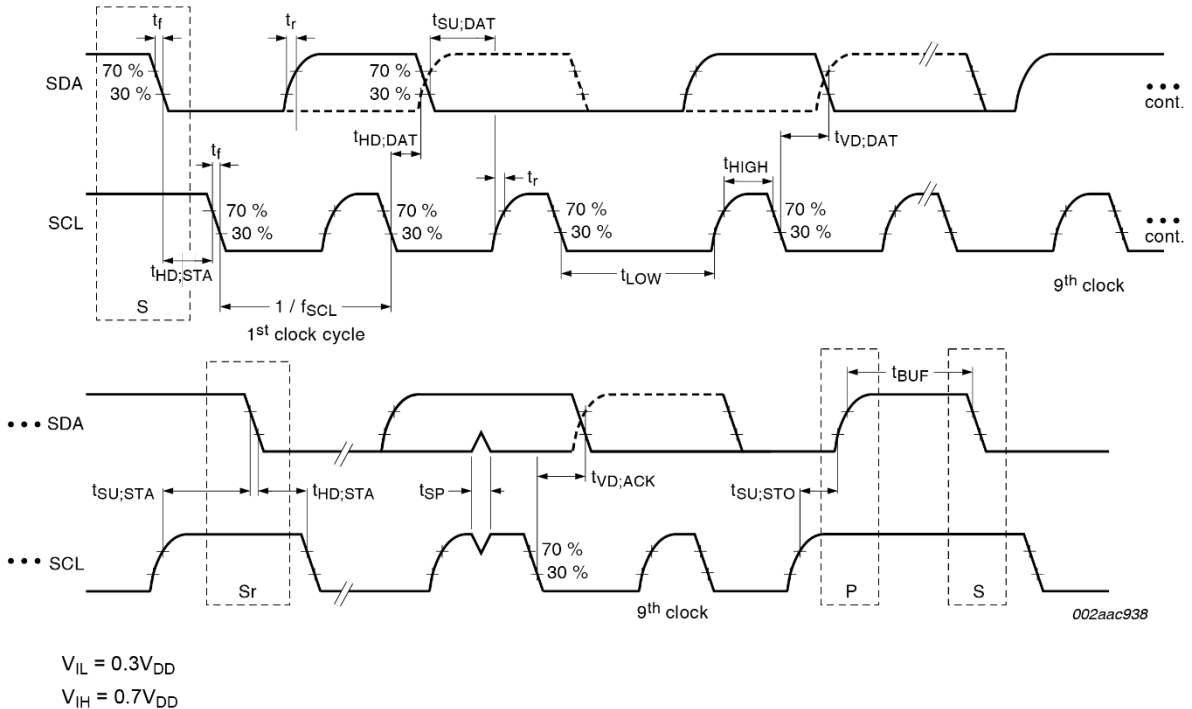
## 5.6 Timing Information



### 5.6.1 I2C Timing Specification

The I2C master interface Fast/Standard mode timing specifications are shown below.

**Figure 5-1 I2C Interface Timing Diagram, FS mode**



**Table 5-9 I2C Interface Timing Specification, SF mode**

Symbol	Parameter	Standard-mode		Fast-mode		Unit
		Min.	Max	Min	Max	
$t_R$	Rise time of SDA and SCL signals	-	1000	-	300	ns
$t_F$	Fall time of SDA and SCL signals	-	300	-	300	ns
$f_{SCL}$	SCL clock frequency	-	100	-	400	KHz
$t_{LOW}$	LOW period of the SCL clock	4.7	-	1.3	-	$\mu$ s
$t_{HIGH}$	HIGH period of the SCL clock	4	-	0.6	-	$\mu$ s
$t_{Su;STA}$	Setup time for START	4.7	-	0.6	-	$\mu$ s
$t_{Su;DAT}$	Setup time for SDA	250	-	100	-	ns
$t_{Su;STO}$	Setup time for STOP	4	-	0.6	-	$\mu$ s

Symbol	Parameter	Standard-mode		Fast-mode		Unit
		Min.	Max	Min	Max	
tHd;STA	Hold time for START	4	-	0.6	-	μs
tHd;DAT	Hold time for SDA	0	3.45	0	0.9	μs
tBuf	Bus free time between stop and start	4.7	-	1.3	-	μs

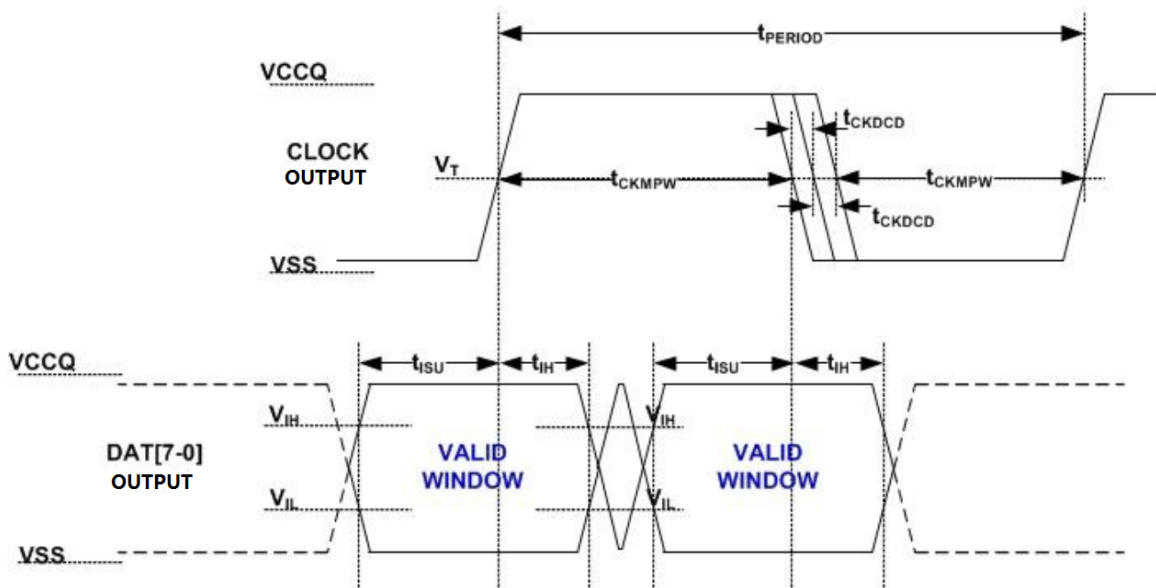
**Note**

Open drain does not support driver strength adjustment.

### 5.6.2 EMMC/SD Timing Specification

Timing specification for EMMC and SDIO are shown as below.

**Figure 5-2 EMMC HS400 Data Output Timing**



NOTE V<sub>T</sub> = 50% of V<sub>CCQ</sub>, indicates clock reference point for timing measurements.

**Table 5-10 HS400 Timing Specification**

Symbol	Parameter	Min	Max	Unit
tPERIOD	Cycle time data transfer mode	5	-	ns
SR	Slew rate	1.125	-	V/ns
tCKDCCD	Duty cycle distortion	0	0.3	ns
tCKMPW	Minimum pulse width	2.2	-	ns
tISU	input set-up time	1.4	-	ns
tIH	input hold time	0.8	-	ns

Symbol	Parameter	Min	Max	Unit
tISUddr	input set-up time	0.4	-	ns
tIHddr	input hold time	0.4	-	ns

Figure 5-3 EMMC HS200 Data Output Timing

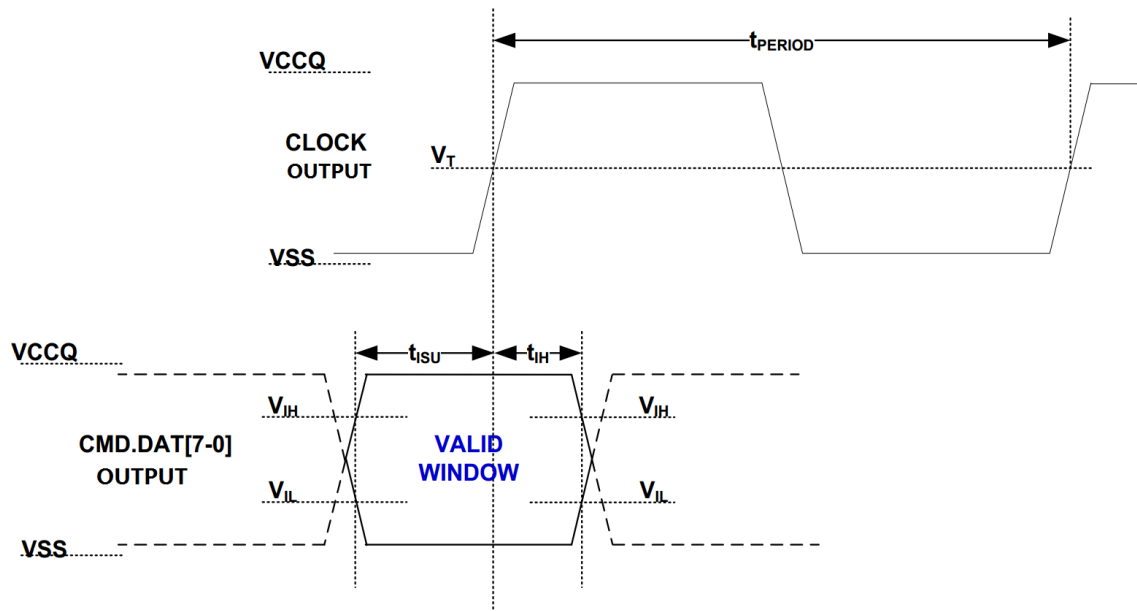


Table 5-11 HS200 Timing Specification

Symbol	Parameter	Min	Max	Unit
tPERIOD	Cycle time data transfer mode	5	-	ns
tISU	output set-up time	1.4	-	ns
tIH	output hold time	0.8	-	ns

Figure 5-4 EMMC HS400 Data Input Timing

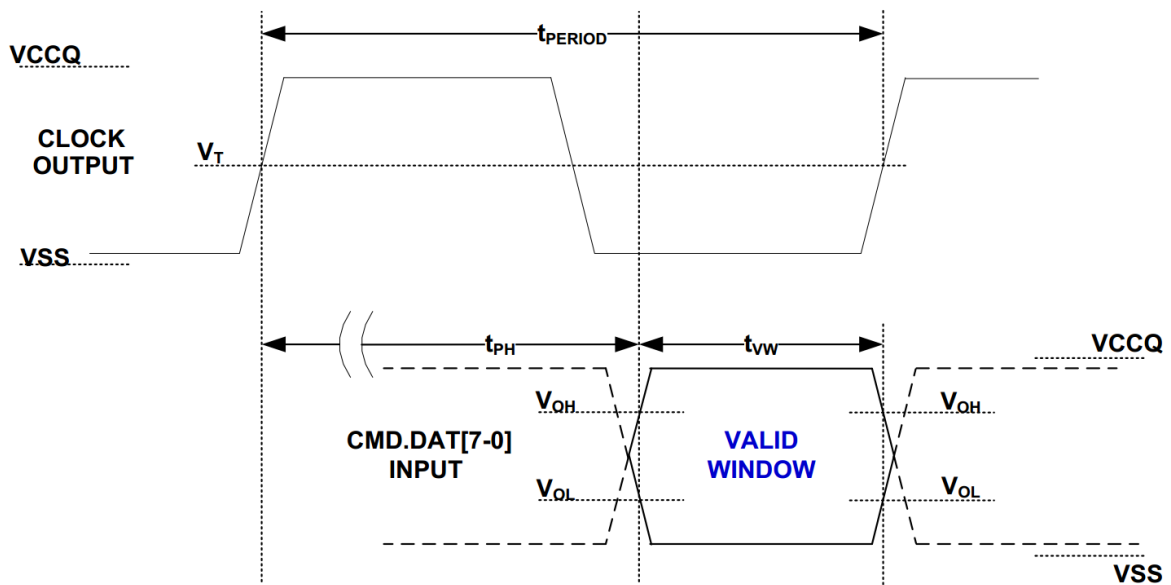


Table 5-12 HS400 Data Input Timing Specification

Symbol	Parameter	Min	Max	Unit
tPERIOD	Cycle time data transfer mode	5	-	ns
SR	Slew rate	1.125	-	V/ns
tCKDCD	Duty cycle distortion	0	0.2	ns
tCKMPW	Minimum pulse width	2	-	ns
tRQ	Input skew	-	0.4	ns
tRQH	input hold skew	-	0.4	ns

Figure 5-5 EMMC HS200 Data Input Timing

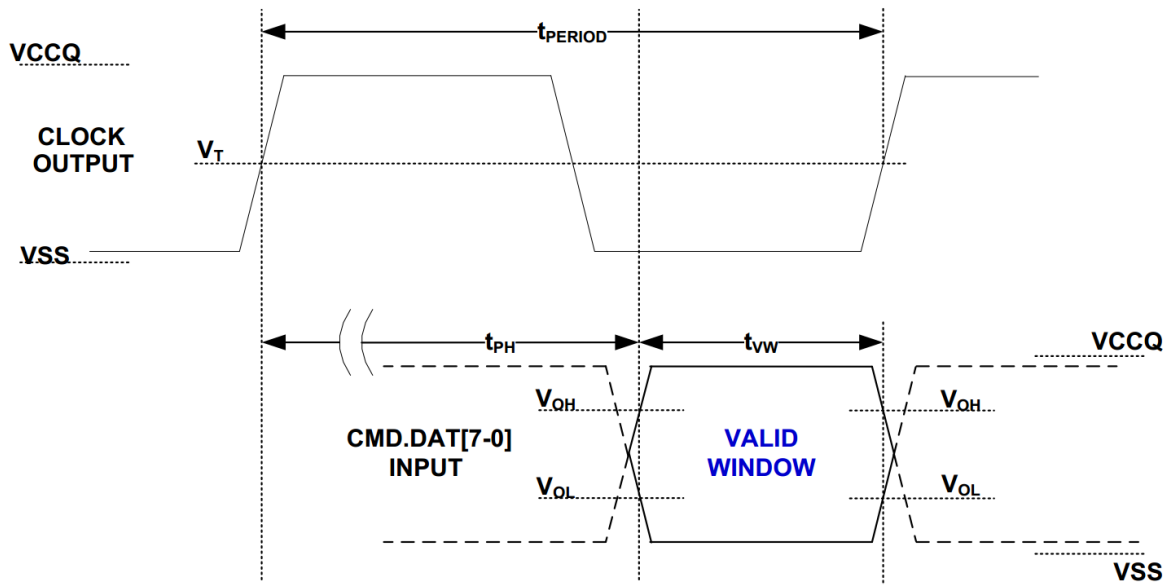
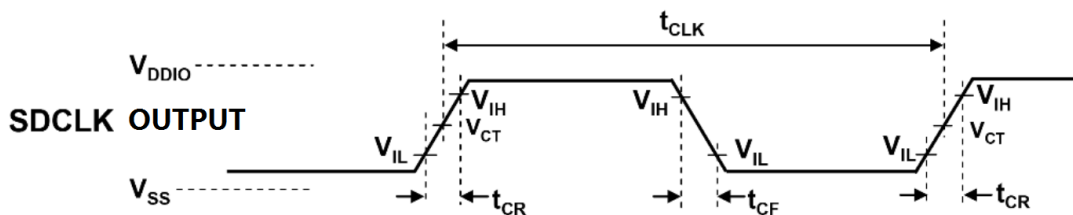


Table 5-13 HS200 Timing Specification

Symbol	Parameter	Min	Max	Unit
tPH	Device output momentary phase from CLK input to CMD or DAT line output. Does not include a longterm temperature drift.	0	2	UI
ΔTPH	Delay variation due to temperature change after tuning. Total allowable shift of output valid window (TVW) from last system Tuning procedure ΔTPH is 2600ps for ΔT from -25 °C to 125 °C during operation.	-350(ΔT=-20deg.C)	1550(ΔT=90deg.C)	ps
tVW	Valid Data Simple window	0.575	-	UI

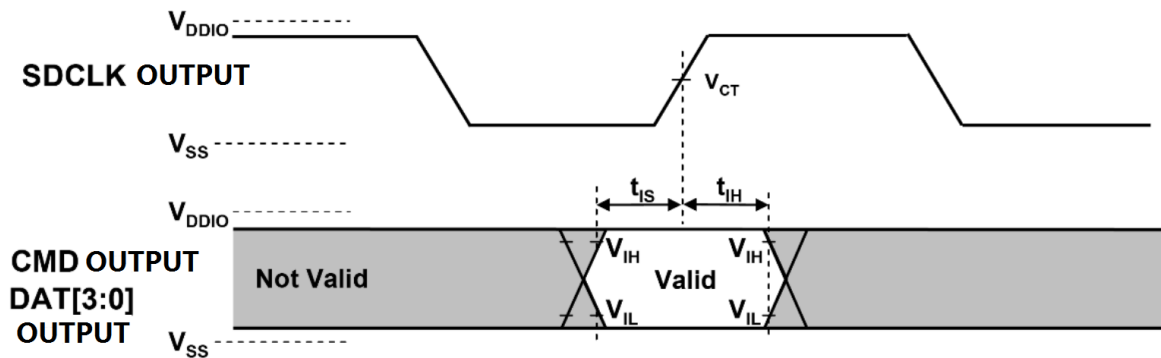
Figure 5-6 SDIO (SDR104) Clock Signal Timing Diagram



**Table 5-14 SDIO (SDR104) Clock Timing Specification**

Symbol	Parameter (SDR104 Mode)	Min	Max	Unit
tCLK	clock period Data Transfer Mode (PP)	4.8	-	ns
Duty	Clock Duty	30	70	%
tCR	clock rise time	-	0.96	ns
tCF	clock fall time	-	0.96	ns

**Figure 5-7 SDIO (SDR104) Output Timing Diagram**



**Table 5-15 SDIO (SDR104) Output Timing Specification**

Inputs CMD, DAT (referenced to CLK)				
Symbol	Parameter	Min	Max	Unit
tIS	input set-up time	1.4	-	ns
tIH	input hold time	0.8	-	ns

**Note**

SD card interface uses SDIO protocol.

**5.6.3 NAND Timing Specification**

Nand timing specifications are shown as below.

Figure 5-8 Async Waveform for Command/Address/Data Output Timing

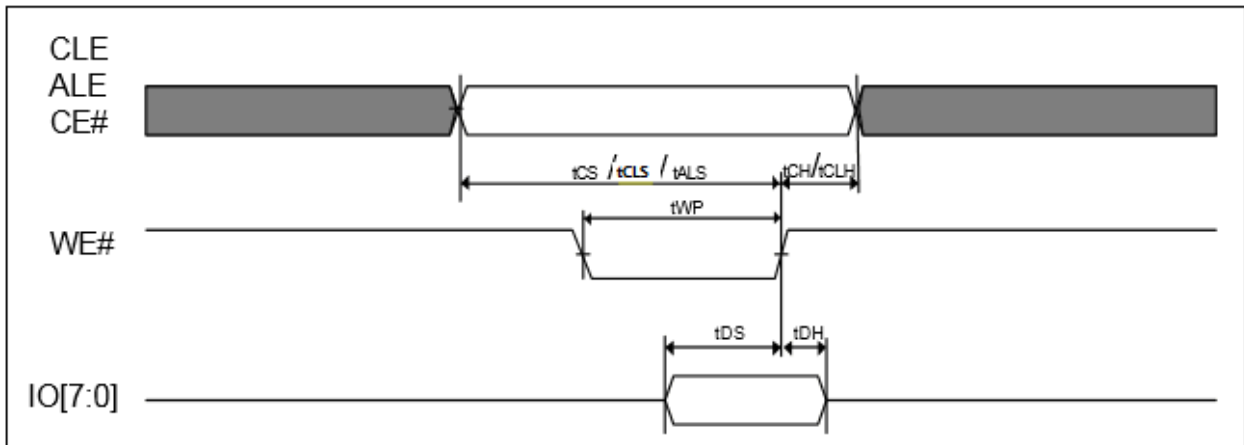


Figure 5-9 Async Waveform for Address Output Cycle

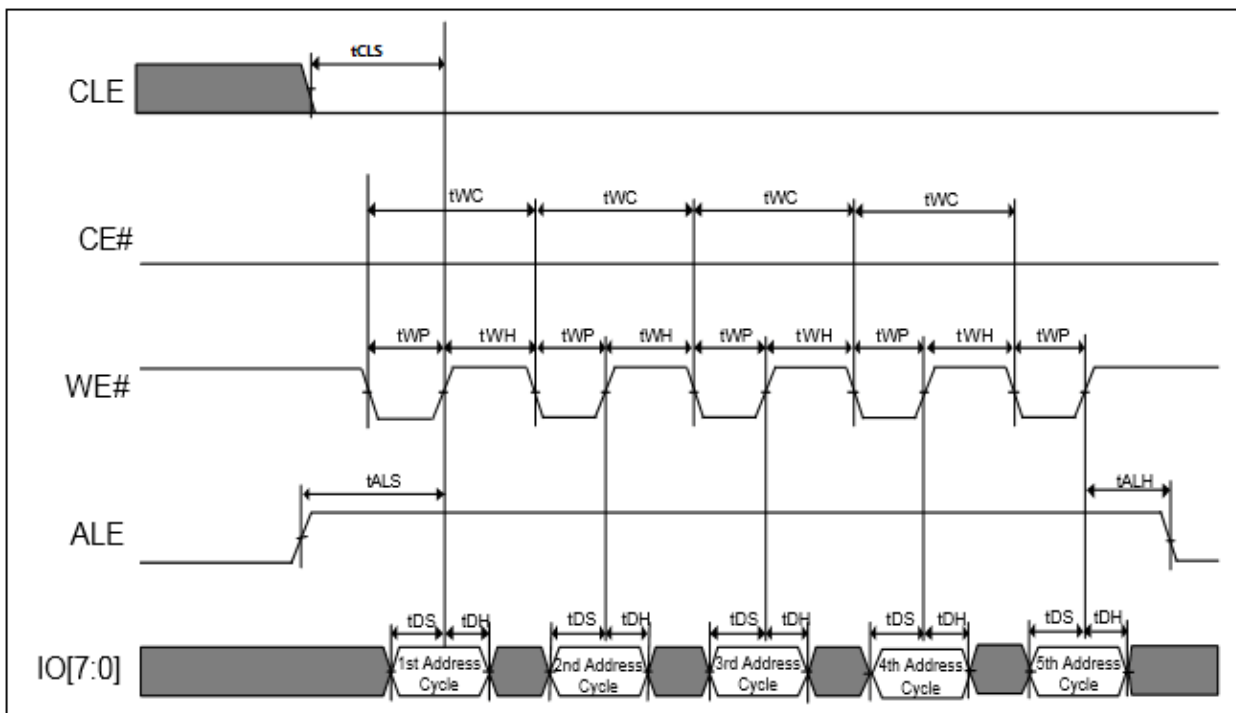


Figure 5-10 Async Waveform for Sequential Data Read Cycle(After Read)-EOD Mode

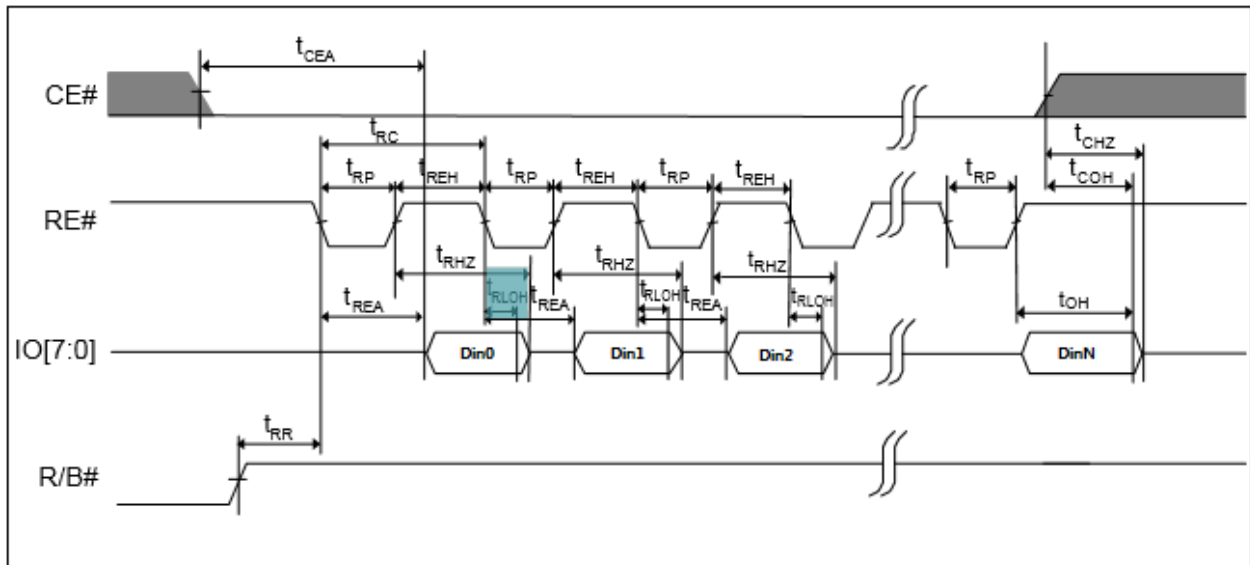


Table 5-16 Nand Timing Specifications

Symbol	Parameter (Asynchronous) (mode 5)	Min	Max	Unit
tCLS	CLE setup time	10	-	ns
tCLH	CLE hold time	5	-	ns
tALS	ALE setup	10	-	ns
tALH	ALE hold	5	-	ns
tDS	Data setup time	7	-	ns
tDH	Data hold time	5	-	ns
tWC	WE# cycle time	20	-	ns
tWP	WE# pulse width	10	-	ns
tWH	WE# high hold time	7	-	ns
tREA	RE# access time	-	16	ns
tOH	Data output hold time	15	-	ns
tRLOH	RE#-low to data hold time (EDO)	5	-	ns
tRP	RE# pulse width	10	-	ns
tREH	RE# high hold time	7	-	ns
tRC	RE# cycle time	20	-	ns



### 5.6.4 SPICC Timing Specification

Figure 5-11 SPICC Timing Diagram

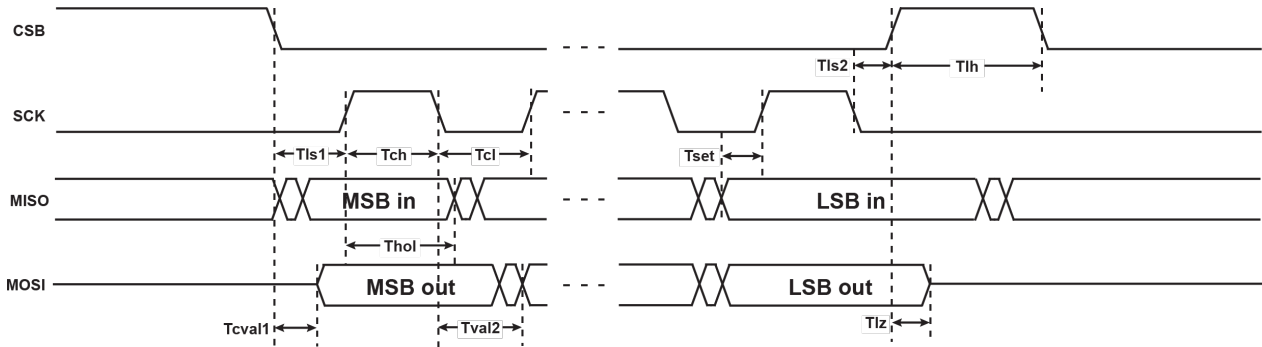


Table 5-17 SPICC Master Timing Specification

Symbol	Description	Min.	Max.	Unit
fCLK	Clock Frequency	1	80	MHz
TCH	Clock high time	5		ns
TCL	Clock low time	5		ns
TLS1	CS fall to First Rising CLK Edge	50		ns
TSET	Data input Setup Time	4		ns
THOL	Data input Hold Time	4		ns
TLH	Minimum idling time between transfers (minimum ss high time)	5		ns

### 5.6.5 SPIFC Timing Specification

Figure 5-12 SPIFC Serial Input Timing Diagram

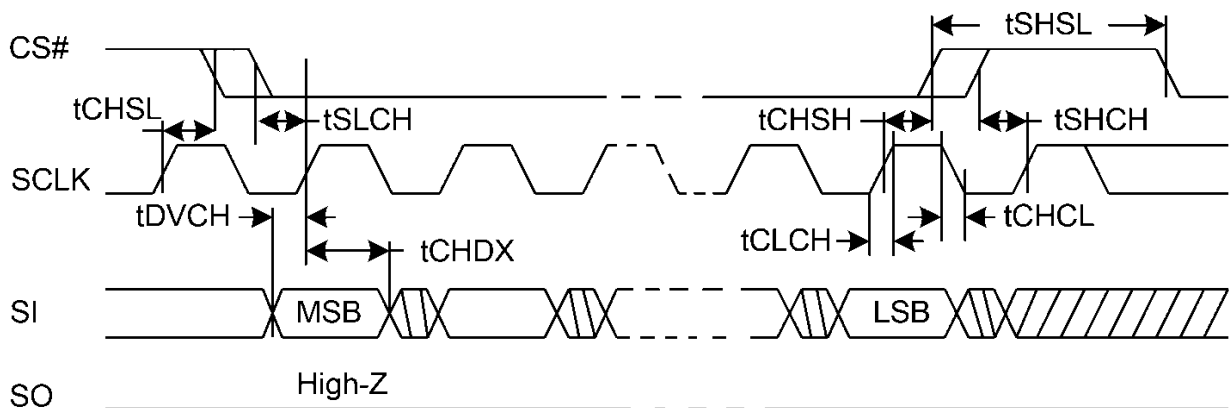


Figure 5-13 SPIFC Out Timing Diagram

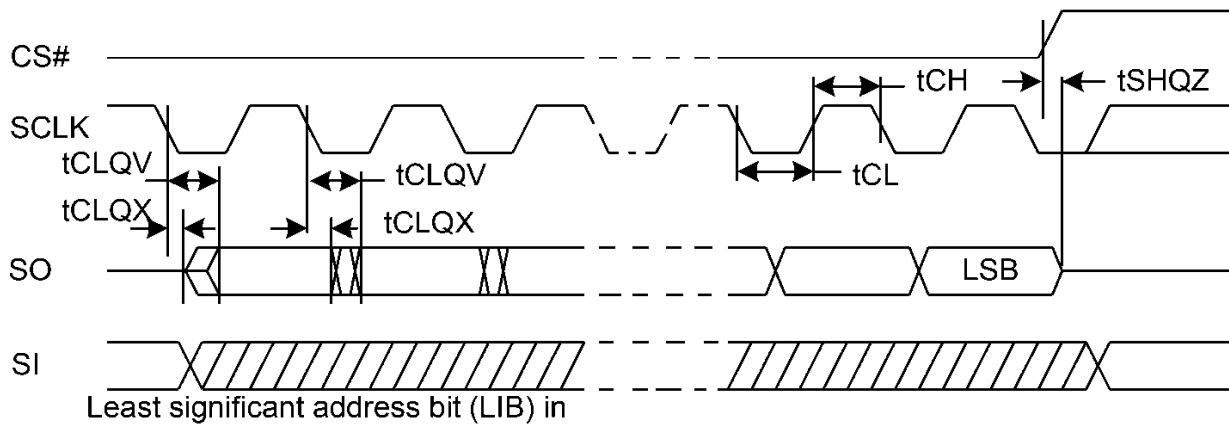
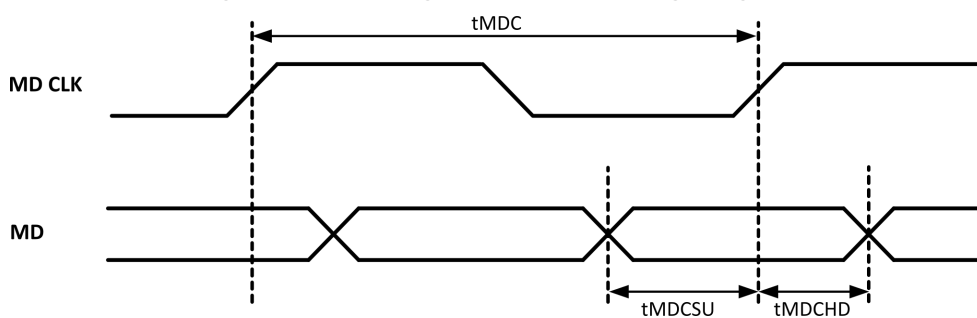


Table 5-18 SPIFC Master Timing Specification

Symbol	Parameter (Clock 41.7MHz)	Min	Max	Unit
fRCLK	Clock Frequency for READ instructions		50	Mhz
tCH	Clock High Time	8		ns
tCL	Clock Low Time	8		ns
tCLCH	Clock Rise Time (peak to peak)	0.1		V/ns
tCHCL	Clock Fall Time (peak to peak)	0.1		V/ns
tSLCH	CS# Active Setup Time (relative to SCLK)	4	-	ns
tCHSH	CS# Active Hold Time (relative to SCLK)	4	-	ns
tDVCH	Data In Setup Time	2	-	ns
tCHDX	Data In Hold Time	3	-	ns
tSHQZ	Output Disable Time (relative to CS#)		8	ns
tCLQV	Clock Low to Output Valid		6	ns
tCLQX	Output Hold Time	1		ns

### 5.6.6 Ethernet Timing Specification

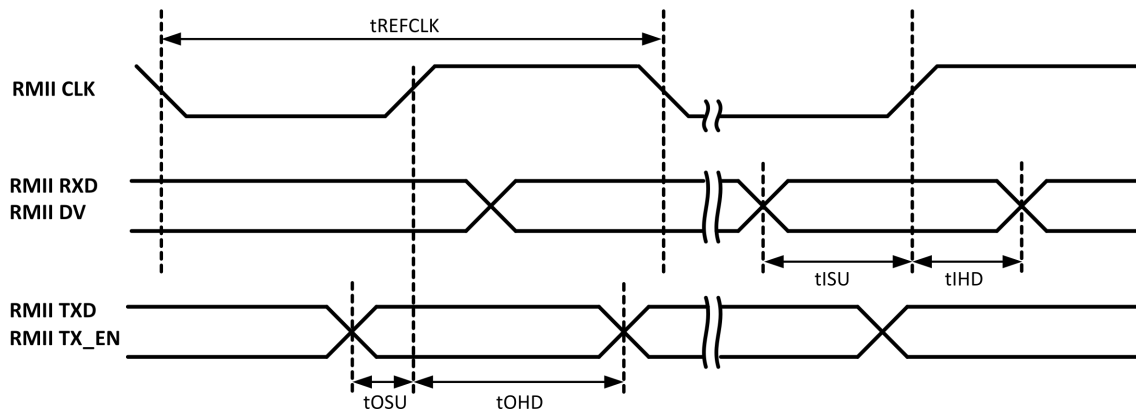
Figure 5-14 Management Data Timing Diagram



**Table 5-19 Management Data Timing Specification**

Symbol	Description	Min.	Typ.	Max.	Unit	Notes
tMDC	MDC clock Period	400	500		ns	From MAC
tMDCSU	Setup time to rising edge of MDC	10			ns	
tMDCHD	Hold time to rising edge of MDC	10			ns	

**Figure 5-15 RMIITiming Diagram**



**Table 5-20 RMIITiming Specification**

Symbol	Description	Min.	Typ.	Max	Unit	Notes
tREFCLK	RMIITiming clock period		20		ns	50MHz from PHY
tOSU	TXD & TX_EN setup time to rising edge of RMIITiming clock	1.8	10		ns	To PHY
tOHD	TXD & TX_EN hold time to rising edge of RMIITiming clock	1.4	10		ns	To PHY
tISU	RXD & DV setup time to rising edge of RMIITiming clock	1.0	10		ns	From PHY
tIHD	RXD & DV hold time to rising edge of RMIITiming clock	1.0	10		ns	From PHY

Figure 5-16 RGMII Receive Timing Diagram

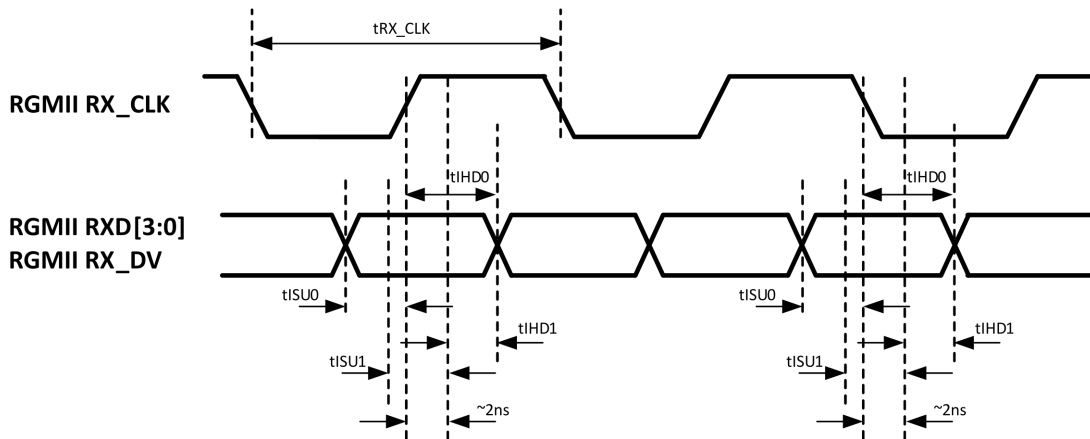


Table 5-21 RGMII Receive Timing Specification

Symbol	Description	Min.	Typ.	Max	Unit	Notes
$t_{RX\_CLK}$	RGMII RX_CLK clock period		8		ns	125MHz from PHY
$t_{SETUP}$	RXD[3:0] & RX_DV setup time (PHY internal delay enabled)	1.2			ns	From PHY
$t_{HOLD}$	RXD[3:0] & RX_DV hold time (PHY internal delay enabled)	1.2			ns	From PHY
$t_{SKEW}$	RXD[3:0] & RX_DV skew between these 5 signals (PHY internal delay disabled)	-0.5		0.5	ns	From PHY

When PHY internal delay is enabled, check setup/hold timing.

When PHY internal delay is disabled, check signal skew.

Figure 5-17 RGMII Transmit Timing Diagram

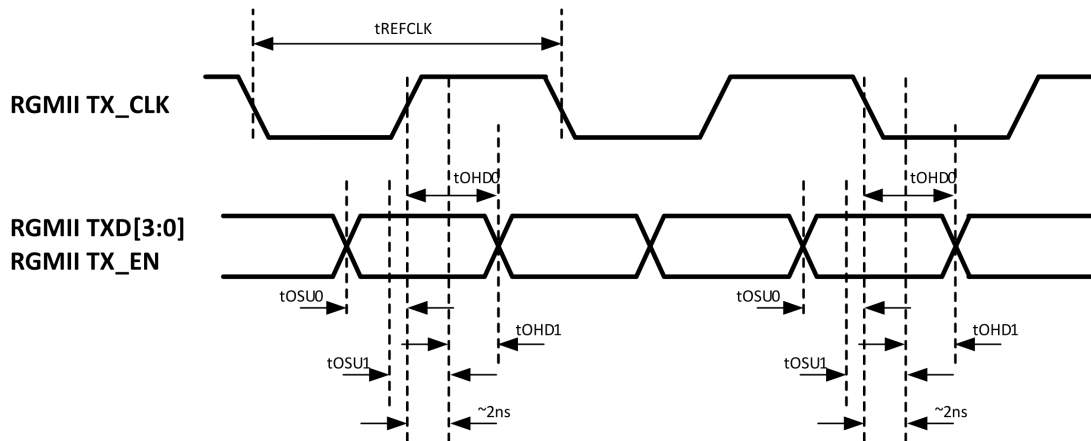


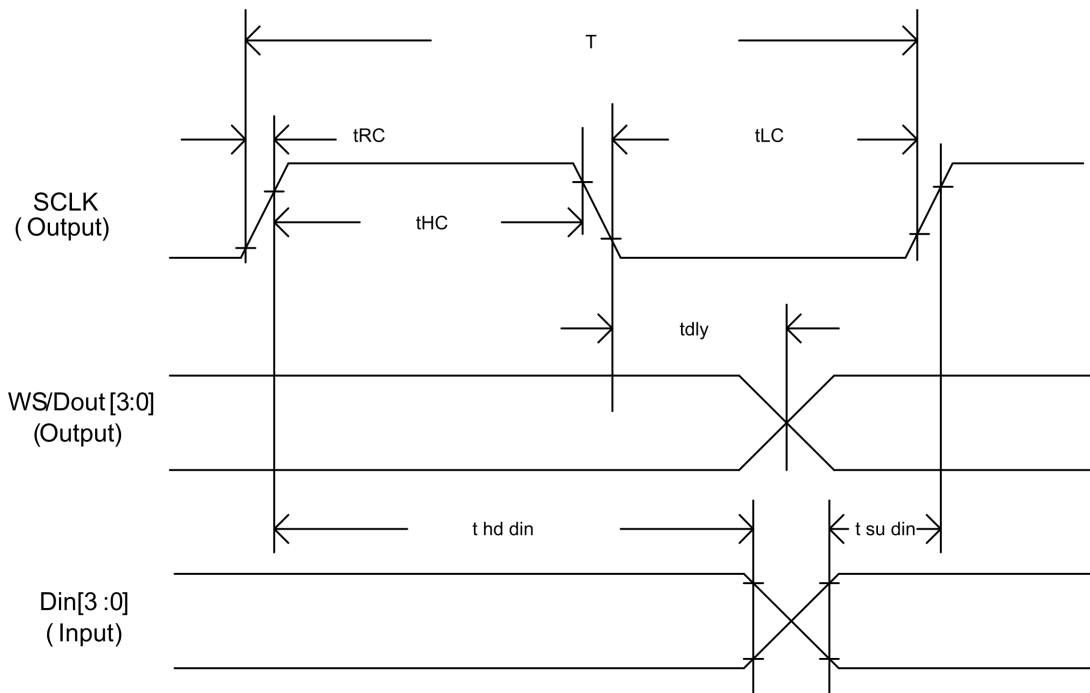
Table 5-22 RGMII Transmit Timing Specification

Symbol	Description	Min.	Typ.	Max	Unit	Notes
$t_{TX\_CLK}$	RGMII TX_CLK clock period		8		ns	125MHz to PHY
$t_{OSU}$	TXD & TX_EN setup time to rising edge of RGMII clock (no clock delay added)	1			ns	From PHY
	TXD & TX_EN setup time to rising edge of RGMII clock (clock delay added)	-0.9			ns	From PHY
$t_{OHD}$	RXD & DV hold time to rising edge of RGMII clock (no clock delay added)	0.8			ns	From PHY
	RXD & DV hold time to rising edge of RGMII clock (clock delay added)	2.7			ns	From PHY

### 5.6.7 Audio Timing Specification

There are two modes for the audio I2S/TDM interface: Master mode and Slave mode, as shown below.

**Figure 5-18 I2S/TDM Timing Diagram, Master Mode**



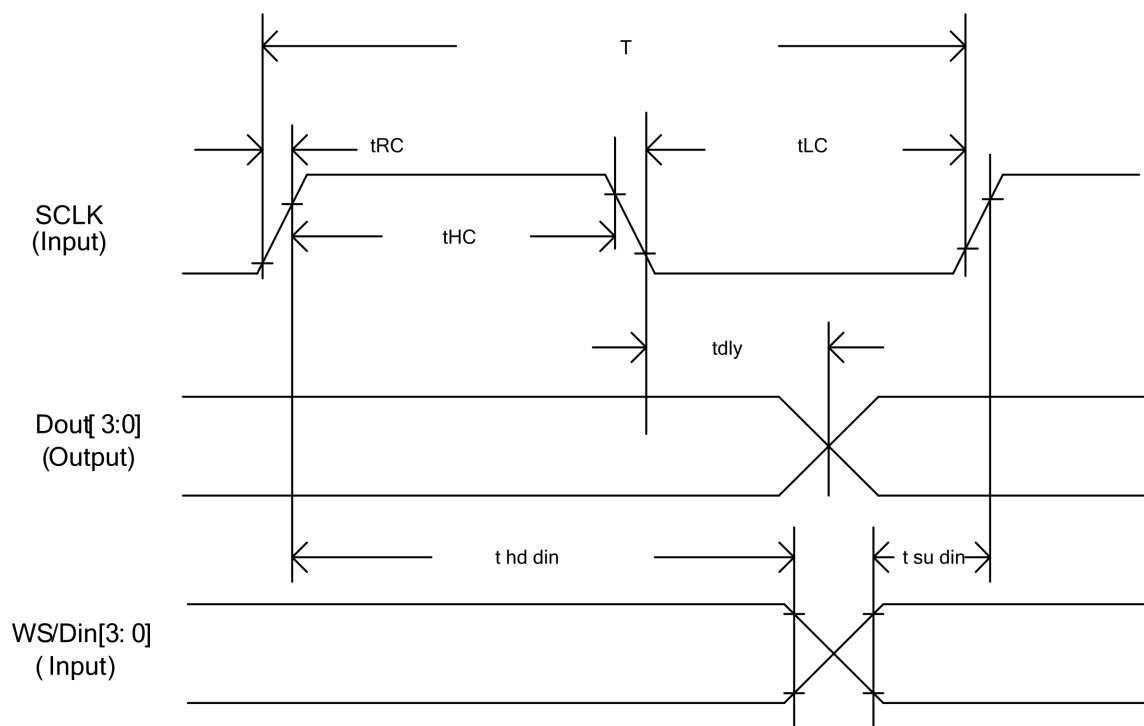
**Table 5-23 Audio I2S/TDM Timing Specification, Transmitter, Master Mode**

Transmitter (master mode)					
Symbol	Parameter	Min	Typ	Max	Unit
T	Clock period	10			ns
tHC	High level of SCLK	0.4			T
tLC	Low level of SCLK	0.4			T
tRC	Edge time of SCLK			0.15	T
tdly	Delay from SCLK to WS	-2	3	5	
tsuin	Setup time of Din	4			ns
thdin	Hold time of Din	4			ns

**Note**

Measure Point refers to VIH, ViL parameter of Normal GPIO Specifications.

Figure 5-19 2S/TDM Timing Diagram, Slave Mode



Transmitter (slave mode)					
Symbol	Parameter	Min	Typ	Max	unit
T(out)	Clock period	40			ns
T(in)	Clock period	10			ns
t <sub>HC</sub>	High level of SCLK	0.4			T
t <sub>LC</sub>	Low level of SCLK	0.4			T
t <sub>RC</sub>	Edge time of SCLK			0.8	ns
t <sub>su in</sub>	Setup time of WS/Din	4			ns
t <sub>hd in</sub>	Hold time of WS/Din	4			ns
t <sub>dly</sub>	Delay between SCLK and Dout	2	12	15	ns



**Note**

Measure Pointrefers to VIH, ViL parameter of Normal GPIO Specifications.

### 5.6.8 PDM Timing Specification

Figure 5-20 PDM Timing Diagram

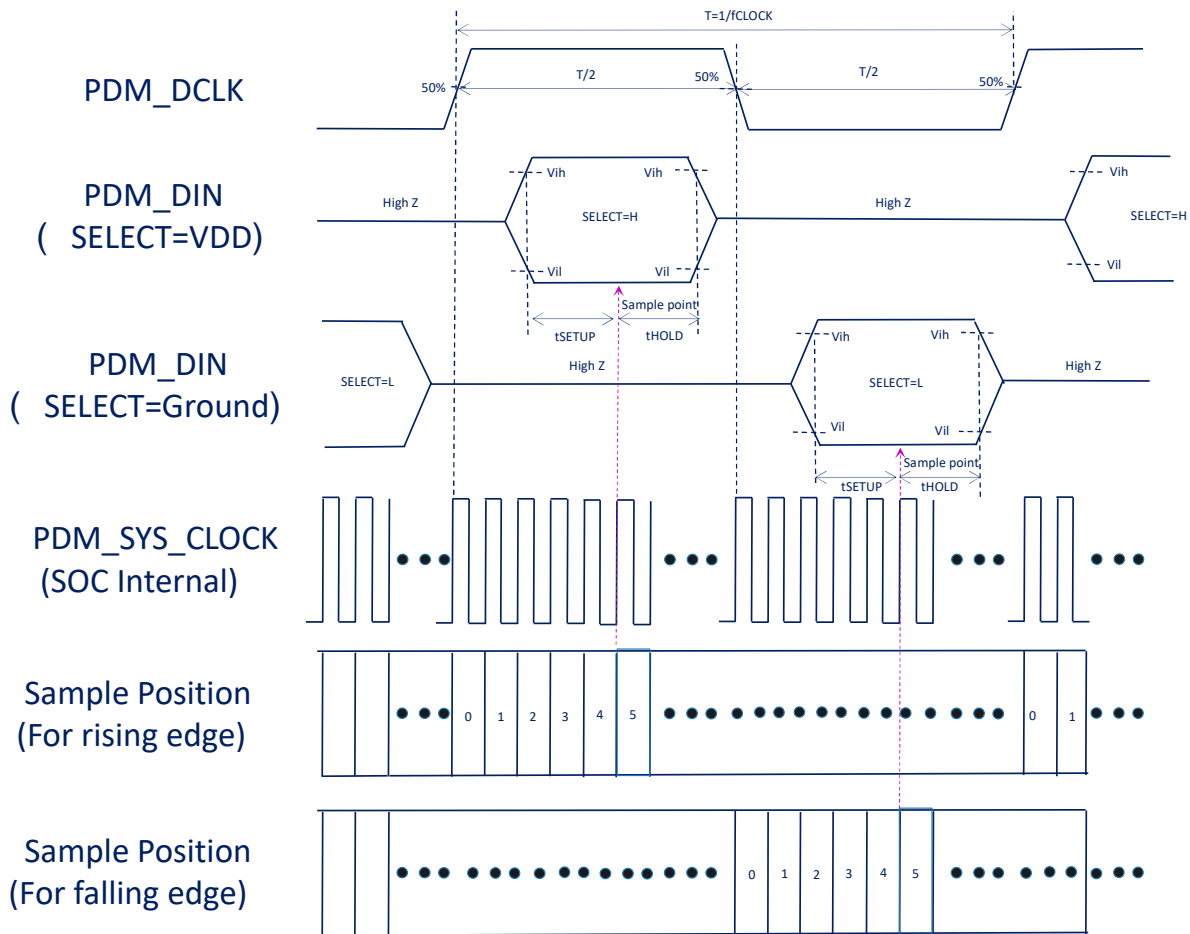


Table 5-24 PDM Timing Specification

Parameter	Symbol	Min.	Typ.	Max.	Units.
PDM clock period	tDCLK	200			ns
PDM clock duty cycle	tHIGH/tLOW	48%		52%	tDCLK
PDM Data setup time	tSETUP	20			ns
PDM Data hold time	tHOLD	20			ns
Sys clock period	tSYSCLK	5	7.5		ns

**Note**

1. Default PDM\_SYS\_CLOCK=133MHz.
2. For Sample position, please refer to PDM register PDM\_CHAN\_CTRL,PDM\_CHAN\_CTRL1.



### 5.6.9 UART Timing Specification

Figure 5-21 UART Timing Diagram

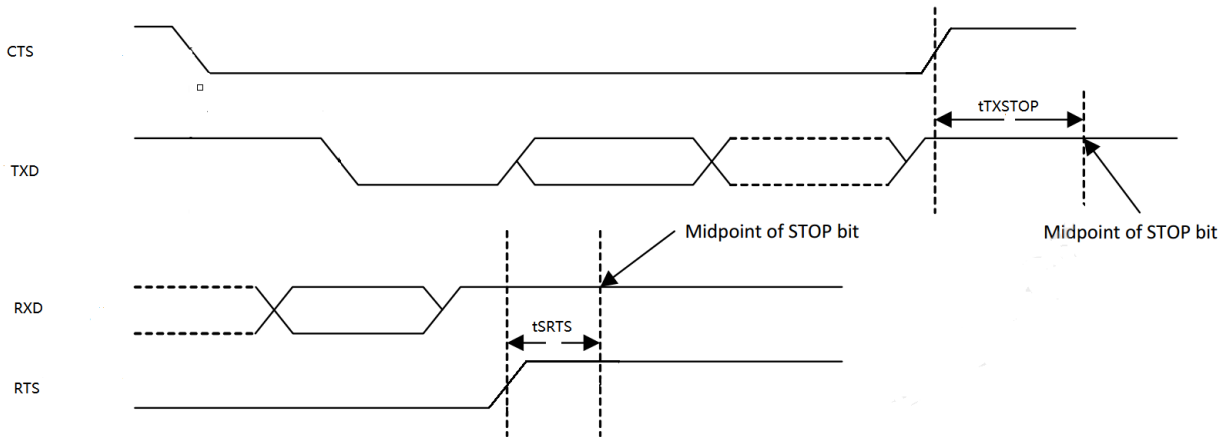


Table 5-25 UART Timing Specification

Parameter	Symbol	Min.	Max.	Units.
Delay time, CTS high before midpoint of stop bit	$t_{TXSTOP}$	-	0.5	Bit Periods
Delay time, midpoint of stop bit to RTS high	$t_{SRTS}$	-	0.5	Bit Periods

## 5.7 Recommended Oscillator Electrical Characteristics

The SoC requires the 24MHz oscillator for generating the main clock source.

Symbol	Description	Min.	Typ.	Max.	Unit	Notes
$F_o$	Nominal Frequency		24		MHz	
$\Delta f/f_o$	Frequency Tolerance	-30		30	ppm	At 25 °C
		-50		50	ppm	At -20~85 °C
$C_L$	Load Capacitance	7.5	12	12.5	pF	
ESR	Equivalent Series Resistance			100	oHm	

 **Note**

1. 10ppm Tolerance is preferred if 24MHz XTAL is also driving WIFI module.
2. For user external clock source, please connect input clock output to SYS\_OSCIN , let SYS\_OSCOUT floating.
3. The threshold of Xin inverter is around 0.9V (Xin range: -0.3V to +2.1V). Therefore, Following suggestion for input clock.
  - Suggestion 1: Without DC blocking capacitor, use a higher Vpp output TCXO. The high voltage should be higher than 1.35V (VSWING >1.35V, 0V to >1.35V).
  - Suggestion 2: With DC blocking capacitor, re-bias the middle voltage at 0.9V, VSWING >2\*0.45V;

## 5.8 Power On Config

3 Boot pins are used as power on config (POC) pins, to set the booting sequence.

POC setting is latched at the rising edge of reset signal.

3 POC pins are all pull high internal, CPU will try to boot from nand/eMMC first, if fails then try to boot from SD CARD, still fails then try to boot from USB (PC).

External 4.7K ohm pull down resistors can be used to change the POC setting. The resistors should be placed on right location, avoid stubs on high speed signals.

The SoC’s power on configuration is listed as following:

**Table 5-26 Power On Configuration Pin Table**

POC	Boot Pin	Name	Pull low	Pull high
POC_2	GPIOB_2	SPINOR_FIRST	SPI NOR first	Default sequence
POC_4	GPIOB_4	SDCARD_FIRST	SDCARD boot first	Default sequence
POC_5	GPIOB_5	USB_FIRST	USB boot first	Default sequence

**Table 5-27 Booting Sequence Diagram**

No.	GPIOB_5 POC5	GPIOB_4 POC4	GPIOB_2 POC2	1st	2nd	3rd	4th
1	0	0	0	SD	SPINOR	eMMC	USB
2	0	0	1	SD	eMMC	SPINOR	USB
3	0	1	0	USB	SPINOR	eMMC	SD
4	0	1	1	USB	eMMC	SPINOR	SD
5	1	0	0	SD	SPINOR	eMMC	USB
6	1	0	1	SD	eMMC	SPINOR	USB
7	1	1	0	SPINOR	eMMC	SD	USB
8	1	1	1	eMMC	SPINOR	SD	USB

## 5.9 Power On Reset

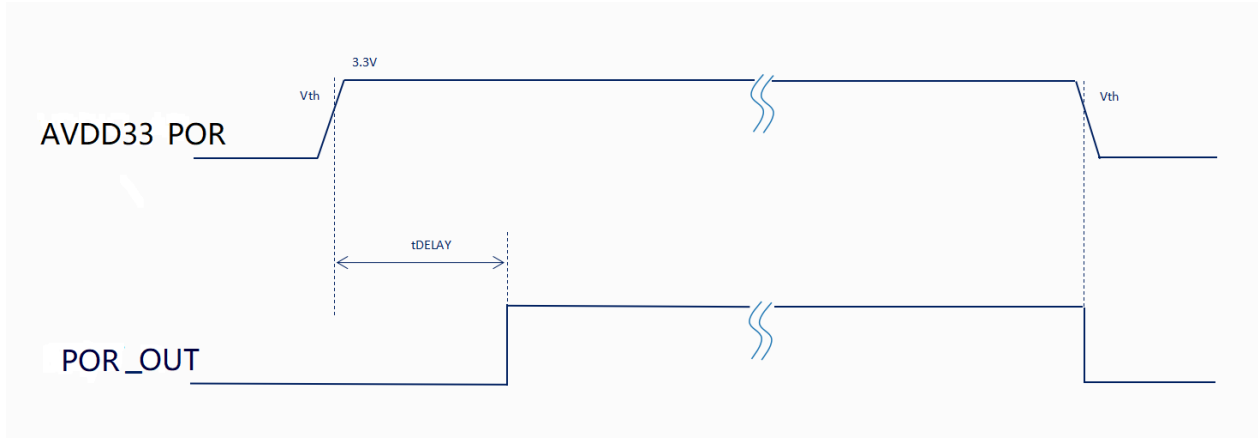
The POR (Power On Reset) monitors VDDIO\_AO power voltage and compares it to a threshold Voltage.

POR\_OUT pin is low (SOC is reset mode) when VDDIO\_AO is below threshold,

**Note**

1. Place 1nF capacitors on RESET\_N Pin.
2. VDDIO\_AO power pin is only support 3.3V, not allow to power off in sleep mode.

**Figure 5-22 POR Wave Diagram**



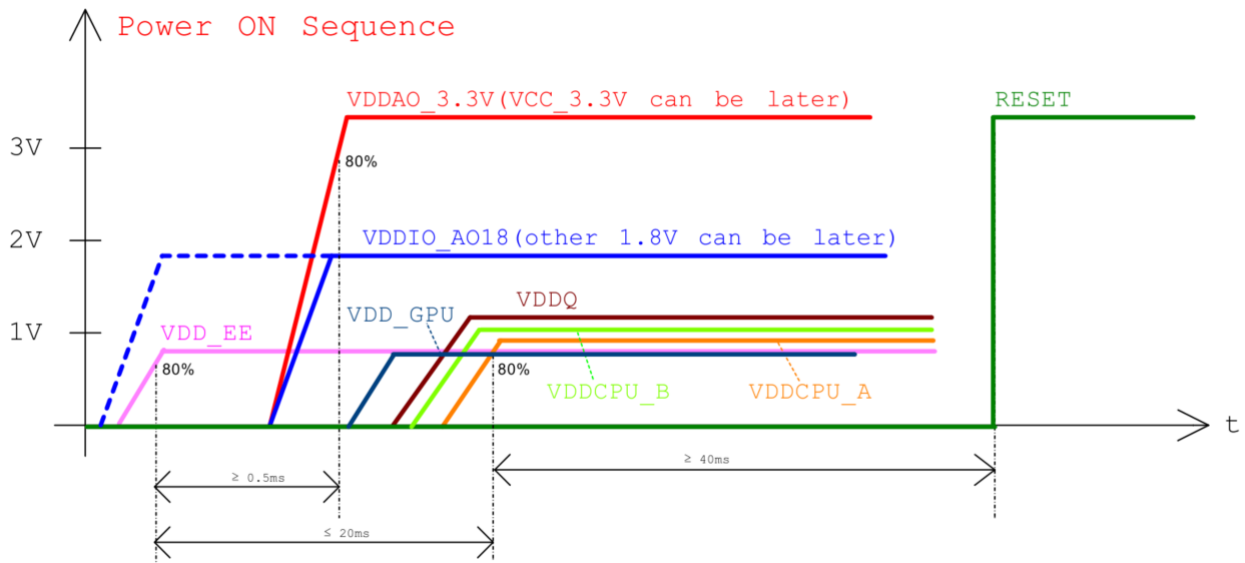
**Table 5-28 POR Specifications**

Parameter	Symbol	Min.	Typ.	Max.	Unit
Reset threshold for power up	Vth+	-	-	2.9	V
Reset threshold for power down	Vth-	2.5	-	-	V
Reset delay time	tDELAY	109	156	218	ms

## 5.10 Recommended Power On Sequence

The example of power on sequence is shown in the following figure.

Figure 5-23 Power On Sequence



**Note**

1. All test values refer to 80% of typical power voltage.
2. VDDAO\_3.3V & VCC\_3.3V should ramp up  $> 0.5ms$  later than VDD\_EE.
3. All power sources should get stable within 20ms (except for DDR\_VDDQ and VDD\_QLP).
4. No sequence requirement between VDD18\_AO and VDD\_EE. No sequence requirement between VDDCPU\_A & VDDCPU\_B & VDD\_GPU & VDDQ & VDD\_QLP & VDD\_DDR and other power source.
5. VDDIO\_AO18 should ramps up earlier or at the same time with VDDAO\_3.3V & VCC3.3V, VDDAO\_3.3V & VCC3.3V should never be 2.5V higher than VDD18\_AO.
6. In some designs, VDDCPU & VDD\_EE are merged to VCC\_CORE, the power on sequence should be same as VDD\_EE.
7. RESET\_n should keep low for at least 30ms after power up (except VDDQ and VDD\_QLP). Please refer to reference schematics.

## 5.11 Power Consumption

**Note**

Value listed here is estimated typical max value tested. Enough margin in circuit needs to be reserved.

Symbol	Maximum Current
VDD_DDR	1500
VDD_EE	3000
VDD_GPU	3000
VDDCPU_A	6000
VDDCPU_B	2000

Symbol	Maximum Current
VDDQ	1500
VDDQLP	600

Symbol	Typical Current (mA)	Maximum Current (mA)	Note
AVDD_DDR0PLL	12.0		
AVDD_DDR1PLL	12.0		
AVDD08_DP	11.0		
AVDD08_HDMIRX	87.1		
AVDD08_HDMITX	11.1		At 6 Gbps mode
AVDD0V8_CSI	18.0		
AVDD0V8_USB_PCIE	54.0		
AVDD18_MCLK	4.0		
AVDD18_DDR_DP_PLL	4.7		
AVDD18_PLL	4.7		
AVDD18_SARADC	1.0		
VDD18_AO	3.0		Including VDD18_EFUSE, Max 100mA when Programming EFUSE
AVDD18_USB_PCIE	85.0		
AVDD18_AUDIO	6.6		
AVDD18_CSI	9.0		
AVDD18_ENET	31.0		
AVDD18_HDMIRX_EARCTX	23.5		
AVDD18_HDMITX_EARCRX	22.3		
AVDD18_DP	210.0		
AVDD33_HDMIRX	90.0		Per connected channel
AVDD33_HDMITX	60.0		
AVDD33_POR	0.1		
AVDD33_USB	24.6		
VDDIO			

## 5.12 Storage and Baking Conditions

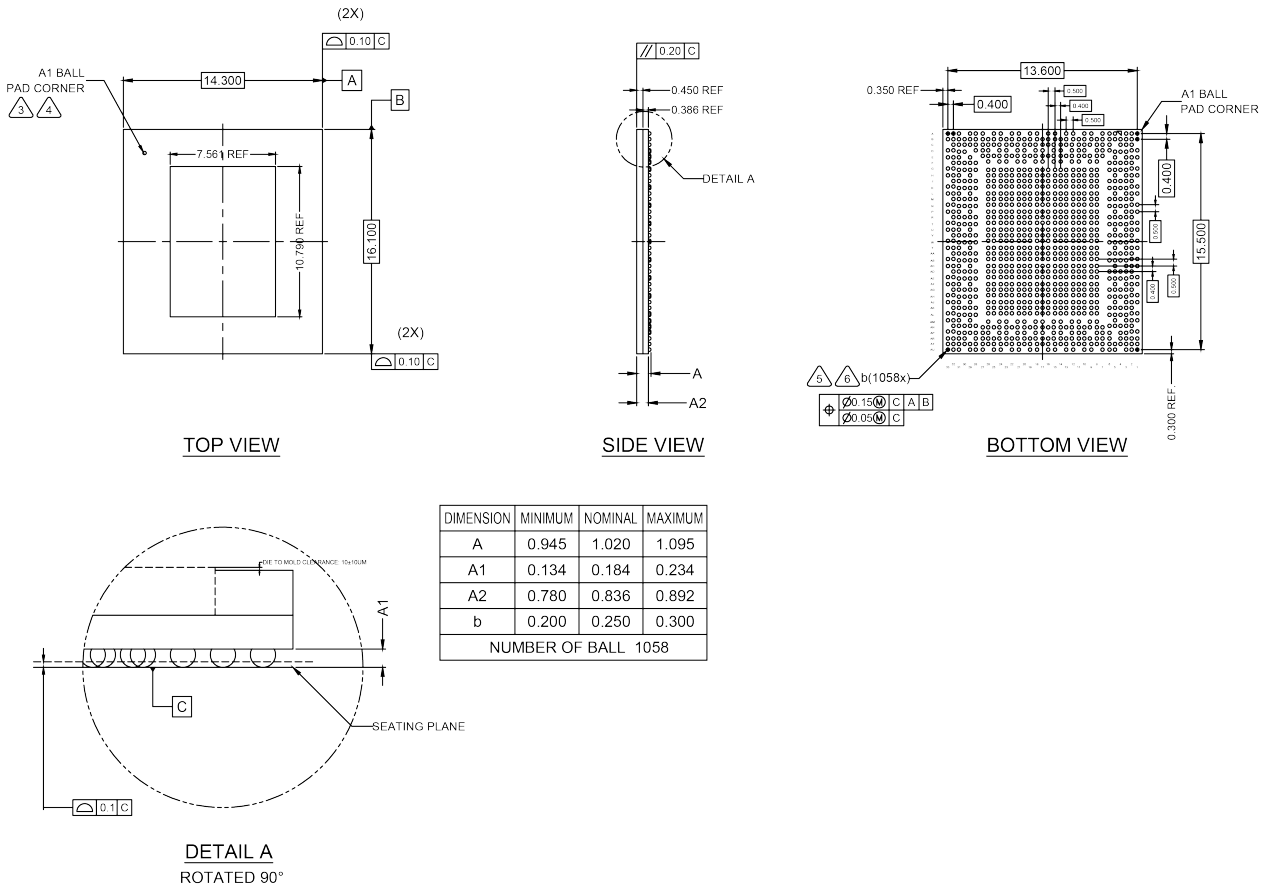
The processor is moisture-sensitive device of MSL level 3, defined by IPC/JEDEC J-STD-020. Please follow the storage and backing guidelines.

1. Calculated shelf life in sealed bag: 12 months at  $<40^{\circ}\text{C}$  and  $<90\%$  relative humidity (RH).
2. After bag is opened, devices that will be subjected to reflow solder or other high temperature process must comply with either of the following principles.
  - a. Mounted with 168 hours of factory conditions  $\leq 30^{\circ}\text{C}/60\%$  RH
  - b. Stored per J-STD-033
3. If the humidity indicator card reads  $>10\%$ , devices should be baked before mounting.
4. If baking is required, see IPC/JEDEC J-STB-033 for baking process.

# 6 Mechanical Dimensions

The processor comes in a 14.30x16.10 ball matrix FCBGA RoHS package. The mechanical dimensions are shown as the following figures.

Figure 6-1 Dimensions



# 7 System

This chapter describes the system architecture.

## 7.1 Memory Map

Memory map is listed in the following table.

Start	End	REGION (NORMAL)
FFFF8000	FFFFFFF	ROM_MMU
FFFF2000	FFFF7FFF	ROM_DFU
FFFE0000	FFFF1FFF	ROM_MAIN
FFF08000	FFFDFFFF	reserved
FFF00000	FFF07FFF	gic
FFE00000	FFEFFFFF	reserved
FF600000	FFDFFFFF	a53_dbg
FF050000	FF5FFFFF	reserved
FF040000	FF04FFFF	ge2d
FF000000	FF03FFFF	vpu
FE800000	FEFFFFFF	a73_dbg
FE700000	FE7FFFFF	CCI
FE500000	FE6FFFFF	reserved
FE480000	FE4FFFFF	reserved
FE440000	FE47FFFF	SECTOP
FE400000	FE43FFFF	mali
FE3C0000	FE3FFFFF	reserved
FE3B0000	FE3BFFFF	mipi_isp
FE3A0000	FE3AFFFF	hdmirx_core
FE390000	FE39FFFF	hdmirx
FE380000	FE38FFFF	hdmitx_core
FE370000	FE37FFFF	nna
FE360000	FE36FFFF	reserved
FE350000	FE35FFFF	dspb
FE340000	FE34FFFF	dspa
FE330000	FE33FFFF	audio
FE320000	FE32FFFF	dos
FE310000	FE31FFFF	wave
FE300000	FE30FFFF	hdmitx
FE0C0000	FE2FFFFF	reserved
FE0BE000	FE0BFFFF	reserved



Start	End	REGION (NORMAL)
FE0BC000	FE0BDFFF	reserved
FE0BA000	FE0BBFFF	reserved
FE0B8000	FE0B9FFF	reserved
FE0B6000	FE0B7FFF	reserved
FE0B4000	FE0B5FFF	reserved
FE0B2000	FE0B3FFF	edptx_ctrl1
FE0B0000	FE0B1FFF	edptx_ctrl0
FE0AE000	FE0AFFFF	reserved
FE0AC000	FE0ADFFF	reserved
FE0AA000	FE0ABFFF	reserved
FE0A8000	FE0A9FFF	reserved
FE0A6000	FE0A7FFF	reserved
FE0A4000	FE0A5FFF	reserved
FE0A2000	FE0A3FFF	reserved
FE0A0000	FE0A1FFF	ddr_pll
FE09E000	FE09FFFF	aucpu
FE09C000	FE09DFFF	aocpu
FE09A000	FE09BFFF	temp_sensor_hevc
FE098000	FE099FFF	temp_sensor_vpu
FE096000	FE097FFF	temp_sensor_nna
FE094000	FE095FFF	temp_sensor_gpu
FE092000	FE093FFF	axi_sramb
FE090000	FE091FFF	axi_srama
FE08E000	FE08FFFF	GDC
FE08C000	FE08DFFF	nand_emmc_C
FE08A000	FE08BFFF	sdio_B
FE088000	FE089FFF	sdio_A
FE086000	FE087FFF	i2c_m_ao_b
FE084000	FE085FFF	ir_top
FE082000	FE083FFF	uart_f
FE080000	FE081FFF	uart_e
FE07E000	FE07FFFF	uart_d
FE07C000	FE07DFFF	uart_c
FE07A000	FE07BFFF	uart_b
FE078000	FE079FFF	uart_a
FE076000	FE077FFF	i2c_m_ao_a

Start	End	REGION (NORMAL)
FE074000	FE075FFF	mipi_dsi_host a
FE072000	FE073FFF	mipi_dsi_host b
FE070000	FE071FFF	i2c_m_f
FE06E000	FE06FFFF	i2c_m_e
FE06C000	FE06DFFF	i2c_m_d
FE06A000	FE06BFFF	i2c_m_c
FE068000	FE069FFF	i2c_m_b
FE066000	FE067FFF	i2c_m_a
FE064000	FE065FFF	i2c_s_a
FE062000	FE063FFF	usb30_apb
FE060000	FE061FFF	pwm_ao_cd
FE05E000	FE05FFFF	pwm_ao_ab
FE05C000	FE05DFFF	pwm_ef
FE05A000	FE05BFFF	pwm_cd
FE058000	FE059FFF	pwm_ab
FE056000	FE057FFF	spifc
FE054000	FE055FFF	spicc_2
FE052000	FE053FFF	spicc_1
FE050000	FE051FFF	spicc_0
FE04E000	FE04FFFF	spicc_5
FE04C000	FE04DFFF	spicc_4
FE04A000	FE04BFFF	spicc_3
FE048000	FE049FFF	mnr_clk
FE046000	FE047FFF	aififo
FE044000	FE045FFF	cec
FE042000	FE043FFF	RSA
FE040000	FE041FFF	dewarp
FE03E000	FE03FFFF	usb20_phy1
FE03C000	FE03DFFF	usb20_phy0
FE03A000	FE03BFFF	usb_comb
FE038000	FE039FFF	Smart Card
FE036000	FE037FFF	dmc0
FE034000	FE035FFF	dmc1
FE032000	FE033FFF	pwm_ao_gh
FE030000	FE031FFF	pwm_ao_ef
FE02E000	FE02FFFF	startup

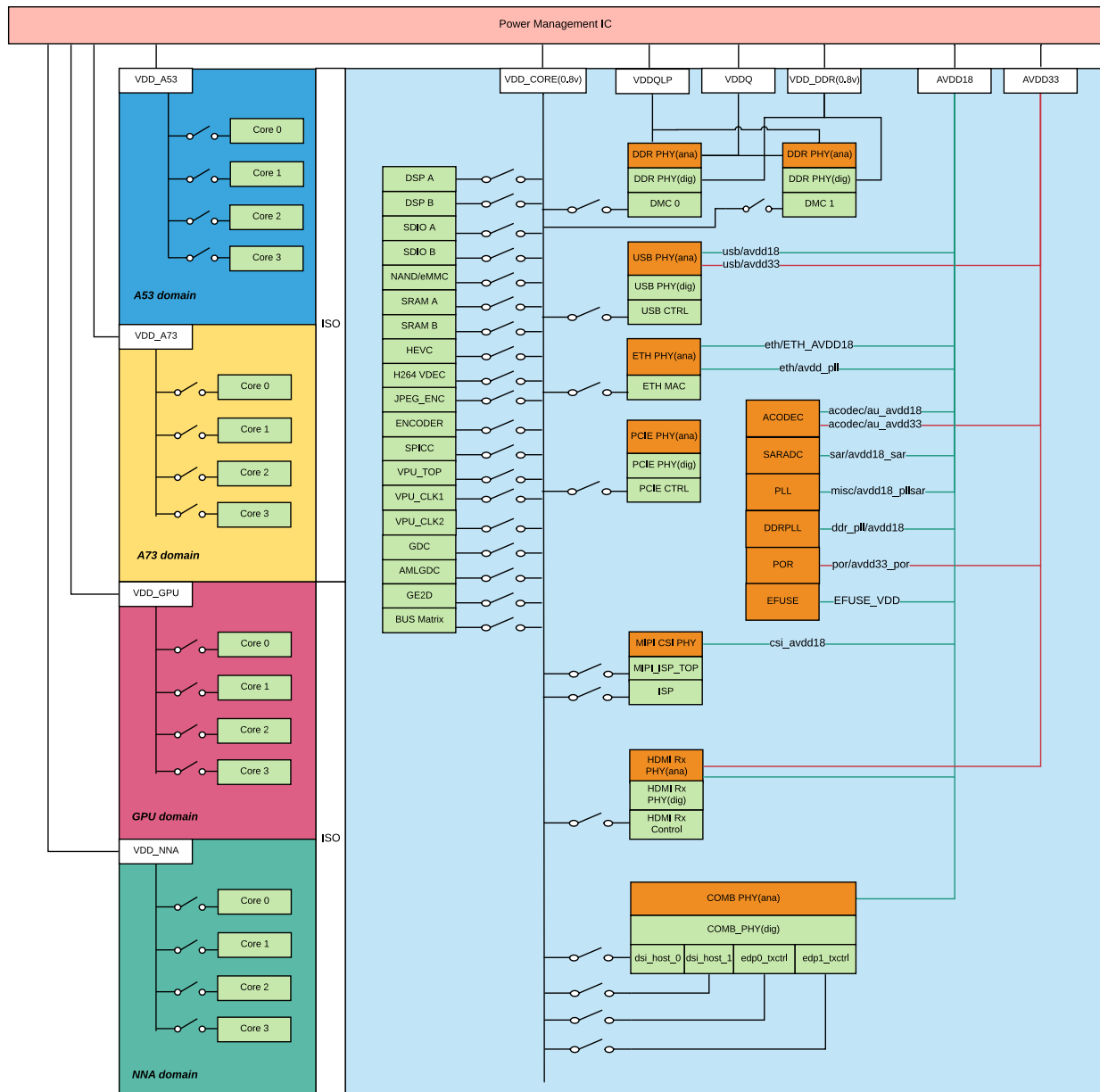
Start	End	REGION (NORMAL)
FE02C000	FE02DFFF	pcie_A_ctrl
FE02A000	FE02BFFF	pcie_apb
FE028000	FE029FFF	eth_phy
FE026000	FE027FFF	sar_adc
FE024000	FE025FFF	eth_top
FE022000	FE023FFF	temp_sensor_a53
FE020000	FE021FFF	temp_sensor_a73
FE01E000	FE01FFFF	hdmi20_aes
FE01C000	FE01DFFF	reserved
FE01A000	FE01BFFF	acodec_ctrl
FE018000	FE019FFF	edp_dsi_vx1_dphy
FE016000	FE017FFF	mipi_dsi_phy1
FE014000	FE015FFF	mipi_dsi_phy0
FE012000	FE013FFF	capu
FE010000	FE011FFF	sys_ctrl
FE00E000	FE00FFFF	cpu_ctrl
FE00C000	FE00DFFF	pwr_ctrl
FE00A000	FE00BFFF	irq_ctrl
FE008000	FE009FFF	analog_ctrl
FE006000	FE007FFF	mailbox
FE004000	FE005FFF	pad_ctrl
FE002000	FE003FFF	reset_ctrl
FE000000	FE001FFF	clk_ctrl
FDF00000	FDFFFFFFFF	reserved
FDE00000	FDEFFFFFFF	usb0(usb3_drd)
FDD00000	FDDFFFFFFF	usb1(usb2_drd)
FDC10000	FDCFFFFFFF	reserved
FDC00000	FDC0FFFF	eth
FDB00000	FDBFFFFFFF	reserved
FD500000	FDAFFFFFFF	reserved
FD400000	FD4FFFFFFF	NIC5_GPV
FD300000	FD3FFFFFFF	NIC4_GPV
FD200000	FD2FFFFFFF	NIC3_GPV
FD100000	FD1FFFFFFF	NIC2_GPV
FD000000	FD0FFFFFFF	NIC1_GPV
FC000000	FCFFFFFFF	ddr_top_0

Start	End	REGION (NORMAL)
FB000000	FBFFFFFF	ddr_top_1
FA040000	FAFFFFFF	reserved
FA000000	FA03FFFF	isp
F8000000	F9FFFFFF	pcie b
F7200000	F7FFFFFF	reserved
F7100000	F71FFFFF	axi sramb
F7000000	F70FFFFF	axi srama
F5000000	F6FFFFFF	pcie a
F1000000	F4FFFFFF	flash

## 7.2 Power Domain

The following figure shows the power domain of the SoC.

Figure 7-1 Power Domain



The power domain ID is matched to the corresponding register bit, it can be used to set the registers for power switching.

Domain Name	ID
PM_A53_PWR	0
PM_A53_CORE0	1
PM_A53_CORE1	2
PM_A53_CORE2	3
PM_A53_CORE3	4
PM_A73_PWR	5
PM_A73_CORE0	6

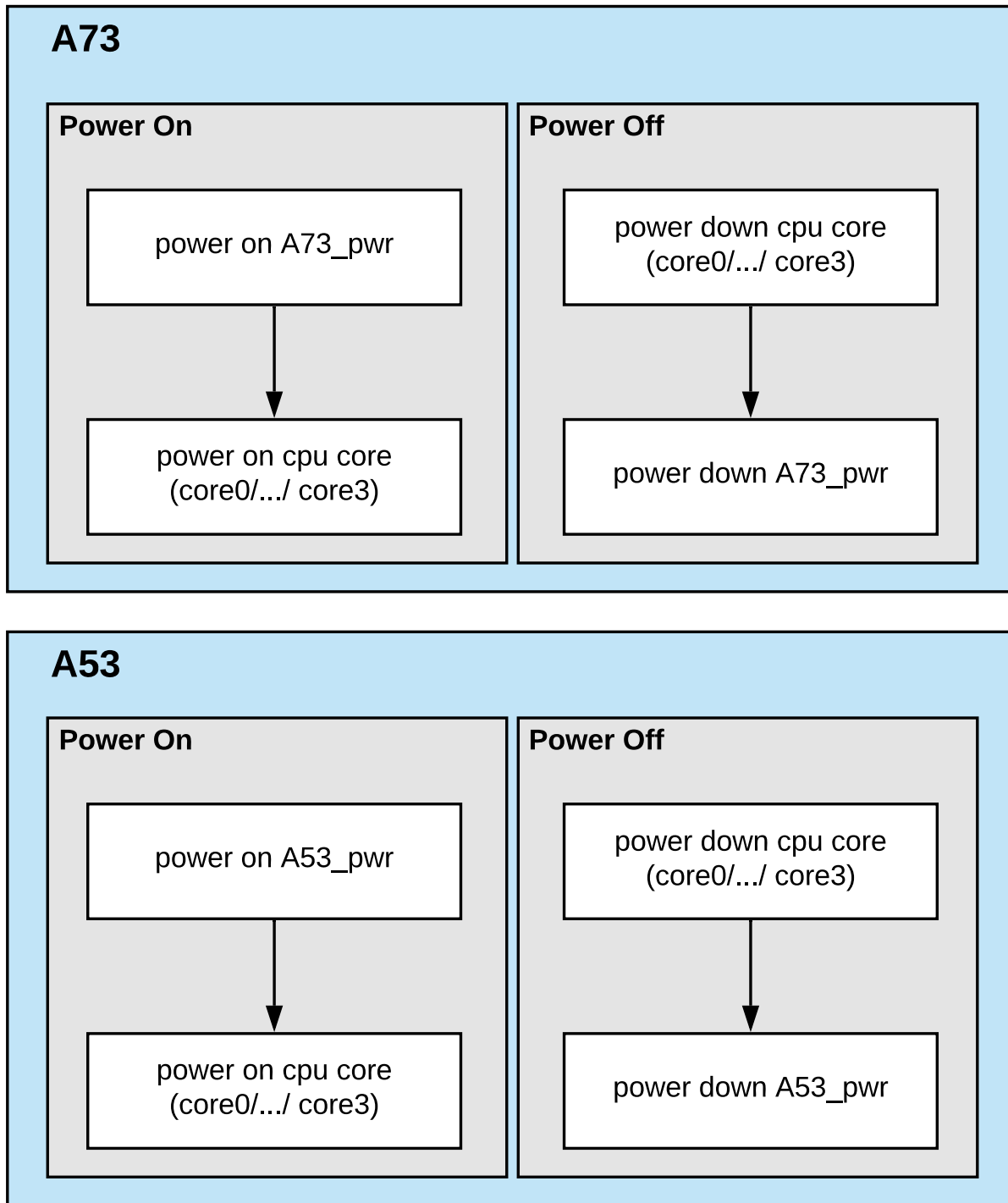
Domain Name	ID
PM_A73_CORE1	7
PM_A73_CORE2	8
PM_A73_CORE3	9
PM_DSPA	10
PM_DSPB	11
PM_DOS_HCODEC	24
PM_DOS_HEVC	25
PM_DOS_VDEC	26
PM_DOS_WAVE	27
PM_VPU_HDMI	28
PM_USB_COMB	30
PM_PCIE	31
PM_GE2D	32
PM_HDMIRX	35
PM_VI_CLK1	36
PM_VI_CLK2	37
PM_ETH	39
PM_ISP	41
PM_MIPI_ISP	42
PM_GDC	43
PM_DEWARP	44
PM_SDEMMC_A	45
PM_SDEMMC_B	46
PM_SDEMMC_C	47
PM_MALI_SC0	48
PM_MALI_SC1	49
PM_MALI_SC2	50
PM_MALI_SC3	51
PM_MALI_TOP	52
PM_NNA_CORE0	56

Domain Name	ID
PM_NNA_CORE1	57
PM_NNA_CORE2	58
PM_NNA_CORE3	59
PM_NNA_TOP	60
PM_DDR0	62
PM_DDR1	63
PM_DMC0	64
PM_DMC1	65
PM_NOC	66
PM_NIC2	67
PM_NIC3	68
PM_CC1	69
PM_MIPI_DSI0	70
PM_SPICC0	71
PM_SPICC1	72
PM_SPICC2	73
PM_SPICC3	74
PM_SPICC4	75
PM_SPICC5	76
PM_EDP0	77
PM_EDP1	78
PM_MIPI_DSI1	79

### 7.2.1 A73 & A53 Power Domain

The A73/A53 domain is 2-levels power domain. The A73/A53 domain itself consists of 4 CPU cores, a L2 cache controller and a SCU. The A73/A53 CPU boots with the SCU/L2 and each one of cores powered. Each CPU core can be enabled and disabled independently using the control bits described below. The diagram below illustrates the transition to each state.

Figure 7-2 Power on & Power off process



### 7.2.2 General Power Domain

The power domains except A73 and A53 power domain are General Power Domains, they are 1-level power domain.

Power domains can be shut down by software way or hardware way.



### 7.2.3 Power Switch Sequence

Power switch sequences include several steps of reset, isolation, power switch etc, functions of each step are listed below:

- Reset: Reset or de-reset the whole domain.
- Isolation: Isolate the power domain from exporting unknown state to powered-on domains. Enable or Disable isolation.
- Power switch: Power on/off memories and logics.

#### Typical Power On Stages

A typical power on process is described below:

1. Keep the reset status of the whole domain;
2. Power on logics with 10us~100us of time delay;
3. Power on memories, with 10us~100us of time delay;
4. Disable isolations;
5. De-assert reset.

In some scenarios, we need to enable the gated-off clocks when domain is powered on. This action is recommended to be done between step 4 and step 5.

#### Typical Power Off Stages

A typical power off process is described below:

1. Keep the reset status of the whole domain;
2. Enable isolations;
3. Power off memories with 5us~10us of time delay;
4. Power off logics with 5us~10us of time delay;

In some scenarios, we need to gate off clocks when domain is powered down. This action is recommended to be done between step 1 step 2.

We can perform the power switch sequence by software way or hardware way. But not all power domains support hardware power switching.

The following power domains support hardware power switching:

- PM\_A73\_PWR
- PM\_A73\_CORE0
- PM\_A73\_CORE1
- PM\_A73\_CORE2
- PM\_A73\_CORE3
- PM\_A53\_PWR
- PM\_A53\_CORE0
- PM\_A53\_CORE1
- PM\_A53\_CORE2
- PM\_A53\_CORE3
- PM\_DSP\_A
- PM\_DSP\_B

Other power domains don't support hardware power switching.

### 7.2.4 Power Control

Depends on applications, the power switch sequence can be performed by software power control or by hardware power control.

For the following scenarios, hardware power control is recommended:

- All processors (DSP and CPU) are powered-off, need IRQ awake one of them;
- No processor is powered-on, but still need to power on the domains other than CPU/DSP domains.
- PM\_CPU\_TOP
- PM\_CPU\_CORE0
- PM\_CPU\_CORE1
- PM\_DSP\_A
- PM\_SP\_TOP

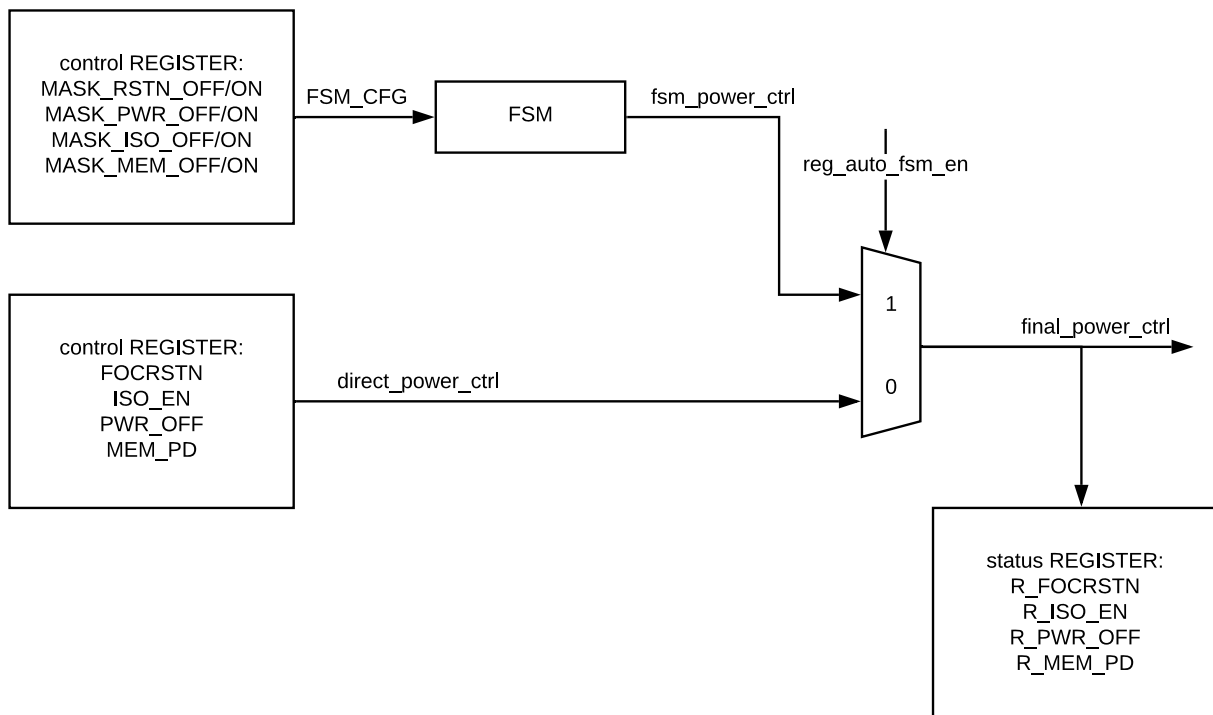
For other power domains that don't support hardware power switching, please use software power control scheme.

#### 7.2.4.1 Hardware Power Control

The SOC integrates a hardware logic module to control power switching, shown in below diagram.

- If PWR\_CTRL FSM is disabled, the power switching is controlled by register configuration;
- If PWR\_CTRL FSM is enabled, the power control will be controlled by PWR\_CTRL FSM.

**Figure 7-3 Power Control Mux**



#### PWR\_CTRL FSM

The diagram of state machine is illustrated at below.

Every step of power switch is implemented in FSM.

Every step has delay which can be adjusted by registers.

IDLE to OFF\_WAIT is controlled by reg\_start\_pwr\_off register. It's the start of domain power-off.

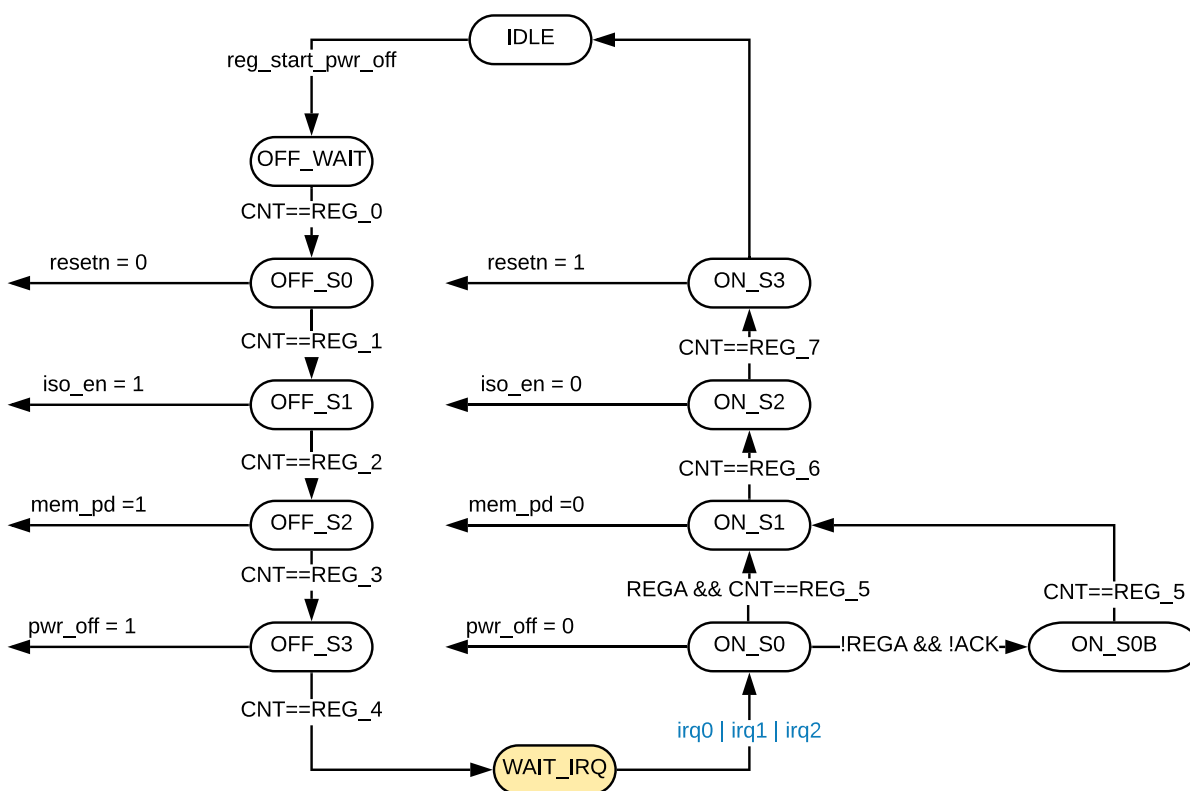
WAIT\_IRQ to ON\_S0 is controlled by IRQ (3 selections from all the SOC interrupts).



**Note**

Only one FSM is integrated in power control, if FSM starts, all the modules' power switching will be done at the same time.

**Figure 7-4 PWR\_CTRL FSM Diagram**



**Configuration**

Mask registers are used to configure the power switching. They can be overlaid with the current power status based on software registers settings.

When shut off/on one domain by hardware, please configure the steps of power switching sequences in advance. For example, enable reset when PWR\_CTRL FSM in OFF\_S0, enable iso\_en when PWR\_CTRL FSM in OFF\_S1. FSM will perform the steps following the script. We should build the state transition script before start the state machine. Every power domain has their mask configuration registers. Including:

1. Reset mask register,
2. Isolation mask register,
3. Memory pd off mask register,
4. Memory pd on mask register,
5. Power switch mask register.

## Switching from Hardware Control to Software Control

Hardware power control is overlaid on the original power state, software power control is not, so if you need to switch to software way when PWR\_CTRL FSM is enabled, we should save the previous power state of hardware switching, otherwise the state may be lost and cause power domain switching to the wrong power state unintentionally.

When switching hardware power control to software power control, perform the following:

1. Save power switch register;
2. Save isolation register;
3. Save reset control register.

### Transition Delay Setting

PWR\_CTRL FSM supports delay configurations between each step. Users can adjust the delay of each step. The delay unit can also be set by register.

3 delay units are supported:

- 1 us
- 10 us
- 100 us

Supported transition delay configurations are listed below:

- OFF\_WAIT to OFF\_S0: Delay between assert power off and assert reset;
- OFF\_S0 to OFF\_S1: Delay between assert reset and enable isolation;
- OFF\_S1 to OFF\_S2: Delay between enable isolation and memory power down;
- OFF\_S2 to OFF\_S3: Delay between memory power down and enable power switch;
- OFF\_S3 to WAIT\_IRQ: Delay between power switch and power-off done;
- ON\_S0 to ON\_S0b: Delay between power on logics and power ack (power ack check is enabled);
- ON\_S0b to ON\_S1: Delay between power ack and power on memory(power ack check is enabled);
- ON\_S0 to ON\_S1: Delay between power on logics and power on memory(power ack check is not enabled);
- ON\_S1 to ON\_S2: Delay between power on memory and disable isolation;
- ON\_S2 to ON\_S3: Delay between disable isolation and de-assert reset.

### Power on IRQ Setting

PWR\_CTRL FSM supports all SOC IRQ sources to awake sleep power domains. Any SOC IRQ can awake power domains.

Supports GIC wake interrupts. That means all the interrupts routed to GIC can be used as the awake interrupt.

## 7.2.4.2 Software Power Control

The processor can perform the power switch sequences by registers.

Below is example code for power switching.

```
void power_switch_to_domains(PM_E domain, uint32_t pwr_ctrl)
{
    if(pwr_ctrl == PWR_ON) {
        st_printf("[PWR]: Power on %s domain.\n", get_domain_name(domain));

        // assert reset
        do_reset(domain, PWR_OFF);

        // Powerup Power Domain
    }
}
```

```

do_power_switch(domain, PWR_ON);

delay_us(50);

// Powerup memories
do_power_memory(domain, PWR_ON);

delay_us(100);

// remove isolations
do_iso_en(domain, PWR_ON);

// deassert reset
do_reset(domain, PWR_ON);
} else {
    st_printf("[PWR]: power off %s domain.\n", get_domain_name(domain));

    // reset
do_reset(domain, PWR_OFF);

    // add isolation to domain
do_iso_en(domain, PWR_OFF);

    // Power down memories
do_power_memory(domain, PWR_OFF);
delay_us(5);
    // Power off domain
do_power_switch(domain, PWR_OFF);
delay_us(10);
}
}

```

## 7.2.5 Register Description

### Register Address

- PWRCTRL\_PWR\_ACK0 0xfe00c000
- PWRCTRL\_PWR\_ACK1 0xfe00c004
- PWRCTRL\_PWR\_OFF0 0xfe00c010
- PWRCTRL\_PWR\_OFF1 0xfe00c014
- PWRCTRL\_ISO\_EN0 0xfe00c020
- PWRCTRL\_ISO\_EN1 0xfe00c024
- PWRCTRL\_FOCRST0 0xfe00c030
- PWRCTRL\_FOCRST1 0xfe00c034
- PWRCTRL\_MEM\_PD0 0xfe00c040
- PWRCTRL\_MEM\_PD1 0xfe00c044
- PWRCTRL\_MEM\_PD2 0xfe00c048
- PWRCTRL\_MEM\_PD3 0xfe00c04c
- PWRCTRL\_MEM\_PD4 0xfe00c050
- PWRCTRL\_MEM\_PD5 0xfe00c054
- PWRCTRL\_MEM\_PD6 0xfe00c058
- PWRCTRL\_MEM\_PD7 0xfe00c05c
- PWRCTRL\_MEM\_PD8 0xfe00c060
- PWRCTRL\_MEM\_PD9 0xfe00c064
- PWRCTRL\_MEM\_PD10 0xfe00c068
- PWRCTRL\_MEM\_PD11 0xfe00c06c
- PWRCTRL\_MEM\_PD12 0xfe00c070
- PWRCTRL\_MEM\_PD13 0xfe00c074

- PWRCTRL\_MEM\_PD14 0xfe00c078
- PWRCTRL\_MEM\_PD15 0xfe00c07c
- PWRCTRL\_MEM\_PD16 0xfe00c080
- PWRCTRL\_MEM\_PD17 0xfe00c084
- PWRCTRL\_MEM\_PD18 0xfe00c088
- PWRCTRL\_MEM\_PD19 0xfe00c08c
- PWRCTRL\_MEM\_PD20 0xfe00c090
- PWRCTRL\_MEM\_PD21 0xfe00c094
- PWRCTRL\_MEM\_PD22 0xfe00c098
- PWRCTRL\_MEM\_PD23 0xfe00c09c
- PWRCTRL\_MEM\_PD24 0xfe00c0a0
- PWRCTRL\_MEM\_PD25 0xfe00c0a4
- PWRCTRL\_IRQ\_EN 0xfe00c0c0
- PWRCTRL\_IRQ\_STS 0xfe00c0c4
- PWRCTRL\_CPU0\_AUTO\_OFF\_CTRL0 0xfe00c100
- PWRCTRL\_CPU0\_AUTO\_OFF\_CTRL1 0xfe00c104
- PWRCTRL\_CPU0\_AUTO\_OFF\_CTRL2 0xfe00c108
- PWRCTRL\_CPU0\_AUTO\_OFF\_CTRL3 0xfe00c10c
- PWRCTRL\_CPU0\_AUTO\_OFF\_CTRL4 0xfe00c110
- PWRCTRL\_CPU0\_AUTO\_OFF\_CTRL5 0xfe00c114
- PWRCTRL\_CPU0\_TIMER\_TH\_01 0xfe00c120
- PWRCTRL\_CPU0\_TIMER\_TH\_23 0xfe00c124
- PWRCTRL\_CPU0\_TIMER\_TH\_45 0xfe00c128
- PWRCTRL\_CPU0\_TIMER\_TH\_67 0xfe00c12c
- PWRCTRL\_CPU0\_TIMER\_TH\_89 0xfe00c130
- PWRCTRL\_CPU0\_IRQ\_MASK0 0xfe00c140
- PWRCTRL\_CPU0\_IRQ\_MASK1 0xfe00c144
- PWRCTRL\_CPU0\_IRQ\_MASK2 0xfe00c148
- PWRCTRL\_CPU0\_IRQ\_MASK3 0xfe00c14c
- PWRCTRL\_CPU0\_IRQ\_MASK4 0xfe00c150
- PWRCTRL\_CPU0\_IRQ\_MASK5 0xfe00c154
- PWRCTRL\_CPU0\_IRQ\_MASK6 0xfe00c158
- PWRCTRL\_CPU0\_IRQ\_MASK7 0xfe00c15c
- PWRCTRL\_CPU0\_IRQ\_MASK8 0xfe00c160
- PWRCTRL\_CPU0\_IRQ\_MASK9 0xfe00c164
- PWRCTRL\_CPU0\_IRQ\_MASK10 0xfe00c168
- PWRCTRL\_CPU0\_IRQ\_MASK11 0xfe00c16c
- PWRCTRL\_CPU0\_IRQ\_MASK12 0xfe00c170
- PWRCTRL\_CPU0\_IRQ\_MASK13 0xfe00c174
- PWRCTRL\_CPU0\_IRQ\_MASK14 0xfe00c178
- PWRCTRL\_CPU0\_IRQ\_MASK15 0xfe00c17c
- PWRCTRL\_CPU0\_MEMPD\_INIT\_SET 0xfe00c180
- PWRCTRL\_CPU0\_MEMPD\_OFF\_SET 0xfe00c184
- PWRCTRL\_CPU0\_MEMPD\_ON\_A\_SET 0xfe00c188

- PWRCTRL\_CPU0\_MEMPD\_ON\_B\_SET 0xfe00c18c
- PWRCTRL\_CPU0\_MEMPD\_ON\_C\_SET 0xfe00c190
- PWRCTRL\_CPU0\_MEMPD\_ON\_D\_SET 0xfe00c194
- PWRCTRL\_CPU0\_MEMPD\_STS 0xfe00c198
- PWRCTRL\_CPU0\_FSM\_STS0 0xfe00c19c
- PWRCTRL\_CPU0\_FSM\_STS1 0xfe00c1a0
- PWRCTRL\_CPU0\_FSM\_STS2 0xfe00c1a4
- PWRCTRL\_CPU0\_FSM\_START\_OFF 0xfe00c1b4
- PWRCTRL\_CPU0\_FSM\_START\_ON 0xfe00c1b8
- PWRCTRL\_CPU0\_FSM\_JUMP 0xfe00c1bc
- PWRCTRL\_CPU1\_AUTO\_OFF\_CTRL0 0xfe00c1c0
- PWRCTRL\_CPU1\_AUTO\_OFF\_CTRL1 0xfe00c1c4
- PWRCTRL\_CPU1\_AUTO\_OFF\_CTRL2 0xfe00c1c8
- PWRCTRL\_CPU1\_AUTO\_OFF\_CTRL3 0xfe00c1cc
- PWRCTRL\_CPU1\_AUTO\_OFF\_CTRL4 0xfe00c1d0
- PWRCTRL\_CPU1\_AUTO\_OFF\_CTRL5 0xfe00c1d4
- PWRCTRL\_CPU1\_TIMER\_TH\_01 0xfe00c1e0
- PWRCTRL\_CPU1\_TIMER\_TH\_23 0xfe00c1e4
- PWRCTRL\_CPU1\_TIMER\_TH\_45 0xfe00c1e8
- PWRCTRL\_CPU1\_TIMER\_TH\_67 0xfe00c1ec
- PWRCTRL\_CPU1\_TIMER\_TH\_89 0xfe00c1f0
- PWRCTRL\_CPU1\_IRQ\_MASK0 0xfe00c200
- PWRCTRL\_CPU1\_IRQ\_MASK1 0xfe00c204
- PWRCTRL\_CPU1\_IRQ\_MASK2 0xfe00c208
- PWRCTRL\_CPU1\_IRQ\_MASK3 0xfe00c20c
- PWRCTRL\_CPU1\_IRQ\_MASK4 0xfe00c210
- PWRCTRL\_CPU1\_IRQ\_MASK5 0xfe00c214
- PWRCTRL\_CPU1\_IRQ\_MASK6 0xfe00c218
- PWRCTRL\_CPU1\_IRQ\_MASK7 0xfe00c21c
- PWRCTRL\_CPU1\_IRQ\_MASK8 0xfe00c220
- PWRCTRL\_CPU1\_IRQ\_MASK9 0xfe00c224
- PWRCTRL\_CPU1\_IRQ\_MASK10 0xfe00c228
- PWRCTRL\_CPU1\_IRQ\_MASK11 0xfe00c22c
- PWRCTRL\_CPU1\_IRQ\_MASK12 0xfe00c230
- PWRCTRL\_CPU1\_IRQ\_MASK13 0xfe00c234
- PWRCTRL\_CPU1\_IRQ\_MASK14 0xfe00c238
- PWRCTRL\_CPU1\_IRQ\_MASK15 0xfe00c23c
- PWRCTRL\_CPU1\_MEMPD\_INIT\_SET 0xfe00c240
- PWRCTRL\_CPU1\_MEMPD\_OFF\_SET 0xfe00c244
- PWRCTRL\_CPU1\_MEMPD\_ON\_A\_SET 0xfe00c248
- PWRCTRL\_CPU1\_MEMPD\_ON\_B\_SET 0xfe00c24c
- PWRCTRL\_CPU1\_MEMPD\_ON\_C\_SET 0xfe00c250
- PWRCTRL\_CPU1\_MEMPD\_ON\_D\_SET 0xfe00c254
- PWRCTRL\_CPU1\_MEMPD\_STS 0xfe00c258

- PWRCTRL\_CPU1\_FSM\_STS0 0xfe00c25c
- PWRCTRL\_CPU1\_FSM\_STS1 0xfe00c260
- PWRCTRL\_CPU1\_FSM\_STS2 0xfe00c264
- PWRCTRL\_CPU1\_FSM\_START\_OFF 0xfe00c274
- PWRCTRL\_CPU1\_FSM\_START\_ON 0xfe00c278
- PWRCTRL\_CPU1\_FSM\_JUMP 0xfe00c27c
- PWRCTRL\_CPU2\_AUTO\_OFF\_CTRL0 0xfe00c280
- PWRCTRL\_CPU2\_AUTO\_OFF\_CTRL1 0xfe00c284
- PWRCTRL\_CPU2\_AUTO\_OFF\_CTRL2 0xfe00c288
- PWRCTRL\_CPU2\_AUTO\_OFF\_CTRL3 0xfe00c28c
- PWRCTRL\_CPU2\_AUTO\_OFF\_CTRL4 0xfe00c290
- PWRCTRL\_CPU2\_AUTO\_OFF\_CTRL5 0xfe00c294
- PWRCTRL\_CPU2\_TIMER\_TH\_01 0xfe00c2a0
- PWRCTRL\_CPU2\_TIMER\_TH\_23 0xfe00c2a4
- PWRCTRL\_CPU2\_TIMER\_TH\_45 0xfe00c2a8
- PWRCTRL\_CPU2\_TIMER\_TH\_67 0xfe00c2ac
- PWRCTRL\_CPU2\_TIMER\_TH\_89 0xfe00c2b0
- PWRCTRL\_CPU2\_IRQ\_MASK0 0xfe00c2c0
- PWRCTRL\_CPU2\_IRQ\_MASK1 0xfe00c2c4
- PWRCTRL\_CPU2\_IRQ\_MASK2 0xfe00c2c8
- PWRCTRL\_CPU2\_IRQ\_MASK3 0xfe00c2cc
- PWRCTRL\_CPU2\_IRQ\_MASK4 0xfe00c2d0
- PWRCTRL\_CPU2\_IRQ\_MASK5 0xfe00c2d4
- PWRCTRL\_CPU2\_IRQ\_MASK6 0xfe00c2d8
- PWRCTRL\_CPU2\_IRQ\_MASK7 0xfe00c2dc
- PWRCTRL\_CPU2\_IRQ\_MASK8 0xfe00c2e0
- PWRCTRL\_CPU2\_IRQ\_MASK9 0xfe00c2e4
- PWRCTRL\_CPU2\_IRQ\_MASK10 0xfe00c2e8
- PWRCTRL\_CPU2\_IRQ\_MASK11 0xfe00c2ec
- PWRCTRL\_CPU2\_IRQ\_MASK12 0xfe00c2f0
- PWRCTRL\_CPU2\_IRQ\_MASK13 0xfe00c2f4
- PWRCTRL\_CPU2\_IRQ\_MASK14 0xfe00c2f8
- PWRCTRL\_CPU2\_IRQ\_MASK15 0xfe00c2fc
- PWRCTRL\_CPU2\_MEMPD\_INIT\_SET 0xfe00c300
- PWRCTRL\_CPU2\_MEMPD\_OFF\_SET 0xfe00c304
- PWRCTRL\_CPU2\_MEMPD\_ON\_A\_SET 0xfe00c308
- PWRCTRL\_CPU2\_MEMPD\_ON\_B\_SET 0xfe00c30c
- PWRCTRL\_CPU2\_MEMPD\_ON\_C\_SET 0xfe00c310
- PWRCTRL\_CPU2\_MEMPD\_ON\_D\_SET 0xfe00c314
- PWRCTRL\_CPU2\_MEMPD\_STS 0xfe00c318
- PWRCTRL\_CPU2\_FSM\_STS0 0xfe00c31c
- PWRCTRL\_CPU2\_FSM\_STS1 0xfe00c320
- PWRCTRL\_CPU2\_FSM\_STS2 0xfe00c324
- PWRCTRL\_CPU2\_FSM\_START\_OFF 0xfe00c334



- PWRCTRL\_CPU2\_FSM\_START\_ON 0xfe00c338
- PWRCTRL\_CPU2\_FSM\_JUMP 0xfe00c33c
- PWRCTRL\_CPU3\_AUTO\_OFF\_CTRL0 0xfe00c340
- PWRCTRL\_CPU3\_AUTO\_OFF\_CTRL1 0xfe00c344
- PWRCTRL\_CPU3\_AUTO\_OFF\_CTRL2 0xfe00c348
- PWRCTRL\_CPU3\_AUTO\_OFF\_CTRL3 0xfe00c34c
- PWRCTRL\_CPU3\_AUTO\_OFF\_CTRL4 0xfe00c350
- PWRCTRL\_CPU3\_AUTO\_OFF\_CTRL5 0xfe00c354
- PWRCTRL\_CPU3\_TIMER\_TH\_01 0xfe00c360
- PWRCTRL\_CPU3\_TIMER\_TH\_23 0xfe00c364
- PWRCTRL\_CPU3\_TIMER\_TH\_45 0xfe00c368
- PWRCTRL\_CPU3\_TIMER\_TH\_67 0xfe00c36c
- PWRCTRL\_CPU3\_TIMER\_TH\_89 0xfe00c370
- PWRCTRL\_CPU3\_IRQ\_MASK0 0xfe00c380
- PWRCTRL\_CPU3\_IRQ\_MASK1 0xfe00c384
- PWRCTRL\_CPU3\_IRQ\_MASK2 0xfe00c388
- PWRCTRL\_CPU3\_IRQ\_MASK3 0xfe00c38c
- PWRCTRL\_CPU3\_IRQ\_MASK4 0xfe00c390
- PWRCTRL\_CPU3\_IRQ\_MASK5 0xfe00c394
- PWRCTRL\_CPU3\_IRQ\_MASK6 0xfe00c398
- PWRCTRL\_CPU3\_IRQ\_MASK7 0xfe00c39c
- PWRCTRL\_CPU3\_IRQ\_MASK8 0xfe00c3a0
- PWRCTRL\_CPU3\_IRQ\_MASK9 0xfe00c3a4
- PWRCTRL\_CPU3\_IRQ\_MASK10 0xfe00c3a8
- PWRCTRL\_CPU3\_IRQ\_MASK11 0xfe00c3ac
- PWRCTRL\_CPU3\_IRQ\_MASK12 0xfe00c3b0
- PWRCTRL\_CPU3\_IRQ\_MASK13 0xfe00c3b4
- PWRCTRL\_CPU3\_IRQ\_MASK14 0xfe00c3b8
- PWRCTRL\_CPU3\_IRQ\_MASK15 0xfe00c3bc
- PWRCTRL\_CPU3\_MEMPD\_INIT\_SET 0xfe00c3c0
- PWRCTRL\_CPU3\_MEMPD\_OFF\_SET 0xfe00c3c4
- PWRCTRL\_CPU3\_MEMPD\_ON\_A\_SET 0xfe00c3c8
- PWRCTRL\_CPU3\_MEMPD\_ON\_B\_SET 0xfe00c3cc
- PWRCTRL\_CPU3\_MEMPD\_ON\_C\_SET 0xfe00c3d0
- PWRCTRL\_CPU3\_MEMPD\_ON\_D\_SET 0xfe00c3d4
- PWRCTRL\_CPU3\_MEMPD\_STS 0xfe00c3d8
- PWRCTRL\_CPU3\_FSM\_STS0 0xfe00c3dc
- PWRCTRL\_CPU3\_FSM\_STS1 0xfe00c3e0
- PWRCTRL\_CPU3\_FSM\_STS2 0xfe00c3e4
- PWRCTRL\_CPU3\_FSM\_START\_OFF 0xfe00c3f4
- PWRCTRL\_CPU3\_FSM\_START\_ON 0xfe00c3f8
- PWRCTRL\_CPU3\_FSM\_JUMP 0xfe00c3fc
- PWRCTRL\_CPUTOP\_AUTO\_OFF\_CTRL0 0xfe00c400
- PWRCTRL\_CPUTOP\_AUTO\_OFF\_CTRL1 0xfe00c404

- PWRCTRL\_CPUTOP\_AUTO\_OFF\_CTRL2 0xfe00c408
- PWRCTRL\_CPUTOP\_AUTO\_OFF\_CTRL3 0xfe00c40c
- PWRCTRL\_CPUTOP\_AUTO\_OFF\_CTRL4 0xfe00c410
- PWRCTRL\_CPUTOP\_AUTO\_OFF\_CTRL5 0xfe00c414
- PWRCTRL\_CPUTOP\_TIMER\_TH\_01 0xfe00c420
- PWRCTRL\_CPUTOP\_TIMER\_TH\_23 0xfe00c424
- PWRCTRL\_CPUTOP\_TIMER\_TH\_45 0xfe00c428
- PWRCTRL\_CPUTOP\_TIMER\_TH\_67 0xfe00c42c
- PWRCTRL\_CPUTOP\_TIMER\_TH\_89 0xfe00c430
- PWRCTRL\_CPUTOP\_IRQ\_MASK0 0xfe00c440
- PWRCTRL\_CPUTOP\_IRQ\_MASK1 0xfe00c444
- PWRCTRL\_CPUTOP\_IRQ\_MASK2 0xfe00c448
- PWRCTRL\_CPUTOP\_IRQ\_MASK3 0xfe00c44c
- PWRCTRL\_CPUTOP\_IRQ\_MASK4 0xfe00c450
- PWRCTRL\_CPUTOP\_IRQ\_MASK5 0xfe00c454
- PWRCTRL\_CPUTOP\_IRQ\_MASK6 0xfe00c458
- PWRCTRL\_CPUTOP\_IRQ\_MASK7 0xfe00c45c
- PWRCTRL\_CPUTOP\_IRQ\_MASK8 0xfe00c460
- PWRCTRL\_CPUTOP\_IRQ\_MASK9 0xfe00c464
- PWRCTRL\_CPUTOP\_IRQ\_MASK10 0xfe00c468
- PWRCTRL\_CPUTOP\_IRQ\_MASK11 0xfe00c46c
- PWRCTRL\_CPUTOP\_IRQ\_MASK12 0xfe00c470
- PWRCTRL\_CPUTOP\_IRQ\_MASK13 0xfe00c474
- PWRCTRL\_CPUTOP\_IRQ\_MASK14 0xfe00c478
- PWRCTRL\_CPUTOP\_IRQ\_MASK15 0xfe00c47c
- PWRCTRL\_CPUTOP\_MEMPD\_INIT\_SET 0xfe00c480
- PWRCTRL\_CPUTOP\_MEMPD\_OFF\_SET 0xfe00c484
- PWRCTRL\_CPUTOP\_MEMPD\_ON\_A\_SET 0xfe00c488
- PWRCTRL\_CPUTOP\_MEMPD\_ON\_B\_SET 0xfe00c48c
- PWRCTRL\_CPUTOP\_MEMPD\_ON\_C\_SET 0xfe00c490
- PWRCTRL\_CPUTOP\_MEMPD\_ON\_D\_SET 0xfe00c494
- PWRCTRL\_CPUTOP\_MEMPD\_STS 0xfe00c498
- PWRCTRL\_CPUTOP\_FSM\_STS0 0xfe00c49c
- PWRCTRL\_CPUTOP\_FSM\_STS1 0xfe00c4a0
- PWRCTRL\_CPUTOP\_FSM\_STS2 0xfe00c4a4
- PWRCTRL\_CPUTOP\_FSM\_START\_OFF 0xfe00c4b4
- PWRCTRL\_CPUTOP\_FSM\_START\_ON 0xfe00c4b8
- PWRCTRL\_CPUTOP\_FSM\_JUMP 0xfe00c4bc
- PWRCTRL\_DSPA\_AUTO\_OFF\_CTRL0 0xfe00c4c0
- PWRCTRL\_DSPA\_AUTO\_OFF\_CTRL1 0xfe00c4c4
- PWRCTRL\_DSPA\_AUTO\_OFF\_CTRL2 0xfe00c4c8
- PWRCTRL\_DSPA\_AUTO\_OFF\_CTRL3 0xfe00c4cc
- PWRCTRL\_DSPA\_AUTO\_OFF\_CTRL4 0xfe00c4d0
- PWRCTRL\_DSPA\_AUTO\_OFF\_CTRL5 0xfe00c4d4

- PWRCTRL\_DSPA\_TIMER\_TH\_01 0xfe00c4e0
- PWRCTRL\_DSPA\_TIMER\_TH\_23 0xfe00c4e4
- PWRCTRL\_DSPA\_TIMER\_TH\_45 0xfe00c4e8
- PWRCTRL\_DSPA\_TIMER\_TH\_67 0xfe00c4ec
- PWRCTRL\_DSPA\_TIMER\_TH\_89 0xfe00c4f0
- PWRCTRL\_DSPA\_IRQ\_MASK0 0xfe00c500
- PWRCTRL\_DSPA\_IRQ\_MASK1 0xfe00c504
- PWRCTRL\_DSPA\_IRQ\_MASK2 0xfe00c508
- PWRCTRL\_DSPA\_IRQ\_MASK3 0xfe00c50c
- PWRCTRL\_DSPA\_IRQ\_MASK4 0xfe00c510
- PWRCTRL\_DSPA\_IRQ\_MASK5 0xfe00c514
- PWRCTRL\_DSPA\_IRQ\_MASK6 0xfe00c518
- PWRCTRL\_DSPA\_IRQ\_MASK7 0xfe00c51c
- PWRCTRL\_DSPA\_IRQ\_MASK8 0xfe00c520
- PWRCTRL\_DSPA\_IRQ\_MASK9 0xfe00c524
- PWRCTRL\_DSPA\_IRQ\_MASK10 0xfe00c528
- PWRCTRL\_DSPA\_IRQ\_MASK11 0xfe00c52c
- PWRCTRL\_DSPA\_IRQ\_MASK12 0xfe00c530
- PWRCTRL\_DSPA\_IRQ\_MASK13 0xfe00c534
- PWRCTRL\_DSPA\_IRQ\_MASK14 0xfe00c538
- PWRCTRL\_DSPA\_IRQ\_MASK15 0xfe00c53c
- PWRCTRL\_DSPA\_MEMPD\_INIT\_SET 0xfe00c540
- PWRCTRL\_DSPA\_MEMPD\_OFF\_SET 0xfe00c544
- PWRCTRL\_DSPA\_MEMPD\_ON\_A\_SET 0xfe00c548
- PWRCTRL\_DSPA\_MEMPD\_ON\_B\_SET 0xfe00c54c
- PWRCTRL\_DSPA\_MEMPD\_ON\_C\_SET 0xfe00c550
- PWRCTRL\_DSPA\_MEMPD\_ON\_D\_SET 0xfe00c554
- PWRCTRL\_DSPA\_MEMPD\_STS 0xfe00c558
- PWRCTRL\_DSPA\_FSM\_STS0 0xfe00c55c
- PWRCTRL\_DSPA\_FSM\_STS1 0xfe00c560
- PWRCTRL\_DSPA\_FSM\_STS2 0xfe00c564
- PWRCTRL\_DSPA\_FSM\_START\_OFF 0xfe00c574
- PWRCTRL\_DSPA\_FSM\_START\_ON 0xfe00c578
- PWRCTRL\_DSPA\_FSM\_JUMP 0xfe00c57c
- PWRCTRL\_DSPB\_AUTO\_OFF\_CTRL0 0xfe00c580
- PWRCTRL\_DSPB\_AUTO\_OFF\_CTRL1 0xfe00c584
- PWRCTRL\_DSPB\_AUTO\_OFF\_CTRL2 0xfe00c588
- PWRCTRL\_DSPB\_AUTO\_OFF\_CTRL3 0xfe00c58c
- PWRCTRL\_DSPB\_AUTO\_OFF\_CTRL4 0xfe00c590
- PWRCTRL\_DSPB\_AUTO\_OFF\_CTRL5 0xfe00c594
- PWRCTRL\_DSPB\_TIMER\_TH\_01 0xfe00c5a0
- PWRCTRL\_DSPB\_TIMER\_TH\_23 0xfe00c5a4
- PWRCTRL\_DSPB\_TIMER\_TH\_45 0xfe00c5a8
- PWRCTRL\_DSPB\_TIMER\_TH\_67 0xfe00c5ac

- PWRCTRL\_DSPB\_TIMER\_TH\_89 0xfe00c5b0
- PWRCTRL\_DSPB\_IRQ\_MASK0 0xfe00c5c0
- PWRCTRL\_DSPB\_IRQ\_MASK1 0xfe00c5c4
- PWRCTRL\_DSPB\_IRQ\_MASK2 0xfe00c5c8
- PWRCTRL\_DSPB\_IRQ\_MASK3 0xfe00c5cc
- PWRCTRL\_DSPB\_IRQ\_MASK4 0xfe00c5d0
- PWRCTRL\_DSPB\_IRQ\_MASK5 0xfe00c5d4
- PWRCTRL\_DSPB\_IRQ\_MASK6 0xfe00c5d8
- PWRCTRL\_DSPB\_IRQ\_MASK7 0xfe00c5dc
- PWRCTRL\_DSPB\_IRQ\_MASK8 0xfe00c5e0
- PWRCTRL\_DSPB\_IRQ\_MASK9 0xfe00c5e4
- PWRCTRL\_DSPB\_IRQ\_MASK10 0xfe00c5e8
- PWRCTRL\_DSPB\_IRQ\_MASK11 0xfe00c5ec
- PWRCTRL\_DSPB\_IRQ\_MASK12 0xfe00c5f0
- PWRCTRL\_DSPB\_IRQ\_MASK13 0xfe00c5f4
- PWRCTRL\_DSPB\_IRQ\_MASK14 0xfe00c5f8
- PWRCTRL\_DSPB\_IRQ\_MASK15 0xfe00c5fc
- PWRCTRL\_DSPB\_MEMPD\_INIT\_SET 0xfe00c600
- PWRCTRL\_DSPB\_MEMPD\_OFF\_SET 0xfe00c604
- PWRCTRL\_DSPB\_MEMPD\_ON\_A\_SET 0xfe00c608
- PWRCTRL\_DSPB\_MEMPD\_ON\_B\_SET 0xfe00c60c
- PWRCTRL\_DSPB\_MEMPD\_ON\_C\_SET 0xfe00c610
- PWRCTRL\_DSPB\_MEMPD\_ON\_D\_SET 0xfe00c614
- PWRCTRL\_DSPB\_MEMPD\_STS 0xfe00c618
- PWRCTRL\_DSPB\_FSM\_STS0 0xfe00c61c
- PWRCTRL\_DSPB\_FSM\_STS1 0xfe00c620
- PWRCTRL\_DSPB\_FSM\_STS2 0xfe00c624
- PWRCTRL\_DSPB\_FSM\_START\_OFF 0xfe00c634
- PWRCTRL\_DSPB\_FSM\_START\_ON 0xfe00c638
- PWRCTRL\_DSPB\_FSM\_JUMP 0xfe00c63c
- PWRCTRL\_A730\_AUTO\_OFF\_CTRL0 0xfe00c640
- PWRCTRL\_A730\_AUTO\_OFF\_CTRL1 0xfe00c644
- PWRCTRL\_A730\_AUTO\_OFF\_CTRL2 0xfe00c648
- PWRCTRL\_A730\_AUTO\_OFF\_CTRL3 0xfe00c64c
- PWRCTRL\_A730\_AUTO\_OFF\_CTRL4 0xfe00c650
- PWRCTRL\_A730\_AUTO\_OFF\_CTRL5 0xfe00c654
- PWRCTRL\_A730\_TIMER\_TH\_01 0xfe00c660
- PWRCTRL\_A730\_TIMER\_TH\_23 0xfe00c664
- PWRCTRL\_A730\_TIMER\_TH\_45 0xfe00c668
- PWRCTRL\_A730\_TIMER\_TH\_67 0xfe00c66c
- PWRCTRL\_A730\_TIMER\_TH\_89 0xfe00c670
- PWRCTRL\_A730\_IRQ\_MASK0 0xfe00c680
- PWRCTRL\_A730\_IRQ\_MASK1 0xfe00c684
- PWRCTRL\_A730\_IRQ\_MASK2 0xfe00c688

- PWRCTRL\_A730\_IRQ\_MASK3 0xfe00c68c
- PWRCTRL\_A730\_IRQ\_MASK4 0xfe00c690
- PWRCTRL\_A730\_IRQ\_MASK5 0xfe00c694
- PWRCTRL\_A730\_IRQ\_MASK6 0xfe00c698
- PWRCTRL\_A730\_IRQ\_MASK7 0xfe00c69c
- PWRCTRL\_A730\_IRQ\_MASK8 0xfe00c6a0
- PWRCTRL\_A730\_IRQ\_MASK9 0xfe00c6a4
- PWRCTRL\_A730\_IRQ\_MASK10 0xfe00c6a8
- PWRCTRL\_A730\_IRQ\_MASK11 0xfe00c6ac
- PWRCTRL\_A730\_IRQ\_MASK12 0xfe00c6b0
- PWRCTRL\_A730\_IRQ\_MASK13 0xfe00c6b4
- PWRCTRL\_A730\_IRQ\_MASK14 0xfe00c6b8
- PWRCTRL\_A730\_IRQ\_MASK15 0xfe00c6bc
- PWRCTRL\_A730\_MEMPD\_INIT\_SET 0xfe00c6c0
- PWRCTRL\_A730\_MEMPD\_OFF\_SET 0xfe00c6c4
- PWRCTRL\_A730\_MEMPD\_ON\_A\_SET 0xfe00c6c8
- PWRCTRL\_A730\_MEMPD\_ON\_B\_SET 0xfe00c6cc
- PWRCTRL\_A730\_MEMPD\_ON\_C\_SET 0xfe00c6d0
- PWRCTRL\_A730\_MEMPD\_ON\_D\_SET 0xfe00c6d4
- PWRCTRL\_A730\_MEMPD\_STS 0xfe00c6d8
- PWRCTRL\_A730\_FSM\_STS0 0xfe00c6dc
- PWRCTRL\_A730\_FSM\_STS1 0xfe00c6e0
- PWRCTRL\_A730\_FSM\_STS2 0xfe00c6e4
- PWRCTRL\_A730\_FSM\_START\_OFF 0xfe00c6f4
- PWRCTRL\_A730\_FSM\_START\_ON 0xfe00c6f8
- PWRCTRL\_A730\_FSM\_JUMP 0xfe00c6fc
- PWRCTRL\_A731\_AUTO\_OFF\_CTRL0 0xfe00c700
- PWRCTRL\_A731\_AUTO\_OFF\_CTRL1 0xfe00c704
- PWRCTRL\_A731\_AUTO\_OFF\_CTRL2 0xfe00c708
- PWRCTRL\_A731\_AUTO\_OFF\_CTRL3 0xfe00c70c
- PWRCTRL\_A731\_AUTO\_OFF\_CTRL4 0xfe00c710
- PWRCTRL\_A731\_AUTO\_OFF\_CTRL5 0xfe00c714
- PWRCTRL\_A731\_TIMER\_TH\_01 0xfe00c720
- PWRCTRL\_A731\_TIMER\_TH\_23 0xfe00c724
- PWRCTRL\_A731\_TIMER\_TH\_45 0xfe00c728
- PWRCTRL\_A731\_TIMER\_TH\_67 0xfe00c72c
- PWRCTRL\_A731\_TIMER\_TH\_89 0xfe00c730
- PWRCTRL\_A731\_IRQ\_MASK0 0xfe00c740
- PWRCTRL\_A731\_IRQ\_MASK1 0xfe00c744
- PWRCTRL\_A731\_IRQ\_MASK2 0xfe00c748
- PWRCTRL\_A731\_IRQ\_MASK3 0xfe00c74c
- PWRCTRL\_A731\_IRQ\_MASK4 0xfe00c750
- PWRCTRL\_A731\_IRQ\_MASK5 0xfe00c754
- PWRCTRL\_A731\_IRQ\_MASK6 0xfe00c758

- PWRCTRL\_A731\_IRQ\_MASK7 0xfe00c75c
- PWRCTRL\_A731\_IRQ\_MASK8 0xfe00c760
- PWRCTRL\_A731\_IRQ\_MASK9 0xfe00c764
- PWRCTRL\_A731\_IRQ\_MASK10 0xfe00c768
- PWRCTRL\_A731\_IRQ\_MASK11 0xfe00c76c
- PWRCTRL\_A731\_IRQ\_MASK12 0xfe00c770
- PWRCTRL\_A731\_IRQ\_MASK13 0xfe00c774
- PWRCTRL\_A731\_IRQ\_MASK14 0xfe00c778
- PWRCTRL\_A731\_IRQ\_MASK15 0xfe00c77c
- PWRCTRL\_A731\_MEMPD\_INIT\_SET 0xfe00c780
- PWRCTRL\_A731\_MEMPD\_OFF\_SET 0xfe00c784
- PWRCTRL\_A731\_MEMPD\_ON\_A\_SET 0xfe00c788
- PWRCTRL\_A731\_MEMPD\_ON\_B\_SET 0xfe00c78c
- PWRCTRL\_A731\_MEMPD\_ON\_C\_SET 0xfe00c790
- PWRCTRL\_A731\_MEMPD\_ON\_D\_SET 0xfe00c794
- PWRCTRL\_A731\_MEMPD\_STS 0xfe00c798
- PWRCTRL\_A731\_FSM\_STS0 0xfe00c79c
- PWRCTRL\_A731\_FSM\_STS1 0xfe00c7a0
- PWRCTRL\_A731\_FSM\_STS2 0xfe00c7a4
- PWRCTRL\_A731\_FSM\_START\_OFF 0xfe00c7b4
- PWRCTRL\_A731\_FSM\_START\_ON 0xfe00c7b8
- PWRCTRL\_A731\_FSM\_JUMP 0xfe00c7bc
- PWRCTRL\_A732\_AUTO\_OFF\_CTRL0 0xfe00c7c0
- PWRCTRL\_A732\_AUTO\_OFF\_CTRL1 0xfe00c7c4
- PWRCTRL\_A732\_AUTO\_OFF\_CTRL2 0xfe00c7c8
- PWRCTRL\_A732\_AUTO\_OFF\_CTRL3 0xfe00c7cc
- PWRCTRL\_A732\_AUTO\_OFF\_CTRL4 0xfe00c7d0
- PWRCTRL\_A732\_AUTO\_OFF\_CTRL5 0xfe00c7d4
- PWRCTRL\_A732\_TIMER\_TH\_01 0xfe00c7e0
- PWRCTRL\_A732\_TIMER\_TH\_23 0xfe00c7e4
- PWRCTRL\_A732\_TIMER\_TH\_45 0xfe00c7e8
- PWRCTRL\_A732\_TIMER\_TH\_67 0xfe00c7ec
- PWRCTRL\_A732\_TIMER\_TH\_89 0xfe00c7f0
- PWRCTRL\_A732\_IRQ\_MASK0 0xfe00c800
- PWRCTRL\_A732\_IRQ\_MASK1 0xfe00c804
- PWRCTRL\_A732\_IRQ\_MASK2 0xfe00c808
- PWRCTRL\_A732\_IRQ\_MASK3 0xfe00c80c
- PWRCTRL\_A732\_IRQ\_MASK4 0xfe00c810
- PWRCTRL\_A732\_IRQ\_MASK5 0xfe00c814
- PWRCTRL\_A732\_IRQ\_MASK6 0xfe00c818
- PWRCTRL\_A732\_IRQ\_MASK7 0xfe00c81c
- PWRCTRL\_A732\_IRQ\_MASK8 0xfe00c820
- PWRCTRL\_A732\_IRQ\_MASK9 0xfe00c824
- PWRCTRL\_A732\_IRQ\_MASK10 0xfe00c828

- PWRCTRL\_A732\_IRQ\_MASK11 0xfe00c82c
- PWRCTRL\_A732\_IRQ\_MASK12 0xfe00c830
- PWRCTRL\_A732\_IRQ\_MASK13 0xfe00c834
- PWRCTRL\_A732\_IRQ\_MASK14 0xfe00c838
- PWRCTRL\_A732\_IRQ\_MASK15 0xfe00c83c
- PWRCTRL\_A732\_MEMPD\_INIT\_SET 0xfe00c840
- PWRCTRL\_A732\_MEMPD\_OFF\_SET 0xfe00c844
- PWRCTRL\_A732\_MEMPD\_ON\_A\_SET 0xfe00c848
- PWRCTRL\_A732\_MEMPD\_ON\_B\_SET 0xfe00c84c
- PWRCTRL\_A732\_MEMPD\_ON\_C\_SET 0xfe00c850
- PWRCTRL\_A732\_MEMPD\_ON\_D\_SET 0xfe00c854
- PWRCTRL\_A732\_MEMPD\_STS 0xfe00c858
- PWRCTRL\_A732\_FSM\_STS0 0xfe00c85c
- PWRCTRL\_A732\_FSM\_STS1 0xfe00c860
- PWRCTRL\_A732\_FSM\_STS2 0xfe00c864
- PWRCTRL\_A732\_FSM\_START\_OFF 0xfe00c874
- PWRCTRL\_A732\_FSM\_START\_ON 0xfe00c878
- PWRCTRL\_A732\_FSM\_JUMP 0xfe00c87c
- PWRCTRL\_A733\_AUTO\_OFF\_CTRL0 0xfe00c880
- PWRCTRL\_A733\_AUTO\_OFF\_CTRL1 0xfe00c884
- PWRCTRL\_A733\_AUTO\_OFF\_CTRL2 0xfe00c888
- PWRCTRL\_A733\_AUTO\_OFF\_CTRL3 0xfe00c88c
- PWRCTRL\_A733\_AUTO\_OFF\_CTRL4 0xfe00c890
- PWRCTRL\_A733\_AUTO\_OFF\_CTRL5 0xfe00c894
- PWRCTRL\_A733\_TIMER\_TH\_01 0xfe00c8a0
- PWRCTRL\_A733\_TIMER\_TH\_23 0xfe00c8a4
- PWRCTRL\_A733\_TIMER\_TH\_45 0xfe00c8a8
- PWRCTRL\_A733\_TIMER\_TH\_67 0xfe00c8ac
- PWRCTRL\_A733\_TIMER\_TH\_89 0xfe00c8b0
- PWRCTRL\_A733\_IRQ\_MASK0 0xfe00c8c0
- PWRCTRL\_A733\_IRQ\_MASK1 0xfe00c8c4
- PWRCTRL\_A733\_IRQ\_MASK2 0xfe00c8c8
- PWRCTRL\_A733\_IRQ\_MASK3 0xfe00c8cc
- PWRCTRL\_A733\_IRQ\_MASK4 0xfe00c8d0
- PWRCTRL\_A733\_IRQ\_MASK5 0xfe00c8d4
- PWRCTRL\_A733\_IRQ\_MASK6 0xfe00c8d8
- PWRCTRL\_A733\_IRQ\_MASK7 0xfe00c8dc
- PWRCTRL\_A733\_IRQ\_MASK8 0xfe00c8e0
- PWRCTRL\_A733\_IRQ\_MASK9 0xfe00c8e4
- PWRCTRL\_A733\_IRQ\_MASK10 0xfe00c8e8
- PWRCTRL\_A733\_IRQ\_MASK11 0xfe00c8ec
- PWRCTRL\_A733\_IRQ\_MASK12 0xfe00c8f0
- PWRCTRL\_A733\_IRQ\_MASK13 0xfe00c8f4
- PWRCTRL\_A733\_IRQ\_MASK14 0xfe00c8f8

- PWRCTRL\_A733\_IRQ\_MASK15 0xfe00c8fc
- PWRCTRL\_A733\_MEMPD\_INIT\_SET 0xfe00c900
- PWRCTRL\_A733\_MEMPD\_OFF\_SET 0xfe00c904
- PWRCTRL\_A733\_MEMPD\_ON\_A\_SET 0xfe00c908
- PWRCTRL\_A733\_MEMPD\_ON\_B\_SET 0xfe00c90c
- PWRCTRL\_A733\_MEMPD\_ON\_C\_SET 0xfe00c910
- PWRCTRL\_A733\_MEMPD\_ON\_D\_SET 0xfe00c914
- PWRCTRL\_A733\_MEMPD\_STS 0xfe00c918
- PWRCTRL\_A733\_FSM\_STS0 0xfe00c91c
- PWRCTRL\_A733\_FSM\_STS1 0xfe00c920
- PWRCTRL\_A733\_FSM\_STS2 0xfe00c924
- PWRCTRL\_A733\_FSM\_START\_OFF 0xfe00c934
- PWRCTRL\_A733\_FSM\_START\_ON 0xfe00c938
- PWRCTRL\_A733\_FSM\_JUMP 0xfe00c93c
- PWRCTRL\_A73TOP\_AUTO\_OFF\_CTRL0 0xfe00c940
- PWRCTRL\_A73TOP\_AUTO\_OFF\_CTRL1 0xfe00c944
- PWRCTRL\_A73TOP\_AUTO\_OFF\_CTRL2 0xfe00c948
- PWRCTRL\_A73TOP\_AUTO\_OFF\_CTRL3 0xfe00c94c
- PWRCTRL\_A73TOP\_AUTO\_OFF\_CTRL4 0xfe00c950
- PWRCTRL\_A73TOP\_AUTO\_OFF\_CTRL5 0xfe00c954
- PWRCTRL\_A73TOP\_TIMER\_TH\_01 0xfe00c960
- PWRCTRL\_A73TOP\_TIMER\_TH\_23 0xfe00c964
- PWRCTRL\_A73TOP\_TIMER\_TH\_45 0xfe00c968
- PWRCTRL\_A73TOP\_TIMER\_TH\_67 0xfe00c96c
- PWRCTRL\_A73TOP\_TIMER\_TH\_89 0xfe00c970
- PWRCTRL\_A73TOP\_IRQ\_MASK0 0xfe00c980
- PWRCTRL\_A73TOP\_IRQ\_MASK1 0xfe00c984
- PWRCTRL\_A73TOP\_IRQ\_MASK2 0xfe00c988
- PWRCTRL\_A73TOP\_IRQ\_MASK3 0xfe00c98c
- PWRCTRL\_A73TOP\_IRQ\_MASK4 0xfe00c990
- PWRCTRL\_A73TOP\_IRQ\_MASK5 0xfe00c994
- PWRCTRL\_A73TOP\_IRQ\_MASK6 0xfe00c998
- PWRCTRL\_A73TOP\_IRQ\_MASK7 0xfe00c99c
- PWRCTRL\_A73TOP\_IRQ\_MASK8 0xfe00c9a0
- PWRCTRL\_A73TOP\_IRQ\_MASK9 0xfe00c9a4
- PWRCTRL\_A73TOP\_IRQ\_MASK10 0xfe00c9a8
- PWRCTRL\_A73TOP\_IRQ\_MASK11 0xfe00c9ac
- PWRCTRL\_A73TOP\_IRQ\_MASK12 0xfe00c9b0
- PWRCTRL\_A73TOP\_IRQ\_MASK13 0xfe00c9b4
- PWRCTRL\_A73TOP\_IRQ\_MASK14 0xfe00c9b8
- PWRCTRL\_A73TOP\_IRQ\_MASK15 0xfe00c9bc
- PWRCTRL\_A73TOP\_MEMPD\_INIT\_SET 0xfe00c9c0
- PWRCTRL\_A73TOP\_MEMPD\_OFF\_SET 0xfe00c9c4
- PWRCTRL\_A73TOP\_MEMPD\_ON\_A\_SET 0xfe00c9c8



- PWRCTRL\_A73TOP\_MEMPD\_ON\_B\_SET 0xfe00c9cc
- PWRCTRL\_A73TOP\_MEMPD\_ON\_C\_SET 0xfe00c9d0
- PWRCTRL\_A73TOP\_MEMPD\_ON\_D\_SET 0xfe00c9d4
- PWRCTRL\_A73TOP\_MEMPD\_STS 0xfe00c9d8
- PWRCTRL\_A73TOP\_FSM\_STS0 0xfe00c9dc
- PWRCTRL\_A73TOP\_FSM\_STS1 0xfe00c9e0
- PWRCTRL\_A73TOP\_FSM\_STS2 0xfe00c9e4
- PWRCTRL\_A73TOP\_FSM\_START\_OFF 0xfe00c9f4
- PWRCTRL\_A73TOP\_FSM\_START\_ON 0xfe00c9f8
- PWRCTRL\_A73TOP\_FSM\_JUMP 0xfe00c9fc

## Register Description

### PWRCTRL\_PWR\_ACK0

Read only, the status of each power domain.

### PWRCTRL\_ISO\_EN0

isolation the interface of each power domain.

### PWRCTRL\_PWR\_OFF0

power off each power domain

**Table 7-1 PWRCTRL\_PWR\_OFF0 Control Bit**

Bit	Module	Default
31	DSI1	OFF
30	EDP1	OFF
29	EDP0	OFF
28	SPICC5	OFF
27	SPICC4	OFF
26	SPICC3	OFF
25	SPICC2	OFF
24	SPICC1	
23	SPICC0	OFF
22	DSI0	OFF
21		
20	NIC3	OFF
19	NIC2	ON
18	NOC	ON
17	DMC1	ON
16	DMC0	ON
15	DDR1	ON
14	DDR0	ON
13		

Bit	Module	Default
12	NNA	OFF
11	NNA	OFF
10	NNA	OFF
9	NNA	OFF
8	NNA	OFF
7		
6		
5		
4	MALI(iso/ack)	
3	MALI(ack)	
2	MALI(ack)	
1	MALI(ack)	
0	MALI(ack)	



#### Note

Mali power off/on inside of IP, ISO\_EN1 [4] is iso\_en\_mali\_top, ACK1 [4:0] are pwr\_ack\_mali [4:0].

#### PWRCTRL\_PWR\_ACK1

read only, the status of each power domain.

#### PWRCTRL\_ISO\_EN1

isolation the interface of each power domain.

#### PWRCTRL\_PWR\_OFF1

power off each power domain

**Table 7-2 PWRCTRL\_MEM\_PD0**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF-FFFF	DOS_VDEC

**Table 7-3 PWRCTRL\_MEM\_PD1**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF-FFFF	DOS_HCODEC

**Table 7-4 PWRCTRL\_MEM\_PD2**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	DOS_HEVC

**Table 7-5 PWRCTRL\_MEM\_PD3**

Bit(s)	R/W	Default	Description
31:0	R/W	0x00	reserved
25:24	R/W	0x0	NOC
23:0	R/W	0x0000- 00	reserved

**Table 7-6 PWRCTRL\_MEM\_PD4**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	AUDIO

**Table 7-7 PWRCTRL\_MEM\_PD5**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	vpu

**Table 7-8 PWRCTRL\_MEM\_PD6**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	vpu

**Table 7-9 PWRCTRL\_MEM\_PD7**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	vpu

**Table 7-10 PWRCTRL\_MEM\_PD8**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	vpu

**Table 7-11 PWRCTRL\_MEM\_PD9**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	vpu

**Table 7-12 PWRCTRL\_MEM\_PD10**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	vpu

**Table 7-13 PWRCTRL\_MEM\_PD11**

Bit(s)	R/W	Default	Description
31:30	R/W	0x0	USBCTRL
29:26	R/W	0xF	PCIECTRL
25:18	R/W	0xFF	GE2D
17:16	R/W	0x0	reserved
15:8	R/W	0xFF	HDMI
7:6	R/W	0x0	reserved
5:4	R/W	0x0	AUCPU
3:2	R/W	0x0	ETH
1:0	R/W	0x0	reserved

**Table 7-14 PWRCTRL\_MEM\_PD12**

Bit(s)	R/W	Default	Description
31:16	R/W	0x0	RAMB
15:0	R/W	0x0	RAMA

**Table 7-15 PWRCTRL\_MEM\_PD13**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	ISP

**Table 7-16 PWRCTRL\_MEM\_PD14**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	ISP

**Table 7-17 PWRCTRL\_MEM\_PD15**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	ISP

**Table 7-18 PWRCTRL\_MEM\_PD16**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	ISP

**Table 7-19 PWRCTRL\_MEM\_PD17**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	ISP

**Table 7-20 PWRCTRL\_MEM\_PD18**

Bit(s)	R/W	Default	Description
31:30	R/W	0x3	reserved
29:28	R/W	0x3	edp0_txctrl_mem_pd
27:26	R/W	0x3	edp1_txctrl_mem_pd
25:24	R/W	0x3	gdc_mem_pd
23:20	R/W	0xf	dewarp_mem_pd
19:18	R/W	0x3	spicc0_mem_pd
17:16	R/W	0x3	Spicc1_mem_pd
15:14	R/W	0x3	Spicc2_mem_pd
13:12	R/W	0x3	Spicc3_mem_pd
11:10	R/W	0x3	Spicc4_mem_pd
9:8	R/W	0x3	Spicc5_mem_pd
7:6	R/W	0x3	reserved
5:4	R/W	0x0	sdioa_mem_pd
3:2	R/W	0x0	sdiob_mem_pd
1:0	R/W	0x0	emmc_mem_pd

**Table 7-21 PWRCTRL\_MEM\_PD19**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	WAVE

**Table 7-22 PWRCTRL\_MEM\_PD20**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	WAVE

**Table 7-23 PWRCTRL\_MEM\_PD21**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	WAVE

**Table 7-24 PWRCTRL\_MEM\_PD22**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	anakin_mem_pd

**Table 7-25 PWRCTRL\_MEM\_PD23**

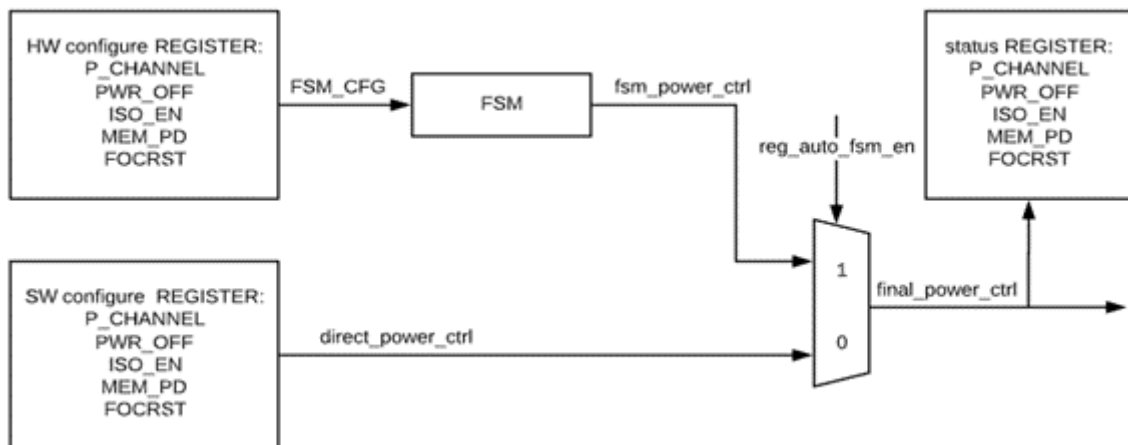
Bit(s)	R/W	Default	Description
31:28	R/W	0xF	anakin_mem_pd
27:26	R/W	0x3	dsi_a_mem_pd
25:24	R/W	0x3	dsi_b_mem_pd
23:8	R/W	0xff	reserved
7:6	R/W	0x0	dmc0_mem_pd
5:4	R/W	0x0	dmc1_mem_pd
3:2	R/W	0x0	ddr0_mem_pd
1:0	R/W	0x0	ddr1_mem_pd

**Table 7-26 PWRCTRL\_MEM\_PD24**

Bit(s)	R/W	Default	Description
31:0	R/W	0xFFFF- FFFF	hdmirx_mem_pd

For each A73/A53/DSP power domain that needs to power off by itself, so PWR\_CTRL contains 7 PWR\_FSM which can update the power related signals by hardware.

Figure 7-5 PWR\_FSM Signal Flow

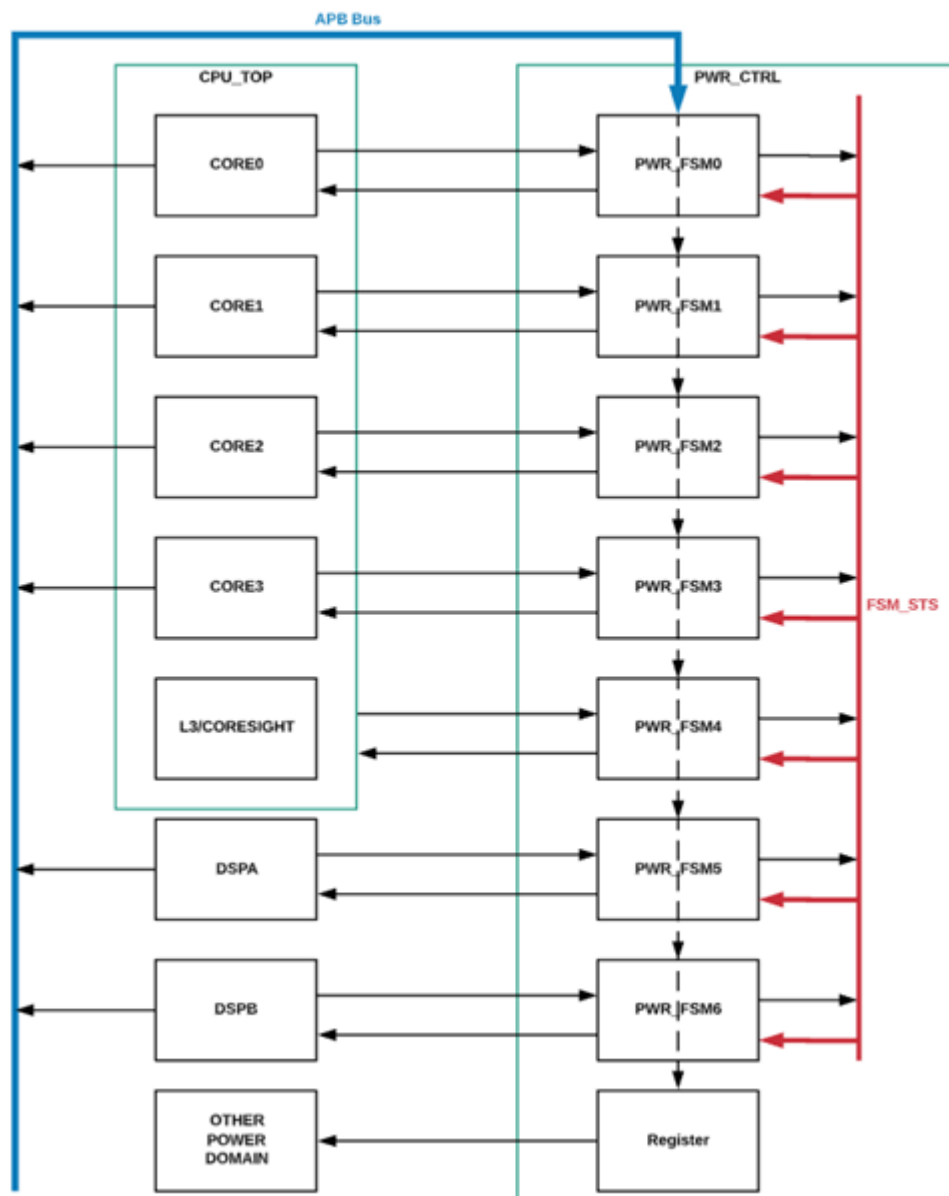


We can turn off/on these 7 cpu/dsp power domains in serial:

1. Turn off `cpu_core0`
2. Turn off `cpu_top`
1. Turn on `cpu_top`
2. Turn on `cpu_core0`

so all `FSM_STS` will send to each FSM.

Figure 7-6 FSM Function



Here's an example of HW(PWR\_FSM) work flow:

1. Configure all related register
2. Start\_pwr\_off
3. Wait for other pwr\_fsm status, check is it finished power off flow
4. Start handshake if need;
5. Start power off flow by sequence;
6. Wait for interrupt to start power on;
7. Wait for other pwr\_fsm status, check is it finished power on flow;
8. Start power on flow by sequence;
9. Start handshake if need;



Figure 7-7 FSM Work Flow Chart

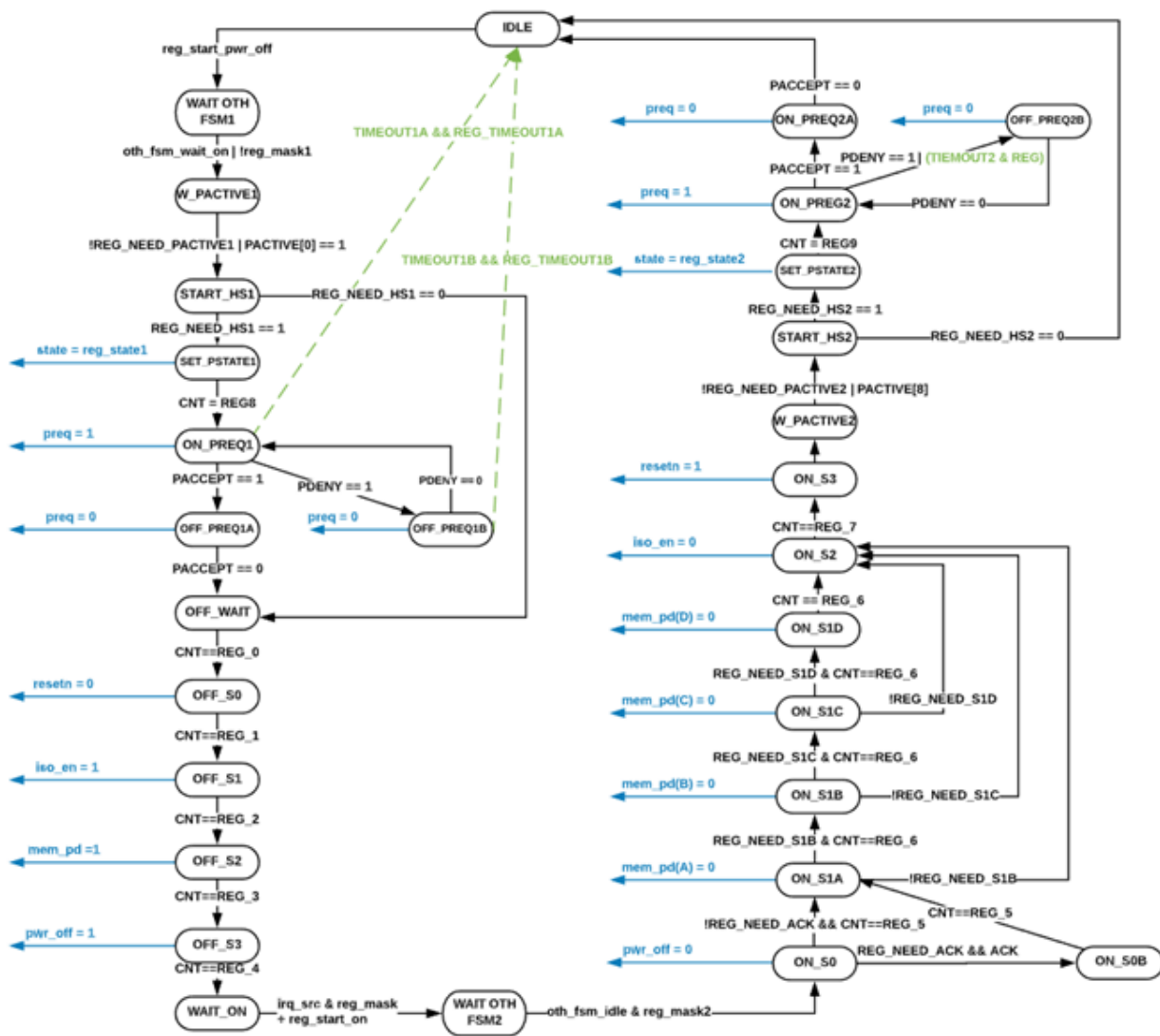


Table 7-27 PWRCTRL\_AUTO\_OFF\_CTRL0

Bits	R/W	Default	Description
31	R/W	0	REG_FSM_EN
30	R/W	0	REG_GATE_ALL_ON; force all clock gate are on;
29	R/W	0	REG_NEED_ACK_EN; set 1 ,ON_S0 will jump to ON_S0B
27	R/W	0	REG_ACK_INV, set 1, will invert PWR_ACK;
25:24	R/W	0	REG_TIMER_CNT_TICK_SEL; 0:timer_tick = sys_clk; 1:timer_tick = 1us; 2:timer_tick = 10us; 3:timer_tick = 100us;
23	R/W	0	REG_NEED_ON_S1B; set 1, ON_S1A will jump to ON_S1B;
22	R/W	0	REG_NEED_ON_S1C; set 1, ON_S1B will jump to ON_S1C;
21	R/W	0	REG_NEED_ON_S1D; set 1, ON_S1C will jump to ON_S1D;

Bits	R/W	Default	Description
20	R/W	0	REG_SET_PSTATE; set 1, force pstate = REG_PSTATE;
19	R/W	0	REG_ON_NEED_HS; set 1, START_HS2 will jump to SET_PSTATE2;
18	R/W	0	REG_ON_NEED_PACTIVE; set 1 will force WAIT_PACTIVE2 jump to START_HS2;
17	R/W	0	REG_OFF_NEED_HS; set 1, START_HS1 will jump to SET_PSTATE1;
16	R/W	0	REG_OFF_NEED_PACTIVE; set 1 will force WAIT_PACTIVE1 jump to START_HS1;
11	R/W	0	REG_PWROFF_OFF_SET; PWR_OFF value when FSM_STS = OFF_S3;
10	R/W	0	REG_PWROFF_ON_SET; PWR_OFF value when FSM_STS = ON_S0;
8	R/W	0	REG_PWROFF_INIT_SET; PWR_OFF value when REG_FSM_EN = 0;
7	R/W	0	REG_ISOEN_OFF_SET; ISO_EN value when FSM_STS = OFF_S1;
6	R/W	0	REG_ISOEN_ON_SET; ISO_EN value when FSM_STS = ON_S2;
4	R/W	0	REG_ISOEN_INIT_SET; ISO_EN value when REG_FSM_EN = 0;
3	R/W	0	REG_FOCRST_OFF_SET; FOC_RST value when FSM_STS = OFF_S0;
2	R/W	0	REG_FOCRST_ON_SET; FOC_RST value when FSM_STS = ON_S3;
0	R/W	0	REG_FOCRST_INIT_SET; FOC_RST value when REG_FSM_EN = 0;

Table 7-28 PWRCTRL\_\*\*\*\_AUTO\_OFF\_CTRL1

Bits	R/W	Default	Description
30:24	R/W	0	reg_timeout_cnt_th_off_b
20:16	R/W	0	reg_fsm_jump_sts
15:8	R/W	0	reg_oth_fsm_on_mask; bit7: reserved; bit6: set 1, FSM_STS will move to ON_S0 when DSPB is IDLE bit5: set 1, FSM_STS will move to ON_S0 when DSPA is IDLE bit4: set 1, FSM_STS will move to ON_S0 when CPU_TOP is IDLE bit3: set 1, FSM_STS will move to ON_S0 when CPU_CORE3 is IDLE bit2: set 1, FSM_STS will move to ON_S0 when CPU_CORE2 is IDLE bit1: set 1, FSM_STS will move to ON_S0 when CPU_CORE1 is IDLE bit0: set 1, FSM_STS will move to ON_S0 when CPU_CORE0 is IDLE
7:0	R/W	0	reg_oth_fsm_off_mask; bit7: reserved; bit6: set 1, FSM_STS will move to WAIT_PACTIVE1 when DSPB is WAIT_ON bit5: set 1, FSM_STS will move to WAIT_PACTIVE1 when DSPA is WAIT_ON bit4: set 1, FSM_STS will move to WAIT_PACTIVE1 when CPU_TOP is WAIT_ON bit3: set 1, FSM_STS will move to WAIT_PACTIVE1 when CPU_CORE3 is WAIT_ON bit2: set 1, FSM_STS will move to WAIT_PACTIVE1 when CPU_CORE2 is WAIT_ON bit1: set 1, FSM_STS will move to WAIT_PACTIVE1 when CPU_CORE1 is WAIT_ON bit0: set 1, FSM_STS will move to WAIT_PACTIVE1 when CPU_CORE0 is WAIT_ON

Table 7-29 PWRCTRL\_\*\*\*\_AUTO\_OFF\_CTRL2

Bits	R/W	Default	Description
31:16	R/W	0	reg_timeout_cnt_th_on_a
15:0	R/W	0	reg_timeout_cnt_th_off_a

Table 7-30 PWRCTRL\_\*\*\*\_AUTO\_OFF\_CTRL3

Bits	R/W	Default	Description
23:16	R/W	0	PSTATE value when REG_SET_PSATE = 0
15:8	R/W	0	PSTATE value when FSM_STS = SET_PSTATE1
7:0	R/W	0	PSTATE value when FSM_STS = SET_PSTATE2

Table 7-31 PWRCTRL\_\*\*\*\_AUTO\_OFF\_CTRL4

Bits	R/W	Default	Description
15:8	R/W	0	irq_clr, each bit can clear pwr_fsm internal irq status;
7:0	R/W	0	irq_en, each bit can enable internal irq to system;

Table 7-32 PWRCTRL\_\*\*\*\_TIMER\_TH\_01

Bits	R/W	Default	Description
31:16	R/W	0	timer count threshold 1; from OFF_S0 to OFF_S1;
15:0	R/W	0	timer count threshold 0; from OFF_WAIT to OFF_S0;

Table 7-33 PWRCTRL\_\*\*\*\_TIMER\_TH\_23

Bits	R/W	Default	Description
31:16	R/W	0	timer count threshold 3; from OFF_S2 to OFF_S3;
15:0	R/W	0	timer count threshold 2; from OFF_S1 to OFF_S2;

Table 7-34 PWRCTRL\_\*\*\*\_TIMER\_TH\_45

Bits	R/W	Default	Description
31:16	R/W	0	timer count threshold 5; from ON_S0/S0B to ON_S1A;
15:0	R/W	0	timer count threshold 4; from OFF_S3 to WAIT_ON;

Table 7-35 PWRCTRL\_\*\*\*\_TIMER\_TH\_67

Bits	R/W	Default	Description
31:16	R/W	0	timer count threshold 7; from ON_S2 to ON_S3;
15:0	R/W	0	timer count threshold 6; from ON_S1A/S1B/S1C/S1D to ON_S1B/S1C/S1D/S2;

**Table 7-36 PWRCTRL\_\*\*\*\_TIMER\_TH\_89**

Bits	R/W	Default	Description
31:16	R/W	0	timer count threshold 9; from SET_PSTATE2 to ON_PREQ2;
15:0	R/W	0	timer count threshold 8; from SET_PSTATE1 to ON_PREQ1;

**Table 7-37 PWRCTRL\_\*\*\*\_IRQ\_MASK0~7**

Bits	R/W	Default	Description
255:0	R/W	0	each bit: 1: jump to WAIT_OTH_FSM2 by this irq;

**Table 7-38 PWRCTRL\_\*\*\*\_MEMPD\_INIT\_SET**

Bits	R/W	Default	Description
31:0	R/W	0	MEMPD value when REG_FSM_EN = 0

**Table 7-39 PWRCTRL\_\*\*\*\_MEMPD\_OFF\_SET**

Bits	R/W	Default	Description
31:0	R/W	0	MEMPD value when FSM_STS = OFF_S2

**Table 7-40 PWRCTRL\_\*\*\*\_MEMPD\_ON\_A\_SET**

Bits	R/W	Default	Description
31:0	R/W	0	MEMPD value when FSM_STS = ON_S1A

**Table 7-41 PWRCTRL\_\*\*\*\_MEMPD\_ON\_B\_SET**

Bits	R/W	Default	Description
31:0	R/W	0	MEMPD value when FSM_STS = ON_S1B

**Table 7-42 PWRCTRL\_\*\*\*\_MEMPD\_ON\_C\_SET**

Bits	R/W	Default	Description
31:0	R/W	0	MEMPD value when FSM_STS = ON_S1C

**Table 7-43 PWRCTRL\_\*\*\*\_MEMPD\_ON\_D\_SET**

Bits	R/W	Default	Description
31:0	R/W	0	MEMPD value when FSM_STS = ON_S1D

**Table 7-44 PWRCTRL\_\*\*\*\_MEMPD\_STS**

Bits	R/W	Default	Description
31:0	R	0	final MEM_PD output

**Table 7-45 PWRCTRL\_\*\*\*\_FSM\_STS0**

Bits	R/W	Default	Description
31:24	R	0	FSM_PWR internal irq sts; bit7: timeout_cnt_a > reg_timeout_cnt_th_off_a when FSM_STS = OFF_PREQ1; bit6: timeout_cnt_a > reg_timeout_cnt_th_on_a when FSM_STS = ON_PREQ1; bit5: timeout_cnt_b > reg_timeout_cnt_th_off_b when FSM_STS = OFF_PREQ1B; bit4: FSM_STS change to ON_START_HS; bit4: FSM_STS change to ON_S1A; bit3: FSM_STS change to OFF_WAIT; bit2: FSM_STS change to WAIT_ON; bit0: FSM_STS change to IDLE;
23:17	R	0	timeout_cnt_b add by FSM sts change: from ON_PREQ1 to OFF_PREQ1B; clear at FSM sts is OFF_PREQ1A and SET_PSTATE1
16:12	R	0	current FSM sts: 0:IDLE 1:WAIT_OTH_FSM1 2:WAIT_PACTIVE1 3:START_HS1 4:SET_PSTATE1 5:ON_PREQ1 6:OFF_PREQ1B 7:OFF_PREQ1A 8:OFF_WAIT 9:OFF_S0 10:OFF_S1 11:OFF_S2 12:OFF_S3 16:WAIT_ON 17:WAIT_OTH_FSM2 18:ON_S0 19:ON_S0B 20:ON_S1A 21:ON_S1B 22:ON_S1C 23:ON_S1D 24:ON_S2 25:ON_S3 26:WAIT_PACTIVE2 27:START_HS2 28:SET_PACTIVE2 29:ON_PREQ2 30:OFF_PREQ2B 31:OFF_PREQ2A
11	R	0	final PREQ output
10	R	0	final PWR_OFF output
9	R	0	final FOC_RST output
8	R	0	final ISO_EN output
7:0	R	0	final PSTATE output

**Table 7-46 PWRCTRL\_\*\*\*\_FSM\_STS1**

Bits	R/W	Default	Description
31:16	R	0	timeout_cnt_a add by clock when FSM sts is ON_PREQ1/ON_PREQ2; clear if FSM sts is other sts;
15:0	R	0	r_timer_cnt add by timer_tick; clear if FSM sts changed;

**Table 7-47 PWRCTRL\_\*\*\*\_FSM\_STS2**

Bits	R/W	Default	Description
29:25	R	0	history5
24:20	R	0	history4
19:15	R	0	history3
14:10	R	0	history2
9:5	R	0	history1
4:0	R	0	history0

**Table 7-48 PWRCTRL\_\*\*\*\_FSM\_START\_OFF**

Bits	R/W	Default	Description
31:0	W	0	write any value will force FSM start from IDLE to WAIT_OTH_FSM1

**Table 7-49 PWRCTRL\_\*\*\*\_FSM\_START\_ON**

Bits	R/W	Default	Description
31:0	W	0	write any value will force FSM start from WAIT_ON to WAIT_OTH_FSM2

**Table 7-50 PWRCTRL\_\*\*\*\_FSM\_JUMP**

Bits	R/W	Default	Description
31:0	W	0	write any value will force FSM jump to reg_fsm_jump_sts

## 7.3 System Booting

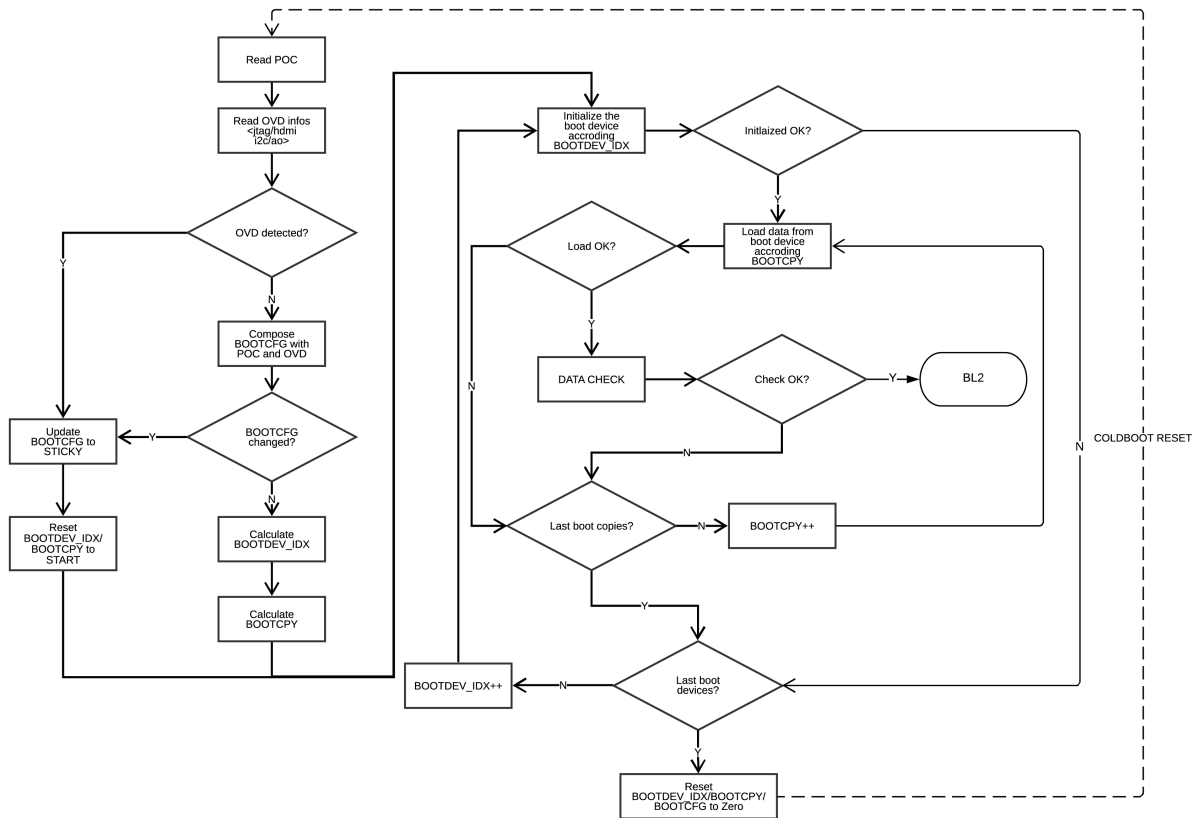
### 7.3.1 Overview

This part describes the power-on mode configuration of the SoC, which include two portions: Cortex-M3 for security control and A53 + A73 for others.

### 7.3.2 Power-on Flow Chart

The following figure illustrates the SoC's power on sequence.

Figure 7-8 Power-on Flow Chart



## 7.4 CPU

### 7.4.1 Overview

#### Cortex-A53

The Cortex™-A53 MP subsystem of the chip is a high-performance, low-power, ARM macro cell with an L1 cache subsystem and an L2 cache subsystem that provide full virtual memory capabilities. The Cortex-A53 processor implements the ARMv8 architecture and runs 32-bit ARM instructions and 64 bit ARMv8 instructions. The developers can follow the ARM official reference documents for programming details.

The Cortex-A53 processor features are:

- Pipeline in-order with dynamic branch prediction
- ARM, Thumb and ThumbEE instruction set support
- TrustZone security extensions
- Harvard level 1 memory system with a Memory Management Unit (MMU)
- 128-bit AXI master interface
- ARM CoreSight debug architecture
- Trace support through an Embedded Trace Macro cell (ETMv4) interface
- Intelligent Energy Manager (IEM) support with
  - Asynchronous AXI wrappers

- Two voltage domains
- Media Processing Engine (MPE) with NEON technology
- Supports FPU

## Cortex-A73

The quad core Cortex™-A73 processor is paired with A73 processor in a big.Little configuration, with each core has L1 instruction and data cache, together with a single shared L2 unified cache with A73. The developers can follow the ARM official reference documents for programming details.

The Cortex-A73 processor features are:

- Armv8-A Architecture
- 1-4x Symmetrical Multiprocessing (SMP) within a single processor cluster, and multiple coherent SMP processor clusters through AMBA 4 ACE technology
- AArch32 for full backward compatibility with Armv7
- AArch64 for 64-bit support and new architectural features
- TrustZone security technology
- NEON advanced SIMD
- DSP & SIMD extensions
- VFPv4 floating point
- trace support through CoreSight SoC-400 interface

## 7.4.2 Register Description

### Register Address

For below registers, the base address is 0xfe00e000.

Each register final address = BASE + address \* 4.

The following lists describe the mapping between each register and its address.

- CPUCTRL\_SYS\_A73\_RESET\_CNTL 0xfe00e100
- CPUCTRL\_SYS\_A73\_CLK\_CTRL0 0xfe00e104
- CPUCTRL\_SYS\_A73\_CLK\_CTRL1 0xfe00e108
- CPUCTRL\_SYS\_A73\_CLK\_CTRL2 0xfe00e10c
- CPUCTRL\_SYS\_CPU\_RESET\_CNTL 0xfe00e140
- CPUCTRL\_SYS\_CPU\_CLK\_CTRL0 0xfe00e144
- CPUCTRL\_SYS\_CPU\_CLK\_CTRL1 0xfe00e148
- CPUCTRL\_SYS\_CPU\_CLK\_RESULT 0xfe00e160
- CPUCTRL\_ROM\_DISABLE 0xfe00e180
- CPUCTRL\_SYS\_CPU\_PERIPHBASE 0xfe00e200
- CPUCTRL\_SYS\_CPU\_TARGETID 0xfe00e204
- CPUCTRL\_SYS\_CPU\_POR\_CFG0 0xfe00e240
- CPUCTRL\_SYS\_CPU\_POR\_CFG1 0xfe00e244
- CPUCTRL\_SYS\_CPU\_CFG0 0xfe00e248
- CPUCTRL\_SYS\_CPU\_CFG1 0xfe00e24c
- CPUCTRL\_SYS\_CPU\_CFG2 0xfe00e250
- CPUCTRL\_SYS\_CPU\_CFG3 0xfe00e254
- CPUCTRL\_SYS\_CPU\_CFG4 0xfe00e258



- CPUCTRL\_SYS\_CPU\_CFG5 0xfe00e25c
- CPUCTRL\_SYS\_CPU\_CFG6 0xfe00e260
- CPUCTRL\_SYS\_CPU\_CFG7 0xfe00e264
- CPUCTRL\_SYS\_CPU\_CFG8 0xfe00e268
- CPUCTRL\_SYS\_CPU\_CFG9 0xfe00e26c
- CPUCTRL\_SYS\_CPU\_STATUS0 0xfe00e280
- CPUCTRL\_SYS\_CPU\_STATUS1 0xfe00e284
- CPUCTRL\_SYS\_CPU\_STATUS2 0xfe00e288
- CPUCTRL\_SYS\_CPU\_STATUS3 0xfe00e28c
- CPUCTRL\_SYS\_CPU\_STATUS4 0xfe00e290
- CPUCTRL\_SYS\_CPU\_STATUS5 0xfe00e294
- CPUCTRL\_SYS\_A73\_PERIPBASE 0xfe00e2c0
- CPUCTRL\_SYS\_A73\_POR\_CFG0 0xfe00e300
- CPUCTRL\_SYS\_A73\_POR\_CFG1 0xfe00e304
- CPUCTRL\_SYS\_A73\_CFG0 0xfe00e308
- CPUCTRL\_SYS\_A73\_CFG1 0xfe00e30c
- CPUCTRL\_SYS\_A73\_CFG2 0xfe00e310
- CPUCTRL\_SYS\_A73\_CFG3 0xfe00e314
- CPUCTRL\_SYS\_A73\_CFG4 0xfe00e318
- CPUCTRL\_SYS\_A73\_CFG5 0xfe00e31c
- CPUCTRL\_SYS\_A73\_CFG6 0xfe00e320
- CPUCTRL\_SYS\_A73\_CFG7 0xfe00e324
- CPUCTRL\_SYS\_A73\_CFG8 0xfe00e328
- CPUCTRL\_SYS\_A73\_CFG9 0xfe00e32c
- CPUCTRL\_SYS\_A73\_STATUS0 0xfe00e340
- CPUCTRL\_SYS\_A73\_STATUS1 0xfe00e344
- CPUCTRL\_SYS\_A73\_STATUS2 0xfe00e348
- CPUCTRL\_SYS\_A73\_STATUS3 0xfe00e34c
- CPUCTRL\_SYS\_A73\_STATUS4 0xfe00e350
- CPUCTRL\_SYS\_A73\_STATUS5 0xfe00e354

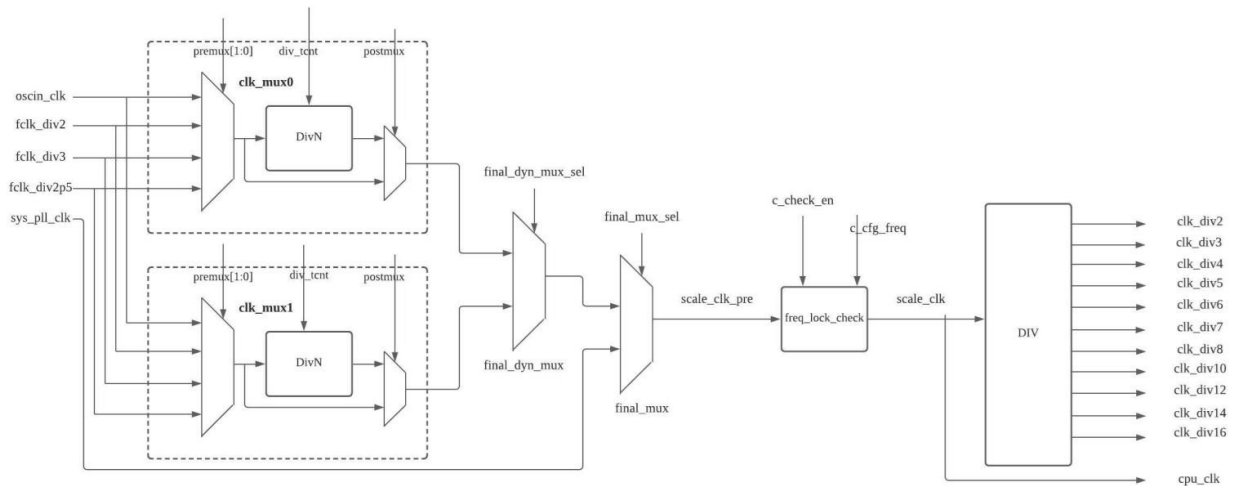
## A53 CPU CTRL

**Table 7-51 CPUCTRL\_SYS\_CPU\_RESET\_CNTL**

Bit(s)	R/W	Default	Description
31:10	R	0x0	Reserved
9	R/W	0x0	nPRESETDBG soft reset source. 0x0: does not provide soft reset to nPRESETDBG 0x1: provide soft reset to nPRESETDBG
8	R/W	0x0	nL2RESET soft reset source 0x0: does not provide soft reset to nL2RESET 0x1: provide soft reset to nL2RESET

Bit(s)	R/W	Default	Description
7:4	R/W	0x0	nCORERESET[3:0] soft reset source 0x0: does not provide soft reset to nCORERESET[3:0] 0x1: provide soft reset to nCORERESET[3:0]
3:0	R/W	0x0	nCPUPORESET[3:0] soft reset source 0x0: does not provide soft reset to nCPUPORESET[3:0] 0x1: provide soft reset to nCPUPORESET[3:0]

**Figure 7-9 A53 Clock Source Selection and Divide Control**



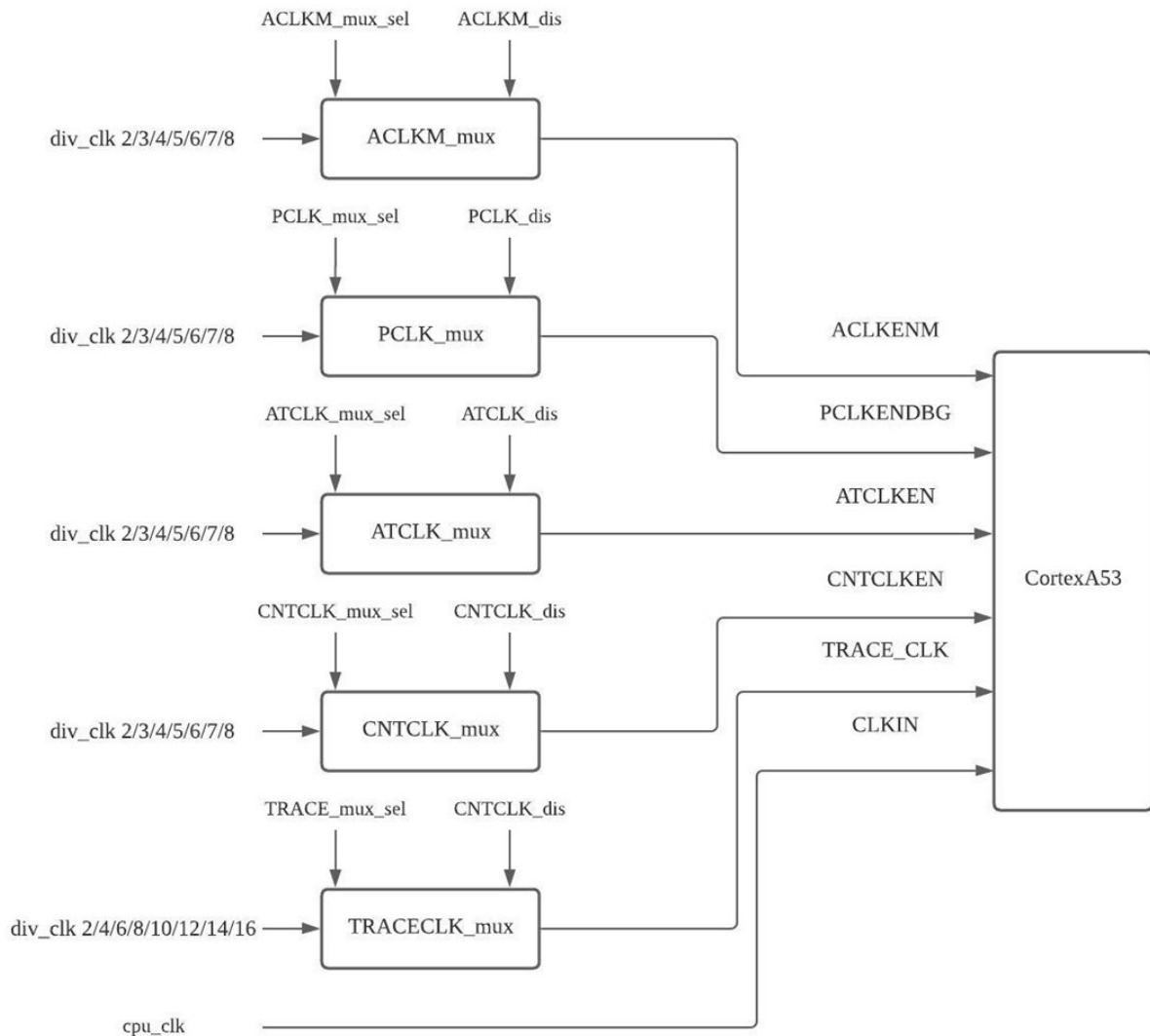
**Table 7-52 CPUCTRL\_SYS\_CPU\_CLK\_CTRL0**

This register controls A53 clock source selection and divide. Refer to the figure above for more details.

Bit(s)	R/W	Default	Description
31:28	R	0x0	Crt sys_cpu_clk_div status {final_mux_sel, final_dyn_mux, busy_cnt, busy}
27	R	0x0	Reserved
26	R/W	0x0	Cortex A53 sys_pll_div: crt_sys_cpu_clk_div clock muxing fsm dyn_enable signal
25:20	R/W	0x0	Clk_mux1: div_tcmt
18	R/W	0x0	Clk_mux1: premux
17:16	R/W	0x0	Clk_mux1: premux
15	R/W	0x0	Cortex A53 sys_pll_div: final mux: clock selection signal mode control. 0x0: use fsm dynamic mux signal to do clock selection. 0x1: directly use register bit[11] to do clock selection.
14	R/W	0x0	Cortex A53 sys_pll_div: Dynamic mux: clock selection signal mode control. 0x0: use fsm dynamic mux signal to do clock selection. 0x1: directly use register bit[10] to do clock selection.
13	R/W	0x0	Cortex A53 sys_pll_div: clk_mux0 clk_mux1 selection signal mode control. 0x0: use fsm dynamic mux signal to do clock selection. 0x1: directly use register bit to do clock selection.

Bit(s)	R/W	Default	Description
			This bit affects the clock selection mode of clk_mux0 (pre_mux and post_mux) and clk_mux1(pre_mux and post_mux)
12	R/W	0x0	Cortex A53 sys_pll_div: crt_sys_cpu_clk_div clock muxing fsm force_update_t signal
11	R/W	0x0	Cortex A53 sys_pll_div: final mux selection signal: final_mux_sel
10	R/W	0x0	Cortex A53 sys_pll_div: dynamic mux clock selection signal: final_dyn_mux_sel
9:4	R/W	0x0	Clk_mux0: div_tcnt
3	R/W	0x0	Rev
2	R/W	0x0	Clk_mux0: postmux
1:0	R/W	0x0	Clk_mux0: premux

Figure 7-10 A53 Clocks Selection



**Table 7-53 CPUCTRL\_SYS\_CPU\_CLK\_CTRL1**

This register controls A53 clock selection. Refer to the figure above for more details.

Bit(s)	R/W	Default	Description
31:25	R/W	0x0	Reserved
24	R/W	0x0	Sys_pll_div16_en
23	R/W	0x0	A53 TPIU TRACECLK_dis: Set to 1 to manually disable the A53 TPIU trace_clk when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glithces. This is a "just in case" bit
22:20	R/W	0x5	A53 TPIU TRACECLK_mux_sel: 0x0: cpu clock divided by 2 0x1: cpu clock divided by 4 0x2: cpu clock divided by 6 0x3: cpu clock divided by 8 0x4: cpu clock divided by 10 0x5: cpu clock divided by 12 0x6: cpu clock divided by 14 0x7: cpu clock divided by 16
19	R/W	0	Timestamp CNTCLKEN_dis: Set to 1 to manually disable the Timestamp CNTCLK clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glithces. This is a "just in case" bit
18	R/W	0	ACLKM_dis: Set to 1 to manually disable the AXI clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glithces. This is a "just in case" bit
17	R/W	0	ATCLK_dis Set to 1 to manually disable the ATB clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glithces. This is a "just in case" bit
16	R/W	0	PCLK_dis: Set to 1 to manually disable the APB clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glithces. This is a "just in case" bit
15	R/W	0	Timestamp CNTCLK enable: 0x0: disable 0x1: enable
14:12	R/W	0x0	Timestamp CNTCLK_mux_sel: 0x0: cpu clock divided by 2 0x1: cpu clock divided by 3 0x2: cpu clock divided by 4 0x3: cpu clock divided by 5 0x4: cpu clock divided by 6 0x5: cpu clock divided by 7 0x6: cpu clock divided by 8 0x7: no clock
11:9	R/W	0x1	ACLKM_mux_sel: 0x0: cpu clock divided by 2 0x1: cpu clock divided by 3 0x2: cpu clock divided by 4 0x3: cpu clock divided by 5 0x4: cpu clock divided by 6 0x5: cpu clock divided by 7 0x6: cpu clock divided by 8 0x7: no clock
8:6	R/W	0x2	ATCLK_mux_sel: 0x0: cpu clock divided by 2 0x1: cpu clock divided by 3 0x2: cpu clock divided by 4 0x3: cpu clock divided by 5 0x4: cpu clock divided by 6 0x5: cpu clock divided by 7

Bit(s)	R/W	Default	Description
			0x6: cpu clock divided by 8 0x7: no clock
5:3	R/W	0x4	PCLK_mux_sel: 0x0: cpu clock divided by 2 0x1: cpu clock divided by 3 0x2: cpu clock divided by 4 0x3: cpu clock divided by 5 0x4: cpu clock divided by 6 0x5: cpu clock divided by 7 0x6: cpu clock divided by 8 0x7: no clock
2	R/W	0x0	Soft reset to the clock divide logic. 0x0: no soft reset 0x1: soft reset clock divide logic
1	R/W	0x0	Sys_cpu_clk_div16_en 0x0: disable 0x1: enable
0	R/W	0x0	DEBUG APB PCLK enable 0x0: disable 0x1: enable

Table 7-54 CPUCTRL\_SYS\_CPU\_CLK\_RESULT

Bit(s)	R/W	Default	Description
31:2	R	0x0	Reserved
1	R	0	Cortex A73 cpu clk result
0	R	0	Cortex A53 cpu clk result

Table 7-55 CPUCTRL\_ROM\_DISABLE

Bit(s)	R/W	Default	Description
15-8	R/W	0xA5	can write only when value is 0xA5 if write to any value except 0xA5, will disable ROM MMU access.
7-0	R/W	0xA5	can write only when value is 0xA5 if write to any value except 0xA5, will disable ROM MAIN access.

Table 7-56 CPUCTRL\_SYS\_CPU\_POR\_CFG0

Bit(s)	R/W	Default	Description
31-29	R	0x0	Reserved
28	R/W	0x0	MBISTREQ
27:24	R	0x0	Reserved
23:20	R/W	0x0	CRYPTODISABLE[3:0] Disable the Cryptography Extensions. This pin is sampled only during reset of the processor.
19:16	R/W	0x0	CP15SDISABLE[3:0] Disable write access to some secure CP15 registers.
15:12	R/W	0x0	VINITHI[3:0]

Bit(s)	R/W	Default	Description
			Location of the exception vectors at reset. It sets the initial value of the V bit in the CP15 SCTLR register: 0 Exception vectors start at address 0x00000000. 1 Exception vectors start at address 0xFFFF0000. This pin is sampled only during reset of the processor.
11:9	R	0x0	Reserved
8	R/W	0x0	DBGL1RSTDISABLE Disable L1 data cache automatic invalidate on reset functionality: 0 Enable automatic invalidation of L1 data cache on reset. 1 Disable automatic invalidation of L1 data cache on reset. This pin is sampled only during reset of the processor.
7	R/W	0x0	BROADCASTINNER Enable broadcasting of Inner Shareable transactions: 0 Inner Shareable transactions are not broadcast externally. 1 Inner Shareable transactions are broadcast externally. If BROADCASTINNER is tied HIGH, you must also tie BROADCASTOUTER HIGH. This pin is sampled only during reset of the Cortex-A53 processor.
6	R/W	0x0	BROADCASTOUTER Enable broadcasting of outer shareable transactions: 0 Outer Shareable transactions are not broadcast externally. 1 Outer Shareable transactions are broadcast externally. This pin is sampled only during reset of the Cortex-A53 processor.
5	R/W	0x0	BROADCASTCACHEMAINT Enable broadcasting of cache maintenance operations to downstream caches: 0 Cache maintenance operations are not broadcast to downstream caches. 1 Cache maintenance operations are broadcast to downstream caches. This pin is sampled only during reset of the Cortex-A53 processor.
4	R/W	0x0	L2RSTDISABLE Disable automatic L2 cache invalidate at reset: 0 Hardware resets L2 cache. 1 Hardware does not reset L2 cache.
3:0	R/W	0x0	CFGEND[3:0] Endianness configuration at reset. It sets the initial value of the EE bits in the CP15 SCTLR_EL3 and SCTR_S registers: 0 EE bit is LOW. 1 EE bit is HIGH. This pin is sampled only during reset of the processor.

Table 7-57 CPUCTRL\_SYS\_CPU\_POR\_CFG1

Bit(s)	R/W	Default	Description
31:11	R	0x1FFF-FF	Reserved
10	R/W	0x1	ACINACTM Snoop interface is inactive and not participating in coherency: 0 Snoop interface is active. 1 Snoop interface is inactive.
9	R/W	0x0	SYSBARDISABLE Disable broadcasting of barriers onto the system bus: 0 Barriers are broadcast onto the system bus. This requires an AMBA4 ACE, or AMBA5 CHI, interconnect. 1 Barriers are not broadcast onto the system bus. This is compatible with an AXI3 interconnect and most AMBA4 interconnects.a This pin is sampled only during reset of the Cortex-A53 processor.

Bit(s)	R/W	Default	Description
8	R/W	0x1	GICCDISABLE Globally disables the GIC CPU interface logic and routes the “External” signals directly to the processor: 0 Enable the GIC CPU interface logic. 1 Disable the GIC CPU interface logic and route the legacy nIRQ, nFIQ, nVIRQ, and nVFIQ signals directly to the processor. Drive this signal HIGH when using a legacy interrupt controller such as GIC-400 which does not support GICv3 or GICv4.
7:4	R/W	0xF	AA64nAA32[3:0] Register width state: 0 AArch32. 1 AArch64. This pin is sampled only during reset of the processor.
3:0	R/W	0xF	CFGTE[3:0] Enable T32 exceptions. It sets the initial value of the TE bit in the CP15 SCTLR register: 0 TE bit is LOW. 1 TE bit is HIGH. This pin is sampled only during reset of the processor.

Table 7-58 CPUCTRL\_SYS\_CPU\_CFG0

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	CLUSTERIDAFF2[7:0] Value read in the Cluster ID Affinity Level 2 field, MPIDR bits[23:16], of the CP15 MPIDR register. These pins are sampled only during reset of the processor.
23:16	R/W	0x0	CLUSTERIDAFF1[7:0] Value read in the Cluster ID Affinity Level 1 field, MPIDR bits[15:8], of the CP15 MPIDR register. These pins are sampled only during reset of the processor.
15:4	R	0x0	Reserved
3:0	R/W	0x0	DBGEN[3:0] Invasive debug enable: 0 Not enabled. 1 Enabled.

Table 7-59 CPUCTRL\_SYS\_CPU\_CFG1

Bit(s)	R/W	Default	Description
31:21	R/W	0x0	Reserved
20	R/W	0x0	DBGROMADDRV Debug ROM base address valid. If the debug ROM address cannot be determined, tie this signal LOW. This pin is sampled only during reset of the processor.
19:0	R/W	0x0	DBGROMADDR[31:12] Debug ROM base address. Specifies bits[39:12] of the ROM table physical address. If the address cannot be determined, tie this signal LOW. This pin is sampled only during reset of the processor.

**Table 7-60 CPUCTRL\_SYS\_CPU\_CFG2**

Bit(s)	R/W	Default	Description
31:2	R/W	0x3FFF_8000	RVBARADDR0[31:2] Core 0 Reset Vector Base Address for executing in 64-bit state. These pins are sampled only during reset of the processor.
1:0	R	0x0	Reserved

**Table 7-61 CPUCTRL\_SYS\_CPU\_CFG3**

Bit(s)	R/W	Default	Description
31:2	R/W	0x3FFF_8000	RVBARADDR1[31:2] Core 1 Reset Vector Base Address for executing in 64-bit state. These pins are sampled only during reset of the processor.
1:0	R	0x0	Reserved

**Table 7-62 CPUCTRL\_SYS\_CPU\_CFG4**

Bit(s)	R/W	Default	Description
31:2	R/W	0x3FFF_8000	RVBARADDR2[31:2] Core 2 Reset Vector Base Address for executing in 64-bit state. These pins are sampled only during reset of the processor.
1:0	R	0x0	Reserved

**Table 7-63 CPUCTRL\_SYS\_CPU\_CFG5**

Bit(s)	R/W	Default	Description
31:2	R/W	0x3FFF_8000	RVBARADDR3[31:2] Core 3 Reset Vector Base Address for executing in 64-bit state. These pins are sampled only during reset of the processor.
1:0	R	0x0	Reserved

**Table 7-64 CPUCTRL\_SYS\_CPU\_CFG6**

Bit(s)	R/W	Default	Description
31:16	R/W	0x0	ICDTDATA[15:0] AXI4 Stream Protocol signal. Distributor to GIC CPU Interface messages. TDATA is the primary payload that is used to provide the data that is passing across the interface.
15	R/W	0x0	ICCTREADY AXI4 Stream Protocol signal. Distributor to GIC CPU Interface messages. TREADY indicates that the slave can accept a transfer in the current cycle.
14:13	R/W	0x0	ICDTDEST[1:0] AXI4 Stream Protocol signal. Distributor to GIC CPU Interface messages. TDEST provides routing information for the data stream.
12	R/W	0x0	ICDTLAST AXI4 Stream Protocol signal. Distributor to GIC CPU Interface messages. TLAST



Bit(s)	R/W	Default	Description
			indicates the boundary of a packet.
11	R/W	0x0	ICDTVALID AXI4 Stream Protocol signal. Distributor to GIC CPU Interface messages. TVALID indicates that the master is driving a valid transfer.
10	R/W	0x0	L2FLUSHREQ L2 hardware flush request.
9	R/W	0x0	EVENTI Event input for processor wake-up from WFE state.
8	R/W	0x0	CLREXMONREQ Clearing of the external global exclusive monitor request. When this signal is asserted, it acts as a WFE wake-up event to all the cores in the MPCore device.
7:4	R	0x0	Reserved
3:0	R/W	0x0	EDBGRQ[3:0] External debug request: 0 No external debug request. 1 External debug request. The processor treats the EDBGRQ input as level-sensitive. The EDBGRQ input must be asserted until the processor asserts DBGACK.

Table 7-65 CPUCTRL\_SYS\_CPU\_CFG7

Bit(s)	R/W	Default	Description
31:26	R	0	Reserved
25	R/W	0x1	CNTVALUEB selection 0x0:select CNTVLAUEB from timestamp unit 0x1:select CNTVLAUEB from internal counter
24	R/W	0x1	L2QREQn Indicates that the power controller is ready to enter or exit retention for the L2 data RAMs
23:20	R/W	0xf	NEONQREQn[3:0] Indicates that the power controller is ready to enter or exit retention for the referenced Advanced SIMD and Floating-point block
19:16	R/W	0xF	DBGPWRDUP[3:0] Core powered up 0 Core is powered down. 1 Core is powered up.
15:12	R/W	0xf	CPUQREQn[3:0] Indicates that the power controller is ready to enter or exit retention for the referenced core
11:8	R/W	0xf	nSEI[3:0] System Error Interrupt request. Active-LOW, edge sensitive: 0 Activate SEI request. 1 Do not activate SEI request. Asserting the nSEI input causes one of the following to occur: • Asynchronous Data Abort, if taken to AArch32. The DFSR.FS field is set to indicate an Asynchronous External Abort. • SError interrupt, if taken to AArch64.

Bit(s)	R/W	Default	Description
7:4	R/W	0xf	nVSEI[3:0] Virtual FIQ request. Active-LOW, level sensitive, asynchronous FIQ interrupt request: 0 Activate FIQ interrupt. 1 Do not activate FIQ interrupt. The processor treats the nVFIQ input as level-sensitive. The nVFIQ input must be asserted until the processor acknowledges the interrupt. If the GIC is enabled by tying the GICCDISABLE input pin LOW, the nVFIQ input pin must be tied off to HIGH. If the GIC is disabled by tying the GICCDISABLE input pin HIGH, the nVFIQ input pin can be driven by an external GIC in the SoC.
3:0	R/W	0xf	nREI[3:0] RAM Error Interrupt request. Active-LOW, edge sensitive: 0 Activate REI request. Reports an asynchronous RAM error in the system. 1 Do not activate REI request. Asserting the nREI input causes one of the following to occur: • Asynchronous Data Abort, if taken to AArch32. The DFSR.FS field is set to indicate an Asynchronous parity error on memory access. • SError interrupt, if taken to AArch64.

Table 7-66 CPUCTRL\_SYS\_CPU\_CFG8

Bit(s)	R/W	Default	Description
31	R/W	0x0	Coresight DAP component debug power up ack signal source selection: 0x0: use DAP debug power up request as system ack signal to DAP. 0x1: use bit[11] or bit[12] as the debug power up ack signal.
30	R/W	0x0	Coresight DAP component system power up ack signal source selection: 0x0: use DAP system power up request as system ack signal to DAP. 0x1: use bit[11] or bit[12] as the system power up ack signal.
29:13	R	0x0	Reserved
12	R/W	0x0	Connects to Coresight DAP component Power Controller Interface System power on request ack for csypwrup
11	R/W	0x0	Connects Coresight DAP component Power Controller Interface Debug power on request ack for cdbgpwrup
10	R/W	0x0	Coresight DAP component JTAG ntrst. Normally JTAG trst comes from chip reset not from internally registers.
9:8	R	0x0	Reserved
7:4	R/W	0x0	Coresight DAP component instanceid If multiple serial wire targets with the same target ID might share a connection, instanceid must be driven differently for each target. Drive a unique value on this input if the system contains multiple SWJ-DPs. In other cases, you must tie this input LOW.
3	R/W	0x0	Coresight DAP component debug reset cdbgrst
2	R/W	0x0	Coresight TPIU component tpctl tie-off signal. If tracectl is not connected then this input must be tied LOW. ARM recommends that you do not connect tracectl, and you tie tpctl LOW, unless you must support a legacy Trace Port Adaptor that cannot support formatter operation in continuous mode.
1:0	R	0x0	Reserved

Table 7-67 CPUCTRL\_SYS\_CPU\_CFG9

Bit(s)	R/W	Default	Description
31:28	R/W	0xf	Coretex A53 CTI Interrupt ack CTIIRQACK[3:0]
27:14	R/W	0x3fff	Reserved
13:10	R/W	0x0	Coretex A53 channel interface H/S bypass CIHSBYPASS[3:0]
9	R/W	0x1	Coretex A53 channel interface bypass CISBYPASS
8	R/W	0x1	Coresight Debug APB clock enable
7	R/W	0x1	Coresight DAP APB-AP tie off signals. If the APB-AP is connected to a debug bus, you must tie device HIGH
6	R/W	0x1	Drives Coresight CTI component niden Non-Invasive debug enable
5	R/W	0x1	Drives Coresight CTI component dbgen Invasive debug enable
4:0	R/W	0x0f	Coresight TPIU tie off signals. Indicates how many pins of tracedata[31:0] are connected to pins of the soc. 0xf means 16bits.

Table 7-68 CPUCTRL\_SYS\_CPU\_STATUS0

Bit(s)	R/W	Default	Description
31:27	R	0x0	Reserved
26	R	0x0	L2FLUSHDONE L2 hardware flush complete.
25	R	0x0	STANDBYWFIL2 Indicates whether the L2 memory system is in WFI low-power state. This signal is active when the following conditions are met: <ul style="list-style-type: none"> <li>• All cores are in WFI low-power state, held in reset, or nL2RESET is asserted LOW.</li> <li>• In an ACE configuration, ACINACTM is asserted HIGH.</li> <li>• In a CHI configuration, SINACT is asserted HIGH.</li> <li>• If ACP has been configured, AINACTS is asserted HIGH.</li> <li>• L2 memory system is idle.</li> </ul>
24:21	R	0x0	STANDBYWFE[3:0] Indicates whether a core is in WFE low-power state: 0x0 Core not in WFE low-power state. 0x1 Core in WFE low-power state
20:17	R	0x0	STANDBYWFI[3:0] Indicates whether a core is in WFI low-power state: 0x0 Core not in WFI low-power state. 0x1 Core in WFI low-power state. This is the reset condition
16	R	0x0	EXTERRIRQ Error indicator for AXI or CHI transactions with a write response error condition. 0x1 interrupt. 0x0 No interrupt.
15:12	R	0x0	CTIIRQ[3:0] CTI interrupt (active-HIGH). 0x1 interrupt. 0x0 No interrupt.
11:8	R	0x0	VCPUMNTIRQ

Bit(s)	R/W	Default	Description
			Virtual CPU interface maintenance interrupt PPI output. 0x1 interrupt. 0x0 No interrupt.
7:4	R	0x0	COMMIRQ[3:0] Communications channel receive or transmit interrupt request. 0x1 interrupt. 0x0 No interrupt.
3:0	R	0x0	PMUIRQ[3:0] PMU interrupt request. 0x1 interrupt. 0x0 No interrupt.

Table 7-69 CPUCTRL\_SYS\_CPU\_STATUS1

Bit(s)	R/W	Default	Description
31:28	R	0x0	DBGNOPWRDWN[3:0] Core no powerdown request 0 Do not request that the core stays powered up. 1 Request that the core stays powered up.
27:24	R	0x0	DBGACK[3:0] Debug acknowledge: 0 External debug request not acknowledged. 1 External debug request acknowledged.
23:20	R	0xf	COMMTX[3:0] Communication transmit channel. Transmit portion of Data Transfer Register empty flag: 0 Full. 1 Empty.
19:16	R	0x0	COMMRX[3:0] Communications channel receive. Receive portion of Data Transfer Register full flag: 0 Empty. 1 Full.
15	R	0x0	Coresight DAP system power up request
14	R	0x0	Coresight DAP debug reset request
13	R	0x0	Coresight timestamp tsforcesync Re-synchronize event indicator. This signal indicates an event that requires all the receivers to re-synchronize to the new tsvalue
12	R	0x0	Coresight DAP component debug power up request. cpwrupreq_cdbgpwrup
11:8	R	0x0	DBGPWRUPREQ[3:0] Core power up request: 0 Do not request that the core is powered up. 1 Request that the core is powered up.
7:4	R	0x0	DBGIRSTREQ[3:0] Warm reset request.
3:0	R	0x0	WARMIRSTREQ[3:0] Processor warm reset request 0 Do not apply warm reset. 1 Apply warm reset.

**Table 7-70 CPUCTRL\_SYS\_CPU\_STATUS2**

Bit(s)	R/W	Default	Description
31:16	R/W		
15:8	R/W		RDMEMATTR[7:0] Read request memory attributes.
7:0	R/W		WRMEMATTR[7:0] Write request memory attributes.

**Table 7-71 CPUCTRL\_SYS\_CPU\_STATUS3**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Reserved

**Table 7-72 CPUCTRL\_SYS\_CPU\_STATUS4**

Bit(s)	R/W	Default	Description
31:7	R	0x0	Reserved
6	R	0x0	CLREXMONACK Clearing of the external global exclusive monitor acknowledge.
5	R	0x0	EVENT0 Event output. Active when a SEV instruction is executed
4	R	0x0	Reserved
3:0	R	0x0	SMPEN[3:0] Indicates whether a core is taking part in coherency.

**Table 7-73 CPUCTRL\_SYS\_CPU\_STATUS5**

Bit(s)	R/W	Default	Description
31:27	R		Reserved
26:23	R		CPUQACTIVE[3:0] Indicates whether the referenced core is active.
22:19	R		CPUQDENY[3:0] Indicates that the referenced core denies the power controller retention request.
18:15	R		CPUQACCEPTn[3:0] Indicates that the referenced core accepts the power controller retention request.
14:11	R		NEONQACTIVE[3:0] Indicates whether the referenced Advanced SIMD and Floating-point block is active.
10:7	R		NEONQDENY[3:0] Indicates that the referenced Advanced SIMD and Floating-point block denies the power controller retention request.
6:3	R		NEONQACCEPTn[3:0] Indicates that the referenced Advanced SIMD and Floating-point block accepts the power controller retention request.
2	R		L2QACTIVE Indicates that the L2 data RAMs accept the power controller retention request.

Bit(s)	R/W	Default	Description
1	R		L2QDENY Indicates that the L2 data RAMs deny the power controller retention request.
0	R		L2QACCEPTn Indicates that the L2 data RAMs accept the power controller retention request.

## A73 CPU CTRL

**Table 7-74 CPUCTRL\_SYS\_A73\_RESET\_CNTL0**

Bit(s)	R/W	Default	Description
31:10	R	0x0	Reserved
9	R/W	0x0	nPRESETDBG soft reset source. 0x0: does not provide soft reset to nPRESETDBG 0x1: provide soft reset to nPRESETDBG
8	R/W	0x0	nL2RESET soft reset source 0x0: does not provide soft reset to nL2RESET 0x1: provide soft reset to nL2RESET
7:4	R/W	0x0	nCORERESET[3:0] soft reset source 0x0: does not provide soft reset to nCORERESET[3:0] 0x1: provide soft reset to nCORERESET[3:0]
3:0	R/W	0x0	nCPUPORESET[3:0] soft reset source 0x0: does not provide soft reset to nCPUPORESET[3:0] 0x1: provide soft reset to nCPUPORESET[3:0]

**Table 7-75 CPUCTRL\_SYS\_A73\_CLK\_CTRL0**

Bit(s)	R/W	Default	Description
31:28	R	0x0	Crt sys_cpu_clk_div status {final_mux_sel, final_dyn_mux, busy_cnt, busy}
27	R	0x0	Reserved
26	R/W	0x0	Cortex 73 sys_pll_div: crt_sys_cpu_clk_div clock muxing fsm dyn_enable signal
25:20	R/W	0x0	Clk_mux1: div_tcnt
18	R/W	0x0	Clk_mux1: premux
17:16	R/W	0x0	Clk_mux1: premux
15	R/W	0x0	Cortex A73 sys_pll_div: final mux: clock selection signal mode control. 0x0: use fsm dynamic mux signal to do clock selection. 0x1: directly use register bit[11] to do clock selection.
14	R/W	0x0	Cortex A73 sys_pll_div: Dynamic mux: clock selection signal mode control. 0x0: use fsm dynamic mux signal to do clock selection. 0x1: directly use register bit[10] to do clock selection.
13	R/W	0x0	Cortex A73 sys_pll_div: clk_mux0 clk_mux1 selection signal mode control. 0x0: use fsm dynamic mux signal to do clock selection. 0x1: directly use register bit to do clock selection. This bit affects the clock selection mode of clk_mux0 (pre_mux and post_mux) and clk_mux1(pre_mux and post_mux)
12	R/W	0x0	Cortex A73 sys_pll_div: crt_sys_cpu_clk_div clock muxing fsm force_update_t signal
11	R/W	0x0	Cortex A73 sys_pll_div: final mux selection signal: final_mux_sel

Bit(s)	R/W	Default	Description
10	R/W	0x0	Cortex A73 sys_pll_div: dynamic mux clock selection signal: final_dyn_mux_sel
9:4	R/W	0x0	Clk_mux0: div_tcnt
3	R/W	0x0	Rev
2	R/W	0x0	Clk_mux0: postmux
1:0	R/W	0x0	Clk_mux0: premux

**Table 7-76 CPUCTRL\_SYS\_A73\_CLK\_CTRL1**

This register controls A53 clock selection. Refer to [Figure 7-10](#) for more details.

Bit(s)	R/W	Default	Description
31:25	R/W	0x0	Reserved
24	R/W	0x0	Sys_pll_div16_en
23	R/W	0x0	A73 TPIU TRACECLK_dis: Set to 1 to manually disable the A53 TPIU trace_clk when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glitches. This is a “just in case” bit
22:20	R/W	0x5	A73 TPIU TRACECLK_mux_sel: 0x0: cpu clock divided by 2 0x1: cpu clock divided by 4 0x2: cpu clock divided by 6 0x3: cpu clock divided by 8 0x4: cpu clock divided by 10 0x5: cpu clock divided by 12 0x6: cpu clock divided by 14 0x7: cpu clock divided by 16
19	R/W	0	Timestamp CNTCLKEN_dis: Set to 1 to manually disable the Timestamp CNTCLK clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glitches. This is a “just in case” bit
18	R/W	0	ACLKM_dis: Set to 1 to manually disable the AXI clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glitches. This is a “just in case” bit
17	R/W	0	ATCLK_dis Set to 1 to manually disable the ATB clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glitches. This is a “just in case” bit
16	R/W	0	PCLK_dis: Set to 1 to manually disable the APB clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glitches. This is a “just in case” bit
15	R/W	0	Timestamp CNTCLK enable: 0x0: disable 0x1: enable
14:12	R/W	0x0	Timestamp CNTCLK_mux_sel: 0x0: cpu clock divided by 2 0x1: cpu clock divided by 3 0x2: cpu clock divided by 4 0x3: cpu clock divided by 5 0x4: cpu clock divided by 6 0x5: cpu clock divided by 7 0x6: cpu clock divided by 8 0x7: no clock
11:9	R/W	0x1	ACLKM_mux_sel: 0x0: cpu clock divided by 2 0x1: cpu clock divided by 3

Bit(s)	R/W	Default	Description
			0x2: cpu clock divided by 4 0x3: cpu clock divided by 5 0x4: cpu clock divided by 6 0x5: cpu clock divided by 7 0x6: cpu clock divided by 8 0x7: no clock
8:6	R/W	0x2	ATCLK_mux_sel: 0x0: cpu clock divided by 2 0x1: cpu clock divided by 3 0x2: cpu clock divided by 4 0x3: cpu clock divided by 5 0x4: cpu clock divided by 6 0x5: cpu clock divided by 7 0x6: cpu clock divided by 8 0x7: no clock
5:3	R/W	0x4	PCLK_mux_sel: 0x0: cpu clock divided by 2 0x1: cpu clock divided by 3 0x2: cpu clock divided by 4 0x3: cpu clock divided by 5 0x4: cpu clock divided by 6 0x5: cpu clock divided by 7 0x6: cpu clock divided by 8 0x7: no clock
2	R/W	0x0	Soft reset to the clock divide logic. 0x0: no soft reset 0x1: soft reset clock divide logic
1	R/W	0x0	Sys_cpu_clk_div16_en 0x0: disable 0x1: enable
0	R/W	0x0	DEBUG APB PCLK enable 0x0: disable 0x1: enable

Table 7-77 CPUCTRL\_SYS\_A73\_CLK\_CTRL2

Bit(s)	R/W	Default	Description
31:16	R/W	0x0	Reserved
15	R/W	0x0	Core3 CORECLKEN[3] force to 1. Which means divide by 1 of cpu clock
14	R/W	0x0	Core3 CORECLKEN[3] clock disable Set to 1 to manually disable clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glitches. This is a "just in case" bit
13:12	R/W	0x0	Core3 CORECLKEN[3] clk_mux: 0x0: cpu clock divided by 1 0x1: cpu clock divided by 2 0x2: cpu clock divided by 3 0x3: cpu clock divided by 4 0x4,0x5,0x6,0x7: no clock
11	R/W	0x0	Core2 CORECLKEN[2] force to 1. Which means divide by 1 of cpu clock
10	R/W	0x0	Core2 CORECLKEN[2] clock disable Set to 1 to manually disable clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glitches. This is a "just in case" bit
9:8	R/W	0x0	Core2 CORECLKEN[2] clk_mux: 0x0: cpu clock divided by 1



Bit(s)	R/W	Default	Description
			0x1: cpu clock divided by 2 0x2: cpu clock divided by 3 0x3: cpu clock divided by 4 0x4,0x5,0x6,0x7: no clock
7	R/W	0x0	Core1 CORECLKEN[1] force to 1. Which means divide by 1 of cpu clock
6	R/W	0x0	Core1 CORECLKEN[1] clock disable Set to 1 to manually disable clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glitches. This is a "just in case" bit
5:4	R/W	0x0	Core1 CORECLKEN[1] clk_mux: 0x0: cpu clock divided by 1 0x1: cpu clock divided by 2 0x2: cpu clock divided by 3 0x3: cpu clock divided by 4 0x4,0x5,0x6,0x7: no clock
3	R/W	0x0	Core0 CORECLKEN[0] force to 1. Which means divide by 1 of cpu clock
2	R/W	0x0	Core0 CORECLKEN[0] clock disable Set to 1 to manually disable clock when changing the mux selection. Typically this bit is set to 0 since the clock muxes can switch without glitches. This is a "just in case" bit
1:0	R/W	0x0	Core0 CORECLKEN[0] clk_mux: 0x0: cpu clock divided by 1 0x1: cpu clock divided by 2 0x2: cpu clock divided by 3 0x3: cpu clock divided by 4 0x4,0x5,0x6,0x7: no clock

Table 7-78 CPUCTRL\_SYS\_A73\_POR\_CFG0

Bit(s)	R/W	Default	Description
31-29	R	0x0	Reserved
28	R/W	0x0	MBISTREQ
27:20	R	0x0	Reserved
19:16	R/W	0x0	CRYPTODISABLE[3:0] Disable the Cryptography Extensions. This pin is sampled only during reset of the processor.
15:12	R/W	0x0	VINITHI[3:0] Location of the exception vectors at reset. It sets the initial value of the V bit in the CP15 SCTLR register: 0 Exception vectors start at address 0x00000000. 1 Exception vectors start at address 0xFFFF0000. This pin is sampled only during reset of the processor.
11:10	R	0x0	Reserved
9	R/W	0x0	Enable broadcasting of cache maintenance operations to downstream caches: 0x0 Cache maintenance operations are not broadcast to downstream caches. 0x1 Cache maintenance operations are broadcast to downstream caches. This pin is sampled only during reset of the Cortex-A73 processor. In a bigLITTLE system with Cortex-A73, this pin must be tied to 0 because Cortex-A73 does not support this feature.
8	R/W	0x0	DBGL1RSTDISABLE Disable L1 data cache automatic invalidate on reset functionality: 0 Enable automatic invalidation of L1 data cache on reset. 1 Disable automatic invalidation of L1 data cache on reset. This pin is sampled only during reset of the processor.

Bit(s)	R/W	Default	Description
7	R/W	0x0	<b>BROADCASTINNER</b> Enable broadcasting of Inner Shareable transactions: 0 Inner Shareable transactions are not broadcast externally. 1 Inner Shareable transactions are broadcast externally. If BROADCASTINNER is tied HIGH, you must also tie BROADCASTOUTER HIGH. This pin is sampled only during reset of the Cortex-A53 processor.
6	R/W	0x0	<b>BROADCASTOUTER</b> Enable broadcasting of outer shareable transactions: 0 Outer Shareable transactions are not broadcast externally. 1 Outer Shareable transactions are broadcast externally. This pin is sampled only during reset of the Cortex-A53 processor.
5	R/W	0x0	<b>BROADCASTCACHEMAINT</b> Enable broadcasting of cache maintenance operations to downstream caches: 0 Cache maintenance operations are not broadcast to downstream caches. 1 Cache maintenance operations are broadcast to downstream caches. This pin is sampled only during reset of the Cortex-A53 processor.
4	R/W	0x0	<b>L2RSTDISABLE</b> Disable automatic L2 cache invalidate at reset: 0 Hardware resets L2 cache. 1 Hardware does not reset L2 cache.
3:0	R/W	0x0	<b>CFGEND[3:0]</b> Endianness configuration at reset. It sets the initial value of the EE bits in the CP15 SCTLR_EL3 and SCTR_S registers: 0 EE bit is LOW. 1 EE bit is HIGH. This pin is sampled only during reset of the processor.

Table 7-79 CPUCTRL\_SYS\_A73\_POR\_CFG1

Bit(s)	R/W	Default	Description
31:11	R	0x1FFF-FF	Reserved
10	R/W	0x1	<b>ACINACTM</b> Snoop interface is inactive and not participating in coherency: 0 Snoop interface is active. 1 Snoop interface is inactive.
9	R/W	0x0	<b>SYSBARDISABLE</b> Disable broadcasting of barriers onto the system bus: 0 Barriers are broadcast onto the system bus. This requires an AMBA4 ACE, or AMBA5 CHI, interconnect. 1 Barriers are not broadcast onto the system bus. This is compatible with an AXI3 interconnect and most AMBA4 interconnects. This pin is sampled only during reset of the Cortex-A53 processor.
8	R/W	0x1	<b>GICCDISABLE</b> Globally disables the GIC CPU interface logic and routes the “External” signals directly to the processor: 0 Enable the GIC CPU interface logic. 1 Disable the GIC CPU interface logic and route the legacy nIRQ, nFIQ, nVIRQ, and nVFIQ signals directly to the processor. Drive this signal HIGH when using a legacy interrupt controller such as GIC-400 which does not support GICv3 or GICv4.

Bit(s)	R/W	Default	Description
7:4	R/W	0xF	AA64nAA32[3:0] Register width state: 0 AArch32. 1 AArch64. This pin is sampled only during reset of the processor.
3:0	R/W	0xF	CFGTE[3:0] Enable T32 exceptions. It sets the initial value of the TE bit in the CP15 SCTLR register: 0 TE bit is LOW. 1 TE bit is HIGH. This pin is sampled only during reset of the processor.

Table 7-80 CPUCTRL\_SYS\_A73\_CFG0

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	CLUSTERIDAFF2[7:0] Value read in the Cluster ID Affinity Level 2 field, MPIDR bits[23:16], of the CP15 MPIDR register. These pins are sampled only during reset of the processor.
23:16	R/W	0x0	CLUSTERIDAFF1[7:0] Value read in the Cluster ID Affinity Level 1 field, MPIDR bits[15:8], of the CP15 MPIDR register. These pins are sampled only during reset of the processor.
15:4	R	0x0	Reserved
3:0	R/W	0x0	DBGEN[3:0] Invasive debug enable: 0 Not enabled. 1 Enabled.

Table 7-81 CPUCTRL\_SYS\_A73\_CFG1

Bit(s)	R/W	Default	Description
31:21	R/W	0x0	Reserved
20	R/W	0x0	DBGROMADDRV Debug ROM base address valid. If the debug ROM address cannot be determined, tie this signal LOW. This pin is sampled only during reset of the processor.
19:0	R/W	0x0	DBGROMADDR[31:12] Debug ROM base address. Specifies bits[39:12] of the ROM table physical address. If the address cannot be determined, tie this signal LOW. This pin is sampled only during reset of the processor.

Table 7-82 CPUCTRL\_SYS\_A73\_CFG2

Bit(s)	R/W	Default	Description
31:2	R/W	0x3FFF_8000	RVBARADDR0[31:2] Core 0 Reset Vector Base Address for executing in 64-bit state. These pins are sampled only during reset of the processor.
1:0	R	0x0	Reserved

**Table 7-83 CPUCTRL\_SYS\_A73\_CFG3**

Bit(s)	R/W	Default	Description
31:2	R/W	0x3FFF_8000	RVBARADDR1[31:2] Core 1 Reset Vector Base Address for executing in 64-bit state. These pins are sampled only during reset of the processor.
1:0	R	0x0	Reserved

**Table 7-84 CPUCTRL\_SYS\_A73\_CFG4**

Bit(s)	R/W	Default	Description
31:2	R/W	0x3FFF_8000	RVBARADDR2[31:2] Core 2 Reset Vector Base Address for executing in 64-bit state. These pins are sampled only during reset of the processor.
1:0	R	0x0	Reserved

**Table 7-85 CPUCTRL\_SYS\_A73\_CFG5**

Bit(s)	R/W	Default	Description
31:2	R/W	0x3FFF_8000	RVBARADDR3[31:2] Core 3 Reset Vector Base Address for executing in 64-bit state. These pins are sampled only during reset of the processor.
1:0	R	0x0	Reserved

**Table 7-86 CPUCTRL\_SYS\_A73\_CFG6**

Bit(s)	R/W	Default	Description
31:16	R/W	0x0	ICDTPDATA[15:0] AXI4 Stream Protocol signal. Distributor to GIC CPU Interface messages. TDATA is the primary payload that is used to provide the data that is passing across the interface.
15	R/W	0x0	ICCTREADY AXI4 Stream Protocol signal. Distributor to GIC CPU Interface messages. TREADY indicates that the slave can accept a transfer in the current cycle.
14:13	R/W	0x0	ICDTDEST[1:0] AXI4 Stream Protocol signal. Distributor to GIC CPU Interface messages. TDEST provides routing information for the data stream.
12	R/W	0x0	ICDTLAST AXI4 Stream Protocol signal. Distributor to GIC CPU Interface messages. TLAST indicates the boundary of a packet.
11	R/W	0x0	ICDTVALID AXI4 Stream Protocol signal. Distributor to GIC CPU Interface messages. TVALID indicates that the master is driving a valid transfer.
10	R/W	0x0	L2FLUSHREQ L2 hardware flush request.
9	R/W	0x0	EVENTI Event input for processor wake-up from WFE state.

Bit(s)	R/W	Default	Description
8	R/W	0x0	CLREXMONREQ Clearing of the external global exclusive monitor request. When this signal is asserted, it acts as a WFE wake-up event to all the cores in the MPCore device.
7:4	R	0x0	Reserved
3:0	R/W	0x0	EDBGRQ[3:0] External debug request: 0 No external debug request. 1 External debug request. The processor treats the EDBGRQ input as level-sensitive. The EDBGRQ input must be asserted until the processor asserts DBGACK.

Table 7-87 CPUCTRL\_SYS\_A73\_CFG7

Bit(s)	R/W	Default	Description
31:26	R	0	Reserved
25	R/W	0x1	CNTVALUEB selection 0x0:select CNTVLAUEB from timestamp unit 0x1:select CNTVLAUEB from internal counter
24	R/W	0x1	L2QREQn Indicates that the power controller is ready to enter or exit retention for the L2 data RAMs
23:20	R/W	0xf	NEONQREQn[3:0] Indicates that the power controller is ready to enter or exit retention for the referenced Advanced SIMD and Floating-point block
19:16	R/W	0xF	DBGPWRDUP[3:0] Core powered up 0 Core is powered down. 1 Core is powered up.
15:12	R/W	0xf	CPUQREQn[3:0] Indicates that the power controller is ready to enter or exit retention for the referenced core
11:8	R/W	0xf	nSEI[3:0] System Error Interrupt request. Active-LOW, edge sensitive: 0 Activate SEI request. 1 Do not activate SEI request. Asserting the nSEI input causes one of the following to occur: • Asynchronous Data Abort, if taken to AArch32. The DFSR.FS field is set to indicate an Asynchronous External Abort. • SError interrupt, if taken to AArch64.

Bit(s)	R/W	Default	Description
7:4	R/W	0xf	nVSEI[3:0] Virtual FIQ request. Active-LOW, level sensitive, asynchronous FIQ interrupt request: 0 Activate FIQ interrupt. 1 Do not activate FIQ interrupt. The processor treats the nVFIQ input as level-sensitive. The nVFIQ input must be asserted until the processor acknowledges the interrupt. If the GIC is enabled by tying the GICCDISABLE input pin LOW, the nVFIQ input pin must be tied off to HIGH. If the GIC is disabled by tying the GICCDISABLE input pin HIGH, the nVFIQ input pin can be driven by an external GIC in the SoC.
3:0	R/W	0xf	nREI[3:0] RAM Error Interrupt request. Active-LOW, edge sensitive: 0 Activate REI request. Reports an asynchronous RAM error in the system. 1 Do not activate REI request. Asserting the nREI input causes one of the following to occur: • Asynchronous Data Abort, if taken to AArch32. The DFSR.FS field is set to indicate an Asynchronous parity error on memory access. • SError interrupt, if taken to AArch64.

Table 7-88 CPUCTRL\_SYS\_A73\_CFG8

Bit(s)	R/W	Default	Description
31	R/W	0x0	Coresight DAP component debug power up ack signal source selection: 0x0: use DAP debug power up request as system ack signal to DAP. 0x1: use bit[11] or bit[12] as the debug power up ack signal.
30	R/W	0x0	Coresight DAP component system power up ack signal source selection: 0x0: use DAP system power up request as system ack signal to DAP. 0x1: use bit[11] or bit[12] as the system power up ack signal.
29:13	R	0x0	Reserved
12	R/W	0x0	Connects to Coresight DAP component Power Controller Interface System power on request ack for csypwrup
11	R/W	0x0	Connects Coresight DAP component Power Controller Interface Debug power on request ack for cdbgpwrup
10	R/W	0x0	Coresight DAP component JTAG ntrst. Normally JTAG trst comes from chip reset not from internally registers.
9:8	R	0x0	Reserved
7:4	R/W	0x0	Coresight DAP component instanceid If multiple serial wire targets with the same target ID might share a connection, instanceid must be driven differently for each target. Drive a unique value on this input if the system contains multiple SWJ-DPs. In other cases, you must tie this input LOW.
3	R/W	0x0	Coresight DAP component debug reset cdbgrst
2	R/W	0x0	Coresight TPIU component tpctl tie-off signal. If tracectl is not connected then this input must be tied LOW. ARM recommends that you do not connect tracectl, and you tie tpctl LOW, unless you must support a legacy Trace Port Adaptor that cannot support formatter operation in continuous mode.
1:0	R	0x0	Reserved

Table 7-89 CPUCTRL\_SYS\_A73\_CFG9

Bit(s)	R/W	Default	Description
31:28	R/W	0xf	Coretex A53 CTI Interrupt ack CTIIRQACK[3:0]
27:14	R/W	0x3fff	Reserved
13:10	R/W	0x0	Coretex A53 channel interface H/S bypass CIHSBYPASS[3:0]
9	R/W	0x1	Coretex A53 channel interface bypass CISBYPASS
8	R/W	0x1	Coresight Debug APB clock enable
7	R/W	0x1	Coresight DAP APB-AP tie off signals. If the APB-AP is connected to a debug bus, you must tie device HIGH
6	R/W	0x1	Drives Coresight CTI component niden Non-Invasive debug enable
5	R/W	0x1	Drives Coresight CTI component dbgen Invasive debug enable
4:0	R/W	0x0f	Coresight TPIU tie off signals. Indicates how many pins of tracedata[31:0] are connected to pins of the soc. 0xf means 16bits.

Table 7-90 CPUCTRL\_SYS\_A73\_STATUS0

Bit(s)	R/W	Default	Description
31:28	R	0x0	Reserved
27	R	0x0	AWAKEUPM
26	R	0x0	L2FLUSHDONE L2 hardware flush complete.
25	R	0x0	STANDBYWFIL2 Indicates whether the L2 memory system is in WFI low-power state. This signal is active when the following conditions are met: <ul style="list-style-type: none"> <li>• All cores are in WFI low-power state, held in reset, or nL2RESET is asserted LOW.</li> <li>• In an ACE configuration, ACINACTM is asserted HIGH.</li> <li>• In a CHI configuration, SINACT is asserted HIGH.</li> <li>• If ACP has been configured, AINACTS is asserted HIGH.</li> <li>• L2 memory system is idle.</li> </ul>
24:21	R	0x0	STANDBYWFE[3:0] Indicates whether a core is in WFE low-power state: 0x0 Core not in WFE low-power state. 0x1 Core in WFE low-power state
20:17	R	0x0	STANDBYWFI[3:0] Indicates whether a core is in WFI low-power state: 0x0 Core not in WFI low-power state. 0x1 Core in WFI low-power state. This is the reset condition
16	R	0x0	AXIERRIRQ Error indicator for AXI or CHI transactions with a write response error condition. 0x1 interrupt. 0x0 No interrupt.
15:12	R	0x0	CTIIRQ[3:0] CTI interrupt (active-HIGH). 0x1 interrupt. 0x0 No interrupt.

Bit(s)	R/W	Default	Description
11:8	R	0x0	VCPUMNTIRQ Virtual CPU interface maintenance interrupt PPI output. 0x1 interrupt. 0x0 No interrupt.
7:4	R	0x0	COMMIRQ[3:0] Communications channel receive or transmit interrupt request. 0x1 interrupt. 0x0 No interrupt.
3:0	R	0x0	PMUIRQ[3:0] PMU interrupt request. 0x1 interrupt. 0x0 No interrupt.

Table 7-91 CPUCTRL\_SYS\_A73\_STATUS1

Bit(s)	R/W	Default	Description
31:28	R	0x0	DBGNOPWRDWN[3:0] Core no powerdown request 0 Do not request that the core stays powered up. 1 Request that the core stays powered up.
27:24	R	0x0	DBGACK[3:0] Debug acknowledge: 0 External debug request not acknowledged. 1 External debug request acknowledged.
23:20	R	0xf	COMMTX[3:0] Communication transmit channel. Transmit portion of Data Transfer Register empty flag: 0 Full. 1 Empty.
19:16	R	0x0	COMMRX[3:0] Communications channel receive. Receive portion of Data Transfer Register full flag: 0 Empty. 1 Full.
15	R	0x0	Coresight DAP system power up request
14	R	0x0	Coresight DAP debug reset request
13	R	0x0	Coresight timestamp tsforcesync Re-synchronize event indicator. This signal indicates an event that requires all the receivers to re-synchronize to the new tsvalue
12	R	0x0	Coresight DAP component debug power up request. cpwrupreq_cdbgwrup
11:8	R	0x0	DBGPWRUPREQ[3:0] Core power up request: 0 Do not request that the core is powered up. 1 Request that the core is powered up.
7:4	R	0x0	DBGRSTREQ[3:0] Warm reset request.
3:0	R	0x0	WARMRSTREQ[3:0] Processor warm reset request 0 Do not apply warm reset. 1 Apply warm reset.



**Table 7-92 CPUCTRL\_SYS\_A73\_STATUS2**

Bit(s)	R/W	Default	Description
31:16	R/W		
15:8	R/W		RDMEMATTR[7:0] Read request memory attributes.
7:0	R/W		WRMEMATTR[7:0] Write request memory attributes.

**Table 7-93 CPUCTRL\_SYS\_A73\_STATUS3**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Reserved

**Table 7-94 CPUCTRL\_SYS\_A73\_STATUS4**

Bit(s)	R/W	Default	Description
31:7	R	0x0	Reserved
6	R	0x0	CLREXMONACK Clearing of the external global exclusive monitor acknowledge.
5	R	0x0	EVENT0 Event output. Active when a SEV instruction is executed
4	R	0x0	Reserved
3:0	R	0x0	SMPEN[3:0] Indicates whether a core is taking part in coherency.

**Table 7-95 CPUCTRL\_SYS\_A73\_STATUS5**

Bit(s)	R/W	Default	Description
31:27	R		Reserved
26:23	R		CPUQACTIVE[3:0] Indicates whether the referenced core is active.
22:19	R		CPUQDENY[3:0] Indicates that the referenced core denies the power controller retention request.
18:15	R		CPUQACCEPTn[3:0] Indicates that the referenced core accepts the power controller retention request.
14:11	R		NEONQACTIVE[3:0] Indicates whether the referenced Advanced SIMD and Floating-point block is active.
10:7	R		NEONQDENY[3:0] Indicates that the referenced Advanced SIMD and Floating-point block denies the power controller retention request.
6:3	R		NEONQACCEPTn[3:0] Indicates that the referenced Advanced SIMD and Floating-point block accepts the power controller retention request.

Bit(s)	R/W	Default	Description
2	R		L2QACTIVE Indicates that the L2 data RAMs accept the power controller retention request.
1	R		L2QDENY Indicates that the L2 data RAMs deny the power controller retention request.
0	R		L2QACCEPTn Indicates that the L2 data RAMs accept the power controller retention request.

## 7.5 GPU

The Mali-G52 MP8 GPU is a hardware accelerator for 2D and 3D graphics system which compatible with the following graphics standards: OpenGL ES 3.2 Vulkan 1.1 and OpenCL 2.0. The developers can follow the ARM and Khronos official reference documents for programming details.

## 7.6 Clock

### 7.6.1 Overview

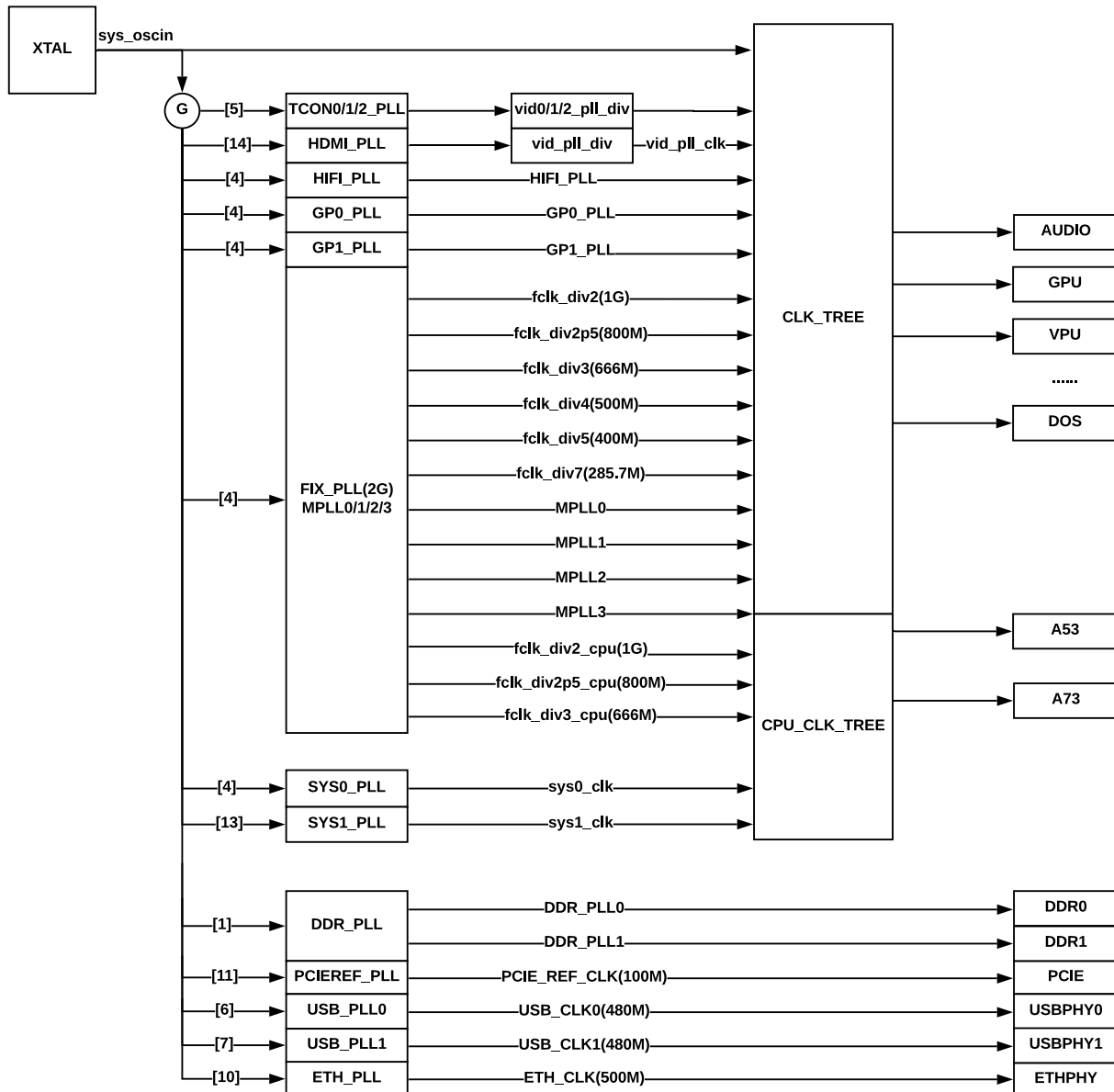
The clock and reset unit is an APB slave module that is designed for generating all of the internal and system clocks, resets of chip. The SoC uses an external 24MHz crystal; there are several internal PLLs which generate clock sources, as shown in the following table.

**Table 7-96 Clock Source**

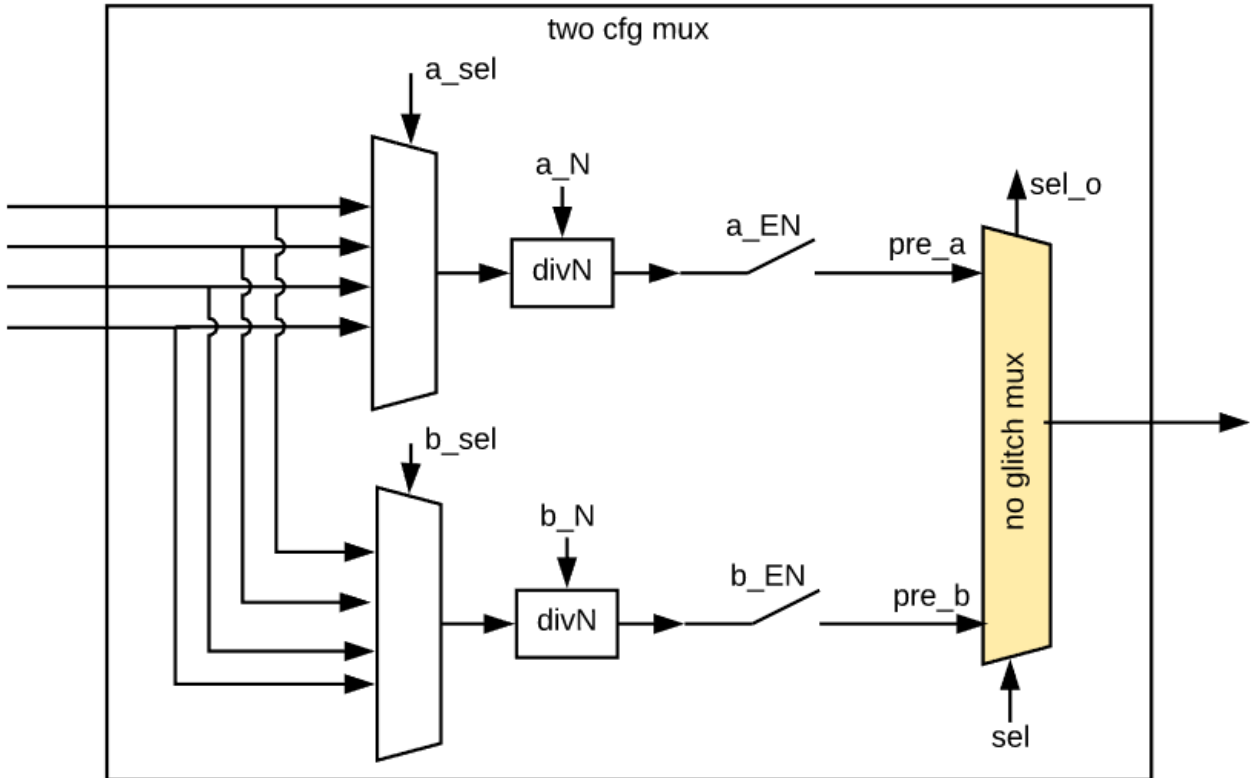
PLL Type	Frequency	Integer/Fractional	Spread Spectrum
SYS0_PLL	1.6G~3.2G	integer	no
SYS1_PLL	1.6G~3.2G	integer	no
GP0_PLL	3~6G	fractional	Yes
GP1_PLL	1.6G~3.2G	integer	no
HIFI_PLL	3~6G	fractional	Yes
HDMI_RXPLL		fractional	Yes
Fixed PLL		integer	Yes
	MPLL0	fractional	Yes
	MPLL1	fractional	Yes
	MPLL2	fractional	Yes
	MPLL3	fractional	Yes
HDMI_TXPLL	3~6G	fractional	Yes
HDMI_RXPLL	3~6G	INT	No
DDR_PLL	3~6G	fractional	Yes
AUD_DDS	200M	fractional	Yes
TCON_PLL1~3	3~6G	fractional	Yes
PCIE_REF	100M	INT	NO
MCLK_PLL_TOP	74.25M	INT	NO

PLL Type	Frequency	Integer/Fractional	Spread Spectrum
ETHPLL		fractional	No
USB2PLL		fractional	No

Figure 7-11 PLLs Connection



Two cfg mux is a special module for dynamic switch clock frequency. It can generate two different clock sources by two muxs and divN. Then selected by a "glitch free mux".



### 7.6.2 24MHz Crystal Oscillator Layout

Oscillator from XTAL pad will connect all PLLs and some analog modules:

**Table 7-97 Xtal & Module Connections**

0	xtal
1	cts_rtc_clk
2	sys_pll_div16
3	ddr_dppll_pt_clk
4	vid_pll0_clk
5	gp0_pll_clk
6	gp1_pll_clk
7	hifi_pll_clk
8	pcie_clk_inn
9	pcie1_clk_inp
10	a73_sys_pll_div16
11	mclk_pll_clk
12	cts_msr_clk

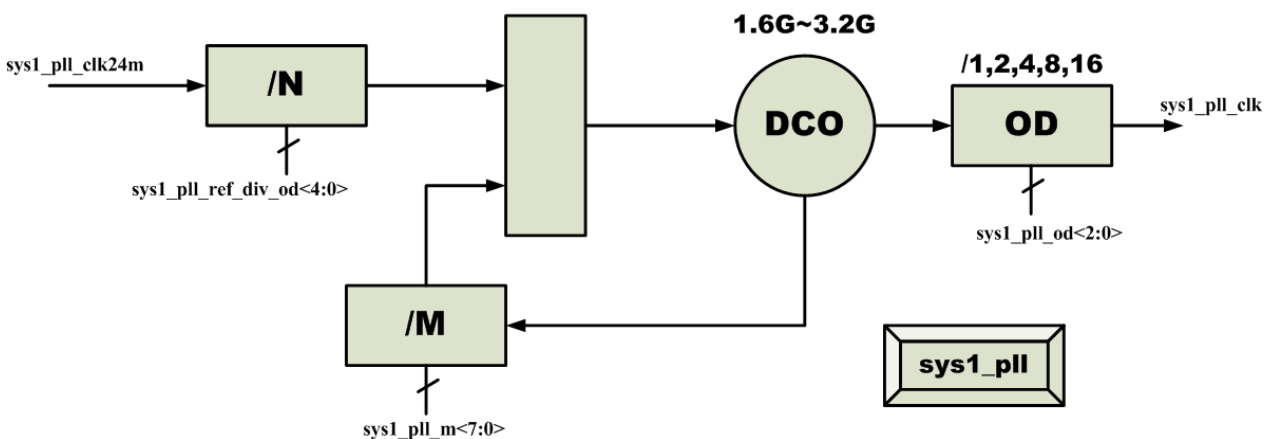
0	xtal
13	hdmirx_aud_pll_clk
14	hdmirx_aud_pll_clk
15	0
16	AU_DAC1_CLK_TO_GPIO
17	sys_cpu_clk_div16
18	a73_sys_cpu_clk_div16
19	fclk_div2
20	fclk_div2p5
21	fclk_div3
22	fclk_div4
23	fclk_div5
24	fclk_div7
25	mp0_clk_out
26	mp1_clk_out
27	mp2_clk_out
28	mp3_clk_out
29	vid_pll1_clk
30	earcrx_pll_ckout
31	vid_pll2_clk

### 7.6.3 Frequency Calculation

#### 7.6.3.1 SYS0 1 PLL / GP1 PLL

SYS0 1 PLL / GP1 PLL diagram is shown in the following figure.

Figure 7-12 SYS0 1 PLL / GP1 PLL



Sys1\_pll\_clk : min=100M, max=3.2G

3 pll share the same core.

Main Function Description:

DCO Cover Range: 1.6G~3.2G

DCO frequency is calculated with the following equation:

$$f_{DCO} = f_{REF} \cdot (M) / N$$

OD control table is as following.

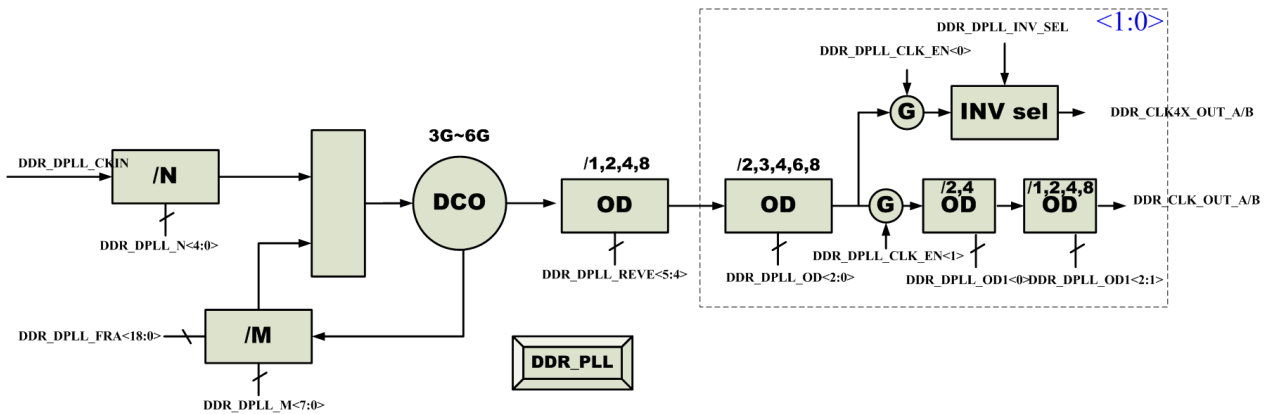
**Table 7-98 OD Control**

Block	Register	Function
OD	SYS1_PLL_OD<2:0>	000:/1
		001:/2
		010:/4
		011:/8
		100:/16

### 7.6.3.2 DDR\_PLL

DDR\_PLL diagram is shown in the following figure.

**Figure 7-13 DDR\_PLL**



DDR\_CLK4X\_OUT: max=2.132G. DDR\_CLK\_OUT:max=1066M

**Note**

DDR\_CLK\_OUT\_A/B for 2 DDR PHY with 0.25UI phase shift  
 For application: DDR3/LP DDR3/DDR4/ LP DDR4

OD control table is as following.

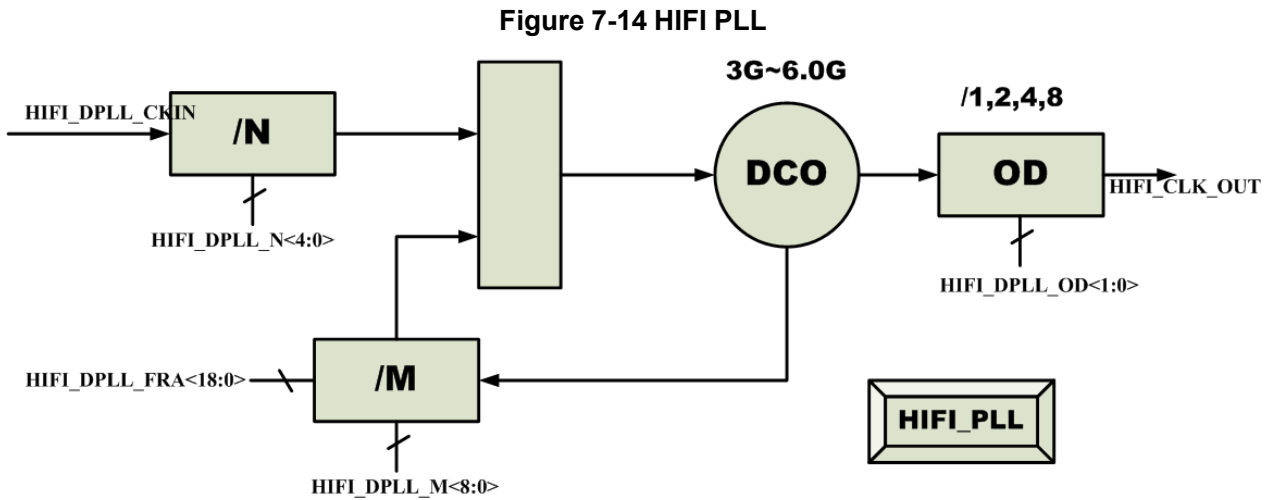
Table 7-99 OD Control

Block	Register	Function
OD	DDR_DPLL_REVE<5:4> /Cntrl5<5:4>	00: /1
		01: /2
		10: /4
		11: /8
OD	DDR_DPLL_OD<2:0> /Cntrl0<18:16>	000: /2
		001: /3
		010: /4
		011: /6
		100: /8
OD	DDR_DPLL_OD1<0>/Cntrl0<19>	0: /2
		1: /4
OD	DDR_DPLL_OD1<2:1>/Cntrl0<21:20>	00: /1
		01: /2
		10: /4
		11: /8

Clock Type	4G bps	2G bps	1G bps	500M bps
VCO_CLK	4G	4G	4G	4G
DFI_CLK	1G	500M	250M	125M
DDR_CLK	2G	1G	500M	250M
Phase shift for DDR_CLKA DDR_CLKB	0.25T	0.25T	0.25T	0.25T
DDR_DPLL_ REVE<5:4>	00	01	10	11

### 7.6.3.3 HIFI PLL

HIFI PLL diagram is shown in the following figure.



HIFI\_CLK\_OUT: min=375MHz max=6G

**Note**

OD add 1/8 option.

DCO frequency is calculated with the following equation:

$$f_{DCO} = f_{REF} \cdot (M + frac) / N$$

OD control table is as following.

**Table 7-100 HIFI PLL OD Control**

Block	Register	Function
OD	HIFI_DPLL_OD<1:0>	00:/1
		01:/2
		10:/4
		10:/8

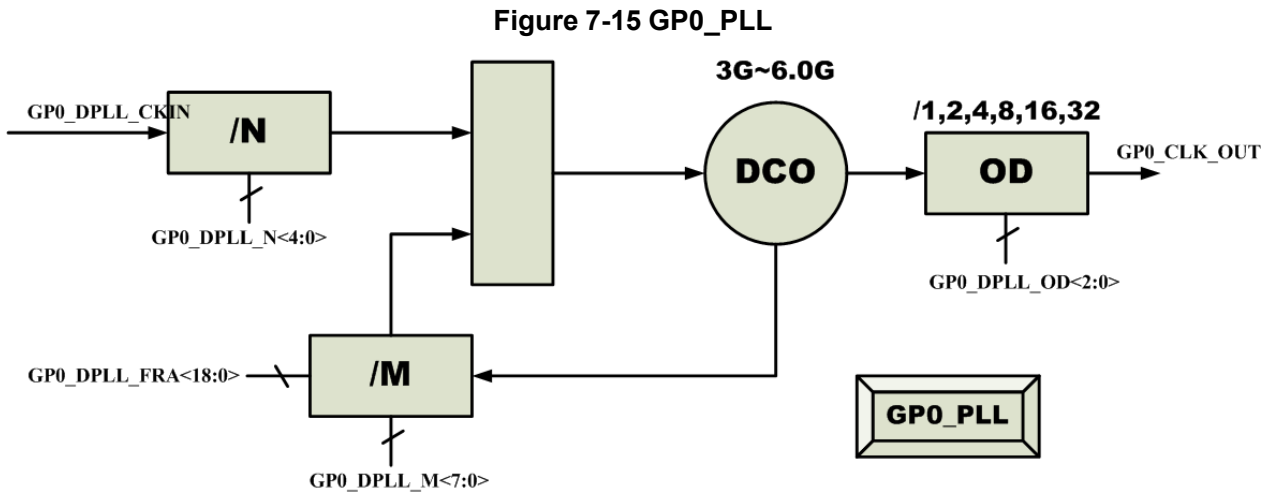
**Table 7-101 HIFI\_DPLL\_FRA<18:0>**

	Bit18	Bit17	Bit16	Bit15	..	Bit0
Weight	+/-	1	frac= value 16:0 / 100000 100000=17'h186a0			

**7.6.3.4 GP0\_PLL**

GP0\_PLL diagram is shown in the following figure.





GP0\_CLK\_OUT: min=375MHz max=6G

DCO frequency is calculated with the following equation:

$$f_{DCO} = f_{REF} \cdot (M + frac) / N$$

OD control table is as following.

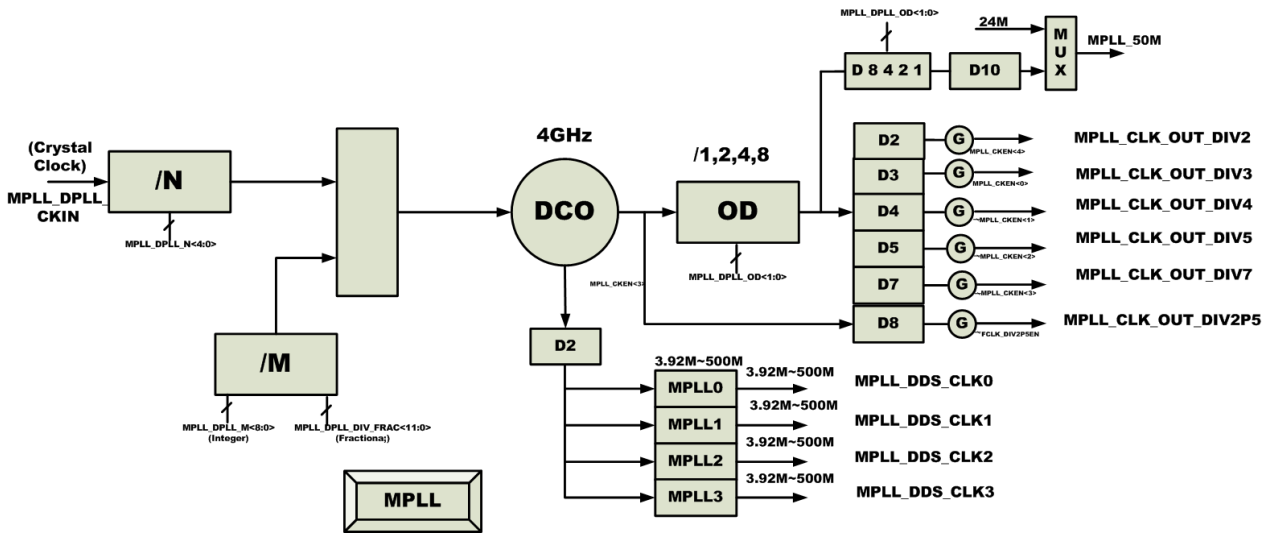
**Table 7-102 GP0\_PLL OD Control**

Block	Register	Function
OD	GP0_DPLL_OD<2:0>	000:/1
		001:/2
		010:/4
		011:/8
		100:/16
		101:/32

### 7.6.3.5 MPLL

MPLL diagram is shown in the following figure.

Figure 7-16 MPLL



Main Function Description

DCO Cover Range: 4GHz

For MPLL: in below equation, N>3. MPLL\_DDS\_CLK0~3 has SSG function

DCO frequency is calculated with the following equation:

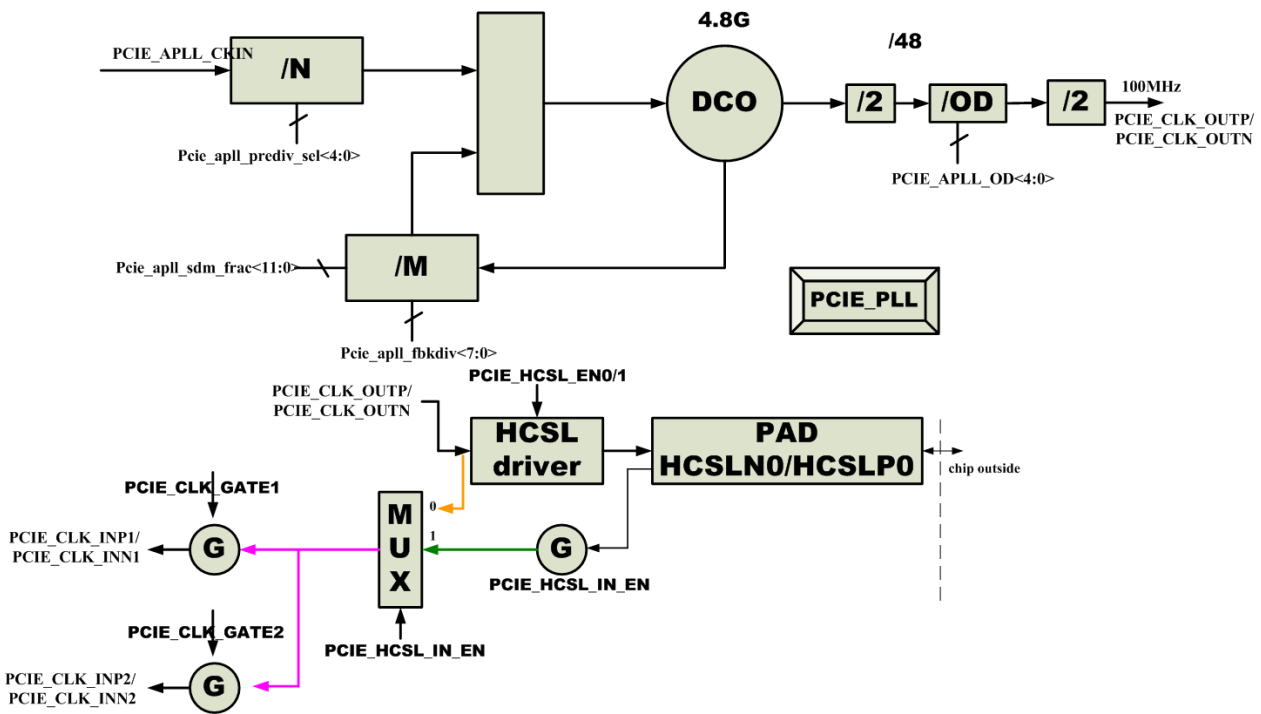
$$F_{out\_2G} = F_{ref} * M * OD\_FB / N$$

$$MPLL\_CLK\_OUT2 = f(N2\_integer, SDM\_IN) = \left( \frac{2Ghz}{(N2\_integer + \frac{SDM\_IN}{16384})} \right)$$

7.6.3.6 PCIE PLL

PCIE PLL diagram is shown in the following figure.

Figure 7-17 PCIE PLL



DCO frequency is calculated with the following equation:

$$f_{DCO} = f_{REF} \cdot (M + frac) / N$$

The PCIE PLL fractional value weight table is shown below.

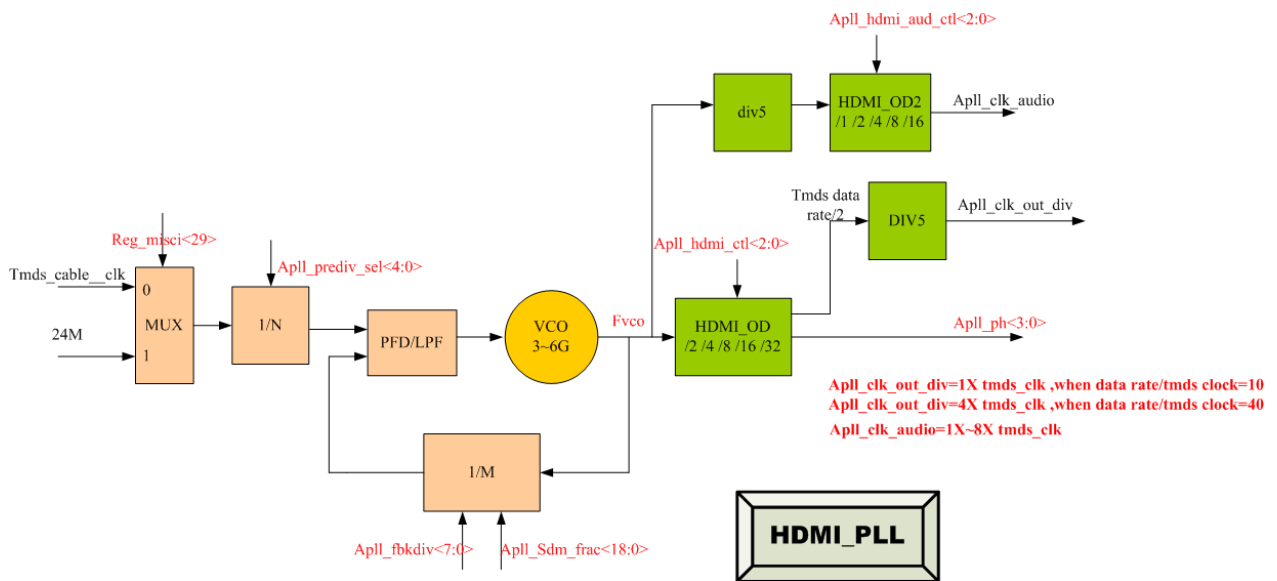
Table 7-103 PCIE PLL OD Control Table

Block	Register	Function
OD	PCIE_APLL_OD<4:0>	OD<4:0>=5'h09 : 1/9

### 7.6.3.7 HDMI RX PLL

HDMI RX PLL diagram is shown in the following figure.

Figure 7-18 HDMI RX PLL



Frac<18:0>:

18 +-	17 1	16 0.5	2^17=131072
-------	------	--------	-------------

OD control table is as following.

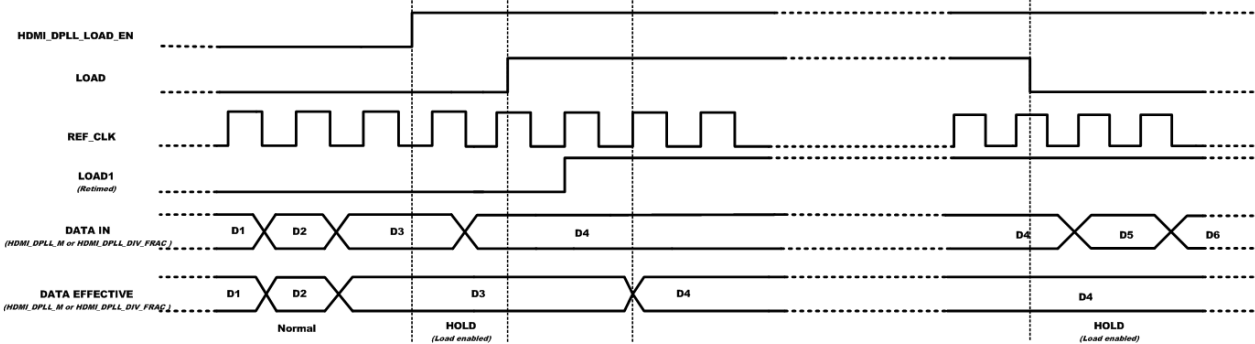
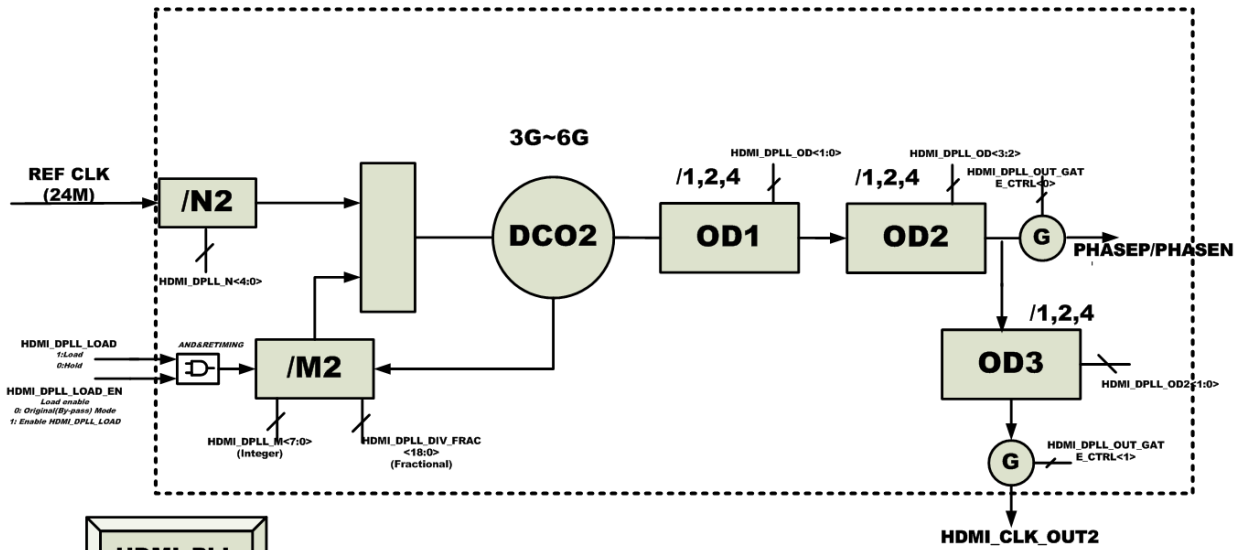
Table 7-104 HDMI PLL OD Control

Block	Register	Function
HDMI_OD	Apll_hdmi_ctl<2:0>	001: /2
		010:/4
		011:/8
		100:/16
		101:/32
HDMI_OD2	Apll_hdmi_aud_ctl<2:0>	000: /1
		001:/2
		010:/4
		011:/8
		100:/16

### 7.6.3.8 HDMI TX PLL

HDMI TX PLL diagram is shown in the following figure.

Figure 7-19 HDMI TX PLL



HDMI\_CLK\_OUT: min=187.5M max=6G.

Frac<18:0>:

18 +-	17 1	16 0.5	2^17=131072
-------	------	--------	-------------

OD control table:

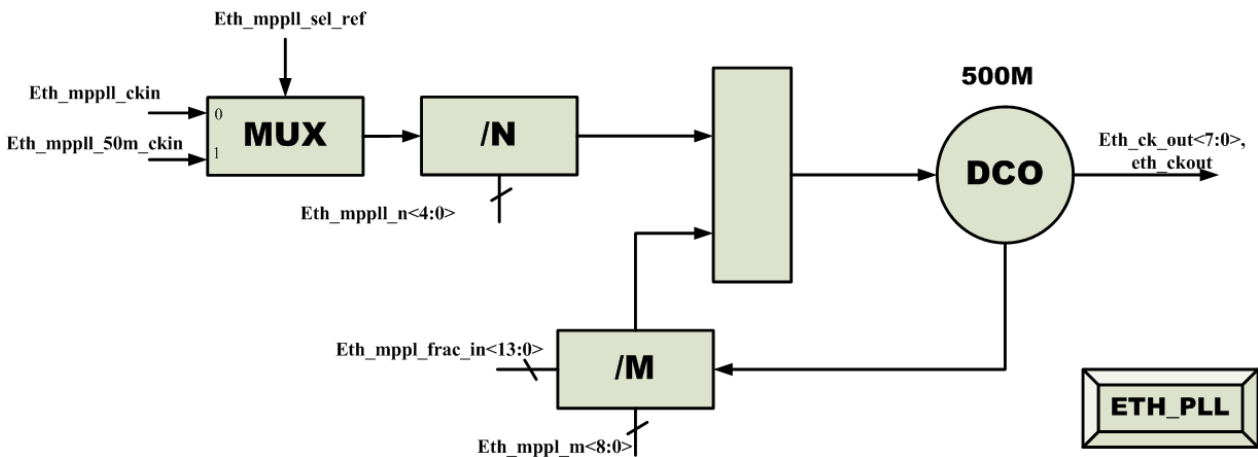
Block	Register	Function
OD1	HDMI_DPLL_OD<1:0>	00: /1
		01: /2
		10: /4
OD2	HDMI_DPLL_OD<3:2>	00: /1
		01: /2
		10: /4
OD3	HDMI_DPLL_OD2<1:0>	00: /1
		01: /2
		10: /4

Block	Register	Function
		11:/8

### 7.6.3.9 ETH PLL

ETH PLL diagram is shown in the following figure.

Figure 7-20 ETH PLL



DCO:500Mz

Eth\_ck\_out<7:0>=eth\_ckout=500MHz

$$f_{DCO} = f_{REF} \cdot (M + frac)/N$$

The ETH PLL fractional value weight table is shown below.

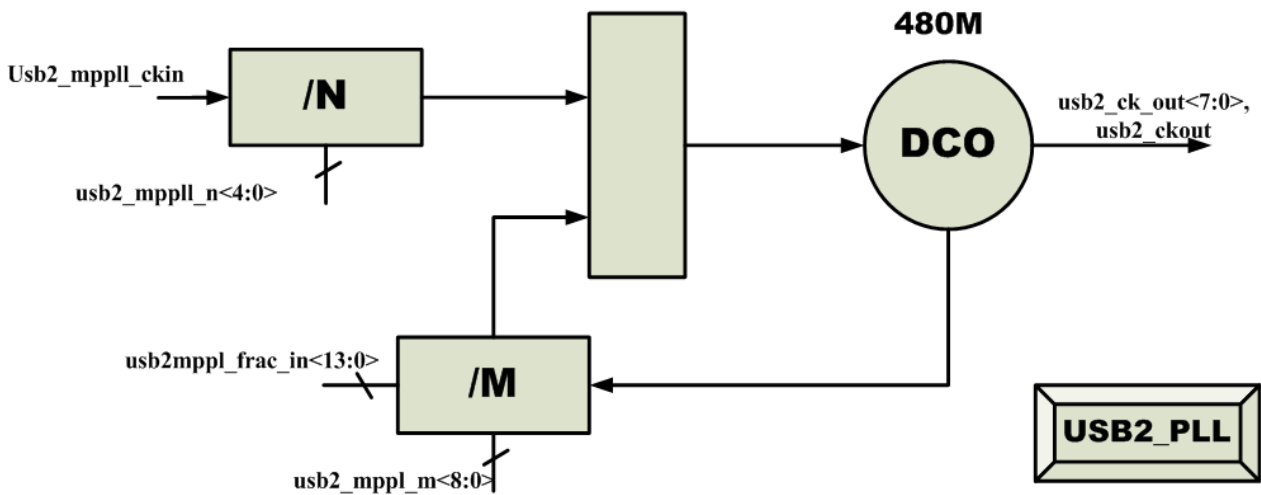
Table 7-105 ETH PLL Fractional Value Weight Table

	Bit13	Bit12	Bit11	Bit10	..	Bit0
Weight	1/2	1/2^2	1/2^3	1/2^4	..	1/2^14

### 7.6.3.10 USB2 PLL

USB2 PLL diagram is shown in the following figure.

Figure 7-21 USB2 PLL



DCO frequency is 480M.

Eth\_ck\_out<7:0>=eth\_ckout=480MHz

$$f_{DCO} = f_{REF} \cdot (M + frac)/N$$

The USB PLL fractional value weight table is shown below.

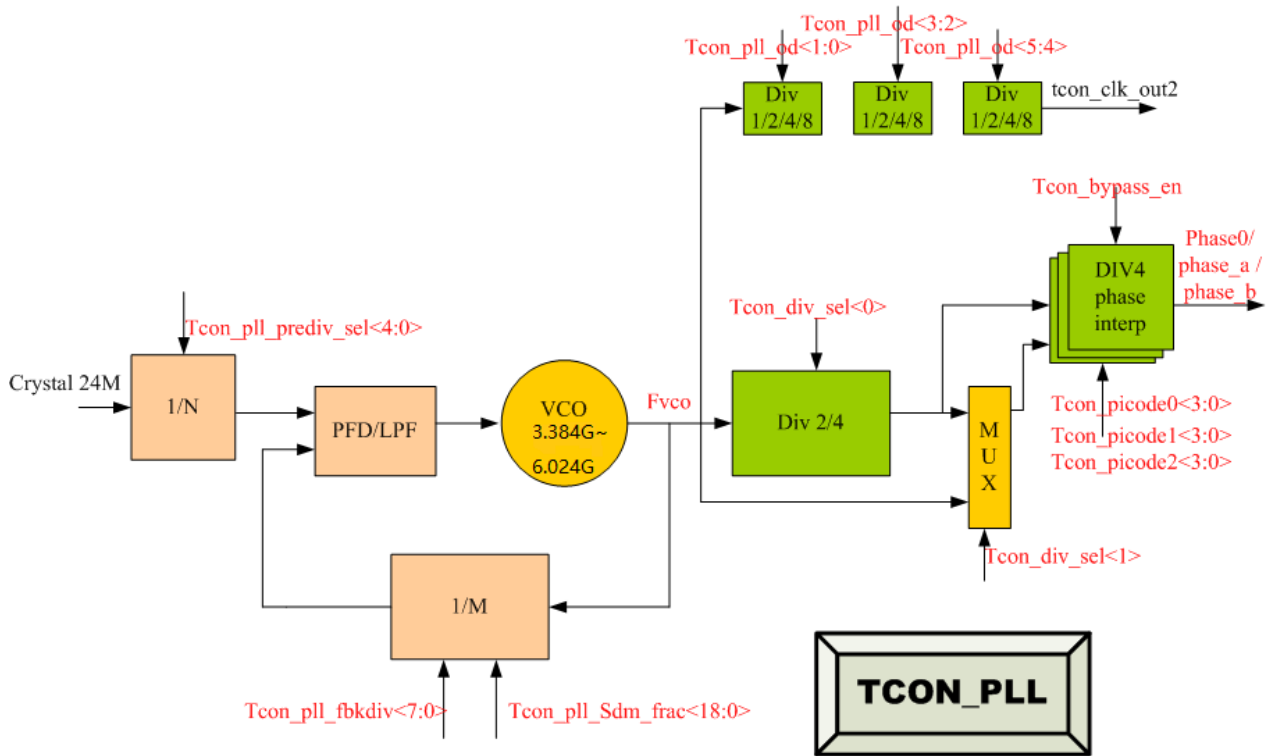
Table 7-106 USB PLL Fractional Value Weight Table

	Bit13	Bit12	Bit11	Bit10	..	Bit0
Weight	1/2	1/2^2	1/2^3	1/2^4	..	1/2^14

### 7.6.3.11 TCON PLL

TCON PLL diagram is shown in the following figure.

Figure 7-22 TCON PLL

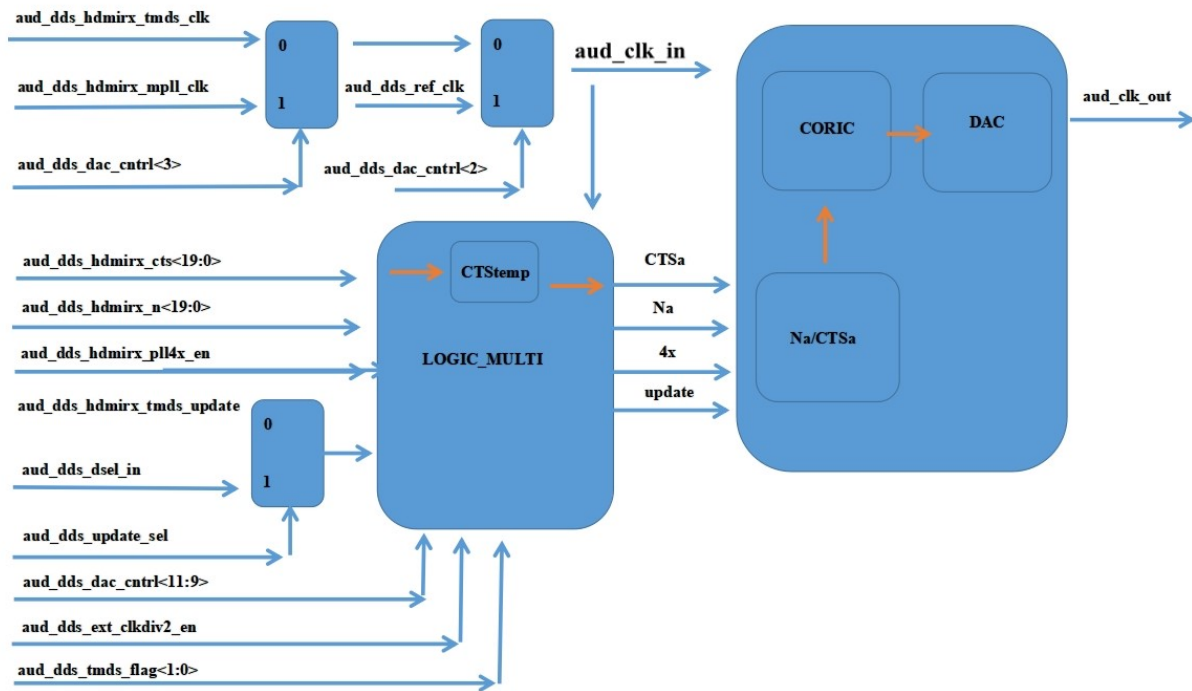


Wire	Register	Function
Tcon_pll_od<5:0>	tcon_pll_ctrl0<24:19>	00:/1
		01:/2
		10:/4
		11:/8
Tcon_bypass_en	tcon_pll_ctrl0<16>	0: /4
		1:/1
Tcon_div_sel<0>	tcon_pll_ctrl1<21>	0:/2
		1:/4
Tcon_div_sel<1>	tcon_pll_ctrl1<22>	0:from div2/4
		1:from vco



### 7.6.3.12 AUD DDS

Figure 7-23 AUD DDS PLL



#### Main function:

$aud\_clk\_out = Na/CTSa * aud\_clk\_in;$

#### Setting:

- **Na:**

`aud_dds_ext_clkdiv2_en=0` , `Na=aud_dds_hdmirx_n<19:0>`

`aud_dds_ext_clkdiv2_en=1` , `Na=aud_dds_hdmirx_n<19:0> *2` ;

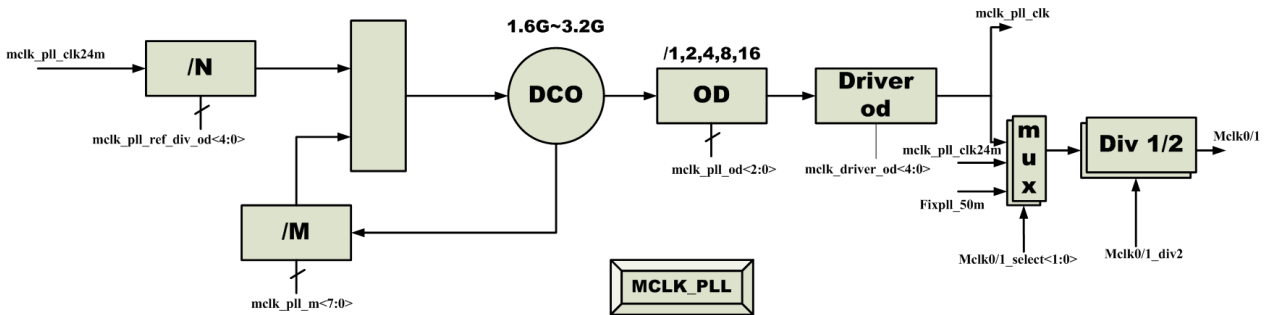
- 

- **CTSa:**

- `aud_dds_tmds_flag<1:0>=2'b0` , `CTSa=aud_dds_hdmirx_cts<19:0>`;
- `aud_dds_tmds_flag<1:0>=2'b1` , `CTSa=aud_dds_hdmirx_cts<19:0>*2`;
- `aud_dds_tmds_flag<1:0>=2'b10` , `CTSa=aud_dds_hdmirx_cts<19:0>*4`;
- `aud_dds_tmds_flag<1:0>=2'b11` , `CTSa=aud_dds_hdmirx_cts<19:0>*8`;
- 
- `aud_dds_dac_cntrl<11:9> =3'b0` , `CTSa= CTSa`
- `aud_dds_dac_cntrl<11:9> =3'b1` , `CTSa= CTSa*40`
- `aud_dds_dac_cntrl<11:9> =3'b010` , `CTSa= CTSa*4`
- `aud_dds_dac_cntrl<11:9> =3'b011` , `CTSa= CTSa*8`
- `aud_dds_dac_cntrl<11:9> =3'b100` , `CTSa= CTSa*5`
- `aud_dds_dac_cntrl<11:9> =3'b101` , `CTSa= CTSa*10`
- `aud_dds_dac_cntrl<11:9> =3'b110` , `CTSa= CTSa*16`
- `aud_dds_dac_cntrl<11:9> =3'b111` , `CTSa= CTSa*20`

### 7.6.3.13 MCLK\_PLL\_TOP

Figure 7-24 MCLK\_PLL\_TOP PLL



Register	Function
Mclk_pll_od	000: /1
	001:/2
	010:/4
	011:/8
	100:16
Mclk_driver_od	N=div N
Mclk0/1_select<1:0>	00:/mclk_pll_clk 01:/24M 10:fixpll_50m
Mclk0/1_div2	0: div1 1:div2

### 7.6.4 Clock Measure

Clock source is as follows:

Table 7-107 Clock Measure Source

No.	Clock Measure Source
222	cts_enc0_clk
221	cts_enc1_clk
220	cts_enc2_clk
219	cts_enc0_if_clk
218	am_ring_osc_clk_out[32]
217	am_ring_osc_clk_out[31]
216	am_ring_osc_clk_out[30]
215	am_ring_osc_clk_out[29]
214	am_ring_osc_clk_out[28]

No.	Clock Measure Source
213	am_ring_osc_clk_out[27]
212	am_ring_osc_clk_out[26]
211	am_ring_osc_clk_out[25]
210	am_ring_osc_clk_out[24]
209	am_ring_osc_clk_out[23]
208	am_ring_osc_clk_out[22]
207	am_ring_osc_clk_out[21]
206	am_ring_osc_clk_out[20]
205	vid_pll2_div_clk_out
204	vid_pll1_div_clk_out
203	mipi_csi_phy0__clk_out
202	mipi_csi_phy1__clk_out
201	mipi_csi_phy2__clk_out
200	mipi_csi_phy3__clk_out
199	am_ring_osc_clk_out[19]
198	am_ring_osc_clk_out[18]
197	am_ring_osc_clk_out[17]
196	am_ring_osc_clk_out[16]
195	am_ring_osc_clk_out[15]
194	am_ring_osc_clk_out[14]
193	am_ring_osc_clk_out[13]
192	am_ring_osc_clk_out[12]
191	am_ring_osc_clk_out[11]
190	am_ring_osc_clk_out[10]
189	am_ring_osc_clk_out[9]
188	am_ring_osc_clk_out[8]
187	am_ring_osc_clk_out[7]
186	am_ring_osc_clk_out[6]
185	am_ring_osc_clk_out[5]
184	am_ring_osc_clk_out[4]
183	am_ring_osc_clk_out[3]
182	am_ring_osc_clk_out[2]

No.	Clock Measure Source
181	am_ring_osc_clk_out[1]
180	am_ring_osc_clk_out[0]
179	rng_ring_3
178	rng_ring_2
177	rng_ring_1
176	rng_ring_0
175	
174	
173	a73_sys_cpu_clk_div16
172	a73_sys_pll_div16
171	mclk_pll_clk
170	cts_ACLKM
169	pwm_a_clk
168	pwm_b_clk
167	pwm_c_clk
166	pwm_d_clk
165	pwm_e_clk
164	pwm_f_clk
163	pwm_ao_a_clk
162	pwm_ao_b_clk
161	pwm_ao_c_clk
160	pwm_ao_d_clk
159	pwm_ao_e_clk
158	pwm_ao_f_clk
157	pwm_ao_g_clk
156	pwm_ao_h_clk
155	
154	
153	
152	
151	
150	
149	

No.	Clock Measure Source
148	earcrx_pll_dmac_ck
147	audio_resampleb_clk
146	audio_pdm_dclk
145	audio_spdifin_mst_clk
144	audio_spdifout_mst_clk
143	audio_spdifoutb_mst_clk
142	audio_pdm_sysclk
141	audio_resamplea_clk
140	audio_tadmin_a_sclk
139	audio_tadmin_b_sclk
138	audio_tadmin_c_sclk
137	audio_tadmin_lb_sclk
136	audio_tdmout_a_sclk
135	audio_tdmout_b_sclk
134	audio_tdmout_c_sclk
133	audio_locker_in_clk
132	audio_locker_out_clk
131	acodec_dac_clk_x128
130	audio_vad_clk
129	
128	
127	
126	
125	
124	ts_nna_clk
123	ts_a53_clk
122	ts_a73_clk
121	cts_ts_clk(temp sensor)
120	
119	cts_anakin_clk
118	cts_spicc_0_clk
117	cts_spicc_1_clk
116	i_gpio_msr

No.	Clock Measure Source
115	cts_sd_emmc_clk_A
114	cts_sd_emmc_clk_B
113	cts_sd_emmc_clk_C
112	
111	cts_sar_adc_clk
110	cts_sc_clk(smartcard)
109	cts_dsi0_phy_clk
108	cts_dsi1_phy_clk
107	
106	
105	hdmirx_dsd_clk
104	hdmirx_aud_clk
103	hdmirx_vid_clk
102	cts_hdmirx_meter_clk
101	cts_hdmirx_acr_ref_clk
100	cts_hdmirx_aud_pll_clk
99	cts_hevcf_clk
98	cts_hevcb_clk
97	cts_hcodec_clk
96	cts_wave521_bclk
95	cts_wave521_cclk
94	cts_wave521_aclk
93	cts_vdec_clk
92	
91	
90	
89	
88	
87	cts_spicc_2_clk
86	cts_spicc_3_clk
85	cts_spicc_4_clk
84	cts_spicc_5_clk
83	cts_hdmi_tx_pnx_clk

No.	Clock Measure Source
82	cts_hdmitx_aud_clk
81	cts_hdmitx_200m_clk
80	cts_hdmitx_prif_clk
79	
78	cts_hdmitx_fe_clk
77	cts_hdmitx_sys_clk
76	HDMITX_TMDS_CLK
75	
74	
73	cts_mipi_isp_clk
72	cts_mipi_csi_phy_clk
71	
70	cts_dsi_a_meas_clk
69	cts_aud_sck
68	cts_aud_pll_clk
67	cts_ge2d_clk
66	cts_vapbclk
65	cts_vid_lock_clk
64	cts_vpu_clkc
61	cts_vpu_clk
63	cts_vpu_clkb_tmp
62	cts_vpu_clkb
60	cts_vdin_meas_clk
59	cts_hdmi_tx_pixel_clk
58	lcd_an_clk_ph3
57	lcd_an_clk_ph2
56	mod_tcon_clko
55	cts_vpu_clk_buf
54	cts_vdac_clk
53	
52	
51	hdmi_vid_pll_clk
50	vid_pll0_div_clk_out

No.	Clock Measure Source
49	cts_hdmirx_hdcp2x_eclk
48	cts_hdmirx_cfg_clk
47	cts_hdmirx_2m_clk
46	cts_hdmirx_5m_clk
45	hdmirx_apll_clk_audio
44	hdmirx_cable_clk
43	hdmirx_tmnds_clk
42	hdmirx_apll_clk_out_div
41	cts_dsi_b_meas_clk
40	ephy_test_clk
39	eth_phy_plltxclk
38	eth_phy_rxclk
37	co_tx_clk
36	co_rx_clk
35	mod_eth_rx_clk_rmii
34	co_clkin_to_mac
33	cts_eth_clk_rmii
32	cts_eth_clk125Mhz
31	mod_eth_tx_clk
30	mod_eth_phy_ref_clk
29	cts_gdc_clk
28	cts_amlgdc_clk
27	paie1_clk_inn
26	paie1_clk_inp
25	earcrx_pll_ckout
24	ddr0_dppll_pt_clk
23	sys_pll_div16
22	eth_mppll_50m_ckout
21	gp1_pll_clk
20	gp0_pll_clk
19	hifi_pll_clk
18	mppll_clk_test_out



No.	Clock Measure Source
17	pcie_clk_inn
16	pcie_clk_inp
15	mp11_clk_50m
14	mp3_clk_out
13	mp2_clk_out
12	mp1_clk_out
11	mp0_clk_out
10	fclk_div5
9	
8	cts_cecb_clk
7	cts_ceca_clk
6	sys_cpu_clk_div16
5	cts_mali_clk
4	cts_dspb_clk
3	cts_dspa_clk
2	cts_rtc_clk
1	cts_axi_clk
0	cts_sys_clk

## Register Address

For below registers the base address is 0xfe048000.

Each register final address = BASE + address \* 4.

The following lists describe the mapping between each MSR\_CLK register and its address.

- MSR\_CLK\_REG0 0xfe048000
- MSR\_CLK\_REG1 0xfe048004
- MSR\_CLK\_REG2 0xfe048008
- MSR\_CLK\_REG3 0xfe04800c
- MSR\_CLK\_REG4 0xfe048010
- MSR\_CLK\_REG5 0xfe048014
- MSR\_CLK\_DUTY 0xfe048018

## Register Description

**Table 7-108 MSR\_CLK\_DUTY**

Bit(s)	R/W	Default	Description
31-16	R	0	DUTY_CNT_HIGH: This value represents the HIGH-count during duty cycle measurements
15-0	R	0	DUTY_CNT_LOW: This value represents the LOW-count during duty cycle measurements

**Table 7-109 MSR\_CLK\_REG0**

Bit(s)	R/W	Default	Description
31	R	0	BUSY: If this bit is set to 1, then the measure logic is busy
30-29	R	0	PROCESS_COUNT: This is a counter that simply increments every time a measurement is started
28	R	0	Limit Error
27-20	R/W	0	MSR_CLK_MUX_SEL: Selects the clock to measure. See example C-code
19	R/W	0	CLK_EN: The incoming clock is gated to prevent glitches from entering the measurement logic. Set this bit to 1 once the external clock has been established and is stable.
18	RW	0	One_shot_enable
17	R/W	0	CONT_EN: Set this bit to 1 to enable continuous frequency measurements
16	R/W	0	ENABLE: This is a general enable to enable the clock measurement circuit. If CONT_EN is set to 1 above, then the circuit will continually measure. If CONT_EN is set to 0, then the circuit will make one measurement. To restart another measurement, you must set ENABLE low then high.
15-0	R/W	0	MSR_TIME: This value establishes the gate time of the measurement with a 1uS resolution. The measurement module will count I2S MCLK clocks for N uS

**Table 7-110 MSR\_CLK\_REG1**

Bit(s)	R/W	Default	Description
31	R	0	DCYCLE_MSR_DONE: If this bit is set to 1, then the duty cycle measurement is done
30	R/W	0	unused
29	R/W	0	DCYCLE_ENABLE: Set this bit to enable start a duty cycle measurement
28	R/W	0	DCYCLE_CLK_EN: Set this bit to enable the high-frequency duty cycle clock
27-20	R/W	0	DCYCLE_TCNT: This value represents the number of uS to measure. This value should be less than 252.
19-0	R/W	0	unused

**Table 7-111 MSR\_CLK\_REG2**

Bit(s)	R/W	Default	Description
19-0	R	0	MEASURED_VALUE: This is the count of cycles measured during the programmed gate time.

**Table 7-112 MSR\_CLK\_REG3**

Bit(s)	R/W	Default	Description
19-0	R	0	Min_exp

**Table 7-113 MSR\_CLK\_REG4**

Bit(s)	R/W	Default	Description
19-0	R	0	Max_exp

**Table 7-114 MSR\_CLK\_REG5**

Bit(s)	R/W	Default	Description
0	R	0	Start_p

## 7.6.5 Register Description

### Register Address

For below registers, the base address is FE000000.

Each register final address = BASE + address \* 4.

The following lists describe the mapping between each register and its address.

- CLKCTRL\_OSCIN\_CTRL 0xfe000004
- CLKCTRL\_RTC\_BY\_OSCIN\_CTRL0 0xfe000008
- CLKCTRL\_RTC\_BY\_OSCIN\_CTRL1 0xfe00000c
- CLKCTRL\_RTC\_CTRL 0xfe000010
- CLKCTRL\_CHECK\_CLK\_RESULT 0xfe000014
- CLKCTRL\_MBIST\_ATSPEED\_CTRL 0xfe000018
- CLKCTRL\_LOCK\_BIT\_REG0 0xfe000020
- CLKCTRL\_LOCK\_BIT\_REG1 0xfe000024
- CLKCTRL\_LOCK\_BIT\_REG2 0xfe000028
- CLKCTRL\_LOCK\_BIT\_REG3 0xfe00002c
- CLKCTRL\_PROT\_BIT\_REG0 0xfe000030
- CLKCTRL\_PROT\_BIT\_REG1 0xfe000034
- CLKCTRL\_PROT\_BIT\_REG2 0xfe000038
- CLKCTRL\_PROT\_BIT\_REG3 0xfe00003c
- CLKCTRL\_SYS\_CLK\_CTRL0 0xfe000040
- CLKCTRL\_SYS\_CLK\_EN0\_REG0 0xfe000044
- CLKCTRL\_SYS\_CLK\_EN0\_REG1 0xfe000048
- CLKCTRL\_SYS\_CLK\_EN0\_REG2 0xfe00004c
- CLKCTRL\_SYS\_CLK\_EN0\_REG3 0xfe000050
- CLKCTRL\_SYS\_CLK\_EN1\_REG0 0xfe000054
- CLKCTRL\_SYS\_CLK\_EN1\_REG1 0xfe000058
- CLKCTRL\_SYS\_CLK\_EN1\_REG2 0xfe00005c

- CLKCTRL\_SYS\_CLK\_EN1\_REG3 0xfe000060
- CLKCTRL\_SYS\_CLK\_VPU\_EN0 0xfe000064
- CLKCTRL\_SYS\_CLK\_VPU\_EN1 0xfe000068
- CLKCTRL\_AXI\_CLK\_CTRL0 0xfe00006c
- CLKCTRL\_TST\_CTRL0 0xfe000080
- CLKCTRL\_TST\_CTRL1 0xfe000084
- CLKCTRL\_CECA\_CTRL0 0xfe000088
- CLKCTRL\_CECA\_CTRL1 0xfe00008c
- CLKCTRL\_CECB\_CTRL0 0xfe000090
- CLKCTRL\_CECB\_CTRL1 0xfe000094
- CLKCTRL\_SC\_CLK\_CTRL 0xfe000098
- CLKCTRL\_DSPA\_CLK\_CTRL0 0xfe00009c
- CLKCTRL\_DSPB\_CLK\_CTRL0 0xfe0000a0
- CLKCTRL\_CLK12\_24\_CTRL 0xfe0000a8
- CLKCTRL\_ANAKIN\_CLK\_CTRL 0xfe0000ac
- CLKCTRL\_GDC\_CLK\_CTRL 0xfe0000b0
- CLKCTRL\_AMLGDC\_CLK\_CTRL 0xfe0000b4
- CLKCTRL\_VID\_CLK0\_CTRL 0xfe0000c0
- CLKCTRL\_VID\_CLK0\_CTRL2 0xfe0000c4
- CLKCTRL\_VID\_CLK0\_DIV 0xfe0000c8
- CLKCTRL\_VIID\_CLK0\_DIV 0xfe0000cc
- CLKCTRL\_VIID\_CLK0\_CTRL 0xfe0000d0
- CLKCTRL\_ENC0\_HDMI\_CLK\_CTRL 0xfe0000d4
- CLKCTRL\_ENC2\_HDMI\_CLK\_CTRL 0xfe0000d8
- CLKCTRL\_ENC\_HDMI\_CLK\_CTRL 0xfe0000dc
- CLKCTRL\_HDMI\_CLK\_CTRL 0xfe0000e0
- CLKCTRL\_VID\_PLL\_CLK0\_DIV 0xfe0000e4
- CLKCTRL\_VPU\_CLK\_CTRL 0xfe0000e8
- CLKCTRL\_VPU\_CLKB\_CTRL 0xfe0000ec
- CLKCTRL\_VPU\_CLKC\_CTRL 0xfe0000f0
- CLKCTRL\_VID\_LOCK\_CLK\_CTRL 0xfe0000f4
- CLKCTRL\_VDIN\_MEAS\_CLK\_CTRL 0xfe0000f8
- CLKCTRL\_VAPBCLK\_CTRL 0xfe0000fc
- CLKCTRL\_MIPIDSI\_PHY\_CLK\_CTRL 0xfe000104
- CLKCTRL\_MIPI\_CSI\_PHY\_CLK\_CTRL 0xfe00010c
- CLKCTRL\_MIPI\_ISP\_CLK\_CTRL 0xfe000110
- CLKCTRL\_WAVE420L\_CLK\_CTRL 0xfe000114
- CLKCTRL\_WAVE420L\_CLK\_CTRL2 0xfe000118
- CLKCTRL\_HTX\_CLK\_CTRL0 0xfe00011c
- CLKCTRL\_HTX\_CLK\_CTRL1 0xfe000120
- CLKCTRL\_HRX\_CLK\_CTRL0 0xfe000128
- CLKCTRL\_HRX\_CLK\_CTRL1 0xfe00012c
- CLKCTRL\_HRX\_CLK\_CTRL2 0xfe000130
- CLKCTRL\_HRX\_CLK\_CTRL3 0xfe000134

- CLKCTRL\_VDEC\_CLK\_CTRL 0xfe000140
- CLKCTRL\_VDEC2\_CLK\_CTRL 0xfe000144
- CLKCTRL\_VDEC3\_CLK\_CTRL 0xfe000148
- CLKCTRL\_VDEC4\_CLK\_CTRL 0xfe00014c
- CLKCTRL\_WAVE521\_CLK\_CTRL 0xfe000150
- CLKCTRL\_WAVE521\_CLK\_CTRL2 0xfe000154
- CLKCTRL\_TS\_CLK\_CTRL 0xfe000158
- CLKCTRL\_MALI\_CLK\_CTRL 0xfe00015c
- CLKCTRL\_VIPNANOQ\_CLK\_CTRL 0xfe000160
- CLKCTRL\_ETH\_CLK\_CTRL 0xfe000164
- CLKCTRL\_NAND\_CLK\_CTRL 0xfe000168
- CLKCTRL\_SD\_EMMC\_CLK\_CTRL 0xfe00016c
- CLKCTRL\_BT656\_CLK\_CTRL 0xfe000170
- CLKCTRL\_SPICC\_CLK\_CTRL 0xfe000174
- CLKCTRL\_GEN\_CLK\_CTRL 0xfe000178
- CLKCTRL\_SAR\_CLK\_CTRL0 0xfe00017c
- CLKCTRL\_PWM\_CLK\_AB\_CTRL 0xfe000180
- CLKCTRL\_PWM\_CLK\_CD\_CTRL 0xfe000184
- CLKCTRL\_PWM\_CLK\_EF\_CTRL 0xfe000188
- CLKCTRL\_PWM\_CLK\_AO\_AB\_CTRL 0xfe0001a0
- CLKCTRL\_PWM\_CLK\_AO\_CD\_CTRL 0xfe0001a4
- CLKCTRL\_PWM\_CLK\_AO\_EF\_CTRL 0xfe0001a8
- CLKCTRL\_PWM\_CLK\_AO\_GH\_CTRL 0xfe0001ac
- CLKCTRL\_SPICC\_CLK\_CTRL1 0xfe0001c0
- CLKCTRL\_SPICC\_CLK\_CTRL2 0xfe0001c4
- CLKCTRL\_VID\_CLK1\_CTRL 0xfe0001cc
- CLKCTRL\_VID\_CLK1\_CTRL2 0xfe0001d0
- CLKCTRL\_VID\_CLK1\_DIV 0xfe0001d4
- CLKCTRL\_VIID\_CLK1\_DIV 0xfe0001d8
- CLKCTRL\_VIID\_CLK1\_CTRL 0xfe0001dc
- CLKCTRL\_VID\_CLK2\_CTRL 0xfe0001e0
- CLKCTRL\_VID\_CLK2\_CTRL2 0xfe0001e4
- CLKCTRL\_VID\_CLK2\_DIV 0xfe0001e8
- CLKCTRL\_VIID\_CLK2\_DIV 0xfe0001ec
- CLKCTRL\_VIID\_CLK2\_CTRL 0xfe0001f0
- CLKCTRL\_VID\_PLL\_CLK1\_DIV 0xfe0001f4
- CLKCTRL\_VID\_PLL\_CLK2\_DIV 0xfe0001f8
- CLKCTRL\_MIPI\_DSI\_MEAS\_CLK\_CTRL 0xfe000200
- CLKCTRL\_HDMI\_VID\_PLL\_CLK\_DIV 0xfe000204
- CLKCTRL\_TIMESTAMP\_CTRL 0xfe000400
- CLKCTRL\_TIMESTAMP\_CTRL1 0xfe000404
- CLKCTRL\_TIMESTAMP\_CTRL2 0xfe00040c
- CLKCTRL\_TIMESTAMP\_RD0 0xfe000410
- CLKCTRL\_TIMESTAMP\_RD1 0xfe000414

- CLKCTRL\_TIMEBASE\_CTRL0 0xfe000418
- CLKCTRL\_TIMEBASE\_CTRL1 0xfe00041c
- CLKCTRL\_EFUSE\_CPU\_CFG0 0xfe000480
- CLKCTRL\_EFUSE\_CPU\_CFG2 0xfe000484
- CLKCTRL\_EFUSE\_ENCP\_CFG0 0xfe000488
- CLKCTRL\_EFUSE\_MALI\_CFG01 0xfe00048c
- CLKCTRL\_EFUSE\_HEVCB\_CFG01 0xfe000490
- CLKCTRL\_EFUSE\_HEVCB\_CFG2 0xfe000494
- CLKCTRL\_EFUSE\_LOCK 0xfe000498
- CLKCTRL\_EFUSE\_A73\_CFG01 0xfe00049c
- CLKCTRL\_EFUSE\_A73\_CFG2 0xfe0004a0

## Register Description

Osc from XTAL pad will connect all PLL and some analog modules:

And it will gated by reg\_oscin\_ctrl;

**Table 7-115 CLKCTRL\_OSCIN\_CTRL**

Bit(s)	R/W	Default	Description
14	R/W	1	clock gate en; xtal -> HDMIPLL
13	R/W	1	Clock gate en Xtal -> sys1
12	R/W	1	clock gate en; xtal -> EARC
11	R/W	1	clock gate en; xtal -> PCIE_REFCLK_PLL
10	R/W	1	clock gate en; xtal -> ETH_PLL
9	R/W	1	clock gate en; xtal -> PCIECTRL/USBCTRL
8	R/W	1	clock gate en; xtal -> MCLK_PLL
7	R/W	1	clock gate en; xtal -> USB_PLL1
6	R/W	1	clock gate en; xtal -> USB_PLL0
5	R/W	1	clock gate en; xtal -> TCON0/1/2PLL
4	R/W	1	clock gate en; xtal -> PLL_TOP(FIXPLL/HIFIPLL/SYS0PLL/GP0PLL/GP1PLL)
3	R/W	1	Clock gate en: Xtal -> aud pll
1	R/W	1	clock gate en; xtal -> DDR_PLL

There are 4 vid\_pll\_clk used, as below table.

Table 7-116 vid\_pll\_clk List

PLL source	Generate Clock	Control Register	Register Address
HDMI_PLL	hdmi_vid_pll_clk	CLKCTRL_VID_PLL_CLK_DIV	0xfe000204
TCON_PLL0	vid_pll0_div_clk_out	COMBO_DPHY_VID_PLL0_DIV	0xfe018008
TCON_PLL1	vid_pll1_div_clk_out	COMBO_DPHY_VID_PLL1_DIV	0xfe01800c
TCON_PLL2	vid_pll2_div_clk_out	COMBO_DPHY_VID_PLL2_DIV	0xfe018010

Table 7-117 CLKCTRL\_VID\_PLL\_CLK\_DIV/COMBO\_DPHY\_VID\_PLL0/1/2\_DIV

Bit(s)	R/W	Default	Description
31~24	R	0	RESERVED
23~20	R/W	0	Reserved
19	R/W	0	CLK_FINAL_EN
18	R/W	0	CLK_DIV1
17~16	R/W	0	CLK_SEL
15	R/W	0	SET_PRESET
14-0	R/W	0	SHIFT_PRESET

RTC clock is a low frequency clock which needn't PLL.

it can generated by XTAL of from PAD.

Table 7-118 CLKCTRL\_RTC\_BY\_OSCIN\_CTRL0

Bit(s)	R/W	Default	Description
31	R/W	1	clock in gate en
30	R/W	1	clock out gate en
28	R/W	0	0: freq_out = freq_in / N0; 1: freq_out = freq_in / ((N0*M0+N1*M1)/(M0+M1)) Because XTAL is 24MHz only, so it can't generate some frequency like 32768Hz. If need 24KHz, set N0=1000 and this bit = 0, then rtc_clk = 24M/1000= 24KHz. If need 32768Hz, set N0=733, N1=732, M0 =2, M1=3, and this bit=1, then rtc_clk = 24M/(733*2+732*3)*(2+3)= 32768.9787Hz
23:12	R/W	'd731	clock div N1, if you want div8, set to 7
11:0	R/W	'd732	clock div N0, if you want div8, set to 7

Table 7-119 CLKCTRL\_RTC\_BY\_OSCIN\_CTRL1

Bit(s)	R/W	Default	Description
24	R/W	1	1: force clock_out = clock_in
23:12	R/W	'd10	clock div M1, if you want div8, set to 7
11:0	R/W	'd7	clock div M0, if you want div8, set to 7

Table 7-120 CLKCTRL\_RTC\_CTRL

Bit(s)	R/W	Default	Description
31	R	0	mux1 sel o
30	R	0	mux0b sel o
29	R	0	mux0a sel o
1:0	R/W	0	cts_rtc_clk source select: 0: oscin; 1:rtc from oscin div(N0/N1); 2:rtc from PAD; 3:off;

Table 7-121 CLKCTRL\_CHECK\_CLK\_RESULT

Bit(s)	R/W	Default	Description
2	R	0	cts_encp_clk frequency check result
1	R	0	cts_hevcb_clk frequency check result
0	R	0	cts_mali_clk frequency check result

Table 7-122 CLKCTRL\_MBIST\_ATSPEED\_CTRL

Bit(s)	R/W	Default	Description
0	R/W	0	0: mbist clk will select oscine; 1: mbist clk will select cts_***_clk;

lock/prot are two different register, they all can lock one clk\_ctrl register by different bit, the difference of lock/prot is lock can write 1 only and prot can write both 0 and 1.

Table 7-123 CLKCTRL\_LOCK/PROT\_BIT\_REG0~3

bit	Reg0	Reg1	Reg2	Reg3
0	CLKCTRL_OSCIN_CTRL	CLKCTRL_VIID_CLK_CTRL	CLKCTRL_SYS_CLK_EN1_REG1	
1		CLKCTRL_VIID_CLK_DIV	CLKCTRL_SYS_CLK_EN1_REG2	
2		CLKCTRL_VID_LOCK_CLK_CTRL	CLKCTRL_SYS_CLK_EN1_REG3	
3			CLKCTRL_RTC_BY_OSCIN_CTRL0	
4			CLKCTRL_RTC_BY_OSCIN_CTRL1	
5		CLKCTRL_SPICC_CLK_CTRL	CLKCTRL_RTC_CTRL	
6		CLKCTRL_PWM_CLK_AB_CTRL	CLKCTRL_RAMA_CLK_CTRL0	
7	CLKCTRL_SC_CLK_CTRL	CLKCTRL_PWM_CLK_CD_CTRL	CLKCTRL_CLK12_24_CTRL	



bit	Reg0	Reg1	Reg2	Reg3
8	CLKCTRL_CECA_CTRL0	CLKCTRL_PWM_CLK_EF_CTRL	CLKCTRL_ANAKIN_CLK_CTRL	
9	CLKCTRL_CECA_CTRL1		CLKCTRL_SPICC_CLK_CTRL1	
10	CLKCTRL_CECB_CTRL0		CLKCTRL_SPICC_CLK_CTRL2	
11	CLKCTRL_CECB_CTRL1	CLKCTRL_ETH_CLK_CTRL	CLKCTRL_	
12	CLKCTRL_TST_CTRL0		CLKCTRL_GDC_CLK_CTRL	
13	CLKCTRL_TST_CTRL1	CLKCTRL_TS_CLK_CTRL	CLKCTRL_AMLGDC_CLK_CTRL	
14	CLKCTRL_MALI_CLK_CTRL	CLKCTRL_TIME-STAMP_CTRL	CLKCTRL_MIPI_DSI_MEAS_CLK_CTRL	
15	CLKCTRL_VDEC_CLK_CTRL	CLKCTRL_TIME-STAMP_CTRL1	CLKCTRL_PWM_CLK_AO_AB_CTRL	
16	CLKCTRL_VDEC2_CLK_CTRL	CLKCTRL_TIME-STAMP_CTRL2	CLKCTRL_PWM_CLK_AO_CD_CTRL	
17	CLKCTRL_VDEC3_CLK_CTRL	CLKCTRL_TIMEBASE_CTRL0	CLKCTRL_PWM_CLK_AO_EF_CTRL	
18	CLKCTRL_VDEC4_CLK_CTRL	CLKCTRL_TIMEBASE_CTRL1	CLKCTRL_PWM_CLK_AO_GH_CTRL	
19	CLKCTRL_MIPI_ISP_CLK_CTRL	CLKCTRL_SAR_CLK_CTRL0	CLKCTRL_VID_CLK0_CTRL	
20	CLKCTRL_MIPI_CSI_PHY_CLK_CTRL	CLKCTRL_DSPA_CLK_CTRL0	CLKCTRL_VID_CLK0_CTRL2	
21	CLKCTRL_VPU_CLK_CTRL	CLKCTRL_DSPB_CLK_CTRL0	CLKCTRL_VID_CLK0_DIV	
22	CLKCTRL_VPU_CLKC_CTRL	CLKCTRL_SYS_CLK_CTRL0	CLKCTRL_VID_CLK1_CTRL	
23	CLKCTRL_VAPBCLK_CTRL	CLKCTRL_AXI_CLK_CTRL0	CLKCTRL_VID_CLK1_CTRL2	
24	CLKCTRL_VPU_CLKB_CTRL	CLKCTRL_MBIST_AT-SPEED_CTRL	CLKCTRL_VID_CLK1_DIV	
25	CLKCTRL_GEN_CLK_CTRL	CLKCTRL_SYS_CLK_VPU_EN0	CLKCTRL_VID_CLK2_CTRL	
26	CLKCTRL_MIPIDSI_PHY_CLK_CTRL	CLKCTRL_SYS_CLK_VPU_EN1	CLKCTRL_VID_CLK2_CTRL2	
27	CLKCTRL_VDIN_MEAS_CLK_CTRL	CLKCTRL_SYS_CLK_EN0_REG0	CLKCTRL_VID_CLK2_DIV	
28	CLKCTRL_NAND_CLK_CTRL	CLKCTRL_SYS_CLK_EN0_REG1	CLKCTRL_HDMI_CLK_CTRL	
29	CLKCTRL_SD_EMMC_CLK_CTRL	CLKCTRL_SYS_CLK_EN0_REG2	CLKCTRL_ENC_HDMI_CLK_CTRL	
30	CLKCTRL_WAVE521_CLK_CTRL	CLKCTRL_SYS_CLK_EN0_REG3	CLKCTRL_ENC0_HDMI_CLK_CTRL	

bit	Reg0	Reg1	Reg2	Reg3
31	CLKCTRL_WAVE521_CLK_CTRL2	CLKCTRL_SYS_CLK_EN1_REG0	CLKCTRL_ENC2_HDMI_CLK_CTRL	
96				CLKCTRL_HTX_CLK_CTRL0
97				CLKCTRL_HTX_CLK_CTRL1
98				CLKCTRL_HRX_CLK_CTRL0
99				CLKCTRL_HRX_CLK_CTRL1
100				CLKCTRL_HRX_CLK_CTRL2
101				CLKCTRL_HRX_CLK_CTRL3
102				CLKCTRL_VIID_CLK1_CTRL
103				CLKCTRL_VIID_CLK1_DIV
104				CLKCTRL_VIID_CLK2_CTRL
105				CLKCTRL_VIID_CLK2_DIV
106				CLKCTRL_HDMI_VID_PLL_CLK_DIV

SYS\_CLK is a system level clock, most of control bus and some data bus worked on this clock domain.

**Table 7-124 CLKCTRL\_SYS\_CLK\_CTRL0**

Bit(s)	R/W	Default	Description
31	R	0	no glitch mux sel out, if it's not equal to sel in for long time, mean the FSM is dead because no clock in
29	R/W	0	pre_b clock gate en
28:26	R/W	0	pre_b source select: 0:cts_oscin_clk; 1:fclk_div2; 2:fclk_div3; 3:fclk_div4 4:fclk_div5 5:axi_clk_frcpu 7:cts_rtc_clk
25:16	R/W	0	pre_b div N, if you want div8, set to 7
15	R/W	0	no glitch mux sel in 0: select pre_a; 1: select pre_b;
13	R/W	1	pre_a clock gate en

Bit(s)	R/W	Default	Description
12:10	R/W	0	pre_a source select: 0:cts_oscclk; 1:fclk_div2; 2:fclk_div3; 3:fclk_div4; 4:fclk_div5 5:axi_clk_frcpu 7:cts_rtc_clk
9:0	R/W	0	pre_a div N, if you want div8, set to 7

Each bit will enabled one user, if some bits are not defined, mean they are reserved.

En0 equivalent to en1.

**Table 7-125 CLKCTRL\_SYS\_CLK\_EN0\_REG0~3/ CLKCTRL\_SYS\_CLK\_EN1\_REG0~3**

bit	user	bit	user	bit	user	bit	user
127		95	Ts_gpu	63	i2c_m_b	31	ir_ctrl
126		94	gic	62	i2c_m_a	30	msr_clk
125		93		61	l2c_ao_b	29	spifc
124		92	sar_adc	60	l2c_m_ao_a	28	acodec
123		91		59	pciephy	27	smart-card
122		90		58	usb	26	sdemmc-C
121		89	vpu_intr	57		25	sdemmcB
120		88		56	pcie	24	sdemmcA
119		87		55		23	Am2axi2
118		86	dspb	54	spicc1	22	Am2axi1
117		85	dspa	53	spicc0	21	Am2axi0
116		84	A73_pclk_sys_cpu_apb	52	ge2d	20	Ampipe_eth
115		83	Pclk_Sys_cpu_apb	51		19	ampipe_nand
114		82	rsa	50		18	deswarp
113		81	Mipi_isp_pclk	49	Spicc_5	17	gdc
112		80		48	ts_a53	16	cec
111		79		47	ts_a73	15	
110		78		46	Spicc_4	14	aucpu
109		77		45	Spicc_3	13	aocpu
108		76		44	Spicc_2	12	
107		75	mmc_pclk	43	aiffo	11	

106		74		42	uart_f	10	
105	pwm_ef	73	Hdmirx_pclk	41	uart_e	9	
104	pwm_cd	72		40	uart_d	8	
103	pwm_ab	71	Hdmi20_ace_clk	39	uart_c	7	
102	Pwm_ao_gh	70		38	uart_b	6	mali
101	Pwm_ao_ef	69	i2c_s_a	37	uart_a	5	
100	Pwm_ao_cd	68	hdmitx_pclk	36		4	ethphy
99	Pwm_ao_ab	67	i2c_m_f	35	eth	3	mipi_dsi_b
98	Ts_hevc	66	i2c_m_e	34		2	mipi_dsi_a
97	Ts_vpu	65	i2c_m_d	33		1	dos
96	Ts_nna	64	i2c_m_c	32	audio	0	ddr

**Table 7-126 CLKCTRL\_SYS\_CLK\_VPU\_EN0/ CLKCTRL\_SYS\_CLK\_VPU\_EN1 the same as CLKCTRL\_SYS\_CLK\_EN0 & EN1, but it's connect to VPU and gate clock internal.**

Bit(s)	R/W	Default	Description
0	R/W	1	TODO

**Table 7-127 CLKCTRL\_AXI\_CLK\_CTRL0**

Bit(s)	R/W	Default	Description
31	R	0	no glitch mux sel out, if it's not equal to sel in for long time, mean the FSM is dead because no clock in
29	R/W	0	pre_b clock gate en
28:26	R/W	0	pre_b source select: 0:cts_oscin_clk; 1:fclk_div2; 2:fclk_div3; 3:fclk_div4 4:fclk_div5 5:axi_clk_frcpu 7:cts_rtc_clk
25:16	R/W	0	pre_b div N, if you want div8, set to 7
15	R/W	0	no glitch mux sel in 0: select pre_a; 1: select pre_b;
13	R/W	1	pre_a clock gate en

Bit(s)	R/W	Default	Description
12:10	R/W	0	pre_a source select: 0:cts_oscclk; 1:fclk_div2; 2:fclk_div3; 3:fclk_div4; 4:fclk_div5; 5:axi_clk_frcpu; 7:cts_rtc_clk
9:0	R/W	0	pre_a div N, if you want div8, set to 7

There are 4 test sources can connect to PINMUX.

Each pin can select by 2bits of CLKCTRL\_TST\_CTRL0/1.

The test sources are listed as blow:

**Table 7-128 CLKCTRL\_TST\_CTRL0/1**

Test Source	Select	Default	src0	src1
TEST_OUT22	CTRL1[13:12]	3	nna[1]	
TEST_OUT21	CTRL1[11:10]	3	nna[0]	
TEST_OUT20	CTRL1[9:8]	3	rama	
TEST_OUT19	CTRL1[7:6]	3	ramb	
TEST_OUT18	CTRL1[5:4]	3	isp	
TEST_OUT17	CTRL1[3:2]	3	ddr1	
TEST_OUT16	CTRL1[1:0]	3	ddr0	
TEST_OUT15	CTRL0[31:30]	3	dos_wave[2]	
TEST_OUT14	CTRL0[29:28]	3	dos_wave[1]	
TEST_OUT13	CTRL0[27:26]	3	dos_wave[0]	
TEST_OUT12	CTRL0[25:24]	3	dos[3]	
TEST_OUT11	CTRL0[23:22]	3	dos[2]	
TEST_OUT10	CTRL0[21:20]	3	dos[1]	
TEST_OUT9	CTRL0[19:18]	3	dos[0]	A73[4]
TEST_OUT8	CTRL0[17:16]	3	vpu[2]	A73[3]
TEST_OUT7	CTRL0[15:14]	3	vpu[1]	A73[2]
TEST_OUT6	CTRL0[13:12]	3	vpu[0]	A73[1]
TEST_OUT5	CTRL0[11:10]	3	gpu[2]	A73[0]
TEST_OUT4	CTRL0[9:8]	3	gpu[1]	A53[4]
TEST_OUT3	CTRL0[7:6]	3	gpu[0]	A53[3]
TEST_OUT2	CTRL0[5:4]	3	dmc1	A53[2]
TEST_OUT1	CTRL0[3:2]	3	dmc0	A53[1]
TEST_OUT0	CTRL0[1:0]	3	dsp	A53[0]

Table 7-129 CLKCTRL\_CECA/B\_CLK\_CTRL0

Bit(s)	R/W	Default	Description
31	R/W	0	Gclk_in: 0:disable clk in;1:enable clk in;
30	R/W	0	Gclk_out: 0:disable clk out; 1: enable clk out;
29-28	R/W	0	Div_list_max: 0:only use clk_div0; 1:use clk_div0/clk_div1 in turn;
27-24	R/W	0	Reserved
23-12	R/W	0	Div_tcmt1: freq_clk_div1 = freq_clk_in/div_tcmt1;
11-0	R/W	0	Div_tcmt0: freq_clk_div0 = freq_clk_in/div_tcmt0;

Table 7-130 CLKCTRL\_CECA/B\_CLK\_CTRL1

Bit(s)	R/W	Default	Description
31	R/W	0	clk_sel for cts_cec_clk 0: from clock divider 1: from cts_rtc_clk
30-25	R	0	Reserved
24	R/W	0	Divide_by_1: 1: clk_out = clk_in; 0: clk_out = clk_div0 or 1;
23-12	R/W	0	Cycle_tcmt1 After Cycle_tcmt1 times clk_div1, clk_out will equal clk_div0; Need set div_list_max = 1;
11-0	R/W	0	Cycle_tcmt0 After Cycle_tcmt0 times clk_div, clk_out will equal clk_div1; Need set div_list_max = 1;

Table 7-131 CLKCTRL\_SC\_CLK\_CTRL

Bit(s)	R/W	Default	Description
10:9	R/W	0	clk_sel for cts_sc_clk(smart card) 0: from fclk_div4; 1: from fclk_div3; 2: from fclk_div5; 3: from cts_oscinc_clk;
8	R/W	0	clock enable
7:0	R/W	0	div N, if you want div8, set to 7
31	R	0	no glitch mux sel out, if it's not equal to sel in for long time, mean the FSM is dead because no clock in
29	R/W	0	pre_b clock gate en
28:26	R/W	0	pre_b source select: 0:cts_oscinc_clk; 1:fclk_div2; 2:fclk_div3; 3:fclk_div4; 4:fclk_div5;

Bit(s)	R/W	Default	Description
			5:axi_clk_frcpu; 6:fclk_div7; 7:cts_rtc_clk;
25:16	R/W	0	pre_b div N, if you want div8, set to 7
15	R/W	0	no glitch mux sel in 0: select pre_a; 1: select pre_b;
13	R/W	0	pre_a clock gate en
12:10	R/W	0	pre_a source select: 0:cts_oscin_clk; 1:fclk_div2; 2:fclk_div3; 3:fclk_div4; 4:fclk_div5; 5:axi_clk_frcpu; 6:fclk_div7; 7:cts_rtc_clk;
9:0	R/W	0	pre_a div N, if you want div8, set to 7

Table 7-132 CLKCTRL\_DSPA/B\_CLK\_CTRL0

Bit(s)	R/W	Default	Description
31	R	0	no glitch mux sel out, if it's not equal to sel in for long time, mean the FSM is dead because no clock in
29	R/W	0	pre_b clock gate en
28:26	R/W	0	pre_b source select: 0:cts_oscin_clk; 1:fclk_div2p5; 2:fclk_div3; 3:fclk_div5 4:hifi_pll 5:fclk_div4 6:fclk_div7 7:cts_rtc_clk
25:16	R/W	0	pre_b div N, if you want div8, set to 7
15	R/W	0	no glitch mux sel in 0: select pre_a; 1: select pre_b;
13	R/W	0	pre_a clock gate en
12:10	R/W	0	pre_a source select: 0:cts_oscin_clk; 1:fclk_div2p5; 2:fclk_div3; 3:fclk_div5 4:hifi_pll 5:fclk_div4 6:fclk_div7 7:cts_rtc_clk
9:0	R/W	0	pre_a div N, if you want div8, set to 7

Table 7-133 CLKCTRL\_CLK12\_24\_CTRL

Bit(s)	R/W	Default	Description
19~13	R/	0	RESERVED
12	R/W	0	crt_clk25_en; for CLK25M(refer to pinmux define) set 1 :CLK25M = 2GHz/ clk25_div; set 0 : CLK25M = 0;
11	R/W	0	crt_clk24_en; for CLK12M_24M(refer to pinmux define) set 1 :CLK12M_24M = 24M; set 0 : CLK12M_24M = 0;
10	R/W	0	clk24_div2_en; for CLK12M_24M(refer to pinmux define) need set crt_clk24_en = 1 first; set 1 :CLK12M_24M = 12M; set 0 : CLK12M_24M = 24M;
7:0	R/W	0	clk25_div; for CLK25M(refer to pinmux define) if need 8, set as 7; if set 0, will div2; etc: set 79, CLK25M = 2000MHz/80=25MHz;

Table 7-134 CLKCTRL\_VID\_CLK\_DIV/CTRL/CTRL2 + CLKCTRL\_VIID\_CLK\_DIV/CTRL +CLKCTRL\_HDMI\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31-21	R/W	0	TCON_CLK0_CTRL
20	R/W	0	CLK_EN1
19	R/W	0	CLK_EN0
18-16	R/W	0	CLK_IN_SEL
15	R/W	0	SOFT_RESET
14	R/W	0	PH23_ENABLE
13	R/W	0	DIV12_PH23
12-5	R/W	0	UNUSED
4	R/W	0	DIV12_EN
3	R/W	0	DIV6_EN
2	R/W	0	DIV4_EN
1	R/W	0	DIV2_EN
0	R/W	0	DIV1_EN

Table 7-135 CLKCTRL\_VID\_CLK1/2\_CTRL2

Bit(s)	R/W	Default	Description
31-17	R/W	0	
15-9	R/W	0	Reserved
10	R/W	1	Gclk_hdmi_tx_pnx_clk
9	R/W	1	Gclk_hdmi_tx_fe_clk
8	R/W	0	Atv demod vdac gated clock control



Bit(s)	R/W	Default	Description
7	R/W	1	LCD_AN_CLK_PHY2 gated clock control. 1 = enable
6	R/W	1	LCD_AN_CLK_PH3 gated clock control
5	R/W	1	HDMI_TX_PIXEL_CLK gated clock control
4	R/W	1	VDAC_clk gated clock control
3	R/W	1	ENCL gated clock control
2	R/W	1	ENCP gated clock control
1	R/W	1	ENCT gated clock control
0	R/W	1	ENCI gated clock control

Table 7-136 CLKCTRL\_VID\_CLK0\_CTRL2

Bit(s)	R/W	Default	Description
31-17	R/W	0	
16	R/W	0	Cts_enc0_if_clk div 0:div1 1:div2
15-9	R/W	0	Reserved
10	R/W	1	Gclk_hdmi_tx_pnx_clk
9	R/W	1	Gclk_hdmi_tx_fe_clk
8	R/W	0	Atv demod vdac gated clock control
7	R/W	1	LCD_AN_CLK_PHY2 gated clock control. 1 = enable
6	R/W	1	LCD_AN_CLK_PH3 gated clock control
5	R/W	1	HDMI_TX_PIXEL_CLK gated clock control
4	R/W	1	VDAC_clk gated clock control
3	R/W	1	ENCL gated clock control
2	R/W	1	ENCP gated clock control
1	R/W	1	ENCT gated clock control
0	R/W	1	ENCI gated clock control

Table 7-137 CLKCTRL\_VID\_CLK0/1/2\_DIV

Bit(s)	R/W	Default	Description
31-28	R/W	0	ENCI_CLK_SEL
27-24	R/W	0	ENCP_CLK_SEL
23-20	R/W	0	ENCT_CLK_SEL
19-18	R/W	0	UNUSED
17	R/W	0	CLK_DIV_RESET
16	R/W	0	CLK_DIV_EN

Bit(s)	R/W	Default	Description
15-8	R/W	1	XD1
7-0	R/W	1	XD0

Table 7-138 CLKCTRL\_VIID\_CLK0/1/2\_CTRL

Bit(s)	R/W	Default	Description
31-20	R/W	0	Unused
19	R/W	0	V2_cntl_clk_en0
18-16	R/W	0	V2_cntl_clk_in_sel
15	R/W	0	V2_cntl_soft_reset
14-5	R/W	0	Unused
4	R/W	0	V2_cntl_div12_en
3	R/W	0	V2_cntl_div6_en
2	R/W	0	V2_cntl_div4_en
1	R/W	0	V2_cntl_div2_en
0	R/W	0	V2_cntl_div1_en

Table 7-139 CLKCTRL\_VIID\_CLK0/1/2\_DIV

Bit(s)	R/W	Default	Description
31-28	R/W	0	DAC0_CLK_SEL
27-24	R/W	0	DAC1_CLK_SEL
23-20	R/W	0	DAC2_CLK_SEL
19	R/W	0	Select adc_pll_clk_b2 to be cts_clk_vdac
18	R/W	0	Unused
17	R/W	0	V2_cntl_clk_div_reset
16	R/W	0	V2_cntl_clk_div_en
15-12	R/W	0	Encl_clk_sel
14-8	R/W	0	Unused
7-0	R/W	0	V2_cntl_xd0

Table 7-140 CLKCTRL\_HDMI\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31-20	R/W	0	Reserved
19~16	R/W	0	Reserved
15~11	R/W	0	Reserved
10~9	R/W	0	CLK_SEL: 0:oscin; 1:fclk_div4;

Bit(s)	R/W	Default	Description
			2:clk_div3; 3:clk_div5;
8	R/W	0	CLK_EN:
7	R/W	0	Reserved
6-0	R/W	0	div N, if you want div8, set to 7

Table 7-141 CLKCTRL\_VID\_LOCK\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31~10	R/W	0	reserved
9-8	R/W	0	Clk_sel: 0:cts_oscin_clk 1:cts_encl_clk 2:cts_enci_clk 3:cts_encp_clk
7	R/W	0	Clk_en
6-0	R/W	0	div N, if you want div8, set to 7

Table 7-142 CLKCTRL\_ETH\_CLK\_CTRL

Bit(s)	R/W	Default	Description
16	R/W	0	invert clk_rmii_pad_i
13	R/W	0	invert clk_rmii_pad_o
12	R/W	0	clk_rmii_pad_o; 0: rmii_clk; 1: rmii_div2;
11:9	R/W	0	rmii_clk; clk_sel; 0: fclk_div2; 7: clk_rmii_pad_i;
8	R/W	0	rmii_clk; clk_en
7	R/W	0	eth_clk125M; clk_en;
6:0	R/W	0	rmii_clk; div N, if you want div8, set to 7

This following clock is generated for temp sensor.

Table 7-143 CLKCTRL\_TS\_CLK\_CTRL

Bit(s)	R/W	Default	Description
8	R/W	0	cts_ts_clk clk_en
7:0	R/W	0	cts_ts_clk div N, if you want div8, set to 7

Table 7-144 CLKCTRL\_MALI\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31	R/W	0	final mux; 0: select mux_a; 1: select mux_b;
30-28	R/W	0	Reserved
27~25	R/W	0	mux_b CLK_SEL: 0: cts_oscin_clk 1: gp0_pll 2: hifi_pll 3: fclk_div2p5 4: fclk_div3 5: fclk_div4 6: fclk_div5 7: fclk_div7
24	R/W	0	mux_b CLK_EN:
23	R/W	0	Reserved
22~16	R/W	0	mux_b div N, if you want div8, set to 7:
15~12	R/W	0	Reserved
11~9	R/W	0	mux_a CLK_SEL: 0: cts_oscin_clk 1: gp0_pll 2: hifi_pll 3: fclk_div2p5 4: fclk_div3 5: fclk_div4 6: fclk_div5 7: fclk_div7
8	R/W	0	mux_a CLK_EN:
7	R/W	0	Reserved
6-0	R/W	0	mux_a div N, if you want div8, set to 7:

CLKCTRL\_VDEC/2/3/4\_CLK\_CTRL

Table 7-145 CLKCTRL\_VDEC\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31-28	R/W	0	Reserved
27~25	R/W	0	cts_hcodec_clk, mux_a CLK_SEL: 0:fclk_div2p5; 1:fclk_div3; 2:fclk_div4; 3:fclk_div5; 4:fclk_div7; 5:hifi_pll; 6:gp0pll; 7:oscin;
24	R/W	0	cts_hcodec_clk, mux_a

Bit(s)	R/W	Default	Description
			CLK_EN:
23	R/W	0	Reserved
22~16	R/W	0	cts_hcodec_clk, mux_a div N, if you want div8, set to 7:
15~12	R/W	0	Reserved
11~9	R/W	0	cts_vdec_clk, mux_a CLK_SEL: 0:fclk_div2p5; 1:fclk_div3; 2:fclk_div4; 3:fclk_div5; 4:fclk_div7; 5:hifi_pll; 6:gp0pll; 7:oscin;
8	R/W	0	cts_vdec_clk, mux_a CLK_EN:
7	R/W	0	Reserved
6-0	R/W	0	cts_vdec_clk, mux_a div N, if you want div8, set to 7:

Table 7-146 \*CLKCTRL\_VDEC2\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31-28	R/W	0	Reserved
27~25	R/W	0	cts_hevcb_clk, mux_a: 0:fclk_div2p5; 1:fclk_div3; 2:fclk_div4; 3:fclk_div5; 4:fclk_div7; 5:hifi_pll; 6:gp0pll; 7:oscin;
24	R/W	0	cts_hevcb_clk, mux_a: CLK_EN:
23	R/W	0	Reserved
22~16	R/W	0	cts_hevcb_clk, mux_a: div N, if you want div8, set to 7:
15~12			Reserved
11~9	R/W	0	cts_hevcf_clk, mux_a: CLK_SEL: 0:fclk_div2p5; 1:fclk_div3; 2:fclk_div4; 3:fclk_div5; 4:fclk_div7; 5:hifi_pll; 6:gp0pll; 7:oscin;
8	R/W	0	cts_hevcf_clk, mux_a: CLK_EN:

Bit(s)	R/W	Default	Description
7	R/W	0	Reserved
6-0	R/W	0	cts_hevcf_clk, mux_a: div N, if you want div8, set to 7:

Table 7-147 CLKCTRL\_VDEC3\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31	R/W	1	cts_hcodec_clk 0: sel mux_a; 1: sel mux_b;
27~25	R/W	0	cts_hcodec_clk, mux_b CLK_SEL: 0:fclk_div2p5; 1:fclk_div3; 2:fclk_div4; 3:fclk_div5; 4:fclk_div7; 5:hifi_pll; 6:gp0pll; 7:oscin;
24	R/W	0	cts_hcodec_clk, mux_b CLK_EN:
23	R/W	0	Reserved
22~16	R/W	0	cts_hcodec_clk, mux_b div N, if you want div8, set to 7:
15	R/W	1	cts_vdec_clk 0: sel mux_a 1: sel mux_b
11~9	R/W	0	cts_vdec_clk, mux_b CLK_SEL: 0:fclk_div2p5; 1:fclk_div3; 2:fclk_div4; 3:fclk_div5; 4:fclk_div7; 5:hifi_pll; 6:gp0pll; 7:oscin;
8	R/W	0	cts_vdec_clk, mux_b CLK_EN:
7	R/W	0	Reserved
6-0	R/W	0	cts_vdec_clk, mux_b div N, if you want div8, set to 7:

Table 7-148 CLKCTRL\_VDEC4\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31	R/W	1	cts_hevc_b_clk 0: use mux_a 1: use mux_b
27~25	R/W	0	cts_hevc_b_clk, mux_b: CLK_SEL:

Bit(s)	R/W	Default	Description
			0:clk_div2p5; 1:clk_div3; 2:clk_div4; 3:clk_div5; 4:clk_div7; 5:hifi_pll; 6:gp0pll; 7:oscin;
24	R/W	0	cts_hevcb_clk, mux_b: CLK_EN:
23	R/W	0	Reserved
22~16	R/W	0	cts_hevcb_clk, mux_b: div N, if you want div8, set to 7:
15	R/W	1	cts_hevcf_clk: 0: use mux_a; 1: use mux_b;
11~9	R/W	0	cts_hevcf_clk, mux_b: CLK_SEL: 0:clk_div2p5; 1:clk_div3; 2:clk_div4; 3:clk_div5; 4:clk_div7; 5:hifi_pll; 6:gp0pll; 7:oscin;
8	R/W	0	cts_hevcf_clk, mux_b: CLK_EN:
7	R/W	0	Reserved
6-0	R/W	0	cts_hevcf_clk, mux_b: div N, if you want div8, set to 7:

Table 7-149 CLKCTRL\_WAVE521\_CLK\_CTRL

Bit(s)	R/W	Default	Description
27:25	R/W	0	Wave521_cclk clk_sel; 0:oscin; 1:clk_div4; 2:clk_div3; 3:clk_div5; 4:clk_div7; 5:mp1l2; 6:mp1l3; 7:gp0pll;
24	R/W	0	Wave521_cclk clk_en
22:16	R/W	0	Wave521_cclk div N, if you want div8, set to 7
11:9	R/W	0	Wave521_bclk clk_sel; 0:oscin; 1:clk_div4; 2:clk_div3; 3:clk_div5; 4:clk_div7; 5:mp1l2; 6:mp1l3; 7:gp0pll;

Bit(s)	R/W	Default	Description
8	R/W	0	Wave521_bclk clk_en
6:0	R/W	0	Wave521_bclk div N, if you want div8, set to 7

Table 7-150 CLKCTRL\_WAVE521\_CLK\_CTRL2

Bit(s)	R/W	Default	Description
11:9	R/W	0	Wave521_ack clk_sel; 0:oscin; 1:fclk_div4; 2:fclk_div3; 3:fclk_div5; 4:fclk_div7; 5:mpll2; 6:mpll3; 7:gp0pll;
8	R/W	0	Wave521_ack clk_en
6:0	R/W	0	Wave521_ack div N, if you want div8, set to 7

Table 7-151 CLKCTRL\_VPU\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31	R/W	0	Final mux sel 0: use mux_a 1: use mux_b
30-29	R/W	0	Reserved
27~25	R/W	0	mux_b CLK_SEL: 0:fclk_div3; 1:fclk_div4; 2:fclk_div5; 3:fclk_div7; 4:mpll1; 5:vid_pll; 6:hifi_pll; 7:gp0_pll;
24	R/W	0	mux_b CLK_EN:
23	R/W	0	Reserved
22~16	R/W	0	mux_b div N, if you want div8, set to 7:
11~9	R/W	0	mux_a CLK_SEL: 0:fclk_div3; 1:fclk_div4; 2:fclk_div5; 3:fclk_div7; 4:mpll1; 5:vid_pll; 6:hifi_pll; 7:gp0_pll;
8	R/W	0	mux_a CLK_EN:
7	R/W	0	Reserved
6-0	R/W	0	mux_a div N, if you want div8, set to 7:



Table 7-152 CLKCTRL\_VPU\_CLKB\_CTRL

Bit(s)	R/W	Default	Description
31-25	R/W	0	Reserved
24	R/W	0	Cts_vpu_clkb_tmp en
21-20	R/W	0	Cts_vpu_clkb_tmp sel 0:cts_vpu_clk 1:fclk_div4 2:fclk_div5 3:fclk_div7
19-16	R/W	0	Cts_vpu_clkb_tmp div
9	R/W	0	Vpu_clkb_latch_en
8	R/W	0	Vpu_clkb_en
7-0	R/W	0	Vpu_clkb_div

Table 7-153 CLKCTRL\_VPU\_CLKC\_CTRL

Bit(s)	R/W	Default	Description
31	R/W	1	Final mux sel 0:mux_a 1:mux_b
30-29	R/W	0	Reserved
27~25	R/W	0	mux_b CLK_SEL: 0:fclk_div4; 1:fclk_div3; 2:fclk_div5; 3:fclk_div7; 4:mp11; 5:vid_pll; 6:mp12; 7:gp0pll;
24	R/W	0	mux_b CLK_EN:
23	R/W	0	Reserved
22~16	R/W	0	mux_b div N, if you want div8, set to 7:
15~12			
11~9	R/W	0	mux_a CLK_SEL: 0:fclk_div4; 1:fclk_div3; 2:fclk_div5; 3:fclk_div7; 4:mp11; 5:vid_pll; 6:mp12; 7:gp0pll;
8	R/W	0	mux_a CLK_EN:
7	R/W	0	Reserved
6-0	R/W	0	mux_a div N, if you want div8, set to 7:

Table 7-154 CLKCTRL\_VAPBCLK\_CTRL

Bit(s)	R/W	Default	Description
31	R/W	0	Final mux sel 0:mux_a 1:mux_b
30	R/W	0	cts_ge2d_clk enable
29	R/W	0	Reserved
27~25	R/W	0	mux_b CLK_SEL: 0:fclk_div4; 1:fclk_div3; 2:fclk_div5; 3:fclk_div7; 4:mp11; 5:vid_pll; 6:mp12; 7:fclk_div2p5;
24	R/W	0	mux_b CLK_EN:
23	R/W	0	Reserved
22~16	R/W	0	mux_b div N, if you want div8, set to 7:
15~12			
11~9	R/W	0	mux_a CLK_SEL: 0:fclk_div4; 1:fclk_div3; 2:fclk_div5; 3:fclk_div7; 4:mp11; 5:vid_pll; 6:mp12; 7:fclk_div2p5;
8	R/W	0	mux_a CLK_EN:
7	R/W	0	Reserved
6-0	R/W	0	mux_a div N, if you want div8, set to 7:

Table 7-155 CLKCTRL\_HDCP22\_CTRL

Bit(s)	R/W	Default	Description
26-25	R/W	0	cts_hdc22_skpclk Clk_sel: 0:cts_oscin_clk 1:fclk_div4 2:fclk_div3 3:fclk_div5
24	R/W	0	cts_hdc22_skpclk Clk_en
22:16	R/W	0	cts_hdc22_skpclk div N, if you want div8, set to 7
10:9	R/W	0	cts_hdc22_esmclk Clk_sel: 0:fclk_div7 1:fclk_div4 2:fclk_div3

Bit(s)	R/W	Default	Description
			3:clk_div5
8	R/W	0	cts_hdcp22_esmclk Clk_en
6:0	R/W	0	cts_hdcp22_esmclk div N, if you want div8, set to 7

Table 7-156 CLKCTRL\_VDIN\_MEAS\_CLK\_CTRL

Bit(s)	R/W	Default	Description
11-9	R/W	0	cts_vdin_meas_clk CLK_SEL: 0:oscin; 1:clk_div4; 2:clk_div3; 3:clk_div5; 4:vid_pll;
8	R/W	0	cts_vdin_meas_clk CLK_EN:
6:0	R/W	0	cts_vdin_meas_clk div N, if you want div8, set to 7

Table 7-157 CLKCTRL\_NAND\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31-12	R/W	0	unused
11-9	R/W	0	CLK_SEL: 0:cts_oscin_clk 1:clk_div2 2:clk_div3 3:hifi_pll_clk 4:clk_div2p5 5:mp2_clk_out 6:mp3_clk_out 7:gp0_pll_clk
8			Reserved
7	R/W	1	CLK_EN:
6-0	R/W	0	div N, if you want div8, set to 7

Table 7-158 CLKCTRL\_SD\_EMMC\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31-28	R/W	0	unused
27-25	R/W	0	Sd_emmc_B_CLK_SEL: 0:cts_oscin_clk 1:clk_div2 2:clk_div3 3:hifi_pll_clk 4:clk_div2p5 5:mp2_clk_out 6:mp3_clk_out 7:gp0_pll_clk

Bit(s)	R/W	Default	Description
24	R/W		Reserved
23	R/W	1	Sd_emmc_B_CLK_EN:
22-16	R/W	0	Sd_emmc_B_CLK div N, if you want div8, set to 7
15-12	R/W	0	reseverd
11-9	R/W	0	Sd_emmc_A_CLK_SEL: 0:cts_oscin_clk 1:fclk_div2 2:fclk_div3 3:hifi_pll_clk 4:fclk_div2p5 5:mp2_clk_out 6:mp3_clk_out 7:gp0_pll_clk
8	R/W		Reserved
7	R/W	1	Sd_emmc_A_CLK_EN:
6-0	R/W	0	Sd_emmc_A_CLK div N, if you want div8, set to 7

Table 7-159 CLKCTRL\_SPICC\_CLK\_CTRL

Bit(s)	R/W	Default	Description
25:23	R/W	0	spicc_1_clk clk_sel 0:osc; 1:sys_clk; 2:fclk_div4; 3:fclk_div3; 4:fclk_div2; 5:fclk_div5; 6:fclk_div7; 7:gp0_pll;
22	R/W	0	spicc_1_clk clk_en
21:16	R/W	0	spicc_1_clk div N, if you want div8, set to 7
9:7	R/W	0	spicc_0_clk clk_sel 0:osc; 1:sys_clk; 2:fclk_div4; 3:fclk_div3; 4:fclk_div2; 5:fclk_div5; 6:fclk_div7; 7:gp0_pll;
6	R/W	0	spicc_0_clk clk_en
5:0	R/W	0	spicc_0_clk div N, if you want div8, set to 7

**Table 7-160 CLKCTRL\_PWM\_CLK\_(AO\_)\*\*\_CTRL**

Bit(s)	R/W	Default	Description
26:25	R/W	0	cts_pwm_j/h/f/d/b_clk clk_sel: 0: osin 1: vid_pll 2: fclk_div4 3: fclk_fiv3
24	R/W	0	cts_pwm_j/h/f/d/b_clk clk_en
23:16	R/W	0	cts_pwm_j/h/f/d/b_clk div N, if you want div8, set to 7
10:9	R/W	0	cts_pwm_i/g/e/c/a_clk clk_sel: 0: osin 1: vid_pll 2: fclk_div4 3: fclk_fiv3
8	R/W	0	cts_pwm_i/g/e/c/a_clk clk_en
7:0	R/W	0	cts_pwm_i/g/e/c/a_clk div N, if you want div8, set to 7

**Table 7-161 CLKCTRL\_SAR\_CLK\_CTRL0**

Bit(s)	R/W	Default	Description
10:9	R/W	0	clk_sel: 0: osin 1: cts_sys_clk1
8	R/W	0	clk_en
7:0	R/W	0	div N, if you want div8, set to 7

**Table 7-162 CLKCTRL\_GEN\_CLK\_CTRL**

Bit(s)	R/W	Default	Description
16:12	R/W	0	gen_clk source select 0: cts_oscin_clk; 1: cts_rtc_clk; 2: sys_pll_div16; 3: ddr_pll_div32 4: vid_pll 5: gp0_pll 6: gp1_pll 7: hifi_pll 8: pcie_ref_pll_n 9: pcie_ref_pll_p 12: clk_msr_src(select from all internal clock except PLLs); 16: ACODEC_DAC1 17: sys_cpu_clk_div16 19: fclk_div2 20: fclk_div2p5 21: fclk_div3 22: fclk_div4 23: fclk_div5 24: fclk_div7 25: mpll0 26: mpll1 27: mpll2 28: mpll3
11	R/W	0	gen_clk gate en
10:0	R/W	0	gen_clk div N, if you want div8, set to 7

**Table 7-163 CLKCTRL\_TIMESTAMP\_CTRL0**

Bit(s)	R/W	Default	Description
14	R/W	0	1: hold count val;
13	R/W	0	1: force count val = {ctrl2,ctrl1}
12	R/W	1	enable
11:10	R/W	0	clk source sel: 0: cts_oscin_clk; 1: cts_rtc_clk; 2: cts_sys_clk; 3: axi_clk;
9	R/W	1	clk gate en
8:0	R/W	23	clk div N, if you want div8, set to 7

**Table 7-164 CLKCTRL\_TIMESTAMP\_CTRL1**

Bit(s)	R/W	Default	Description
31-0	R/W	0	force value[31:0]

**Table 7-165 CLKCTRL\_TIMESTAMP\_CTRL2**

Bit(s)	R/W	Default	Description
31-0	R/W	0	force value[63:32]

**Table 7-166 CLKCTRL\_TIMESTAMP\_VAL0**

Bit(s)	R/W	Default	Description
31:0	R/W	0	count val [31:0], need update by write any value first

**Table 7-167 CLKCTRL\_TIMESTAMP\_VAL1**

Bit(s)	R/W	Default	Description
31:0	R	0	count val [63:32], need update by write val0 register any value first

**Table 7-168 CLKCTRL\_TIMEBASE\_CTRL0**

Bit(s)	R/W	Default	Description
31:24	R/W	0	N1, if you want div8, set to 8
23:19	R/W	0	N2, if you want div8, set to 8
18	R/W	0	1: enable xtal(24M) tick
17	R/W	1	Crystal clock enable.
16	R/W	0	Soft Reset
15:14	R/W	0	1MS_TICK: 0 = use (100US_TICK/10), 1 = hold to 1, 2 = use counter1, 3 = use counter 2
13:12	R/W	0	100US_TICK: 0 = use (10US_TICK/10), 1 = hold to 1, 2 = use counter1, 3 = use counter 2
11:10	R/W	0	10US_TICK: 0 = use (1US_TICK/10), 1 = hold to 1, 2 = use counter1, 3 = use counter 2
9:8	R/W	0	1US_TICK: 0 = use (Crystal/XTAL_DIV), 1 = hold to 1, 2 = use counter1, 3 = use counter 2
7:6	R/W	0	XTAL_DIV3_TICK: 0 = use Crystal/3, 1 = hold to 1, 2 = use counter1, 3 = use counter 2
5:0	R/W	24	XTAL_DIV, if you want div24000, set to 24

**Table 7-169 CLKCTRL\_TIMEBASE\_CTRL1**

Bit(s)	R/W	Default	Description
1	R/W	0	1: sync xtal_div2 by fall edge of xtal;
0	R/W	0	1: enable xtal_div2(12M) tick

**Table 7-170 CLKCTRL\_EFUSE\_CPU\_CFG01**

Bit(s)	R/W	Default	Description
31:16	R/W	0xffff	efuse lock a53 clk cfg1; cfg1 must < cfg0;
15:0	R/W	0xffff	efuse lock a53 clk cfg0;

**Table 7-171 CLKCTRL\_EFUSE\_CPU\_CFG2**

Bit(s)	R/W	Default	Description
15:0	R/W	0xffff	efuse lock a53 clk cfg2; cfg2 must < cfg1;

**Table 7-172 CLKCTRL\_EFUSE\_ENCP\_CFG0**

Bit(s)	R/W	Default	Description
15:0	R/W	0xffff	efuse lock cts_encp_clk cfg0

**Table 7-173 CLKCTRL\_EFUSE\_MALI\_CFG01**

Bit(s)	R/W	Default	Description
31:16	R/W	0xffff	efuse lock cts_mali_clk cfg1, cfg1 must < cfg0
15:0	R/W	0xffff	efuse lock cts_mali_clk cfg0

**Table 7-174 CLKCTRL\_EFUSE\_HEVCB\_CFG01**

Bit(s)	R/W	Default	Description
31:16	R/W	0xffff	efuse lock cts_hevcb_clk cfg1, cfg1 must < cfg0
15:0	R/W	0xffff	efuse lock cts_hevcb_clk cfg0

**Table 7-175 CLKCTRL\_EFUSE\_HEVCB\_CFG2**

Bit(s)	R/W	Default	Description
15:0	R/W	0xffff	efuse lock cts_hevcb_clk cfg2, cfg2 must < cfg1

**Table 7-176 CLKCTRL\_EFUSE\_A73\_CFG01**

Bit(s)	R/W	Default	Description
31:16	R/W	0xffff	efuse lock a73 clk cfg1; cfg1 must < cfg0;
15:0	R/W	0xffff	efuse lock a73 clk cfg0;

**Table 7-177 CLKCTRL\_EFUSE\_A73\_CFG2**

Bit(s)	R/W	Default	Description
15:0	R/W	0xffff	efuse lock a73 clk cfg2; cfg2 must < cfg1;

**Table 7-178 CLKCTRL\_EFUSE\_LOCK**

Bit(s)	R/W	Default	Description
3	R/W	0	lock EFUSE_HEVCB_CFG01/2
2	R/W	0	lock EFUSE_MALI_CFG01



Bit(s)	R/W	Default	Description
1	R/W	0	lock EFUSE_ENCP_CFG0
0	R/W	0	lock EFUSE_CPU_CFG01/2

Table 7-179 CLKCTRL\_MIPI\_ISP\_PHY\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31	R	1	mux sel out,
24	R/W	0	Mux1 clock gate en
27:25	R/W	0	Mux1 source select: 0:cts_oscinc_clk; 1:gp0_pll_clk; 2:mp1_clk; 3:mp2_clk 4:fclk_div3 5:fclk_div4 6:fclk_div5 7:fclk_div7
22:16	R/W	0	Mux1 div N, if you want div8, set to 7
8	R/W	1	Mux0 clock gate en
11:9	R/W	0	Mux0 source select: 0:cts_oscinc_clk; 1:gp0_pll_clk; 2:mp1_clk; 3:mp2_clk 4:fclk_div3 5:fclk_div4 6:fclk_div5 7:fclk_div7
6:0	R/W	0	Mux0 div N, if you want div8, set to 7

Table 7-180 CLKCTRL\_MIPIDSI\_PHY\_CLK\_CTRL

Bit(s)	R/W	Default	Description
27-25	R/W	0	cts_dsi1_phy_clk: 0:vid_pll1_clk 1:gp1_pll_clk 2:hifi_pll 3:mp1_clk 4:fclk_div2 5:fclk_div2p5 6:fclk_div3 7:fclk_div7
24	R/W	0	cts_dsi1_phy_clkClk_en
22:16	R/W	0	cts_dsi1_phy_clkl div N, if you want div8, set to 7
14:12	R/W	0	cts_dsi0_phy_clk: 0:vid_pll1_clk 1:gp1_pll_clk 2:hifi_pll 3:mp1_clk 4:fclk_div2 5:fclk_div2p5

Bit(s)	R/W	Default	Description
			6:fclk_div3 7:fclk_div7
8	R/W	0	cts_dsi0_phy_clk Clk_en
6:0	R/W	0	cts_dsi0_phy_clk div N, if you want div8, set to 7

Table 7-181 CLKCTRL\_ANAKIN\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31	R/W	1	Final_mux_sel
30	R/W	0	Clk_en
27-25	R/W	0	cts_anakin_clk: 0:fclk_div4 1:fclk_div3 2:fclk_div5 3fclk_div2 4:mp1_clk 5:vid_pll1_clk 6mp2_clk 7:fclk_div2p5
24	R/W	0	cts_anakin_mux1_Clk_en
22:16	R/W	0	cts_anakin_mux1_Clk div N, if you want div8, set to 7
11:9	R/W	0	cts_anakin_clk: 0:fclk_div4 1:fclk_div3 2:fclk_div5 3fclk_div2 4:mp1_clk 5:vid_pll1_clk 6mp2_clk 7:fclk_div2p5
8	R/W	0	cts_anakin_mux0_Clk_en
6:0	R/W	0	cts_anakin_mux0_Clk div N, if you want div8, set to 7

Table 7-182 CLKCTRL\_GDC\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31	R/W	1	Final_mux_sel
30	R/W	0	Clk_en
27-25	R/W	0	cts_gdc_clk: 0:fclk_div4 1:fclk_div3 2:fclk_div5 3fclk_div2 4:mp1_clk 5:vid_pll1_clk 6mp2_clk 7:fclk_div2p5

Bit(s)	R/W	Default	Description
24	R/W	0	cts_gdc_mux1_Clk_en
22:16	R/W	0	cts_gdc_mux1_Clk div N, if you want div8, set to 7
11:9	R/W	0	cts_gdc_clkl: 0:fclk_div4 1:fclk_div3 2:fclk_div5 3fclk_div2 4:mp1_clk 5:vid_pll1_clk 6mp2_clk 7:fclk_div2p5
8	R/W	0	cts_gdc_mux0_Clk_en
6:0	R/W	0	cts_gdc_mux0_Clk div N, if you want div8, set to 7

Table 7-183 CLKCTRL\_AMLGDC\_CLK\_CTRL

Bit(s)	R/W	Default	Description
31	R/W	1	Final_mux_sel
30	R/W	0	Clk_en
27-25	R/W	0	cts_amlgdc_clkl: 0:fclk_div4 1:fclk_div3 2:fclk_div5 3fclk_div2 4:mp1_clk 5:vid_pll1_clk 6mp2_clk 7:fclk_div2p5
24	R/W	0	cts_amlgdc_mux1_Clk_en
22:16	R/W	0	cts_amlgdc_mux1_Clk div N, if you want div8, set to 7
11:9	R/W	0	cts_amlgdc_clkl: 0:fclk_div4 1:fclk_div3 2:fclk_div5 3fclk_div2 4:mp1_clk 5:vid_pll1_clk 6mp2_clk 7:fclk_div2p5
8	R/W	0	cts_amlgdc_mux0_Clk_en
6:0	R/W	0	cts_amlgdc_mux0_Clk div N, if you want div8, set to 7

Table 7-184 CLKCTRL\_MIPI\_DSI\_MEAS\_CLK\_CTRL

Bit(s)	R/W	Default	Description
23-21	R/W	0	cts_mipi_dsi_b_meas_clk: 0:oscin 1:fclk_div4 2:fclk_div3 3fclk_div5 4:vid_pll0_clk 5:gp0 6fclk_div2 7:fclk_div7
20	R/W	0	cts_mipi_dsi_b_meas_mux1_Clk_en
18:12	R/W	0	cts_mipi_dsi_b_meas_mux1_Clk div N, if you want div8, set to 7
11:9	R/W	0	cts_mipi_dsi_a_meas_clk: 0:oscin 1:fclk_div4 2:fclk_div3 3fclk_div5 4:vid_pll0_clk 5:gp0 6fclk_div2 7:fclk_div7
8	R/W	0	cts_mipi_dsi_a_meas_mux0_Clk_en
6:0	R/W	0	cts_mipi_dsi_a_meas_mux0_Clk div N, if you want div8, set to 7

Table 7-185 CLKCTRL\_ENC\_HDMI\_CLK\_CTRL

Bit(s)	R/W	Default	Description
22:21	R/W	0	cts_hdmi_tx_pnx_clk: 0:enc0_hdmi_tx_pnx_clk 1:0 2:enc2_hdmi_tx_pnx_clk 30
20	R/W	0	cts_hdmi_tx_pnx_Clk_en
19:16	R/W	0	cts_hdmi_tx_pnx_Clk div N, if you want div8, set to 7
14:13	R/W	0	cts_hdmi_tx_fe_clk: 0:enc0_hdmi_tx_fe_clk 1:0 2:enc2_hdmi_tx_fe_clk 30
12	R/W	0	cts_hdmi_tx_fe_Clk_en
11:8	R/W	0	cts_hdmi_tx_fe_Clk div N, if you want div8, set to 7
6:5	R/W	0	cts_hdmi_tx_pixel_clk: 0:enc0_hdmi_tx_pixel_clk 1:0 2:enc2_hdmi_tx_pixel_clk 30

Bit(s)	R/W	Default	Description
4	R/W	0	cts_hdmi_tx_pixel_Clk_en
3:0	R/W	0	cts_hdmi_tx_pixel_Clk div N, if you want div8, set to 7

**Table 7-186 CLKCTRL\_ENC0/2\_HDMI\_CLK\_CTRL**

Bit(s)	R/W	Default	Description
27:24	R/W	0	hdmi_tx_pnx_Clk sel
23:20	R/W	0	hdmi_tx_fe_Clk sel
19:16	R/W	0	hdmi_tx_pixel_Clk sel
15:0	RW	0	unused

**Table 7-187 CLKCTRL\_HTX\_CLK\_CTRL**

Bit(s)	R/W	Default	Description
26:25	R/W	0	cts_hdmitx_200m_clk: 0:oscin 1:fclk_div4 2:fclk_div3 3fclk_div5
24	R/W	0	cts_hdmitx_200m_Clk_en
22:16	R/W	0	cts_hdmitx_200m_Clk div N, if you want div8, set to 7
10:9	R/W	0	cts_hdmitx_prifl_clk: 0:oscin 1:fclk_div4 2:fclk_div3 3fclk_div5
8	R/W	0	cts_hdmitx_prif_Clk_en
6:0	R/W	0	cts_hdmitx_prif_Clk div N, if you want div8, set to 7

**Table 7-188 CLKCTRL\_HTX\_CLK\_CTRL1**

Bit(s)	R/W	Default	Description
10:9	R/W	0	cts_hdmitx_aud_clk: 0:oscin 1:fclk_div4 2:fclk_div3 3fclk_div5
8	R/W	0	cts_hdmitx_aud_Clk_en
6:0	R/W	0	cts_hdmitx_aud_Clk div N, if you want div8, set to 7

**Table 7-189 CLKCTRL\_HRX\_CLK\_CTRL**

Bit(s)	R/W	Default	Description
26:25	R/W	0	cts_hdmirx_2m_clk: 0:oscin 1:flck_div4 2:flck_div3 3flck_div5
24	R/W	0	cts_hdmirx_2m_Clk_en
22:16	R/W	0	cts_hdmirx_2m_Clk div N, if you want div8, set to 7
10:9	R/W	0	cts_hdmirx_5m_clk: 0:oscin 1:flck_div4 2:flck_div3 3flck_div5
8	R/W	0	cts_hdmirx_5m_Clk_en
6:0	R/W	0	cts_hdmirx_5m_Clk div N, if you want div8, set to 7

**Table 7-190 CLKCTRL\_HRX\_CLK\_CTRL1**

Bit(s)	R/W	Default	Description
26:25	R/W	0	cts_hdmirx_hdcp2x_eckl: 0:oscin 1:flck_div4 2:flck_div3 3flck_div5
24	R/W	0	cts_hdmirx_hdcp2x_eclk_en
22:16	R/W	0	cts_hdmirx_hdcp2x_eclk div N, if you want div8, set to 7
10:9	R/W	0	cts_hdmirx_cfg_clk: 0:oscin 1:flck_div4 2:flck_div3 3flck_div5
8	R/W	0	cts_hdmirx_cfg_Clk_en
6:0	R/W	0	cts_hdmirx_cfg_Clk div N, if you want div8, set to 7

**Table 7-191 CLKCTRL\_HRX\_CLK\_CTRL2**

Bit(s)	R/W	Default	Description
26:25	R/W	0	cts_hdmirx_acr_ref_clk: 0:oscin 1:flck_div4 2:flck_div3 3flck_div5
24	R/W	0	cts_hdmirx_acr_ref_Clk_en
22:16	R/W	0	cts_hdmirx_acr_ref_Clk div N, if you want div8, set to 7

Bit(s)	R/W	Default	Description
10:9	R/W	0	cts_hdmirx_aud_pll_clk: 0:oscin 1:fclk_div4 2:fclk_div3 3fclk_div5
8	R/W	0	cts_hdmirx_aud_pll_Clk_en
6:0	R/W	0	cts_hdmirx_aud_pll_Clk div N, if you want div8, set to 7

Table 7-192 CLKCTRL\_HRX\_CLK\_CTRL3

Bit(s)	R/W	Default	Description
10:9	R/W	0	cts_hdmirx_meter_clk: 0:oscin 1:fclk_div4 2:fclk_div3 3fclk_div5
8	R/W	0	cts_hdmirx_meter_Clk_en
6:0	R/W	0	cts_hdmirx_meter_Clk div N, if you want div8, set to 7

Table 7-193 CLKCTRL\_HDMI\_VID\_PLL\_CLK\_DIV

Bit(s)	R/W	Default	Description
25	R/W	0	Reg_vid_pll2_clk_sel_hdmi
24	R/W	0	Reg_vid_pll0_clk_sel_hdmi

Table 7-194 CLKCTRL\_MIPI\_ISP\_CLK\_CTRL

Bit(s)	R/W	Default	Description
11:9	R/W	0	cts_pwm_i/g/e/c/a_clk clk_sel: 0: osin 1: fclk_div4 2: fclk_div3 3: fclk_fiv5 4: fclk_div7 5: mp2_clk_out 6: mp3_clk_out 7: gp1_pll_clk
8	R/W	0	Cts_mipi_isp_clk clk_en
6:0	R/W	0	cts_nipi_isp_clk div N, if you want div8, set to 7

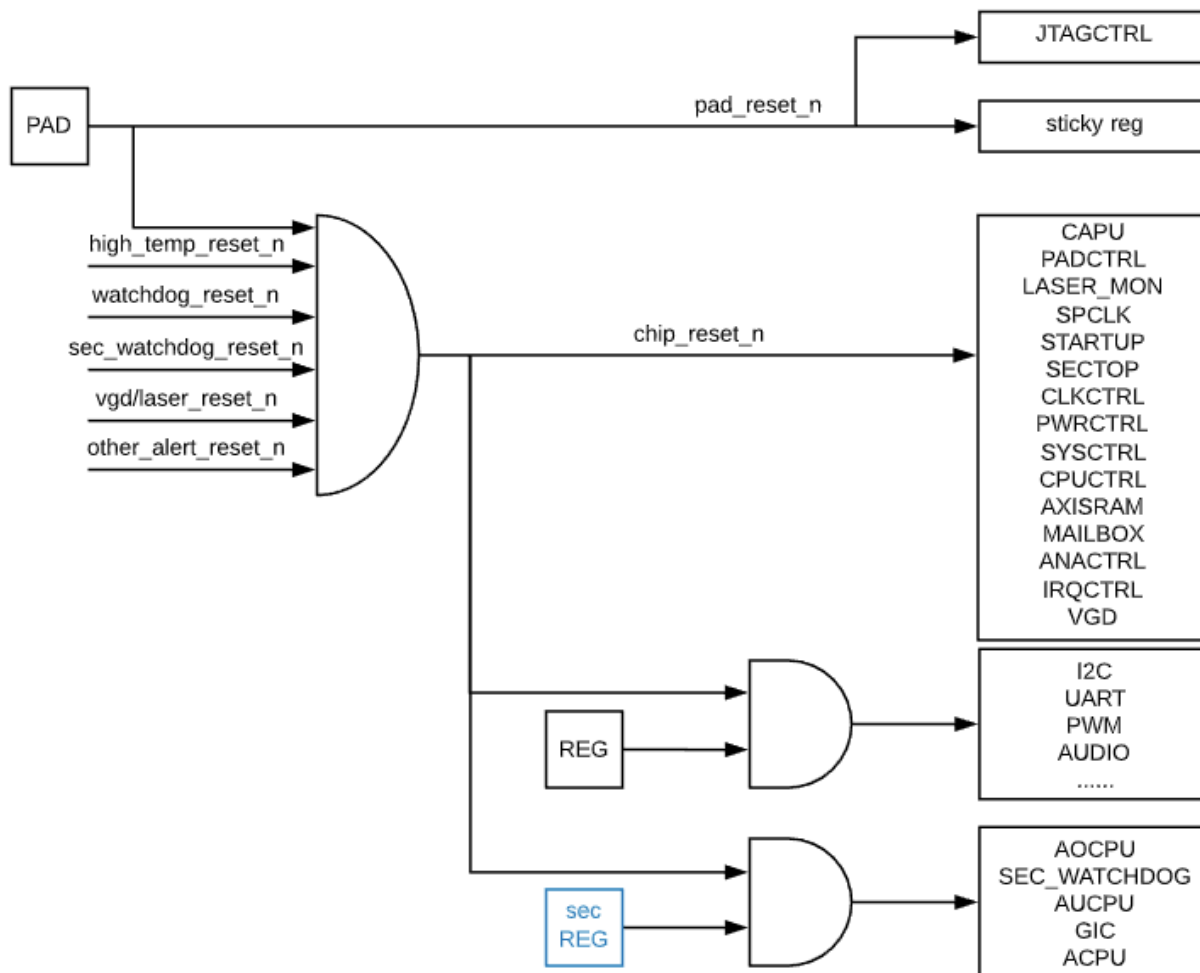
## 7.7 Reset

### 7.7.1 Overview

#### Reset

This module is control module level reset which is shown in the following diagram as “REG/SEC\_REG”, and the control of watchdog is also shown.

Figure 7-25 Reset Design



RESETn\_module is low active.

$$RESETn\_module = (\sim RESET\_reg \& RESET\_LEVEL) | RESET\_MASK;$$

If set RESET\_reg = 1, will reset the module.

RESET\_reg is an auto-recover register which will recover to 0 after some cycles.

If set RESET\_LEVEL = 0, will keep RESETn\_module as low, mean the module will keep at reset status.

If set RESET\_MASK = 1, will keep RESETn\_module as high, mean the module will never be reset.

#### Watchdog

There are two watchdogs, one can be used in non-secure mode, and the other can only be used in secure mode.

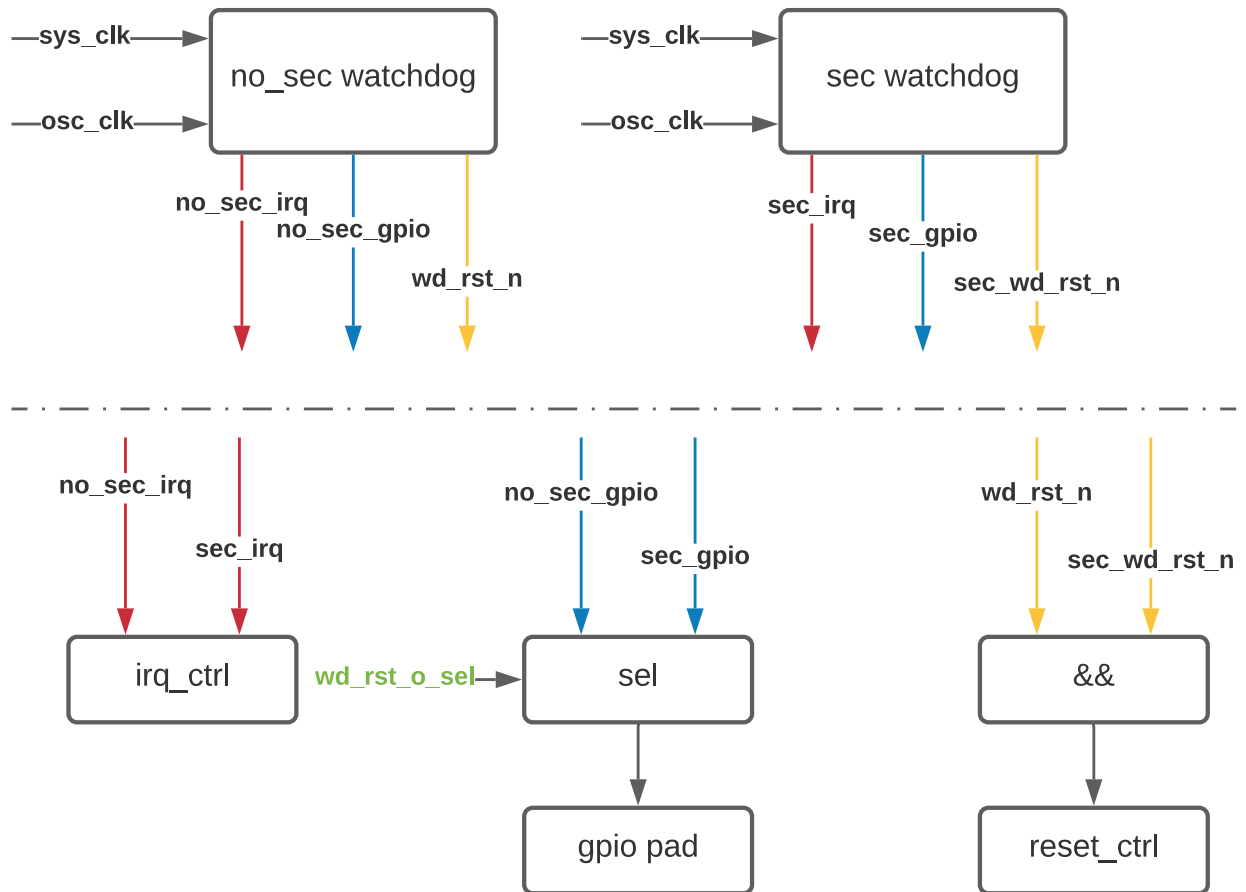
The time tick used by watchdog is based on sys\_clk or osc\_clk.



There are 3 features that can be enabled separately after time out:

- Generate interrupt (can be enabled independently)
- Reset the chip (can be enabled independently)
- Generate gpio output (pulse or level, controlled by register)

**Figure 7-26 Watchdog Design**



### 7.7.2 Register Description

#### Register Address

For below registers, the base address is 0xfe002000.

Each register final address = BASE + address \* 4.

The following lists describe the mapping between each RESET\_CTRL register and its address.

- RESETCTRL\_RESET0 0xfe002000
- RESETCTRL\_RESET1 0xfe002004
- RESETCTRL\_RESET2 0xfe002008
- RESETCTRL\_RESET3 0xfe00200c
- RESETCTRL\_RESET4 0xfe002010
- RESETCTRL\_RESET5 0xfe002014
- RESETCTRL\_RESET6 0xfe002018

- RESETCTRL\_RESET0\_LEVEL 0xfe002040
- RESETCTRL\_RESET1\_LEVEL 0xfe002044
- RESETCTRL\_RESET2\_LEVEL 0xfe002048
- RESETCTRL\_RESET3\_LEVEL 0xfe00204c
- RESETCTRL\_RESET4\_LEVEL 0xfe002050
- RESETCTRL\_RESET5\_LEVEL 0xfe002054
- RESETCTRL\_RESET6\_LEVEL 0xfe002058
- RESETCTRL\_RESET0\_MASK 0xfe002080
- RESETCTRL\_RESET1\_MASK 0xfe002084
- RESETCTRL\_RESET2\_MASK 0xfe002088
- RESETCTRL\_RESET3\_MASK 0xfe00208c
- RESETCTRL\_RESET4\_MASK 0xfe002090
- RESETCTRL\_RESET5\_MASK 0xfe002094
- RESETCTRL\_RESET6\_MASK 0xfe002098
- RESETCTRL\_RESET\_HOLD\_CYC 0xfe0020c0
- RESETCTRL\_WATCHDOG\_CTRL0 0xfe002100
- RESETCTRL\_WATCHDOG\_CTRL1 0xfe002104
- RESETCTRL\_WATCHDOG\_CNT 0xfe002108
- RESETCTRL\_WATCHDOG\_CLR 0xfe00210c
- RESETCTRL\_SEC\_WATCHDOG\_CTRL0 0xfe002110
- RESETCTRL\_SEC\_WATCHDOG\_CTRL1 0xfe002114
- RESETCTRL\_SEC\_WATCHDOG\_CNT 0xfe002118
- RESETCTRL\_SEC\_WATCHDOG\_CLR 0xfe00211c
- RESETCTRL\_SEC\_RESET0 0xfe002140
- RESETCTRL\_SEC\_RESET1 0xfe002144
- RESETCTRL\_SEC\_RESET2 0xfe002148
- RESETCTRL\_SEC\_RESET0\_LEVEL 0xfe002180
- RESETCTRL\_SEC\_RESET1\_LEVEL 0xfe002184
- RESETCTRL\_SEC\_RESET2\_LEVEL 0xfe002188
- RESETCTRL\_SEC\_RESET0\_MASK 0xfe0021c0
- RESETCTRL\_SEC\_RESET1\_MASK 0xfe0021c4
- RESETCTRL\_SEC\_RESET2\_MASK 0xfe0021c8
- RESETCTRL\_RESET0\_LOCK 0xfe002200
- RESETCTRL\_RESET1\_LOCK 0xfe002204
- RESETCTRL\_RESET2\_LOCK 0xfe002208
- RESETCTRL\_RESET3\_LOCK 0xfe00220c
- RESETCTRL\_RESET4\_LOCK 0xfe002210
- RESETCTRL\_RESET5\_LOCK 0xfe002214
- RESETCTRL\_RESET6\_LOCK 0xfe002218
- RESETCTRL\_SEC\_RESET0\_LOCK 0xfe002220
- RESETCTRL\_SEC\_RESET1\_LOCK 0xfe002224
- RESETCTRL\_SEC\_RESET2\_LOCK 0xfe002228
- RESETCTRL\_RESET0\_PROT 0xfe002240
- RESETCTRL\_RESET1\_PROT 0xfe002244

- RESETCTRL\_RESET2\_PROT 0xfe002248
- RESETCTRL\_RESET3\_PROT 0xfe00224c
- RESETCTRL\_RESET4\_PROT 0xfe002250
- RESETCTRL\_RESET5\_PROT 0xfe002254
- RESETCTRL\_RESET6\_PROT 0xfe002258
- RESETCTRL\_SEC\_RESET0\_PROT 0xfe002260
- RESETCTRL\_SEC\_RESET1\_PROT 0xfe002264
- RESETCTRL\_SEC\_RESET2\_PROT 0xfe002268

## Register Description

**Table 7-195 RESETCTRL\_RESET0~6**

RESET\_reg, write each bit to 1 can reset related module, the reset will auto-cover to 0 by HW.

\*mean it can reset by power control signals because it need keep reset when power off/on.

bit	reg_re-set0_n	reg_re-set1_n	reg_re-set2_n	reg_re-set3_n	reg_re-set4_n	reg_re-set5_n	reg_re-set6_n	reg_sec_re-set0_n	reg_sec_re-set1_n	reg_sec_re-set2_n
31	venc*	mipi_dsi_b_host*		brg_cci*	ts_hevc	brg_nic4_all	brg_am2a_xi2			
30	viu*	mipi_dsi_a_host*		brg_adb_a53	ts_vpu	brg_nic4_main	brg_am2a_xi1			
29	hdmit-x*	mipi_dsi0_phy		brg_adb_a73	ts_nna	brg_nic4_clk81	brg_am2a_xi0			
28	rdma*	mipi_dsi1_phy		brg_adb_mali0	ts_gpu	brg_nic4_vapb		vpu_sec		
27	venc1*	edp0_pipeline	watch-dog	brg_adb_mali1	i2c_m_ao_b	brg_nic4_dspa	brg_am_pipe_eth	dspb_sec		
26	venc2*	edp1_pipeline	afifo	mipi_isp*	sd_emmc_C*	brg_nic4_dspb	brg_am_pipe_nand	dspa_sec		
25	vdac*		cec		sd_emmc_B*			isp_sec		
24	venc0*	ddr1	aco-dec		sd_emmc_A*			dew-arp_sec		
23	vid_lock*	pci-e1_apb			i2c_m_ao_a			gdc_sec		

bit	reg_re-set0_n	reg_re-set1_n	reg_re-set2_n	reg_re-set3_n	reg_re-set4_n	reg_re-set5_n	reg_re-set6_n	reg_sec_re-set0_n	reg_sec_re-set1_n	reg_sec_re-set2_n
22	hdmit-xphy	pci-e1_phy			i2c_m_f	brg_nic5_vpu		ge2d_sec		
21	ge2d*	dsi_lvds_edp_top			i2c_m_e	brg_nic3_all*		mali_sec		
20	vdi6*	combo_dphy_chan1			i2c_m_d	brg_nic3_main*		wave*		
19	vid_pll_div	combo_dphy_chan0	bt		i2c_m_c	brg_nic3_amlg-dc*		hevc*		
18	vcbus*	edp1_ctrl*	sar_adc		i2c_m_b	brg_nic3_gdc*		vdec*		
17	brg_vcbus_dec	edp0_ctrl*	spifc		i2c_m_a	brg_nic3_ge2d*		hco-dec*		
16	hdmit-x_cap-b3*	eth*	msr_clk		i2c_s_a	brg_nic3_hco-dec*		gic		
15		anakin*			uart_f	brg_nic3_hevc-b*		sys_cpu_3*		a73_sys_cpu_3*
14	dew-arp*	pcie_apb			uart_e	brg_nic3_hevcf*		sys_cpu_2*		a73_sys_cpu_2*
13	hdmir-x_apb*	pcie_phy			uart_d	brg_nic3_vdec*	brg_fris-p0_pipel	sys_cpu_1*		a73_sys_cpu_1*
12	hdmir-x*	pcie_A*			uart_c	brg_nic3_wave*	brg_fris-p1_pipel	auc-pu_dma	sys_cpu_0*	a73_sys_cpu_0*
11	hdm-i20_aes	i_dsipa	rsa		uart_b	brg_nic2_all*	brg_fris-p2_pipel	auc-pu_sys	sys_cpu_core_3*	a73_sys_cpu_core_3*
10	gdc*	i_dspb*	spicc_1*		uart_a	brg_nic2_hdmi*	brg_fris-p3_pipel	auc-pu_por	sys_cpu_core_2*	a73_sys_cpu_core_2*

bit	reg_re-set0_n	reg_re-set1_n	reg_re-set2_n	reg_re-set3_n	reg_re-set4_n	reg_re-set5_n	reg_re-set6_n	reg_sec_re-set0_n	reg_sec_re-set1_n	reg_sec_re-set2_n
9	u2phy21	i_debuga*	spicc_0*			brg_nic2_main*	brg_nnato-noc_pipel	aocpu_all	sys_cpu_core_1*	a73_sys_cpu_core_1*
8	u2phy20	i_debugb*	smart_card			brg_nic2_sys*	brg_nic2-to-noc_pipel	sec_watchdog	sys_cpu_core_0*	a73_sys_cpu_core_0*
7	u3dr-d_pipe0*	combo_dphy_chan2	spicc_5*				brg_nic3-to-noc_pipel		sys_pll_div*	a73_sys_pll_div*
6	u3dr-d*	dos	spicc_4*		pwm_ef		brg_nic4-to-noc_pipel		Psys_cpu_capb3*	a73_Psys_cpu_capb3*
5	u2dr-d*	dos_capb3	spicc_3*		pwm_cd		brg_vputo-noc_pipel		rom_boot	
4	usb*	ddr	spicc_2*	brg_nic5_gpv	pwm_ab		brg_section-ic4_pipel		sys_cpu_axi*	a73_sys_cpu_axi*
3		ddr_apb	ts_a53	brg_nic4_gpv	pwm_ao_gh	brg_noc_all*	brg_hdmir-xto-nic2_pipel		sys_cpu_l2*	a73_sys_cpu_l2*
2		mali*	ts_a73	brg_nic3_gpv	pwm_ao_ef	brg_noc_main*	brg_nic2-ton-ic4_pipel	aocpu_por	sys_cpu_p*	a73_sys_cpu_p*
1		mali_capb3*	ir_ctrl	brg_nic2_gpv	pwm_ao_cd	brg_noc_ddr0*	brg_hevc-f_dmc_pipel	aocpu_core	sys_cpu_mbist*	a73_sys_cpu_mbist*
0		audio	device_mmc_arb	brg_nic1_gpv	pwm_ao_ab	brg_noc_ddr1*	brg_vdec_pipel	aocpu		

**RESETCTRL\_SEC\_RESET0**

Write each bit to 1 can reset related module, the reset will auto-recover to 0 by HW.

**RESETCTRL\_RESET0~6\_LEVEL**

Write each bit to 0 will hold reset as 1.

The list refer to resetctrl\_reset0~6

**RESETCTRL\_SEC\_RESET0\_LEVEL**

Write each bit to 0 will hold reset as 1.

Refer to resetctrl\_sec\_reset0

**RESETCTRL\_RESET0~6\_MASK**

Write each bit to 1 will hold reset as 0.

The list refer to resetctrl\_reset0~6

**RESETCTRL\_SEC\_RESET0\_MASK**

Write each bit to 1 will hold reset as 0.

The list refer to resetctrl\_sec\_reset0

**RESETCTRL\_RESET0~6\_LOCK**

Each bit can write 1 only

Write each bit to 1 will lock reg reset/level/mask.

The list refer to resetctrl\_reset0~6

**RESETCTRL\_SEC\_RESET0\_LOCK**

Each bit can write 1 only

Write each bit to 1 will lock reg reset/level/mask.

The list refer to resetctrl\_sec\_reset0

**RESETCTRL\_RESET0~6\_PROT**

Write each bit to 1 will protect(can't write) reg reset/level/mask.

The list refer to resetctrl\_reset0~6

**RESETCTRL\_SEC\_RESET0\_PROT**

Write each bit to 1 will protect(can't write) reg reset/level/mask.

The list refer to resetctrl\_sec\_reset0

**Table 7-196 RESETCTRL\_RESET\_HOLD\_CYC**

Bit(s)	R/W	Default	Description
4:0	R/W	3	the cycle number of auto-recover

**Table 7-197 RESETCTRL\_WATCHDOG\_CTRL0**

Bit(s)	R/W	Default	Description
31	R	0	watchdog status(RESET value)
27	R/W	0	the RESET OUT value when watchdog_en = 0; high active;
25	R/W	0	Clk_div_en

Bit(s)	R/W	Default	Description
24	R/W	1	Clk_en
23	R/W	0	Interrupt_en
22	R/W	0	reset_n_en
19	R/W	0	Clk_sel 0: cts_oscin_clk; 1: cts_sys_clk;
18	R/W	0	Watchdog_en
17-0	R/W	23999	Clk_div_tcnt

Table 7-198 RESETCTRL\_WATCHDOG\_CTRL1

Bit(s)	R/W	Default	Description
17	R/W	0	Gpio_pulse
16	R/W	0	Gpio_polarity
15-0	R/W	0	Gpio_pulse_tcnt

Table 7-199 RESETCTRL\_WATCHDOG\_CNT

Bit(s)	R/W	Default	Description
31-16	R	0	Watchdog count read
15-0	R/W	0	Watchdog_count set

Table 7-200 RESETCTRL\_WATCHDOG\_CLR

Bit(s)	R/W	Default	Description
31:0	W	0	write any value will clear watchdog status

**RESETCTRL\_SEC\_WATCHDOG\_CTRL0**

The description/function is same as RESETCTRL\_WATCHDOG\_CTRL0, access permission is different, it can be only accessed in secure mode.

**RESETCTRL\_SEC\_WATCHDOG\_CTRL1**

The description/function is same as RESETCTRL\_WATCHDOG\_CTRL1, access permission is different, it can be only accessed in secure mode

**RESETCTRL\_SEC\_WATCHDOG\_CNT**

The description/function is same as RESETCTRL\_WATCHDOG\_CNT, access permission is different, it can be only accessed in secure mode

**RESETCTRL\_SEC\_WATCHDOG\_CLR**

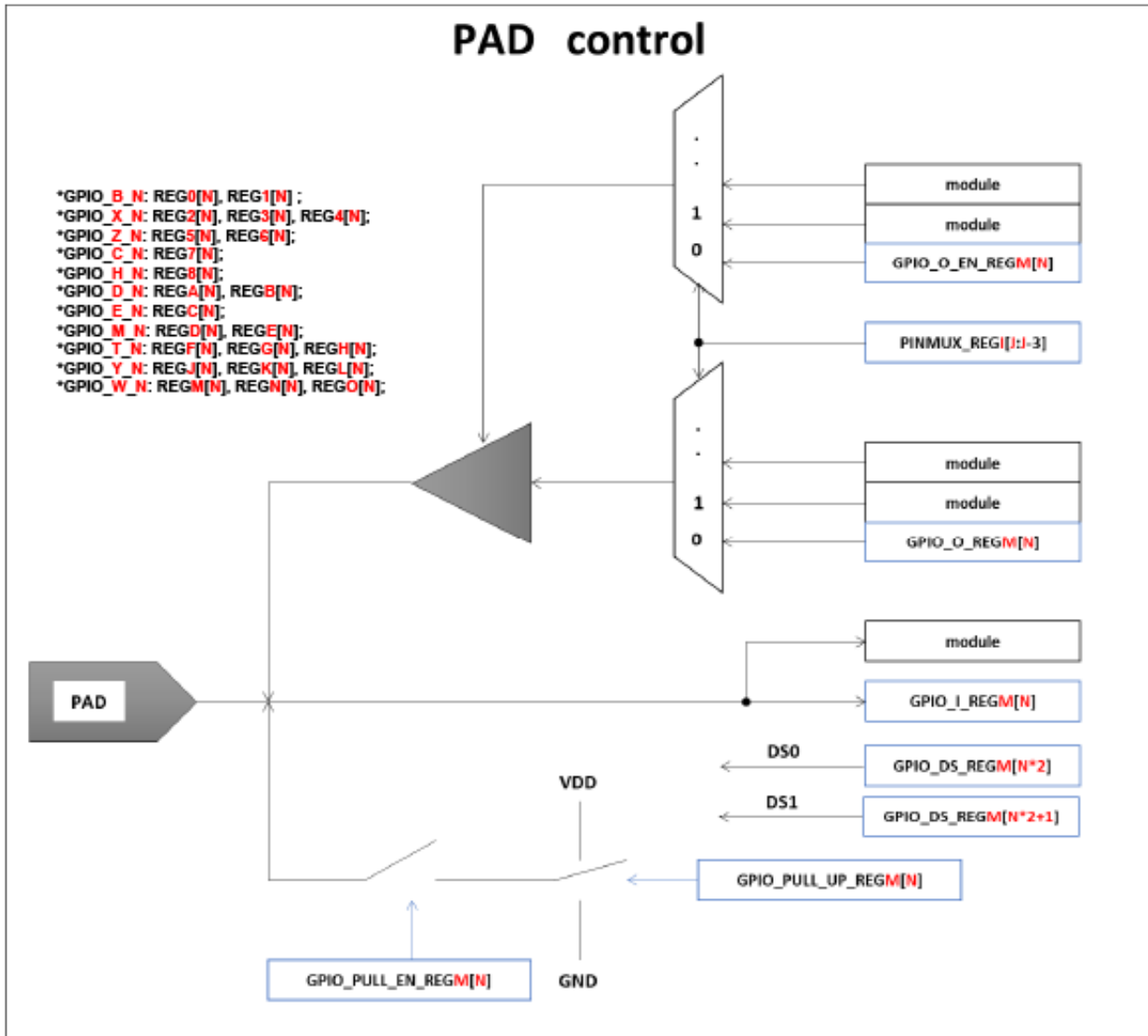
The description/function is same as RESETCTRL\_WATCHDOG\_CLR, access permission is different, it can be only accessed in secure mode

## 7.8 GPIO

### 7.8.1 Overview

The SOC has a number of multi-function digital I/O pads that can be multiplexed to a number of internal resources (e.g. PWM generators, SDIO controllers). When a digital I/O is not being used for any specific purpose, it is converted to a general purpose GPIO pin. A GPIO pin can be statically set to high/low logical levels. The structure of a GPIO is given below.

Figure 7-27 GPIO Structure



T08FC01

### 7.8.2 GPIO Multiplex Function

The GPIO multiplex functions are shown in the sections below, where the RegNN[MM] corresponds to CBUS registers defined in the following table.

The base address is 0xfe004000 for the PAD control registers.

The final address is calculated as: final address = base address + offset\*4.



Table 7-201 GPIOW\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2	Func3
GPIOW_0	PADCTRL_PIN_MUX_REGM	3:0	HDMIRX_A_HPDP		
GPIOW_1		7:4	HDMIRX_A_5VDET		
GPIOW_2		11:8	HDMIRX_A_SDA	UART_AO_A_TX	HDMITX_SDA
GPIOW_3		15:12	HDMIRX_A_SCL	UART_AO_A_RX	HDMITX_SCL
GPIOW_4		19:16	HDMIRX_C_HPDP		
GPIOW_5		23:20	HDMIRX_C_5VDET		
GPIOW_6		27:24	HDMIRX_C_SDA	UART_AO_A_TX	
GPIOW_7		31:28	HDMIRX_C_SCL	UART_AO_A_RX	
GPIOW_8	PADCTRL_PIN_MUX_REGN	3:0	HDMIRX_B_HPDP		
GPIOW_9		7:4	HDMIRX_B_5VDET		
GPIOW_10		11:8	HDMIRX_B_SDA	UART_AO_A_TX	
GPIOW_11		15:12	HDMIRX_B_SCL	UART_AO_A_RX	
GPIOW_12		19:16	CEC_A		
GPIOW_13		23:20	HDMITX_SDA		
GPIOW_14		27:24	HDMITX_SCL		
GPIOW_15		31:28	HDMITX_HPDP_IN		
GPIOW_16	PADCTRL_PIN_MUX_REGO	3:0	CEC_B		

Table 7-202 GPIOD\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2	Func3	Func4
GPIOD_0	PADCTRL_PIN_MUX_REGA	3:0	UART_AO_A_TX			
GPIOD_1		7:4	UART_AO_A_RX			
GPIOD_2		11:8	I2CM_AO_A_SCL	I2CS_AO_SCL	UART_AO_B_TX	
GPIOD_3		15:12	I2CM_AO_A_SDA	I2CS_AO_SDA	UART_AO_B_RX	
GPIOD_4		19:16	IR_OUT	RTC_CLK_IN	UART_AO_B_CTS	

Pin Name	Control Register	Control Bit	Func1	Func2	Func3	Func4
GPIOD_5		23:20	IR_IN	PWM_AO_H		
GPIOD_6		27:24	JTAG_A_CLK	PWM_AO_C	PWM_AO_C_HIZ	IR_OUT
GPIOD_7		31:28	JTAG_A_TMS	PWM_AO_G	PWM_AO_G_HIZ	
GPIOD_8	PADCTRL_PIN_MUX_REGB	3:0	JTAG_A_TDI	SPDIF_OUT		
GPIOD_9		7:4	JTAG_A_TDO	SPDIF_IN		
GPIOD_10		11:8	GEN_CLK_OUT	PWM_AO_H	UART_AO_B_RTS	
GPIOD_11		15:12	PWM_AO_G			
GPIOD_12		19:16	WD_RSTO			

Table 7-203 GPIOE\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2	Func3
GPIOE_0	PADCTRL_PIN_MUX_REGC	3:0	PWM_AO_A	I2CM_AO_A_SCL	
GPIOE_1		7:4	PWM_AO_B	I2CM_AO_A_SDA	
GPIOE_2		11:8	PWM_AO_C	CLK25M	
GPIOE_3		15:12	PWM_AO_D	I2CM_AO_B_SCL	
GPIOE_4		19:16	PWM_AO_E	I2CM_AO_B_SDA	CLK12M_24M
GPIOE_5		23:20	PWM_AO_F	RTC_CLK_OUT	
GPIOE_6		27:24	PWM_AO_G		

Table 7-204 GPIOB\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2
GPIOB_0	PADCTRL_PIN_MUX_REG0	3:0	EMMC_D0	
GPIOB_1		7:4	EMMC_D1	
GPIOB_2		11:8	EMMC_D2	
GPIOB_3		15:12	EMMC_D3	SPIF_HOLD
GPIOB_4		19:16	EMMC_D4	SPIF_MO
GPIOB_5		23:20	EMMC_D5	SPIF_MI
GPIOB_6		27:24	EMMC_D6	SPIF_CLK
GPIOB_7	31:28	EMMC_D7	SPIF_WP	
GPIOB_8	PADCTRL_PIN_MUX_REG1	3:0	EMMC_CLK	

Pin Name	Control Register	Control Bit	Func1	Func2
GPIOB_9		7:4		
GPIOB_10		11:8	EMMC_CMD	
GPIOB_11		15:12	EMMC_DS	
GPIOB_12		19:16		SPIF_CS

Table 7-205 GPIOC\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2	Func3
GPIOC_0	PADCTRL_PIN_MUX_REG7	3:0	SDCARD_D0	JTAG_B_TDO	SPI_B_MOSI
GPIOC_1		7:4	SDCARD_D1	JTAG_B_TDI	SPI_B_MISO
GPIOC_2		11:8	SDCARD_D2	UART_AO_A_RX	SPI_B_SCLK
GPIOC_3		15:12	SDCARD_D3	UART_AO_A_TX	SPI_B_SS0
GPIOC_4		19:16	SDCARD_CLK	JTAG_B_CLK	
GPIOC_5		23:20	SDCARD_CMD	JTAG_B_TMS	
GPIOC_6		27:24	GEN_CLK_OUT		

Table 7-206 GPIOZ\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2	Func3	Func4
GPIOZ_0	PADCTRL_PIN_MUX_REG5	3:0	ETH_MDIO	ISO7816_CLK		SPI_E_MOSI
GPIOZ_1		7:4	ETH_MDC	ISO7816_DATA		SPI_E_MISO
GPIOZ_2		11:8	ETH_RGMII_RX_CLK	TSIN_B_VALID		SPI_E_SCLK
GPIOZ_3		15:12	ETH_RX_DV	TSIN_B_SOP		SPI_E_SS0
GPIOZ_4		19:16	ETH_RXD0	TSIN_B_DIN0		SPI_F_MOSI
GPIOZ_5		23:20	ETH_RXD1	TSIN_B_CLK		SPI_F_MISO
GPIOZ_6		27:24	ETH_RXD2_RGMII	TSIN_B_FAIL	TSIN_C_VALID	SPI_F_SCLK
GPIOZ_7	PADCTRL_PIN_MUX_REG6	31:28	ETH_RXD3_RGMII	TSIN_B_DIN1	TSIN_C_SOP	SPI_F_SS0
GPIOZ_8		3:0	ETH_RGMII_TX_CLK	TSIN_B_DIN2	TSIN_C_DIN0	
GPIOZ_9		7:4	ETH_TXEN	TSIN_B_DIN3	TSIN_C_CLK	
GPIOZ_10		11:8	ETH_TXD0	TSIN_B_DIN4	TSIN_D_VALID	
GPIOZ_11		15:12	ETH_TXD1	TSIN_B_DIN5	TSIN_D_SOP	

Pin Name	Control Register	Control Bit	Func1	Func2	Func3	Func4
GPIOZ_12		19:16	ETH_TXD2_RGMII	TSIN_B_DIN6	TSIN_D_DIN0	
GPIOZ_13		23:20	ETH_TXD3_RGMII	TSIN_B_DIN7	TSIN_D_CLK	

Table 7-207 GPIOH\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2	Func3
GPIOH_0	PADCTRL_PIN_MUX_REG8	3:0	MIC_MUTE_KEY		
GPIOH_1		7:4	Mic_Mute_LED	PWM_VS	
GPIOH_2		11:8	I2CM_D_SCL	UART_F_TX	PCIECK_REQN
GPIOH_3		15:12	I2CM_D_SDA	UART_F_RX	
GPIOH_4		19:16	I2CM_E_SCL	UART_F_CTS	
GPIOH_5		23:20	I2CM_E_SDA	UART_F_RTS	
GPIOH_6		27:24	ETH_LINK_LED	I2CM_A_SDA	
GPIOH_7		31:28	ETH_ACT_LED	I2CM_A_SCL	

Table 7-208 GPIOM\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2
GPIOM_0	PADCTRL_PIN_MUX_REGD	3:0	TDM_D12	PDM_DIN1
GPIOM_1		7:4	TDM_D13	PDM_DIN2
GPIOM_2		11:8	TDM_D14	PDM_DIN3
GPIOM_3		15:12	TDM_D15	PDM_DCCLK
GPIOM_4		19:16	TDM_SCLK3	PDM_DIN0
GPIOM_5		23:20	TDM_FS3	PDM_DIN1
GPIOM_6		27:24	I2CM_D_SCL	
GPIOM_7		31:28	I2CM_D_SDA	
GPIOM_8	PADCTRL_PIN_MUX_REGE	3:0	SPI_B_MOSI	UART_D_TX
GPIOM_9		7:4	SPI_B_MISO	UART_D_RX
GPIOM_10		11:8	SPI_B_SCLK	UART_D_CTS
GPIOM_11		15:12	SPI_B_SS0	UART_D_RTS
GPIOM_12		19:16	SPI_B_SS1	I2CM_C_SCL
GPIOM_13		23:20	SPI_B_SS2	I2CM_C_SDA

Table 7-209 GPIOX\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2
GPIOX_0	PADCTRL_PIN_MUX_REG2	3:0	SDIO_D0	
GPIOX_1		7:4	SDIO_D1	
GPIOX_2		11:8	SDIO_D2	
GPIOX_3		15:12	SDIO_D3	
GPIOX_4		19:16	SDIO_CLK	
GPIOX_5		23:20	SDIO_CMD	
GPIOX_6		27:24	PWM_B	
GPIOX_7		31:28	PWM_C	
GPIOX_8	PADCTRL_PIN_MUX_REG3	3:0	TDM_D0	
GPIOX_9		7:4	TDM_D1	
GPIOX_10		11:8	TDM_FS0	
GPIOX_11		15:12	TDM_SCLK0	
GPIOX_12		19:16	UART_C_TX	
GPIOX_13		23:20	UART_C_RX	
GPIOX_14		27:24	UART_C_CTS	CLK12M_24M
GPIOX_15		31:28	UART_C_RTS	
GPIOX_16	PADCTRL_PIN_MUX_REG4	3:0	PWM_A	
GPIOX_17		7:4	I2CM_C_SDA	
GPIOX_18		11:8	I2CM_C_SCL	
GPIOX_19		15:12	PWM_D	

Table 7-210 GPIOT\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2	Func3
GPIOT_0	PADCTRL_PIN_MUX_REGF	3:0	MCLK_1		
GPIOT_1		7:4	TDM_SCLK1		
GPIOT_2		11:8	TDM_FS1		
GPIOT_3		15:12	TDM_D2	SPDIF_IN	
GPIOT_4		19:16	TDM_D3	SPDIF_OUT	
GPIOT_5		23:20	TDM_D4	ISO7816_CLK	
GPIOT_6		27:24	TDM_D5	ISO7816_DATA	SPI_D_MOSI
GPIOT_7		31:28	TDM_D6	TSIN_A_SOP	SPI_D_MISO
GPIOT_8	PADCTRL_PIN_MUX_REGG	3:0	TDM_D7	TSIN_A_DIN0	SPI_D_SCLK
GPIOT_9		7:4	TDM_D8	TSIN_A_CLK	SPI_D_SS0
GPIOT_10		11:8	TDM_D9	TSIN_A_VALID	

Pin Name	Control Register	Control Bit	Func1	Func2	Func3
GPIOT_11		15:12	TDM_D10		
GPIOT_12		19:16	TDM_D11		
GPIOT_13		23:20	MCLK_2		
GPIOT_14		27:24	TDM_SCLK2		
GPIOT_15		31:28	TDM_FS2		
GPIOT_16	PADCTRL_PIN_MUX_REGH	3:0	I2CM_B_SCL		
GPIOT_17		7:4	I2CM_B_SDA		
GPIOT_18		11:8	SPI_A_MOSI		
GPIOT_19		15:12	SPI_A_MISO		
GPIOT_20		19:16	SPI_A_SCLK	I2CM_A_SCL	
GPIOT_21		23:20	SPI_A_SS0	I2CM_A_SDA	
GPIOT_22		27:24	SPI_A_SS1	I2CM_C_SCL	
GPIOT_23		31:28	SPI_A_SS2	I2CM_C_SDA	

Table 7-211 GPIOY\_x Multi-Function Pin

Pin Name	Control Register	Control Bit	Func1	Func2	Func3	Func4
GPIYOY_0	PADCTRL_PIN_MUX_REGJ	3:0	SPI_C_MOSI			
GPIYOY_1		7:4	SPI_C_MISO		PWM_E	
GPIYOY_2		11:8	SPI_C_SCLK			
GPIYOY_3		15:12	SPI_C_SS0			
GPIYOY_4		19:16	SPI_C_SS1	TSIN_C_SOP	HSYNC	
GPIYOY_5		23:20	SPI_C_SS2	TSIN_C_DIN0	VSYNC	
GPIYOY_6		27:24	UART_E_TX	TSIN_C_CLK		
GPIYOY_7		31:28	UART_E_RX	TSIN_C_VALID		
GPIYOY_8	PADCTRL_PIN_MUX_REGK	3:0	UART_E_CTS	TSIN_D_SOP	PWM_F	
GPIYOY_9		7:4	UART_E_RTS	TSIN_D_DIN0	3D_SYNC_OUT	
GPIYOY_10		11:8	UART_D_CTS	TSIN_D_CLK	VX1_A_HTPDN	eDP_A_HPD
GPIYOY_11		15:12	UART_D_RTS	TSIN_D_VALID	VX1_B_HTPDN	eDP_B_HPD
GPIYOY_12		19:16	UART_D_TX		VX1_A_LOCKN	
GPIYOY_13		23:20	UART_D_RX		VX1_B_LOCKN	
GPIYOY_14		27:24			PWM_VS	

Pin Name	Control Register	Control Bit	Func1	Func2	Func3	Func4
GPIOY_15		31:28	I2CM_E_SCL			
GPIOY_16	PADCTRL_PIN_MUX_REGL	3:0	I2CM_E_SDA			
GPIOY_17		7:4	I2CM_F_SCL			
GPIOY_18		11:8	I2CM_F_SDA	PCIECK_REQN		

## 7.8.3 Register Description

### 7.8.3.1 Pin MUX Registers

#### Register Address

The following lists describe the mapping between each register and its address.

- PADCTRL\_PIN\_MUX\_REG0 0xfe004000
- PADCTRL\_PIN\_MUX\_REG1 0xfe004004
- PADCTRL\_PIN\_MUX\_REG2 0xfe004008
- PADCTRL\_PIN\_MUX\_REG3 0xfe00400c
- PADCTRL\_PIN\_MUX\_REG4 0xfe004010
- PADCTRL\_PIN\_MUX\_REG5 0xfe004014
- PADCTRL\_PIN\_MUX\_REG6 0xfe004018
- PADCTRL\_PIN\_MUX\_REG7 0xfe00401c
- PADCTRL\_PIN\_MUX\_REG8 0xfe004020
- PADCTRL\_PIN\_MUX\_REG9 0xfe004024
- PADCTRL\_PIN\_MUX\_REGA 0xfe004028
- PADCTRL\_PIN\_MUX\_REGB 0xfe00402c
- PADCTRL\_PIN\_MUX\_REGC 0xfe004030
- PADCTRL\_PIN\_MUX\_REGD 0xfe004034
- PADCTRL\_PIN\_MUX\_REGE 0xfe004038
- PADCTRL\_PIN\_MUX\_REGF 0xfe00403c
- PADCTRL\_PIN\_MUX\_REGG 0xfe004040
- PADCTRL\_PIN\_MUX\_REGH 0xfe004044
- PADCTRL\_PIN\_MUX\_REGI 0xfe004048
- PADCTRL\_PIN\_MUX\_REGJ 0xfe00404c
- PADCTRL\_PIN\_MUX\_REGK 0xfe004050
- PADCTRL\_PIN\_MUX\_REGL 0xfe004054
- PADCTRL\_PIN\_MUX\_REGM 0xfe004058
- PADCTRL\_PIN\_MUX\_REGN 0xfe00405c
- PADCTRL\_PIN\_MUX\_REGO 0xfe004060

## Register Description

**Table 7-212 PADCTRL\_PIN\_MUX\_REG0**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioB_7_sel
27:24	R/W	0	gpioB_6_sel
23:20	R/W	0	gpioB_5_sel
19:16	R/W	0	gpioB_4_sel
15:12	R/W	0	gpioB_3_sel
11:8	R/W	0	gpioB_2_sel
7:4	R/W	0	gpioB_1_sel
3:0	R/W	0	gpioB_0_sel

**Table 7-213 PADCTRL\_PIN\_MUX\_REG1**

Bit(s)	R/W	Default	Description
19:16	R/W	0	gpioB_12_sel
15:12	R/W	0	gpioB_11_sel
11:8	R/W	0	gpioB_10_sel
7:4	R/W	0	gpioB_9_sel
3:0	R/W	0	gpioB_8_sel

**Table 7-214 PADCTRL\_PIN\_MUX\_REG2**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioX_7_sel
27:24	R/W	0	gpioX_6_sel
23:20	R/W	0	gpioX_5_sel
19:16	R/W	0	gpioX_4_sel
15:12	R/W	0	gpioX_3_sel
11:8	R/W	0	gpioX_2_sel
7:4	R/W	0	gpioX_1_sel
3:0	R/W	0	gpioX_0_sel

**Table 7-215 PADCTRL\_PIN\_MUX\_REG3**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioX_15_sel
27:24	R/W	0	gpioX_14_sel
23:20	R/W	0	gpioX_13_sel



Bit(s)	R/W	Default	Description
19:16	R/W	0	gpioX_12_sel
15:12	R/W	0	gpioX_11_sel
11:8	R/W	0	gpioX_10_sel
7:4	R/W	0	gpioX_9_sel
3:0	R/W	0	gpioX_8_sel

**Table 7-216 PADCTRL\_PIN\_MUX\_REG4**

Bit(s)	R/W	Default	Description
15:12	R/W	0	gpioX_19_sel
11:8	R/W	0	gpioX_18_sel
7:4	R/W	0	gpioX_17_sel
3:0	R/W	0	gpioX_16_sel

**Table 7-217 PADCTRL\_PIN\_MUX\_REG5**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioZ_7_sel
27:24	R/W	0	gpioZ_6_sel
23:20	R/W	0	gpioZ_5_sel
19:16	R/W	0	gpioZ_4_sel
15:12	R/W	0	gpioZ_3_sel
11:8	R/W	0	gpioZ_2_sel
7:4	R/W	0	gpioZ_1_sel
3:0	R/W	0	gpioZ_0_sel

**Table 7-218 PADCTRL\_PIN\_MUX\_REG6**

Bit(s)	R/W	Default	Description
23:20	R/W	0	gpioZ_13_sel
19:16	R/W	0	gpioZ_12_sel
15:12	R/W	0	gpioZ_11_sel
11:8	R/W	0	gpioZ_10_sel
7:4	R/W	0	gpioZ_9_sel
3:0	R/W	0	gpioZ_8_sel

**Table 7-219 PADCTRL\_PIN\_MUX\_REG7**

Bit(s)	R/W	Default	Description
27:24	R/W	0	gpioC_6_sel
23:20	R/W	2	gpioC_5_sel
19:16	R/W	2	gpioC_4_sel
15:12	R/W	0	gpioC_3_sel
11:8	R/W	0	gpioC_2_sel
7:4	R/W	2	gpioC_1_sel
3:0	R/W	2	gpioC_0_sel

**Table 7-220 PADCTRL\_PIN\_MUX\_REG8**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioH_7_sel
27:24	R/W	0	gpioH_6_sel
23:20	R/W	0	gpioH_5_sel
19:16	R/W	0	gpioH_4_sel
15:12	R/W	0	gpioH_3_sel
11:8	R/W	0	gpioH_2_sel
7:4	R/W	0	gpioH_1_sel
3:0	R/W	0	gpioH_0_sel

**Table 7-221 PADCTRL\_PIN\_MUX\_REGA**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioD_7_sel
27:24	R/W	0	gpioD_6_sel
23:20	R/W	0	gpioD_5_sel
19:16	R/W	0	gpioD_4_sel
15:12	R/W	0	gpioD_3_sel
11:8	R/W	0	gpioD_2_sel
7:4	R/W	0	gpioD_1_sel
3:0	R/W	0	gpioD_0_sel

**Table 7-222 PADCTRL\_PIN\_MUX\_REGB**

Bit(s)	R/W	Default	Description
19:16	R/W	0	gpioD_12_sel
15:12	R/W	0	gpioD_11_sel

Bit(s)	R/W	Default	Description
11:8	R/W	0	gpioD_10_sel
7:4	R/W	0	gpioD_9_sel
3:0	R/W	0	gpioD_8_sel

Table 7-223 PADCTRL\_PIN\_MUX\_REGC

Bit(s)	R/W	Default	Description
27:24	R/W	0	gpioE_6_sel
23:20	R/W	0	gpioE_5_sel
19:16	R/W	0	gpioE_4_sel
15:12	R/W	0	gpioE_3_sel
11:8	R/W	0	gpioE_2_sel
7:4	R/W	0	gpioE_1_sel
3:0	R/W	0	gpioE_0_sel

Table 7-224 PADCTRL\_PIN\_MUX\_REGD

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioM_7_sel
27:24	R/W	0	gpioM_6_sel
23:20	R/W	0	gpioM_5_sel
19:16	R/W	0	gpioM_4_sel
15:12	R/W	0	gpioM_3_sel
11:8	R/W	0	gpioM_2_sel
7:4	R/W	0	gpioM_1_sel
3:0	R/W	0	gpioM_0_sel

Table 7-225 PADCTRL\_PIN\_MUX\_REGE

Bit(s)	R/W	Default	Description
23:20	R/W	0	gpioM_13_sel
19:16	R/W	0	gpioM_12_sel
15:12	R/W	0	gpioM_11_sel
11:8	R/W	0	gpioM_10_sel
7:4	R/W	0	gpioM_9_sel
3:0	R/W	0	gpioM_8_sel

**Table 7-226 PADCTRL\_PIN\_MUX\_REGF**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioT_7_sel
27:24	R/W	0	gpioT_6_sel
23:20	R/W	0	gpioT_5_sel
19:16	R/W	0	gpioT_4_sel
15:12	R/W	0	gpioT_3_sel
11:8	R/W	0	gpioT_2_sel
7:4	R/W	0	gpioT_1_sel
3:0	R/W	0	gpioT_0_sel

**Table 7-227 PADCTRL\_PIN\_MUX\_REGG**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioT_15_sel
27:24	R/W	0	gpioT_14_sel
23:20	R/W	0	gpioT_13_sel
19:16	R/W	0	gpioT_12_sel
15:12	R/W	0	gpioT_11_sel
11:8	R/W	0	gpioT_10_sel
7:4	R/W	0	gpioT_9_sel
3:0	R/W	0	gpioT_8_sel

**Table 7-228 PADCTRL\_PIN\_MUX\_REGH**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioT_23_sel
27:24	R/W	0	gpioT_22_sel
23:20	R/W	0	gpioT_21_sel
19:16	R/W	0	gpioT_20_sel
15:12	R/W	0	gpioT_19_sel
11:8	R/W	0	gpioT_18_sel
7:4	R/W	0	gpioT_17_sel
3:0	R/W	0	gpioT_16_sel

**Table 7-229 PADCTRL\_PIN\_MUX\_REGJ**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioY_7_sel
27:24	R/W	0	gpioY_6_sel
23:20	R/W	0	gpioY_5_sel
19:16	R/W	0	gpioY_4_sel
15:12	R/W	0	gpioY_3_sel
11:8	R/W	0	gpioY_2_sel
7:4	R/W	0	gpioY_1_sel
3:0	R/W	0	gpioY_0_sel

**Table 7-230 PADCTRL\_PIN\_MUX\_REGK**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioY_15_sel
27:24	R/W	0	gpioY_14_sel
23:20	R/W	0	gpioY_13_sel
19:16	R/W	0	gpioY_12_sel
15:12	R/W	0	gpioY_11_sel
11:8	R/W	0	gpioY_10_sel
7:4	R/W	0	gpioY_9_sel
3:0	R/W	0	gpioY_7_sel

**Table 7-231 PADCTRL\_PIN\_MUX\_REGL**

Bit(s)	R/W	Default	Description
11:8	R/W	0	gpioY_18_sel
7:4	R/W	0	gpioY_17_sel
3:0	R/W	0	gpioY_16_sel

**Table 7-232 PADCTRL\_PIN\_MUX\_REGM**

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioW_7_sel
27:24	R/W	0	gpioW_6_sel
23:20	R/W	0	gpioW_5_sel
19:16	R/W	0	gpioW_4_sel
15:12	R/W	0	gpioW_3_sel
11:8	R/W	0	gpioW_2_sel

Bit(s)	R/W	Default	Description
7:4	R/W	0	gpioW_1_sel
3:0	R/W	0	gpioW_0_sel

Table 7-233 PADCTRL\_PIN\_MUX\_REGN

Bit(s)	R/W	Default	Description
31:28	R/W	0	gpioM_15_sel
27:24	R/W	0	gpioM_14_sel
23:20	R/W	0	gpioM_13_sel
19:16	R/W	0	gpioM_12_sel
15:12	R/W	0	gpioM_11_sel
11:8	R/W	0	gpioM_10_sel
7:4	R/W	0	gpioM_9_sel
3:0	R/W	0	gpioM_7_sel

Table 7-234 PADCTRL\_PIN\_MUX\_REGO

Bit(s)	R/W	Default	Description
3:0	R/W	0	gpioM_16_sel

### 7.8.3.2 PADCTRL\_GPIOx Registers

### 7.8.3.3 PADCTRL\_LOCK/PROT\_PIN\_MUX

### 7.8.3.4 PADCTRL Other Registers

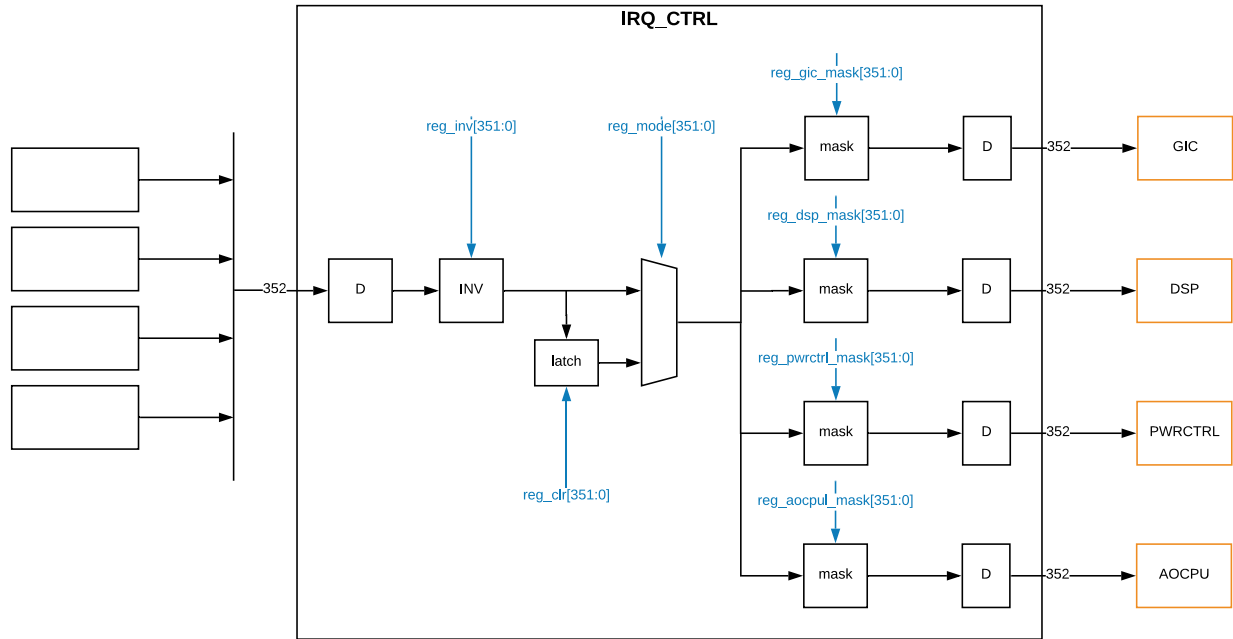
## 7.9 Interrupt Control

### 7.9.1 Overview

Generic Interrupt Controller (GIC) is a centralized resource that supports and manages interrupts in a system. For more details about GIC, please refer to the ARM GIC Architecture Specification.

The irq\_ctrl function includes following aspects:

- collect all irq source together and sync it;
- add invert or latch if needed;
- send to different destination after masked;
- all registers are TEE W/R only except REG\_CLR;
- all registers can be locked except REG\_CLR;



### 7.9.2 Interrupt Source

There are 351 interrupt sources in the chip. The control Bits of interrupts are listed in the following table.

GIC Bit	Interrupt Source	GIC Bit	Interrupt Source	GIC Bit	Interrupt Source
351	reg_soft_irq[7]	333	dmc_prot_irq	315	mipi_ism_irq[27]
350	reg_soft_irq[6]	332	dmc_mon_irq	314	mipi_ism_irq[26]
349	reg_soft_irq[5]	331	i2c_m_ao_b_irq	313	mipi_ism_irq[25]
348	reg_soft_irq[4]	330	i2c_m_ao_a_irq	312	mipi_ism_irq[24]
347	reg_soft_irq[3]	329	eth_qos_irq[9]	311	mipi_ism_irq[23]
346	reg_soft_irq[2]	328	eth_qos_irq[8]	310	mipi_ism_irq[22]
345	reg_soft_irq[1]	327	eth_qos_irq[7]	309	mipi_ism_irq[21]
344	reg_soft_irq[0]	326	eth_qos_irq[6]	308	mipi_ism_irq[20]
343	1'b0	325	eth_qos_irq[5]	307	mipi_ism_irq[19]
342	ddr1_phy_irq	324	eth_qos_irq[4]	306	mipi_ism_irq[18]
341	ddr_phy_irq	323	eth_qos_irq[3]	305	mipi_ism_irq[17]
340	eth_phy_irq_or	322	eth_qos_irq[2]	304	mipi_ism_irq[16]
339	dmc1_sec_irq	321	eth_qos_irq[1]	303	mipi_ism_irq[15]
338	dmc1_test_irq	320	eth_qos_irq[0]	302	mipi_ism_irq[14]
337	dmc1_prot_irq	319	mipi_ism_irq[31]	301	mipi_ism_irq[13]
336	dmc1_mon_irq	318	mipi_ism_irq[30]	300	mipi_ism_irq[12]
335	dmc_sec_irq	317	mipi_ism_irq[29]	299	mipi_ism_irq[11]
334	dmc_test_irq	316	mipi_ism_irq[28]	298	mipi_ism_irq[10]

GIC Bit	Interrupt Source	GIC Bit	Interrupt Source	GIC Bit	Interrupt Source
297	mipi_isp_irq[9]	260	A73IRQ[0]	227	STANDBYWFI[3]
296	mipi_isp_irq[8]	259	A73_STANDBYWFI[3]	226	STANDBYWFI[2]
295	mipi_isp_irq[7]	258	A73_STANDBYWFI[2]	225	STANDBYWFI[1]
294	mipi_isp_irq[6]	257	A73_STANDBYWFI[1]	224	STANDBYWFI[0]
293	mipi_isp_irq[5]	256	A73_STANDBYWFI[0]	223	gdc_irq
292	mipi_isp_irq[4]	255	assist_mbox_irq_ee[3]	222	edptx_int_1
291	mipi_isp_irq[3]	254	assist_mbox_irq_ee[2]	221	edptx_int_0
290	mipi_isp_irq[2]	253	assist_mbox_irq_ee[1]	220	vpu_sec_int
289	mipi_isp_irq[1]	252	assist_mbox_irq_ee[0]	219	mute_irq
288	mipi_isp_irq[0]	251	mbox_irq[3]	218	1'b0
287	1'b0	250	mbox_irq[2]	217	ge2d_int
286	1'b0	249	mbox_irq[1]	216	cusad_interrupt
285	1'b0	248	mbox_irq[0]	215	rdma_done_int
284	ldim_done_int	247	nIRQOUT[3:0][3]	214	vid1_wr_irq
283	venc1_vx1_int	246	nIRQOUT[3:0][2]	213	vid0_wr_irq
282	venc0_vx1_int	245	nIRQOUT[3:0][1]	212	vdin1_vsync_int
281	usb_iddig_irq1	244	nIRQOUT[3:0][0]	211	vdin1_hsync_int
280	usb_vbusdig_irq1	243	nFIQOUT[3:0][3]	210	vdin0_vsync_int
279	a73_nIRQOUT[3]	242	nFIQOUT[3:0][2]	209	vdin0_hsync_int
278	a73_nIRQOUT[2]	241	nFIQOUT[3:0][1]	208	viu1_mail_afbc_int[2]
277	a73_nIRQOUT[1]	240	nFIQOUT[3:0][0]	207	lc_curve_int
276	a73_nIRQOUT[0]	239	mbox_irq[4]	206	vpu_crash_int
275	a73_nFIQOUT[3]	238	STANDBYWFIL2	205	viu1_mail_afbc_int[1]
274	a73_nFIQOUT[2]	237	L2FLUSHDONE	204	hdmitx_interrupt
273	a73_nFIQOUT[1]	236	A53IRQ[8]	203	deint_pre_irq
272	a73_nFIQOUT[0]	235	A53IRQ[7]	202	deint_post_irq
271	1'b0	234	A53IRQ[6]	201	viff_empty_int_cpu
270	A73_STANDBYWFIL2	233	A53IRQ[5]	200	viu1_mail_afbc_int[0]
269	A73_L2FLUSHDONE	232	A53IRQ[4]	199	viu1_wm_int
268	A73IRQ[8]	231	A53IRQ[3]	198	viu1_dolby_int
267	A73IRQ[7]	230	A53IRQ[2]	197	viu1_vsync_int
266	A73IRQ[6]	229	A53IRQ[1]	196	viu1_hsync_int
265	A73IRQ[5]	228	A53IRQ[0]	195	viu1_line_n_int
264	A73IRQ[4]			194	viu2_vsync_int
263	A73IRQ[3]			193	viu2_hsync_int
262	A73IRQ[2]			192	viu2_line_n_int
261	A73IRQ[1]			191	spicc_5_int



GIC Bit	Interrupt Source	GIC Bit	Interrupt Source	GIC Bit	Interrupt Source
190	spicc_4_int	168	uart_a_irq	146	mali_irq_pp
189	spicc_3_int	167	spicc_2_int	145	mali_irq_gpmmu
188	anakin_irq[3]	166	i2c_s_a_irq	144	mali_irq_gp
187	anakin_irq[2]	165	i2c_m_f_irq	143	pcie_A_irq[7]
186	anakin_irq[1]	164	i2c_m_e_irq	142	pcie_A_irq[6]
185	anakin_irq[0]	163	i2c_m_d_irq	141	pcie_A_irq[5]
184	spicc_1_int	162	i2c_m_c_irq	140	pcie_A_irq[4]
183	spicc_0_int	161	i2c_m_b_irq	139	pcie_A_irq[3]
182	spi_int	160	i2c_m_a_irq	138	pcie_A_irq[2]
181	sar_adc_irq	159	dspb_error_irq[1]	137	pcie_A_irq[1]
180	cecb_irq	158	dspb_error_irq[0]	136	pcie_A_irq[0]
179	ceca_irq	157	dspa_error_irq[1]	135	pcie_A_edma_int[1]
178	sd_emmc_C_irq	156	dspa_error_irq[0]	134	pcie_A_edma_int[0]
177	sd_emmc_B_irq	155	ts_demux_irq	133	sectop_irq[1]
176	sd_emmc_A_irq	154	mali_irq_pp3	132	sectop_irq[0]
175	nand_irq	153	mali_irq_ppmmu2	131	u2drd_interrupt
174	smartcard_irq	152	mali_irq_pp2	130	u3drd_interrupt
173	uart_f_irq	151	mali_irq_ppmmu1	129	usb_iddig_irq0
172	uart_e_irq	150	mali_irq_pp1	128	usb_vbusdig_irq0
171	uart_d_irq	149	mali_irq_ppmmu0		
170	uart_c_irq	148	mali_irq_pp0		
169	uart_b_irq	147	mali_irq_pmu		

### 7.9.3 Register Description

#### Register Address

For below registers the base address is 0xfe00a000.

Each register final address = BASE + address \* 4.

The following lists describe the mapping between each IRQ\_CTRL register and its address.

- IRQCTRL\_IRQ\_INV0 0xfe00a000
- IRQCTRL\_IRQ\_INV1 0xfe00a004
- IRQCTRL\_IRQ\_INV2 0xfe00a008
- IRQCTRL\_IRQ\_INV3 0xfe00a00c
- IRQCTRL\_IRQ\_INV4 0xfe00a010
- IRQCTRL\_IRQ\_INV5 0xfe00a014
- IRQCTRL\_IRQ\_INV6 0xfe00a018
- IRQCTRL\_IRQ\_INV7 0xfe00a01c
- IRQCTRL\_IRQ\_INV8 0xfe00a020
- IRQCTRL\_IRQ\_INV9 0xfe00a024

- IRQCTRL\_IRQ\_INV10 0xfe00a028
- IRQCTRL\_MASK0\_GIC 0xfe00a040
- IRQCTRL\_MASK1\_GIC 0xfe00a044
- IRQCTRL\_MASK2\_GIC 0xfe00a048
- IRQCTRL\_MASK3\_GIC 0xfe00a04c
- IRQCTRL\_MASK4\_GIC 0xfe00a050
- IRQCTRL\_MASK5\_GIC 0xfe00a054
- IRQCTRL\_MASK6\_GIC 0xfe00a058
- IRQCTRL\_MASK7\_GIC 0xfe00a05c
- IRQCTRL\_MASK8\_GIC 0xfe00a060
- IRQCTRL\_MASK9\_GIC 0xfe00a064
- IRQCTRL\_MASK10\_GIC 0xfe00a068
- IRQCTRL\_MASK0\_DSP 0xfe00a080
- IRQCTRL\_MASK1\_DSP 0xfe00a084
- IRQCTRL\_MASK2\_DSP 0xfe00a088
- IRQCTRL\_MASK3\_DSP 0xfe00a08c
- IRQCTRL\_MASK4\_DSP 0xfe00a090
- IRQCTRL\_MASK5\_DSP 0xfe00a094
- IRQCTRL\_MASK6\_DSP 0xfe00a098
- IRQCTRL\_MASK7\_DSP 0xfe00a09c
- IRQCTRL\_MASK8\_DSP 0xfe00a0a0
- IRQCTRL\_MASK9\_DSP 0xfe00a0a4
- IRQCTRL\_MASK10\_DSP 0xfe00a0a8
- IRQCTRL\_MASK0\_PWRCTRL 0xfe00a0c0
- IRQCTRL\_MASK1\_PWRCTRL 0xfe00a0c4
- IRQCTRL\_MASK2\_PWRCTRL 0xfe00a0c8
- IRQCTRL\_MASK3\_PWRCTRL 0xfe00a0cc
- IRQCTRL\_MASK4\_PWRCTRL 0xfe00a0d0
- IRQCTRL\_MASK5\_PWRCTRL 0xfe00a0d4
- IRQCTRL\_MASK6\_PWRCTRL 0xfe00a0d8
- IRQCTRL\_MASK7\_PWRCTRL 0xfe00a0dc
- IRQCTRL\_MASK8\_PWRCTRL 0xfe00a0e0
- IRQCTRL\_MASK9\_PWRCTRL 0xfe00a0e4
- IRQCTRL\_MASK10\_PWRCTRL 0xfe00a0e8
- IRQCTRL\_MASK0\_AOCPU 0xfe00a100
- IRQCTRL\_MASK1\_AOCPU 0xfe00a104
- IRQCTRL\_MASK2\_AOCPU 0xfe00a108
- IRQCTRL\_MASK3\_AOCPU 0xfe00a10c
- IRQCTRL\_MASK4\_AOCPU 0xfe00a110
- IRQCTRL\_MASK5\_AOCPU 0xfe00a114
- IRQCTRL\_MASK6\_AOCPU 0xfe00a118
- IRQCTRL\_MASK7\_AOCPU 0xfe00a11c
- IRQCTRL\_MASK8\_AOCPU 0xfe00a120
- IRQCTRL\_MASK9\_AOCPU 0xfe00a124

- IRQCTRL\_MASK10\_AOCPU 0xfe00a128
- IRQCTRL\_IRQ\_MODE0 0xfe00a140
- IRQCTRL\_IRQ\_MODE1 0xfe00a144
- IRQCTRL\_IRQ\_MODE2 0xfe00a148
- IRQCTRL\_IRQ\_MODE3 0xfe00a14c
- IRQCTRL\_IRQ\_MODE4 0xfe00a150
- IRQCTRL\_IRQ\_MODE5 0xfe00a154
- IRQCTRL\_IRQ\_MODE6 0xfe00a158
- IRQCTRL\_IRQ\_MODE7 0xfe00a15c
- IRQCTRL\_IRQ\_MODE8 0xfe00a160
- IRQCTRL\_IRQ\_MODE9 0xfe00a164
- IRQCTRL\_IRQ\_MODE10 0xfe00a168
- IRQCTRL\_IRQ\_LATCH0 0xfe00a180
- IRQCTRL\_IRQ\_LATCH1 0xfe00a184
- IRQCTRL\_IRQ\_LATCH2 0xfe00a188
- IRQCTRL\_IRQ\_LATCH3 0xfe00a18c
- IRQCTRL\_IRQ\_LATCH4 0xfe00a190
- IRQCTRL\_IRQ\_LATCH5 0xfe00a194
- IRQCTRL\_IRQ\_LATCH6 0xfe00a198
- IRQCTRL\_IRQ\_LATCH7 0xfe00a19c
- IRQCTRL\_IRQ\_LATCH8 0xfe00a1a0
- IRQCTRL\_IRQ\_LATCH9 0xfe00a1a4
- IRQCTRL\_IRQ\_LATCH10 0xfe00a1a8
- IRQCTRL\_IRQ\_LATCH\_CLR0 0xfe00a1c0
- IRQCTRL\_IRQ\_LATCH\_CLR1 0xfe00a1c4
- IRQCTRL\_IRQ\_LATCH\_CLR2 0xfe00a1c8
- IRQCTRL\_IRQ\_LATCH\_CLR3 0xfe00a1cc
- IRQCTRL\_IRQ\_LATCH\_CLR4 0xfe00a1d0
- IRQCTRL\_IRQ\_LATCH\_CLR5 0xfe00a1d4
- IRQCTRL\_IRQ\_LATCH\_CLR6 0xfe00a1d8
- IRQCTRL\_IRQ\_LATCH\_CLR7 0xfe00a1dc
- IRQCTRL\_IRQ\_LATCH\_CLR8 0xfe00a1e0
- IRQCTRL\_IRQ\_LATCH\_CLR9 0xfe00a1e4
- IRQCTRL\_IRQ\_LATCH\_CLR10 0xfe00a1e8
- IRQCTRL\_LOCK\_BIT 0xfe00a200
- IRQCTRL\_PROT\_BIT 0xfe00a204
- IRQCTRL\_SOFT\_IRQ 0xfe00a220

## Register Description

Table 7-235 RQCTRL\_IRQ\_INV0~10

Bits	R/W	Defaults	Description
351:0	R/W	0	if set 1, will invert irq level

**Table 7-236 IRQCTRL\_IRQ\_MODE0~10**

Bits	R/W	Defaults	Description
351:0	R/W	0	if set 1, will select latched irq; if set 0, will select original irq source;

**Table 7-237 IRQCTRL\_IRQ\_LATCH0~10**

Bits	R/W	Defaults	Description
351:0	R	0	the latched irq value

**Table 7-238 IRQCTRL\_IRQ\_LATCH\_CLR0~10**

Bits	R/W	Defaults	Description
351:0	R/W	0	if set 1, will keep latched irq to low

**Table 7-239 IRQCTRL\_IRQ\_MASK0~10\_\*\*\***

Bits	R/W	Defaults	Description
351:0	R/W	all 1	if set 0, will keep irq as low

**Table 7-240 IRQCTRL\_LOCK\_BIT**

Bits	R/W	Defaults	Description
6	R/W	0	write 1 once, if set 1 ,will lock all IRQ_MODE
4	R/W	0	write 1 once, if set 1 ,will lock all IRQ_MASK_AOCPU
3	R/W	0	write 1 once, if set 1 ,will lock all IRQ_MASK_PWRCTRL
2	R/W	0	write 1 once, if set 1 ,will lock all IRQ_MASK_DSP
1	R/W	0	write 1 once, if set 1 ,will lock all IRQ_MASK_GIC
0	R/W	0	write 1 once, if set 1 ,will lock all IRQ_INV

**Table 7-241 IRQCTRL\_PROT\_BIT**

Bits	R/W	Defaults	Description
6	R/W	0	if set 1 ,will lock all IRQ_MODE
4	R/W	0	if set 1 ,will lock all IRQ_MASK_AOCPU
3	R/W	0	if set 1 ,will lock all IRQ_MASK_PWRCTRL
2	R/W	0	if set 1 ,will lock all IRQ_MASK_DSP
1	R/W	0	if set 1 ,will lock all IRQ_MASK_GIC
0	R/W	0	if set 1 ,will lock all IRQ_INV

## 7.10 Timer

### 7.10.1 Overview

The SOC contains 15 general purpose timers.

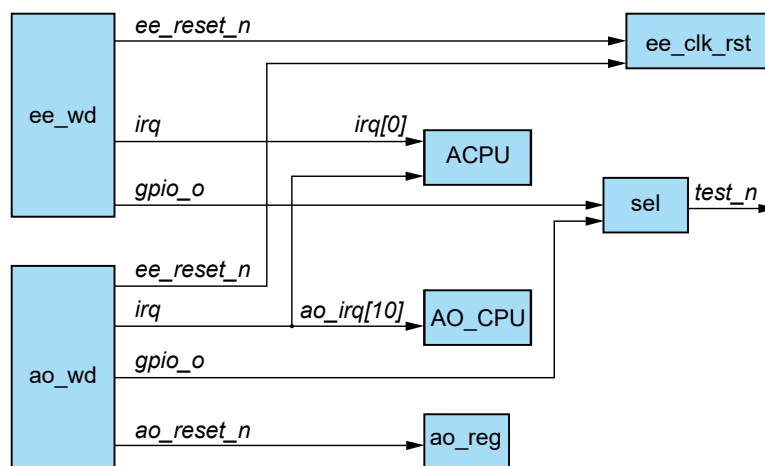
### 7.10.2 General-Purpose Timer

The SOC contains a number of general-purpose timers that can be used as general counters or interrupt generators. Each counter (except TIMER E) can be configured as a periodic counter (for generating periodic interrupts) or a simple count-down and stop counter. Additionally, the timers have a programmable count rate ranging from 1uS to 1mS. The table below outlines the general-purpose timers available in the chip.

Timer	Timebase Options	Counter size	Comment
Timer A	1uS, 10uS, 100uS	16-bits	The 16-bit counter allows the timer to generate interrupts as infrequent as every 65.535 Seconds
Timer B	1uS, 10uS, 100uS	16-bits	
Timer C	1uS, 10uS, 100uS	16-bits	
Timer D	1uS, 10uS, 100uS	16-bits	
Timer E	1uS, 10uS, 100uS	64-bits	Doesn't generate an interrupt. This is a count up counter that counts from 0 to 0xFFFFFFFF. The counter can be written at any time to reset the value to 0.
Timer F	1uS, 10uS, 100uS	16-bits	
Timer G	1uS, 10uS, 100uS	16-bits	
Timer H	1uS, 10uS, 100uS	16-bits	
Timer I	1uS, 10uS, 100uS	16-bits	
Timer J	1uS, 10uS, 100uS	16-bits	

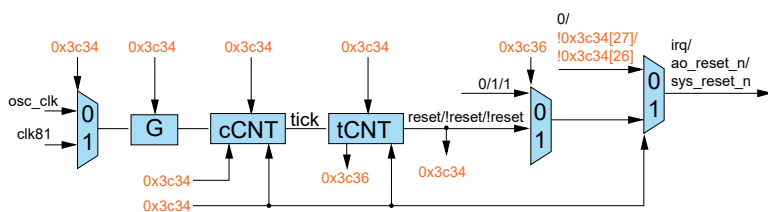
### 7.10.3 Watchdog Timer

There are also two watchdog timers, one in AO and the other in EE domain, illustrated as following:



T02FC37

Figure 7-28 EE domain Watchdog Timer Design



T02FC38

The AO Domain watchdog timer is driven from the system clock (typically 157Mhz). It is a 16-bit counter that is periodically reset by either the AO CPU or the System CPU. This AO-watchdog timer can be used to generate an interrupt of the AO domain. Additionally, the AO-watchdog timer can be used to “enable” a delay generator that can toggle a GPIO pin (currently the TEST\_n I/O pad). The “delay generator” allows an interrupt to first be acknowledged by the AO-CPU before the TEST\_N pad is toggled. The “delay generator” is programmable from 1 to 65535 system clocks (typically 417uS).

It should be noted that the AO watchdog timer can also be used to reset the AO domain but this feature is only used when operating in a suspend mode (only the AO-domain is powered). As long as the system periodically resets the AO-watchdog timer the WD\_GPIO\_CNT (delay generator) will not be enabled and the I/O pad will not toggle.

**Note**

The maximum delay between two AO-watchdog periodic resets is about 100mS (assuming a 157Mhz system clock).

The EE Domain watchdog timer is driven by the 24Mhz crystal clock and can be used to generate an interrupt to the system CPU or optionally, the watchdog timer can completely reset the chip (causing a cold boot). There are a few registers that are not affected by watchdog timer. These registers are only reset by the external RESET\_n I/O pad and can be used to store information related to a possible watchdog event. As long as the system CPU periodically resets the EE-watchdog timer, it will never timeout and cause an interrupt or system reset.

**Note**

The maximum delay between two EE-watchdog periodic resets is about 8.3 Seconds. This time is independent of the system clocks and is driven by the external 24Mhz crystal.

**7.10.4 Register Description**

Each register final address = 0xfe010000 + address \* 4

The following lists describe the mapping between each timer register and its address.

SYSCTRL_TIMER_A_CTRL	0xfe0100c0
SYSCTRL_TIMER_A	0xfe0100c4
SYSCTRL_TIMER_B_CTRL	0xfe0100c8
SYSCTRL_TIMER_B	0xfe0100cc
SYSCTRL_TIMER_C_CTRL	0xfe0100d0
SYSCTRL_TIMER_C	0xfe0100d4
SYSCTRL_TIMER_D_CTRL	0xfe0100d8

SYSCTRL_TIMERD	0xfe0100dc
SYSCTRL_SEC_TIMER_A_CTRL	0xfe0100e0
SYSCTRL_SEC_TIMER_A	0xfe0100e4
SYSCTRL_TIMER_E_CTRL	0xfe0100e8
SYSCTRL_TIMER_E	0xfe0100ec
SYSCTRL_TIMER_E_HI	0xfe0100f0
SYSCTRL_TIMER_F_CTRL	0xfe0100f4
SYSCTRL_TIMER_F	0xfe0100f8
SYSCTRL_TIMER_F_HI	0xfe0100fc
SYSCTRL_TIMER_G_CTRL	0xfe010100
SYSCTRL_TIMER_G	0xfe010104
SYSCTRL_TIMER_H_CTRL	0xfe010108
SYSCTRL_TIMER_H	0xfe01010c
SYSCTRL_TIMER_I_CTRL	0xfe010110
SYSCTRL_TIMER_I	0xfe010114
SYSCTRL_TIMER_J_CTRL	0xfe010118
SYSCTRL_TIMER_J	0xfe01011c
SYSCTRL_SEC_TIMER_E_CTRL	0xfe010120
SYSCTRL_SEC_TIMER_E	0xfe010124
SYSCTRL_SEC_TIMER_E_HI	0xfe010128
SYSCTRL_TIMER_90K	0xfe01015c
SYSCTRL_HPG_TIMER	0xfe010164

**Table 7-242 SYSCTRL\_TIMER\_A/B/C/D/G/H/I/J\_CTRL**

Bit(s)	R/W	Default	Description
7	R/W	1	timer en
6	R/W	1	timer mode: 0: generate irq once; 1: generate irq periodic;
1:0	R/W	0	clk_sel: 0: cts_sys_clk; 1:1us; 2:10us; 3:100us

**Table 7-243 SYSCTRL\_TIMER\_A/B/C/D/G/H/I/J**

Bit(s)	R/W	Default	Description
31:16	R	0	ttimer threshold
15:0	R/W	0	write: clean timer count; set timer threshold. irq will generate when timer count equal to threshold read: timer count value

**Table 7-244 SYSCTRL\_SEC\_TIMER\_A\_CTRL**

Bit(s)	R/W	Default	Description
7	R/W	1	timer en
6	R/W	1	timer mode: 0: generate irq once; 1: generate irq periodic;
1:0	R/W	0	clk_sel: 0: cts_sys_clk; 1:1us; 2:10us; 3:100us

**Table 7-245 SYSCTRL\_SEC\_TIMER\_A**

Bit(s)	R/W	Default	Description
31:16	R	0	timer threshold
15:0	R/W	0	write: clean timer count; set timer threshold. irq will generate when timer count equal to threshold read: timer count value

**Table 7-246 SYSCTRL\_TIMER\_E/F\_CTRL**

Bit(s)	R/W	Default	Description
7	R/W	1	timer en
1:0	R/W	1	clk_sel: 0: cts_sys_clk; 1:1us; 2:10us; 3:100us

**Table 7-247 SYSCTRL\_TIMER\_E/F**

Bit(s)	R/W	Default	Description
31:0	R/W	0	write: clean timer count; read: update timer count[63:32] to TIMER_HI timer count[31:0]



**Table 7-248 SYSCTRL\_TIMERE/F\_HI**

Bit(s)	R/W	Default	Description
31:0	R	0	timer count[63:32]

**Table 7-249 SYSCTRL\_SEC\_TIMERE\_CTRL**

Bit(s)	R/W	Default	Description
7	R/W	1	timer en
1:0	R/W	1	clk_sel: 0: cts_sys_clk; 1: 1us; 2: 10us; 3: 100us

**Table 7-250 SYSCTRL\_SEC\_TIMERE**

Bit(s)	R/W	Default	Description
31:0	R/W	0	write: clean timer count; read: update timer count[63:32] to TIMER_HI timer count[31:0]

**Table 7-251 SYSCTRL\_SEC\_TIMERE\_HI**

Bit(s)	R/W	Default	Description
31:0	R	0	timer count[63:32]

**Table 7-252 SYSCTRL\_TIMER90K**

Bit(s)	R/W	Default	Description
15:0	R/W	0x384	90khz timer divider

**Table 7-253 SYSCTRL\_HPG\_TIMER**

Bit(s)	R/W	Default	Description
15:0	R/W	0x0	generate h2tmr_irq if timer_cnt equal to HPG_TIMER; the timer_cnt will increment by 90khz timer; the timer_cnt will clear when generated h2tmr_irq;

## 7.11 Crypto

### 7.11.1 Overview

The crypto engine is one encrypt/decrypt function accelerator. Crypto engine supports 2 different modes, i.e. SCP (System Control Processor) secure and SCP non secure. The crypto engine has special internal DMA controller to transfer data.

It has the following features.

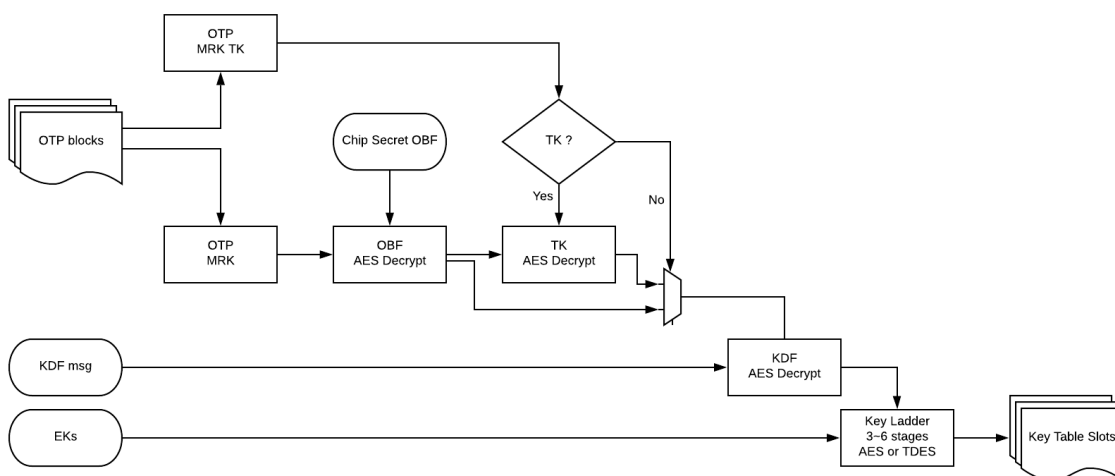
- Trustzone based Trusted Execution Environment (TEE)
- Secured boot, encrypted OTP, encrypted DRAM with memory integrity checker, hardware key ladder and internal control buses and storage
- Separated secure/non-secure Entropy true RNG
- Pre-region/ID memory security control and electric fence
- Hardware based Trusted Video Path (TVP) , video watermarking and secured contents (needs SecureOS software)
- Secured IO and secured clock

### 7.11.2 Key Ladder

The Key Ladder is a series of TDES / AES crypto processes that iterates on different user supplied and OTP keys. The key ladder module uses a single AES / TDES crypto module and iterates by using internal storage to hold temporary states.

The key ladder flow figure is shown below.

Figure 7-29 Key Ladder Flow



The key ladder supports MKL and ETSI modes.

### 7.11.3 RNG

Functionality the Random Number Generator (RNG) contains two main modules, True Random Number Generator (TRNG) and Deterministic Random Number Generator (DRNG).

- True Random Number Generator (TRNG): is realized by using metastability and jitter for random bit generation based on four free running ring oscillator

- Deterministic Random Number Generator (DRNG): which has 32-bit random number generator, is mainly designed to increase the throughput and do post-processing of the Digital TRNG, which will need hundreds cycles to collect entropy.

### 7.11.4 CTR\_DRBG

CTR\_DRBG contains two main modules, True Random Number Generator (TRNG) and Deterministic Random Number Generator (DRNG).

- True Random Number Generator (TRNG): is realized by using metastability and jitter for random bit generation based on four free running ring oscillator
- Deterministic Random Number Generator (DRNG): which has 32-bit random number generator, is mainly designed to increase the throughput and do post-processing of the Digital TRNG, which will need hundreds cycles to collect entropy.

It supports health test, which contains two functions: repetition count test and adaptive proportion test.

### 7.11.5 EFUSE

Synopsys Design Ware OTP NVM IP, Previous Kilopass OTP, is selected as OTP IP for this project.

TSMC 12FFC 1.8V 32Kbits

Model: XPM KPTS12FFC32-R08W01

Program x1 bit

Read x8 bits

Internal charge pump, support field programming

32 Kbits OTP is divided into 256 blocks, each block has 128 bits.

The blocks are labeled from 0 to 255.

The bits in each block are labeled from 0 to 127.

OTP controller takes job from users, there are 4 types of tasks.

- Sandwich Read
- Check
- Write
- Test

## 7.12 MIPI\_ISP TOP

### 7.12.1 MIPI\_CSI\_DPHY

The SoC has 4 DPHYs: dphy0/1 and dphy2/3 as two groups.

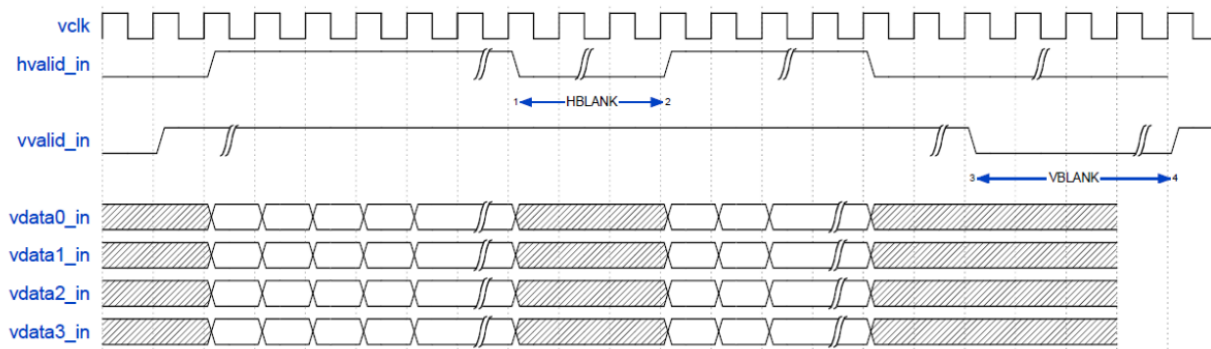
- Each DPHY can support up to 4 data lane and 1 clock lane.
- It can support up to 2.5Gbps/lane.

### 7.12.2 MIPI\_ISP\_ADAPT

MIPI\_ISP\_ADAPT is used to change IDI interface to pixel format to meet isp\_core input requirements, which is:

The high bit of the data valid bit is valid, and the input of different channels must be aligned, HBLANK> 32 cycles VBLANK> 48 lines

**Figure 7-30 ISP Core Input Requirement**



mipi\_isp\_adapt supports 4 work modes for RAW input.

- SDR with ddr path mode
- SDR with direct path mode
- HDR with ddr path+direct path mode
- HDR with ddr path+ddr path mode

And supports 2 work modes for YUV/RGB input:

- SDR with ddr path mode
- SDR with direct path mode(comb direct path)

**mipi\_adapt\_frontend0/1/2/3**

Change IDI data to pixel format, write to ddr to direct path.

- Support virtual channel mode or line info mode for HDR RAW input
- Support pixel/line crop for RAW input, only line crop for RGB/YUV input(pixel crop done in mipi\_adapt\_pixel module)

Data Type	RAW Data Arrangement in DDR
RAW6	{P15,2'h0,P14,2'h0...P3,2'h0,P2,2'h0,P1,2'h0,P0,2'h0}
RAW7	{P15,1'h0,P14,1'h0...P3,1'h0,P2,1'h0,P1,1'h0,P0,1'h0}
RAW8	{P15,P14...P3,P2,P1,P0}
RAW10	{ P12[9:2],P11[1:0],P10[1:0],P9[1:0],P8[1:0],P11[9:2],P10[9:2],P9[9:2],P8[9:2] ...P3[1:0],P2[1:0],P1[1:0],P0[1:0],P3[9:2],P2[9:2],P1[9:2],P0[9:2]}
RAW12	{P10[11:4],P9[3:0],P8[3:0],P9[11:4],P8[11:4]...P1[3:0],P0[3:0],P1[11:4],P0[11:4]}
RAW14	{P3[5:0],2'h0,P3[13:6],P2[5:0],2'h0,P2[13:6],P2[5:0],2'h0,P1[13:6],P0[5:0],2'h0,P0[13:6]}

**mipi\_adapt\_drrd0/1**

This module reads data which is written by frontend, or accept direct path/direct comb path data from frontend.

It also writes to line buf.

**mipi\_adapt\_pixel0/1**

This module change data in line buf to pixel format.

For YUV/RGB format, it can do pixel crop in this module.

For YUV422, it can convert YUV422 to YUV444.

### mipi\_adapt\_alig

The module is used to generate the frame structure required by the ISP and to align pixel data of the two field exposures in HDR mode.

### mipi\_fe\_mux

The module is used to mux single from frontend0~3 to ddr\_rd.

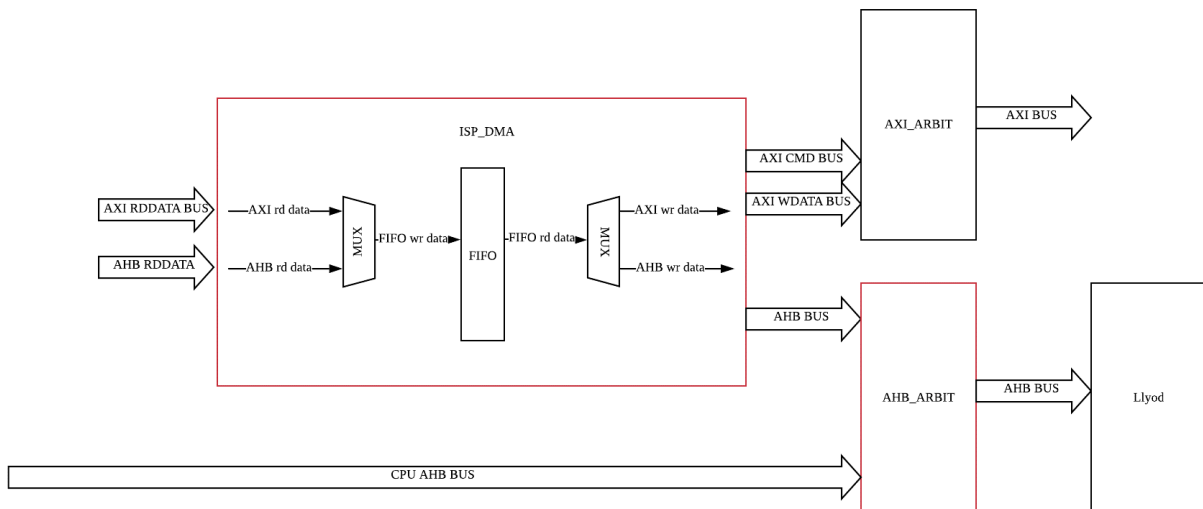
It can add delay for each frame\_vs single, and monitor delay from frame\_vs to 1<sup>st</sup> valid data.

## 7.12.3 ISP\_AHB\_DMA

ISP\_AHB\_DMA is used to set ISP\_CORE registers.

- Supports up to 8 multiple tasks, each task is a one-way transmission.
- When DMA works, the CPU can normally access the ISP\_CORE registers.

Figure 7-31 ISP\_AHB\_DMA Diagram



## 7.12.4 ISP\_APB\_DMA

ISP\_APB\_DMA is used to set apb bus registers, there are 16 trigger sources for APB\_DMA.

Each trigger can trig 2 tasks, each task can either write or read.

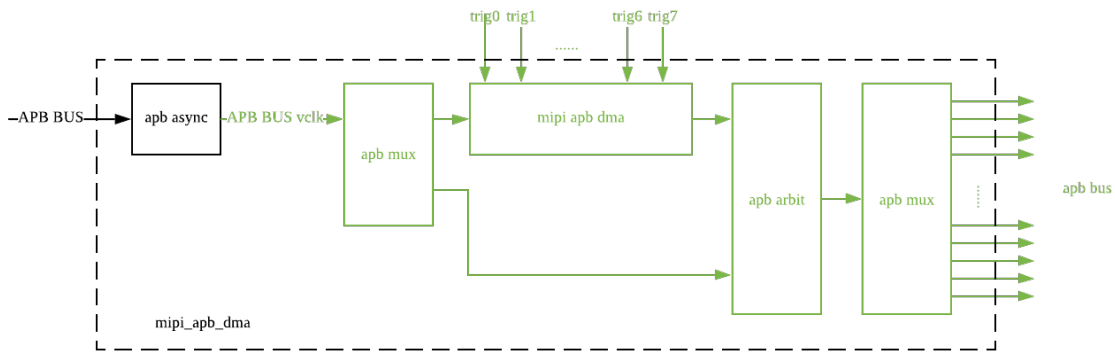
APB CMD store in DDR in such format:

- WR CMD: 32bit waddr + 32bit wdata, dword-aligned
- RD CMD: 32bit raddr + 32bit raddr, word-aligned

And each trigger source have pingpong task setting.

When DMA work, the CPU can normally access apb registers.

Figure 7-32 ISP\_APB\_DMA Diagram



APB trigger source list:

Source Index	Trigger Source	Source Index	Trigger Source
15	NA	7	adapt_de_wr
14	NA	6	crop_mif
13	NA	5	ds2
12	NA	4	ds1
11	NA	3	ds0
10	tnr_de_rd	2	md2
9	tnr_de_wr	1	flicker
8	adapt_de_rd	0	md

## 7.12.5 Register Description

### 7.12.5.1 MIPI ISP Registers

Table 7-254 Registers Base Address

Module	Base Address
mipi_isp_adapt_frontend0	0xfe3b0000
mipi_isp_adapt_frontend1	0xfe3b0400
mipi_isp_adapt_frontend2	0xfe3b0800
mipi_isp_adapt_cfg	0xfe3b1000
mipi_isp_ds0	0xfe3b1400
mipi_isp_ds1	0xfe3b1800
mipi_isp_ds2	0xfe3b1c00
mipi_isp_cropmif	0xfe3b2000
mipi_isp_md	0xfe3b2400
mipi_isp_md2	0xfe3b2800

Module	Base Address
mipi_isp_fliker	0xfe3b2c00
mipi_isp_top	0xfe3b3000
mipi_isp_ahb_dma	0xfe3b3400
mipi_isp_axi_arbit0	0xfe3b3800
mipi_isp_axi_arbit1	0xfe3b3c00
mipi_isp_axi_adapt_de	0xfe3b4000
mipi_isp_axi_tnr_de	0xfe3b4800
mipi_isp_axi_arbit2	0xfe3b5800
mipi_isp_apb_dma	0xfe3bbc00
mipi_isp_ana	0xfe3bdc00
mipi_isp_host3	0xfe3be000
mipi_isp_host2	0xfe3be400
mipi_isp_host1	0xfe3be800
mipi_isp_host0	0xfe3bec00
mipi_isp_dphy3	0xfe3bf000
mipi_isp_dphy2	0xfe3bf400
mipi_isp_dphy1	0xfe3bf800
mipi_isp_dphy0	0xfe3bfc00

## MIPI\_ISP\_PHY

Table 7-255 reg\_csi\_phy\_cntl0 0X0

Bit	R/W	Default	Description
<31:30>		2'b0	reserved
<29>		1'b1	lane 5 hs digital logic path enable
<28>		1'b1	lane 4 hs digital logic path enable
<27>		1'b1	lane 3 hs digital logic path enable
<26>		1'b1	lane 2 hs digital logic path enable
<25>		1'b1	lane 1 hs digital logic path enable
<24>		1'b1	lane 0 hs digital logic path enable
<23:20>		4'b0100	lp high-vth control(one hot)
<19:16>		4'b0010	lp low-vth control(one hot)
<15>		1'b0	reserved
<14:11>		4'b1011	hs mode rx termination trimming control
<10>		1'b1	hs/lp bias current enable control
<9>		1'b1	hs eq cml resistor control(2.5G:1; 1.5G:0)

Bit	R/W	Default	Description
<8>		1'b1	hs d2d cml resistor control(2.5G:1; 1.5G:0)
<7:6>		2'b10	hs RX receiver eq bias current control(2.5G:10; 1.5G:00)
<5:4>		2'b10	hs RX receiver d2d bias current control(2.5G:10; 1.5G:00)
<3:2>		2'b10	hs RX receiver d2s bias current control(2.5G:10; 1.5G:00)
<1:0>		2'b0	lp RX receiver bias current control

Table 7-256 reg\_csi\_phy\_cntl1 0X1

Bit	R/W	Default	Description
<31:30>		2'b0	reserved
<29>		1'b0	hs hsd3_r/f use clock hsout5_ck(1'b1) or hsout2_ck(1'b0)
<28>		1'b0	hs hsd2_r/f use clock hsout5_ck(1'b1) or hsout2_ck(1'b0)
<27>		1'b0	hs hsd1_r/f use clock hsout5_ck(1'b1) or hsout2_ck(1'b0)
<26>		1'b0	hs hsd0_r/f use clock hsout5_ck(1'b1) or hsout2_ck(1'b0)
<25>		1'b1	hs sync block enable control
<24>		1'b1	common bg enable
<23>		1'b0	data channel eq enable
<22>		1'b0	clock channel eq enable
<21>		1'b1	data channel eq max gain
<20>		1'b1	clock channel eq max gain
<19:18>		2'b10	hs eq res select, min:11; medium:10; max:01
<17:16>		2'b10	hs eq cap select, small:10 ; big:11
<15:6>		10'b0	reserved
<5:0>		6'b0	lane5 to lane1 P N pins swap(default:0; P N swap:1)

Table 7-257 reg\_csi\_phy\_cntl2 0X2

Bit	R/W	Default	Description
<31>		1'b0	reg_csi2_pt_en : production test
<26>		1'b0	reg_skew_from_dphy, 1'b1□skew from dphy, 1'b0:from fixed register
<25>		1'b0	reg_lane5_sel_dphy, lane5 MUX to 1'b1:DPHY1,1'b0:DPHY0
<24>		1'b0	reg_lane4_sel_dphy, lane4 MUX to 1'b1:DPHY1,1'b0:DPHY0
<23>		1'b0	reg_lane3_sel_dphy, lane3 MUX to 1'b1:DPHY1,1'b0:DPHY0
<22>		1'b0	reg_lane2_sel_dphy, lane2 MUX to 1'b1:DPHY1,1'b0:DPHY0
<21>		1'b0	reg_lane1_sel_dphy, lane1 MUX to 1'b1:DPHY1,1'b0:DPHY0
<20>		1'b0	reg_lane0_sel_dphy: lane0 MUX to 1'b1:DPHY1,1'b0:DPHY0
<19:0>		20'b0	reg_csi2_pt_ctrl



**Table 7-258 reg\_csi\_phy\_cntl3 0X3**

Bit	R/W	Default	Description
<29:25>		4'b0	reg_lane0_skew, lane0 fixed skew value
<24:20>		4'b0	reg_lane0_skew, lane0 fixed skew value
<19:15>		4'b0	reg_lane0_skew, lane0 fixed skew value
<14:10>		4'b0	reg_lane0_skew, lane0 fixed skew value
<9:5>		4'b0	reg_lane0_skew, lane0 fixed skew value
<4:0>		4'b0	reg_lane0_skew, lane0 fixed skew value

reg\_csi\_phy\_cntl4 0x4 for analog phy 1, description same as reg\_phy\_cnt0

reg\_csi\_phy\_cntl5 0x5 for analog phy 1, description same as reg\_phy\_cnt1

reg\_csi\_phy\_cntl6 0x6 for analog phy 1, description same as reg\_phy\_cnt2

reg\_csi\_phy\_cntl7 0x7 for analog phy 1, description same as reg\_phy\_cnt3

## MIPI\_ISP\_ADAPT

**Table 7-259 MIPI\_ISP\_ADAPT Register Address**

Addr	Name	RW	Function
0x00	CSI2_CLK_RESET	RW	Clock and reset control.
0x01	CSI2_GEN_CTRL0	RW	General control.
0x02	CSI2_GEN_CTRL1	RW	General control.
0x03	CSI2_X_START_END_ISP	RW	Direct path X start/end.
0x04	CSI2_Y_START_END_ISP	RW	Direct path Y start/end.
0x05	CSI2_X_START_END_MEM	RW	MEM path X start/end.
0x06	CSI2_Y_START_END_MEM	RW	MEM path Y start/end.
0x07	CSI2_VC_MODE	RW	Use virtual channel to separate direct and MEM path.
0x08	CSI2_VC_MODE2_MATCH_MASK_A_L	RW	To separate direct and MEM path.
0x09	CSI2_VC_MODE2_MATCH_MASK_A_H	RW	To separate direct and MEM path.
0x0a	CSI2_VC_MODE2_MATCH_A_L	RW	To separate direct and MEM path.
0x0b	CSI2_VC_MODE2_MATCH_A_H	RW	To separate direct and MEM path.
0x0c	CSI2_VC_MODE2_MATCH_B_L	RW	To separate direct and MEM path.
0x0d	CSI2_VC_MODE2_MATCH_B_H	RW	To separate direct and MEM path.
0x0e	CSI2_DDR_START_PIX	RW	DDR start address for pixel-based data.
0x0f	CSI2_DDR_START_PIX_ALT	RW	Alternative DDR start address for pixel-based data. Ping-pong mode.

Addr	Name	RW	Function
0x10	CSI2_DDR_STRIDE_PIX	RW	DDR line stride for pixel-based data.
0x11	CSI2_DDR_START_OTHER	RW	DDR start address for non-pixel-based data.
0x12	CSI2_DDR_START_OTHER_ALT	RW	Alternative DDR start address for non-pixel-based data. Ping-pong mode.
0x13	CSI2_DDR_MAX_BYTES_OTHER	RW	For DDR store non-pixel-based.
0x14	CSI2_INTERRUPT_CTRL_STAT	RW	Interrupt control/status.
0x15	CSI2_VC_MODE2_MATCH_MASK_B_L	RW	To separate direct and MEM path.
0x16	CSI2_VC_MODE2_MATCH_MASK_B_H	RW	To separate direct and MEM path.
0x17	CSI2_DDR_LOOP_LINES_PIX	RW	DDR loopback for pixel-based.
0x20	CSI2_GEN_STAT0	R	General status.
0x21	CSI2_ERR_STAT0	RW	Error status.
0x22	CSI2_PIC_SIZE_STAT	R	Picture size seen at direct path.
0x23	CSI2_DDR_WPTR_STAT_PIX	R	Current DDR address for pixel-based.
0x24	CSI2_DDR_WPTR_STAT_OTHER	R	Current DDR address for non-pixel-based.
0x25	CSI2_STAT_MEM_0	R	Status for MEM path.
0x26	CSI2_STAT_MEM_1	R	Status for MEM path.
0x27	CSI2_STAT_MEM_2	R	Status for MEM path.
0x28	CSI2_STAT_GEN_SHORT_08	R	Generic short packet 0x08.
0x29	CSI2_STAT_GEN_SHORT_09	R	Generic short packet 0x09.
0x2a	CSI2_STAT_GEN_SHORT_0A	R	Generic short packet 0x0A.
0x2b	CSI2_STAT_GEN_SHORT_0B	R	Generic short packet 0x0B.
0x2c	CSI2_STAT_GEN_SHORT_0C	R	Generic short packet 0x0C.
0x2d	CSI2_STAT_GEN_SHORT_0D	R	Generic short packet 0x0D.
0x2e	CSI2_STAT_GEN_SHORT_0E	R	Generic short packet 0x0E.
0x2f	CSI2_STAT_GEN_SHORT_0F	R	Generic short packet 0x0F.
0x30	CSI2_STAT_TYPE_RCVD_L	RW	Record received packet types.
0x31	CSI2_STAT_TYPE_RCVD_H	RW	Record received packet types.
0x32	CSI2_DDR_START_PIX_B	RW	DDR start address for pixel-based data (isp2ddr).
0x33	CSI2_DDR_START_PIX_B_ALT	RW	Alternative DDR start address for pixel-based data. Ping-pong mode (isp2ddr).
0x34	CSI2_DDR_STRIDE_PIX_B	RW	DDR line stride for pixel-based data (isp2ddr).
0x35	CSI2_DDR_WPTR_STAT_PIX_B	R	Current DDR address for pixel-based (isp2ddr).
0x36	CSI2_STAT_MEM_3	R	Status for MEM path.

Addr	Name	RW	Function
0x37	CSI2_AXI_UGT_CNTL0	RW	AXI bus urgent control
0x38	CSI2_AXI_UGT_CNTL1	RW	AXI bus urgent control
0x3a	CSI2_AXI_UGT_ST	R	AXI bus urgent status
0x3b	CSI2_LINE_SUP_CNTL0	RW	add data to mem path meet min data number
0x3c	CSI2_LINE_SUP_ST	R	Status for mem path data add

Table 7-260 CSI2\_CLK\_RESET

Bit	R/W	Default	Description
31:3	R	0	Reserved
2	RW	0	clk_auto_gate_off: Enable/disable clock gating. 0=Enable auto clock gating for power saving; 1=Disable auto clock gating.
1	RW	0	clk_enable: To enable/disable clock to mipi_csi_adapt_frontend module. 0=Disable clock; 1=Enable clock.
0	RW	1	sw_reset: To reset mipi_csi_adapt_frontend module. 0=Release from reset; 1=Apply reset.

Table 7-261 CSI2\_GEN\_CTRL0

Bit	R/W	Default	Description
31:21	R	0	Reserved
29	RW	0	awfifo_cnt_check. 0=original behaviour,mem2ddr/isp2ddr_done generate don't check awfifo empty 1=mem2ddr/isp2ddr_done generate need until awfifo empty
26	RW	0	isp2comb_enable. 0=Original behaviour, direct-path data go to downstream module; 1=Enable direct-path data go to comb path(RGB/YUV direct path) instead, not go to downstream module.
25	RW	0	isp2ddr_enable. 0=Original behaviour, direct-path data go to downstream module; 1=Enable direct-path data go to memory instead, not go to downstream module. Memory location is defined by CSI2_DDR_START_PIX_B/_ALT.
24	RW	0	rst_on_vs_disable. 0=Reset MEM path FIFOs on Frame_Start; 1=Do not reset on Frame_Start.
20:16	RW	0x1f	enable_packets. [16] enable_raw. 1=Enable receive Raw data; 0=Ignore Raw data. [17] enable_yuv. 1=Enable receive YUV data; 0=Ignore YUV data. [18] enable_rgb. 1=Enable receive RGB data; 0=Ignore RGB data. [19] enable_embedded. 1=Enable receive Embedded data; 0=Ignore. [20] enable_user. 1=Enable receive User-define data; 0=Ignore User-define data.
15	RW	0	isp2ddr_wdone_mode. 0=After FrameEnd, assert interrupt when all previous writes are completed; 1=After (x_end_isp, y_end_isp) pixel are written to DDR, assert interrupt. However if somehow due to error, e.g. less lines are received, so that (x_end_isp, y_end_isp) never happens, then the interrupt will behave the same way as if cfg_

Bit	R/W	Default	Description
			ddr_wdone_mode=0 – assert interrupt after FrameEnd, when all previous writes are completed.
14	RW	0	buffer_ddr_wptr_other. 0=Reports the current ddr_wptr_other status; 1=Reports the buffered ddr_wptr_other status that was buffered at Frame_Start.
13	RW	0	ddr_64byte. HW uses it to calculate stride for Embedded or User-Defined address increment per line-end. 0=DDR is most efficient when address is 16-byte aligned; 1=DDR is most efficient when address is 64-byte aligned.
12	RW	0	mem2ddr_wdone_mode. 0=After FrameEnd, assert interrupt when all previous writes are completed; 1=After (x_end_mem, y_end_mem) pixel are written to DDR, assert interrupt. However if somehow due to error, e.g. less lines are received, so that (x_end_mem, y_end_mem) never happens, then the interrupt will behave the same way as if cfg_ddr_wdone_mode=0 – assert interrupt after FrameEnd, when all previous writes are completed.
11	RW	0	buffer_pic_size. Applicable only to direct path. 0=For active video size status, use non-buffered/instantaneous line/pixel count; 1=For active video size status, use line/pixel count that are buffered at end of field.
10	RW	0	use_null_packet. 0=Ignore Null packet; 1=Regard Null packet as one line.
9	RW	0	inv_field. Applicable only to interlace video to direct path. 0=first field is odd; 1=first field is even.
8	RW	0	interlace_en. Applicable only to direct path. 0=Progressive; 1=Interlace.
7	W	0	ddr_pingpong_init_set. Always read 0. Set to 1 when need to load ddr_pingpong_init_val into effect.
6	RW	0	ddr_pingpong_init_val. Only valid when ddr_pingpong_init_set=1. 0=Very 1st frame use start_addr; 1=Very 1st frame use start_addr_alt.
5	RW	0	ddr_pingpong_en. 0=DDR update with a new start_addr at every Frame_Start; 1=DDR set a pair of start_addr and start_addr_alt at INIT, and then DDR pointer will be ping-ponged at every Frame_Start.
4	RW	0	all_to_mem. 0=Depends on data type, some type might go to direct path, some might go to memory; 1=All data will go to memory, regardless of data type. In DOL2 mode, if isp2ddr_enable=1, direct-path and mem-path will go to different MEM locations; if isp2ddr_enable=0, both paths will go to same MEM location defined by CSI2_DDR_START_PIX/_ALT.
3:0	RW	0	virtual_channel_en: Enable one or more virtual channels at input. Bit[0] 1=Enable virtual channel 0, 0=Disable virtual channel 0; Bit[1] 1=Enable virtual channel 1, 0=Disable virtual channel 1; Bit[2] 1=Enable virtual channel 2, 0=Disable virtual channel 2; Bit[3] 1=Enable virtual channel 3, 0=Disable virtual channel 3;

Table 7-262 CSI2\_GEN\_CTRL1

Bit	R/W	Default	Description
31:8	R	0	Reserved
7:6	RW	3	din_byte3_sel: Re-map the received 32-bit data's byte3. 0=Use input data's byte0 as actual byte3; 1=Use input data's byte1 as actual byte3; 2=Use input data's byte2 as actual byte3; 3=Use input data's byte3 as actual byte3.
5:4	RW	2	din_byte2_sel: Re-map the received 32-bit data's byte2. 0=Use input data's byte0 as actual byte2; 1=Use input data's byte1 as actual byte2; 2=Use input data's byte2 as actual byte2; 3=Use input data's byte3 as actual byte2.
3:2	RW	1	din_byte1_sel: Re-map the received 32-bit data's byte1. 0=Use input data's byte0 as actual byte1; 1=Use input data's byte1 as actual byte1; 2=Use input data's byte2 as actual byte1; 3=Use input data's byte3 as actual byte1.
1:0	RW	0	din_byte0_sel: Re-map the received 32-bit data's byte0. 0=Use input data's byte0 as actual byte0; 1=Use input data's byte1 as actual byte0; 2=Use input data's byte2 as actual byte0; 3=Use input data's byte3 as actual byte0.

Table 7-263 CSI2\_X\_START\_END\_ISP

Bit	R/W	Default	Description
31:16	RW	0xffff	x_end_isp. Applicable only to Raw data, direct ISP path.
15:0	RW	0	x_start_isp . Applicable only to Raw data, direct ISP path.

Table 7-264 CSI2\_Y\_START\_END\_ISP

Bit	R/W	Default	Description
31:16	RW	0xffff	y_end_isp. Applicable only to Raw data, direct ISP path.
15:0	RW	0	y_start_isp . Applicable only to Raw data, direct ISP path.

Table 7-265 CSI2\_X\_START\_END\_MEM

Bit	R/W	Default	Description
31:16	RW	0xffff	x_end_mem. Applicable only to Raw data, direct MEM path.
15:0	RW	0	x_start_mem. Applicable only to Raw data, MEM path.

Table 7-266 CSI2\_Y\_START\_END\_MEM

Bit	R/W	Default	Description
31:16	RW	0xffff	y_end_mem. Applicable only to Raw data, direct MEM path.
15:0	RW	0	y_start_mem. Applicable only to Raw data, MEM path.

Table 7-267 CSI2\_VC\_MODE

Bit	R/W	Default	Description
31:28	RW	0xf	vs_oth_sel_vc. The VS signal for non-pix-based data, is the logic operation of  (VS[all_4_channel_pre_reassign] & vs_oth_sel_vc[3:0]).
27:24	RW	0xf	vs_mem_sel_vc. The VS signal that goes to MEM path, is the logic operation of  (VS[all_4_channel_pre_reassign] & vs_mem_sel_vc[3:0]).
23:20	RW	0xf	hs_isp_sel_vc. The HS signal that goes to direct path, is the logic operation of  (HS[all_4_channel_pre_reassign] & hs_isp_sel_vc[3:0]).
19:16	RW	0xf	vs_isp_sel_vc. The VS signal that goes to direct path, is the logic operation of  (VS[all_4_channel_pre_reassign] & vs_isp_sel_vc[3:0]).
15:14	RW	0	vc_mode3[3]. Applicable only to Raw/RGB/YUV data.
13:12	RW	0	vc_mode3[2]. Applicable only to Raw/RGB/YUV data.
11:10	RW	0	vc_mode3[1]. Applicable only to Raw/RGB/YUV data.
9:8	RW	0	vc_mode3[0]. Applicable only to Raw/RGB/YUV data.
7:6	RW	0	vc_to_isp. Applicable only to Raw/RGB/YUV data. After VC re-assignment (if applicable), if the input data's VC==vc_to_isp, then the data go to direct path, otherwise it go to MEM path.
5:4	RW	0	vc_val1. Applicable only to Raw/RGB/YUV data.
3:2	RW	0	vc_val0. Applicable only to Raw/RGB/YUV data.
1:0	RW	0	assign_vc_mode. Applicable only to Raw/RGB/YUV data. 0=No virtual_channel re-assign. 1=Even line's VC=vc_val0, Odd line's VC=vc_val1. 2=If the first 64-bit (Payload   vc_mode2_match_mask_A) == (vc_mode2_match_A   vc_mode2_match_mask_A), this line's VC=vc_val0; If the first 64-bit (Payload   vc_mode2_match_mask_B) == (vc_mode2_match_B   vc_mode2_match_mask_B), this line is VC=vc_val1; Otherwise this line is to be ignored. 3=Reassign this line's VC to vc_mode3[VC].

Table 7-268 CSI2\_VC\_MODE2\_MATCH\_MASK\_A\_L

Bit	R/W	Default	Description
31:0	RW	0	vc_mode2_match_mask_A[31:0]. Refer to assign_vc_mode.

Table 7-269 CSI2\_VC\_MODE2\_MATCH\_MASK\_A\_H

Bit	R/W	Default	Description
31:0	RW	0	vc_mode2_match_mask_A[63:32]. Refer to assign_vc_mode.

Table 7-270 CSI2\_VC\_MODE2\_MATCH\_A\_L

Bit	R/W	Default	Description
31:0	RW	0	vc_mode2_match_A[31:0]. Refer to assign_vc_mode.

**Table 7-271 CSI2\_VC\_MODE2\_MATCH\_A\_H**

Bit	R/W	Default	Description
31:0	RW	0	vc_mode2_match_A[63:32]. Refer to assign_vc_mode.

**Table 7-272 CSI2\_VC\_MODE2\_MATCH\_B\_L**

Bit	R/W	Default	Description
31:0	RW	0	vc_mode2_match_B[31:0]. Refer to assign_vc_mode.

**Table 7-273 CSI2\_VC\_MODE2\_MATCH\_B\_H**

Bit	R/W	Default	Description
31:0	RW	0	vc_mode2_match_B[63:32]. Refer to assign_vc_mode.

**Table 7-274 CSI2\_DDR\_START\_PIX**

Bit	R/W	Default	Description
31:0	RW	0	DDR start address for pixel-based data (Raw/RGB/YUV). Must be 16-byte aligned, so low 4-bit will be ignored. For MEM path.

**Table 7-275 CSI2\_DDR\_START\_PIX\_ALT**

Bit	R/W	Default	Description
31:0	RW	0	Alternative DDR start address for pixel-based data (Raw/RGB/YUV). Refer to ddr_pingpong_en. Must be 16-byte aligned, so low 4-bit will be ignored. For MEM path.

**Table 7-276 CSI2\_DDR\_STRIDE\_PIX**

Bit	R/W	Default	Description
31:0	RW	0	DDR line stride for pixel-based data (Raw/RGB/YUV). Must be 16-byte aligned, so low 4-bit will be ignored. For MEM path.

**Table 7-277 CSI2\_DDR\_START\_OTHER**

Bit	R/W	Default	Description
31:0	RW	0	DDR start address for non-pixel-based data (Embedded or User-defined). Must be 16-byte aligned, so low 4-bit will be ignored.

Table 7-278 CSI2\_DDR\_START\_OTHER\_ALT

Bit	R/W	Default	Description
31:0	RW	0	Alternative DDR start address for non-pixel-based data (Embedded or User-defined). Refer to ddr_pingpong_en. Must be 16-byte aligned, so low 4-bit will be ignored.

Table 7-279 CSI2\_DDR\_MAX\_BYTES\_OTHER

Bit	R/W	Default	Description
31:0	RW	0	Maximum bytes allocated to DDR for storing non-pixel-based data (Embedded or User-defined).

**Note**

About reading Embedded or User-defined packets from DDR The first packet starts from CSI2\_DDR\_START\_OTHER(\_ALT). Byte0={virtual\_channel[1:0],data\_type[5:0]}; Byte1=word\_count[7:0]; Byte2=word\_count[15:8]; The rest of the bytes are packet payload bytes, total bytes of payload is word\_count, with lowest byte first, highest byte last. SW calculates where to read the next packet, (if there is any):  $\text{new\_packet\_start\_addr} = \text{last\_packet\_start\_addr} + (((\text{last\_packet\_word\_count} + 3) + M - 1) / M) * M$  Where M is defined by CSI2\_GEN\_CTRL0's bit[13] – ddr\_64byte. If  $\text{cfg\_ddr\_64byte} = 0$ ,  $M = 16$ ; if  $\text{ddr\_64byte} = 1$ ,  $M = 64$ . If the new\_packet\_start\_addr  $\geq$  the value read from CSI2\_DDR\_WPTR\_STAT\_OTHER. Then there is no more packets. Do not read the content starting from new\_packet\_start\_addr. CSI2\_DDR\_WPTR\_STAT\_OTHER's value can be current address pointer or previous address pointer buffered by Frame\_Start.

Table 7-280 CSI2\_INTERRUPT\_CTRL\_STAT

Bit	R/W	Default	Description
31:22	R	0	Reserved
21:16	RW	0	interrupt status/clear. [16] vs_rise_isp sticky status, write 1 to clear it; [17] vs_fall_isp sticky status, write 1 to clear it; [18] mem2ddr_wdone sticky status, write 1 to clear it; [19] vs_rise_mem sticky status, write 1 to clear it; [20] vs_fall_mem sticky status, write 1 to clear it; [21] isp2ddr_wdone sticky status, write 1 to clear it.
15:6	R	0	Reserved
5:0	RW	0	interrupt_sel. Enable one or more interrupts. [0]=1 to enable vs_rise_isp interrupt; [1]=1 to enable vs_fall_isp interrupt; [2]=1 to enable mem2ddr_wdone interrupt; [3]=1 to enable vs_rise_mem interrupt; [4]=1 to enable vs_fall_mem interrupt; [5]=1 to enable isp2ddr_wdone interrupt.

Table 7-281 CSI2\_VC\_MODE2\_MATCH\_MASK\_B\_L

Bit	R/W	Default	Description
31:0	RW	0	vc_mode2_match_mask_B[31:0]. Refer to assign_vc_mode.



**Table 7-282 CSI2\_VC\_MODE2\_MATCH\_MASK\_B\_H**

Bit	R/W	Default	Description
31:0	RW	0	vc_mode2_match_mask_B[63:32]. Refer to assign_vc_mode.

**Table 7-283 CSI2\_DDR\_LOOP\_LINES\_PIX**

Bit	R/W	Default	Description
31:16	RW	0xffff	isp2ddr_loop_lines_pix. For direct path. DDR address increment (isp2ddr_loop_lines_pix+1) lines, before loops back to start address.
15:0	RW	0xffff	mem2ddr_loop_lines_pix. For MEM path. DDR address increment (mem2ddr_loop_lines_pix+1) lines, before loops back to start address.

**Table 7-284 CSI2\_GEN\_STAT0**

Bit	R/W	Default	Description
31:10	R	0	Reserved
9	R	0	afifo_nempty.
8	R	0	afifo_full.
7:6	R	0	Reserved
5:0	R	0	afifo_count. The counter of async FIFO at input.

**Table 7-285 CSI2\_ERR\_STAT0**

Bit	R/W	Default	Description
31:15	R	0	Reserved
14	R	0	err_axi_overdue_oth
13	R	0	err_axi_overdue_isp
12	R	0	err_addr_other_overfl
11	R	0	err_axi_overdue_mem
10	R	0	err_bfifo_underfl
9	R	0	err_bfifo_overfl
8	R	0	err_axi_bresp
7	R	0	err_axi_bresp_extra
6	R	0	err_wfifo_underfl
5	R	0	err_wfifo_overfl
4	R	0	err_awfifo_underfl
3	R	0	err_awfifo_overfl
2	R	0	err_afifo_ovfl

Bit	R/W	Default	Description
1	R	0	err_wc_ovfl
0	R	0	err_wc_unfl

**Table 7-286 CSI2\_PIC\_SIZE\_STAT**

Bit	R/W	Default	Description
31:16	R	0	line_count measure for direct path only. Maybe buffered or non-buffered, depends on buffer_pic_size.
15:0	R	0	pix_count measure for direct path only. Maybe buffered or non-buffered, depends on buffer_pic_size.

**Table 7-287 CSI2\_DDR\_WPTR\_STAT\_PIX**

Bit	R/W	Default	Description
31:0	R	0	Memory pointer for pixel lines. Always current status. For MEM path.

**Table 7-288 CSI2\_DDR\_WPTR\_STAT\_OTHER**

Bit	R/W	Default	Description
31:0	R	0	Memory pointer for non-pixel lines. Can be buffered or current status depending on buffer_ddr_wptr_other.

**Table 7-289 CSI2\_STAT\_MEM\_0**

Bit	R/W	Default	Description
31:20	R	0	Reserved
19:10	R	0	wfifo_count.
9	R	0	axi_w_state. 0=AXI_W_IDLE; 1=AXI_W_WAIT_READY
8:2	R	0	awfifo_cnt.
1:0	R	0	axi_aw_state. 0=AXI_AW_IDEL; 1= AXI_AW_WAIT_READY; 2= AXI_AW_WAIT_W.

**Table 7-290 CSI2\_STAT\_MEM\_1**

Bit	R/W	Default	Description
31:16	R	0	aw_pending_cnt_mem.
15:0	R	0	y_pos_mem. Current line number, for MEM path's RAW/RGB/YUV data only.

**Table 7-291 CSI2\_STAT\_MEM\_2**

Bit	R/W	Default	Description
31:16	R	0	aw_pending_cnt_isp.
15:0	R	0	y_pos_isp. Current line number, for ISP path's RAW/RGB/YUV data only.

**Table 7-292 CSI2\_STAT\_GEN\_SHORT\_08**

Bit	R/W	Default	Description
18	R	0	General Short Packet (type=0x08) receive flag. The flag will be cleared once the register is read. 0=Not received; 1=Received new packet.
17:16	R	0	Virtual Channel for General Short Packet (type=0x08).
15:0	R	0	Received General Short Packet (type=0x08).

**Table 7-293 CSI2\_STAT\_GEN\_SHORT\_09**

Bit	R/W	Default	Description
18	R	0	General Short Packet (type=0x09) receive flag. The flag will be cleared once the register is read. 0=Not received; 1=Received new packet.
17:16	R	0	Virtual Channel for General Short Packet (type=0x09).
15:0	R	0	Received General Short Packet (type=0x09).

**Table 7-294 CSI2\_STAT\_GEN\_SHORT\_0A**

Bit	R/W	Default	Description
18	R	0	General Short Packet (type=0x0A) receive flag. The flag will be cleared once the register is read. 0=Not received; 1=Received new packet.
17:16	R	0	Virtual Channel for General Short Packet (type=0x0A).
15:0	R	0	Received General Short Packet (type=0x0A).

**Table 7-295 CSI2\_STAT\_GEN\_SHORT\_0B**

Bit	R/W	Default	Description
18	R	0	General Short Packet (type=0x0B) receive flag. The flag will be cleared once the register is read. 0=Not received; 1=Received new packet.
17:16	R	0	Virtual Channel for General Short Packet (type=0x0B).
15:0	R	0	Received General Short Packet (type=0x0B).

**Table 7-296 CSI2\_STAT\_GEN\_SHORT\_0C**

Bit	R/W	Default	Description
18	R	0	General Short Packet (type=0x0C) receive flag. The flag will be cleared once the register is read. 0=Not received; 1=Received new packet.
17:16	R	0	Virtual Channel for General Short Packet (type=0x0C).
15:0	R	0	Received General Short Packet (type=0x0C).

**Table 7-297 CSI2\_STAT\_GEN\_SHORT\_0D**

Bit	R/W	Default	Description
18	R	0	General Short Packet (type=0x0D) receive flag. The flag will be cleared once the register is read. 0=Not received; 1=Received new packet.
17:16	R	0	Virtual Channel for General Short Packet (type=0x0D).
15:0	R	0	Received General Short Packet (type=0x0D).

**Table 7-298 CSI2\_STAT\_GEN\_SHORT\_0E**

Bit	R/W	Default	Description
18	R	0	General Short Packet (type=0x0E) receive flag. The flag will be cleared once the register is read. 0=Not received; 1=Received new packet.
17:16	R	0	Virtual Channel for General Short Packet (type=0x0E).
15:0	R	0	Received General Short Packet (type=0x0E).

**Table 7-299 CSI2\_STAT\_GEN\_SHORT\_0F**

Bit	R/W	Default	Description
18	R	0	General Short Packet (type=0x0F) receive flag. The flag will be cleared once the register is read. 0=Not received; 1=Received new packet.
17:16	R	0	Virtual Channel for General Short Packet (type=0x0F).
15:0	R	0	Received General Short Packet (type=0x0F).

**Table 7-300 CSI2\_STAT\_TYPE\_RCVD\_L**

Together with CSI2\_STAT\_TYPE\_RCVD\_H, to form 64-bit flags to record type of packets that have been received. Write 1 to each bit to clear the corresponding flag for each type.

Bit	R/W	Default	Description
31	RW	0	If read back 1, DataType=31 has been received. Write 1 to this bit to clear the receive flag.
30:2	RW	0	Same as the rest...
1	RW	0	If read back 1, DataType=1 has been received. Write 1 to this bit to clear the receive flag.
0	RW	0	If read back 1, DataType=0 has been received. Write 1 to this bit to clear the receive flag.

**Table 7-301 CSI2\_STAT\_TYPE\_RCVD\_H**

Bit	R/W	Default	Description
31	RW	0	If read back 1, DataType=63 has been received. Write 1 to this bit to clear the receive flag.
30:2	RW	0	Same as the rest...
1	RW	0	If read back 1, DataType=33 has been received. Write 1 to this bit to clear the receive flag.
0	RW	0	If read back 1, DataType=32 has been received. Write 1 to this bit to clear the receive flag.

**Table 7-302 CSI2\_DDR\_START\_PIX\_B**

Bit	R/W	Default	Description
31:0	RW	0	DDR start address for pixel-based data (Raw/RGB/YUV). Must be 16-byte aligned, so low 4-bit will be ignored. For direct path.

**Table 7-303 CSI2\_DDR\_START\_PIX\_B\_ALT**

Bit	R/W	Default	Description
31:0	RW	0	Alternative DDR start address for pixel-based data (Raw/RGB/YUV). Refer to ddr_pingpong_en. Must be 16-byte aligned, so low 4-bit will be ignored. For direct path.

**Table 7-304 CSI2\_DDR\_STRIDE\_PIX\_B**

Bit	R/W	Default	Description
31:0	RW	0	DDR line stride for pixel-based data (Raw/RGB/YUV). Must be 16-byte aligned, so low 4-bit will be ignored. For direct path.

**Table 7-305 CSI2\_DDR\_WPTR\_STAT\_PIX\_B**

Bit	R/W	Default	Description
31:0	R	0	Memory pointer for pixel lines. Always current status. For direct path.

**Table 7-306 CSI2\_STAT\_MEM\_3**

Bit	R/W	Default	Description
31:16	R	0	aw_pending_cnt_oth.
15:0	R	0	Reserved

**Table 7-307 CSI2\_AXI\_UGT\_CNTL0**

Bit	R/W	Default	Description
31	RW	0	reg_ugt_auto_en 1.frontend axi auto urgent enable
30:16	RW	0	reg_ugt_th1 axi urgent change to 0,when wfifo_cnt < th1
14:0	RW	0	reg_ugt_th0 axi urgent change to 1,when wfifo_cnt >= th0 NOTE: th0 >= th1

**Table 7-308 CSI2\_AXI\_UGT\_CNTL1**

Bit	R/W	Default	Description
31	RW	0	reg_sideband_auto_en 1.frontend axi auto sideband enable
30:16	RW	0	reg_sideband_th1 axi sideband change to 0,when wfifo_cnt < th1
14:0	RW	0	reg_sideband_th0 axi sideband change to 1,when wfifo_cnt >= th0 NOTE: th0 >= th1

**Table 7-309 CSI2\_AXI\_UGT\_ST**

Bit	R/W	Default	Description
17	R	0	axi_sideband value
16	R	0	axi_ugt value
15:0	R	0	fifo_count

**Table 7-310 CSI2\_LINE\_SUP\_CNTL0**

Bit	R/W	Default	Description
31:20	RW	0	src_err_mem_en fifo err define in CSI2_ERR_STAT0,can trigger sub_overflow when each bit set to 1
19	RW	0	both_ov_en 1:will set both isp2ddr path and mem2ddr path sub_overflow
18	RW	0	soft_rst set 1 to reset this module
17	RW	0	line_sup_sel 1:line supply on mem2ddr path 0:line supply on isp2ddr path

Bit	R/W	Default	Description
16	RW	0	line_sup_en line supply enable
15	RW	0	line_sup_vs_sel 1:use mem2ddr vsync to sync 0:use isp2ddr vsync to sync
14	RW	0	ovflow_st_clr set 1 to clear overflow status
13:0	RW	0	line_min_num min data number per each line,N-1 Byte

Table 7-311 CSI2\_LINE\_SUP\_ST

Bit	R/W	Default	Description
20	R	0	r_fifo_ov_st
19:18	R	0	r_sup_din_bytes_en
17	R	0	r_fifo_rd_mask
16	R	0	r_fifo_ov
15:0	R	0	r_pixel_cnt

Table 7-312 MIPI\_ADAPT\_DDR\_RD0\_CNTL0 0x00

Bit(s)	R/W	Default	Description
31	RW	0x0	frame_rd_start start read frame data from ddr, used in single frame read scenario, this bit is self-cleared
29:28	RW	0x0	burst_type burst type for AXI read command 0: single 1: incr2 2: incr4 3: inc8
27:26	RW	0x0	reg_sample_sel By default, control register will be latched from apb clk domain to vclk domain by enable and soft_load, you can use the following signal to latch the control register too 0: at posedge of frame_rd_start 1: din_vs,isp path vsync 2: fe_line_wr_vs,ddr path vsync 3: reserved
25	RW	0x0	soft_load control register will be latched from apb clk domain to vclk domain at posedge of soft_load
24	RW	0x0	continue_mode Used in ddr mode, if this bit is high, continuous frames will be read from ddr without any software control in every vblank period 0: single frame scenario 1: multiple frame scenario
23	RW	0x0	dol_mode 0: single exposure mode 1: DOL mode for HDR, 2-exposure mode

Bit(s)	R/W	Default	Description
22	RW	0x0	pingpong_mode This bit is used for frame base address control. 0: frame_base_address_0 is used 1: frame_base_address_0 and frame_base_address_1 are used alternately
21	RW	0x0	dol_time_out_en Enable for the timeout control in DOL mode. 0: disable 1: enable
18	RW	0x0	dol_frame_end_en Enable ddr_rd module read the missing lines when frontend module write less lines than expectation, only used in DOL mode
17	RW	0x0	dol_vblank_clr_sel In DOL mode, vblank flag is set by frame write done flag from frontend module, and clear by the following signal: 0: ddr_rd has read all the lines from ddr for a frame 1: vsync from frontend module
15:14	RW	0x0	dol_time_tick_sel don_timeout count tick 0:vclk 1:xtal3_tick 2:1us_tick 3:10us_tick
13:4	RW	0x0	line_stride mem stride for a line, unit: 128bit
3:2	RW	0x0	lbuf_soft_rst_sel Line buffer 0 soft reset source select 0: reserved 1: din_vs 2: fe_line_wr_vs 3: frame_vs_rst
1	RW	0x0	frame loop mode enable
0	RW	0x0	Module enable for ddr_rd

Table 7-313 MIPI\_ADAPT\_DDR\_RD0\_CNTL1 0x01

Bit(s)	R/W	Default	Description
31:30	RW	0x0	port_sel 0: direct path mode, data from direct path input with RAW 1: ddr mode, data from ddr with RAW/YUV/RGB 2: direct comb mode, data from direct path input with RGB/YUV
29	RW	0x0	frame_vs_rst_en reset at frame_vs, frame_vs sel by frame_vs_rst_sel
28:16	RW	0x0	line_number Line number in a frame, when N, set N
15:12	RW	0x0	dol_line_threshold In DOL mode, ddr_rd start to read frame data from ddr when frontend module has written "dol_line_threshold" lines into ddr
11:10	RW	0x0	frame_vs_rst_sel 0: din_vs 1: fe_line_wr_vs 2: din_vs & lbuf_overflow 3: fe_line_wr_vs & lbuf_overflow
9:0	RW	0x0	line_size Data size for a line, unit: 128bit



Table 7-314 MIPI\_ADAPT\_DDR\_RD0\_CNTL2 0x02

Bit(s)	R/W	Default	Description
31:0	RW	0x0	frame_base_address_0 Base address 0 for frame data

Table 7-315 MIPI\_ADAPT\_DDR\_RD0\_CNTL3 0x03

Bit(s)	R/W	Default	Description
31:0	RW	0x0	frame_base_address_1 Base address 1 for frame data, valid in pingpong mode

Table 7-316 MIPI\_ADAPT\_DDR\_RD0\_CNTL4 0x04

Bit(s)	R/W	Default	Description
31:13	RW	0x0	dol_time_out Set the IDLE period in vck cycle for frontend-ddr_rd interface to set the time out flag, valid in DOL mode
12:0	RW	0x0	line_threshold line number threshold for interrupt, when ddr_rd has read "line_threshold" lines, it will send out an interrupt flag

Table 7-317 MIPI\_ADAPT\_DDR\_RD0\_CNTL5 0x05

Bit(s)	R/W	Default	Description
31:0	RW	0x0	frame_end_address_0 end address 0 for frame data in loop mode

Table 7-318 MIPI\_ADAPT\_DDR\_RD0\_CNTL6 0x06

Bit(s)	R/W	Default	Description
31:0	RW	0x0	frame_end_address_1 end address 1 for frame data in loop mode

Table 7-319 MIPI\_ADAPT\_DDR\_RD0\_UGT\_CNTL0 0x07

Bit(s)	R/W	Default	Description
31	RW	0	reg_ugt_auto_en 1. axi auto urgent enable
30:16	RW	0	reg_ugt_th1 axi urgent change to 0,when wfifo_cnt >= th1
14:0	RW	0	reg_ugt_th0 axi urgent change to 1,when wfifo_cnt < th0 NOTE: th0 >= th1

**Table 7-320 MIPI\_ADAPT\_DDR\_RD0\_UGT\_CNTL1 0x08**

Bit(s)	R/W	Default	Description
31	RW	0	reg_sideband_auto_en 1:axi auto sidebandenable
30:16	RW	0	reg_ugt_th1 axi sideband change to 0,when wfifo_cnt >= th1
14:0	RW	0	reg_ugt_th0 axi sideband change to 1,when wfifo_cnt < th0 NOTE: th0 >= th1

**Table 7-321 MIPI\_ADAPT\_DDR\_RD0\_ST0 0x0a**

Bit(s)	R/W	Default	Description
31	R	0x0	pixel_line_valid Valid line period for direct path
19:8	R	0x0	lbuf_counter Valid data number in line buffer
7:4	R	0x0	fe_frame_wr_counter Frame numbers present in DDR for read
3	R	0x0	lbuf_not_full full flag for line buffer 0: full 1: not full
2:0	R	0x0	ddr_rd_state Internal FSM state for ddr read control

**Table 7-322 MIPI\_ADAPT\_DDR\_RD0\_ST1 0x0b**

Bit(s)	R/W	Default	Description
31	R	0x0	frame_base_address_sel Indicate which base address is used for current frame 0: frame_base_address_0 1: frame_base_address_1
28:16	R	0x0	v_counter Line number that has been read from ddr for a frame
11:0	R	0x0	line_count data number that has been read from ddr for a line

**Table 7-323 MIPI\_ADAPT\_DDR\_RD0\_ST2 0x0c**

Bit(s)	R/W	Default	Description
29	R	0x0	fe_line_wr_done_mask Indicate frontend has written more lines than expectation to ddr for a frame
28:16	R	0x0	fe_line_wr_counter Valid Line number present in ddr, valid in DOL mode
13	R	0x0	dol_timeout_valid Time out flag in DOL mode
12:0	R	0x0	dol_frame_line_cnt Line number that frontend has written to ddr, valid in DOL mode

**Table 7-324 MIPI\_ADAPT\_DDR\_RD0\_UGT\_ST 0x0d**

Bit(s)	R/W	Default	Description
29	R	0x0	axi_sideband value
28:16	R	0x0	axi_ugt value
13	R	0x0	fifo_count

**Table 7-325 DDR\_RD0\_LBUF\_STATUS 0x7a**

Bit(s)	R/W	Default	Description
19	R	0x0	ddr_rd_0_lbuf_overflow
18	R	0x0	ddr_rd_0_lbuf_underflow
17	R	0x0	ddr_rd_0_lbuf_full
16	R	0x0	ddr_rd_0_lbuf_halffull
15	R	0x0	ddr_rd_0_lbuf_empty
12:0	R	0x0	ddr_rd_0_lbuf_count

MIPI\_ADAPT\_DDR\_RD1\_CNTL0 0x10 Same as MIPI\_ADAPT\_DDR\_RD0\_CNTL0

MIPI\_ADAPT\_DDR\_RD1\_CNTL1 0x11 Same as MIPI\_ADAPT\_DDR\_RD0\_CNTL1

MIPI\_ADAPT\_DDR\_RD1\_CNTL2 0x12 Same as MIPI\_ADAPT\_DDR\_RD0\_CNTL2

MIPI\_ADAPT\_DDR\_RD1\_CNTL3 0x13 Same as MIPI\_ADAPT\_DDR\_RD0\_CNTL3

MIPI\_ADAPT\_DDR\_RD1\_CNTL4 0x14 Same as MIPI\_ADAPT\_DDR\_RD0\_CNTL4

MIPI\_ADAPT\_DDR\_RD1\_CNTL5 0x15 Same as MIPI\_ADAPT\_DDR\_RD0\_CNTL5

MIPI\_ADAPT\_DDR\_RD1\_CNTL6 0x16 Same as MIPI\_ADAPT\_DDR\_RD0\_CNTL6

MIPI\_ADAPT\_DDR\_RD1\_UGT\_CNTL0 0x17 Same as MIPI\_ADAPT\_DDR\_RD0\_UGT\_CNTL0

MIPI\_ADAPT\_DDR\_RD1\_UGT\_CNTL1 0x18 Same as MIPI\_ADAPT\_DDR\_RD0\_UGT\_CNTL1

MIPI\_ADAPT\_DDR\_RD1\_ST0 0x1a Same as MIPI\_ADAPT\_DDR\_RD0\_ST0

MIPI\_ADAPT\_DDR\_RD1\_ST1 0x1b Same as MIPI\_ADAPT\_DDR\_RD0\_ST1

MIPI\_ADAPT\_DDR\_RD1\_ST2 0x1c Same as MIPI\_ADAPT\_DDR\_RD0\_ST2

MIPI\_ADAPT\_DDR\_RD0\_UGT\_ST 0x1d Same as MIPI\_ADAPT\_DDR\_RD1\_UGT\_ST

DDR\_RD1\_LBUF\_STATUS 0x7b same as DDR\_RD0\_LBFU\_STATUS

**Table 7-326 MIPI\_ADAPT\_PIXEL0\_CNTL0 0x20**

Bit(s)	R/W	Default	Description
31	RW	0x0	start_en Start enable,write 1 to start ,will auto clear to 0
30	RW	0x0	soft_load reload register value, will auto clear to 0
25:20	RW	0x0	data_type 6'h1e:YUV422 8BIT

Bit(s)	R/W	Default	Description
			6'h1f:YUV422 10BIT 6'h20:RGB444 6'h21:RGB555 6'h22:RGB565 6'h23:RGB666 6'h24:RGB888 6'h28:RAW6 6'h29:RAW7 6'h2a:RAW8 6'h2b:RAW10 6'h2c:RAW12 6'h2d:RAW14
17:16	RW	0x0	work mode 0:RAW from ddr path 1:RAW from direct path 2:YUV/RGB from ddr path 3:YUV/RGB from direct path
14	RW	0x0	color_expand 0=Expand less than 10-bit component to 10-bit, by appending 0; 1=Expand less than 10-bit component to 10-bit, by appending MSBs.
12:0	RW	0x0	fifo_threshold start work after lbuf valid level >= fifo_threshold value

Table 7-327 MIPI\_ADAPT\_PIXEL0\_CNTL1 0x21

Bit(s)	R/W	Default	Description
31:16	RW	0x0	pixel_x_start
15:0	RW	0xffff	pixel_x_end

Table 7-328 MIPI\_ADAPT\_PIXEL0\_CNTL2 0x22

Bit(s)	R/W	Default	Description
27:15	RW	0x0	pixel_num pixel number per line, set N
9:0	RW	0x0	fifo_rd_size 128bit number per line ,set N

Table 7-329 MIPI\_ADAPT\_PIXEL0\_CNTL3 0x23

Bit(s)	R/W	Default	Description
31	RW	0x0	frame_sync_delay_en add delay to frame_sync
30:16	RW	0x0	frame_sync_delay_time
2	RW	0x0	frame_sync_en frame_sync enable
1	RW	0x0	frame_sync_rst_en use frame_sync reset module
0	RW	0x0	frame_sync_load_en use frame_sync reload registers

**Table 7-330 MIPI\_ADAPT\_PIXEL0\_ST0 0x2a**

Bit(s)	R/W	Default	Description
31:30	R	0x0	r_pixel_st
29	R	0x0	r_pixel_dsr
28:16	R	0x0	r_ddr_mode_pixel_set_cnt
15	R	0x0	fifo_rd_start
14	R	0x0	trans_start
13	R	0x0	r_last_pixel_finish
12	R	0x0	fifo_rd_pixel_mask
11:10	R	0x0	pixel_fifo_count_v
9:0	R	0x0	fifo_rd_cnt

**Table 7-331 MIPI\_ADAPT\_PIXEL0\_ST1 0x2b**

Bit(s)	R/W	Default	Description
31:30	R	0x0	pixel_fifo_count
29:28	R	0x0	r_buf_step_cnt
24:16	R	0x0	r_data_buf_cnt
14:8	R	0x0	r_info_buf_cnt
6:0	R	0x0	r_data_buf_remain_cnt

MIPI\_ADAPT\_PIXEL1\_CNTL0 0x30

Same as MIPI\_ADAPT\_PIXEL0\_CNTL0

MIPI\_ADAPT\_PIXEL1\_CNTL1 0x31

Same as MIPI\_ADAPT\_PIXEL0\_CNTL1

MIPI\_ADAPT\_PIXEL1\_CNTL2 0x32

Same as MIPI\_ADAPT\_PIXEL0\_CNTL2

MIPI\_ADAPT\_PIXEL1\_CNTL3 0x33

Same as MIPI\_ADAPT\_PIXEL0\_CNTL3

MIPI\_ADAPT\_PIXEL1\_ST0 0x3a

Same as MIPI\_ADAPT\_PIXEL0\_ST0

MIPI\_ADAPT\_PIXEL1\_ST1 0x3b

Same as MIPI\_ADAPT\_PIXEL0\_ST0

**Table 7-332 MIPI\_ADAPT\_ALIG\_CNTL0 0x40**

Bit(s)	R/W	Default	Description
31:16	RW	0x0	h_total_num H total in a line,when N,set N
15:0	RW	0x0	v_total_num V total in a frame,when N,set N

**Table 7-333 MIPI\_ADAPT\_ALIG\_CNTL1 0x41**

Bit(s)	R/W	Default	Description
31:16	RW	0x0	hpe_num End pixel in a line,when N,set N
15:0	RW	0x0	hps_num Start pixel in a line,when N,set N+1

**Table 7-334 MIPI\_ADAPT\_ALIG\_CNTL2 0x42**

Bit(s)	R/W	Default	Description
31:16	RW	0x0	vpe_num End line in a frame,when N,set N
15:0	RW	0x0	vps_num Start line in a frame,when N,set N+1

**Table 7-335 MIPI\_ADAPT\_ALIG\_CNTL3 0x43**

Bit(s)	R/W	Default	Description
31:16	RW	0x0	frame_start_line Will valid ,When frame_sync_en =1 Can start at N line of frame at the time of frame_sync Should not greater than the vps_num(first valid line position)
15:0	RW	0x0	frame_start_pixel Will valid ,When frame_sync_en =1 Can start at N pixel of frame_start_line at the time of frame_sync Should be less than the h_num(h total)

**Table 7-336 MIPI\_ADAPT\_ALIG\_CNTL4 0x44**

Bit(s)	R/W	Default	Description
29:0	RW	0x0	fill_data Fill data value when less line or less pixel in direct path,will use this value to fill the missing pixel

Table 7-337 MIPI\_ADAPT\_ALIG\_CNTL5 0x45

Bit(s)	R/W	Default	Description
31:20	RW	0x0	flow_done_src_en generate frame_end irq until source done_irq bit31:isp_core bit30:tnr_de_rd bit29:tnr_de_wr bit28:adapt_de_rd bit27:adapt_de_wr bit26:md2 bit25:flicker bit24:md bit23:ds2 bit22:ds1 bit21:ds0 bit20:crop mif
1	RW	0x0	frame_end_int_final_sel 1:frame_end_int use frame_end,last vblank cycle 0:frame_end_int use frame_last ,last valid pixel cycle
0	RW	0x0	dst_hold_gate set 1 to disable auto clk gate ,include source hold

Table 7-338 MIPI\_ADAPT\_ALIG\_CNTL6 0x46

Bit(s)	R/W	Default	Description
31	RW	0x0	yuvrgb_mode need set 1 when input data format is YUV/RGB
31	RW	0x0	frame_end_load set 1 will auto load register value at frame_en
15	RW	0x0	vdata3_en 0:vdata3 disable 1:vdata3 enable
14	RW	0x0	vdata2_en 0:vdata2 disable 1:vdata2 enable
13	RW	0x0	vdata1_en 0:vdata1 disable 1:vdata1 enable
12	RW	0x0	vdata0_en 0:vdata0 disable 1:vdata0 enable
11	RW	0x0	vdata3_sel When camera_mode = 1,this bit replaced by camera_sel When camera_mode = 0,vdata3 select 0:lane0 data 1:lane1 data
10	RW	0x0	vdata2_sel When camera_mode = 1,this bit replaced by camera_sel When camera_mode = 0,vdata2 select 0:lane0 data 1:lane1 data
9	RW	0x0	vdata1_sel When camera_mode = 1,this bit replaced by camera_sel When camera_mode = 0,vdata1 select 0:lane0 data 1:lane1 data

Bit(s)	R/W	Default	Description
8	RW	0x0	vdata0_sel When camera_mode = 1, this bit replaced by camera_sel When camera_mode = 0, vdata0 select 0: lane0 data 1: lane1 data
5	RW	0x0	pix_datamode_1 Pixel 1 data mode 0 ddr mode 1 direct mode
4	RW	0x0	pix_datamode_0 Pixel 0 data mode 0 ddr mode 1 direct mode
3	RW	0x0	0 lane1_sel Lane1 input source select pixel 0 pixel 1
2	RW	0x0	lane1_en 0 lane1 disable 1 lane1 enable When camera_mode = 1, camera_sel should set to 1 to make lane1 enable
1	RW	0x0	lane0_sel Lane0 input source select 0 pixel 0 1 pixel 1
0	RW	0x0	lane0_en 0 lane0 disable 1 lane0 enable When camera_mode = 1, camera_sel should set to 0 to make lane0 enable

Table 7-339 MIPI\_ADAPT\_ALIG\_CNTL7 0x47

Bit(s)	R/W	Default	Description
31:16	RW	0x0	inc_mask_timeout When inc_mask_timeout_en = 1, those bits are valid will be timeout after when no valid data with direct mode in vvalid period after N x vclk cycles
15	RW	0x0	inc_mask_timeout_en 0: in_mask timeout disable 1: in_mask timeout enable
14	RW	0x0	frame_sync_rst_en 1. use frame_sync to reset this module
13	RW	0x0	frame_sync_load_en 1. use frame_sync to reload register value
1:0	RW	0x0	inc_mask_tick_sel 0: vclk 1: xtal3_tick 2: 1us_tick 3: 10us_tick



Table 7-340 MIPI\_ADAPT\_ALIG\_CNTL8 0x48

Bit(s)	R/W	Default	Description
31	RW	0x0	start_en Aligment start enable,will auto clear to 0
30	RW	0x0	Soft register load enable
30	RW	0x0	frame_vs_dst_dly_exc_clr clear all frame_vs_dst_dly_exc status
15	RW	0x0	Line irq enable,generate irq in configure number of line start
13	RW	0x0	no_frame_sv_sync_en When frame_sync_en =0,can start frame at frame_start_line and frame_start_pixel after counter equal h_num and v_num
12	RW	0x0	exc_mask_dis When in direct mode pixel or line is received is more than setting,and the inc_mask_timeout_en is valid,will treat it as pixel or line exceed,can select stop counter or not stop counter until received the lineend/frameend information. 0 hold counter when pixel or line exceed 1 not hold counter when pixel or line exceed
11	RW	0x0	mult_camera_req_sf_en mult_camera_req_sf disable 1 mult_camera_req_sf enable
10	RW	0x0	mult_camera_req_sf ISP in mult camera mode will output isp_multictx_frame_req, we should start transmit when isp_multictx_frame_req.But we can use this register instand of isp_multictx_frame_req when mult_camera_req_sf_en = 1
6	RW	0x0	isp_mult_frame_req_en 0.no check isp_multictx_frame_req value when start new frame trans 1.check isp_multictx_frame_req value when start new frame trans
5	RW	0x0	frame_continue disable continue mode,should set start_en each frame 1 enable continue mode,will auto start the next frame transmit

Table 7-341 MIPI\_ADAPT\_ALIG\_CNTL9 0x49

Bit(s)	R/W	Default	Description
31	RW	0x0	frame_sync_dely_en Can use frame_sync from frontend after delay some cycles of vclk 0 Frame_sync delay disable 1 Frame_sync delay enable
30:16	RW	0x0	frame_sync_dely_time Frame_sync delay time,N x vclk cycles,valid when frame_sync_dely_en = 1
15:0	RW	0x0	generate irq in configure number of line start

Table 7-342 MIPI\_ADAPT\_ALIG\_CNTL10 0x4a

Bit(s)	R/W	Default	Description
31:24	RW	0x0	isp_hold_src_en bit30:md2 hold cke enable bit29:flicker hold cke enable bit28:md hold cke enable bit27:ds2 hold cke enable bit26:ds1 hold cke enable bit25:ds0 hold cke enable bit24:crop_mif hold cke enable
18:16	RW	0x0	frame_vs_sel Alignment will send out a frame_vs single to top for sync. frame_vs select 0 frame_sync_tm = frame_sync_dely_en ? frame_syunc_dly : frame_sync 1 frame_vs_p,trans start,vcnt ==0 and hcnt == 0 2 frame_sync,input from fe_mux 4 frame_vs_cke_dly,output vvalid_pos with cke delay 6 frame_start_irq_set,vvalid_pos other frame_end
14:0	RW	0x0	frame_vs_cke_delay Frame_vs delay time when cke valid,N x vclk cycles,valid when frame_vs_sel = 4

Table 7-343 MIPI\_ADAPT\_ALIG\_CNTL11 0x4b

frame\_vs single for crop\_mif

Bit(s)	R/W	Default	Description
18:16	RW	0x0	frame_vs_sel Alignment will send out a frame_vs single to motion detect for sync. frame_vs to swrmif select 0 frame_sync_tm = frame_sync_dely_en ? frame_syunc_dly : frame_sync 1 frame_vs_p,trans start,vcnt ==0 and hcnt == 0 2 frame_sync,input from fe_mux 4 frame_vs_cke_dly,output vvalid_pos with cke delay other frame_end
14:0	RW	0x0	frame_vs_cke_delay Frame_vs delay time when cke valid,N x vclk cycles,valid when frame_vs_sel = 4

MIPI\_ADAPT\_ALIG\_CNTL12 0x4c frame\_vs single for ds0 , same as MIPI\_ADAPT\_ALIG\_CNTL11;  
MIPI\_ADAPT\_ALIG\_CNTL13 0x4d frame\_vs single for ds1 , same as MIPI\_ADAPT\_ALIG\_CNTL11;  
MIPI\_ADAPT\_ALIG\_CNTL14 0x4e frame\_vs single for ds2 , same as MIPI\_ADAPT\_ALIG\_CNTL11;  
MIPI\_ADAPT\_ALIG\_CNTL15 0x4f frame\_vs single for md , same as MIPI\_ADAPT\_ALIG\_CNTL11;  
MIPI\_ADAPT\_ALIG\_CNTL16 0x50 frame\_vs single for flicker , same as MIPI\_ADAPT\_ALIG\_CNTL11;  
MIPI\_ADAPT\_ALIG\_CNTL17 0x51 frame\_vs single for md2 , same as MIPI\_ADAPT\_ALIG\_CNTL11;  
MIPI\_ADAPT\_ALIG\_CNTL18 0x52 frame\_vs single for tnr\_de\_wr , same as MIPI\_ADAPT\_ALIG\_CNTL11;  
MIPI\_ADAPT\_ALIG\_CNTL19 0x53 frame\_vs single for tnr\_de\_rd , same as MIPI\_ADAPT\_ALIG\_CNTL11

Table 7-344 MIPI\_ADAPT\_ALIG\_ST0 0x5a

Bit(s)	R/W	Default	Description
31	R	0x0	frame_end_d_hold
30	R	0x0	trans_start_d
29	R	0x0	pixel_inc_mask

Bit(s)	R/W	Default	Description
28	R	0x0	line_inc_mask
27:20	R	0x0	isp_hold_src_v
19	R	0x0	trans_en
18	R	0x0	isp_multictx_frame_req
17	R	0x0	isp_hold_v
16:4	R	0x0	flow_done_src_hold
2	R	0x0	cke_d
1	R	0x0	hvalid_p
0	R	0x0	vvalid_p

Table 7-345 MIPI\_ADAPT\_ALIG\_ST1 0x5b

Bit(s)	R/W	Default	Description
31:16	R	0x0	h_cnt
15:0	R	0x0	v_cnt

Table 7-346 MIPI\_ADAPT\_ALIG\_ST2 0x5c

Bit(s)	R/W	Default	Description
24:16	R	0x0	frame_vs_dst_dly_exc
15:0	R	0x0	dir_frame_cnt

Table 7-347 MIPI\_ADAPT\_FE\_MUX\_CTRL0 0x90

Bit(s)	R/W	Default	Description
29:28	RW	0x0	alig_frame_vs : frame_valid_sel 2'b11 : pixel_0_ack   pixel_1_ack 2'b10 : pixel_1_ack 2'b01 : pixel_0_ack 2'b00 : 0
27:24	RW	0x0	alig_frame_vs : frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs 2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs 8:ddr_rd0_lbuf_soft_rst 9:ddr_rd1_lbuf_soft_rst other:0
23:20	RW	0x0	pixel1_frame_vs: frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs

Bit(s)	R/W	Default	Description
			2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs 8:ddr_rd1_lbuf_soft_rst other:0
19:16	RW	0x0	pixel0_frame_vs: frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs 2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs 8:ddr_rd0_lbuf_soft_rst other:0
14	RW	0x0	ddr_rd1_line_wr_vs_sel 1:from fe isp path 0:from fe mem path
13	RW	0x0	ddr_rd1_frame_wr_done_sel 1:from fe isp path 0:from fe mem path
12	RW	0x0	ddr_rd1_line_wr_done_sel 1:from fe isp path 0:from fe mem path
9:8	RW	0x0	ddr_rd1_fe_single_sel 0:fe0 1:fe1 2:fe2 3:fe3
6	RW	0x0	ddr_rd0_line_wr_vs_sel 1:from fe isp path 0:from fe mem path
5	RW	0x0	ddr_rd0_frame_wr_done_sel 1:from fe isp path 0:from fe mem path
4	RW	0x0	ddr_rd0_line_wr_done_sel 1:from fe isp path 0:from fe mem path
1:0	RW	0x0	ddr_rd0_fe_single_sel 0:fe0 1:fe1 2:fe2 3:fe3

Table 7-348 MIPI\_ADAPT\_FE\_MUX\_CTRL1 0x91

Bit(s)	R/W	Default	Description
29:28	RW	0x0	lbuf1_frame_vs : frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs 2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs 8:ddr_rd1_lbuf_soft_rst other:0
23:20	RW	0x0	lbuf0_frame_vs : frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs 2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs 8:ddr_rd0_lbuf_soft_rst other:0
16	RW	0x0	adapt_de_rd_frame_vs□ frame_valid_sel 1:ddr_rd1 0:ddr_rd0
15:12	RW	0x0	adpat_de_rd_frame_vs: frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs 2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs 8:ddr_rd0_rd_start 0:ddr_rd1_rd_start other:0
11:8	RW	0x0	adpat_de_wr_frame_vs: frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs 2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs other:0

Bit(s)	R/W	Default	Description
7:4	RW	0x0	adpat_ds_noblock_frame_vs: frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs 2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs other:0
3:0	RW	0x0	adpat_fr_noblock_frame_vs: frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs 2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs other:0

Table 7-349 MIPI\_ADAPT\_FE\_MUX\_CTRL2 0xbe

Bit(s)	R/W	Default	Description
31:30	RW	0x0	frame_vs single for ds0 frame_vs_in_sel 0:0 1:isp_core_inf select bit pos 2:isp_core_inf select bit neg 3:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL12
29:24	RW	0x0	frame_vs single for ds0 isp_core_inf_sel select 1 bit info from isp_core as frame_vs source
23:20	RW	0x0	frame_vs single for crop_mif frame_vs_in_sel 0:0 1:isp_core_inf select bit pos 2:isp_core_inf select bit neg 3:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL11
19:16	RW	0x0	frame_vs single for crop_mif isp_core_inf_sel select 1 bit info from isp_core as frame_vs source
8	RW	0x0	isp_core_inf invert

Bit(s)	R/W	Default	Description
7:4	RW	0x0	frame_vs_1 for vlock: frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs 2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs 8:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL10 other:0
3:0	RW	0x0	frame_vs_0 for vlock: frame_vs_in_sel 0:fe0 isp path vs 1:fe1 isp path vs 2:fe2 isp path vs 3:fe3 isp path vs 4:fe0 mem path vs 5:fe1 mem path vs 6:fe2 mem path vs 7:fe3 mem path vs 8:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL10 other:0

Table 7-350 MIPI\_ADAPT\_FE\_MUX\_CTRL3 0xbf

Bit(s)	R/W	Default	Description
31:30	RW	0x0	frame_vs single for flicker frame_vs_in_sel 0:0 1:isp_core_inf select bit pos 2:isp_core_inf select bit neg 3:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL16
29:24	RW	0x0	frame_vs single for flicker isp_core_inf_sel select 1 bit info from isp_core as frame_vs source
23:20	RW	0x0	frame_vs single for md frame_vs_in_sel 0:0 1:isp_core_inf select bit pos 2:isp_core_inf select bit neg 3:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL15
19:16	RW	0x0	frame_vs single for md isp_core_inf_sel select 1 bit info from isp_core as frame_vs source
15:14	RW	0x0	frame_vs single for ds2 frame_vs_in_sel 0:0 1:isp_core_inf select bit pos 2:isp_core_inf select bit neg 3:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL14
13:8	RW	0x0	frame_vs single for ds2 isp_core_inf_sel select 1 bit info from isp_core as frame_vs source

Bit(s)	R/W	Default	Description
7:6	RW	0x0	frame_vs single for ds1 frame_vs_in_sel 0:0 1:isp_core_inf select bit pos 2:isp_core_inf select bit neg 3:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL13
5:0	RW	0x0	frame_vs single for ds1 isp_core_inf_sel select 1 bit info from isp_core as frame_vs source

Table 7-351 MIPI\_ADAPT\_FE\_MUX\_CTRL4 0xc0

Bit(s)	R/W	Default	Description
31	RW	0x0	isp_core_inf_sel for isp_core_flow_hold 0:isp_core_inf select bit pos 1:isp_core_inf select bit neg
29:24	RW	0x0	isp_core_inf_sel for isp_core_flow_hold select 1 bit info from isp_core as isp_core flow_hold to alig module
23:20	RW	0x0	frame_vs single for tnr_de_rd frame_vs_in_sel 0:0 1:isp_core_inf select bit pos 2:isp_core_inf select bit neg 3:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL19
19:16	RW	0x0	frame_vs single for tnr_de_rd isp_core_inf_sel select 1 bit info from isp_core as frame_vs source
15:14	RW	0x0	frame_vs single for tnr_de_wr frame_vs_in_sel 0:0 1:isp_core_inf select bit pos 2:isp_core_inf select bit neg 3:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL18
13:8	RW	0x0	frame_vs single for tnr_de_wr isp_core_inf_sel select 1 bit info from isp_core as frame_vs source
7:6	RW	0x0	frame_vs single for md2 frame_vs_in_sel 0:0 1:isp_core_inf select bit pos 2:isp_core_inf select bit neg 3:alig_frame_vs select in MIPI_ADAPT_ALIG_CNTL17
5:0	RW	0x0	frame_vs single for md2 isp_core_inf_sel select 1 bit info from isp_core as frame_vs source

Table 7-352 MIPI\_ADAPT\_FE\_MUX\_CTRL5 0xc1

Bit(s)	R/W	Default	Description
31:0	RW	0x0	tnr_wr_fail_mask[31:0],use 43 bit isp_core_info as isp wr fail flag,set mask to enable each bit



**Table 7-353 MIPI\_ADAPT\_FE\_MUX\_CTRL6 0xc2**

Bit(s)	R/W	Default	Description
31	RW	0x0	tnr_wr_fail_enable
30	RW	0x0	tnr_wr_fail_sel: 0:neg 1:pos
10:0	RW	0x0	tnr_wr_fail_mask[42:32],use 43 bit isp_core_info as isp wr fail flag,set mask to enable each bit

**Table 7-354 MIPI\_ADAPT\_FE\_MUX\_CTRL7 0xc3**

Bit(s)	R/W	Default	Description
31:0	RW	0x0	tnr_rdr_fail_mask[31:0],use 43 bit isp_core_info as isp rd fail flag,set mask to enable each bit

**Table 7-355 MIPI\_ADAPT\_FE\_MUX\_CTRL8 0xc4**

Bit(s)	R/W	Default	Description
31	RW	0x0	tnr_rdr_fail_enable
30	RW	0x0	tnr_rd_fail_sel: 0:neg 1:pos
10:0	RW	0x0	tnr_rd_fail_mask[42:32],use 43 bit isp_core_info as isp rd fail flag,set mask to enable each bit

**Table 7-356 MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0 0x92 frame\_vs for ddr\_rd0 isp path delay and monitor**

Bit(s)	R/W	Default	Description
31	RW	0x0	soft_rst
29	RW	0x0	monitor clear
28	RW	0x0	monitor enable
27	RW	0x0	vs_delay exceed one frame flag clear
19	RW	0x0	vs_delay enable
17:16	RW	0x0	vs_delay tick select: 0:vclk 1:xtal3 2:1us 3:10us
14:0	RW	0x0	vs_delay count

**Table 7-357 MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0 0x93**

Bit(s)	R/W	Default	Description
31	R	0x0	vs_delay exceed one frame flag
30:16	R	0x0	vs_monitor max value
14:0	R	0x0	vs_monitor min value

MIPI_ADAPT_FE_MUX1_DLY_CNTRL0	0x94
frame_vs for ddr_rd1 isp path delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_CNTRL0	
MIPI_ADAPT_FE_MUX1_DLY_STAT0	0x95
frame_vs for ddr_rd1 isp path delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_STAT0	
MIPI_ADAPT_FE_MUX2_DLY_CNTRL0	0x96
frame_vs for pixel0 delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_CNTRL0	
MIPI_ADAPT_FE_MUX2_DLY_STAT0	0x97
frame_vs for pixel0 delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_STAT0	
MIPI_ADAPT_FE_MUX3_DLY_CNTRL0	0x98
frame_vs for pixel1 delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_CNTRL0	
MIPI_ADAPT_FE_MUX3_DLY_STAT0	0x99
frame_vs for pixel1 delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_STAT0	
MIPI_ADAPT_FE_MUX4_DLY_CNTRL0	0x9a
frame_vs for alig delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_CNTRL0	
MIPI_ADAPT_FE_MUX4_DLY_STAT0	0x9b
frame_vs for alig delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_STAT0	
MIPI_ADAPT_FE_MUX5_DLY_CNTRL0	0x9c
frame_vs for fr_noblock delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_CNTRL0	
MIPI_ADAPT_FE_MUX5_DLY_STAT0	0x9d
frame_vs for fr_noblock delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_STAT0	
MIPI_ADAPT_FE_MUX6_DLY_CNTRL0	0x9e
frame_vs for ds_noblock delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_CNTRL0	
MIPI_ADAPT_FE_MUX6_DLY_STAT0	0x9f
frame_vs for ds_noblock delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_STAT0	
MIPI_ADAPT_FE_MUX7_DLY_CNTRL0	0xa0
frame_vs for adapt_de_wr delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_CNTRL0	
MIPI_ADAPT_FE_MUX7_DLY_STAT0	0xa1
frame_vs for adapt_de_wr delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_STAT0	
MIPI_ADAPT_FE_MUX8_DLY_CNTRL0	0xa2
frame_vs for adapt_de_rd delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_CNTRL0	
MIPI_ADAPT_FE_MUX8_DLY_STAT0	0xa3
frame_vs for adapt_de_rd delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_STAT0	
MIPI_ADAPT_FE_MUX9_DLY_CNTRL0	0xa4
frame_vs for lbuf0 delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_CNTRL0	
MIPI_ADAPT_FE_MUX9_DLY_STAT0	0xa5
frame_vs for lbuf0 delay and monitor, same as MIPI_ADAPT_FE_MUX0_DLY_STAT0	
MIPI_ADAPT_FE_MUX10_DLY_CNTRL0	0xa6

frame\_vs for lbuf1 delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0  
MIPI\_ADAPT\_FE\_MUX10\_DLY\_STAT0 0xa7  
frame\_vs for lbuf1 delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0  
MIPI\_ADAPT\_FE\_MUX11\_DLY\_CNTRL0 0xba  
frame\_vs for ddr\_rd0 mem path delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0  
MIPI\_ADAPT\_FE\_MUX11\_DLY\_STAT0 0xbb  
frame\_vs for ddr\_rd0 mem path delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0  
MIPI\_ADAPT\_FE\_MUX12\_DLY\_CNTRL0 0xbc  
frame\_vs for ddr\_rd1 mem path delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0  
MIPI\_ADAPT\_FE\_MUX12\_DLY\_STAT0 0xbd  
frame\_vs for ddr\_rd1 mem path delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0  
MIPI\_ADAPT\_DST\_VS\_0\_DLY\_CNTRL0 0xa8  
frame\_vs for crop\_mif delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0  
MIPI\_ADAPT\_DST\_VS\_0\_DLY\_STAT0 0xa9  
frame\_vs for crop\_mif delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0  
MIPI\_ADAPT\_DST\_VS\_1\_DLY\_CNTRL0 0xaa  
frame\_vs for ds0 delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0  
MIPI\_ADAPT\_DST\_VS\_1\_DLY\_STAT0 0xab  
frame\_vs for ds0 delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0  
MIPI\_ADAPT\_DST\_VS\_2\_DLY\_CNTRL0 0xac  
frame\_vs for ds1 delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0  
MIPI\_ADAPT\_DST\_VS\_2\_DLY\_STAT0 0xad  
frame\_vs for ds1 delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0  
MIPI\_ADAPT\_DST\_VS\_3\_DLY\_CNTRL0 0xae  
frame\_vs for ds2 delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0  
MIPI\_ADAPT\_DST\_VS\_3\_DLY\_STAT0 0xaf  
frame\_vs for ds2 delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0  
MIPI\_ADAPT\_DST\_VS\_4\_DLY\_CNTRL0 0xb0  
frame\_vs for md delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0  
MIPI\_ADAPT\_DST\_VS\_4\_DLY\_STAT0 0xb1  
frame\_vs for md delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0  
MIPI\_ADAPT\_DST\_VS\_5\_DLY\_CNTRL0 0xb2  
frame\_vs for flicker delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0  
MIPI\_ADAPT\_DST\_VS\_5\_DLY\_STAT0 0xb3  
frame\_vs for flicker delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0  
MIPI\_ADAPT\_DST\_VS\_6\_DLY\_CNTRL0 0xb4  
frame\_vs for md2 delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0  
MIPI\_ADAPT\_DST\_VS\_6\_DLY\_STAT0 0xb5

frame\_vs for md2 delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0

MIPI\_ADAPT\_DST\_VS\_7\_DLY\_CNTRL0 0xb6

frame\_vs for tnr\_de\_wr delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0

MIPI\_ADAPT\_DST\_VS\_7\_DLY\_STAT0 0xb7

frame\_vs for tnr\_de\_wr delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0

MIPI\_ADAPT\_DST\_VS\_8\_DLY\_CNTRL0 0xb8

frame\_vs for tnr\_de\_rd delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_CNTRL0

MIPI\_ADAPT\_DST\_VS\_8\_DLY\_STAT0 0xb9

frame\_vs for tnr\_de\_rd delay and monitor, same as MIPI\_ADAPT\_FE\_MUX0\_DLY\_STAT0

## MPI\_ISP\_AHB\_DMA

**Table 7-358 ISP\_DMA\_BUS\_CTL0 0x00**

Bit(s)	R/W	Default	Description
21	R/W	0	reg_axi_arugt
20	R/W	0	reg_axi_awugt
15:8	R/W	0	reg_axi_id
1:0	R/W	0	reg_axi_burst_type: 00 single 01 INC2 10 inc4 11 inc8

**Table 7-359 ISP\_DMA\_CTL 0x01**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_dma_start,auto clear
30	R/W	0	reg_soft_rst
29	R/W	0	reg_fifo_rst
28	R/W	0	reg_fifo_noclk_gate
27	R/W	0	reg_dma_all_done_clr
26	R/W	0	reg_ahb_resp_err_clr
23:16	R/W	0	reg_dma_done_clr
15:8	R/W	0	reg_dma_done_en
7	R/W	0	reg_dma_all_done_en
6	R/W	0	reg_ahb_resp_err_en
4	R/W	0	reg_isp_dma_en
2:0	R/W	0	reg_task_num

**Table 7-360 ISP\_DMA\_SRC\_ADDR0 0x02**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_src_addr

**Table 7-361 ISP\_DMA\_DST\_ADDR0 0x03**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_dst_addr

**Table 7-362 ISP\_DMA\_CTL\_TASK0 0x04**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_dma_dir 1: ddr to ISP 0: ISP to ddr
15:0	R/W	0	reg_dma_len, (N-1)xword

ISP\_DMA\_SRC\_ADDR1                    0x05

same as ISP\_DMA\_SRC\_ADDR0

ISP\_DMA\_DST\_ADDR1    0x06

same as ISP\_DMA\_DST\_ADDR0

ISP\_DMA\_CTL\_TASK1                    0x07

same as ISP\_DMA\_CTL\_ADDR0

ISP\_DMA\_SRC\_ADDR2                    0x08

same as ISP\_DMA\_SRC\_ADDR0

ISP\_DMA\_DST\_ADDR2    0x09

same as ISP\_DMA\_DST\_ADDR0

ISP\_DMA\_CTL\_TASK2                    0x0a

same as ISP\_DMA\_CTL\_ADDR0

ISP\_DMA\_SRC\_ADDR3                    0x0b

same as ISP\_DMA\_SRC\_ADDR0

ISP\_DMA\_DST\_ADDR3    0x0c

same as ISP\_DMA\_DST\_ADDR0

ISP\_DMA\_CTL\_TASK3                    0x0d

same as ISP\_DMA\_CTL\_ADDR0

ISP\_DMA\_SRC\_ADDR4                    0x0e

same as ISP\_DMA\_SRC\_ADDR0

ISP\_DMA\_DST\_ADDR4    0x0f

same as ISP\_DMA\_DST\_ADDR0

ISP\_DMA\_CTL\_TASK4                    0x10

same as ISP\_DMA\_CTL\_ADDR0  
 ISP\_DMA\_SRC\_ADDR5 0x11  
 same as ISP\_DMA\_SRC\_ADDR0  
 ISP\_DMA\_DST\_ADDR5 0x12  
 same as ISP\_DMA\_DST\_ADDR0  
 ISP\_DMA\_CTL\_TASK5 0x13  
 same as ISP\_DMA\_CTL\_ADDR0  
 ISP\_DMA\_SRC\_ADDR6 0x14  
 same as ISP\_DMA\_SRC\_ADDR0  
 ISP\_DMA\_DST\_ADDR6 0x15  
 same as ISP\_DMA\_DST\_ADDR0  
 ISP\_DMA\_CTL\_TASK6 0x16  
 same as ISP\_DMA\_CTL\_ADDR0  
 ISP\_DMA\_SRC\_ADDR7 0x17  
 same as ISP\_DMA\_SRC\_ADDR0  
 ISP\_DMA\_DST\_ADDR7 0x18  
 same as ISP\_DMA\_DST\_ADDR0  
 ISP\_DMA\_CTL\_TASK7 0x19  
 same as ISP\_DMA\_CTL\_ADDR0

**Table 7-363 ISP\_DMA\_ST0 0x1a**

Bit(s)	R/W	Default	Description
31:0	RO	0	reg_axi_waddr

**Table 7-364 ISP\_DMA\_ST1 0x1b**

Bit(s)	R/W	Default	Description
31:0	RO	0	reg_axi_raddr

**Table 7-365 ISP\_DMA\_ST2 0x1c**

Bit(s)	R/W	Default	Description
31:0	RO	0	reg_ahb_raddr

**Table 7-366 ISP\_DMA\_ST3 0x1d**

Bit(s)	R/W	Default	Description
23:16	RO	0	r_dma_done
15	RO	0	r_dma_all_done
14	RO	0	r_ahb_resp_err

Bit(s)	R/W	Default	Description
12:8	RO	0	c_fifo_count
64:4	RO	0	r_task_cnt
2:0	RO	0	r_task_num

Table 7-367 ISP\_DMA\_ST4 0x1e

Bit(s)	R/W	Default	Description
31:16	RO	0	r_ahb_cmd_num
15:0	RO	0	r_axi_cmd_num

Table 7-368 ISP\_DMA\_ST5 0x1f

Bit(s)	R/W	Default	Description
29:24	RO	0	r_axi_wcmd_cnt
23:20	RO	0	c_axi_wdata_len
19:16	RO	0	r_axi_wdata_cnt
15:0	RO	0	r_axi_wdata_num

Table 7-369 ISP\_DMA\_ST6 0x20

Bit(s)	R/W	Default	Description
31:29	RO	0	r_ahb_fifo_rdat_valid
28:27	RO	0	r_ahb_rdata_valid_flag
26:21	RO	0	r_axi_wrsp_cnt
20:16	RO	0	r_axi_wr_fifo_vcount
15:0	RO	0	r_axi_rdata_cnt

Table 7-370 ISP\_DMA\_PENDING 0x23

Bit(s)	R/W	Default	Description
11	R/W	0	task7 done pending
10	R/W	0	task6 done pending
9	R/W	0	task5 done pending
8	R/W	0	task4 done pending
7	R/W	0	task3 done pending
6	R/W	0	task2 done pending
5	R/W	0	task1 done pending
4	R/W	0	task0 done pending

Bit(s)	R/W	Default	Description
1	R/W	0	ahb resp err pending
0	R/W	0	all task done pending

Table 7-371 ISP\_DMA\_IRQ\_MASK 0x24

Bit(s)	R/W	Default	Description
11	R/W	0	task7 done pending mask
10	R/W	0	task6 done pending mask
9	R/W	0	task5 done pending mask
8	R/W	0	task4 done pending mask
7	R/W	0	task3 done pending mask
6	R/W	0	task2 done pending mask
5	R/W	0	task1 done pending mask
4	R/W	0	task0 done pending mask
1	R/W	0	ahb resp err pending mask
0	R/W	0	all task done pending mask

## MPI\_ISP\_APB\_DMA

Table 7-372 ISP\_DMA\_AXI\_CTL 0x00

Bit(s)	R/W	Default	Description
21	R/W	0	reg_axi_arugt
20	R/W	0	reg_axi_awugt
12:8	R/W	0	reg_axi_id
3:2	R/W	0	reg_axi_wr_burst_type: 00 single 01 INC2 10 inc4 11 inc8
1:0	R/W	0	reg_axi_rd_burst_type: 00 single 01 INC2 10 inc4 11 inc8

Table 7-373 ISP\_DMA\_CTL0 0x01

Bit(s)	R/W	Default	Description
31	R/W	0	apb_dma_enable
27	R/W	0	reg_soft_rst
26	R/W	0	reg_trig_fifo_rst
25	R/W	0	reg_axi_rfifo_rst



Bit(s)	R/W	Default	Description
26	R/W	0	reg_axi_wfifo_rst
18:16	R/W	0	reg_trig_fifo_stat_sel,for trig_fifo_stat read value select

Table 7-374 ISP\_DMA\_PENDING0 0x02

Bit(s)	R/W	Default	Description
31:0	R/W	0	apb_dma pending[31:0],write 1 to clear

Table 7-375 ISP\_DMA\_PENDING1 0x03

Bit(s)	R/W	Default	Description
31:0	R/W	0	apb_dma pending[63:32],write 1 to clear

Table 7-376 ISP\_DMA\_IRQ\_MASK0 0x04

Bit(s)	R/W	Default	Description
31:0	R/W	0	apb_dma pending[31:0] enable,write 1 to enable each pending bit

Table 7-377 ISP\_DMA\_IRQ\_MASK1 0x05

Bit(s)	R/W	Default	Description
<b>bit</b>	<b>irq source</b>	<b>bit</b>	<b>irq source</b>
31:0	R/W	0	apb_dma pending[63:32] enable,write 1 to enable each pending bit

Bit	irq Source	Bit	irq Source
63	NA	31	src15_trig_overflow
62	NA	30	src14_trig_overflow
61	NA	29	src13_trig_overflow
60	NA	28	src12_trig_overflow
59	NA	27	src11_trig_overflow
58	NA	26	src10_trig_overflow
57	NA	25	src9_trig_overflow
56	NA	24	src8_trig_overflow
55	NA	23	src7_trig_overflow
54	NA	22	src6_trig_overflow
53	NA	21	src5_trig_overflow
52	NA	20	src4_trig_overflow
51	NA	19	src3_trig_overflow

Bit	irq Source	Bit	irq Source
50	NA	18	src2_trig_overflow
49	bus_mon_fast_irq	17	src1_trig_overflow
48	trig_fifo_overflow	16	src0_trig_overflow
47	src15_task_done without valid task	15	src15 task done
46	src14_task_done without valid task	14	src14 task done
45	src13_task_done without valid task	13	src13 task done
44	src12_task_done without valid task	12	src12 task done
43	src11_task_done without valid task	11	src11 task done
42	src10_task_done without valid task	10	src10 task done
41	src9_task_done without valid task	9	src9 task done
40	src8_task_done without valid task	8	src8 task done
39	src7_task_done without valid task	7	src7 task done
38	src6_task_done without valid task	6	src6 task done
37	src5_task_done without valid task	5	src5 task done
36	src4_task_done without valid task	4	src4 task done
35	src3_task_done without valid task	3	src3 task done
34	src2_task_done without valid task	2	src2 task done
33	src1_task_done without valid task	1	src1 task done
32	src0_task_done without valid task	0	src0 task done

Table 7-378 ISP\_DMA\_ARBIT\_CNTL0 0x20

Bit(s)	R/W	Default	Description
3:0	R/W	0	apb arbiter :bus_monitor_cntl

Table 7-379 ISP\_DMA\_ARBIT\_CNTL1 0x21

Bit(s)	R/W	Default	Description
31:0	R/W	0	apb arbiter :bus_monitor_addr

Table 7-380 ISP\_DMA\_ARBIT\_CNTL2 0x22

Bit(s)	R/W	Default	Description
31:0	R/W	0	apb arbiter :bus_monitor_data

Table 7-381 ISP\_DMA\_ARBIT\_CNTL3 0x23

Bit(s)	R/W	Default	Description
31:0	R/W	0	apb arbiter :bus_monitor_data_msk

**Table 7-382 ISP\_DMA\_SRC0\_CTL 0x30**

Bit(s)	R/W	Default	Description
3	R/W	0	pingpong task force sel value: 1 pong 0 ping
2	R/W	0	pingpong task force enable
1	R/W	0	pingpong task initial start value 1 pong 0 ping
0	R/W	0	source trig enable

**Table 7-383 ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0 0x31**

Bit(s)	R/W	Default	Description
31:0	R/W	0	task0 cmd store address

**Table 7-384 ISP\_DMA\_SRC0\_PING\_DST\_ADDR0 0x32**

Bit(s)	R/W	Default	Description
31:0	R/W	0	task0 dst data address,for apb read only

**Table 7-385 ISP\_DMA\_SRC0\_PING\_TASK0 0x33**

Bit(s)	R/W	Default	Description
31	R/W	0	task0 valid,auto clear after task done
30	R/W	0	task0 type 1 apb write 0 apb read
15:0	R/W	0	task0 cmd length,N-1 words

**Table 7-386 ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1 0x34**

Bit(s)	R/W	Default	Description
31:0	R/W	0	task1 cmd store address

**Table 7-387 ISP\_DMA\_SRC0\_PING\_DST\_ADDR1 0x35**

Bit(s)	R/W	Default	Description
31:0	R/W	0	task1 dst data address,for apb read only

**Table 7-388 ISP\_DMA\_SRC0\_PING\_TASK1 0x36**

Bit(s)	R/W	Default	Description
31	R/W	0	task1 valid,auto clear after task done
30	R/W	0	task1 type 1 apb write 0 apb read
15:0	R/W	0	task1 cmd length,N-1 words

ISP\_DMA\_SRC0\_PONG\_CMD\_ADDR0 0x37

same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC0\_PONG\_DST\_ADDR0 0x38

same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC0\_PONG\_TASK0 0x39

same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC0\_PONG\_CMD\_ADDR1 0x3a

same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC0\_PONG\_DST\_ADDR1 0x3b

same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC0\_PONG\_TASK1 0x3c

same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC1\_CTL 0x3d

same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC1\_PING\_CMD\_ADDR0 0x3e

same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC1\_PING\_DST\_ADDR0 0x3f

same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC1\_PING\_TASK0 0x40

same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC1\_PING\_CMD\_ADDR1 0x41

same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC1\_PING\_DST\_ADDR1 0x42

same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC1\_PING\_TASK1 0x43

same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC1\_PONG\_CMD\_ADDR0 0x44

same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC1\_PONG\_DST\_ADDR0 0x45

same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC1\_PONG\_TASK0 0x46  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC1\_PONG\_CMD\_ADDR1 0x47  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC1\_PONG\_DST\_ADDR1 0x48  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC1\_PONG\_TASK1 0x49  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC2\_CTL 0x4a  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC2\_PING\_CMD\_ADDR0 0x4b  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC2\_PING\_DST\_ADDR0 0x4c  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC2\_PING\_TASK0 0x4d  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC2\_PING\_CMD\_ADDR1 0x4e  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC2\_PING\_DST\_ADDR1 0x4f  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC2\_PING\_TASK1 0x50  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC2\_PONG\_CMD\_ADDR0 0x51  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC2\_PONG\_DST\_ADDR0 0x52  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC2\_PONG\_TASK0 0x53  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC2\_PONG\_CMD\_ADDR1 0x54  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC2\_PONG\_DST\_ADDR1 0x55  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC2\_PONG\_TASK1 0x56  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC3\_CTL 0x57  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC3\_PING\_CMD\_ADDR0 0x58  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC3\_PING\_DST\_ADDR0 0x59  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC3\_PING\_TASK0 0x5a  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC3\_PING\_CMD\_ADDR1 0x5b  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC3\_PING\_DST\_ADDR1 0x5c  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC3\_PING\_TASK1 0x5d  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC3\_PONG\_CMD\_ADDR0 0x5e  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC3\_PONG\_DST\_ADDR0 0x5f  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC3\_PONG\_TASK0 0x60  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC3\_PONG\_CMD\_ADDR1 0x61  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC3\_PONG\_DST\_ADDR1 0x62  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC3\_PONG\_TASK1 0x63  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC4\_CTL 0x64  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC4\_PING\_CMD\_ADDR0 0x65  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC4\_PING\_DST\_ADDR0 0x66  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC4\_PING\_TASK0 0x67  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC4\_PING\_CMD\_ADDR1 0x68  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC4\_PING\_DST\_ADDR1 0x69  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC4\_PING\_TASK1 0x6a  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC4\_PONG\_CMD\_ADDR0 0x6b  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC4\_PONG\_DST\_ADDR0 0x6c  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC4\_PONG\_TASK0 0x6d  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC4\_PONG\_CMD\_ADDR1 0x6e  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC4\_PONG\_DST\_ADDR1 0x6f  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC4\_PONG\_TASK1 0x70  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC5\_CTL 0x71  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC5\_PING\_CMD\_ADDR0 0x72  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC5\_PING\_DST\_ADDR0 0x73  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC5\_PING\_TASK0 0x74  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC5\_PING\_CMD\_ADDR1 0x75  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC5\_PING\_DST\_ADDR1 0x76  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC5\_PING\_TASK1 0x77  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC5\_PONG\_CMD\_ADDR0 0x78  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC5\_PONG\_DST\_ADDR0 0x79  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC5\_PONG\_TASK0 0x7a  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC5\_PONG\_CMD\_ADDR1 0x7b  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC5\_PONG\_DST\_ADDR1 0x7c  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC5\_PONG\_TASK1 0x7d  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC6\_CTL 0x7e  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC6\_PING\_CMD\_ADDR0 0x7f  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC6\_PING\_DST\_ADDR0 0x80  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC6\_PING\_TASK0 0x81  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC6\_PING\_CMD\_ADDR1 0x82  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC6\_PING\_DST\_ADDR1 0x83  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC6\_PING\_TASK1 0x84  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC6\_PONG\_CMD\_ADDR0 0x85  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC6\_PONG\_DST\_ADDR0 0x86  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC6\_PONG\_TASK0 0x87  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC6\_PONG\_CMD\_ADDR1 0x88  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC6\_PONG\_DST\_ADDR1 0x89  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC6\_PONG\_TASK1 0x8a  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC7\_CTL 0x8b  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC7\_PING\_CMD\_ADDR0 0x8c  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC7\_PING\_DST\_ADDR0 0x8d  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC7\_PING\_TASK0 0x8e  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC7\_PING\_CMD\_ADDR1 0x8f  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC7\_PING\_DST\_ADDR1 0x90  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC7\_PING\_TASK1 0x91  
same as ISP\_DMA\_SRC0\_PING\_TASK1



ISP\_DMA\_SRC7\_PONG\_CMD\_ADDR0 0x92  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC7\_PONG\_DST\_ADDR0 0x93  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC7\_PONG\_TASK0 0x94  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC7\_PONG\_CMD\_ADDR1 0x95  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC7\_PONG\_DST\_ADDR1 0x96  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC7\_PONG\_TASK1 0x97  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC8\_CTL 0x98  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC8\_PING\_CMD\_ADDR0 0x99  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC8\_PING\_DST\_ADDR0 0x9a  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC8\_PING\_TASK0 0x9b  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC8\_PING\_CMD\_ADDR1 0x9c  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC8\_PING\_DST\_ADDR1 0x9d  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC8\_PING\_TASK1 0x9e  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC8\_PONG\_CMD\_ADDR0 0x9f  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC8\_PONG\_DST\_ADDR0 0xa0  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC8\_PONG\_TASK0 0xa1  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC8\_PONG\_CMD\_ADDR1 0xa2  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC8\_PONG\_DST\_ADDR1 0xa3  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC8\_PONG\_TASK1 0xa4  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC9\_CTL 0xa5  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC9\_PING\_CMD\_ADDR0 0xa6  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC9\_PING\_DST\_ADDR0 0xa7  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC9\_PING\_TASK0 0xa8  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC9\_PING\_CMD\_ADDR1 0xa9  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC9\_PING\_DST\_ADDR1 0xaa  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC9\_PING\_TASK1 0xab  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC9\_PONG\_CMD\_ADDR0 0xac  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC9\_PONG\_DST\_ADDR0 0xad  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC9\_PONG\_TASK0 0xae  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC9\_PONG\_CMD\_ADDR1 0xaf  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC9\_PONG\_DST\_ADDR1 0xb0  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC9\_PONG\_TASK1 0xb1  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC10\_CTL 0xb2  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC10\_PING\_CMD\_ADDR0 0xb3  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC10\_PING\_DST\_ADDR0 0xb4  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC10\_PING\_TASK0 0xb5  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC10\_PING\_CMD\_ADDR1 0xb6  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC10\_PING\_DST\_ADDR1 0xb7  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC10\_PING\_TASK1 0xb8  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC10\_PONG\_CMD\_ADDR0 0xb9  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC10\_PONG\_DST\_ADDR0 0xba  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC10\_PONG\_TASK0 0xbb  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC10\_PONG\_CMD\_ADDR1 0xbc  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC10\_PONG\_DST\_ADDR1 0xbd  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC10\_PONG\_TASK1 0xbe  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC11\_CTL 0xbf  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC11\_PING\_CMD\_ADDR0 0xc0  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC11\_PING\_DST\_ADDR0 0xc1  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC11\_PING\_TASK0 0xc2  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC11\_PING\_CMD\_ADDR1 0xc3  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC11\_PING\_DST\_ADDR1 0xc4  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC11\_PING\_TASK1 0xc5  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC11\_PONG\_CMD\_ADDR0 0xc6  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC11\_PONG\_DST\_ADDR0 0xc7  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC11\_PONG\_TASK0 0xc8  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC11\_PONG\_CMD\_ADDR1 0xc9  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC11\_PONG\_DST\_ADDR1 0xca  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC11\_PONG\_TASK1 0xcb  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC12\_CTL 0xcc  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC12\_PING\_CMD\_ADDR0 0xcd  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC12\_PING\_DST\_ADDR0 0xce  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC12\_PING\_TASK0 0xcf  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC12\_PING\_CMD\_ADDR1 0xd0  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC12\_PING\_DST\_ADDR1 0xd1  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC12\_PING\_TASK1 0xd2  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC12\_PONG\_CMD\_ADDR0 0xd3  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC12\_PONG\_DST\_ADDR0 0xd4  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC12\_PONG\_TASK0 0xd5  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC12\_PONG\_CMD\_ADDR1 0xd6  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC12\_PONG\_DST\_ADDR1 0xd7  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC12\_PONG\_TASK1 0xd8  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC13\_CTL 0xd9  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC13\_PING\_CMD\_ADDR0 0xda  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC13\_PING\_DST\_ADDR0 0xdb  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC13\_PING\_TASK0 0xdc  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC13\_PING\_CMD\_ADDR1 0xdd  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC13\_PING\_DST\_ADDR1 0xde  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC13\_PING\_TASK1 0xdf  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC13\_PONG\_CMD\_ADDR0 0xe0  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC13\_PONG\_DST\_ADDR0 0xe1  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC13\_PONG\_TASK0 0xe2  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC13\_PONG\_CMD\_ADDR1 0xe3  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC13\_PONG\_DST\_ADDR1 0xe4  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC13\_PONG\_TASK1 0xe5  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC14\_CTL 0xe6  
same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC14\_PING\_CMD\_ADDR0 0xe7  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC14\_PING\_DST\_ADDR0 0xe8  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC14\_PING\_TASK0 0xe9  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC14\_PING\_CMD\_ADDR1 0xea  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC14\_PING\_DST\_ADDR1 0xeb  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC14\_PING\_TASK1 0xec  
same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC14\_PONG\_CMD\_ADDR0 0xed  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC14\_PONG\_DST\_ADDR0 0xee  
same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC14\_PONG\_TASK0 0xef  
same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC14\_PONG\_CMD\_ADDR1 0xf0  
same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC14\_PONG\_DST\_ADDR1 0xf1  
 same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC14\_PONG\_TASK1 0xf2  
 same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC15\_CTL 0xf3  
 same as ISP\_DMA\_SRC0\_CTL

ISP\_DMA\_SRC15\_PING\_CMD\_ADDR0 0xf4  
 same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC15\_PING\_DST\_ADDR0 0xf5  
 same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC15\_PING\_TASK0 0xf6  
 same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC15\_PING\_CMD\_ADDR1 0xf7  
 same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC15\_PING\_DST\_ADDR1 0xf8  
 same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC15\_PING\_TASK1 0xf9  
 same as ISP\_DMA\_SRC0\_PING\_TASK1

ISP\_DMA\_SRC15\_PONG\_CMD\_ADDR0 0xfa  
 same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR0

ISP\_DMA\_SRC15\_PONG\_DST\_ADDR0 0xfb  
 same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR0

ISP\_DMA\_SRC15\_PONG\_TASK0 0xfc  
 same as ISP\_DMA\_SRC0\_PING\_TASK0

ISP\_DMA\_SRC15\_PONG\_CMD\_ADDR1 0xfd  
 same as ISP\_DMA\_SRC0\_PING\_CMD\_ADDR1

ISP\_DMA\_SRC15\_PONG\_DST\_ADDR1 0xfe  
 same as ISP\_DMA\_SRC0\_PING\_DST\_ADDR1

ISP\_DMA\_SRC15\_PONG\_TASK1 0xff  
 same as ISP\_DMA\_SRC0\_PING\_TASK1

## MPI\_ISP\_TOP

**Table 7-389 MIPI\_TOP\_CSI2\_CTRL0 0xc0**

Bit(s)	R/W	Default	Description
31:28	RW	0x0	debug_sel
27	RW	0x0	isp_ahb_force force isp_ahb bus hready_out to 1,set when isp_core power off
20:0	RW	0x0	soft_reset for each sub module

Table 7-390 MIPI\_TOP\_CSI2\_CTRL1 0xbf

Bit(s)	R/W	Default	Description
23:22	RW	0x3	flicker_data3_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0
21:20	RW	0x2	flicker_data2_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0
19:18	RW	0x1	flicker_data1_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0
17:16	RW	0x0	flicker_data0_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0
15:14	RW	0x3	md2_data3_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0
13:12	RW	0x2	md2_data2_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0
11:10	RW	0x1	md2_data1_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0
9:8	RW	0x0	md2_data0_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0
7:6	RW	0x3	md_data3_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0
5:4	RW	0x2	md_data2_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0

Bit(s)	R/W	Default	Description
3:2	RW	0x1	md_data1_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0
1:0	RW	0x0	md_data0_sel 3:isp_vdata3 2:isp_vdata2 1:isp_vdata1 0:isp_vdata0

Table 7-391 MIPI\_TOP\_ADAPT\_DE\_CTRL0 0xc1

Bit(s)	R/W	Default	Description
31:30	RW	0x0	adapt_de soft reset
7	RW	0x0	adapt_de AXI wr bus bypass enable
5:4	RW	0x0	adapt_de AXI wr bus sel: 0:fe0 1:fe1 2:fe2 3:fe3
3	RW	0x0	adapt_de AXI rd bus bypass enable
0	RW	0x0	adapt_de AXI rd bus sel: 0:ddr_rd0 1:ddr_rd1

Table 7-392 MIPI\_TOP\_ADAPT\_DE\_CTRL1 0xc2

Bit(s)	R/W	Default	Description
31	RW	0x0	adapt_de axi write bus div 2 control soft_reset
30	RW	0x0	adapt_de axi write bus div 2 control cmd_fifo_sf_rst
29	RW	0x0	adapt_de axi write bus div 2 control wdata_fifo_sf_rst
3:1	RW	0x0	adapt_de axi write bus div 2 control stat_sel,stat0 read value select
0	RW	0x0	adapt_de axi write bus div 2 control enable

Table 7-393 MIPI\_TOP\_ADAPT\_DE\_CTRL2 0xc3

Bit(s)	R/W	Default	Description
31:0	RW	0x0	MIN axi address to adapt_de



**Table 7-394 MIPI\_TOP\_ADAPT\_DE\_CTRL3 0xc4**

Bit(s)	R/W	Default	Description
31:0	RW	0x0	MAX axi address to adapt_de

**Table 7-395 MIPI\_TOP\_TNR\_DE\_CTRL0 0xc5**

Bit(s)	R/W	Default	Description
31:30	RW	0x0	tnr_de soft reset
0	RW	0x0	tnr_de AXI bus bypass enable

**Table 7-396 MIPI\_TOP\_ISP\_FR\_AXI\_CNTRL0 0xe4**

isp\_core fr axi bus aysnc control

Bit(s)	R/W	Default	Description
31	RW	0x0	axi_async: req_en
30	RW	0x0	axi_async :auto_gclk
29	RW	0x0	axi_async :dis_clk

**Table 7-397 MIPI\_TOP\_ARIBIT0\_CTRL 0xc6**

Bit(s)	R/W	Default	Description
31:30	RW	0x0	axi_aribit0 control : gclk_ctrl
15:0	RW	0x0	axi_aribit0 control : force urgent bit15:NA bit14:adapt_de_wr bit13:ahb_dma wr bit12:apb_dma wr bit11:fe3 wr bit10:fe2 wr bit9 :fe1 wr bit8:fe0 wr bit7:NA bit6:NA bit5:NA bit4:NA bit3:ahb_dma rd bit2:apb_dma rd bit1:ddr_rd1 rd bit0:ddr_rd0 rd

Table 7-398 MIPI\_TOP\_ARIBIT1\_CTRL 0xc7

Bit(s)	R/W	Default	Description
31:30	RW	0x0	axi_aribit1 control : gclk_ctrl
15:0	RW	0x0	axi_aribit1 control : force urgent bit15:md wr bit14:flicker wr bit13:md2 wr bit12:crop_mif wr bit11:ds2 wr bit10:ds1 wr bit9 :ds0 wr bit8:isp_core ds wr bit7:NA bit6:NA bit5:NA bit4:NA bit3:NA bit2:NA bit1:NA bit0:md rd

Table 7-399 MIPI\_TOP\_ARIBIT2\_CTRL 0xc8

Bit(s)	R/W	Default	Description
31:30	RW	0x0	axi_aribit2 control : gclk_ctrl
15:0	RW	0x0	axi_aribit2 control : force urgent bit15:NA bit14:NA bit13:NA bit12:tnr_de1 stream wr bit11:tnr_de1 meta wr bit10:tnr_de0 stream wr bit9 :tnr_de0 meta wr bit8:isp_core tnr wr bit7:NA bit6:NA bit5:NA bit4:tnr_de1 stream rd bit3tnr_de1 meta rd bit2:tnr_de0 stream rd bit1 :tnr_de0 meta rd bit0:isp_core tnr rd

Table 7-400 MIPI\_TOP\_AXI\_CMD\_MASK\_DDR\_RD0\_CTRL 0xc9

Bit(s)	R/W	Default	Description
31	RW	0x0	0 cmd_mask_en ddr_rd0 axi read cmd bus can be hold intermittent by this bit to reduce the bandwidth loading : disable : enable
15:8	RW	0x0	cmd_number When cmd_mask_en = 1,will hold ddr_rd0 axi read cmd bus after send those number of read cmd
7:0	RW	0x0	cmd_block_cycle When cmd_mask_en =1,after the number of read cmd equal cmd_number will hold the cmd bus for cmd_block_cycle cycles of vclk.

MIPI\_TOP\_AXI\_CMD\_MASK\_DDR\_RD1\_CTRL 0xca

same as MIPI\_TOP\_AXI\_CMD\_MASK\_DDR\_RD0\_CTRL,for ddr\_rd1 axi read bus mask

**Table 7-401 MIPI\_TOP\_AXI\_DS\_WBUS\_NOBLOCK\_CTRL 0xcb**

Bit(s)	R/W	Default	Description
31	RW	0x0	noblock_en Isp ds axi wrbus can not wait bresp when this bit set 0: disable 1: enable
30	RW	0x0	soft_rst ds_noblock soft_rst,write 1 to reset ds_noblock module
29	RW	0x0	cmd_cnt_clr Set cmd_cnt to 0,cmd_cnt used to count the number of cmd have not receive bresp from bus
18:16	RW	0x0	frame_sync_sel Check cmd_cnt equal 0 at frame_sync 5 : isp_vvalid_neg : isp_vvalid_pos : frame_vs (from alignment) 1: adapt_ds_no_vs 0 : disable
15	RW	0x0	frame_sync_dely_en 0 : frame_sync delay disable 1 : frame_sync delay enable
14:0	RW	0x0	frame_sync_dely_time frame_sync delay time,N x vclk cycles,valid when frame_sync_dely_en = 1

**Table 7-402 MIPI\_TOP\_AXI\_FS\_WBUS\_NOBLOCK\_CTRL 0xcc**

Bit(s)	R/W	Default	Description
31	RW	0x0	noblock_en Isp fr axi wrbus can not wait bresp when this bit set 0: disable 1: enable
30	RW	0x0	soft_rst fr_noblock soft_rst,write 1 to reset fr_noblock module
29	RW	0x0	cmd_cnt_clr Set cmd_cnt to 0,cmd_cnt used to count the number of cmd have not receive bresp from bus
18:16	RW	0x0	frame_sync_sel Check cmd_cnt equal 0 at frame_sync 5 : isp_vvalid_neg : isp_vvalid_pos : frame_vs (from alignment) 1: adapt_ds_no_vs 0 : disable
15	RW	0x0	frame_sync_dely_en 0 : frame_sync delay disable 1 : frame_sync delay enable
14:0	RW	0x0	frame_sync_dely_time frame_sync delay time,N x vclk cycles,valid when frame_sync_dely_en = 1

**Table 7-403 MIPI\_TOP\_FR\_WR\_UGT\_CTRL0 0xd0**

isp\_core fr axi wr control

Bit(s)	R/W	Default	Description
31	RW	0x0	auto_urgent_enable
30	RW	0x0	axi auto urgent enable
26:16	RW	0x0	axi auto urgent control THD1 mem0 cnt < THD1 ,will set qos to high value
15:12	RW	0x0	qos high value
10:0	RW	0x0	axi auto urgent control THD0 mem0 cnt >= THD0 ,will set qos to low value

**Table 7-404 MIPI\_TOP\_FR\_WR\_UGT\_CTRL1 0xd1**

isp\_core fr uv axi wr control

Bit(s)	R/W	Default	Description
30	RW	0x0	axi auto urgent enable
26:16	RW	0x0	axi auto urgent control THD1 mem1 cnt < THD1 ,will set qos to high value
15:12	RW	0x0	qos low value
10:0	RW	0x0	axi auto urgent control THD0 mem1 cnt >= THD0 ,will set qos to low value

**Table 7-405 MIPI\_TOP\_FR\_WR\_UGT\_CTRL2 0xd2**

isp\_core fr axi wr control

Bit(s)	R/W	Default	Description
31	RW	0x0	auto_sideband_enable
30	RW	0x0	axi auto sideband enable
26:16	RW	0x0	axi auto sideband control THD1 mem0 cnt < THD1 ,will set sideband to 1
10:0	RW	0x0	axi auto sideband control THD0 mem0 cnt >= THD0 ,will set sideband to 0

**Table 7-406 MIPI\_TOP\_FR\_WR\_UGT\_CTRL3 0xd3**

isp\_core fr uv axi wr control

Bit(s)	R/W	Default	Description
30	RW	0x0	axi auto sideband enable
26:16	RW	0x0	axi auto sideband control THD1 mem1 cnt < THD1 ,will set sideband to 1
10:0	RW	0x0	axi auto sideband control THD0 mem1 cnt >= THD0 ,will set sideband to 0

**Table 7-407 MIPI\_TOP\_FR\_WR\_UGT\_ST 0xd4**

isp\_core fr axi wr control

Bit(s)	R/W	Default	Description
25:22	R	0x0	axi qos
21	R	0x0	axi side_band
20	R	0x0	axi urgent
19:10	R	0x0	mem1_cnt
9:0	R	0x0	mem0_cnt

MIPI\_TOP\_DS\_WR\_UGT\_CTRL0 0xd5

isp\_core ds axi wr control,same as MIPI\_TOP\_FR\_WR\_UGT\_CTRL0

MIPI\_TOP\_DS\_WR\_UGT\_CTRL1 0xd6

isp\_core ds uv axi wr control,same as MIPI\_TOP\_FR\_WR\_UGT\_CTRL1

MIPI\_TOP\_DS\_WR\_UGT\_CTRL2 0xd7

isp\_core ds axi wr control,same as MIPI\_TOP\_FR\_WR\_UGT\_CTRL2

MIPI\_TOP\_DS\_WR\_UGT\_CTRL3 0xd8

isp\_core ds uv axi wr control,same as MIPI\_TOP\_FR\_WR\_UGT\_CTRL3

MIPI\_TOP\_DS\_WR\_UGT\_ST 0xd9

isp\_core ds axi wr control,same as MIPI\_TOP\_FR\_WR\_UGT\_ST

**Table 7-408 MIPI\_TOP\_TNR\_WR\_UGT\_CTRL0 0xda**

isp\_core tnr msb axi wr control

Bit(s)	R/W	Default	Description
31	RW	0x0	auto_urgent_enable
30	RW	0x0	axi auto urgent enable
26:16	RW	0x0	axi auto urgent control THD1 mem0 cnt < THD1 ,will set urgent to 1
10:0	RW	0x0	axi auto urgent control THD0 mem0 cnt >= THD0 ,will set urgent to 0

**Table 7-409 MIPI\_TOP\_TNR\_WR\_UGT\_CTRL1 0xdb**

isp\_core tnr lsb axi wr control

Bit(s)	R/W	Default	Description
30	RW	0x0	axi auto urgent enable
26:16	RW	0x0	axi auto urgent control THD1 mem1 cnt < THD1 ,will set urgent to 1
10:0	RW	0x0	axi auto urgent control THD0 mem1 cnt >= THD0 ,will set urgent to 0

**Table 7-410 MIPI\_TOP\_TNR\_WR\_UGT\_CTRL2 0xdc**

isp\_core tnr msb axi wr control

Bit(s)	R/W	Default	Description
31	RW	0x0	auto_sideband_enable
30	RW	0x0	axi auto sideband enable
26:16	RW	0x0	axi auto sideband control THD1 mem0 cnt < THD1 ,will set sideband to 1
10:0	RW	0x0	axi auto sideband control THD0 mem0 cnt >= THD0 ,will set sideband to 0

**Table 7-411 MIPI\_TOP\_TNR\_WR\_UGT\_CTRL3 0xdd**

isp\_core tnr lsb axi wr control

Bit(s)	R/W	Default	Description
30	RW	0x0	axi auto sideband enable
26:16	RW	0x0	axi auto sideband control THD1 mem1 cnt < THD1 ,will set sideband to 1
10:0	RW	0x0	axi auto sideband control THD0 mem1 cnt >= THD0 ,will set sideband to 0

**Table 7-412 MIPI\_TOP\_TNR\_WR\_UGT\_ST 0xde**

isp\_core tnr axi wr control

Bit(s)	R/W	Default	Description
21	R	0x0	axi side_band
20	R	0x0	axi urgent
19:10	R	0x0	mem1_cnt
9:0	R	0x0	mem0_cnt

**Table 7-413 MIPI\_TOP\_TNR\_RD\_UGT\_CTRL0 0xdf**

isp\_core tnr msb axi rd control

Bit(s)	R/W	Default	Description
31	RW	0x0	auto_urgent_enable
30	RW	0x0	axi auto urgent enable
26:16	RW	0x0	axi auto urgent control THD1 mem0 cnt > THD1 ,will set urgent to 1
10:0	RW	0x0	axi auto urgent control THD0 mem0 cnt <= THD0 ,will set urgent to 0

**Table 7-414 MIPI\_TOP\_TNR\_RD\_UGT\_CTRL1 0xe0**

isp\_core tnr lsb axi rd control

Bit(s)	R/W	Default	Description
30	RW	0x0	axi auto urgent enable
26:16	RW	0x0	axi auto urgent control THD1 mem1 cnt > THD1 ,will set urgent to 1
10:0	RW	0x0	axi auto urgent control THD0 mem1 cnt <= THD0 ,will set urgent to 0

**Table 7-415 MIPI\_TOP\_TNR\_RD\_UGT\_CTRL2 0xe1**

isp\_core tnr msb axi rd control

Bit(s)	R/W	Default	Description
31	RW	0x0	auto_sideband_enable
30	RW	0x0	axi auto sideband enable
26:16	RW	0x0	axi auto sideband control THD1 mem0 cnt > THD1 ,will set sideband to 1
10:0	RW	0x0	axi auto sideband control THD0 mem0 cnt <= THD0 ,will set sideband to 0

**Table 7-416 MIPI\_TOP\_TNR\_RD\_UGT\_CTRL3 0xe2**

isp\_core tnr lsb axi rd control

Bit(s)	R/W	Default	Description
30	RW	0x0	axi auto sideband enable
26:16	RW	0x0	axi auto sideband control THD1 mem1 cnt > THD1 ,will set sideband to 1
10:0	RW	0x0	axi auto sideband control THD0 mem1 cnt <= THD0 ,will set sideband to 0

**Table 7-417 MIPI\_TOP\_TNR\_RD\_UGT\_ST 0xe3**

isp\_core tnr axi rd control

Bit(s)	R/W	Default	Description
21	R	0x0	axi side_band
20	R	0x0	axi urgent
19:10	R	0x0	mem1_cnt
9:0	R	0x0	mem0_cnt

**Table 7-418 MIPI\_TOP\_ISP\_FR\_WBUS\_PROT\_CNTL 0xf6**

isp\_core isp axi wr protect control

Bit(s)	R/W	Default	Description
3	RW	0x0	protect enable
1	RW	0x0	clear axi bus hold status
0	RW	0x0	protect start

**Table 7-419 MIPI\_TOP\_ISP\_FR\_WBUS\_PROT\_CNTL 0xf7**

isp\_core isp axi wr protect control

Bit(s)	R/W	Default	Description
17	R	0x0	all cmd done status
16	R	0x0	mask_en
15:0	R	0x0	cmd number not finish

**Table 7-420 MIPI\_TOP\_ISP\_SIDEHAND\_CNTL0 0xe5**

Bit(s)	R/W	Default	Description
31:0	RW	0x0	all axi wr bus sideband enable

**Table 7-421 MIPI\_TOP\_ISP\_SIDEHAND\_CNTL1 0xe6**

Bit(s)	R/W	Default	Description
31:0	RW	0x0	all axi rd bus sideband enable

**Table 7-422 MIPI\_TOP\_ISP\_PENDING\_MASK0 0xf0**

Bit(s)	R/W	Default	Description
31:0	RW	0x0	pending mask for irq[31:0]

**Table 7-423 MIPI\_TOP\_ISP\_PENDING\_MASK1 0xf1**

Bit(s)	R/W	Default	Description
31:0	RW	0x0	pending mask for irq[63:32]

**Table 7-424 MIPI\_TOP\_ISP\_PENDING\_MASK2 0xf2**

Bit(s)	R/W	Default	Description
31:0	RW	0x0	pending mask for irq[95:64]

**Table 7-425 MIPI\_TOP\_ISP\_PENDING0 0xf0**

Bit(s)	R/W	Default	Description
31:0	RW	0x0	irq pending [31:0],write 1 to clear pending bit



Table 7-426 MIPI\_TOP\_ISP\_PENDING1 0xf1

Bit(s)	R/W	Default	Description
31:0	RW	0x0	irq pending [63:32],write 1 to clear pending bit

Table 7-427 MIPI\_TOP\_ISP\_PENDING2 0xf2

Bit(s)	R/W	Default	Description
31:0	RW	0x0	irq pending [95:64],write 1 to clear pending bit

Table 7-428 top\_isp\_pending list:

bit	irq source	bit	irq source	bit	irq source
95	NA	63	NA	31	fe_0_pending[5]:frame isp data have been write to ddr
94	NA	62	NA	30	fe_0_pending[4]:frame end for ddr mode data
93	NA	61	NA	29	fe_0_pending[3]:frame start for ddr mode data
92	NA	60	NA	28	fe_0_pending[2]:frame ddr data have been write to ddr
91	NA	59	tnr_de_pending	27	fe_0_pending[1]:frame end for direct mode data
90	NA	58	adapt_de_pending	26	fe_0_pending[0]:frame start for direct mode data
89	NA	57	crop_mif_pending	25	ddr_rd0_lbuf_wr_ov_pending When lbuf0 overflow will set this pending
88	NA	56	ds2_pending	24	ddr_rd0_dol_hang_pending When ddr_rd in dol mode,will set this pending when the next frame_sync come before ddr_rd send out all valid pixel
87	NA	55	ds1_pending	23	ddr_rd0_dol_more_line_pending When ddr_rd in dol mode and with ddr mode,will set this pending when ddr_rd received more lines than the setting for ddr_rd0
86	isp_core_pending [22]	54	ds0_pending	22	ddr_rd0_dol_less_line_pending When ddr_rd in dol mode and with ddr mode,will set this pending at 2 cases: 1 ddr_rd received the next frame_vsync and the number of lines is less than the setting for ddr_rd0 2 when set dol_time_out_en=1,and ddr_rd did not received valid line in the setting time period for ddr_rd0
85	isp_core_pending [21]	53	ahb_dma_pending	21	ddr_rd0_frame_rd_done_pending When ddr_rd0 read over one frame with setting in ddr_rd0,will set this pending
84	isp_core_pending [20]	52	apb_dma_pending	20	ddr_rd0_frame_v_count_pending When ddr_rd0 has read "line_threshold" lines,will set this pending

bit	irq source	bit	irq source	bit	irq source
83	isp_core_pending [19]	51	isp_core_ds_wbus_noblock_pending	19	ddr_rd1_lbuf_wr_ov_pending When lbuf1 overflow will set this pending
82	isp_core_pending [18]	50	isp_core_fr_wbus_noblock_pending	18	ddr_rd1_dol_hang_pending When ddr_rd in dol mode,will set this pending when the next frame_sync come before ddr_rd send out all valid pixel
81	isp_core_pending [17]	49	fe_3_pending[5]:frame isp data have been write to ddr	17	ddr_rd1_dol_more_line_pending When ddr_rd in dol mode and with ddr mode,will set this pending when ddr_rd received more lines than the setting for ddr_rd1
80	isp_core_pending [16]	48	fe_3_pending[4]:frame end for ddr mode data	16	ddr_rd1_dol_less_line_pending When ddr_rd in dol mode and with ddr mode,will set this pending at 2 cases: 1 ddr_rd received the next frame_vsync and the number of lines is less than the setting for ddr_rd1 2 when set dol_time_out_en=1,and ddr_rd did not received valid line in the setting time period for ddr_rd1
79	isp_core_pending [15]	47	fe_3_pending[3]:frame start for ddr mode data	15	ddr_rd1_frame_rd_done_pending When ddr_rd1 read over one frame with setting in ddr_rd1,will set this pending
78	isp_core_pending [14]	46	fe_3_pending[2]:frame ddr data have been write to ddr	14	ddr_rd1_frame_v_count_pending When ddr_rd1 has read "line_threshold" lines,will set this pending
77	isp_core_pending [13]	45	fe_3_pending[1]:frame end for direct mode data	13	alig_hsync_num_irq_set_pending When Line irq enable,will set this pending at line start at the setting number of line
76	isp_core_pending [12]	44	fe_3_pending[0]:frame start for direct mode data	12	alig_lane0_dir_data_timeout_pending When lane0 valid,will set this pending if did not receive valid data in valid period for 65535 vclk cycles
75	isp_core_pending [11]	43	fe_2_pending[5]:frame isp data have been write to ddr	11	alig_lane1_dir_data_timeout_pending When lane1 valid,will set this pending if did not receive valid data in valid period for 65535 vclk cycles
74	isp_core_pending [10]	42	fe_2_pending[4]:frame end for ddr mode data	10	alig_data_less_timeout_pending When inc_mask_timeout_en is valid, will set this pending when no valid data from pixel last inc_mask_timeout x vclk cycles
73	isp_core_pending [9]	41	fe_2_pending[3]:frame start for ddr mode data	9	alig_frame_end_pending When alignment send out the entire frame with camera setting in alignment (both valid period and blank period),will set this pending
72	isp_core_pending [8]	40	fe_2_pending[2]:frame ddr data have been write to ddr	8	alig_frame_start_pending alignment output valid pos set this pending
71	isp_core_pending [7]	39	fe_2_pending[1]:frame end for direct mode data	7	alig_lane0_pixel_less_pending When lane0 is in direct mode,will set this pending when the number of pixel in a line from source is less than camera setting in alignment

bit	irq source	bit	irq source	bit	irq source
70	isp_core_pending [6]	38	fe_2_pending[0]:frame start for direct mode data	6	alig_lane0_pixel_exc_pending When lane0 is in direct mode,will set this pending when the number of pixel in a line from source is more than camera setting in alignment
69	isp_core_pending [5]	37	fe_1_pending[5]:frame isp data have been write to ddr	5	alig_lane0_line_less_pending When lane0 is in direct mode,and inc_mask_timeout_en is valid, will set this pending when the number of line from source is less than camera setting in alignment
68	isp_core_pending [4]	36	fe_1_pending[4]:frame end for ddr mode data	4	alig_lane0_line_exc_pending When lane0 is in direct mode,will set this pending when the number of line from source is more than camera setting in alignment
67	isp_core_pending [3]	35	fe_1_pending[3]:frame start for ddr mode data	3	alig_lane1_pixel_less_pending When lane1 is in direct mode,will set this pending when the number of pixel in a line from source is less than camera setting in alignment
66	isp_core_pending [2]	34	fe_1_pending[2]:frame ddr data have been write to ddr	2	alig_lane1_pixel_exc_pending When lane1 is in direct mode,will set this pending when the number of pixel in a line from source is more than camera setting in alignment
65	isp_core_pending [1]	33	fe_1_pending[1]:frame end for direct mode data	1	alig_lane1_line_less_pending When lane1 is in direct mode,and inc_mask_timeout_en is valid, will set this pending when the number of line from source is less than camera setting in alignment
64	isp_core_pending [0]	32	fe_1_pending[0]:frame start for direct mode data	0	alig_lane1_line_exc_pending When lane1 is in direct mode,will set this pending when the number of line from source is more than camera setting in alignment

**Table 7-429 MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ST 0x0**

Bits	R/W	Default	Description
31:0	R/W	0	ping frame0 start

**Table 7-430 MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ED 0x1**

Bits	R/W	Default	Description
31:0	R/W	0	ping frame0 end

**Table 7-431 MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ST 0x2**

Bits	R/W	Default	Description
31:0	R/W	0	ping frame1 start only used when y,chroma seperated ladder, for chroma channel

Table 7-432 MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ED 0x3

Bits	R/W	Default	Description
31:0	R/W	0	ping frame1 end

Table 7-433 MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ST 0x4

Bits	R/W	Default	Description
31:0	R/W	0	pong frame0 start

Table 7-434 MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ED 0x5

Bits	R/W	Default	Description
31:0	R/W	0	pong frame0 end

Table 7-435 MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ST 0x6

Bits	R/W	Default	Description
31:0	R/W	0	pong frame1 end

Table 7-436 MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ED 0x7

Bits	R/W	Default	Description
31:0	R/W	0	pong frame1 end

Table 7-437 MIPI\_BL\_ISP\_FR\_FRAME\_TH 0x8

Bits	R/W	Default	Description
31	R/W	0	awaddr handshake and total frame count calculation reset
30	R/W	0	frame count reset
29	R/W	0	drop mode count reset
28	R/W	0	total buffer count reset
27:7	R/W	0	reserved
6	R/W	0	pre frame ping/pong status reset
5	R/W	0	frame change when address jump between Y channel and Chroma channel
4	R/W	0	hold frame change when address jump between Ping/Pong
3:1	R/W	0	reserved
0	R/W	0	fake delay count reset

**Table 7-438 MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0 0x9**

Bits	R/W	Default	Description
31:0	R/W	0	frame buffer target buffer start address

**Table 7-439 MIPI\_BL\_ISP\_FR\_FRAME\_SIZE0 0xa**

Bits	R/W	Default	Description
31:0	R/W	0	frame size stride in buffer

**Table 7-440 MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE0 0xb**

Bits	R/W	Default	Description
31:0	R/W	0	frame buffer size

**Table 7-441 MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR1 0xc**

Bits	R/W	Default	Description
31:0	R/W	0	frame1 buffer target buffer start address, only used when y,chroma seperated ladder, for chroma channel

**Table 7-442 MIPI\_BL\_ISP\_FR\_FRAME\_SIZE1 0xd**

Bits	R/W	Default	Description
31:0	R/W	0	frame1 size stride in buffer

**Table 7-443 MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE1 0xe**

Bits	R/W	Default	Description
31:0	R/W	0	frame1 buffer size

**Table 7-444 MIPI\_BL\_ISP\_FR\_CTRL0 0xf**

Bits	R/W	Default	Description
31:16	R/W	0	frame count thm1 for stop_th mode
15	R/W	0	ladder async reset
14	R/W	0	ladder sw hold handshake
13	R/W	0	ladder latch mode
12	R/W	0	ladder sw bypass mode
11:6	R/W	0	reserved
5	R/W	0	ladder drop mode enable
4	R/W	0	ladder stop next frame
3	R/W	0	ladder reset stop_th count
2	R/W	0	ladder stop_th mode enable

Bits	R/W	Default	Description
1	R/W	0	ladder based address reset
0	R/W	0	ladder enable

Table 7-445 MIPI\_BL\_ISP\_FR\_CTRL1 0x10

Bits	R/W	Default	Description
31:28	R/W	0	fake handshake wready delay threshold
27:24	R/W	0	fake handshake bvalid delay threshold
23:16	R/W	0	drop mode jump threshold between write frames
15:8	R/W	0	drop mode write threshold between jump frames
7:0	R/W	0	drop mode skip threshold when drop mode beginning

Table 7-446 MIPI\_BL\_ISP\_FR\_FRAME\_COUNT 0x11

Bits	R/W	Default	Description
31:16	R	0	reserved
15:0	R	0	total frame count, can be reset by frame_th[31]

Table 7-447 MIPI\_BL\_ISP\_FR\_AWADDR\_O 0x12

Bits	R/W	Default	Description
31:0	R	0	last output handshake address

Table 7-448 MIPI\_BL\_ISP\_FR\_AWADDR\_I 0x13

Bits	R/W	Default	Description
31:0	R	0	last input handshake address

Table 7-449 MIPI\_BL\_ISP\_FR\_STAT0 0x14

Bits	R/W	Default	Description
31:21	R	0	reserved
20	R	0	output ports handshake idle
19	R	0	work in stop_th mode
18	R	0	work in stop mode
17:16	R	0	drop mode, 0 for idle, 1 for skip, 2 for write, 3 for jump
15:0	R	0	error address handshake count

**Table 7-450 MIPI\_BL\_ISP\_FR\_STAT1 0x15**

Bits	R/W	Default	Description
31:24	R	0	outside ping/pong frame count
23:12	R	0	ping frame count
11:0	R	0	pong frame count

**Table 7-451 MIPI\_BL\_ISP\_FR\_STAT2 0x16**

Bits	R/W	Default	Description
31:24	R	0	drop mode skip frame count
23:12	R	0	drop mode write frame count
11:0	R	0	drop mode jump frame count

**Table 7-452 MIPI\_BL\_ISP\_FR\_STAT3 0x17**

Bits	R/W	Default	Description
31:24	R	0	frame loop back count
23	R	0	reserved
22	R	0	master awvalid total > wvalid total
21	R	0	master awvalid total > bvalid total
20	R	0	master wvalid total > bvalid total
19	R	0	slave awvalid total > wvalid total
18	R	0	slave awvalid total > bvalid total
17	R	0	slave wvalid total > bvalid total
16:12	R	0	fake hadkshake awvalid-bvalid count[4:0]
11	R	0	reserved
10:0	R	0	normal handshake awvalid-bvalid count[10:0]

**Table 7-453 MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0P 0x18**

Bits	R/W	Default	Description
31:0	R/W	0	frame bufferP target buffer start address, should be the same with buffer0 unless p/p buffer

MIPI\_BL\_ISP\_DS\_PING\_ADDR0\_ST 0x20

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ST

MIPI\_BL\_ISP\_DS\_PING\_ADDR0\_ED 0x21

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ED

MIPI\_BL\_ISP\_DS\_PING\_ADDR1\_ST 0x22

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ST

MIPI\_BL\_ISP\_DS\_PING\_ADDR1\_ED 0x23

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ED

MIPI\_BL\_ISP\_DS\_PONG\_ADDR0\_ST 0x24

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ST

MIPI\_BL\_ISP\_DS\_PONG\_ADDR0\_ED 0x25

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ED

MIPI\_BL\_ISP\_DS\_PONG\_ADDR1\_ST 0x26

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ST

MIPI\_BL\_ISP\_DS\_PONG\_ADDR1\_ED 0x27

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ED

MIPI\_BL\_ISP\_DS\_FRAME\_TH 0x28

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_TH

MIPI\_BL\_ISP\_DS\_FRAME\_BUFFER\_START\_ADDR0 0x29

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0

MIPI\_BL\_ISP\_DS\_FRAME\_SIZE0 0x2a

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_SIZE0

MIPI\_BL\_ISP\_DS\_FRAME\_BUF\_SIZE0 0x2b

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE0

MIPI\_BL\_ISP\_DS\_FRAME\_BUFFER\_START\_ADDR1 0x2c

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR1

MIPI\_BL\_ISP\_DS\_FRAME\_SIZE1 0x2d

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_SIZE1

MIPI\_BL\_ISP\_DS\_FRAME\_BUF\_SIZE1 0x2e

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE1

MIPI\_BL\_ISP\_DS\_CTRL0 0x2f

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_CTRL0

MIPI\_BL\_ISP\_DS\_CTRL1 0x30

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_CTRL1

MIPI\_BL\_ISP\_DS\_FRAME\_COUNT 0x31

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_COUNT

MIPI\_BL\_ISP\_DS\_AWADDR\_O 0x32

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_AWADDR\_O

MIPI\_BL\_ISP\_DS\_AWADDR\_I 0x33

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_AWADDR\_I

MIPI\_BL\_ISP\_DS\_STAT0 0x34

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_STAT0

MIPI\_BL\_ISP\_DS\_STAT1 0x35

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_STAT1



MIPI\_BL\_ISP\_DS\_STAT2 0x36

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_STAT2

MIPI\_BL\_ISP\_DS\_STAT3 0x37

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_STAT3

MIPI\_BL\_ISP\_DS\_FRAME\_BUFFER\_START\_ADDR0P 0x38

for isp\_core ds ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0P

MIPI\_BL\_DS0\_PING\_ADDR0\_ST 0x40

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ST

MIPI\_BL\_DS0\_PING\_ADDR0\_ED 0x41

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ED

MIPI\_BL\_DS0\_PING\_ADDR1\_ST 0x42

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ST

MIPI\_BL\_DS0\_PING\_ADDR1\_ED 0x43

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ED

MIPI\_BL\_DS0\_PONG\_ADDR0\_ST 0x44

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ST

MIPI\_BL\_DS0\_PONG\_ADDR0\_ED 0x45

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ED

MIPI\_BL\_DS0\_PONG\_ADDR1\_ST 0x46

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ST

MIPI\_BL\_DS0\_PONG\_ADDR1\_ED 0x47

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ED

MIPI\_BL\_DS0\_FRAME\_TH 0x48

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_TH

MIPI\_BL\_DS0\_FRAME\_BUFFER\_START\_ADDR0 0x49

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0

MIPI\_BL\_DS0\_FRAME\_SIZE0 0x4a

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_SIZE0

MIPI\_BL\_DS0\_FRAME\_BUF\_SIZE0 0x4b

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE0

MIPI\_BL\_DS0\_FRAME\_BUFFER\_START\_ADDR1 0x4c

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR1

MIPI\_BL\_DS0\_FRAME\_SIZE1 0x4d

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_SIZE1

MIPI\_BL\_DS0\_FRAME\_BUF\_SIZE1 0x4e

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE1

MIPI\_BL\_DS0\_CTRL0 0x4f

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_CTRL0

MIPI\_BL\_DS0\_CTRL1 0x50

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_CTRL1

MIPI\_BL\_DS0\_FRAME\_COUNT 0x51

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_COUNT

MIPI\_BL\_DS0\_AWADDR\_O 0x52

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_AWADDR\_O

MIPI\_BL\_DS0\_AWADDR\_I 0x53

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_AWADDR\_I

MIPI\_BL\_DS0\_STAT0 0x54

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_STAT0

MIPI\_BL\_DS0\_STAT1 0x55

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_STAT1

MIPI\_BL\_DS0\_STAT2 0x56

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_STAT2

MIPI\_BL\_DS0\_STAT3 0x57

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_STAT3

MIPI\_BL\_DS0\_FRAME\_BUFFER\_START\_ADDR0P 0x58

for DS0 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0P

MIPI\_BL\_DS1\_PING\_ADDR0\_ST 0x60

for DS1 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ST

MIPI\_BL\_DS1\_PING\_ADDR0\_ED 0x61

for DS1 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ED

MIPI\_BL\_DS1\_PING\_ADDR1\_ST 0x62

for DS1 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ST

MIPI\_BL\_DS1\_PING\_ADDR1\_ED 0x63

for DS1 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ED

MIPI\_BL\_DS1\_PONG\_ADDR0\_ST 0x64

for DS1 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ST

MIPI\_BL\_DS1\_PONG\_ADDR0\_ED 0x65

for DS1 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ED

MIPI\_BL\_DS1\_PONG\_ADDR1\_ST 0x66

for DS1 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ST

MIPI\_BL\_DS1\_PONG\_ADDR1\_ED 0x67

for DS1 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ED

MIPI\_BL\_DS1\_FRAME\_TH 0x68

for DS1 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_TH

MIPI\_BL\_DS1\_FRAME\_BUFFER\_START\_ADDR0 0x69

for DS1 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0

MIPI\_BL\_DS1\_FRAME\_SIZE0 0x6a  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_SIZE0

MIPI\_BL\_DS1\_FRAME\_BUF\_SIZE0 0x6b  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE0

MIPI\_BL\_DS1\_FRAME\_BUFFER\_START\_ADDR1 0x6c  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR1

MIPI\_BL\_DS1\_FRAME\_SIZE1 0x6d  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_SIZE1

MIPI\_BL\_DS1\_FRAME\_BUF\_SIZE1 0x6e  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE1

MIPI\_BL\_DS1\_CTRL0 0x6f  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_CTRL0

MIPI\_BL\_DS1\_CTRL1 0x70  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_CTRL1

MIPI\_BL\_DS1\_FRAME\_COUNT 0x71  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_COUNT

MIPI\_BL\_DS1\_AWADDR\_O 0x72  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_AWADDR\_O

MIPI\_BL\_DS1\_AWADDR\_I 0x73  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_AWADDR\_I

MIPI\_BL\_DS1\_STAT0 0x74  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_STAT0

MIPI\_BL\_DS1\_STAT1 0x75  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_STAT1

MIPI\_BL\_DS1\_STAT2 0x76  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_STAT2

MIPI\_BL\_DS1\_STAT3 0x77  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_STAT3

MIPI\_BL\_DS1\_FRAME\_BUFFER\_START\_ADDR0P 0x78  
for DS1 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0P

MIPI\_BL\_DS2\_PING\_ADDR0\_ST 0x80  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ST

MIPI\_BL\_DS2\_PING\_ADDR0\_ED 0x81  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ED

MIPI\_BL\_DS2\_PING\_ADDR1\_ST 0x82  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ST

MIPI\_BL\_DS2\_PING\_ADDR1\_ED 0x83  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ED

MIPI\_BL\_DS2\_PONG\_ADDR0\_ST 0x84  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ST

MIPI\_BL\_DS2\_PONG\_ADDR0\_ED 0x85  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ED

MIPI\_BL\_DS2\_PONG\_ADDR1\_ST 0x86  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ST

MIPI\_BL\_DS2\_PONG\_ADDR1\_ED 0x87  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ED

MIPI\_BL\_DS2\_FRAME\_TH 0x88  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_TH

MIPI\_BL\_DS2\_FRAME\_BUFFER\_START\_ADDR0 0x89  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0

MIPI\_BL\_DS2\_FRAME\_SIZE0 0x8a  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_SIZE0

MIPI\_BL\_DS2\_FRAME\_BUF\_SIZE0 0x8b  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE0

MIPI\_BL\_DS2\_FRAME\_BUFFER\_START\_ADDR1 0x8c  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR1

MIPI\_BL\_DS2\_FRAME\_SIZE1 0x8d  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_SIZE1

MIPI\_BL\_DS2\_FRAME\_BUF\_SIZE1 0x8e  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE1

MIPI\_BL\_DS2\_CTRL0 0x8f  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_CTRL0

MIPI\_BL\_DS2\_CTRL1 0x90  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_CTRL1

MIPI\_BL\_DS2\_FRAME\_COUNT 0x91  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_COUNT

MIPI\_BL\_DS2\_AWADDR\_O 0x92  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_AWADDR\_O

MIPI\_BL\_DS2\_AWADDR\_I 0x93  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_AWADDR\_I

MIPI\_BL\_DS2\_STAT0 0x94  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_STAT0

MIPI\_BL\_DS2\_STAT1 0x95  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_STAT1

MIPI\_BL\_DS2\_STAT2 0x96  
for DS2 ,description same as MIPI\_BL\_ISP\_FR\_STAT2

MIPI\_BL\_DS2\_STAT3 0x97

for DS2 ,description same as MIPI\_BL\_ISP\_FR\_STAT3

MIPI\_BL\_DS2\_FRAME\_BUFFER\_START\_ADDR0P 0x98

for DS2 ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0P

MIPI\_BL\_CR0P\_MIF\_PING\_ADDR0\_ST 0xa0

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ST

MIPI\_BL\_CR0P\_MIF\_PING\_ADDR0\_ED 0xa1

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR0\_ED

MIPI\_BL\_CR0P\_MIF\_PING\_ADDR1\_ST 0xa2

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ST

MIPI\_BL\_CR0P\_MIF\_PING\_ADDR1\_ED 0xa3

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_PING\_ADDR1\_ED

MIPI\_BL\_CR0P\_MIF\_PONG\_ADDR0\_ST 0xa4

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ST

MIPI\_BL\_CR0P\_MIF\_PONG\_ADDR0\_ED 0xa5

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR0\_ED

MIPI\_BL\_CR0P\_MIF\_PONG\_ADDR1\_ST 0xa6

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ST

MIPI\_BL\_CR0P\_MIF\_PONG\_ADDR1\_ED 0xa7

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_PONG\_ADDR1\_ED

MIPI\_BL\_CR0P\_MIF\_FRAME\_TH 0xa8

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_TH

MIPI\_BL\_CR0P\_MIF\_FRAME\_BUFFER\_START\_ADDR0 0xa9

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0

MIPI\_BL\_CR0P\_MIF\_FRAME\_SIZE0 0xaa

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_SIZE0

MIPI\_BL\_CR0P\_MIF\_FRAME\_BUF\_SIZE0 0xab

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE0

MIPI\_BL\_CR0P\_MIF\_FRAME\_BUFFER\_START\_ADDR1 0xac

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR1

MIPI\_BL\_CR0P\_MIF\_FRAME\_SIZE1 0xad

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_SIZE1

MIPI\_BL\_CR0P\_MIF\_FRAME\_BUF\_SIZE1 0xae

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUF\_SIZE1

MIPI\_BL\_CR0P\_MIF\_CTRL0 0xaf

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_CTRL0

MIPI\_BL\_CR0P\_MIF\_CTRL1 0xb0

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_CTRL1

MIPI\_BL\_CR0P\_MIF\_FRAME\_COUNT 0xb1

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_COUNT

MIPI\_BL\_CR0P\_MIF\_AWADDR\_O 0xb2

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_AWADDR\_O

MIPI\_BL\_CR0P\_MIF\_AWADDR\_I 0xb3

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_AWADDR\_I

MIPI\_BL\_CR0P\_MIF\_STAT0 0xb4

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_STAT0

MIPI\_BL\_CR0P\_MIF\_STAT1 0xb5

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_STAT1

MIPI\_BL\_CR0P\_MIF\_STAT2 0xb6

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_STAT2

MIPI\_BL\_CR0P\_MIF\_STAT3 0xb7

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_STAT3

MIPI\_BL\_CR0P\_MIF\_FRAME\_BUFFER\_START\_ADDR0P 0xb8

for crop\_mif ,description same as MIPI\_BL\_ISP\_FR\_FRAME\_BUFFER\_START\_ADDR0P

## 7.12.5.2 ISP Compress Core Registers

### ISP Compress Core Registers

The base address of compress core are listed below:

- Adapt Sub0 Compress Core: 0xFE3B4400
- TNR Sub0 Compress Core: 0xFE3B4c00
- TNR Sub1 Compress Core: 0xFE3B5000

**Table 7-454 ISP\_ENC\_LOSS\_CTRL 0x00**

Bit(s)	R/W	Default	Description
31:24	R/W	6	reg_hblank1: should >=6, blank cycle between two line
23:16	R/W	6	reg_hblank0: should >=6
15:8	R/W	0	reg_sync_ctrl: shaddow control, 0 to shaddow the register
7:4	R/W	0	reg_use_inter_fmt: 0 to use external format setting: for T7, it need set to 4'b1111
3	R/W	0	reg_status_rdbk_mode: 0 to shaddow the ro status register
2:1	R/W	3	reg_stats_en:[0] to enable max_err/err_acc stats; [1] to enable error_count2 stats
0	R/W	1	reg_enable: 1 to enable

**Table 7-455 ISP\_ENC\_LOSS\_FRAME\_HOLD 0x01**

Bit(s)	R/W	Default	Description
31:0	R/W	1	reg_frame_hold_nums: hold clock number from frame_rst to wait register ready Note: set bit[31] to 1 indicates the compress would start till RDMA transfer finished or write [31:0] to 0xffffffff

**Table 7-456 ISP\_ENC\_LOSS\_GCLK\_CTRL 0x02**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_gclk_ctrl: gated-clock control

**Table 7-457 ISP\_ENC\_LOSS\_RO\_CODEC\_STATUS 0x03**

Bit(s)	R/W	Default	Description
31:0	R/W	0	ro_codec_status:codec status, write 1 to clear

**Table 7-458 ISP\_ENC\_LOSS\_LOSS\_MISC 0x04**

Bit(s)	R/W	Default	Description
31:0	R/W	0	[4] MISC control register : reg_not_wait_input4busy : reg_busy_status_use4abort

**Table 7-459 ISP\_ENC\_LOSS\_BASIS 0x08**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_lossless_en: 1 to use lossless mode
28:24	R/W	12	reg_src_bit_depth: source data bit depth from sensor
23:21	R/W	1	reg_raw_mode:0: mono, 1:Bayer, 2/3: RGBIR2x2 4:RGBIR4x4
20	R/W	0	reg_mono_comp_mode:compression mode of mono data : 0 compression in Y, 1: compression in bayer
19:18	R/W	1	reg_xphase_ofst:phase offset in x dimension
17:16	R/W	0	reg_yphase_ofst:phase offset in y dimension
15:13	R/W		reserved
12:4	R/W	80	reg_ratio_bppx16: bits per pixel x16, fracbits_bpp = x/16
2:0	R/W	2	reg_comp_chn_size: color space mode for snr1 input

**Table 7-460 ISP\_ENC\_LOSS\_PIC\_SIZE 0x09**

Bit(s)	R/W	Default	Description
31:16	R/W	1080	reg_pic_ysize:picture verital size
15:0	R/W	1920	reg_pic_xsize:picture horizontal size

**Table 7-461 ISP\_ENC\_LOSS\_SLICE\_SIZE 0x0a**

Bit(s)	R/W	Default	Description
31	R/W	1	reg_use_sw_preslc_bitaccum
29:28	R/W	0	reg_slice_num_mode: 0 slice num 1, 1:slice num 2 2:slice num 4 3:slice num 8
15:0	R/W	1080	reg_slice_ysize_0:vertical slice window size for slice0

**Table 7-462 ISP\_ENC\_LOSS\_SLICE\_SIZE\_1 0x0b**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_slice_ysize_2:vertical slice window size for slice2
15:0	R/W	0	reg_slice_ysize_1:vertical slice window size for slice1

**Table 7-463 ISP\_ENC\_LOSS\_SLICE\_SIZE\_2 0x0c**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_slice_ysize_4:vertical slice window size for slice4
15:0	R/W	0	reg_slice_ysize_3:vertical slice window size for slice3

**Table 7-464 ISP\_ENC\_LOSS\_SLICE\_SIZE\_3 0x0d**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_slice_ysize_6:vertical slice window size for slice6
15:0	R/W	0	reg_slice_ysize_5:vertical slice window size for slice5

**Table 7-465 ISP\_ENC\_LOSS\_PRESL\_LAST\_BITS 0x0e**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_pre_slc_bitaccum: pre slice last bits add to next slice

**Table 7-466 ISP\_ENC\_LOSS\_PRESL\_FIFO\_LEVEL 0x0f**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_use_sw_preslc_fifolevel
15:0	R/W	16	reg_pre_slc_fifolevel:pre slice fifolevel add to next slice

**Table 7-467 ISP\_ENC\_LOSS\_DEFAULT\_VALUE 0x10**

Bit(s)	R/W	Default	Description
31:16	R/W	2048	reg_default_value_0:default value of comp 0
15:0	R/W	2048	reg_default_value_1:default value of comp 1



**Table 7-468 ISP\_ENC\_LOSS\_DEFAULT\_VALUE\_1 0x11**

Bit(s)	R/W	Default	Description
15:0	R/W	2048	reg_default_value_2:default value of comp 2

**Table 7-469 ISP\_ENC\_LOSS\_ERROR\_COUNT\_THD 0x12**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_error_count_region_half_en:enable of static the left half of the image,1: only static left half of the image, 0: all image
15:0	R/W	100	reg_error_count_thd:threshold to count number of error bigger thd

**Table 7-470 ISP\_ENC\_LOSS\_DEBUG 0x13**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_debug_mode: 1 to oupput qlevel for decoder only
24:16	R/W	0	reg_dbg_qlevel_2: [24] =1 to force qlevel_2 to reg_dbg_qlevel_2[4:0]
15:8	R/W	0	reg_dbg_qlevel_1: [15] =1 to force qlevel_1 to reg_dbg_qlevel_1[4:0]
7:0	R/W	0	reg_dbg_qlevel_0: [7] =1 to force qlevel_0 to reg_dbg_qlevel_0[4:0]

**Table 7-471 ISP\_ENC\_LOSS\_GLOBAL\_PHASE\_LUT 0x14**

Bit(s)	R/W	Default	Description
30:28	R/W	3	reg_raw_phslut_15: raw phase lut
26:24	R/W	2	reg_raw_phslut_14: raw phase lut
22:20	R/W	3	reg_raw_phslut_13: raw phase lut
18:16	R/W	2	reg_raw_phslut_12: raw phase lut
14:12	R/W	1	reg_raw_phslut_11: raw phase lut
10:8	R/W	0	reg_raw_phslut_10: raw phase lut
6:4	R/W	1	reg_raw_phslut_9: raw phase lut
2:0	R/W	0	reg_raw_phslut_8: raw phase lut

**Table 7-472 ISP\_ENC\_LOSS\_GLOBAL\_PHASE\_LUT\_1 0x15**

Bit(s)	R/W	Default	Description
30:28	R/W	3	reg_raw_phslut_7: raw phase lut
26:24	R/W	2	reg_raw_phslut_6: raw phase lut
22:20	R/W	3	reg_raw_phslut_5: raw phase lut
18:16	R/W	2	reg_raw_phslut_4: raw phase lut
14:12	R/W	1	reg_raw_phslut_3: raw phase lut
10:8	R/W	0	reg_raw_phslut_2: raw phase lut

Bit(s)	R/W	Default	Description
6:4	R/W	1	reg_raw_phslut_1: raw phase lut
2:0	R/W	0	reg_raw_phslut_0: raw phase lut

Table 7-473 ISP\_ENC\_LOSS\_PHASE\_LUT 0x16

Bit(s)	R/W	Default	Description
18:16	R/W	0	reg_comp_chn_lut_4: compression phase lut
14:12	R/W	0	reg_comp_chn_lut_3: compression phase lut
10: 8	R/W	1	reg_comp_chn_lut_2: compression phase lut
6: 4	R/W	1	reg_comp_chn_lut_1: compression phase lut
2: 0	R/W	0	reg_comp_chn_lut_0: compression phase lut

Table 7-474 ISP\_ENC\_LOSS\_FLATNESS\_0 0x17

Bit(s)	R/W	Default	Description
31	R/W	0	reg_flat_en: enable signal for the flatness mode.
30	R/W	0	reg_flatness_pixel_flag: enable signal for the flatness
29	R/W	0	reg_flatness_qerr_flag:
24:20	R/W	4	reg_flatness_qp_thresh: thresh qp for flatness to be used
16:12	R/W	2	reg_flatness_qp_reduce: if the cell is flatness qllevel = qllevel- flatness_qp_reduce
11: 0	R/W	1024	reg_flatness_accum_thresh: thresh ibits_accum for flatness to be usedx16

Table 7-475 ISP\_ENC\_LOSS\_FLATNESS\_1 0x18

Bit(s)	R/W	Default	Description
11: 0	R/W	4	reg_flatness_q_err_thresh: MAX_Q_ERR-MINQ_ERR for all components is required to be less than this value , flatness to be used

Table 7-476 ISP\_ENC\_LOSS\_FLATNESS\_TH 0x19

Bit(s)	R/W	Default	Description
31:16	R/W	25	reg_flatness_det_thresh: if MAX-MIN for all components is required to be less than this value , flatness to be used
15: 0	R/W	32	reg_flatness_det_thresh_max: if MAX-MIN for all components is required to be less than this value , flatness to be used

**Table 7-477 ISP\_ENC\_LOSS\_PRED 0x1a**

Bit(s)	R/W	Default	Description
26:24	R/W	3	reg_pred_filter_model: mode to do filter to get side value for the pred_value, sid= 0: ref; 1: side0, 2:(sid1+sid2)/2; 3: (2sid0+sid1 +sid2)
20	R/W	0	reg_pred_vertical_model: bayer RB vertical prediction mode ,0: real n , 1: (nw + ne )/2
16	R/W	0	reg_pred_dir5flat_en: valid direction,the prediction is same as flatness
12	R/W	1	reg_pred_errlp: enable of lpf on error, 0: no lpf; 1:lpf on.
11: 8	R/W	4	reg_pred_drt5_rate: dynamic ratio to max_err of coring for no valid edge det base on min_err, norm to 32 as 1x
6: 4	R/W	4	reg_pred_dirconf_thrd: threshold to max_err for adaptive dir_conf, thrd= x*thred
2	R/W	0	reg_pred_dirconf_valid_2: for adaptive dir_conf
1	R/W	0	reg_pred_dirconf_valid_1: for adaptive dir_conf
0	R/W	0	reg_pred_dirconf_valid_0: for adaptive dir_confx

**Table 7-478 ISP\_ENC\_LOSS\_PRED\_TH 0x1b**

Bit(s)	R/W	Default	Description
31:16	R/W	2048	reg_pred_err0: y_g = 0 horizontal default err
15: 0	R/W	0	reg_pred_flat_thd: threshold of flat in pred

**Table 7-479 ISP\_ENC\_LOSS\_PRED\_TH\_1 0x1c**

Bit(s)	R/W	Default	Description
31:16	R/W	2048	reg_pred_drt5flat_thd: dynamic ratio to min_err for no valid edge det base on err_flat
15: 0	R/W	20	reg_pred_drt5_thrd: static threshold to detect no valid edge base on min_err

**Table 7-480 ISP\_ENC\_LOSS\_QP\_MAP\_CHN0 0x1d**

Bit(s)	R/W	Default	Description
28:24	R/W	11	reg_lut_budget2qp_0_3: from budget2x to master_qp mapping
20:16	R/W	12	reg_lut_budget2qp_0_2
12: 8	R/W	12	reg_lut_budget2qp_0_1
4: 0	R/W	12	reg_lut_budget2qp_0_0

**Table 7-481 ISP\_ENC\_LOSS\_QP\_MAP\_CHN0\_1 0x1e**

Bit(s)	R/W	Default	Description
28:24	R/W	9	reg_lut_budget2qp_0_7
20:16	R/W	10	reg_lut_budget2qp_0_6

Bit(s)	R/W	Default	Description
12: 8	R/W	10	reg_lut_budget2qp_0_5
4: 0	R/W	11	reg_lut_budget2qp_0_4

**Table 7-482 ISP\_ENC\_LOSS\_QP\_MAP\_CHN0\_2 0x1f**

Bit(s)	R/W	Default	Description
28:24	R/W	8	reg_lut_budget2qp_0_11
20:16	R/W	8	reg_lut_budget2qp_0_10
12: 8	R/W	8	reg_lut_budget2qp_0_9
4: 0	R/W	9	reg_lut_budget2qp_0_8

**Table 7-483 ISP\_ENC\_LOSS\_QP\_MAP\_CHN0\_3 0x20**

Bit(s)	R/W	Default	Description
28:24	R/W	6	reg_lut_budget2qp_0_15
20:16	R/W	6	reg_lut_budget2qp_0_14
12: 8	R/W	7	reg_lut_budget2qp_0_13
4: 0	R/W	7	reg_lut_budget2qp_0_12

**Table 7-484 ISP\_ENC\_LOSS\_QP\_MAP\_CHN0\_4 0x21**

Bit(s)	R/W	Default	Description
28:24	R/W	4	reg_lut_budget2qp_0_19
20:16	R/W	4	reg_lut_budget2qp_0_18
12: 8	R/W	5	reg_lut_budget2qp_0_17
4: 0	R/W	5	reg_lut_budget2qp_0_16

**Table 7-485 ISP\_ENC\_LOSS\_QP\_MAP\_CHN0\_5 0x22**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_0_23
20:16	R/W	1	reg_lut_budget2qp_0_22
12: 8	R/W	2	reg_lut_budget2qp_0_21
4: 0	R/W	3	reg_lut_budget2qp_0_20

**Table 7-486 ISP\_ENC\_LOSS\_QP\_MAP\_CHN0\_6 0x23**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_0_27
20:16	R/W	0	reg_lut_budget2qp_0_26
12: 8	R/W	0	reg_lut_budget2qp_0_25
4: 0	R/W	0	reg_lut_budget2qp_0_24

**Table 7-487 ISP\_ENC\_LOSS\_QP\_MAP\_CHN0\_7 0x24**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_0_31
20:16	R/W	0	reg_lut_budget2qp_0_30
12: 8	R/W	0	reg_lut_budget2qp_0_29
4: 0	R/W	0	reg_lut_budget2qp_0_28

**Table 7-488 ISP\_ENC\_LOSS\_QP\_MAP\_CHN0\_8 0x25**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_0_35
20:16	R/W	0	reg_lut_budget2qp_0_34
12: 8	R/W	0	reg_lut_budget2qp_0_33
4: 0	R/W	0	reg_lut_budget2qp_0_32

**Table 7-489 ISP\_ENC\_LOSS\_QP\_MAP\_CHN0\_9 0x26**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_0_39
20:16	R/W	0	reg_lut_budget2qp_0_38
12: 8	R/W	0	reg_lut_budget2qp_0_37
4: 0	R/W	0	reg_lut_budget2qp_0_36

**Table 7-490 ISP\_ENC\_LOSS\_QP\_MAP\_CHN1 0x27**

Bit(s)	R/W	Default	Description
28:24	R/W	11	reg_lut_budget2qp_1_3
20:16	R/W	12	reg_lut_budget2qp_1_2
12: 8	R/W	12	reg_lut_budget2qp_1_1
4: 0	R/W	12	reg_lut_budget2qp_1_0

**Table 7-491 ISP\_ENC\_LOSS\_QP\_MAP\_CHN1\_1 0x28**

Bit(s)	R/W	Default	Description
28:24	R/W	9	reg_lut_budget2qp_1_7
20:16	R/W	10	reg_lut_budget2qp_1_6
12: 8	R/W	10	reg_lut_budget2qp_1_5
4: 0	R/W	11	reg_lut_budget2qp_1_4

**Table 7-492 ISP\_ENC\_LOSS\_QP\_MAP\_CHN1\_2 0x29**

Bit(s)	R/W	Default	Description
28:24	R/W	8	reg_lut_budget2qp_1_11
20:16	R/W	8	reg_lut_budget2qp_1_10
12: 8	R/W	8	reg_lut_budget2qp_1_9
4: 0	R/W	9	reg_lut_budget2qp_1_8

**Table 7-493 ISP\_ENC\_LOSS\_QP\_MAP\_CHN1\_3 0x2a**

Bit(s)	R/W	Default	Description
28:24	R/W	6	reg_lut_budget2qp_1_15
20:16	R/W	6	reg_lut_budget2qp_1_14
12: 8	R/W	7	reg_lut_budget2qp_1_13
4: 0	R/W	7	reg_lut_budget2qp_1_12

**Table 7-494 ISP\_ENC\_LOSS\_QP\_MAP\_CHN1\_4 0x2b**

Bit(s)	R/W	Default	Description
28:24	R/W	4	reg_lut_budget2qp_1_19
20:16	R/W	4	reg_lut_budget2qp_1_18
12: 8	R/W	5	reg_lut_budget2qp_1_17
4: 0	R/W	5	reg_lut_budget2qp_1_16

**Table 7-495 ISP\_ENC\_LOSS\_QP\_MAP\_CHN1\_5 0x2c**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_1_23
20:16	R/W	1	reg_lut_budget2qp_1_22
12: 8	R/W	2	reg_lut_budget2qp_1_21
4: 0	R/W	3	reg_lut_budget2qp_1_20

**Table 7-496 ISP\_ENC\_LOSS\_QP\_MAP\_CHN1\_6 0x2d**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_1_27
20:16	R/W	0	reg_lut_budget2qp_1_26
12: 8	R/W	0	reg_lut_budget2qp_1_25
4: 0	R/W	0	reg_lut_budget2qp_1_24

**Table 7-497 ISP\_ENC\_LOSS\_QP\_MAP\_CHN1\_7 0x2e**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_1_31
20:16	R/W	0	reg_lut_budget2qp_1_30
12: 8	R/W	0	reg_lut_budget2qp_1_29
4: 0	R/W	0	reg_lut_budget2qp_1_28

**Table 7-498 ISP\_ENC\_LOSS\_QP\_MAP\_CHN1\_8 0x2f**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_1_35
20:16	R/W	0	reg_lut_budget2qp_1_34
12: 8	R/W	0	reg_lut_budget2qp_1_33
4: 0	R/W	0	reg_lut_budget2qp_1_32

**Table 7-499 ISP\_ENC\_LOSS\_QP\_MAP\_CHN1\_9 0x30**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_1_39
20:16	R/W	0	reg_lut_budget2qp_1_38
12: 8	R/W	0	reg_lut_budget2qp_1_37
4: 0	R/W	0	reg_lut_budget2qp_1_36

**Table 7-500 ISP\_ENC\_LOSS\_QP\_MAP\_CHN2 0x31**

Bit(s)	R/W	Default	Description
28:24	R/W	11	reg_lut_budget2qp_2_3
20:16	R/W	12	reg_lut_budget2qp_2_2
12: 8	R/W	12	reg_lut_budget2qp_2_1
4: 0	R/W	12	reg_lut_budget2qp_2_0

**Table 7-501 ISP\_ENC\_LOSS\_QP\_MAP\_CHN2\_1 0x32**

Bit(s)	R/W	Default	Description
28:24	R/W	9	reg_lut_budget2qp_2_7
20:16	R/W	10	reg_lut_budget2qp_2_6
12: 8	R/W	10	reg_lut_budget2qp_2_5
4: 0	R/W	11	reg_lut_budget2qp_2_4

**Table 7-502 ISP\_ENC\_LOSS\_QP\_MAP\_CHN2\_2 0x33**

Bit(s)	R/W	Default	Description
28:24	R/W	8	reg_lut_budget2qp_2_11
20:16	R/W	8	reg_lut_budget2qp_2_10
12: 8	R/W	8	reg_lut_budget2qp_2_9
4: 0	R/W	9	reg_lut_budget2qp_2_8

**Table 7-503 ISP\_ENC\_LOSS\_QP\_MAP\_CHN2\_3 0x34**

Bit(s)	R/W	Default	Description
28:24	R/W	6	reg_lut_budget2qp_2_15
20:16	R/W	6	reg_lut_budget2qp_2_14
12: 8	R/W	7	reg_lut_budget2qp_2_13
4: 0	R/W	7	reg_lut_budget2qp_2_12

**Table 7-504 ISP\_ENC\_LOSS\_QP\_MAP\_CHN2\_4 0x35**

Bit(s)	R/W	Default	Description
28:24	R/W	4	reg_lut_budget2qp_2_19
20:16	R/W	4	reg_lut_budget2qp_2_18
12: 8	R/W	5	reg_lut_budget2qp_2_17
4: 0	R/W	5	reg_lut_budget2qp_2_16

**Table 7-505 ISP\_ENC\_LOSS\_QP\_MAP\_CHN2\_5 0x36**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_2_23
20:16	R/W	1	reg_lut_budget2qp_2_22
12: 8	R/W	2	reg_lut_budget2qp_2_21
4: 0	R/W	3	reg_lut_budget2qp_2_20



**Table 7-506 ISP\_ENC\_LOSS\_QP\_MAP\_CHN2\_6 0x37**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_2_27
20:16	R/W	0	reg_lut_budget2qp_2_26
12: 8	R/W	0	reg_lut_budget2qp_2_25
4: 0	R/W	0	reg_lut_budget2qp_2_24

**Table 7-507 ISP\_ENC\_LOSS\_QP\_MAP\_CHN2\_7 0x38**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_2_31
20:16	R/W	0	reg_lut_budget2qp_2_30
12: 8	R/W	0	reg_lut_budget2qp_2_29
4: 0	R/W	0	reg_lut_budget2qp_2_28

**Table 7-508 ISP\_ENC\_LOSS\_QP\_MAP\_CHN2\_8 0x39**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_2_35
20:16	R/W	0	reg_lut_budget2qp_2_34
12: 8	R/W	0	reg_lut_budget2qp_2_33
4: 0	R/W	0	reg_lut_budget2qp_2_32

**Table 7-509 ISP\_ENC\_LOSS\_QP\_MAP\_CHN2\_9 0x3a**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_2_39
20:16	R/W	0	reg_lut_budget2qp_2_38
12: 8	R/W	0	reg_lut_budget2qp_2_37
4: 0	R/W	0	reg_lut_budget2qp_2_36

**Table 7-510 ISP\_ENC\_LOSS\_RC\_GROUP\_2 0x3b**

Bit(s)	R/W	Default	Description
31:16	R/W	1080	reg_rc_group_y: vertical size of group (super block) within picture for rate control algorithm
13: 0	R/W	24	reg_slcn_ratio: dynamic ratio to idx of extra bit budget for the cells within the line

**Table 7-511 ISP\_ENC\_LOSS\_RC\_BUDGET\_0 0x3c**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_rc_perln_16pec_xbdgt_3: extra bit budget (in pct= x/128) for the cells within the line, defined by 17 nodes to split the line into 16 pieces.
23:16	R/W	0	reg_rc_perln_16pec_xbdgt_2
15: 8	R/W	0	reg_rc_perln_16pec_xbdgt_1
7: 0	R/W	0	reg_rc_perln_16pec_xbdgt_0

**Table 7-512 ISP\_ENC\_LOSS\_RC\_BUDGET\_1 0x3d**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_rc_perln_16pec_xbdgt_7
23:16	R/W	0	reg_rc_perln_16pec_xbdgt_6
15: 8	R/W	0	reg_rc_perln_16pec_xbdgt_5
7: 0	R/W	0	reg_rc_perln_16pec_xbdgt_4

**Table 7-513 ISP\_ENC\_LOSS\_RC\_BUDGET\_2 0x3e**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_rc_perln_16pec_xbdgt_11
23:16	R/W	0	reg_rc_perln_16pec_xbdgt_10
15: 8	R/W	0	reg_rc_perln_16pec_xbdgt_9.
7: 0	R/W	0	reg_rc_perln_16pec_xbdgt_8

**Table 7-514 ISP\_ENC\_LOSS\_RC\_BUDGET\_3 0x3f**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_rc_perln_16pec_xbdgt_15
23:16	R/W	0	reg_rc_perln_16pec_xbdgt_14
15: 8	R/W	0	reg_rc_perln_16pec_xbdgt_13
7: 0	R/W	0	reg_rc_perln_16pec_xbdgt_12

**Table 7-515 ISP\_ENC\_LOSS\_RC\_BUDGET\_4 0x40**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_rc_perln_16pec_xbdgt_16
23:16	R/W	0	reg_rc_1stln_slice_xbdgt: extra bit budget (in pct= x/128) for the first line of the slice, no prediction from pre-line;
15: 8	R/W	0	reg_rc_2ndln_slice_xbdgt: extra bit budget (in pct= x/128) for the 2nd line of the slice, with limited prediction from pre-line;
7: 0	R/W	0	reg_rc_1stln_group_xbdgt: extra bit budget (in pct= x/128) for the first line of the group, still with prediction from pre-line;

**Table 7-516 ISP\_ENC\_LOSS\_RC\_BUDGET\_5 0x41**

Bit(s)	R/W	Default	Description
29:24	R/W	23	reg_rc_master_qpx2_max: maximum qpx2 during the rc_loop
21:16	R/W	14	reg_rc_master_qpx2_min: minimum qpx2 during the rc_loop
15: 8	R/W	16	reg_rc_dynamic_speed_long: dynamic speed for using up the accrued bits, 2nd order gain of loop filter. normalized to 16 as 1;
7: 0	R/W	16	reg_rc_dynamic_speed_short: dynamic speed for using up the accrued bits, 1st order gain of loop filter. normalized to 64 as 1;

**Table 7-517 ISP\_ENC\_LOSS\_RC\_BUDGET\_6 0x42**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_rc_precedell_bits_reset: enable for bits_cell_pre reset of each group 1, bits_cell_pre =0 0:bits_cell_pre value unchanged
28:16	R/W	240	reg_rc_dynamic_mxblk_long: maximum number of blocks in group for long term dynamic factor, if set to 0, then no limit; otherwise set limit
12: 0	R/W	240	reg_rc_qpx2_margin3_blkth:final guard margin threshold to blocks_left_in_group

**Table 7-518 ISP\_ENC\_LOSS\_RC\_QP\_MARGIN 0x43**

Bit(s)	R/W	Default	Description
29:16	R/W	-12	reg_rc_qpx2_margin_thd_1: threshold x16 to ibits_accum to assign extra pq margin to avoid buffer down flow
13: 0	R/W	-4	reg_rc_qpx2_margin_thd_0

**Table 7-519 ISP\_ENC\_LOSS\_RC\_QP\_MARGIN\_1 0x44**

Bit(s)	R/W	Default	Description
29:16	R/W	8	reg_rc_qpx2_margin_thd_3
13: 0	R/W	-16	reg_rc_qpx2_margin_thd_2

**Table 7-520 ISP\_ENC\_LOSS\_RC\_QP\_MARGIN\_2 0x45**

Bit(s)	R/W	Default	Description
29:24	R/W	6	reg_rc_qpx2_margin_dlt_3: delta qp margin to qp_max when ibits_accum lower than rc_qp_margin_thd[2]
21:16	R/W	5	reg_rc_qpx2_margin_dlt_2
13: 8	R/W	3	reg_rc_qpx2_margin_dlt_1
5: 0	R/W	2	reg_rc_qpx2_margin_dlt_0

**Table 7-521 ISP\_ENC\_LOSS\_RC\_QP\_MARGIN\_3 0x46**

Bit(s)	R/W	Default	Description
31:30	R/W	1	reg_rc_qp_x2_margin_dlt_mode
27:16	R/W	250	reg_rc_fifo_qp_x2_margin_thd_1: threshold to fifo to assign extra pq margin to avoid buffer down flow,
11: 0	R/W	150	reg_rc_fifo_qp_x2_margin_thd_0

**Table 7-522 ISP\_ENC\_LOSS\_RC\_QP\_MARGIN\_4 0x47**

Bit(s)	R/W	Default	Description
27:16	R/W	500	reg_rc_fifo_qp_x2_margin_thd_3
11: 0	R/W	350	reg_rc_fifo_qp_x2_margin_thd_2

**Table 7-523 ISP\_ENC\_LOSS\_RC\_QP\_MARGIN\_5 0x48**

Bit(s)	R/W	Default	Description
29:24	R/W	16	reg_rc_fifo_qp_x2_margin_dlt_3: delta qp margin to qp_max when fifo lower than rc_qp_margin_thd[2]
21:16	R/W	7	reg_rc_fifo_qp_x2_margin_dlt_2
13: 8	R/W	5	reg_rc_fifo_qp_x2_margin_dlt_1
5: 0	R/W	3	reg_rc_fifo_qp_x2_margin_dlt_0

**Table 7-524 ISP\_ENC\_LOSS\_FLATNESS\_ADJ0 0x49**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_adjsize_flatness_flag: enable signal for the flatness mode,adj_predicted_size should be small in flatness
30	R/W	0	reg_adjsize_complex_flag: enable signal for the complex mode,adj_predicted_size should be large in complex
29	R/W	0	reg_rc_fifo_avgspeed_use_sbudget: enable to use s_budget_block as budget_block: 0 use bits_perblk programable register, 1: use rc calculated s_budget_blk
28	R/W	0	reg_rc_bits_gap_dlt_down_en: to dlt_buget2x reduced based on bits_gap value
27:16	R/W	50	reg_adjsize_flatness_pixthd: area if(MAX-MIN) bigger than pixthd
12: 8	R/W	7	reg_adjsize_flatness_sizedlt: delta size margin to adj_predicted_size when adj_predicted_size lower than reg_adjsize_flatness_sizedlt
4: 0	R/W	2	reg_adjsize_flatness_reduce: the cell is flatness and adj_predicted_size > thd, adj_predicted_size = adj_predicted_size- reg_adjsize_flatness_reduce

**Table 7-525 ISP\_ENC\_LOSS\_FLATNESS\_ADJ1 0x4a**

Bit(s)	R/W	Default	Description
27:16	R/W	512	reg_adjsize_complex_pixthd : area if(MAX-MIN) bigger than pixthd
12: 8	R/W	7	reg_adjsize_complex_sizedlt : delta size margin to adj_predicted_size when adj_predicted_size bigger than reg_adjsize_complex_sizedlt
4: 0	R/W	2	reg_adjsize_complex_increase: if the cell is complex and adj_predicted_size < thd,adj_predicted_size = adj_predicted_size + reg_adjsize_complex_increase

**Table 7-526 ISP\_ENC\_LOSS\_FIFO\_THD\_0 0x4b**

Bit(s)	R/W	Default	Description
23:12	R/W	280	reg_rc_fifo_margin_thd_5:threshold of fifo level(in words=16bits) to guard the rc loop by adding delta to p_budget2x
11: 0	R/W	250	reg_rc_fifo_margin_thd_4

**Table 7-527 ISP\_ENC\_LOSS\_FIFO\_THD\_1 0x4c**

Bit(s)	R/W	Default	Description
23:12	R/W	220	reg_rc_fifo_margin_thd_3
11: 0	R/W	165	reg_rc_fifo_margin_thd_2

**Table 7-528 ISP\_ENC\_LOSS\_FIFO\_THD\_2 0x4d**

Bit(s)	R/W	Default	Description
23:12	R/W	145	reg_rc_fifo_margin_thd_1
11: 0	R/W	130	reg_rc_fifo_margin_thd_0

**Table 7-529 ISP\_ENC\_LOSS\_FIFO\_AVG 0x4e**

Bit(s)	R/W	Default	Description
25:20	R/W	12	reg_rc_fifo_margin_dlt_5: delta of fifo level to guard the rc loop by adding delta to p_budget2x
17:12	R/W	14	reg_rc_fifo_margin_dlt_4:
11: 0	R/W	80	reg_rc_fifo_avgspeed_bits_perblk: bits perblock fifo read/write speed, set to budget_block = (SIZE_BLK*(ratio_bppx16) )/16 as default

**Table 7-530 ISP\_ENC\_LOSS\_FIFO\_DLT 0x4f**

Bit(s)	R/W	Default	Description
29:24	R/W	11	reg_rc_fifo_margin_dlt_3: delta of fifo level to guard the rc loop by adding delta to p_budget2x
21:16	R/W	8	reg_rc_fifo_margin_dlt_2

Bit(s)	R/W	Default	Description
13: 8	R/W	5	reg_rc_fifo_margin_dlt_1
5: 0	R/W	2	reg_rc_fifo_margin_dlt_0

Table 7-531 ISP\_ENC\_LOSS\_BITSGAP\_THD\_0 0x50

Bit(s)	R/W	Default	Description
27:16	R/W	50	reg_rc_bits_gap_margin_thd_1: threshold of bits gap to reduced dlt_buget2x
11: 0	R/W	30	reg_rc_bits_gap_margin_thd_0

Table 7-532 ISP\_ENC\_LOSS\_BITSGAP\_THD\_1 0x51

Bit(s)	R/W	Default	Description
11: 0	R/W	82	reg_rc_bits_gap_margin_thd_2:threshold of bits gap to reduced dlt_buget2x

Table 7-533 ISP\_ENC\_LOSS\_ICH 0x52

Bit(s)	R/W	Default	Description
25	R/W	1	reg_ich_enable: enable signal for the ich mode.
24	R/W	1	reg_ich_error_compare_pred_en: ICH maxerror compare prediction maxerror en
20:16	R/W	5	reg_ich_max_error_sel: ICH error th sel, th = 1<<sel [0,DW-1)
12: 8	R/W	8	reg_ich_bits_weight: cost bits weight(0-16)
4: 0	R/W	8	reg_pre_bits_weight: cost bits weight (0-16)

Table 7-534 ISP\_ENC\_LOSS\_ICH\_UPTH 0x53

Bit(s)	R/W	Default	Description
7: 0	R/W	4	reg_ich_update_th: thresh for ICH history update

Table 7-535 ISP\_ENC\_LOSS\_LUMA 0x54

Bit(s)	R/W	Default	Description
16	R/W	0	reg_luma_adjust_en: enable of luma(G) adjust
12: 8	R/W	1	reg_pixel_luma_adj_dlt_1: of qelvel according to luma
4: 0	R/W	2	reg_pixel_luma_adj_dlt_0:

Table 7-536 ISP\_ENC\_LOSS\_LUMA\_TH 0x55

Bit(s)	R/W	Default	Description
31:16	R/W	128	reg_pixel_luma_adj_th_1: luma level
15: 0	R/W	256	reg_pixel_luma_adj_th_0:

**Table 7-537 ISP\_ENC\_LOSS\_ACCUM\_OFSET\_0 0x56**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_accum_add_ofset_2
23:16	R/W	5	reg_accum_add_ofset_1
15: 8	R/W	2	reg_accum_add_ofset_0
3: 1	R/W	2	reg_accum_ofset_shift
0	R/W	0	reg_accum_ofset_en

**Table 7-538 ISP\_ENC\_LOSS\_ACCUM\_OFSET\_1 0x57**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_accum_add_ofset_6
23:16	R/W	5	reg_accum_add_ofset_5
15: 8	R/W	2	reg_accum_add_ofset_4
7: 0	R/W	0	reg_accum_add_ofset_3

**Table 7-539 ISP\_ENC\_LOSS\_ACCUM\_OFSET\_2 0x58**

Bit(s)	R/W	Default	Description
15: 8	R/W	255	reg_accum_add_ofset_8
7: 0	R/W	0	reg_accum_add_ofset_7

**Table 7-540 ISP\_ENC\_LOSS\_ACCUM\_OFSET\_3 0x59**

Bit(s)	R/W	Default	Description
11: 0	R/W	0	reg_normalize_idx_ratio

**Table 7-541 ISP\_ENC\_LOSS\_DERIVA\_ADJ 0x5a**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_derivative_lut_0_9: lut of derivative adjusted
26:24	R/W	0	reg_derivative_lut_0_8
22:20	R/W	0	reg_derivative_lut_0_7
18:16	R/W	0	reg_derivative_lut_0_6
14:12	R/W	0	reg_derivative_lut_0_5
10: 8	R/W	0	reg_derivative_lut_0_4
6: 4	R/W	0	reg_derivative_lut_1_9
0	R/W	0	reg_derivative_en: 1 to enable derivative adjusted

**Table 7-542 ISP\_ENC\_LOSS\_DERIVA\_ADJ\_1 0x5b**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_derivative_lut_1_8
26:24	R/W	0	reg_derivative_lut_1_7
22:20	R/W	0	reg_derivative_lut_1_6
18:16	R/W	0	reg_derivative_lut_1_5
14:12	R/W	0	reg_derivative_lut_2_9
10: 8	R/W	0	reg_derivative_lut_2_8
6: 4	R/W	0	reg_derivative_lut_2_7
2: 0	R/W	0	reg_derivative_lut_2_6

**Table 7-543 ISP\_ENC\_LOSS\_DERIVA\_ADJ\_2 0x5c**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_derivative_lut_9_5
26:24	R/W	0	reg_derivative_lut_9_4
22:20	R/W	0	reg_derivative_lut_9_3
18:16	R/W	0	reg_derivative_lut_9_2
14:12	R/W	0	reg_derivative_lut_9_1
10: 8	R/W	0	reg_derivative_lut_9_0
6: 4	R/W	0	reg_derivative_lut_8_4
2: 0	R/W	0	reg_derivative_lut_8_3

**Table 7-544 ISP\_ENC\_LOSS\_DERIVA\_ADJ\_3 0x5d**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_derivative_lut_8_2
26:24	R/W	0	reg_derivative_lut_8_1
22:20	R/W	0	reg_derivative_lut_8_0
18:16	R/W	0	reg_derivative_lut_7_4
14:12	R/W	0	reg_derivative_lut_7_3
10: 8	R/W	0	reg_derivative_lut_7_2
6: 4	R/W	0	reg_derivative_lut_7_1
2: 0	R/W	0	reg_derivative_lut_7_0



**Table 7-545 ISP\_ENC\_LOSS\_WDR\_LINE\_DELAY 0x5e**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_wdr_mode_en: mode enable,(in this mode have line delay between long and short frame)
15: 0	R/W	4	reg_wdr_delay_line: delay between long and short frame in wdr mode

**Table 7-546 ISP\_ENC\_LOSS\_STATS\_RAM\_MODE 0x5f**

Bit(s)	R/W	Default	Description
1	R/W	0	reg_stats_slice_rmode: 0 to read slice according to reg_slice_num_mode
0	R/W	0	reg_stats_ram_rden: 1 to read stats ram enable

**Table 7-547 ISP\_ENC\_LOSS\_STATS\_RAM\_ADDR 0x60**

Bit(s)	R/W	Default	Description
8: 0	R/W	0	reg_stats_ram_addr: statistics info ram status address

**Table 7-548 ISP\_ENC\_LOSS\_RO\_STATS\_RAM\_DATA 0x61**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_stats_ram_data: statistics info from ram data

**Table 7-549 ISP\_ENC\_LOSS\_MIF\_CTRL 0x70**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_sync_sel: 0 to sync with frm rst [24]: for enable [25]: for reg_canvas_id [26]: for dol mode
23:16	R/W	0	reg_canvas_id: axi canvas id num
14:12	R/W	1	reg_cmd_intr_len: interrupt send cmd when how many series axi cmd, 0=12 1=16 2=24 3=32 4=40 5=48 6=56 7=64
9: 8	R/W	1	reg_cmd_req_size: how many room fifo have, then axi send series req, 0=16 1=32 2=48 3=64
6: 4	R/W	3	reg_burst_len: burst type: 0->1; 1->2; 2->4; 3->8; 4->16, others reserved
2	R/W	0	reg_check_resp_id: 1 to check RESP ID
1	R/W	0	reg_dol_mode: 1: DOL mode
0	R/W	1	reg_mif_enable: 1 to enable mif

**Table 7-550 ISP\_ENC\_LOSS\_MIF\_QOS 0x71**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_qos_ctrl: Qos control [18]: reg_qos_auto_en [17]: reg_qos_sup [16]: reg_qos_ini [15:8]: reg_qos_up_th [7:0]: reg_qos_dn_th

**Table 7-551 ISP\_ENC\_LOSS\_MIF\_URGENT 0x72**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_urgent_ctrl: urgent control [18]: reg_urgent_auto_en [17]: reg_urgent_sup [16]: reg_urgent_ini [15:8]: reg_urgent_up_th [7:0]: reg_urgent_dn_th

**Table 7-552 ISP\_ENC\_LOSS\_MIF\_MISC 0x73**

Bit(s)	R/W	Default	Description
3	R/W	1	reg_mif_busy_noresp_check: for mif write, dont take "respond busy" as mif unit busy.
2	R/W	0	reg_mif_wait_input4busy: use in encoder, mif is busy till input data comming. only for mif wr
1	R/W	0	reg_mif_dbl_baddr_init: write 1 to use reg_mif_baddr as base-addr for next frame,it would be clear automatically.
0	R/W	0	reg_mif_dbl_baddr_en:

**Table 7-553 ISP\_ENC\_LOSS\_MIF\_BADDR 0x74**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_mif_baddr: mif base address

**Table 7-554 ISP\_ENC\_LOSS\_MIF\_BADDR1 0x75**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_mif_baddr1: mif base address

**Table 7-555 ISP\_ENC\_LOSS\_MIF\_FIFO\_CTRL 0x77**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_mif_gclk_ctrl: gate clock control
7:0	R/W	128	reg_mif_fifo_size: data FIFO max size, default = 128,

**Table 7-556 ISP\_ENC\_LOSS\_MIF\_FRM\_HOLD 0x78**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_mif_frm_holds: clock cycle holder from frm_rst to wait register ready, default = 16 Note: set bit[31] to 1 indicates the compress would start to work till RDMA transfer finished or write [31:0] to 0xffffffff

**Table 7-557 ISP\_ENC\_LOSS\_MIF\_RO\_STATS 0x79**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_mif_status: mif status

**Table 7-558 ISP\_DEC\_LOSS\_CTRL 0x80**

Bit(s)	R/W	Default	Description
31:24	R/W	6	reg_hblank1: should >=6, blank cycle between two line
23:16	R/W	6	reg_hblank0: should >=6
15:8	R/W	0	reg_sync_ctrl: shaddow control, 0 to shaddow the register
7:4	R/W	0	reg_use_inter_fmt: 0 to use external format setting: for T7, it need set to 4'b1111
3	R/W	0	reg_status_rdbk_mode: 0 to shaddow the ro status register
2:1	R/W	3	reg_stats_en:[0] to enable max_err/err_acc stats; [1] to enable error_count2 stats
0	R/W	1	reg_enable: 1 to enable

**Table 7-559 ISP\_DEC\_LOSS\_FRAME\_HOLD 0x81**

Bit(s)	R/W	Default	Description
31:0	R/W	1	reg_frame_hold_nums: hold clock number from frame_rst to wait register ready Note: set bit[31] to 1 indicates the compress would start till RDMA transfer finished or write [31:0] to 0xffffffff

**Table 7-560 ISP\_DEC\_LOSS\_GCLK\_CTRL 0x82**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_gclk_ctrl: gated-clock control

**Table 7-561 ISP\_DEC\_LOSS\_RO\_CODEC\_STATUS 0x83**

Bit(s)	R/W	Default	Description
31:0	R/W	0	ro_codec_status: codec status, write 1 to clear

**Table 7-562 ISP\_DEC\_LOSS\_LOSS\_MISC 0x84**

Bit(s)	R/W	Default	Description
31:0	R/W	0	[5] MISC control register : reg_not_wait_input4busy : reg_busy_status_use4abort

**Table 7-563 ISP\_DEC\_LOSS\_BASIS 0x88**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_lossless_en: 1 to use lossless mode
28:24	R/W	12	reg_src_bit_depth: source data bit depth from sensor
23:21	R/W	1	reg_raw_mode:0: mono, 1:Bayer, 2/3: RGBIR2x2 4:RGBIR4x4
20	R/W	0	reg_mono_comp_mode:compression mode of mono data : 0 compression in Y, 1: compression in bayer
19:18	R/W	1	reg_xphase_ofst:phase offset in x dimmension
17:16	R/W	0	reg_yphase_ofst:phase offset in y dimmension
15:13	R/W		reserved
12:4	R/W	80	reg_ratio_bppx16: bits per pixel x16, fracbits_bpp = x/16
2:0	R/W	2	reg_comp_chn_size: color space mode for snr1 input

**Table 7-564 ISP\_DEC\_LOSS\_PIC\_SIZE 0x89**

Bit(s)	R/W	Default	Description
31:16	R/W	1080	reg_pic_ysize:picture verital size
15:0	R/W	1920	reg_pic_xsize:picture horizontal size

**Table 7-565 ISP\_DEC\_LOSS\_SLICE\_SIZE 0x8a**

Bit(s)	R/W	Default	Description
31	R/W	1	reg_use_sw_preslc_bitaccum
29:28	R/W	0	reg_slice_num_mode: 0 slice num 1, 1:slice num 2 2:slice num 4 3:slice num 8
15:0	R/W	1080	reg_slice_ysize_0:vertical slice window size for slice0

**Table 7-566 ISP\_DEC\_LOSS\_SLICE\_SIZE\_1 0x8b**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_slice_ysize_2:vertical slice window size for slice2
15:0	R/W	0	reg_slice_ysize_1:vertical slice window size for slice1

**Table 7-567 ISP\_DEC\_LOSS\_SLICE\_SIZE\_2 0x8c**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_slice_ysize_4:vertical slice window size for slice4
15:0	R/W	0	reg_slice_ysize_3:vertical slice window size for slice3

**Table 7-568 ISP\_DEC\_LOSS\_SLICE\_SIZE\_3 0x8d**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_slice_ysize_6:vertical slice window size for slice6
15:0	R/W	0	reg_slice_ysize_5:vertical slice window size for slice5

**Table 7-569 ISP\_DEC\_LOSS\_PRESL\_LAST\_BITS 0x8e**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_pre_slc_bitaccum: pre slice last bits add to next slice

**Table 7-570 ISP\_DEC\_LOSS\_PRESL\_FIFO\_LEVEL 0x8f**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_use_sw_preslc_fifolevel
15:0	R/W	16	reg_pre_slc_fifolevel:pre slice fifolevel add to next slice

**Table 7-571 ISP\_DEC\_LOSS\_DEFAULT\_VALUE 0x90**

Bit(s)	R/W	Default	Description
31:16	R/W	2048	reg_default_value_0:default value of comp 0
15:0	R/W	2048	reg_default_value_1:default value of comp 1

**Table 7-572 ISP\_DEC\_LOSS\_DEFAULT\_VALUE\_1 0x91**

Bit(s)	R/W	Default	Description
15:0	R/W	2048	reg_default_value_2:default value of comp 2

**Table 7-573 ISP\_DEC\_LOSS\_ERROR\_COUNT\_THD 0x92**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_error_count_region_half_en:enable of static the left half of the image, 1: only static left half of the image, 0: all image
15:0	R/W	100	reg_error_count_thd:threshold to count number of error bigger thd

**Table 7-574 ISP\_DEC\_LOSS\_DEBUG 0x93**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_debug_mode: 1 to oupput qllevel for decoder only
24:16	R/W	0	reg_dbg_qllevel_2: [24] =1 to force qllevel_2 to reg_dbg_qllevel_2[4:0]
15:8	R/W	0	reg_dbg_qllevel_1: [15] =1 to force qllevel_1 to reg_dbg_qllevel_1[4:0]
7:0	R/W	0	reg_dbg_qllevel_0: [7] =1 to force qllevel_0 to reg_dbg_qllevel_0[4:0]

**Table 7-575 ISP\_DEC\_LOSS\_GLOBAL\_PHASE\_LUT 0x94**

Bit(s)	R/W	Default	Description
30:28	R/W	3	reg_raw_phslut_15: raw phase lut
26:24	R/W	2	reg_raw_phslut_14: raw phase lut
22:20	R/W	3	reg_raw_phslut_13: raw phase lut
18:16	R/W	2	reg_raw_phslut_12: raw phase lut
14:12	R/W	1	reg_raw_phslut_11: raw phase lut
10:8	R/W	0	reg_raw_phslut_10: raw phase lut
6:4	R/W	1	reg_raw_phslut_9: raw phase lut
2:0	R/W	0	reg_raw_phslut_8: raw phase lut

**Table 7-576 ISP\_DEC\_LOSS\_GLOBAL\_PHASE\_LUT\_1 0x95**

Bit(s)	R/W	Default	Description
30:28	R/W	3	reg_raw_phslut_7: raw phase lut
26:24	R/W	2	reg_raw_phslut_6: raw phase lut
22:20	R/W	3	reg_raw_phslut_5: raw phase lut
18:16	R/W	2	reg_raw_phslut_4: raw phase lut
14:12	R/W	1	reg_raw_phslut_3: raw phase lut
10:8	R/W	0	reg_raw_phslut_2: raw phase lut
6:4	R/W	1	reg_raw_phslut_1: raw phase lut
2:0	R/W	0	reg_raw_phslut_0: raw phase lut

**Table 7-577 ISP\_DEC\_LOSS\_PHASE\_LUT 0x96**

Bit(s)	R/W	Default	Description
18:16	R/W	0	reg_comp_chn_lut_4: compression phase lut
14:12	R/W	0	reg_comp_chn_lut_3: compression phase lut
10: 8	R/W	1	reg_comp_chn_lut_2: compression phase lut
6: 4	R/W	1	reg_comp_chn_lut_1: compression phase lut
2: 0	R/W	0	reg_comp_chn_lut_0: compression phase lut

Table 7-578 ISP\_DEC\_LOSS\_FLATNESS\_0 0x97

Bit(s)	R/W	Default	Description
31	R/W	0	reg_flat_en: enable signal for the flatness mode.
30	R/W	0	reg_flatness_pixel_flag: enable signal for the flatness
29	R/W	0	reg_flatness_qerr_flag:
24:20	R/W	4	reg_flatness_qp_thresh: thresh qp for flatness to be used
16:12	R/W	2	reg_flatness_qp_reduce: if the cell is flatness qlevel = qlevel- flatness_qp_reduce
11: 0	R/W	1024	reg_flatness_accum_thresh: thresh ibits_accum for flatness to be usedx16

Table 7-579 ISP\_DEC\_LOSS\_FLATNESS\_1 0x98

Bit(s)	R/W	Default	Description
11: 0	R/W	4	reg_flatness_q_err_thresh: MAX_Q_ERR-MINQ_ERR for all components is required to be less than this value , flatness to be used

Table 7-580 ISP\_DEC\_LOSS\_FLATNESS\_TH 0x99

Bit(s)	R/W	Default	Description
31:16	R/W	25	reg_flatness_det_thresh: if MAX-MIN for all components is required to be less than this value , flatness to be used
15: 0	R/W	32	reg_flatness_det_thresh_max: if MAX-MIN for all components is required to be less than this value , flatness to be used

Table 7-581 ISP\_DEC\_LOSS\_PRED 0x9a

Bit(s)	R/W	Default	Description
26:24	R/W	3	reg_pred_filter_model: mode to do filter to get side value for the pred_value, sid= 0: ref; 1: side0, 2:(sid1+sid2)/2; 3: (2sid0+sid1 +sid2)
20	R/W	0	reg_pred_vertical_model: bayer RB vertical prediction mode ,0: real n , 1: (nw + ne )/2
16	R/W	0	reg_pred_dir5flat_en: valid direction,the prediction is same as flatness
12	R/W	1	reg_pred_errlp: enable of lpf on error, 0: no lpf; 1:lpf on.
11: 8	R/W	4	reg_pred_drt5_rate: dynamic ratio to max_err of coring for no valid edge det base on min_err, norm to 32 as 1x
6: 4	R/W	4	reg_pred_dirconf_thrd: threshold to max_err for adaptive dir_conf, thrd= x*thred
2	R/W	0	reg_pred_dirconf_valid_2: for adaptive dir_conf
1	R/W	0	reg_pred_dirconf_valid_1: for adaptive dir_conf
0	R/W	0	reg_pred_dirconf_valid_0: for adaptive dir_confx

**Table 7-582 ISP\_DEC\_LOSS\_PRED\_TH 0x9b**

Bit(s)	R/W	Default	Description
31:16	R/W	2048	reg_pred_err0: y_g = 0 horizontal default err
15: 0	R/W	0	reg_pred_flat_thd: threshold of flat in pred

**Table 7-583 ISP\_DEC\_LOSS\_PRED\_TH\_1 0x9c**

Bit(s)	R/W	Default	Description
31:16	R/W	2048	reg_pred_drt5flat_thd: dynamic ratio to min_err for no valid edge det base on err_flat
15: 0	R/W	20	reg_pred_drt5_thrd: static threshold to detect no valid edge base on min_err

**Table 7-584 ISP\_DEC\_LOSS\_QP\_MAP\_CHN0 0x9d**

Bit(s)	R/W	Default	Description
28:24	R/W	11	reg_lut_budget2qp_0_3: from budget2x to master_qp mapping
20:16	R/W	12	reg_lut_budget2qp_0_2
12: 8	R/W	12	reg_lut_budget2qp_0_1
4: 0	R/W	12	reg_lut_budget2qp_0_0

**Table 7-585 ISP\_DEC\_LOSS\_QP\_MAP\_CHN0\_1 0x9e**

Bit(s)	R/W	Default	Description
28:24	R/W	9	reg_lut_budget2qp_0_7
20:16	R/W	10	reg_lut_budget2qp_0_6
12: 8	R/W	10	reg_lut_budget2qp_0_5
4: 0	R/W	11	reg_lut_budget2qp_0_4

**Table 7-586 ISP\_DEC\_LOSS\_QP\_MAP\_CHN0\_2 0x9f**

Bit(s)	R/W	Default	Description
28:24	R/W	8	reg_lut_budget2qp_0_11
20:16	R/W	8	reg_lut_budget2qp_0_10
12: 8	R/W	8	reg_lut_budget2qp_0_9
4: 0	R/W	9	reg_lut_budget2qp_0_8

**Table 7-587 ISP\_DEC\_LOSS\_QP\_MAP\_CHN0\_3 0xa0**

Bit(s)	R/W	Default	Description
28:24	R/W	6	reg_lut_budget2qp_0_15
20:16	R/W	6	reg_lut_budget2qp_0_14



Bit(s)	R/W	Default	Description
12: 8	R/W	7	reg_lut_budget2qp_0_13
4: 0	R/W	7	reg_lut_budget2qp_0_12

**Table 7-588 ISP\_DEC\_LOSS\_QP\_MAP\_CHN0\_4 0xa1**

Bit(s)	R/W	Default	Description
28:24	R/W	4	reg_lut_budget2qp_0_19
20:16	R/W	4	reg_lut_budget2qp_0_18
12: 8	R/W	5	reg_lut_budget2qp_0_17
4: 0	R/W	5	reg_lut_budget2qp_0_16

**Table 7-589 ISP\_DEC\_LOSS\_QP\_MAP\_CHN0\_5 0xa2**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_0_23
20:16	R/W	1	reg_lut_budget2qp_0_22
12: 8	R/W	2	reg_lut_budget2qp_0_21
4: 0	R/W	3	reg_lut_budget2qp_0_20

**Table 7-590 ISP\_DEC\_LOSS\_QP\_MAP\_CHN0\_6 0xa3**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_0_27
20:16	R/W	0	reg_lut_budget2qp_0_26
12: 8	R/W	0	reg_lut_budget2qp_0_25
4: 0	R/W	0	reg_lut_budget2qp_0_24

**Table 7-591 ISP\_DEC\_LOSS\_QP\_MAP\_CHN0\_7 0xa4**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_0_31
20:16	R/W	0	reg_lut_budget2qp_0_30
12: 8	R/W	0	reg_lut_budget2qp_0_29
4: 0	R/W	0	reg_lut_budget2qp_0_28

**Table 7-592 ISP\_DEC\_LOSS\_QP\_MAP\_CHN0\_8 0xa5**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_0_35
20:16	R/W	0	reg_lut_budget2qp_0_34
12: 8	R/W	0	reg_lut_budget2qp_0_33
4: 0	R/W	0	reg_lut_budget2qp_0_32

**Table 7-593 ISP\_DEC\_LOSS\_QP\_MAP\_CHN0\_9 0xa6**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_0_39
20:16	R/W	0	reg_lut_budget2qp_0_38
12: 8	R/W	0	reg_lut_budget2qp_0_37
4: 0	R/W	0	reg_lut_budget2qp_0_36

**Table 7-594 ISP\_DEC\_LOSS\_QP\_MAP\_CHN1 0xa7**

Bit(s)	R/W	Default	Description
28:24	R/W	11	reg_lut_budget2qp_1_3
20:16	R/W	12	reg_lut_budget2qp_1_2
12: 8	R/W	12	reg_lut_budget2qp_1_1
4: 0	R/W	12	reg_lut_budget2qp_1_0

**Table 7-595 ISP\_DEC\_LOSS\_QP\_MAP\_CHN1\_1 0xa8**

Bit(s)	R/W	Default	Description
28:24	R/W	9	reg_lut_budget2qp_1_7
20:16	R/W	10	reg_lut_budget2qp_1_6
12: 8	R/W	10	reg_lut_budget2qp_1_5
4: 0	R/W	11	reg_lut_budget2qp_1_4

**Table 7-596 ISP\_DEC\_LOSS\_QP\_MAP\_CHN1\_2 0xa9**

Bit(s)	R/W	Default	Description
28:24	R/W	8	reg_lut_budget2qp_1_11
20:16	R/W	8	reg_lut_budget2qp_1_10
12: 8	R/W	8	reg_lut_budget2qp_1_9
4: 0	R/W	9	reg_lut_budget2qp_1_8

**Table 7-597 ISP\_DEC\_LOSS\_QP\_MAP\_CHN1\_3 0xaa**

Bit(s)	R/W	Default	Description
28:24	R/W	6	reg_lut_budget2qp_1_15
20:16	R/W	6	reg_lut_budget2qp_1_14
12: 8	R/W	7	reg_lut_budget2qp_1_13
4: 0	R/W	7	reg_lut_budget2qp_1_12

**Table 7-598 ISP\_DEC\_LOSS\_QP\_MAP\_CHN1\_4 0xab**

Bit(s)	R/W	Default	Description
28:24	R/W	4	reg_lut_budget2qp_1_19
20:16	R/W	4	reg_lut_budget2qp_1_18
12: 8	R/W	5	reg_lut_budget2qp_1_17
4: 0	R/W	5	reg_lut_budget2qp_1_16

**Table 7-599 ISP\_DEC\_LOSS\_QP\_MAP\_CHN1\_5 0xac**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_1_23
20:16	R/W	1	reg_lut_budget2qp_1_22
12: 8	R/W	2	reg_lut_budget2qp_1_21
4: 0	R/W	3	reg_lut_budget2qp_1_20

**Table 7-600 ISP\_DEC\_LOSS\_QP\_MAP\_CHN1\_6 0xad**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_1_27
20:16	R/W	0	reg_lut_budget2qp_1_26
12: 8	R/W	0	reg_lut_budget2qp_1_25
4: 0	R/W	0	reg_lut_budget2qp_1_24

**Table 7-601 ISP\_DEC\_LOSS\_QP\_MAP\_CHN1\_7 0xae**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_1_31
20:16	R/W	0	reg_lut_budget2qp_1_30
12: 8	R/W	0	reg_lut_budget2qp_1_29
4: 0	R/W	0	reg_lut_budget2qp_1_28

**Table 7-602 ISP\_DEC\_LOSS\_QP\_MAP\_CHN1\_8 0xaf**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_1_35
20:16	R/W	0	reg_lut_budget2qp_1_34
12: 8	R/W	0	reg_lut_budget2qp_1_33
4: 0	R/W	0	reg_lut_budget2qp_1_32

**Table 7-603 ISP\_DEC\_LOSS\_QP\_MAP\_CHN1\_9 0xb0**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_1_39
20:16	R/W	0	reg_lut_budget2qp_1_38
12: 8	R/W	0	reg_lut_budget2qp_1_37
4: 0	R/W	0	reg_lut_budget2qp_1_36

**Table 7-604 ISP\_DEC\_LOSS\_QP\_MAP\_CHN2 0xb1**

Bit(s)	R/W	Default	Description
28:24	R/W	11	reg_lut_budget2qp_2_3
20:16	R/W	12	reg_lut_budget2qp_2_2
12: 8	R/W	12	reg_lut_budget2qp_2_1
4: 0	R/W	12	reg_lut_budget2qp_2_0

**Table 7-605 ISP\_DEC\_LOSS\_QP\_MAP\_CHN2\_1 0xb2**

Bit(s)	R/W	Default	Description
28:24	R/W	9	reg_lut_budget2qp_2_7
20:16	R/W	10	reg_lut_budget2qp_2_6
12: 8	R/W	10	reg_lut_budget2qp_2_5
4: 0	R/W	11	reg_lut_budget2qp_2_4

**Table 7-606 ISP\_DEC\_LOSS\_QP\_MAP\_CHN2\_2 0xb3**

Bit(s)	R/W	Default	Description
28:24	R/W	8	reg_lut_budget2qp_2_11
20:16	R/W	8	reg_lut_budget2qp_2_10
12: 8	R/W	8	reg_lut_budget2qp_2_9
4: 0	R/W	9	reg_lut_budget2qp_2_8

**Table 7-607 ISP\_DEC\_LOSS\_QP\_MAP\_CHN2\_3 0xb4**

Bit(s)	R/W	Default	Description
28:24	R/W	6	reg_lut_budget2qp_2_15
20:16	R/W	6	reg_lut_budget2qp_2_14
12: 8	R/W	7	reg_lut_budget2qp_2_13
4: 0	R/W	7	reg_lut_budget2qp_2_12

**Table 7-608 ISP\_DEC\_LOSS\_QP\_MAP\_CHN2\_4 0xb5**

Bit(s)	R/W	Default	Description
28:24	R/W	4	reg_lut_budget2qp_2_19
20:16	R/W	4	reg_lut_budget2qp_2_18
12: 8	R/W	5	reg_lut_budget2qp_2_17
4: 0	R/W	5	reg_lut_budget2qp_2_16

**Table 7-609 ISP\_DEC\_LOSS\_QP\_MAP\_CHN2\_5 0xb6**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_2_23
20:16	R/W	1	reg_lut_budget2qp_2_22
12: 8	R/W	2	reg_lut_budget2qp_2_21
4: 0	R/W	3	reg_lut_budget2qp_2_20

**Table 7-610 ISP\_DEC\_LOSS\_QP\_MAP\_CHN2\_6 0xb7**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_2_27
20:16	R/W	0	reg_lut_budget2qp_2_26
12: 8	R/W	0	reg_lut_budget2qp_2_25
4: 0	R/W	0	reg_lut_budget2qp_2_24

**Table 7-611 ISP\_DEC\_LOSS\_QP\_MAP\_CHN2\_7 0xb8**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_2_31
20:16	R/W	0	reg_lut_budget2qp_2_30
12: 8	R/W	0	reg_lut_budget2qp_2_29
4: 0	R/W	0	reg_lut_budget2qp_2_28

**Table 7-612 ISP\_DEC\_LOSS\_QP\_MAP\_CHN2\_8 0xb9**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_2_35
20:16	R/W	0	reg_lut_budget2qp_2_34
12: 8	R/W	0	reg_lut_budget2qp_2_33
4: 0	R/W	0	reg_lut_budget2qp_2_32

**Table 7-613 ISP\_DEC\_LOSS\_QP\_MAP\_CHN2\_9 0xba**

Bit(s)	R/W	Default	Description
28:24	R/W	0	reg_lut_budget2qp_2_39
20:16	R/W	0	reg_lut_budget2qp_2_38
12: 8	R/W	0	reg_lut_budget2qp_2_37
4: 0	R/W	0	reg_lut_budget2qp_2_36

**Table 7-614 ISP\_DEC\_LOSS\_RC\_GROUP\_2 0xbb**

Bit(s)	R/W	Default	Description
31:16	R/W	1080	reg_rc_group_y: vertical size of group (super block) within picture for rate control algorithm
13: 0	R/W	24	reg_slcln_ratio: dynamic ratio to idx of extra bit budget for the cells within the line

**Table 7-615 ISP\_DEC\_LOSS\_RC\_BUDGET\_0 0xbc**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_rc_perln_16pec_xbdgt_3: extra bit budget (in pct= x/128) for the cells within the line, defined by 17 nodes to split the line into 16 pieces.
23:16	R/W	0	reg_rc_perln_16pec_xbdgt_2
15: 8	R/W	0	reg_rc_perln_16pec_xbdgt_1
7: 0	R/W	0	reg_rc_perln_16pec_xbdgt_0

**Table 7-616 ISP\_DEC\_LOSS\_RC\_BUDGET\_1 0xbd**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_rc_perln_16pec_xbdgt_7
23:16	R/W	0	reg_rc_perln_16pec_xbdgt_6
15: 8	R/W	0	reg_rc_perln_16pec_xbdgt_5
7: 0	R/W	0	reg_rc_perln_16pec_xbdgt_4

**Table 7-617 ISP\_DEC\_LOSS\_RC\_BUDGET\_2 0xae**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_rc_perln_16pec_xbdgt_11
23:16	R/W	0	reg_rc_perln_16pec_xbdgt_10
15: 8	R/W	0	reg_rc_perln_16pec_xbdgt_9.
7: 0	R/W	0	reg_rc_perln_16pec_xbdgt_8

**Table 7-618 ISP\_DEC\_LOSS\_RC\_BUDGET\_3 0xaf**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_rc_perln_16pec_xbdgt_15
23:16	R/W	0	reg_rc_perln_16pec_xbdgt_14
15: 8	R/W	0	reg_rc_perln_16pec_xbdgt_13
7: 0	R/W	0	reg_rc_perln_16pec_xbdgt_12

**Table 7-619 ISP\_DEC\_LOSS\_RC\_BUDGET\_4 0xb0**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_rc_perln_16pec_xbdgt_16
23:16	R/W	0	reg_rc_1stln_slice_xbdgt: extra bit budget (in pct= x/128) for the first line of the slice, no prediction from pre-line;
15: 8	R/W	0	reg_rc_2ndln_slice_xbdgt: extra bit budget (in pct= x/128) for the 2nd line of the slice, with limited prediction from pre-line;
7: 0	R/W	0	reg_rc_1stln_group_xbdgt: extra bit budget (in pct= x/128) for the first line of the group, still with prediction from pre-line;

**Table 7-620 ISP\_DEC\_LOSS\_RC\_BUDGET\_5 0xb1**

Bit(s)	R/W	Default	Description
29:24	R/W	23	reg_rc_master_qpx2_max: maxmum qpx2 during the rc_loop
21:16	R/W	14	reg_rc_master_qpx2_min: minmum qpx2 during the rc_loop
15: 8	R/W	16	reg_rc_dynamic_speed_long: dynamic speed for using up the accumed bits, 2nd order gain of loop filter. normalized to 16 as 1;
7: 0	R/W	16	reg_rc_dynamic_speed_short: dynamic speed for using up the accumed bits, 1st order gain of loop filter. normalized to 64 as 1;

**Table 7-621 ISP\_DEC\_LOSS\_RC\_BUDGET\_6 0xb2**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_rc_precell_bits_reset: enable for bits_cell_pre reset of each group 1, bits_cell_pre =0 0:bits_cell_pre value unchanged
28:16	R/W	240	reg_rc_dynamic_mxblk_long: maximum number of blocks in group for long term dynamic factor, if set to 0, then no limit; otherwise set limit
12: 0	R/W	240	reg_rc_qpx2_margin3_blkth:final guard margin threshold to blocks_left_in_group

**Table 7-622 ISP\_DEC\_LOSS\_RC\_QP\_MARGIN 0xb3**

Bit(s)	R/W	Default	Description
29:16	R/W	-12	reg_rc_qpx2_margin_thd_1: threshold x16 to ibits_accum to assign extra pq margin to avoid buffer down flow
13: 0	R/W	-4	reg_rc_qpx2_margin_thd_0

**Table 7-623 ISP\_DEC\_LOSS\_RC\_QP\_MARGIN\_1 0xb4**

Bit(s)	R/W	Default	Description
29:16	R/W	8	reg_rc_qpx2_margin_thd_3
13: 0	R/W	-16	reg_rc_qpx2_margin_thd_2

**Table 7-624 ISP\_DEC\_LOSS\_RC\_QP\_MARGIN\_2 0xb5**

Bit(s)	R/W	Default	Description
29:24	R/W	6	reg_rc_qpx2_margin_dlt_3: delta qp margin to qp_max when ibits_accum lower than rc_qp_margin_thd[2]
21:16	R/W	5	reg_rc_qpx2_margin_dlt_2
13: 8	R/W	3	reg_rc_qpx2_margin_dlt_1
5: 0	R/W	2	reg_rc_qpx2_margin_dlt_0

**Table 7-625 ISP\_DEC\_LOSS\_RC\_QP\_MARGIN\_3 0xb6**

Bit(s)	R/W	Default	Description
31:30	R/W	1	reg_rc_qpx2_margin_dlt_mode
27:16	R/W	250	reg_rc_fifo_qpx2_margin_thd_1: threshold to fifo to assign extra pq margin to avoid buffer down flow,
11: 0	R/W	150	reg_rc_fifo_qpx2_margin_thd_0

**Table 7-626 ISP\_DEC\_LOSS\_RC\_QP\_MARGIN\_4 0xb7**

Bit(s)	R/W	Default	Description
27:16	R/W	500	reg_rc_fifo_qpx2_margin_thd_3
11: 0	R/W	350	reg_rc_fifo_qpx2_margin_thd_2



**Table 7-627 ISP\_DEC\_LOSS\_RC\_QP\_MARGIN\_5 0xb8**

Bit(s)	R/W	Default	Description
29:24	R/W	16	reg_rc_fifo_qpx2_margin_dlt_3: delta qp margin to qp_max when fifo lower than rc_qp_margin_thd[2]
21:16	R/W	7	reg_rc_fifo_qpx2_margin_dlt_2
13: 8	R/W	5	reg_rc_fifo_qpx2_margin_dlt_1
5: 0	R/W	3	reg_rc_fifo_qpx2_margin_dlt_0

**Table 7-628 ISP\_DEC\_LOSS\_FLATNESS\_ADJ0 0xb9**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_adjsize_flatness_flag: enable signal for the flatness mode,adj_predicted_size should be small in flatness
30	R/W	0	reg_adjsize_complex_flag: enable signal for the complex mode,adj_predicted_size should be large in complex
29	R/W	0	reg_rc_fifo_avgspeed_use_sbudget: enable to use s_budget_block as budget_block: 0 use bits_perblk programable register, 1: use rc calculated s_budget_blk
28	R/W	0	reg_rc_bits_gap_dlt_down_en: to dlt_buget2x reduced based on bits_gap value
27:16	R/W	50	reg_adjsize_flatness_pixthd: area if(MAX-MIN) bigger than pixthd
12: 8	R/W	7	reg_adjsize_flatness_sizedlt: delta size margin to adj_predicted_size when adj_predicted_size lower than reg_adjsize_flatness_sizedlt
4: 0	R/W	2	reg_adjsize_flatness_reduce: the cell is flatness and adj_predicted_size > thd, adj_predicted_size = adj_predicted_size- reg_adjsize_flatness_reduce

**Table 7-629 ISP\_DEC\_LOSS\_FLATNESS\_ADJ1 0xba**

Bit(s)	R/W	Default	Description
27:16	R/W	512	reg_adjsize_complex_pixthd : area if(MAX-MIN) bigger than pixthd
12: 8	R/W	7	reg_adjsize_complex_sizedlt : delta size margin to adj_predicted_size when adj_predicted_size bigger than reg_adjsize_complex_sizedlt
4: 0	R/W	2	reg_adjsize_complex_increase: if the cell is complex and adj_predicted_size < thd,adj_predicted_size = adj_predicted_size + reg_adjsize_complex_increase

**Table 7-630 ISP\_DEC\_LOSS\_FIFO\_THD\_0 0xbb**

Bit(s)	R/W	Default	Description
23:12	R/W	280	reg_rc_fifo_margin_thd_5:threshold of fifo level(in words=16bits) to guard the rc loop by adding delta to p_budget2x
11: 0	R/W	250	reg_rc_fifo_margin_thd_4

Table 7-631 ISP\_DEC\_LOSS\_FIFO\_THD\_1 0xbc

Bit(s)	R/W	Default	Description
23:12	R/W	220	reg_rc_fifo_margin_thd_3
11: 0	R/W	165	reg_rc_fifo_margin_thd_2

Table 7-632 ISP\_DEC\_LOSS\_FIFO\_THD\_2 0xbd

Bit(s)	R/W	Default	Description
23:12	R/W	145	reg_rc_fifo_margin_thd_1
11: 0	R/W	130	reg_rc_fifo_margin_thd_0

Table 7-633 ISP\_DEC\_LOSS\_FIFO\_AVG 0xbe

Bit(s)	R/W	Default	Description
25:20	R/W	12	reg_rc_fifo_margin_dlt_5: delta of fifo level to guard the rc loop by adding delta to p_budget2x
17:12	R/W	14	reg_rc_fifo_margin_dlt_4:
11: 0	R/W	80	reg_rc_fifo_avgspeed_bits_perblk: bits perblock fifo read/write speed, set to budget_block = (SIZE_BLK*(ratio_bppx16) )/16 as default

Table 7-634 ISP\_DEC\_LOSS\_FIFO\_DLT 0xbf

Bit(s)	R/W	Default	Description
29:24	R/W	11	reg_rc_fifo_margin_dlt_3: delta of fifo level to guard the rc loop by adding delta to p_budget2x
21:16	R/W	8	reg_rc_fifo_margin_dlt_2
13: 8	R/W	5	reg_rc_fifo_margin_dlt_1
5: 0	R/W	2	reg_rc_fifo_margin_dlt_0

Table 7-635 ISP\_DEC\_LOSS\_BITSGAP\_THD\_0 0xc0

Bit(s)	R/W	Default	Description
27:16	R/W	50	reg_rc_bits_gap_margin_thd_1: threshold of bits gap to reduced dlt_buget2x
11: 0	R/W	30	reg_rc_bits_gap_margin_thd_0

Table 7-636 ISP\_DEC\_LOSS\_BITSGAP\_THD\_1 0xc1

Bit(s)	R/W	Default	Description
11: 0	R/W	82	reg_rc_bits_gap_margin_thd_2:threshold of bits gap to reduced dlt_buget2x

**Table 7-637 ISP\_DEC\_LOSS\_ICH\_0xc2**

Bit(s)	R/W	Default	Description
25	R/W	1	reg_ich_enable: enable signal for the ich mode.
24	R/W	1	reg_ich_error_compare_pred_en: ICH maxerror compare prediction maxerror en
20:16	R/W	5	reg_ich_max_error_sel: ICH error th sel, th = 1<<sel [0,DW-1)
12: 8	R/W	8	reg_ich_bits_weight: cost bits weight(0-16)
4: 0	R/W	8	reg_pre_bits_weight: cost bits weight (0-16)

**Table 7-638 ISP\_DEC\_LOSS\_ICH\_UPTH\_0xc3**

Bit(s)	R/W	Default	Description
7: 0	R/W	4	reg_ich_update_th: thresh for ICH history update

**Table 7-639 ISP\_DEC\_LOSS\_LUMA\_0xc4**

Bit(s)	R/W	Default	Description
16	R/W	0	reg_luma_adjust_en: enable of luma(G) adjust
12: 8	R/W	1	reg_pixel_luma_adj_dlt_1: of qelvel according to luma
4: 0	R/W	2	reg_pixel_luma_adj_dlt_0:

**Table 7-640 ISP\_DEC\_LOSS\_LUMA\_TH\_0xc5**

Bit(s)	R/W	Default	Description
31:16	R/W	128	reg_pixel_luma_adj_th_1: luma level
15: 0	R/W	256	reg_pixel_luma_adj_th_0:

**Table 7-641 ISP\_DEC\_LOSS\_ACCUM\_OFFSET\_0\_0xc6**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_accum_add_offset_2
23:16	R/W	5	reg_accum_add_offset_1
15: 8	R/W	2	reg_accum_add_offset_0
3: 1	R/W	2	reg_accum_offset_shift
0	R/W	0	reg_accum_offset_en

**Table 7-642 ISP\_DEC\_LOSS\_ACCUM\_OFFSET\_1\_0xc7**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_accum_add_offset_6
23:16	R/W	5	reg_accum_add_offset_5

Bit(s)	R/W	Default	Description
15: 8	R/W	2	reg_accum_add_offset_4
7: 0	R/W	0	reg_accum_add_offset_3

Table 7-643 ISP\_DEC\_LOSS\_ACCUM\_OFFSET\_2 0xc8

Bit(s)	R/W	Default	Description
15: 8	R/W	255	reg_accum_add_offset_8
7: 0	R/W	0	reg_accum_add_offset_7

Table 7-644 ISP\_DEC\_LOSS\_ACCUM\_OFFSET\_3 0xc9

Bit(s)	R/W	Default	Description
11: 0	R/W	0	reg_normalize_idx_ratio

Table 7-645 ISP\_DEC\_LOSS\_DERIVA\_ADJ 0xca

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_derivative_lut_0_9: lut of derivative adjusted
26:24	R/W	0	reg_derivative_lut_0_8
22:20	R/W	0	reg_derivative_lut_0_7
18:16	R/W	0	reg_derivative_lut_0_6
14:12	R/W	0	reg_derivative_lut_0_5
10: 8	R/W	0	reg_derivative_lut_0_4
6: 4	R/W	0	reg_derivative_lut_1_9
0	R/W	0	reg_derivative_en: 1 to enable derivative adjusted

Table 7-646 ISP\_DEC\_LOSS\_DERIVA\_ADJ\_1 0xcb

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_derivative_lut_1_8
26:24	R/W	0	reg_derivative_lut_1_7
22:20	R/W	0	reg_derivative_lut_1_6
18:16	R/W	0	reg_derivative_lut_1_5
14:12	R/W	0	reg_derivative_lut_2_9
10: 8	R/W	0	reg_derivative_lut_2_8
6: 4	R/W	0	reg_derivative_lut_2_7
2: 0	R/W	0	reg_derivative_lut_2_6

**Table 7-647 ISP\_DEC\_LOSS\_DERIVA\_ADJ\_2 0xcc**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_derivative_lut_9_5
26:24	R/W	0	reg_derivative_lut_9_4
22:20	R/W	0	reg_derivative_lut_9_3
18:16	R/W	0	reg_derivative_lut_9_2
14:12	R/W	0	reg_derivative_lut_9_1
10: 8	R/W	0	reg_derivative_lut_9_0
6: 4	R/W	0	reg_derivative_lut_8_4
2: 0	R/W	0	reg_derivative_lut_8_3

**Table 7-648 ISP\_DEC\_LOSS\_DERIVA\_ADJ\_3 0xcd**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_derivative_lut_8_2
26:24	R/W	0	reg_derivative_lut_8_1
22:20	R/W	0	reg_derivative_lut_8_0
18:16	R/W	0	reg_derivative_lut_7_4
14:12	R/W	0	reg_derivative_lut_7_3
10: 8	R/W	0	reg_derivative_lut_7_2
6: 4	R/W	0	reg_derivative_lut_7_1
2: 0	R/W	0	reg_derivative_lut_7_0

**Table 7-649 ISP\_DEC\_LOSS\_WDR\_LINE\_DELAY 0xce**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_wdr_mode_en: mode enable,(in this mode have line delay between long and short frame)
15: 0	R/W	4	reg_wdr_delay_line: delay between long and short frame in wdr mode

**Table 7-650 ISP\_DEC\_LOSS\_STATS\_RAM\_MODE 0xcf**

Bit(s)	R/W	Default	Description
1	R/W	0	reg_stats_slice_rdmode: 0 to read slice according to reg_slice_num_mode
0	R/W	0	reg_stats_ram_rden: 1 to read stats ram enable

**Table 7-651 ISP\_DEC\_LOSS\_STATS\_RAM\_ADDR 0xd0**

Bit(s)	R/W	Default	Description
8: 0	R/W	0	reg_stats_ram_addr: statistics info ram status address

Table 7-652 ISP\_DEC\_LOSS\_RO\_STATS\_RAM\_DATA 0xd1

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_stats_ram_data: statistics info from ram data

Table 7-653 ISP\_DEC\_LOSS\_MIF\_CTRL 0xf0

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_sync_sel: 0 to sync with frm rst [24]: for enable [25]: for reg_canvas_id [26]: for dol mode
23:16	R/W	0	reg_canvas_id: axi canvas id num
14:12	R/W	1	reg_cmd_intr_len: interrupt send cmd when how many series axi cmd, 0=12 1=16 2=24 3=32 4=40 5=48 6=56 7=64
9:8	R/W	1	reg_cmd_req_size: how many room fifo have, then axi send series req, 0=16 1=32 2=48 3=64
6:4	R/W	3	reg_burst_len: burst type: 0->1; 1->2; 2->4; 3->8; 4->16, others reserved
2	R/W	0	reg_check_resp_id: 1 to check RESP ID
1	R/W	0	reg_dol_mode: 1: DOL mode
0	R/W	1	reg_mif_enable: 1 to enable mif

Table 7-654 ISP\_DEC\_LOSS\_MIF\_QOS 0xf1

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_qos_ctrl: Qos control [18]: reg_qos_auto_en [17]: reg_qos_sup [16]: reg_qos_ini [15:8]: reg_qos_up_th [7:0]: reg_qos_dn_th

Table 7-655 ISP\_DEC\_LOSS\_MIF\_URGENT 0xf2

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_urgent_ctrl: urgent control [18]: reg_urgent_auto_en [17]: reg_urgent_sup [16]: reg_urgent_ini [15:8]: reg_urgent_up_th [7:0]: reg_urgent_dn_th

**Table 7-656 ISP\_DEC\_LOSS\_MIF\_MISC 0xf3**

Bit(s)	R/W	Default	Description
3	R/W	1	reg_mif_busy_noresp_check: for mif write, dont take "respond busy" as mif unit busy.
2	R/W	0	reg_mif_wait_input4busy: use in encoder, mif is busy till input data comming. only for mif wr
1	R/W	0	reg_mif_dbl_baddr_init: write 1 to use reg_mif_baddr as base-addr for next frame,it would be clear automatically.
0	R/W	0	reg_mif_dbl_baddr_en:

**Table 7-657 ISP\_DEC\_LOSS\_MIF\_BADDR 0xf4**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_mif_baddr: mif base address

**Table 7-658 ISP\_DEC\_LOSS\_MIF\_BADDR1 0xf5**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_mif_baddr1: mif base address

**Table 7-659 ISP\_DEC\_LOSS\_MIF\_FIFO\_CTRL 0xf7**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_mif_gclk_ctrl: gate clock control
7:0	R/W	128	reg_mif_fifo_size: data FIFO max size, default = 128,

**Table 7-660 ISP\_DEC\_LOSS\_MIF\_FRM\_HOLD 0xf8**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_mif_frm_holds: clock cycle holder from frm_rst to wait register ready, default = 16 Note: set bit[31] to 1 indicates the compress would start to work till RDMA transfer finished or write [31:0] to 0xffffffff

**Table 7-661 ISP\_DEC\_LOSS\_MIF\_RO\_STATS 0xf9**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_mif_status: mif status

### 7.12.5.3 MIPI APAPT Registers

The base address is 0xFE3B4000

**Table 7-662 MIPI\_ADAPT\_CMPR\_SPLIT\_CTRL 0x00**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_split_sync_ctrl: 0 for register sync (shadow)
3	R/W	1	reg_split_ignore_frag_cmd: ignore no used axi cmd before cmd with address = "reg_wsub0_iaddr_bgn"
1	R/W	0	reg_split_big_endian: 128bit data big_endian
0	R/W	1	reg_split_enable: 1 to enable

**Table 7-663 MIPI\_ADAPT\_CMPR\_SPLIT\_SIZE 0x01**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_split_gclk_ctrl: gate clock control
15:0	R/W	4608	reg_split_hsize: horizontal size

**Table 7-664 MIPI\_ADAPT\_CMPR\_SPLIT\_FRMRST\_CTRL 0x02**

Bit(s)	R/W	Default	Description
20	R/W	1	reg_split_abort_after_cmd_1st
19	R/W	1	reg_split_internal_abort_en
18	R/W	1	reg_split_abort_hold_input
17	R/W	1	reg_split_frmrst_hold_input: 1 to hold axi input during frmrst generate, only valid for reg_frmrst_mode =0/2
16	R/W	0	reg_split_rdma_trigger_mode: 0 to use frmrst, 1 to use frame_end
15	R/W	0	reg_split_rdma_enable: 1 to enable RDMA
13	R/W	0	reg_split_frmrst_hold4abort_mode: 1 to start frmrst delay counter caused by abort after enc path idle.
12	R/W	0	reg_split_frmrst_hold4gen_mode: 1 to start frmrst delay counter after enc path idle.
11	R/W	0	reg_split_dis_abort: disable abnormal abort processing.
10	R/W	0	reg_split_dis_frmrst_by_abort: disable frmrst generate caused by abnormal abort
9	R/W	1	reg_split_wait_idle_for_abort: wait enc path idle before frmrst generate cause by abnormal abort.
8	R/W	1	reg_split_wait_idle_for_frmrst: wait enc path idle before frmrst generate.
5 : 4	R/W	0	reg_split_frmrst_mode:frmrst mode, 0:use input frmrst; 1:use reg_soft_frmrst; 2: use axi iaddr_end to generate; 3: use iaddr_bgn to generate
0	R/W	0	reg_split_soft_frmrst: write 1 to generate a soft frmrst

**Table 7-665 MIPI\_ADAPT\_CMPR\_SPLIT\_FRMRST\_DLY0 0x03**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_split_frmrst_dlys4gen: clock delays for frmrst generate except caused by "abnormal abort"



**Table 7-666 MIPI\_ADAPT\_CMPR\_SPLIT\_FRMRST\_DLY1 0x04**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_split_frmrst_dlys4abort: clock delays for frmrst generate caused by "abnormal abort"

**Table 7-667 MIPI\_ADAPT\_CMPR\_SPLIT\_FRMHOLD 0x05**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_split_frm_holds: clock cycle holder from frm_rst,to wait register ready, default = 16

**Table 7-668 MIPI\_ADAPT\_CMPR\_RO\_SPLIT\_STATS 0x07**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_split_status

**Table 7-669 MIPI\_ADAPT\_CMPR\_WSUB0\_CTRL 0x08**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_wsub0_raw67_ext:
18:16	R/W	1	reg_wsub0_raw_mode: raw6/7/8/10/12/14
9 : 0	R/W	32	reg_wsub0_dfifo_size: data FIFO size

**Table 7-670 MIPI\_ADAPT\_CMPR\_WSUB0\_IADDRS 0x09**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_wsub0_iaddr_bgn: start address for WSUB0

**Table 7-671 MIPI\_ADAPT\_CMPR\_WSUB0\_IADDRE 0x0a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_wsub0_iaddr_end: end address for WSUB0

**Table 7-672 MIPI\_ADAPT\_CMPR\_PACK\_CTRL 0x10**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_pack_sync_ctrl: register sync (shadow) ctrl
6	R/W	0	reg_pack_skip_flag_init: skip flag for pack, 1 to skip pack axi cmd for current frame
5	R/W	0	reg_pack_skip_auto_ctrl: auto control for skip flag, if 1, the 1st frame use the reg_pack_skip_init as skip flag, and the next frames use 0 for skip flag
3	R/W	1	reg_pack_ignore_frag_cmd: ignore no used axi cmd before cmd with address = "reg_rsub0_iaddr_bgn"

Bit(s)	R/W	Default	Description
1	R/W	0	reg_pack_big_endian: Big_endian
0	R/W	1	reg_pack_enable: 1 to enable

Table 7-673 MIPI\_ADAPT\_CMPR\_PACK\_SIZE 0x11

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_pack_gclk_ctrl: gate clock control
15:0	R/W	4608	reg_pack_hsize: hsize

Table 7-674 MIPI\_ADAPT\_CMPR\_PACK\_FRMRST\_CTRL 0x12

Bit(s)	R/W	Default	Description
24	R/W	0	reg_pack_busy_care_rdmif : 1 to take rdmif busy as pack busy
20	R/W	1	reg_pack_abort_after_cmd_1st :
19	R/W	1	reg_pack_internal_abort_en :
18	R/W	1	reg_pack_abort_hold_input :
17	R/W	1	reg_pack_frmrst_hold_input: 1 to hold axi input during frmrst generate, only valid for reg_frmrst_mode =0/2
16	R/W	0	reg_pack_rdma_trigger_mode: 0 to use frmrst, 1 to use frame_end
15	R/W	0	reg_pack_rdma_enable: 1 to enable RDMA
13	R/W	0	reg_pack_frmrst_hold4abort_mode: 1 to start frmrst delay counter caused by abort after enc path idle.
12	R/W	0	reg_pack_frmrst_hold4gen_mode:1 to start frmrst delay counter after enc path idle.
11	R/W	0	reg_pack_dis_abort: disable abnormal abort processing.
10	R/W	0	reg_pack_dis_frmrst_by_abort: disable frmrst generate caused by abnormal abort
9	R/W	1	reg_pack_wait_idle_for_abort: wait enc path idle before frmrst generate cause by abnormal abort.
8	R/W	1	reg_pack_wait_idle_for_frmrst: wait enc path ready before frmrst generate.
5 : 4	R/W	0	reg_pack_frmrst_mode: frmrst mode, 0:use input frmrst; 1:use reg_soft_frmrst; 2: use axi iaddr_end to generate; 3: use iaddr_bgn to generate
0	R/W	0	reg_pack_soft_frmrst: write 1 to generate a soft frmrst

Table 7-675 MIPI\_ADAPT\_CMPR\_PACK\_FRMRST\_DLY0 0x13

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_pack_frmrst_dlys4gen: clock delays for frmrst generate except caused by "abnormal abort"

**Table 7-676 MIPI\_ADAPT\_CMPR\_PACK\_FRMRST\_DLY1 0x14**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_pack_frmrst_dlys4abort: clock delays for frmrst generate caused by "abnormal abort"

**Table 7-677 MIPI\_ADAPT\_CMPR\_PACK\_FRMHOLD 0x15**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_pack_frm_holds: clock cycle holder from frm_rst,to wait register ready, default = 16

**Table 7-678 MIPI\_ADAPT\_CMPR\_RO\_PACK\_STATS 0x17**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_pack_status:

**Table 7-679 MIPI\_ADAPT\_CMPR\_RSUB0\_CTRL 0x18**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_rsub0_raw67_ext,
18:16	R/W	1	reg_rsub0_raw_mode: raw6/7/8/10/12/14
9 : 0	R/W	32	reg_rsub0_dfifo_size: Data FIFO size

**Table 7-680 MIPI\_ADAPT\_CMPR\_RSUB0\_IADDRS 0x19**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_rsub0_iaddr_bgn: start address for RSUB0

**Table 7-681 MIPI\_ADAPT\_CMPR\_RSUB0\_IADDRE 0x1a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_rsub0_iaddr_end: end address for RSUB0

**Table 7-682 MIPI\_ADAPT\_CMPR\_INT\_EN 0x20**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_adapt_int_en: interrupt mask

**Table 7-683 MIPI\_ADAPT\_CMPR\_RO\_INT\_STATUS 0x21**

Bit(s)	R/W	Default	Description
31:0	R.O	0x0	ro_adapt_int_status: interrupt status, write 1 to clear : TNR RSUB (for compress dec) RDMA END : TNR WSUB (for compress enc) RDMA END : RSUB (dec) internal frm_rst generate : RSUB (dec) abort occurred : RSUB(dec) frame end : WSUB (enc) internal frm_rst generate : WSUB (enc) abort occurred [2] : WSUB(enc) frame end

### 7.12.5.4 MIPI TNR\_CMPR Registers

The base address is 0xFE3B4800

**Table 7-684 MIPI\_TNR\_CMPR\_SPLIT\_CTRL 0x00**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_split_sync_ctrl: 0 for register sync (shadow)
3	R/W	1	reg_split_ignore_frag_cmd: ignore no used axi cmd before cmd with address = iaddr_bgn
2	R/W	1	reg_split_tnr_temp3_en: 1 to indicate TNR works in Temp3 mode
1	R/W	0	reg_split_big_endian: 128bit data big_endian
0	R/W	1	reg_split_enable: 1 to enable

**Table 7-685 MIPI\_TNR\_CMPR\_SPLIT\_SIZE 0x01**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_split_gclk_ctrl: gate clock control
15:0	R/W	4608	reg_split_hsize: horizontal size

**Table 7-686 MIPI\_TNR\_CMPR\_SPLIT\_FRMRST\_CTRL 0x02**

Bit(s)	R/W	Default	Description
25	R/W	0	reg_split_addrs_cnt_mode: 1 to hold in_abort after all sub addrs come, otherwise one sub addrs is ok.
20	R/W	1	reg_split_abort_after_cmd_1st :
19	R/W	1	reg_split_internal_abort_en:
18	R/W	1	reg_split_abort_hold_input:
17	R/W	1	reg_split_frmrst_hold_input: 1 to hold axi input during frmrst generate
16	R/W	0	reg_split_rdma_trigger_mode: 0 to use frmrst, 1 to use frame_end
15	R/W	0	reg_split_rdma_enable: 1 to enable RDMA
13	R/W	0	reg_split_frmrst_hold4abort_mode: 1 to start frmrst delay counter caused by abort after enc path idle.

Bit(s)	R/W	Default	Description
12	R/W	0	reg_split_frmrst_hold4gen_mode: 1 to start frmrst delay counter after enc path idle.
11	R/W	0	reg_split_dis_abort: disable abnormal abort processing.
10	R/W	0	reg_split_dis_frmrst_by_abort: disable frmrst generate caused by abnormal abort
9	R/W	1	reg_split_wait_idle_for_abort: wait enc path idle before frmrst generate cause by abnormal abort.
8	R/W	1	reg_split_wait_idle_for_frmrst: wait enc path idle before frmrst generate.
5 : 4	R/W	0	reg_split_frmrst_mode: frmrst mode, 0:use input frmrst; 1:use reg_soft_frmrst; 2: use axi iaddr_end to generate; 3: use iaddr_bgn to generate
0	R/W	0	reg_split_soft_frmrst: write 1 to generate a soft frmrst

**Table 7-687 MIPI\_TNR\_CMPR\_SPLIT\_FRMRST\_DLY0 0x03**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_split_frmrst_dlys4gen: clock delays for frmrst generate except caused by "- abnormal abort"

**Table 7-688 MIPI\_TNR\_CMPR\_SPLIT\_FRMRST\_DLY1 0x04**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_split_frmrst_dlys4abort: clock delays for frmrst generate caused by "- abnormal abort"

**Table 7-689 MIPI\_TNR\_CMPR\_SPLIT\_FRMHOLD 0x05**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_split_frm_holds: clock cycle holder from frm_rst,to wait register ready, default = 16

**Table 7-690 MIPI\_TNR\_CMPR\_RO\_SPLIT\_STATS 0x07**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_split_status

**Table 7-691 MIPI\_TNR\_CMPR\_WSUB0\_CTRL 0x08**

Bit(s)	R/W	Default	Description
16	R/W	1	reg_wsub0_bit16_mode: 0 for 12bit mode; 1 for 16bit mode
9 : 0	R/W	128	reg_wsub0_dfifo_size: Data FIFO size

**Table 7-692 MIPI\_TNR\_CMPR\_WSUB0\_IADDRS 0x09**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_wsub0_iaddr_bgn: start address for WSUB0

**Table 7-693 MIPI\_TNR\_CMPR\_WSUB0\_IADDRE 0x0a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_wsub0_iaddr_end: end address for WSUB0

**Table 7-694 MIPI\_TNR\_CMPR\_WSUB1\_CTRL 0x0b**

Bit(s)	R/W	Default	Description
16	R/W	1	reg_wsub1_bit16_mode: 0 for 12bit mode; 1 for 16bit mode
9 : 0	R/W	128	reg_wsub1_dfifo_size : Data FIFO size

**Table 7-695 MIPI\_TNR\_CMPR\_WSUB1\_IADDRS 0x0c**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_wsub1_iaddr_bgn: start address for WSUB1

**Table 7-696 MIPI\_TNR\_CMPR\_WSUB1\_IADDRE 0x0d**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_wsub1_iaddr_end: end address for WSUB1

**Table 7-697 MIPI\_TNR\_CMPR\_PACK\_CTRL 0x10**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_pack_sync_ctrl: register sync (shadow) ctrl
6	R/W	0	reg_pack_skip_flag_init: skip flag for pack, 1 to skip pack axi cmd for current frame
5	R/W	0	reg_pack_skip_auto_ctrl: auto control for skip flag, if 1, the 1st frame use the reg_pack_skip_init as skip flag, and the next frames use 0 for skip flag
4	R/W	1	reg_pack_keep_axicmd_order: 1 to keep axi rdata cmd order same as axi cmd order
3	R/W	1	reg_pack_ignore_frag_cmd: ignore no used axi cmd before cmd with address = iaddr_bgn
2	R/W	1	reg_pack_tnr_temp3_en: 1 to indicate TNR works in Temp3 mode
1	R/W	0	reg_pack_big_endian: Big_endian
0	R/W	1	reg_pack_enable: 1 to enable

**Table 7-698 MIPI\_TNR\_CMPR\_PACK\_SIZE 0x11**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_pack_gclk_ctrl: gate clock size
15:0	R/W	4608	reg_pack_hsize: hsize

**Table 7-699 MIPI\_TNR\_CMPR\_PACK\_FRMRST\_CTRL 0x12**

Bit(s)	R/W	Default	Description
25	R/W	0	reg_pack_addrs_cnt_mode: 1 to hold in_abort after all sub addrs come, otherwise one sub addrs is ok.
24	R/W	0	reg_pack_busy_care_rdmif: 1 to take rdmif busy as pack busy
20	R/W	1	reg_pack_abort_after_cmd_1st:
19	R/W	1	reg_pack_internal_abort_en:
18	R/W	1	reg_pack_abort_hold_input:
17	R/W	1	reg_pack_frmrst_hold_input: 1 to hold axi input during frmrst generate, only valid for reg_frmrst_mode =0/2
16	R/W	0	reg_pack_rdma_trigger_mode: 0 to use frmrst, 1 to use frame_end
15	R/W	0	reg_pack_rdma_enable: 1 to enable RDMA
13	R/W	0	reg_pack_frmrst_hold4abort_mode: 1 to start frmrst delay counter caused by abort after enc path idle.
12	R/W	0	reg_pack_frmrst_hold4gen_mode: 1 to start frmrst delay counter after enc path idle.
11	R/W	0	reg_pack_dis_abort: disable abnormal abort processing.
10	R/W	0	reg_pack_dis_frmrst_by_abort : disable frmrst generate caused by abnormal abort
9	R/W	1	reg_pack_wait_idle_for_abort : wait enc path idle before frmrst generate cause by abnormal abort.
8	R/W	1	reg_pack_wait_idle_for_frmrst: wait enc path ready before frmrst generate.
5 : 4	R/W	0	reg_pack_frmrst_mode: frmrst mode, 0:use input frmrst; 1:use reg_soft_frmrst; 2: use axi iaddr_end to generate; 3: use iaddr_bgn to generate
0	R/W	0	reg_pack_soft_frmrst: write 1 to generate a soft frmrst

**Table 7-700 MIPI\_TNR\_CMPR\_PACK\_FRMRST\_DLY0 0x13**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_pack_frmrst_dlys4gen: clock delays for frmrst generate except caused by "-abnormal abort"

**Table 7-701 MIPI\_TNR\_CMPR\_PACK\_FRMRST\_DLY1 0x14**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_pack_frmrst_dlys4abort: clock delays for frmrst generate caused by "-abnormal abort"

**Table 7-702 MIPI\_TNR\_CMPR\_PACK\_FRMHOLD 0x15**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_pack_frm_holds: clock cycle holder from frm_rst,to wait register ready, default = 16

**Table 7-703 MIPI\_TNR\_CMPR\_RO\_PACK\_STATS 0x17**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_pack_status

**Table 7-704 MIPI\_TNR\_CMPR\_RSUB0\_CTRL 0x18**

Bit(s)	R/W	Default	Description
16	R/W	1	reg_rsub0_bit16_mode: 0 for 12 bit mode, 1 for 16 bit mode
9 : 0	R/W	128	reg_rsub0_dfifo_size: Data FIFO size

**Table 7-705 MIPI\_TNR\_CMPR\_RSUB0\_IADDRS 0x19**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_rsub0_iaddr_bgn: start address for RSUB0

**Table 7-706 MIPI\_TNR\_CMPR\_RSUB0\_IADDRE 0x1a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_rsub0_iaddr_end: end address for RSUB0

**Table 7-707 MIPI\_TNR\_CMPR\_RSUB1\_CTRL 0x1b**

Bit(s)	R/W	Default	Description
16	R/W	1	reg_rsub1_bit16_mode : 0 for 12 bit mode, 1 for 16 bit mode
9 : 0	R/W	128	reg_rsub1_dfifo_size: Data FIFO size

**Table 7-708 MIPI\_TNR\_CMPR\_RSUB1\_IADDRS 0x1c**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_rsub1_iaddr_bgn: start address for RSUB1

**Table 7-709 MIPI\_TNR\_CMPR\_RSUB1\_IADDRE 0x1d**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_rsub1_iaddr_end: end address for RSUB1



**Table 7-710 MIPI\_TNR\_CMPR\_INT\_EN 0x20**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_tnr_int_en: interrupt mask

**Table 7-711 MIPI\_TNR\_CMPR\_RO\_INT\_STATUS 0x21**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_tnr_int_status: interrupt status, write 1 to clear : TNR RSUB (for compress dec) RDMA END : TNR WSUB (for compress enc) RDMA END : RSUB (dec) internal frm_rst generate : RSUB (dec) abort occurred : RSUB(dec) frame end : WSUB (enc) internal frm_rst generate : WSUB (enc) abort occurred [2] : WSUB(enc) frame end

**Table 7-712 MIPI\_TNR\_META\_WSUB0\_MIF\_CTRL 0x80**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_meta_wsub0_sync_sel: sync with frm rst
23:16	R/W	0	reg_meta_wsub0_canvas_id: axi canvas id num
14:12	R/W	1	reg_meta_wsub0_cmd_intr_len: interrupt send cmd when how many series axi cmd, 0=12 1=16 2=24 3=32 4=40 5=48 6=56 7=64
9: 8	R/W	1	reg_meta_wsub0_cmd_req_size: how many room fifo have, then axi send series req, 0=16 1=32 2=24 3=64
6: 4	R/W	3	reg_meta_wsub0_burst_len: burst type: 0->1; 1->2; 2->4; 3->8; 4->16, others reserved
2	R/W	0	reg_meta_wsub0_check_resp_id:
0	R/W	1	reg_meta_wsub0_mif_enable: 1 to enable mif

**Table 7-713 ISP\_TNR\_META\_WSUB0\_MIF\_QOS 0x81**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_wsub0_qos_ctrl: Qos control [18] : reg_qos_auto_en [17] : reg_qos_sup [16] : reg_qos_ini [15:8]: reg_qos_up_th [ 7:0]: reg_qos_dn_th

**Table 7-714 MIPI\_TNR\_META\_WSUB0\_MIF\_URGENT 0x82**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_wsub0_urgent_ctrl: urgent control [18] : reg_urgent_auto_en [17] : reg_urgent_sup [16] : reg_urgent_ini [15:8]: reg_urgent_up_th [ 7:0]: reg_urgent_dn_th

**Table 7-715 MIPI\_TNR\_META\_WSUB0\_MIF\_MISC 0x83**

Bit(s)	R/W	Default	Description
3	R/W	1	reg_meta_wsub0_mif_busy_noresp_check: for mif write, dont take "respond busy" as mif unit busy.
2	R/W	0	reg_meta_wsub0_mif_wait_input4busy: use in encoder, mif is busy till input data comming.
1	R/W	0	reg_meta_wsub0_mif_dbl_baddr_init: 1 to use reg_mif_baddr as base-addr for next frame,it would be clear automatically
0	R/W	0	reg_meta_wsub0_mif_dbl_baddr_en: 1 to use ping-pong base-addr based on frame.

**Table 7-716 MIPI\_TNR\_META\_WSUB0\_MIF\_BADDR 0x84**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_wsub0_mif_baddr: mif base address

**Table 7-717 MIPI\_TNR\_META\_WSUB0\_MIF\_BADDR1 0x85**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_wsub0_mif_baddr1: mif base address

**Table 7-718 MIPI\_TNR\_META\_WSUB0\_MIF\_SIZE 0x86**

Bit(s)	R/W	Default	Description
23:0	R/W	0	reg_meta_wsub0_mif_size: mif trnasfer size, based on 128bits

**Table 7-719 MIPI\_TNR\_META\_WSUB0\_MIF\_FIFO 0x87**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_meta_wsub0_mif_gclk_ctrl: gate clock control
7 :0	R/W	128	reg_meta_wsub0_mif_fifo_size: data FIFO ize

**Table 7-720 MIPI\_TNR\_META\_WSUB0\_MIF\_FRMHOLD 0x88**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_meta_wsub0_mif_frm_holds:generate mif_start after reg_mif_frm_holds clock cycle from frm_rst

**Table 7-721 MIPI\_TNR\_META\_WSUB0\_MIF\_RO\_STATS 0x89**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_meta_wsub0_mif_status: mif status

Table 7-722 MIPI\_TNR\_META\_WSUB1\_MIF\_CTRL 0x90

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_meta_wsub1_sync_sel: sync with frm rst
23:16	R/W	0	reg_meta_wsub1_canvas_id: axi canvas id num
14:12	R/W	1	reg_meta_wsub1_cmd_intr_len: interrupt send cmd when how many series axi cmd, 0=12 1=16 2=24 3=32 4=40 5=48 6=56 7=64
9: 8	R/W	1	reg_meta_wsub1_cmd_req_size: how many room fifo have, then axi send series req, 0=16 1=32 2=24 3=64
6: 4	R/W	3	reg_meta_wsub1_burst_len: burst type: 0->1; 1->2; 2->4; 3->8; 4->16, others reserved
2	R/W	0	reg_meta_wsub1_check_resp_id:
0	R/W	1	reg_meta_wsub1_mif_enable: 1 to enable mif

Table 7-723 ISP\_TNR\_META\_WSUB1\_MIF\_QOS 0x91

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_wsub1_qos_ctrl: Qos control [18]: reg_qos_auto_en [17]: reg_qos_sup [16]: reg_qos_ini [15:8]: reg_qos_up_th [ 7:0]: reg_qos_dn_th

Table 7-724 MIPI\_TNR\_META\_WSUB1\_MIF\_URGENT 0x92

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_wsub1_urgent_ctrl: urgent control [18]: reg_urgent_auto_en [17]: reg_urgent_sup [16]: reg_urgent_ini [15:8]: reg_urgent_up_th [ 7:0]: reg_urgent_dn_th

Table 7-725 MIPI\_TNR\_META\_WSUB1\_MIF\_MISC 0x93

Bit(s)	R/W	Default	Description
3	R/W	1	reg_meta_wsub1_mif_busy_noresp_check: for mif write, dont take "respond busy" as mif unit busy.
2	R/W	0	reg_meta_wsub1_mif_wait_input4busy: use in encoder, mif is busy till input data comming.
1	R/W	0	reg_meta_wsub1_mif_dbl_baddr_init: 1 to use reg_mif_baddr as base-addr for next frame,it would be clear automatically
0	R/W	0	reg_meta_wsub1_mif_dbl_baddr_en: 1 to use ping-pong base-addr based on frame.

**Table 7-726 MIPI\_TNR\_META\_WSUB1\_MIF\_BADDR 0x94**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_wsub1_mif_baddr: mif base address

**Table 7-727 MIPI\_TNR\_META\_WSUB1\_MIF\_BADDR1 0x95**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_wsub1_mif_baddr1: mif base address

**Table 7-728 MIPI\_TNR\_META\_WSUB1\_MIF\_SIZE 0x96**

Bit(s)	R/W	Default	Description
23:0	R/W	0	reg_meta_wsub1_mif_size: mif transfer size, based on 128bits

**Table 7-729 MIPI\_TNR\_META\_WSUB1\_MIF\_FIFO 0x97**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_meta_wsub1_mif_gclk_ctrl: gate clock control
7:0	R/W	64	reg_meta_wsub1_mif_fifo_size: data FIFO size

**Table 7-730 MIPI\_TNR\_META\_WSUB1\_MIF\_FRMHOLD 0x98**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_meta_wsub1_mif_frm_holds: generate mif_start after reg_mif_frm_holds clock cycle from frm_rst

**Table 7-731 MIPI\_TNR\_META\_WSUB1\_MIF\_RO\_STATS 0x99**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_meta_wsub1_mif_status: mif status

**Table 7-732 MIPI\_TNR\_META\_RSUB0\_MIF\_CTRL 0xa0**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_meta_rsub0_sync_sel: sync with frm rst
23:16	R/W	0	reg_meta_rsub0_canvas_id: axi canvas id num
14:12	R/W	1	reg_meta_rsub0_cmd_intr_len: interrupt send cmd when how many series axi cmd, 0=12 1=16 2=24 3=32 4=40 5=48 6=56 7=64
9:8	R/W	1	reg_meta_rsub0_cmd_req_size: how many room fifo have, then axi send series req, 0=16 1=32 2=24 3=64
6:4	R/W	3	reg_meta_rsub0_burst_len: burst type: 0->1; 1->2; 2->4; 3->8; 4->16, others reserved
2	R/W	0	reg_meta_rsub0_check_resp_id:
0	R/W	1	reg_meta_rsub0_mif_enable: 1 to enable mif

**Table 7-733 ISP\_TNR\_META\_RSUB0\_MIF\_QOS 0xa1**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_rsub0_qos_ctrl: Qos control [18]: reg_qos_auto_en [17]: reg_qos_sup [16]: reg_qos_ini [15:8]: reg_qos_up_th [7:0]: reg_qos_dn_th

**Table 7-734 MIPI\_TNR\_META\_RSUB0\_MIF\_URGENT 0xa2**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_rsub0_urgent_ctrl: urgent control [18]: reg_urgent_auto_en [17]: reg_urgent_sup [16]: reg_urgent_ini [15:8]: reg_urgent_up_th [7:0]: reg_urgent_dn_th

**Table 7-735 MIPI\_TNR\_META\_RSUB0\_MIF\_MISC 0xa3**

Bit(s)	R/W	Default	Description
1	R/W	0	reg_meta_rsub0_mif_dbl_baddr_init: 1 to use reg_mif_baddr as base-addr for next frame,it would be clear automatically
0	R/W	0	reg_meta_rsub0_mif_dbl_baddr_en: 1 to use ping-pong base-addr based on frame.

**Table 7-736 MIPI\_TNR\_META\_RSUB0\_MIF\_BADDR 0xa4**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_rsub0_mif_baddr: mif base address

**Table 7-737 MIPI\_TNR\_META\_RSUB0\_MIF\_BADDR1 0xa5**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_rsub0_mif_baddr1: mif base address

**Table 7-738 MIPI\_TNR\_META\_RSUB0\_MIF\_SIZE 0xa6**

Bit(s)	R/W	Default	Description
23:0	R/W	0	reg_meta_rsub0_mif_size: mif transfer size, based on 128bits

**Table 7-739 MIPI\_TNR\_META\_RSUB0\_MIF\_FIFO 0xa7**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_meta_rsub0_mif_gclk_ctrl: gate clock control
7 :0	R/W	64	reg_meta_rsub0_mif_fifo_size: data FIFO size

**Table 7-740 MIPI\_TNR\_META\_RSUB0\_MIF\_FRMHOLD 0xa8**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_meta_rsub0_mif_frm_holds: clock cycle holder from frm_rst to wait register ready

**Table 7-741 MIPI\_TNR\_META\_RSUB0\_MIF\_RO\_STATS 0xa9**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_meta_rsub0_mif_status: mif status

**Table 7-742 MIPI\_TNR\_META\_RSUB1\_MIF\_CTRL 0xb0**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_meta_rsub1_sync_sel: sync with frm rst
23:16	R/W	0	reg_meta_rsub1_canvas_id: axi canvas id num
14:12	R/W	1	reg_meta_rsub1_cmd_intr_len: interrupt send cmd when how many series axi cmd, 0=12 1=16 2=24 3=32 4=40 5=48 6=56 7=64
9: 8	R/W	1	reg_meta_rsub1_cmd_req_size: how many room fifo have, then axi send series req, 0=16 1=32 2=24 3=64
6: 4	R/W	3	reg_meta_rsub1_burst_len: burst type: 0->1; 1->2; 2->4; 3->8; 4->16, others reserved
2	R/W	0	reg_meta_rsub1_check_resp_id:
0	R/W	1	reg_meta_rsub1_mif_enable: 1 to enable mif

**Table 7-743 ISP\_TNR\_META\_RSUB1\_MIF\_QOS 0xb1**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_rsub1_qos_ctrl: Qos control [18] : reg_qos_auto_en [17] : reg_qos_sup [16] : reg_qos_ini [15:8]: reg_qos_up_th [ 7:0]: reg_qos_dn_th

**Table 7-744 MIPI\_TNR\_META\_RSUB1\_MIF\_URGENT 0xb2**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_rsub1_urgent_ctrl: urgent control [18]: reg_urgent_auto_en [17]: reg_urgent_sup [16]: reg_urgent_ini [15:8]: reg_urgent_up_th [7:0]: reg_urgent_dn_th

**Table 7-745 MIPI\_TNR\_META\_RSUB1\_MIF\_MISC 0xb3**

Bit(s)	R/W	Default	Description
1	R/W	0	reg_meta_rsub1_mif_dbl_baddr_init: 1 to use reg_mif_baddr as base-addr for next frame,it would be clear automatically
0	R/W	0	reg_meta_rsub1_mif_dbl_baddr_en: 1 to use ping-pong base-addr based on frame.

**Table 7-746 MIPI\_TNR\_META\_RSUB1\_MIF\_BADDR 0xb4**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_rsub1_mif_baddr: mif base address

**Table 7-747 MIPI\_TNR\_META\_RSUB1\_MIF\_BADDR1 0xb5**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_meta_rsub1_mif_baddr1: mif base address

**Table 7-748 MIPI\_TNR\_META\_RSUB1\_MIF\_SIZE 0xb6**

Bit(s)	R/W	Default	Description
23:0	R/W	0	reg_meta_rsub1_mif_size: mif transfer size, based on 128bits

**Table 7-749 MIPI\_TNR\_META\_RSUB1\_MIF\_FIFO 0xb7**

Bit(s)	R/W	Default	Description
31:16	R/W	0	reg_meta_rsub1_mif_gclk_ctrl: gate clock control
7:0	R/W	128	reg_meta_rsub1_mif_fifo_size: data FIFO size

**Table 7-750 MIPI\_TNR\_META\_RSUB1\_MIF\_FRMHOLD 0xb8**

Bit(s)	R/W	Default	Description
31:0	R/W	16	reg_meta_rsub1_mif_frm_holds:clock cycle holder from frm_rst,to wait register ready

Table 7-751 MIPI\_TNR\_META\_RSUB1\_MIF\_RO\_STATS 0xb9

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_meta_rsub1_mif_status: mif status

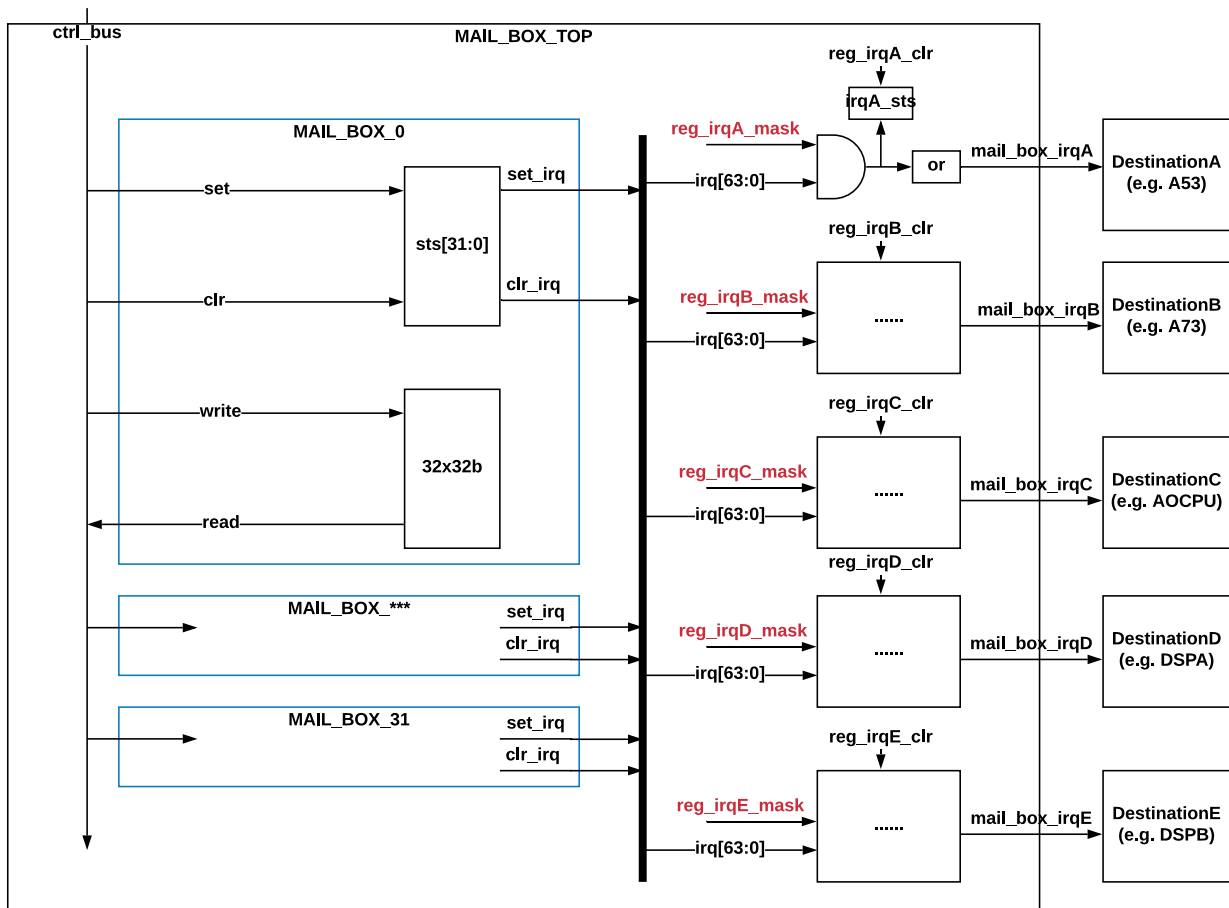
## 7.13 Mailbox

### 7.13.1 Overview

There are 32 mailboxes with the following features:

- Each mailbox includes set\_irq, clr\_irq and 32x32 bits buffer;
- Each mailbox can be configured the access permission;
- The total 64 irqs (32 set and 32 clr) can be configured the destination;
- All configurable register are TEE W/R only;

Figure 7-33 Mailbox Top Diagram



### 7.13.2 Register Description

#### Register Address

- MAILBOX\_BUF\_MBOX00 0xfe006000



- MAILBOX\_BUF\_MBOX01 0xfe006080
- MAILBOX\_BUF\_MBOX02 0xfe006100
- MAILBOX\_BUF\_MBOX03 0xfe006180
- MAILBOX\_BUF\_MBOX04 0xfe006200
- MAILBOX\_BUF\_MBOX05 0xfe006280
- MAILBOX\_BUF\_MBOX06 0xfe006300
- MAILBOX\_BUF\_MBOX07 0xfe006380
- MAILBOX\_BUF\_MBOX08 0xfe006400
- MAILBOX\_BUF\_MBOX09 0xfe006480
- MAILBOX\_BUF\_MBOX10 0xfe006500
- MAILBOX\_BUF\_MBOX11 0xfe006580
- MAILBOX\_BUF\_MBOX12 0xfe006600
- MAILBOX\_BUF\_MBOX13 0xfe006680
- MAILBOX\_BUF\_MBOX14 0xfe006700
- MAILBOX\_BUF\_MBOX15 0xfe006780
- MAILBOX\_BUF\_MBOX16 0xfe006800
- MAILBOX\_BUF\_MBOX17 0xfe006880
- MAILBOX\_BUF\_MBOX18 0xfe006900
- MAILBOX\_BUF\_MBOX19 0xfe006980
- MAILBOX\_BUF\_MBOX20 0xfe006a00
- MAILBOX\_BUF\_MBOX21 0xfe006a80
- MAILBOX\_BUF\_MBOX22 0xfe006b00
- MAILBOX\_BUF\_MBOX23 0xfe006b80
- MAILBOX\_BUF\_MBOX24 0xfe006c00
- MAILBOX\_BUF\_MBOX25 0xfe006c80
- MAILBOX\_BUF\_MBOX26 0xfe006d00
- MAILBOX\_BUF\_MBOX27 0xfe006d80
- MAILBOX\_BUF\_MBOX28 0xfe006e00
- MAILBOX\_BUF\_MBOX29 0xfe006e80
- MAILBOX\_BUF\_MBOX30 0xfe006f00
- MAILBOX\_BUF\_MBOX31 0xfe006f80
- MAILBOX\_LOCK\_BIT0 0xfe007000
- MAILBOX\_LOCK\_BIT1 0xfe007004
- MAILBOX\_LOCK\_BIT2 0xfe007008
- MAILBOX\_LOCK\_BIT3 0xfe00700c
- MAILBOX\_ACCESS\_EN\_IRQA\_CLR 0xfe007020
- MAILBOX\_ACCESS\_EN\_IRQB\_CLR 0xfe007024
- MAILBOX\_ACCESS\_EN\_IRQC\_CLR 0xfe007028
- MAILBOX\_ACCESS\_EN\_IRQD\_CLR 0xfe00702c
- MAILBOX\_ACCESS\_EN\_IRQE\_CLR 0xfe007030
- MAILBOX\_IRQ\_TYPE0 0xfe007034
- MAILBOX\_IRQ\_TYPE1 0xfe007038
- MAILBOX\_IRQA\_MASK0 0xfe007040
- MAILBOX\_IRQA\_MASK1 0xfe007044

- MAILBOX\_IRQB\_MASK0 0xfe007048
- MAILBOX\_IRQB\_MASK1 0xfe00704c
- MAILBOX\_IRQC\_MASK0 0xfe007050
- MAILBOX\_IRQC\_MASK1 0xfe007054
- MAILBOX\_IRQD\_MASK0 0xfe007058
- MAILBOX\_IRQD\_MASK1 0xfe00705c
- MAILBOX\_IRQE\_MASK0 0xfe007060
- MAILBOX\_IRQE\_MASK1 0xfe007064
- MAILBOX\_IRQA\_CLR0 0xfe007080
- MAILBOX\_IRQA\_CLR1 0xfe007084
- MAILBOX\_IRQB\_CLR0 0xfe007088
- MAILBOX\_IRQB\_CLR1 0xfe00708c
- MAILBOX\_IRQC\_CLR0 0xfe007090
- MAILBOX\_IRQC\_CLR1 0xfe007094
- MAILBOX\_IRQD\_CLR0 0xfe007098
- MAILBOX\_IRQD\_CLR1 0xfe00709c
- MAILBOX\_IRQE\_CLR0 0xfe0070a0
- MAILBOX\_IRQE\_CLR1 0xfe0070a4
- MAILBOX\_IRQA\_STS0 0xfe0070c0
- MAILBOX\_IRQA\_STS1 0xfe0070c4
- MAILBOX\_IRQB\_STS0 0xfe0070c8
- MAILBOX\_IRQB\_STS1 0xfe0070cc
- MAILBOX\_IRQC\_STS0 0xfe0070d0
- MAILBOX\_IRQC\_STS1 0xfe0070d4
- MAILBOX\_IRQD\_STS0 0xfe0070d8
- MAILBOX\_IRQD\_STS1 0xfe0070dc
- MAILBOX\_IRQE\_STS0 0xfe0070e0
- MAILBOX\_IRQE\_STS1 0xfe0070e4
- MAILBOX\_ACCESS\_EN\_MBOX00 0xfe007100
- MAILBOX\_ACCESS\_EN\_MBOX01 0xfe007104
- MAILBOX\_ACCESS\_EN\_MBOX02 0xfe007108
- MAILBOX\_ACCESS\_EN\_MBOX03 0xfe00710c
- MAILBOX\_ACCESS\_EN\_MBOX04 0xfe007110
- MAILBOX\_ACCESS\_EN\_MBOX05 0xfe007114
- MAILBOX\_ACCESS\_EN\_MBOX06 0xfe007118
- MAILBOX\_ACCESS\_EN\_MBOX07 0xfe00711c
- MAILBOX\_ACCESS\_EN\_MBOX08 0xfe007120
- MAILBOX\_ACCESS\_EN\_MBOX09 0xfe007124
- MAILBOX\_ACCESS\_EN\_MBOX10 0xfe007128
- MAILBOX\_ACCESS\_EN\_MBOX11 0xfe00712c
- MAILBOX\_ACCESS\_EN\_MBOX12 0xfe007130
- MAILBOX\_ACCESS\_EN\_MBOX13 0xfe007134
- MAILBOX\_ACCESS\_EN\_MBOX14 0xfe007138
- MAILBOX\_ACCESS\_EN\_MBOX15 0xfe00713c

- MAILBOX\_ACCESS\_EN\_MBOX16 0xfe007140
- MAILBOX\_ACCESS\_EN\_MBOX17 0xfe007144
- MAILBOX\_ACCESS\_EN\_MBOX18 0xfe007148
- MAILBOX\_ACCESS\_EN\_MBOX19 0xfe00714c
- MAILBOX\_ACCESS\_EN\_MBOX20 0xfe007150
- MAILBOX\_ACCESS\_EN\_MBOX21 0xfe007154
- MAILBOX\_ACCESS\_EN\_MBOX22 0xfe007158
- MAILBOX\_ACCESS\_EN\_MBOX23 0xfe00715c
- MAILBOX\_ACCESS\_EN\_MBOX24 0xfe007160
- MAILBOX\_ACCESS\_EN\_MBOX25 0xfe007164
- MAILBOX\_ACCESS\_EN\_MBOX26 0xfe007168
- MAILBOX\_ACCESS\_EN\_MBOX27 0xfe00716c
- MAILBOX\_ACCESS\_EN\_MBOX28 0xfe007170
- MAILBOX\_ACCESS\_EN\_MBOX29 0xfe007174
- MAILBOX\_ACCESS\_EN\_MBOX30 0xfe007178
- MAILBOX\_ACCESS\_EN\_MBOX31 0xfe00717c
- MAILBOX\_SET\_MBOX00 0xfe007180
- MAILBOX\_SET\_MBOX01 0xfe007184
- MAILBOX\_SET\_MBOX02 0xfe007188
- MAILBOX\_SET\_MBOX03 0xfe00718c
- MAILBOX\_SET\_MBOX04 0xfe007190
- MAILBOX\_SET\_MBOX05 0xfe007194
- MAILBOX\_SET\_MBOX06 0xfe007198
- MAILBOX\_SET\_MBOX07 0xfe00719c
- MAILBOX\_SET\_MBOX08 0xfe0071a0
- MAILBOX\_SET\_MBOX09 0xfe0071a4
- MAILBOX\_SET\_MBOX10 0xfe0071a8
- MAILBOX\_SET\_MBOX11 0xfe0071ac
- MAILBOX\_SET\_MBOX12 0xfe0071b0
- MAILBOX\_SET\_MBOX13 0xfe0071b4
- MAILBOX\_SET\_MBOX14 0xfe0071b8
- MAILBOX\_SET\_MBOX15 0xfe0071bc
- MAILBOX\_SET\_MBOX16 0xfe0071c0
- MAILBOX\_SET\_MBOX17 0xfe0071c4
- MAILBOX\_SET\_MBOX18 0xfe0071c8
- MAILBOX\_SET\_MBOX19 0xfe0071cc
- MAILBOX\_SET\_MBOX20 0xfe0071d0
- MAILBOX\_SET\_MBOX21 0xfe0071d4
- MAILBOX\_SET\_MBOX22 0xfe0071d8
- MAILBOX\_SET\_MBOX23 0xfe0071dc
- MAILBOX\_SET\_MBOX24 0xfe0071e0
- MAILBOX\_SET\_MBOX25 0xfe0071e4
- MAILBOX\_SET\_MBOX26 0xfe0071e8
- MAILBOX\_SET\_MBOX27 0xfe0071ec

- MAILBOX\_SET\_MBOX28 0xfe0071f0
- MAILBOX\_SET\_MBOX29 0xfe0071f4
- MAILBOX\_SET\_MBOX30 0xfe0071f8
- MAILBOX\_SET\_MBOX31 0xfe0071fc
- MAILBOX\_CLR\_MBOX00 0xfe007200
- MAILBOX\_CLR\_MBOX01 0xfe007204
- MAILBOX\_CLR\_MBOX02 0xfe007208
- MAILBOX\_CLR\_MBOX03 0xfe00720c
- MAILBOX\_CLR\_MBOX04 0xfe007210
- MAILBOX\_CLR\_MBOX05 0xfe007214
- MAILBOX\_CLR\_MBOX06 0xfe007218
- MAILBOX\_CLR\_MBOX07 0xfe00721c
- MAILBOX\_CLR\_MBOX08 0xfe007220
- MAILBOX\_CLR\_MBOX09 0xfe007224
- MAILBOX\_CLR\_MBOX10 0xfe007228
- MAILBOX\_CLR\_MBOX11 0xfe00722c
- MAILBOX\_CLR\_MBOX12 0xfe007230
- MAILBOX\_CLR\_MBOX13 0xfe007234
- MAILBOX\_CLR\_MBOX14 0xfe007238
- MAILBOX\_CLR\_MBOX15 0xfe00723c
- MAILBOX\_CLR\_MBOX16 0xfe007240
- MAILBOX\_CLR\_MBOX17 0xfe007244
- MAILBOX\_CLR\_MBOX18 0xfe007248
- MAILBOX\_CLR\_MBOX19 0xfe00724c
- MAILBOX\_CLR\_MBOX20 0xfe007250
- MAILBOX\_CLR\_MBOX21 0xfe007254
- MAILBOX\_CLR\_MBOX22 0xfe007258
- MAILBOX\_CLR\_MBOX23 0xfe00725c
- MAILBOX\_CLR\_MBOX24 0xfe007260
- MAILBOX\_CLR\_MBOX25 0xfe007264
- MAILBOX\_CLR\_MBOX26 0xfe007268
- MAILBOX\_CLR\_MBOX27 0xfe00726c
- MAILBOX\_CLR\_MBOX28 0xfe007270
- MAILBOX\_CLR\_MBOX29 0xfe007274
- MAILBOX\_CLR\_MBOX30 0xfe007278
- MAILBOX\_CLR\_MBOX31 0xfe00727c
- MAILBOX\_STS\_MBOX00 0xfe007280
- MAILBOX\_STS\_MBOX01 0xfe007284
- MAILBOX\_STS\_MBOX02 0xfe007288
- MAILBOX\_STS\_MBOX03 0xfe00728c
- MAILBOX\_STS\_MBOX04 0xfe007290
- MAILBOX\_STS\_MBOX05 0xfe007294
- MAILBOX\_STS\_MBOX06 0xfe007298
- MAILBOX\_STS\_MBOX07 0xfe00729c

- MAILBOX\_STS\_MBOX08 0xfe0072a0
- MAILBOX\_STS\_MBOX09 0xfe0072a4
- MAILBOX\_STS\_MBOX10 0xfe0072a8
- MAILBOX\_STS\_MBOX11 0xfe0072ac
- MAILBOX\_STS\_MBOX12 0xfe0072b0
- MAILBOX\_STS\_MBOX13 0xfe0072b4
- MAILBOX\_STS\_MBOX14 0xfe0072b8
- MAILBOX\_STS\_MBOX15 0xfe0072bc
- MAILBOX\_STS\_MBOX16 0xfe0072c0
- MAILBOX\_STS\_MBOX17 0xfe0072c4
- MAILBOX\_STS\_MBOX18 0xfe0072c8
- MAILBOX\_STS\_MBOX19 0xfe0072cc
- MAILBOX\_STS\_MBOX20 0xfe0072d0
- MAILBOX\_STS\_MBOX21 0xfe0072d4
- MAILBOX\_STS\_MBOX22 0xfe0072d8
- MAILBOX\_STS\_MBOX23 0xfe0072dc
- MAILBOX\_STS\_MBOX24 0xfe0072e0
- MAILBOX\_STS\_MBOX25 0xfe0072e4
- MAILBOX\_STS\_MBOX26 0xfe0072e8
- MAILBOX\_STS\_MBOX27 0xfe0072ec
- MAILBOX\_STS\_MBOX28 0xfe0072f0
- MAILBOX\_STS\_MBOX29 0xfe0072f4
- MAILBOX\_STS\_MBOX30 0xfe0072f8
- MAILBOX\_STS\_MBOX31 0xfe0072fc

### Register Description

MAILBOX\_BUF\_MBOX\*\*

Buffer base address for mailbox 0~31.

The following diagram shows the example of mbox0:

Figure 7-34 Mailbox Buffer

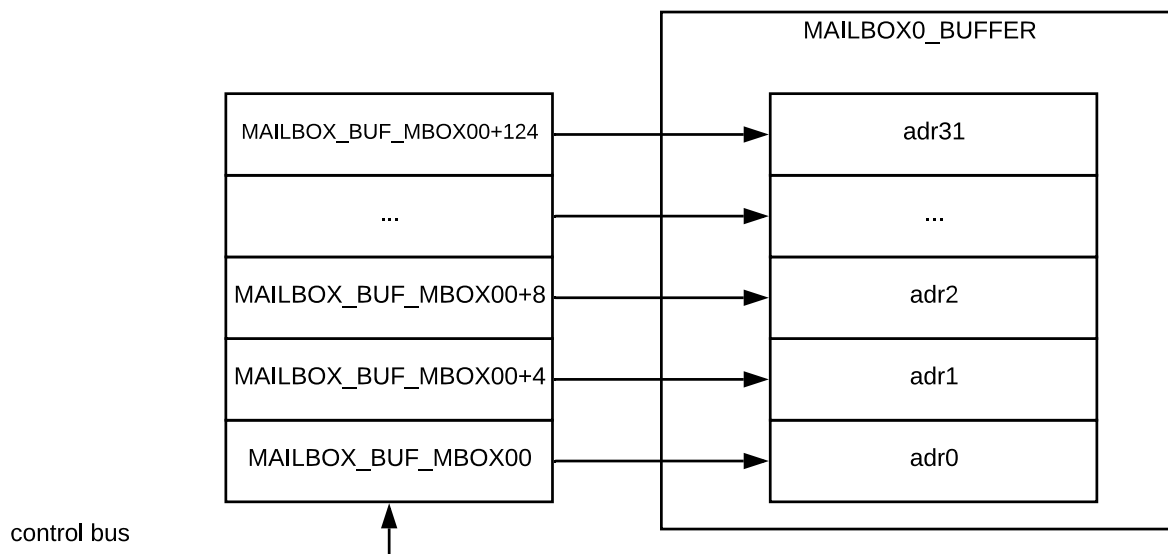


Table 7-752 MAILBOX\_SET\_MBOX\*\*\*

Bit(s)	R/W	Default	Description
31:0	W	0	write 32bits data to mailbox sts

Table 7-753 MAILBOX\_CLR\_MBOX\*\*\*

Bit(s)	R/W	Default	Description
31:0	W	0	will clear some bits of mailbox sts like new_sts = old_sts & ~write_data

Table 7-754 MAILBOX\_STS\_MBOX\*\*\*

Bit(s)	R/W	Default	Description
31:0	R	0	the sts of mailbox

Table 7-755 MAILBOX\_IRQ\_TYPE0

Bit(s)	R/W	Default	Description
31	W/R	0	mbox15 irq_clr type;
30	W/R	0	mbox15 irq_set type;
...	...	...	...
3	W/R	0	mbox01 irq_clr type;
2	W/R	0	mbox01 irq_set type;

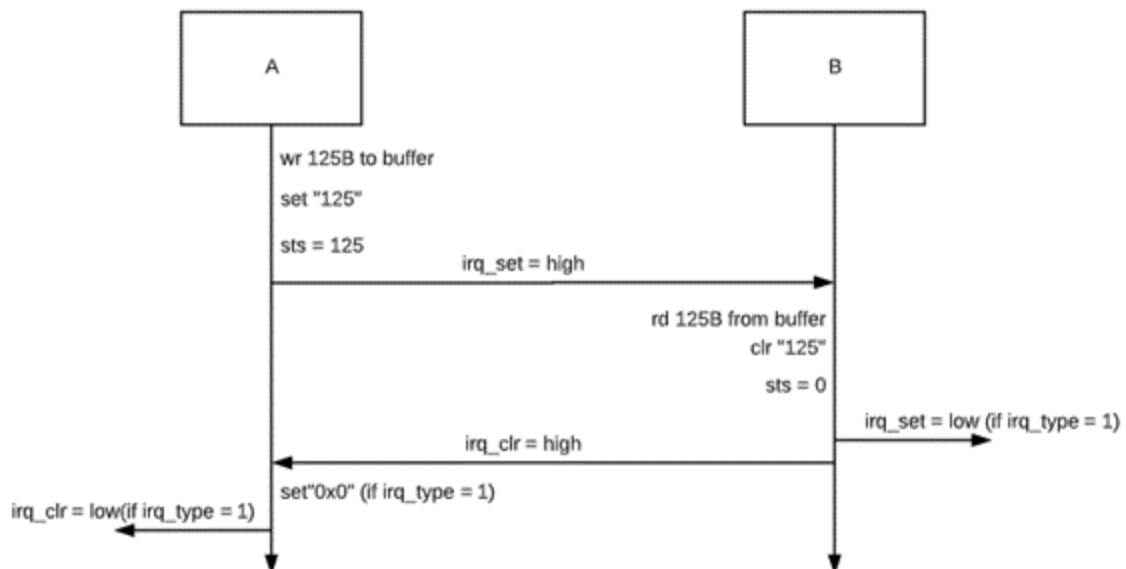
Bit(s)	R/W	Default	Description
1	W/R	0	mbox00 irq_clr type; 0: pulse irq; irq_clr generated by :sts changed & new sts == 0; 1: level irq; irq_clr pull high by: sts changed & new sts == 0; irq_clr pull low by : set 0x0;
0	W/R	0	mbox00 irq_set type; 0: pulse irq; irq_set generated by :sts changed & new sts != 0; 1: level irq; irq_set = (sts!= 0);

**Table 7-756 MAILBOX\_IRQ\_TYPE1**

Bit(s)	R/W	Default	Description
31	W/R	0	mbox31 irq_clr type;
30	W/R	0	mbox31 irq_set type;
...	...	...	...
3	W/R	0	mbox17 irq_clr type;
2	W/R	0	mbox17 irq_set type;
1	W/R	0	mbox16 irq_clr type;
0	W/R	0	mbox16 irq_set type;

Example flowchart of MAILBOX\_IRQ\_TYPE1:

**Figure 7-35 MAILBOX\_IRQ\_TYPE1 Flowchart**



**Table 7-757 MAILBOX\_IRQ\*\*\*\_MASK0**

Bit(s)	R/W	Default	Description
31	W/R	0	1: enable mbox15 irq_clr connect to IRQ***
30	W/R	0	1: enable mbox15 irq_set connect to IRQ***
...	...	...	...
3	W/R	0	1: enable mbox01 irq_clr connect to IRQ***
2	W/R	0	1: enable mbox01 irq_set connect to IRQ***
1	W/R	0	1: enable mbox00 irq_clr connect to IRQ***
0	W/R	0	1: enable mbox00 irq_set connect to IRQ***

**Table 7-758 MAILBOX\_IRQ\*\*\*\_MASK1**

Bit(s)	R/W	Default	Description
31	W/R	0	1: enable mbox31 irq_clr connect to IRQ***
30	W/R	0	1: enable mbox31 irq_set connect to IRQ***
...	...	...	...
3	W/R	0	1: enable mbox17 irq_clr connect to IRQ***
2	W/R	0	1: enable mbox17 irq_set connect to IRQ***
1	W/R	0	1: enable mbox16 irq_clr connect to IRQ***
0	W/R	0	1: enable mbox16 irq_set connect to IRQ***

**Table 7-759 MAILBOX\_IRQ\*\*\*\_CLR0**

Bit(s)	R/W	Default	Description
31	W	0	1: clear IRQ*** sts bit31(latched by mbox15 irq_clr)
30	W	0	1: clear IRQ*** sts bit30(latched by mbox15 irq_set)
...	...	...	...
3	W	0	1: clear IRQ*** sts bit3(latched by mbox01 irq_clr)
2	W	0	1: clear IRQ*** sts bit2(latched by mbox01 irq_set)
1	W	0	1: clear IRQ*** sts bit1(latched by mbox00 irq_clr)
0	W	0	1: clear IRQ*** sts bit0(latched by mbox00 irq_set)

**Table 7-760 MAILBOX\_IRQ\*\*\*\_CLR1**

Bit(s)	R/W	Default	Description
31	W	0	1: clear IRQ*** sts bit63(latched by mbox31 irq_clr)
30	W	0	1: clear IRQ*** sts bit62(latched by mbox31 irq_set)
...	...	...	...
3	W	0	1: clear IRQ*** sts bit35(latched by mbox17 irq_clr)



Bit(s)	R/W	Default	Description
2	W	0	1: clear IRQ*** sts bit34(latched by mbox17 irq_set)
1	W	0	1: clear IRQ*** sts bit33(latched by mbox16 irq_clr)
0	W	0	1: clear IRQ*** sts bit32(latched by mbox16 irq_set)

Table 7-761 MAILBOX\_IRQ\*\*\*\_STS0

Bit(s)	R/W	Default	Description
31	R	0	latched by mbox15 irq_clr
30	R	0	latched by mbox15 irq_set
...	...	...	...
3	R	0	latched by mbox01 irq_clr
2	R	0	latched by mbox01 irq_set
1	R	0	latched by mbox00 irq_clr
0	R	0	latched by mbox00 irq_set

Table 7-762 MAILBOX\_IRQ\*\*\*\_STS1

Bit(s)	R/W	Default	Description
31	R	0	latched by mbox31 irq_clr
30	R	0	latched by mbox31 irq_set
...	...	...	...
3	R	0	latched by mbox17 irq_clr
2	R	0	latched by mbox17 irq_set
1	R	0	latched by mbox16 irq_clr
0	R	0	latched by mbox16 irq_set

Table 7-763 MAILBOX\_ACCESS\_EN\_IRQ\*\*\*\_CLR

Bit(s)	R/W	Default	Description
7:0	W/R	0	access_en of CLR_IRQ_STS bit4 = 1: DSP can access bit3 = 1: AOCPUI can access bit2 = 1: JTAG(PROD) can access bit1 = 1: REE can access bit0 = 1: TEE can access

**Table 7-764 MAILBOX\_ACCESS\_EN\_MBOX\*\*\***

Bit(s)	R/W	Default	Description
31:24	W/R	0	access_en of RD command bit4 = 1: DSP can access bit3 = 1: AOCPU can access bit2 = 1: JTAG(PROD) can access bit1 = 1: REE can access bit0 = 1: TEE can access
23:16	W/R	0	access_en of WR command bit4 = 1: DSP can access bit3 = 1: AOCPU can access bit2 = 1: JTAG(PROD) can access bit1 = 1: REE can access bit0 = 1: TEE can access
15:8	W/R	0	access_en of CLR command bit4 = 1: DSP can access bit3 = 1: AOCPU can access bit2 = 1: JTAG(PROD) can access bit1 = 1: REE can access bit0 = 1: TEE can access
7:0	W/R	0	access_en of SET command bit4 = 1: DSP can access bit3 = 1: AOCPU can access bit2 = 1: JTAG(PROD) can access bit1 = 1: REE can access bit0 = 1: TEE can access

**Table 7-765 MAILBOX\_LOCK\_BIT0**

Bit(s)	R/W	Default	Description
31	W/R	0	write 1 once; 1: lock reg_access_en_mbox00
...	...	...	...
2	W/R	0	write 1 once; 1: lock reg_access_en_mbox02
1	W/R	0	write 1 once; 1: lock reg_access_en_mbox01
0	W/R	0	write 1 once; 1: lock reg_access_en_mbox00

**Table 7-766 MAILBOX\_LOCK\_BIT1**

Bit(s)	R/W	Default	Description
31	W/R	0	write 1 once; 1: lock mbox15 irq_clr destination (all reg_irq***_mask bit31)
30	W/R	0	write 1 once; 1: lock mbox15 irq_set destination (all reg_irq***_mask bit30)
...	...	...	...
1	W/R	0	write 1 once; 1: lock mbox00 irq_clr destination (all reg_irq***_mask bit1)
0	W/R	0	write 1 once; 1: lock mbox00 irq_set destination (all reg_irq***_mask bit0)

**Table 7-767 MAILBOX\_LOCK\_BIT2**

Bit(s)	R/W	Default	Description
31	W/R	0	write 1 once; 1: lock mbox31 irq_clr destination (all reg_irq***_mask bit63)
30	W/R	0	write 1 once; 1: lock mbox31 irq_set destination (all reg_irq***_mask bit62)
...	...	...	...
1	W/R	0	write 1 once; 1: lock mbox16 irq_clr destination (all reg_irq***_mask bit33)
0	W/R	0	write 1 once; 1: lock mbox16 irq_set destination (all reg_irq***_mask bit32)

**Table 7-768 MAILBOX\_LOCK\_BIT3**

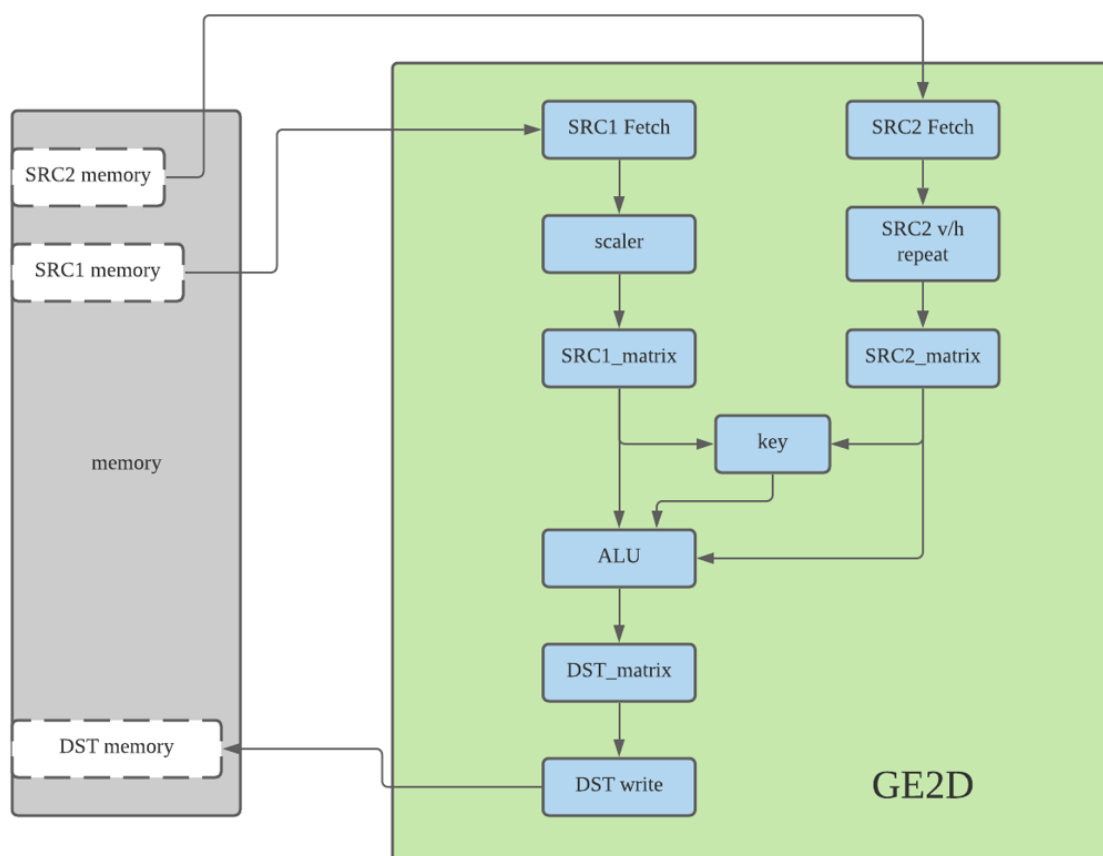
Bit(s)	R/W	Default	Description
31	W/R	0	write 1 once; 1: lock reg_irq_type
20	W/R	0	write 1 once; 1: lock reg_access_en_irqe_clr
19	W/R	0	write 1 once; 1: lock reg_access_en_irqd_clr
18	W/R	0	write 1 once; 1: lock reg_access_en_irqc_clr
17	W/R	0	write 1 once; 1: lock reg_access_en_irqb_clr
16	W/R	0	write 1 once; 1: lock reg_access_en_irqa_clr
15:0	W/R	0	Reserved

## 8 GE2D

### 8.1 Overview

The basic structure of GE2D is shown in figure below.

Figure 8-1 GE2D Structure



GE2D main features are shown in the following list.

#### GE2D SRC1

- Supports following input formats
  - 8bit component or 8bit lookup table -> 32bit
  - 16bit: RGB655, RGB844, RGBA6442, RGBA4444, RGB565, ARGB4444, ARGB1555, ARGB4642
  - 24bit: RGB888, RGBA5658, ARGB8565, RGBA6666, ARGB6666, BGR888
  - 32bit: RGBA8888, ARGB8888, ABGR888, BGRA8888
  - YUV, 8bit, 4:2:0, 4:2:2, 4:4:4: One Plane YUV/YUYV, 2 Planes (Y, UV) or 3 planes (Y, U, V)
- Supports X Rev, Y Rev read

#### GE2D SRC2

- Supports following input formats

- 8bit: One component, No lookup table
- 16bit: RGB655, RGB844, RGBA6442, RGBA4444, RGB565, ARGB4444, ARGB1555, ARGB4642
- 24bit: RGB888, RGBA5658, ARGB8565, RGBA6666, ARGB6666, BGR888
- 32bit: RGBA8888, ARGB8888, ABGR888, BGRA8888
- YUV, 8bit: One Plane 4:4:4 only
- Supports X Rev, Y Rev read

### GE2D Scaler

- Poly-phase vertical and horizontal scaler
- Anti-flicking filter

### GE2D SRC1 Matrix

- 3x3 programmable coefficients For YUV to RGB or RGB to YUV conversion

### GE2D SRC2 V/H Repeat

- Repeat the pixel 2, 4, or 8 times in vertical or horizontal

### GE2D SRC2 Matrix

- 3x3 programmable coefficients for YUV to RGB or RGB to YUV conversion

### GE2D Key

- 32bit SRC1 Key -> SRC1\_MASK
- 32bit SRC2 Key -> SRC2\_MASK

### GE2D ALU

- Independent Color, Alpha operation
- Color(RGB) Operation:
  - Blending mode selection
    - ◆  $Cs * CFs + Cd * CFd$
    - ◆  $Cs * CFs - Cd * CFd$
    - ◆  $Cd * CFd - Cs * CFs$
    - ◆  $\text{Min}(Cs * CFs, Cd * CFd)$
    - ◆  $\text{Max}(Cs * CFs, Cd * CFd)$
    - ◆ Logic OP
  - Independent CFs and CFd Selection: 0, 1, Cs, 1-Cs, 1-Cd, As, 1-As, Ad, 1-Ad, Cc, 1-Cc, Ac, 1-Ac, min(As, 1-Ad)
  - Logic OP if blending mode = Logic OP: 0, S, D, 1, ~S, ~D, S & ~D, S | ~D, S & D, S | D, ~(S&D), ~(S|D), S^D, ~(S^D), ~S & D, ~S|D
- Alpha Operation
  - Alpha blend mode
    - ◆  $As * AFs + Ad * AFd$
    - ◆  $As * AFs - Ad * AFd$
    - ◆  $Ad * AFd - As * AFs$
    - ◆  $\text{min}(As * AFs, Ad * AFd)$
    - ◆  $\text{max}(As * AFs, Ad * AFd)$
    - ◆ As op Ad
  - Independent CFs and CFd Selection: -0, 1, As, 1-As, Ad, 1-Ad, Ac, 1-Ac

- Logic OP if blending mode = Logic OP
  - ◆ 0, S, D, 1, ~S, ~D, S & ~D, S | ~D, S & D
  - ◆ S | D, ~(S&D), ~(S|D), S^D, ~(S^D), ~S & D, ~S|D
- Dst Matrix
  - 3x3 programmable coefficients for YUV to RGB or RGB to YUV conversion
  - support the signed mode output. Range: -128~127

### GE2D Dst Write

- output support repeat pixels mode, support 2 times, 4 times, 8times, 16times.
- 8bit component
- 16bit: RGB655, RGB844, RGBA6442, RGBA4444, RGB565, ARGB4444, ARGB1555, ARGB4642
- 24bit: RGB888, RGBA5658, ARGB8565, RGBA6666, ARGB6666, BGR888
- 32bit: RGBA8888, ARGB8888, ABGR888, BGRA8888
- Support YUV, 8bit: One Plane YUV(4:4:4), 2 Planes (Y, UV) NV21

### GE2D Secure Mode

- ge2d's secure mode bit send out in SRC1/SRC2/DST AXI ports AxUSER bit.
- ge2d's secure mode configured by TEE/REE. It's latched by each ge2d's processing start command.

## 8.2 Register Description

For below registers the base address is 0xd0160000.

Each register final address = BASE + address \* 4.

**Table 8-1 GE2D\_GEN\_CTRL0 0x8A0**

Bit(s)	R/W	Default	Description
31	R/W	0	dst_bytemask_only: Applicable only if dst_bitmask_en=1.
30	R/W	0	dst_bitmask_en: destination bitmask enable. 0: disable; 1: enable.
29	R/W	0	src2_key_en: source2 key enable. 0: disable; 1: enable.
28	R/W	0	src2_key_mode: source2 key mode. 0: mask data when match; 1: mask data when unmatched.
27	R/W	0	src1_key_en: source1 key enable. 0: disable; 1: enable.
26	R/W	0	src1_key_mode: source1 key mode. 0: mask data when match; 1: mask data when unmatched.
25-24	R/W	0	dst1_8b_mode_sel: Destination 1's 8-bit mode component selection. 0: Select Y or R; 1: Select Cb or G; 2: Select Cr or B; 3: Select Alpha.
23	R/W	0	dst_clip_mode: 0: Write inside clip window;

Bit(s)	R/W	Default	Description
			1: Write outside clip window.
22-17	R	0	dst_rpt[5:0], 1~0bit: 0: no rpt, 1:x0, 2:0x, 11:xx, 3~2bit: 0, repeat64, 1: 128, 2:256)
16-15	R/W	0	src2_8b_mode_sel: Applicable only when src2_format=0, define the property of the 8-bit output data. 0: The 8-bit output data is for Y or R; 1: The 8-bit output data is for Cb or G; 2: The 8-bit output data is for Cr or B; 3: The 8-bit output data is for Alpha.
14	R/W	0	src2_fill_mode: When the display window is outside the boundary of the clipping window, this field defines how to fill the area outside the clipping window. 0: Fill with the pixels at the boundary of the clipping window; 1: Fill with the default color defined by register GE2D_SRC2_DEF_COLOR.
13-12	R/W	0	src2_pic_struct: Define how source2 reads the picture stored in DDR memory. 0: Read all lines; 1: Reserved; 2: Read even lines only; 3: Read odd lines only.
11	R/W	0	src2_X_yc_ratio: Source2 x direction yc ratio. 0: 1:1; 1: 2:1.
9-7	R	0	Unused
6-5	R/W	0	src1_8b_mode_sel: Applicable only when src1_sep_en=0, src1_format=0 and src1_lut_en=0, define the property of the 8-bit output data. 0: the 8-bit output data is for Y or R; 1: the 8-bit output data is for Cb or G; 2: the 8-bit output data is for Cr or B; 3: the 8-bit output data is for Alpha.
4	R/W	0	src1_fill_mode: When the display window is outside the boundary of the clipping window, this field defines how to fill the area outside the clipping window. 0: Fill with the pixels at the boundary of the clipping window; 1: Fill with the default color defined by register GE2D_SRC1_DEF_COLOR.
3	R/W	0	src1_lut_en: Applicable only when src1_sep_en=0 and src1_format=0, define whether to enable 8-bit input data to look up a 32-bit pixel for output. 0: Disable; 1: Enable.
2-1	R/W	0	src1_pic_struct: Define how source1 reads the picture stored in DDR memory. 0: Read all lines; 1: Reserved; 2: Read even lines only; 3: Read odd lines only.

Table 8-2 GE2D\_GEN\_CTRL1 0x8A1

Bit(s)	R/W	Default	Description
31	R/W	0	soft_rst: If true, reset GE2D.
30	R/W	0	dst write response counter reset
29	R/W	0	disable adding dst write response count to busy bit
28-27	R	0	Unused
26	R/W	0	Color_conversion_mode[1] in alu. Mode[1:0]

Bit(s)	R/W	Default	Description
			3 : color_out = color; 2 : color_out = (color != 255) ? color : color + 1; 1 : color_out = (color < 128) ? color : color + 1; 0 : color_out = (color == 0) ? color : color + 1.
25-24	R/W	0	interrupt_ctrl: If bit[0] true, generate interrupt when one command done; if bit[1] true, generate interrupt when ge2d change from busy to not busy.
23-22	R/W	0x3	src2_burst_size_ctrl: Source2 DDR request burst size control. (Note: data to source2 are stored together in one DDR memory block.) 0: Burst size = 24 x 64-bit; 1: Burst size = 32 x 64-bit; 2: Burst size = 48 x 64-bit; 3: Burst size = 64 x 64-bit.
21-16	R/W	0x3F	src1_burst_size_ctrl: Source1 DDR request burst size control. Bit[21:20] control Y burst size, bit[19:18] control Cb burst size, bit[17:16] control Cr burst size. Each 2-bit is decoded as below: 0: Burst size = 24 x 64-bit; 1: Burst size = 32 x 64-bit; 2: Burst size = 48 x 64-bit; 3: Burst size = 64 x 64-bit.
15-14	R/W	0	dst1_pic_struct: Define how destination 1 write the picture to DDR memory. 0: Write all lines (whole frame); 1: Reserved; 2: Write even lines only (top); 3: Write odd lines only (bottom).
13-12	R/W	0	src_rd_ctrl: Bit[13] if true, force read src1, bit[12] if true, force read src2.
11	R/W	0	dst2_urgent_en: Destination 2 DDR request urgent enable.
10	R/W	0	src1_urgent_en: Source1 DDR request urgent enable.
9	R/W	0	src2_urgent_en: Source2 DDR request urgent enable.
8	R/W	0	dst1_urgent_en: Destination 1 DDR request urgent enable.
7-0	R/W	0	src1_gb_alpha: Source1 global alpha.

Table 8-3 GE2D\_GEN\_CTRL2 0x8A2

Bit(s)	R/W	Default	Description
31	R/W	0	alpha_conversion_mode[0] in alu. Mode[1:0]: 2,3 : alpha_out = (alpha!=255) ? alpha : alpha + 1; 1 : alpha_out = (alpha < 128) ? alpha : alpha + 1; 0 : alpha_out = (alpha == 0) ? alpha : alpha + 1.
30	R/W	0	Color_conversion_mode[0] in alu. Mode[1:0] 3 : color_out = color; 2 : color_out = (color != 255) ? color : color + 1; 1 : color_out = (color < 128) ? color : color + 1; 0 : color_out = (color == 0) ? color : color + 1.
29	R/W	0	src1_gb_alpha_en in alu. As = src1_gb_alpha_en ? Asr * Ag: Asr
28	R/W	0	dst1_COLOR_round_MODE. 1 = Truncate the full bit color components to required output bit width; 0 = Round (+ 0.5) the full bit color components to required output bit width.
27	R/W	0	src2_COLOR_EXPAND_MODE.



Bit(s)	R/W	Default	Description
			1 = Expand the color components to 8-bit by padding LSBs with MSBs. E.g. If the input is 5'b11000, the output is expanded to 8'b11000110; 0 = Expand the color components to 8-bit by padding LSBs with 0.
26	R/W	0	src2_ALPHA_EXPAND_MODE. 1 = Expand alpha value to 8-bit by padding LSBs with MSBs. E.g. If the input is 5'b11000, the output is expanded to 8'b11000110; 0 = If input alpha value is all 1, then expand the value to 8-bit by padding LSBs with 1; otherwise, pad LSBs with 0.
25	R/W	0	src1_COLOR_EXPAND_MODE. 1 = Expand the color components to 8-bit by padding LSBs with MSBs. E.g. If the input is 5'b11000, the output is expanded to 8'b11000110; 0 = Expand the color components to 8-bit by padding LSBs with 0.
24	R/W	0	src1_ALPHA_EXPAND_MODE. 1 = Expand alpha value to 8-bit by padding LSBs with MSBs. E.g. If the input is 5'b11000, the output is expanded to 8'b11000110; 0 = If input alpha value is all 1, then expand the value to 8-bit by padding LSBs with 1; otherwise, pad LSBs with 0.
23	R/W	0	dst_little_endian: define the endianness of SRC2 input data. 1 = Little endian; 0 = Big endian.
22-19	R/W	0	dst1_color_map: Applicable to 16-bit, 24-bit and 32-bit pixel, defines the bit-field allocation of the pixel data. For whether to truncate or round full 8-bit to output, refer to ge2d_gen_ctrl2.dst1_color_round_mode. For 16-bit mode (dst1_format=1): 0 = Unused; 1 = 6:5:5 format. Bit[15:10] is Y [7:2] or R[7:2], bit[9:5] is Cb[7:3] or G[7:3], bit[4:0] is Cr[7:3] or B[7:3]; 2 = 8:4:4 format. Bit[15:8] is Y or R, bit[7:4] is Cb[7:4] or G[7:4], bit[3:0] is Cr[7:4] or B[7:4]; 3 = 6:4:4:2 format. Bit[15:10] is Y[7:2] or R[7:2], bit[9:6] is Cb[7:4] or G[7:4], bit [5:2] is Cr[7:4] or B[7:4], bit[1:0] is Alpha[7:6]; 4 = 4:4:4:4 format. Bit[15:12] is Y [7:4] or R[7:4], bit[11:8] is Cb[7:4] or G[7:4], bit[7:4] is Cr[7:4] or B[7:4], bit[3:0] is Alpha[7:4]; 5 = 5:6:5 format. Bit[15:11] is Y[7:3] or R[7:3], bit[10:5] is Cb[7:2] or G [7:2], bit[4:0] is Cr[7:3] or B[7:3]; 6 = 4:4:4:4 format. Bit[15:12] is Alpha[7:4], bit [11:8] is Y[7:4] or R[7:4], bit[7:4] is Cb[7:4] or G[7:4], bit[3:0] is Cr[7:4] or B[7:4]; 7 = 1:5:5:5 format. Bit[15] is Alpha[7], bit[14:10] is Y[7:3] or R[7:3], bit[9:5] is Cb [7:3] or G[7:3], bit[4:0] is Cr[7:3] or B[7:3]; 8 = 4:6:4:2 format. Bit[15:12] is Y[7:4] or R[7:4], bit[11:6] is Cb[7:2] or G[7:2], bit[5:2] is Cr[7:4] or B[7:4], bit[1:0] is Alpha [1:0]. 9 = CbCr format. Bit[15:8] is Cb, bit[7:0] is Cr; 10 = CrCb format. Bit[15:8] is Cr, bit[7:0] is Cb. For 24-bit mode (dst1_format=2): 0 = RGB 8:8:8 mode. Bit [23:16] is Y or R, bit[15:8] is Cb or G, bit[7:0] is Cr or B; 1 = RGBA 5:6:5:8 mode. Bit[23:19] is Y[7:3] or R[7:3], bit[18:13] is Cb[7:2] or G[7:2], bit[12:8] is Cr[7:3] or B[7:3], bit[7:0] is Alpha; 2 = ARGB 8:5:6:5 mode. Bit[23:16] is Alpha, bit[15:11] is Y[7:3] or R[7:3], bit[10:5] is Cb[7:2] or G[7:2], bit[4:0] is Cr[7:3] or B[7:3]; 3 = RGBA 6:6:6:6 mode. Bit[23:18] is Y[7:2] or R[7:2], bit[17:12] is Cb[7:2] or G[7:2], bit[11:6] is Cr[7:2] or B[7:2], bit[5:0] is Alpha[7:2]; 4 = ARGB 6:6:6:6 mode. Bit [23:18] is Alpha[7:2];, bit[17:12] is Y[7:2] or R[7:2], bit[11:6] is Cb[7:2] or G[7:2], bit[5:0] is Cr[7:2] or B[7:2]; 5 = BGR 8:8:8 mode. Bit[23:16] is Cr or B, bit[15:8] is Cb or G, bit[7:0] is Y or R. For 32-bit mode (dst1_format=3): 0 = RGBA 8:8:8:8 format. Bit[31:24] is Y or R, bit[23:16] is Cb or G; bit[15:8] is Cr or B; bit[7:0] is Alpha; 1 = ARGB 8:8:8:8 format. Bit[31:24] is Alpha, bit[23:16] is Y or R; bit[15:8] is Cb or G; bit[7:0] is Cr or B; 2 = ABGR 8:8:8:8 format. Bit[31:24] is Alpha, bit [23:16] is Cr or B; bit[15:8] is Cb or G; bit[7:0] is Y or R; 3 = BGRA 8:8:8:8 format. Bit[31:24] is Cr or B, bit[23:16] is Cb or G; bit[15:8] is Y or R; bit[7:0] is Alpha.
18	R/W	0	ALU_MULT_MODE: 1: mult result rounding else truncation
17-16	R/W	0	dst1_format: define output pixel byte-width. 0: Output pixel is 1-byte (8-bit) color component; 1: Output pixel is 2-byte (16-bit), refer to GE2D_GEN_CTRL2.dst1_COLOR_map for further pixel color mapping; 2: Output pixel is 3-byte (24-bit), refer to GE2D_GEN_CTRL2.dst1_COLOR_MAP for further pixel color mapping; 3: Output pixel is 4-byte (32-bit), refer to GE2D_GEN_CTRL2.dst1_COLOR_MAP for further pixel color mapping.

Bit(s)	R/W	Default	Description
15	R/W	0	src2_little_endian: define the endianness of SRC2 input data. 1 = Little endian; 0 = Big endian.
14-11	R/W	0	src2_color_map: Applicable to 16-bit, 24-bit and 32-bit pixel, defines the bit-field allocation of the pixel data. For expanding the bit-fields to full 8-bit, refer to ge2d_gen_ctrl2.src2_color_expand_mode and ge2d_gen_ctrl2.src2_alpha_expand_mode. For 16-bit mode (src2_format=1): 0 = Unused; 1 = 6:5:5 format. Bit[15:10] is Y[7:2] or R[7:2], bit[9:5] is Cb[7:3] or G[7:3], bit[4:0] is Cr[7:3] or B[7:3]; 2 = 8:4:4 format. Bit[15:8] is Y or R, bit[7:4] is Cb[7:4] or G[7:4], bit[3:0] is Cr[7:4] or B[7:4]; 3 = 6:4:4:2 format. Bit[15:10] is Y[7:2] or R[7:2], bit[9:6] is Cb[7:4] or G[7:4], bit[5:2] is Cr[7:4] or B[7:4], bit[1:0] is Alpha[7:6]; 4 = 4:4:4:4 format. Bit[15:12] is Y[7:4] or R[7:4], bit[11:8] is Cb[7:4] or G[7:4], bit[7:4] is Cr[7:4] or B[7:4], bit[3:0] is Alpha[7:4]; 5 = 5:6:5 format. Bit[15:11] is Y[7:3] or R[7:3], bit[10:5] is Cb[7:2] or G[7:2], bit[4:0] is Cr[7:3] or B[7:3]; 6 = 4:4:4:4 format. Bit[15:12] is Alpha[7:4], bit[11:8] is Y[7:4] or R[7:4], bit[7:4] is Cb[7:4] or G[7:4], bit[3:0] is Cr[7:4] or B[7:4]; 7 = 1:5:5:5 format. Bit[15] is Alpha[7], bit[14:10] is Y[7:3] or R[7:3], bit[9:5] is Cb[7:3] or G[7:3], bit[4:0] is Cr[7:3] or B[7:3]; 8 = 4:6:4:2 format. Bit[15:12] is Y[7:4] or R[7:4], bit[11:6] is Cb[7:2] or G[7:2], bit[5:2] is Cr[7:4] or B[7:4], bit[1:0] is Alpha[1:0]. For 24-bit mode (src2_format=2): 0 = RGB 8:8:8 mode. Bit[23:16] is Y or R, bit[15:8] is Cb or G, bit[7:0] is Cr or B; 1 = RGBA 5:6:5:8 mode. Bit[23:19] is Y[7:3] or R[7:3], bit[18:13] is Cb[7:2] or G[7:2], bit[12:8] is Cr[7:3] or B[7:3], bit[7:0] is Alpha; 2 = ARGB 8:5:6:5 mode. Bit[23:16] is Alpha, bit[15:11] is Y[7:3] or R[7:3], bit[10:5] is Cb[7:2] or G[7:2], bit[4:0] is Cr[7:3] or B[7:3]; 3 = RGBA 6:6:6:6 mode. Bit[23:18] is Y[7:2] or R[7:2], bit[17:12] is Cb[7:2] or G[7:2], bit[11:6] is Cr[7:2] or B[7:2], bit[5:0] is Alpha[7:2]; 4 = ARGB 6:6:6:6 mode. Bit[23:18] is Alpha[7:2]; bit[17:12] is Y[7:2] or R[7:2], bit[11:6] is Cb[7:2] or G[7:2], bit[5:0] is Cr[7:2] or B[7:2]; 5 = BGR 8:8:8 mode. Bit[23:16] is Cr or B, bit[15:8] is Cb or G, bit[7:0] is Y or R. For 32-bit mode (src2_format=3): 0 = RGBA 8:8:8:8 format. Bit[31:24] is Y or R, bit[23:16] is Cb or G; bit[15:8] is Cr or B; bit[7:0] is Alpha; 1 = ARGB 8:8:8:8 format. Bit[31:24] is Alpha, bit[23:16] is Y or R; bit[15:8] is Cb or G; bit[7:0] is Cr or B; 2 = ABGR 8:8:8:8 format. Bit[31:24] is Alpha, bit[23:16] is Cr or B; bit[15:8] is Cb or G; bit[7:0] is Y or R; 3 = BGRA 8:8:8:8 format. Bit[31:24] is Cr or B, bit[23:16] is Cb or G; bit[15:8] is Y or R; bit[7:0] is Alpha.
10	R/W	0	alpha_conversion_mode[1] in alu. Mode[1:0]: 2,3 : alpha_out = (alpha!=255) ? alpha : alpha + 1; 1 : alpha_out = (alpha < 128) ? alpha : alpha + 1; 0 : alpha_out = (alpha == 0) ? alpha : alpha + 1.
9-8	R/W	0	src2_format: define input pixel byte-width. 0: Input pixel is 1-byte (8-bit) color component; 1: Input pixel is 2-byte (16-bit), refer to GE2D_GEN_CTRL2.SRC2_COLOR_MAP for further pixel color mapping; 2: Input pixel is 3-byte (24-bit), refer to GE2D_GEN_CTRL2.SRC2_COLOR_MAP for further pixel color mapping; 3: Input pixel is 4-byte (32-bit), refer to GE2D_GEN_CTRL2.SRC2_COLOR_MAP for further pixel color mapping.
7	R/W	0	src1_little_endian: define the endianness of SRC1 input data. 1 = Little endian; 0 = Big endian.
6-3	R/W	0	src1_color_map: Note: If SRC1_DEEPCOLOR=0, the SRC1_COLOR_MAP's definitions is as below. If SRC1_DEEPCOLOR=1, please refer to SRC1_DEEPCOLOR entry for new meaning. Applicable to 16-bit, 24-bit and 32-bit pixel, defines the bit-field allocation of the pixel data. For expanding the bit-fields to full 8-bit, refer to ge2d_gen_ctrl2.src1_color_expand_mode and ge2d_gen_ctrl2.src1_alpha_expand_mode. For 16-bit mode (src1_format=1): 0 = 4:2:2 format (Y0Cb0Y1Cr0); 1 = 6:5:5 format. Bit[15:10] is Y[7:2] or R[7:2], bit[9:5] is Cb[7:3] or G[7:3], bit[4:0] is Cr[7:3] or B[7:3]; 2 = 8:4:4 format. Bit[15:8] is Y or R, bit[7:4] is Cb[7:4] or G[7:4], bit[3:0] is Cr[7:4] or B[7:4]; 3 = 6:4:4:2 format. Bit[15:10] is Y[7:2] or R[7:2], bit[9:6] is Cb[7:4] or G[7:4], bit[5:2] is Cr[7:4] or B[7:4], bit[1:0] is Alpha[7:6]; 4 = 4:4:4:4 format. Bit

Bit(s)	R/W	Default	Description
			[15:12] is Y[7:4] or R[7:4], bit[11:8] is Cb[7:4] or G[7:4], bit[7:4] is Cr[7:4] or B [7:4], bit[3:0] is Alpha[7:4]; 5 = 5:6:5 format. Bit[15:11] is Y[7:3] or R[7:3], bit[10:5] is Cb[7:2] or G[7:2], bit[4:0] is Cr[7:3] or B[7:3]; 6 = 4:4:4:4 format. Bit[15:12] is Alpha[7:4], bit[11:8] is Y[7:4] or R[7:4], bit[7:4] is Cb[7:4] or G[7:4], bit[3:0] is Cr[7:4] or B[7:4]; 7 = 1:5:5:5 format. Bit[15] is Alpha[7], bit[14:10] is Y[7:3] or R[7:3], bit [9:5] is Cb[7:3] or G[7:3], bit[4:0] is Cr[7:3] or B[7:3]; 8 = 4:6:4:2 format. Bit[15:12] is Y[7:4] or R[7:4], bit[11:6] is Cb[7:2] or G[7:2], bit[5:2] is Cr[7:4] or B[7:4], bit[1:0] is Alpha[1:0]. For 24-bit mode (src1_format=2): 0 = RGB 8:8:8 mode. Bit[23:16] is Y or R, bit[15:8] is Cb or G, bit[7:0] is Cr or B; 1 = RGBA 5:6:5:8 mode. Bit [23:19] is Y[7:3] or R[7:3], bit[18:13] is Cb[7:2] or G[7:2], bit[12:8] is Cr[7:3] or B [7:3], bit[7:0] is Alpha; 2 = ARGB 8:5:6:5 mode. Bit[23:16] is Alpha, bit[15:11] is Y [7:3] or R[7:3], bit[10:5] is Cb[7:2] or G[7:2], bit[4:0] is Cr[7:3] or B[7:3]; 3 = RGBA 6:6:6:6 mode. Bit[23:18] is Y[7:2] or R[7:2], bit[17:12] is Cb[7:2] or G[7:2], bit [11:6] is Cr[7:2] or B[7:2], bit[5:0] is Alpha[7:2]; 4 = ARGB 6:6:6:6 mode. Bit [23:18] is Alpha[7:2];, bit[17:12] is Y[7:2] or R[7:2], bit[11:6] is Cb[7:2] or G[7:2], bit[5:0] is Cr[7:2] or B[7:2]; 5 = BGR 8:8:8 mode. Bit[23:16] is Cr or B, bit[15:8] is Cb or G, bit[7:0] is Y or R. 14=NV12 format. 8-bit Y and 16-bit CbCr; 15=NV21 format. 8-bit Y and 16-bit CrCb; For 32-bit mode (src1_format=3): 0 = RGBA 8:8:8:8 format. Bit[31:24] is Y or R, bit[23:16] is Cb or G; bit[15:8] is Cr or B; bit [7:0] is Alpha; 1 = ARGB 8:8:8:8 format. Bit[31:24] is Alpha, bit[23:16] is Y or R; bit[15:8] is Cb or G; bit[7:0] is Cr or B; 2 = ABGR 8:8:8:8 format. Bit[31:24] is Al-pha, bit[23:16] is Cr or B; bit[15:8] is Cb or G; bit[7:0] is Y or R; 3 = BGRA 8:8:8:8 format. Bit[31:24] is Cr or B, bit[23:16] is Cb or G; bit[15:8] is Y or R; bit[7:0] is Alpha.
2	R/W	0	src1_DEEPCOLOR: 1 = Enable deepcolor formats support, the formats are defined by SRC1_FOR- MAT and SRC1_COLOR_MAP; 0 = Disable deepcolor. The supported deepcolor formats are as below: src1_deep_color=1, src1_for- mat=2'b01, src1_color_map=4'b0000 or 4'b0001: 10-bit 422 in one canvas – 10- bit Y + 10-bit C = 20-bit per pixel in canvas. If src1_color_map=4'b0000, the se- quence is Y0Cb0, Y1Cr0, ... If src1_color_map=4'b0001, the sequence is Y0Cr0, Y1Cb0, ... src1_deep_color=1, src1_format=2'b01, src1_color_map=4'b1000 or 4'b1001: 12-bit 422 in one canvas – 12-bit Y + 12-bit C = 24-bit per pixel in can- vas. If src1_color_map=4'b1000, the sequence is Y0Cb0, Y1Cr0, ... If src1_col- or_map=4'b1001, the sequence is Y0Cr0, Y1Cb0, ... src1_deep_color=1, src1_ format=2'b10: 10-bit 444 in one canvas – 10-bit Y + 10-bit Cb + 10-bit Cr + 2-bit stuffing = 32-bit per pixel in canvas. If src1_color_map=4'b0000, the sequence is Y0Cb0Cr0, Y1Cb1Cr1, ... If src1_color_map=other value, the sequence is Cr0Cb0Y0, Cr1Cb1Y1, ...
1-0	R/W	0	src1_format: define input pixel byte-width. Note: If SRC1_DEEPCOLOR=0, the SRC1_FORMAT's definitions is as below. If SRC1_DEEPCOLOR=1, please refer to SRC1_DEEPCOLOR entry for new meaning. 0: Input pixel is 1-byte (8-bit), it is either an 8-bit color component or 8-bit address to look up a 32-bit pixel, refer to GE2D_GEN_CTRL0.src1_lut_en; 1: Input pixel is 2-byte (16-bit), refer to GE2D_GEN_CTRL2.SRC1_COLOR_MAP for further pixel color mapping; 2: Input pixel is 3-byte (24-bit), refer to GE2D_GEN_CTRL2. SRC1_COLOR_MAP for further pixel color mapping; 3: Input pixel is 4-byte (32- bit), refer to GE2D_GEN_CTRL2.SRC1_COLOR_MAP for further pixel color mapping.

**Table 8-4 GE2D\_CMD\_CTRL 0x8A3**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused
28	R/W	0	sec_mode: if true, ge2d works with secure domain
15-14	R/W	0	src2_x_interp_ctrl: if true, do interpolation mode
13-12	R/W	0	SRC2 X repeat: 0 will not repeat, 1 for 1:2, 2 for 1:4, 3 for 1:8

Bit(s)	R/W	Default	Description
11-10	R/W	0	SRC2 y repeat: 0 will not repeat, 1 for 1:2, 2 for 1:4, 3 for 1:8
9	R/W	0	src2_fill_color_en: if true, all src2 data use default color.
8	R/W	0	src1_fill_color_en: if true, all src1 data use default color.
7	R/W	0	dst_xy_swap: if true, dst x/y swap.
6	R/W	0	dst_x_rev: if true, dst x direction reversely read.
5	R/W	0	dst_y_rev: if true, dst y direction reversely read.
4	R/W	0	src2_x_rev: if true, src2 x direction reversely read.
3	R/W	0	src2_y_rev: if true, src2 y direction reversely read.
2	R/W	0	src1_x_rev: if true, src1 x direction reversely read.
1	R/W	0	src1_y_rev: if true, src1 y direction reversely read.
0	R/W	0	cbus_cmd_wr: If true, generate a pulse to validate a GE2D command, the command is described by the rest of the field of this register.

Table 8-5 GE2D\_STATUS0 0x8A4

Bit(s)	R/W	Default	Description
31-29	R	0	unused
28-17	R	0	dst write response counter, for debug only.
16-7	R	0	dp_status: ge2d_dp status, for debug only.
6	R	0	r1cmd_rdy: read src2 cmd ready.
5	R	0	r2cmd_rdy: read src2 cmd ready.
4	R	0	pdpcmd_v: pre dpcmd ready.
3	R	0	dpcmd_rdy: GE2D dpcmd ready.
2	R	0	buf_cmd_v: GE2D buffer command valid.
1	R	0	curr_cmd_v: GE2D current command valid.
0	R	0	ge2d_busy: GE2D busy.

Table 8-6 GE2D\_STATUS1 0x8A5

Bit(s)	R/W	Default	Description
31-30	R	0	unused
29-16	R	0	ge2d_dst1_status, for debug only.
15	R	1	ge2d_rd_src2 core.fifo_empty.
14	R	0	ge2d_rd_src2 core.fifo_overflow.
13-12	R	0	ge2d_rd_src2 core.req_st. Same as req_st_y.
11	R	0	ge2d_rd_src2 cmd_if.cmd_err, true if cmd_format=1.
10	R	0	ge2d_rd_src2 cmd_if.cmd_st, 0=IDLE state, 1=BUSY state.

Bit(s)	R/W	Default	Description
9	R	1	ge2d_rd_src1 luma_core(chroma_core).fifo_empty.
8	R	0	ge2d_rd_src1 luma_core(chroma_core).fifo_overflow.
7-6	R	0	ge2d_rd_src1 chroma_core.req_st_cr. Same as req_st_y.
5-4	R	0	ge2d_rd_src1 chroma_core.req_st_cb. Same as req_st_y.
3-2	R	0	ge2d_rd_src1 luma_core.req_st_y. 0: IDLE; 1: WAIT_FIFO_ROOM; 2: REQUEST; 3: WAIT_FINISH.
1	R	0	ge2d_rd_src1 cmd_if.stat_read_window_err, 1=reading/clipping window setting exceed limit.
0	R	0	ge2d_rd_src1 cmd_if.cmd_st, 0=IDLE state, 1=BUSY state.

Table 8-7 GE2D\_SRC1\_DEF\_COLOR 0x8A6

Bit(s)	R/W	Default	Description
31-24	R/W	0	Default Y or R.
23-16	R/W	0x80	Default Cb or G.
15-8	R/W	0x80	Default Cr or B.
7-0	R/W	0	Default Alpha.

Table 8-8 GE2D\_SRC1\_CLIPX\_START\_END 0x8A7

Bit(s)	R/W	Default	Description
31	R/W	0	SRC1 clip x start extra, if true, one more data is read for chroma.
30-29	R	0	Unused.
28-16	R/W	0	SRC1 clip x start.
15	R/W	0	SRC1 clip x end extra, if true, one more data is read for chroma.
14-13	R	0	Unused.
12-0	R/W	0x1FFF	SRC1 clip x end.

Table 8-9 GE2D\_SRC1\_CLIPY\_START\_END 0x8A8

Bit(s)	R/W	Default	Description
30-29	R	0	Unused.
28-16	R/W	0	SRC1 clip y start.
14-13	R	0	Unused.
12-0	R/W	0x1FFF	SRC1 clip y end.

**Table 8-10 GE2D\_SRC1\_CANVAS 0x8A9**

Bit(s)	R/W	Default	Description
-24	R/W	0	SRC1 canvas address0, for Y only or Y/Cb/Cr stored together.
7-3	R	0	Unused.
2-0	R/W	0	Blk32 mode

**Table 8-11 GE2D\_SRC1\_X\_START\_END 0x8AA**

Bit(s)	R/W	Default	Description
31	R/W	0	SRC1 x start extra bit1, if true, one more chroma data is read for x even start chroma data when y/c ratio = 2 or x even/odd start chroma extra data when y/c ratio = 1
30	R/W	0	SRC1 x start extra bit0, if true, one more chroma data is read for x odd start chroma data when y/c ratio = 2
29-16	R/W	0	SRC1 x start, signed data
15	R/W	0	SRC1 x end extra bit1, if true, one more chroma data is read for x odd end chroma data when y/c ratio = 2 or x even/odd end chroma extra data when y/c ratio = 1
14	R/W	0	SRC1 x end extra bit0, if true, one more chroma data is read for x even end chroma data when y/c ratio = 2
13-0	R/W	0	SRC1 x end, signed data.

**Table 8-12 GE2D\_SRC1\_Y\_START\_END 0x8AB**

Bit(s)	R/W	Default	Description
31	R/W	0	SRC1 y start extra bit1, if true, one more chroma line is read for y even start chroma data when y/c ratio = 2 or x even/odd start chroma extra data when y/c ratio = 1
30	R/W	0	SRC1 y start extra bit0, if true, one more chroma line is read for y odd start chroma data when y/c ratio = 2
29-16	R/W	0	SRC1 y start, signed data
15	R/W	0	SRC1 y end extra bit1, if true, one more chroma line is read for y odd end chroma data when y/c ratio = 2 or y even/odd end chroma extra data when y/c ratio = 1
14	R/W	0	SRC1 y end extra bit0, if true, one more chroma line is read for y even end chroma data when y/c ratio = 2
13-0	R/W	0	SRC1 y end, signed data.

**Table 8-13 GE2D\_SRC1\_LUT\_ADDR 0x8AC**

Bit(s)	R/W	Default	Description
31-9	R	0	Unused.
8	R/W	1	0 = Write LUT, 1 = Read LUT.
7-0	R/W	0	lut_addr: The initial read or write address of the look-up table

**Table 8-14 GE2D\_SRC1\_LUT\_DAT 0x8AD**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Current LUT entry's Y or R
23-16	R/W	0	Current LUT entry's Cb or G
15-8	R/W	0	Current LUT entry's Cr or B
7-0	R/W	0	Current LUT entry's Alpha.

**Table 8-15 GE2D\_SRC1\_FMT\_CTRL 0x8AE**

Bit(s)	R/W	Default	Description
31-20	R	0	Unused.
19	R/W	0	src1_chfmt_rpt_pix: if true, horizontal formatter using repeat to get the pixel, otherwise using interpolation.
18	R/W	0	src1_chfmt_en: horizontal formatter enable.
17	R/W	0	src1_cvfmt_rpt_pix: if true, vertical formatter using repeat to get the pixel, otherwise using interpolation.
16	R/W	0	src1_cvfmt_en: vertical formatter enable.
15-8	R/W	0	src1_x_chr_phase: X direction chroma phase, Bit[15:12] for x direction even start/end chroma phase when y/c ratio = 2 or start/end even/odd chroma phase when y/c ratio = 1; Bit[11:8] for x direction odd start/end chroma phase only when y/c ratio = 2.
7-0	R/W	0	src1_y_chr_phase: Y direction chroma phase. Bit[7:4] for y direction even start/end chroma phase when y/c ratio = 2 or start/end even/odd chroma phase when y/c ratio = 1; Bit[3:0] for y direction odd start/end chroma phase only when y/c ratio = 2.

**Table 8-16 GE2D\_SRC2\_DEF\_COLOR 0x8AF**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Default Y or R.
23-16	R/W	0x80	Default Cb or G.
15-8	R/W	0x80	Default Cr or B.
7-0	R/W	0	Default Alpha.

**Table 8-17 GE2D\_SRC2\_CLIPX\_START\_END 0x8B0**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	SRC2 clip x start.
15-13	R	0	Unused.
12-0	R/W	0x1FFF	SRC2 clip x end.

**Table 8-18 GE2D\_SRC2\_CLIPY\_START\_END 0x8B1**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	SRC2 clip y start.
15-13	R	0	Unused.
12-0	R/W	0x1FFF	SRC2 clip y end.

**Table 8-19 GE2D\_SRC2\_X\_START\_END 0x8B2**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	SRC2 x start.
15-13	R	0	Unused.
12-0	R/W	0	SRC2 x end.

**Table 8-20 GE2D\_SRC2\_Y\_START\_END 0x8B3**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	SRC2 y start.
15-13	R	0	Unused.
12-0	R/W	0	SRC2 y end.

**Table 8-21 GE2D\_DST\_CLIPX\_START\_END 0x8B4**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	DST clip x start.
15-13	R	0	Unused.
12-0	R/W	0x1FFF	DST clip x end.

**Table 8-22 GE2D\_DST\_CLIPY\_START\_END 0x8B5**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	DST clip y start.
15-13	R	0	Unused.
12-0	R/W	0x1FFF	DST clip y end.



**Table 8-23 GE2D\_DST\_X\_START\_END 0x8B6**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	DST x start.
15-13	R	0	Unused.
12-0	R/W	0	DST x end.

**Table 8-24 GE2D\_DST\_Y\_START\_END 0x8B7**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	DST y start.
15-13	R	0	Unused.
12-0	R/W	0	DST y end.

**Table 8-25 GE2D\_SRC2\_DST\_CANVAS 0x8B8**

Bit(s)	R/W	Default	Description
31-27	R	0	Unused.
26	R/W	0	Dst2 32blk
25	R/W	0	Src2 32blk
23	R/W	0	Dst1 32blk
23-16	R/W	0	DST2 canvas address.
15-8	R/W	0	SRC2 canvas address.
7-0	R/W	0	DST1 canvas address.

**Table 8-26 GE2D\_VSC\_START\_PHASE\_STEP 0x8B9**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-0	R/W	0x0100-0000	5.24 format.

**Table 8-27 GE2D\_VSC\_PHASE\_SLOPE 0x8BA**

Bit(s)	R/W	Default	Description
31-25	R	0	Unused.
24-0	R/W	0	Signed data.

**Table 8-28 GE2D\_VSC\_INI\_CTRL 0x8BB**

Bit(s)	R/W	Default	Description
31	R	0	Unused.
30-29	R/W	0	vertical repeat line0 number.
28-24	R	0	Unused.
24	R/W	0	Rpt last In mode
23-0	R/W	0	vertical scaler initial phase.

**Table 8-29 GE2D\_HSC\_START\_PHASE\_STEP 0x8BC**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-0	R/W	0x0100-0000	5.24 format.

**Table 8-30 GE2D\_HSC\_PHASE\_SLOPE 0x8BD**

Bit(s)	R/W	Default	Description
31-25	R	0	Unused.
24-0	R/W	0	Signed data.

**Table 8-31 GE2D\_HSC\_INI\_CTRL 0x8BE**

Bit(s)	R/W	Default	Description
31	R	0	Unused.
30-29	R/W	0	horizontal repeat line0 number.
28	R	0	Unused.
27-24	R/W	0	advance number in this round, if horizontal scaler is working on dividing mode. (hsc_adv_num[11:8])
23-0	R/W	0	horizontal scaler initial phase.

**Table 8-32 GE2D\_HSC\_ADV\_CTRL 0x8BF**

Bit(s)	R/W	Default	Description
31-24	R/W	0	advance number in this round, if horizontal scaler is working on dividing mode. (hsc_avd_num[7:4])
23-0	R/W	0	horizontal scaler advance phase in this round, if horizontal scaler is working on dividing mode.

**Table 8-33 GE2D\_SC\_MISC\_CTRL 0x8C0**

Bit(s)	R/W	Default	Description
31	R	0	Unused.
30	R/W	0	vsc_nearest_en: vertical nearest mode enable, must set vt_bank_length = 4.
29	R/W	0	hsc_nearest_en: horizontal nearest mode enable, must set hz_bank_length = 4.
28	R/W	0	hsc_div_en: horizontal scaler dividing mode enable.
27-15	R/W	0	hsc_div_length: horizontal dividing length, if bit 25 is enable.
14	R/W	0	pre horizontal scaler enable.
13	R/W	0	pre vertical scale enable.
12	R/W	0	vertical scale enable.
11	R/W	0	horizontal scaler enable.
10	R	0	Unused.
9	R/W	0	HSc_rpt_ctrl: if true, treat horizontal repeat line number(GE2D_HSC_INI_CTRL bit 30:29) as repeating line, otherwise using treat horizontal repeat line number as minus line number.
8	R/W	0	VSc_rpt_ctrl: if true, treat vertical repeat line number(GE2D_VSC_INI_CTRL bit 30:29) as repeating line, otherwise using treat vertical repeat line number as minus line number.
7	R/W	0	vsc_phase0_always_en: if true, always use phase0 in vertical scaler.
6-4	R/W	2	vsc_bank_length: vertical scaler bank length.
3	R/W	0	hsc_phase0_always_en: if true, always use phase0 in horizontal scaler.
2-0	R/W	2	hsc_bank_length: horizontal scaler bank length.

**Table 8-34 GE2D\_VSC\_NRND\_POINT 0x8C1**

Bit(s)	R/W	Default	Description
31-14	R	0	Unused.
13-0	R	0	vertical scaler next round integer pixel pointer, signed data.

**Table 8-35 GE2D\_VSC\_NRND\_PHASE 0x8C2**

Bit(s)	R/W	Default	Description
31-24	R	0	Unused.
23-0	R	0	vertical scaler next round phase.

**Table 8-36 GE2D\_HSC\_NRND\_POINT 0x8C3**

Bit(s)	R/W	Default	Description
31-14	R	0	Unused.
13-0	R	0	horizontal scaler next round integer pixel pointer, signed data.

**Table 8-37 GE2D\_HSC\_NRND\_PHASE 0x8C4**

Bit(s)	R/W	Default	Description
31-24	R	0	Unused.
23-0	R	0	horizontal scaler next round phase.

**Table 8-38 GE2D\_MATRIX\_PRE\_OFFSET 0x8C5**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-20	R/W	0	pre_offset0.
19	R	0	Unused.
18-10	R/W	0	pre_offset1.
9	R	0	Unused.
8-0	R/W	0	pre_offset2.

**Table 8-39 GE2D\_MATRIX\_COEF00\_01 0x8C6**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	coef00.
15-13	R	0	Unused.
12-0	R/W	0	coef01.

**Table 8-40 GE2D\_MATRIX\_COEF02\_10 0x8C7**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	Coef02.
15-13	R	0	Unused.
12-0	R/W	0	Coef10.

**Table 8-41 GE2D\_MATRIX\_COEF11\_12 0x8C8**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	Coef11.
15-13	R	0	Unused.
12-0	R/W	0	Coef12.

Table 8-42 GE2D\_MATRIX\_COEF20\_21 0x8C9

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	Coef20.
15-13	R	0	Unused.
12-0	R/W	0	Coef21.

Table 8-43 GE2D\_MATRIX\_COEF22\_CTRL 0x8CA

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	coef22.
15-8	R	0	Unused.
7	R/W	0	input y/cb/cr saturation enable.
6-1	R	0	Unused.
0	R/W	0	conversion matrix enable.

Table 8-44 GE2D\_MATRIX\_OFFSET 0x8CB

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-20	R/W	0	offset0.
19	R	0	Unused.
18-10	R/W	0	offset1.
9	R	0	Unused.
8-0	R/W	0	offset2.

Table 8-45 GE2D\_ALU\_OP\_CTRL 0x8CC

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-27	R/W	0	Src2 Ad mux. 0:Ad=Adr*Ag, 1:Ad=Adr*Ag*Adr*Ag, 2:Ad=Adr*Ag*Adr, 3:Ad = 1
26-25	R/W	0	SRC1 color multiplier alpha selection. if 00, Cs = Csr if 01, Cs = Csr * Asr * Ag (if source is not premultiplied) if 10, Cs = Csr * Ag (if source is premultiplied).
24-23	R/W	0	SRC2 color multiplier alpha selection. 0: Cd=Cdr, 1:Cd=Cdr*Adr*Adg, 2:Cd=Cdr*Adg
22-12	R/W	0x010	ALU color operation. Bit[22:20] Blending Mode Parameter. 3'b000: ADD Cs*Fs + Cd*Fd 3'b001: SUBTRACT Cs*Fs - Cd*Fd 3'b010: REVERSE SUBTRACT Cd*Fd - Cs*Fs 3'b011: MIN min(Cs*Fs, Cd*Fd) 3'b100: MAX max(Cs*Fs, Cd*Fd) 3'b101: LOGIC OP Cs op Cd

Bit(s)	R/W	Default	Description
			3'b110: DARKEN $Cs*Fs + Cd*Fd + \min(Cs,Cd)$ 3'b111: LIGHTEN $Cs*Fs + Cd*Fd + \max(Cs,Cd)$ .... reserved Bit[19:16] Source Color Blending Factor CFs. 4'b0000: ZERO 0 4'b0001: ONE 1 4'b0010: SRC_COLOR Cs(RGBs) 4'b0011: ONE_MINUS_SRC_COLOR 1 - Cs(RGBs) 4'b0100: DST_COLOR Cd(RGBd) 4'b0101: ONE_MINUS_DST_COLOR 1 - Cd(RGBd) 4'b0110: SRC_ALPHA As 4'b0111: ONE_MINUS_SRC_ALPHA 1 - As 4'b1000: DST_ALPHA Ad 4'b1001: ONE_MINUS_DST_ALPHA 1 - Ad 4'b1010: CONST_COLOR Cc(RGBc) 4'b1011: ONE_MINUS_CONST_COLOR 1 - Cc(RGBc) 4'b1100: CONST_ALPHA Ac 4'b1101: ONE_MINUS_CONST_ALPHA 1 - Ac 4'b1110: SRC_ALPHA_SATURATE $\min(As,1-Ad)$ .... reserved Bit[15:12] dest Color Blending Factor CFd, when bit[22:20] != LOGIC OP. 4'b0000: ZERO 0 4'b0001: ONE 1 4'b0010: SRC_COLOR Cs(RGBs) 4'b0011: ONE_MINUS_SRC_COLOR 1 - Cs(RGBs) 4'b0100: DST_COLOR Cd(RGBd) 4'b0101: ONE_MINUS_DST_COLOR 1 - Cd(RGBd) 4'b0110: SRC_ALPHA As 4'b0111: ONE_MINUS_SRC_ALPHA 1 - As 4'b1000: DST_ALPHA Ad 4'b1001: ONE_MINUS_DST_ALPHA 1 - Ad 4'b1010: CONST_COLOR Cc(RGBc) 4'b1011: ONE_MINUS_CONST_COLOR 1 - Cc(RGBc) 4'b1100: CONST_ALPHA Ac 4'b1101: ONE_MINUS_CONST_ALPHA 1 - Ac 4'b1110: SRC_ALPHA_SATURATE $\min(As,1-Ad)$ .... reserved Bit[15:12] logic operations, when bit[22:20] == LOGIC OP. 4'b0000: CLEAR 0 4'b0001: COPY s 4'b0010: NOOP d 4'b0011: SET 1 4'b0100: COPY_INVERT ~s 4'b0101: INVERT ~d 4'b0110: AND_REVERSE s & ~d 4'b0111: OR_REVERSE s   ~d 4'b1000: AND s & d 4'b1001: OR s   d 4'b1010: NAND ~(s & d) 4'b1011: NOR ~(s   d) 4'b1100: XOR s ^ d 4'b1101: EQUIV ~(s ^ d) 4'b1110: AND_INVERTED ~s & d 4'b1111: OR_INVERTED ~s   d
11	R	0	Unused.

Bit(s)	R/W	Default	Description
10-0	R/W	0x010	ALU alpha operation. Bit[10:8] Blending Equation Math Operation. 3'b000: ADD $As*Fs + Ad*Fd$ 3'b001: SUBTRACT $As*Fs - Ad*Fd$ 3'b010: REVERSE SUBTRACT $Ad*Fd - As*Fs$ 3'b011: MIN $\min(As*Fs, Ad*Fd)$ 3'b100: MAX $\max(As*Fs, Ad*Fd)$ 3'b101: LOGIC OP $As \text{ op } Ad$ .... reserved Bit[7:4] Source alpha Blending Factor AFs. 4'b0000 0 4'b0001 1 4'b0010 $As$ 4'b0011 $1 - As$ 4'b0100 $Ad$ 4'b0101 $1 - Ad$ 4'b0110 $Ac$ 4'b0111 $1 - Ac$ .... reserved Bit[3:0] Destination alpha Blending Factor AFd, when bit[10:8] != LOGIC OP. 4'b0000 0 4'b0001 1 4'b0010 $As$ 4'b0011 $1 - As$ 4'b0100 $Ad$ 4'b0101 $1 - Ad$ 4'b0110 $Ac$ 4'b0111 $1 - Ac$ .... reserved Bit[3:0] logic operations, when bit[10:8] == LOGIC OP. 4'b0000: CLEAR 0 4'b0001: COPY $s$ 4'b0010: NOOP $d$ 4'b0011: SET 1 4'b0100: COPY_INVERT $\sim s$ 4'b0101: INVERT $\sim d$ 4'b0110: AND_REVERSE $s \& \sim d$ 4'b0111: OR_REVERSE $s   \sim d$ 4'b1000: AND $s \& d$ 4'b1001: OR $s   d$ 4'b1010: NAND $\sim(s \& d)$ 4'b1011: NOR $\sim(s   d)$ 4'b1100: XOR $s \wedge d$ 4'b1101: EQUIV $\sim(s \wedge d)$ 4'b1110: AND_INVERTED $\sim s \& d$ 4'b1111: OR_INVERTED $\sim s   d$

Table 8-46 GE2D\_ALU\_CONST\_COLOR 0x8CD

Bit(s)	R/W	Default	Description
31-0	R/W	0x0080-8000	RGBA or YCbCrA.

Table 8-47 GE2D\_SRC1\_KEY 0x8CE

Bit(s)	R/W	Default	Description
31-0	R/W	0	SRC1 Key.

**Table 8-48 GE2D\_SRC1\_KEY\_MASK 0x8CF**

Bit(s)	R/W	Default	Description
31-0	R/W	0	SRC1 Key Mask.

**Table 8-49 GE2D\_SRC2\_KEY 0x8D0**

Bit(s)	R/W	Default	Description
31-0	R/W	0	SRC2 Key.

**Table 8-50 GE2D\_SRC2\_KEY\_MASK 0x8D1**

Bit(s)	R/W	Default	Description
31-0	R/W	0	SRC2 Key Mask.

**Table 8-51 GE2D\_DST\_BITMASK 0x8D2**

Bit(s)	R/W	Default	Description
31-0	R/W	0	Destination Bit Mask.

**Table 8-52 GE2D\_DP\_ONOFF\_CTRL 0x8D3**

Bit(s)	R/W	Default	Description
31	R/W	0	DP onoff mode. 0: on_counter means how many pixels will output before ge2d turns off; 1: on_counter means how many clocks will ge2d turn on before ge2d turns off.
30-16	R/W	0	DP on counter.
15	R/W	0	0: vd_format doesnt have onoff mode, 1: vd format has onoff mode.
14-0	R/W	0	DP off counter.

**Table 8-53 GE2D\_SCALE\_COEF\_IDX 0x8D4**

Because there are many coefficients used in the vertical filter and horizontal filters, indirect access the coefficients of vertical filter and horizontal filter is used. For vertical filter, there are 33x4 coefficients. For horizontal filter, there are 33x4 coefficients.

Bit(s)	R/W	Default	Description
31-16	R	0	Unused.
15	R/W	0	index increment, if bit9 == 1 then (0: index increase 1, 1: index increase 2) else (index increase 2).
14	R/W	0	1: read coef through cbus enable, just for debug purpose in case when we wanna check the coef in ram in correct or not.
13-10	R	0	Unused.
9	R/W	0	if true, use 9bit resolution coef, other use 8bit resolution coef.
8	R/W	0	type of index, 0: vertical coef; 1: horizontal coef.



Bit(s)	R/W	Default	Description
7	R	0	Unused.
6-0	R/W	0	coef index.

Table 8-54 GE2D\_SCALE\_COEF 0x8D5

Bit(s)	R/W	Default	Description
31-0	R/W	0	coefficients for vertical filter and horizontal filter.

Table 8-55 GE2D\_SRC\_OUTSIDE\_ALPHA 0x8D6

Bit(s)	R/W	Default	Description
31-25	R	0	Unused.
24	R/W	0	src2 alpha fill mode: together with GE2D_GEN_CTRL0[14](fill_mode), define what alpha values are used. 0: repeat inner alpha, 1: fill src2 outside alpha for the area outside the clipping window. As below: fill_mode=0, alpha_fill_mode=0 : use inner alpha, (or default_alpha if src data have no alpha values); fill_mode=0, alpha_fill_mode=1 : use outside_alpha; fill_mode=1, alpha_fill_mode=0 : use default_alpha; fill_mode=1, alpha_fill_mode=1 : use outside_alpha.
23-16	R/W	16	src2 outside alpha.
15-9	R	0	Unused.
8	R/W	1	src1 alpha fill mode, refer to src2 alpha fill mode above.
7-0	R/W	0	src1 outside alpha.

Table 8-56 GE2D\_ANTIFLICK\_CTRL0 0x8D8

Bit(s)	R/W	Default	Description
31	R/W	0	antiflick enable
24	R/W	0	1: alpha value for the first line use repeated alpha, 0: use bit 23:16 as the first line alpha
23-16	R/W	0	register value for the first line alpha when bit 24 is 1.
8	R/W	0	1: alpha value for the last line use repeated alpha, 0: use bit 7:0 as the last line alpha
7-0	R/W	0	register value for the last line alpha when bit 8 is 1.

Table 8-57 GE2D\_ANTIFLICK\_CTRL1 0x8D9

Bit(s)	R/W	Default	Description
25	R/W	0	rgb_sel, 1: antiflick RGBA, 0: antiflick YCbCrA
24	R/W	0	cbr_en, 1: also filter cbr in case of antiflicking YCbCrA, 0: no filter on cbr in case of antiflicking YCbCrA
23-16	R/W	0	R mult coef for converting RGB to Y

Bit(s)	R/W	Default	Description
15- 8	R/W	0	G mult coef for converting RGB to Y
7-0	R/W	0	B mult coef for converting RGB to Y

$$Y = (R * y_r + G * y_g + B * y_b) / 256$$

**Table 8-58 GE2D\_ANTIFLICK\_COLOR\_FILT0 0x8DA**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Y threshold1, when $0 < Y \leq th1$ , use filter0.
23-16	R/W	0	color antiflick filter0 n3
15- 8	R/W	0	color antiflick filter0 n2
7-0	R/W	0	color antiflick filter0 n1

$$Y = (line\_up * n1 + line\_center * n2 + line\_dn * n3) / 128$$

**Table 8-59 GE2D\_ANTIFLICK\_COLOR\_FILT1 0x8DB**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Y threshold2, when $th1 < Y \leq th2$ , use filter1.
23-16	R/W	0	color antiflick filter1 n3
15- 8	R/W	0	color antiflick filter1 n2
7-0	R/W	0	color antiflick filter1 n1

**Table 8-60 GE2D\_ANTIFLICK\_COLOR\_FILT2 0x8DC**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Y threshold3, when $th2 < Y \leq th3$ , use filter2 ; $Y > th3$ , use filter3.
23-16	R/W	0	color antiflick filter2 n3
15- 8	R/W	0	color antiflick filter2 n2
7-0	R/W	0	color antiflick filter2 n1

**Table 8-61 GE2D\_ANTIFLICK\_COLOR\_FILT3 0x8DD**

Bit(s)	R/W	Default	Description
23-16	R/W	0	color antiflick filter3 n3
15- 8	R/W	0	color antiflick filter3 n2
7-0	R/W	0	color antiflick filter3 n1

**Table 8-62 GE2D\_ANTIFLICK\_ALPHA\_FILT0 0x8DE**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Alpha threshold1, when $0 < Alpha \leq th1$ , use filter0.
23-16	R/W	0	Alpha antiflick filter0 n3

Bit(s)	R/W	Default	Description
15- 8	R/W	0	Alpha antiflick filter0 n2
7-0	R/W	0	Alpha antiflick filter0 n1

$$\text{Alpha} = (\text{line\_up} * n1 + \text{line\_center} * n2 + \text{line\_dn} * n3) / 128$$

**Table 8-63 GE2D\_ANTIFLICK\_ALPHA\_FILT1 0x8DF**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Alpha threshold2, when $th1 < \text{Alpha} \leq th2$ , use filter1.
23-16	R/W	0	Alpha antiflick filter1 n3
15- 8	R/W	0	Alpha antiflick filter1 n2
7-0	R/W	0	Alpha antiflick filter1 n1

**Table 8-64 GE2D\_ANTIFLICK\_ALPHA\_FILT2 0x8E0**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Alpha threshold3, when $th2 < \text{Alpha} \leq th3$ , use filter2; $\text{Alpha} > th3$ , use filter3.
23-16	R/W	0	Alpha antiflick filter2 n3
15- 8	R/W	0	Alpha antiflick filter2 n2
7-0	R/W	0	Alpha antiflick filter2 n1

**Table 8-65 GE2D\_ANTIFLICK\_ALPHA\_FILT3 0x8E1**

Bit(s)	R/W	Default	Description
23-16	R/W	0	Alpha antiflick filter3 n3
15- 8	R/W	0	Alpha antiflick filter3 n2
7-0	R/W	0	Alpha antiflick filter3 n1

**Table 8-66 GE2D\_SRC1\_RANGE\_MAP\_Y\_CTRL 0x8E3**

Bit(s)	R/W	Default	Description
30-22	R/W	0	din_offset (signed data)
21-14	R/W	0	map_coef (unsigned data)
13- 10	R/W	0	map_sr (unsigned data)
9-1	R/W	0	dout_offset (signed data)
0	R/W	0	enable

$$\text{dout} = \text{clipto\_0\_255}(((\text{din} + \text{din\_offset}) * \text{map\_coef} + ((1 \ll (\text{map\_sr} - 1))) \gg \text{map\_sr} + \text{dout\_offset})$$

**Table 8-67 GE2D\_SRC1\_RANGE\_MAP\_CB\_CTRL 0x8E4**

Bit(s)	R/W	Default	Description
30-22	R/W	0	din_offset (signed data)
21-14	R/W	0	map_coef (unsigned data)
13- 10	R/W	0	map_sr (unsigned data)
9-1	R/W	0	dout_offset (signed data)
0	R/W	0	enable

dout = clipto\_0\_255(((din + din\_offset) \* map\_coef + ((1 << (map\_sr - 1))) >> map\_sr + dout\_offset)

**Table 8-68 GE2D\_SRC1\_RANGE\_MAP\_CR\_CTRL 0x8E5**

Bit(s)	R/W	Default	Description
30-22	R/W	0	din_offset (signed data)
21-14	R/W	0	map_coef (unsigned data)
13- 10	R/W	0	map_sr (unsigned data)
9-1	R/W	0	dout_offset (signed data)
0	R/W	0	enable

dout = clipto\_0\_255(((din + din\_offset) \* map\_coef + ((1 << (map\_sr - 1))) >> map\_sr + dout\_offset)

**Table 8-69 GE2D\_ARB\_BURST\_NUM 0x8E6**

Bit(s)	R/W	Default	Description
21-16	R/W	0x3f	Src1 prearbitor burst number
13-8	R/W	0x3f	Src2 prearbitor burst number
5-0	R/W	0x3f	dst prearbitor burst number

**Table 8-70 GE2D\_TID\_TOKEN 0x8E7**

Bit(s)	R/W	Default	Description
21-16	R/W	0x3f	Src1 ID. High 4bit are thread ID, low 2bits are the token
13-8	R/W	0x3f	Src2 ID. High 4bit are thread ID, low 2bits are the token
5-0	R/W	0x3f	dst ID. High 4bit are thread ID, low 2bits are the token

**Table 8-71 GE2D\_GEN\_CTRL3 0x8E8**

Bit(s)	R/W	Default	Description
31-28	R/W	0	dst2_bytemask_val: 1-bit mask for each byte (8-bit). Applicable only if both dst_bitmask_en=1 and dst_bytemask_only=1.
27-26	R/W	0	dst2_pic_struct: Define how destination 2 write the picture to DDR memory. 0: Write all lines (whole frame); 1: Reserved; 2: Write even lines only (top);

Bit(s)	R/W	Default	Description
			3: Write odd lines only (bottom).
25-24	R/W	0	dst2_8b_mode_sel: Destination 8-bit mode component selection. 0: Select Y or R; 1: Select Cb or G; 2: Select Cr or B; 3: Select Alpha.
23	R	0	Unused
22-19	R/W	0	dst2_color_map: Applicable to 16-bit, 24-bit and 32-bit pixel, defines the bit-field allocation of the pixel data. For whether to truncate or round full 8-bit to output, refer to ge2d_gen_ctrl3.dst2_color_round_mode. For 16-bit mode (dst2_format=1): 0 = Unused; 1 = 6:5:5 format. Bit[15:10] is Y [7:2] or R[7:2], bit[9:5] is Cb[7:3] or G[7:3], bit[4:0] is Cr[7:3] or B[7:3]; 2 = 8:4:4 format. Bit[15:8] is Y or R, bit[7:4] is Cb[7:4] or G[7:4], bit[3:0] is Cr[7:4] or B[7:4]; 3 = 6:4:4:2 format. Bit[15:10] is Y[7:2] or R[7:2], bit[9:6] is Cb[7:4] or G[7:4], bit [5:2] is Cr[7:4] or B[7:4], bit[1:0] is Alpha[7:6]; 4 = 4:4:4:4 format. Bit[15:12] is Y [7:4] or R[7:4], bit[11:8] is Cb[7:4] or G[7:4], bit[7:4] is Cr[7:4] or B[7:4], bit[3:0] is Alpha[7:4]; 5 = 5:6:5 format. Bit[15:11] is Y[7:3] or R[7:3], bit[10:5] is Cb[7:2] or G [7:2], bit[4:0] is Cr[7:3] or B[7:3]; 6 = 4:4:4:4 format. Bit[15:12] is Alpha[7:4], bit [11:8] is Y[7:4] or R[7:4], bit[7:4] is Cb[7:4] or G[7:4], bit[3:0] is Cr[7:4] or B[7:4]; 7 = 1:5:5:5 format. Bit[15] is Alpha[7], bit[14:10] is Y[7:3] or R[7:3], bit[9:5] is Cb [7:3] or G[7:3], bit[4:0] is Cr[7:3] or B[7:3]; 8 = 4:6:4:2 format. Bit[15:12] is Y[7:4] or R[7:4], bit[11:6] is Cb[7:2] or G[7:2], bit[5:2] is Cr[7:4] or B[7:4], bit[1:0] is Alpha [1:0]; 9 = CbCr format. Bit[15:8] is Cb, bit[7:0] is Cr; 10 = CrCb format. Bit[15:8] is Cr, bit[7:0] is Cb. For 24-bit mode (dst2_format=2): 0 = RGB 8:8:8 mode. Bit [23:16] is Y or R, bit[15:8] is Cb or G, bit[7:0] is Cr or B; 1 = RGBA 5:6:5:8 mode. Bit[23:19] is Y[7:3] or R[7:3], bit[18:13] is Cb[7:2] or G[7:2], bit[12:8] is Cr[7:3] or B[7:3], bit[7:0] is Alpha; 2 = ARGB 8:5:6:5 mode. Bit[23:16] is Alpha, bit[15:11] is Y[7:3] or R[7:3], bit[10:5] is Cb[7:2] or G[7:2], bit[4:0] is Cr[7:3] or B[7:3]; 3 = RGBA 6:6:6:6 mode. Bit[23:18] is Y[7:2] or R[7:2], bit[17:12] is Cb[7:2] or G[7:2], bit[11:6] is Cr[7:2] or B[7:2], bit[5:0] is Alpha[7:2]; 4 = ARGB 6:6:6:6 mode. Bit [23:18] is Alpha[7:2]; bit[17:12] is Y[7:2] or R[7:2], bit[11:6] is Cb[7:2] or G[7:2], bit[5:0] is Cr[7:2] or B[7:2]; 5 = BGR 8:8:8 mode. Bit[23:16] is Cr or B, bit[15:8] is Cb or G, bit[7:0] is Y or R. For 32-bit mode (dst2_format=3): 0 = RGBA 8:8:8:8 format. Bit[31:24] is Y or R, bit[23:16] is Cb or G; bit[15:8] is Cr or B; bit[7:0] is Alpha; 1 = ARGB 8:8:8:8 format. Bit[31:24] is Alpha, bit[23:16] is Y or R; bit[15:8] is Cb or G; bit[7:0] is Cr or B; 2 = ABGR 8:8:8:8 format. Bit[31:24] is Alpha, bit [23:16] is Cr or B; bit[15:8] is Cb or G; bit[7:0] is Y or R; 3 = BGRA 8:8:8:8 format. Bit[31:24] is Cr or B, bit[23:16] is Cb or G; bit[15:8] is Y or R; bit[7:0] is Alpha.
18	R	0	Unused
17-16	R/W	0	dst2_format: define DST2 output pixel byte-width. 0: Output pixel is 1-byte (8-bit) color component; 1: Output pixel is 2-byte (16-bit), refer to GE2D_GEN_CTRL3.dst2_COLOR_ map for further pixel color mapping; 2: Output pixel is 3-byte (24-bit), refer to GE2D_GEN_CTRL3.dst2_COLOR_ MAP for further pixel color mapping; 3: Output pixel is 4-byte (32-bit), refer to GE2D_GEN_CTRL3.dst2_COLOR_ MAP for further pixel color mapping.
15	R	0	Unused
14	R/W	0	dst2_COLOR_round_MODE. 1 = Truncate the full bit color components to required output bit width; 0 = Round (+ 0.5) the full bit color components to required output bit width.
13-12	R/W	0	dst2_x_discard_mode: Define how DST2 discard X direction data before writing to DDR. Note: x is post reverse/rotation. 0: No discard; 1: Reserved; 2: discard even x; 3: discard odd x.
11-10	R/W	0	dst2_y_discard_mode: Define how DST2 discard Y direction data before writing to DDR.

Bit(s)	R/W	Default	Description
			Note: y is post reverse/rotation. 0: No discard; 1: Reserved; 2: discard even y; 3: discard odd y.
9	R	0	Unused
8	R/W	0	dst2_enable: 0: Disable destination 2; 1: Enable destination 2.
7-6	R	0	Unused
5-4	R/W	0	dst1_x_discard_mode: Define how DST1 discard X direction data before writing to DDR. Note: x is post reverse/rotation. 0: No discard; 1: Reserved; 2: discard even x; 3: discard odd x.
3-2	R/W	0	dst1_y_discard_mode: Define how DST1 discard Y direction data before writing to DDR. Note: y is post reverse/rotation. 0: No discard; 1: Reserved; 2: discard even y; 3: discard odd y.
1	R	0	Unused
0	R/W	1	dst1_enable: 0: Disable destination 1; 1: Enable destination 1.

Table 8-72 GE2D\_STATUS2 0x8E9

Bit(s)	R/W	Default	Description
13-12	R	0	ge2d_dst1.ctrl_status
11	R	0	ge2d_dst1.map_srdy
10	R	0	ge2d_dst1.map_d1_srdy
9	R	0	ge2d_dst1.s_v
8	R	0	ge2d_dst1.ofifo_dout_srdy
7-1	R	0	ge2d_dst1.ofifo_cnt
0	R	0	ge2d_dst1.dst_busy

Table 8-73 GE2D\_GEN\_CTRL4 0x8EA

Bit(s)	R/W	Default	Description
28	R/W	0	am2axi_len4_mode
0	R/W	0	dis_dp_hang_bugfix.

**Table 8-74 GE2D\_GEN\_CTRL5 0x8f1**

Bit(s)	R/W	Default	Description
8	R/W	0	Src2_gb_alpha_en
7-0	R/W	0	Src2_gb_alpha

**Table 8-75 GE2D\_MATRIX2\_PRE\_OFFSET 0x890**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-20	R/W	0	pre_offset0.
19	R	0	Unused.
18-10	R/W	0	pre_offset1.
9	R	0	Unused.
8-0	R/W	0	pre_offset2.

**Table 8-76 GE2D\_MATRIX2\_COEF00\_01 0x891**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	coef00.
15-13	R	0	Unused.
12-0	R/W	0	coef01.

**Table 8-77 GE2D\_MATRIX2\_COEF02\_10 0x892**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	Coef02.
15-13	R	0	Unused.
12-0	R/W	0	Coef10.

**Table 8-78 GE2D\_MATRIX2\_COEF11\_12 0x893**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	Coef11.
15-13	R	0	Unused.
12-0	R/W	0	Coef12.

**Table 8-79 GE2D\_MATRIX2\_COEF20\_21 0x894**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	Coef20.
15-13	R	0	Unused.
12-0	R/W	0	Coef21.

**Table 8-80 GE2D\_MATRIX2\_COEF22\_CTRL 0x895**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	coef22.
15-8	R	0	Unused.
7	R/W	0	input y/cb/cr saturation enable.
6	R/w	0	Output signed mode
5-1	R	0	Unused.
0	R/W	0	conversion matrix enable.

**Table 8-81 GE2D\_MATRIX2\_OFFSET 0x896**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-20	R/W	0	offset0.
19	R	0	Unused.
18-10	R/W	0	offset1.
9	R	0	Unused.
8-0	R/W	0	offset2.

**Table 8-82 GE2D\_MATRIX3\_PRE\_OFFSET 0x897**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-20	R/W	0	pre_offset0.
19	R	0	Unused.
18-10	R/W	0	pre_offset1.
9	R	0	Unused.
8-0	R/W	0	pre_offset2.



**Table 8-83 GE2D\_MATRIX3\_COEF00\_01 0x898**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	coef00.
15-13	R	0	Unused.
12-0	R/W	0	coef01.

**Table 8-84 GE2D\_MATRIX3\_COEF02\_10 0x899**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	Coef02.
15-13	R	0	Unused.
12-0	R/W	0	Coef10.

**Table 8-85 GE2D\_MATRIX3\_COEF11\_12 0x89a**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	Coef11.
15-13	R	0	Unused.
12-0	R/W	0	Coef12.

**Table 8-86 GE2D\_MATRIX3\_COEF20\_21 0x89b**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	Coef20.
15-13	R	0	Unused.
12-0	R/W	0	Coef21.

**Table 8-87 GE2D\_MATRIX3\_COEF22\_CTRL 0x89c**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-16	R/W	0	coef22.
15-8	R	0	Unused.
7	R/W	0	input y/cb/cr saturation enable.
6	R/W	0	Output signed mode
5-1	R	0	Unused.
0	R/W	0	conversion matrix enable.

**Table 8-88 GE2D\_MATRIX3\_OFFSET 0x89d**

Bit(s)	R/W	Default	Description
31-29	R	0	Unused.
28-20	R/W	0	offset0.
19	R	0	Unused.
18-10	R/W	0	offset1.
9	R	0	Unused.
8-0	R/W	0	offset2.

**Table 8-89 GE2D\_DST1\_BADDR\_CTRL 0x8f2**

Bit(s)	R/W	Default	Description
31-0	R/W	0	DST1 base address in 64bits.

**Table 8-90 GE2D\_DST1\_STRIDE\_CTRL 0x8f3**

Bit(s)	R/W	Default	Description
19	R/W	0	Blk32 mode
18-17	R/W	0	DST1 stride mode 0: linear, 1:32x32, 2: 64x32
16-0	R/W	0	DST1 stride size in 64bits.

**Table 8-91 GE2D\_DST2\_BADDR\_CTRL 0x8f4**

Bit(s)	R/W	Default	Description
31-0	R/W	0	DST2 base address in 64bits.

**Table 8-92 GE2D\_DST2\_STRIDE\_CTRL 0x8f5**

Bit(s)	R/W	Default	Description
19	R/W	0	Blk32 mode
18-17	R/W	0	DST2 stride mode 0: linear, 1:32x32, 2: 64x32
16-0	R/W	0	DST2 stride size in 64bits.

**Table 8-93 GE2D\_SRC1\_BADDR\_CTRL\_Y 0x8f6**

Bit(s)	R/W	Default	Description
31-0	R/W	0	SRC1 Y_channel base address in 64bits.

**Table 8-94 GE2D\_SRC1\_STRIDE\_CTRL\_Y 0x8f7**

Bit(s)	R/W	Default	Description
19	R/W	0	Blk32 mode
18-17	R/W	0	src1 Y stride mode 0: linear, 1:32x32, 2: 64x32
16-0	R/W	0	src1 Y stride size in 64bits.

**Table 8-95 GE2D\_SRC1\_BADDR\_CTRL\_CB 0x8f8**

Bit(s)	R/W	Default	Description
31-0	R/W	0	SRC1 cb_channel base address in 64bits.

**Table 8-96 GE2D\_SRC1\_STRIDE\_CTRL\_CB 0x8f9**

Bit(s)	R/W	Default	Description
19	R/W	0	Blk32 mode
18-17	R/W	0	src1 cb stride mode 0: linear, 1:32x32, 2: 64x32
16-0	R/W	0	src1 cb stride size in 64bits.

**Table 8-97 GE2D\_SRC1\_BADDR\_CTRL\_CR 0x8fa**

Bit(s)	R/W	Default	Description
31-0	R/W	0	SRC1 cr_channel base address in 64bits.

**Table 8-98 GE2D\_SRC1\_STRIDE\_CTRL\_CR 0x8fb**

Bit(s)	R/W	Default	Description
19	R/W	0	Blk32 mode
18-17	R/W	0	src1 cr stride mode 0: linear, 1:32x32, 2: 64x32
16-0	R/W	0	src1 cr stride size in 64bits.

**Table 8-99 GE2D\_SRC2\_BADDR\_CTRL 0x8fc**

Bit(s)	R/W	Default	Description
31-0	R/W	0	SRC2 base address in 64bits.

**Table 8-100 GE2D\_SRC2\_STRIDE\_CTRL 0x8fd**

Bit(s)	R/W	Default	Description
19	R/W	0	Blk32 mode
18-17	R/W	0	Src2 stride mode 0: linear, 1:32x32, 2: 64x32
16-0	R/W	0	Src2 stride size in 64bits.

# 9 Video Path

## 9.1 Video Input

### 9.1.1 Features

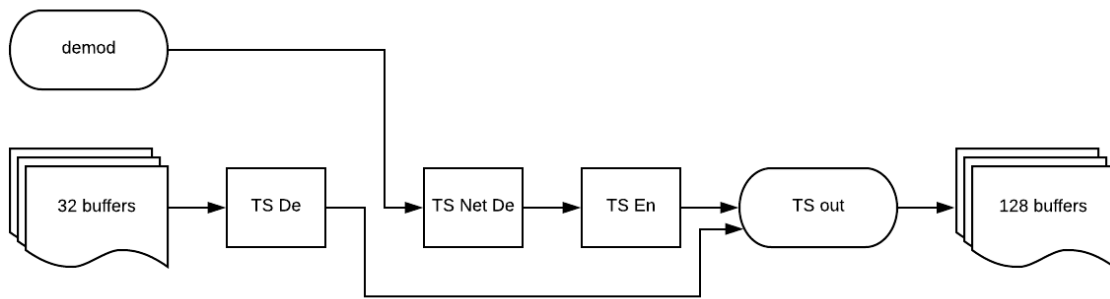
Support 16 TS streams from demodulators.

Support 32 TS streams from DDR memory, include TS streams from internet.

Support TS stream local decrypt, network decrypt, local encrypt.

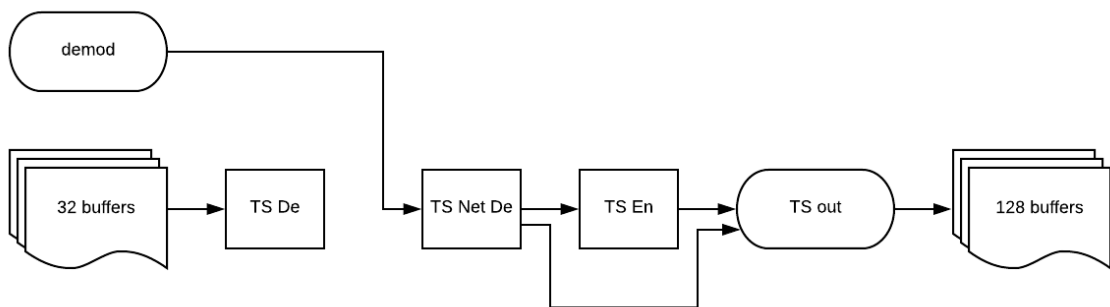
Support multiple TS streams demux and output data in TS, PES or ES format.

**Figure 9-1 connect to both demod and buffer**



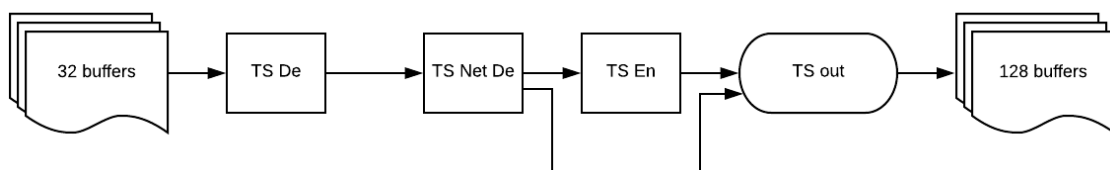
Record programs from Demod and view other programs from Demod or IP buffer

**Figure 9-2 connect to demod, play and record**



Record and view the same program from Demod, record other programs from Demod, read buffers are not used

**Figure 9-3 connect to buffer, play and record**



Record programs from IP, view programs from IP or buffer.

View and record the same program is achieved by output TSN packets to both TSE and TSOUT, the packets send to TSOUT with the stream SID MSB flipped.

In case of Demod, only the 32~63 are used, after MSB flipped, the SID is converted from 32~63 to 0~31 for view, the original SID packets are encrypted by local TS encryption and saved.

In case of IP Buffers, only the 0~31 are used, after MSB flipped, the SID is converted from 0~31 to 32~63 for view, the original SID packets are encrypted by local TS encryption and saved.

Setup the SIDs based on use case.

## 9.1.2 TS Input

### 9.1.2.1 Demodulator TS

Demodulator inputs from PINs or from internal demodulator in serial or parallel format

Multiple TS streams can be muxed by adding extra prefix bytes before input into the SOC, the extra bytes are removed by TS input, a side band signal “stream ID” or “SID” is used to identify different TS streams.

Demod TS packet size:

**Table 9-1 demod TS interface format**

Mode	Size	Note
Standard	188	Physical connection as SID Support 4 TS inputs
Extra bytes	192, 196, 200	Extract SID from extra bytes Support up to 16 SIDs
Extra RS parity	204	Removed by interface

### 9.1.2.2 Stream from Memory

If a TS stream is downloaded from internet or read from external storage, it is read from DDR memory by TS demux

TS demux supports up to 32 streams data read from DDR memory, TS packet tagged with SID and sent to TS crypto engine, to avoid the same PID from different stream sent to the same TS, 6 bits stream ID is used to separate the same PID case.

Stream ID combined with PID are used to select crypto algorithm and keys.

Secure level of input stream is the same as memory secure level.

When TS read from a memory location with secure level x, then the data is secure level x.

**Table 9-2 DDR buffer interface SID**

Source	Stream ID ( 6 bits )	Secure Level ( 3 bits )
DDR memory	0~31	The same as memory secure level

### 9.1.3 Crypto Algorithm

Only the following algorithms are supported in TS crypto.

TS Keys are only from Key Table

Host keys, MKL keys, ETSI keys and CAS IP keys are deposited into key table.

**Table 9-3 TS descrambler algorithm list**

Mode	Algorithm	EMI	TSD 8	TSN 9	TSE 10	IV
0	Pass through	N/A	P	P	P	
1	AES ECB clear end	0021	D	D	E	No IV
2	AES ECB clear front		D	D	E	No IV
3	AES CBC clear end	0022 0023	D	D	E	128 bits
4	AES CBC IDSA	0020	D	D	E	All zero
5	CSA2	0000	-	D	-	No IV
6	DES SCTE41	0031	-	D	-	No IV
7	DES SCTE52	0030	-	D	-	64 bits for CBC 64 bits for short payload
8	TDES ECB clear end	0041	-	D	-	No IV
9	CPCM LSA MDI CBC		D	D	E	MDI
10	CPCM LSA MDD CBC		D	D	E	MDD
11	CSA3	0001	-	D	-	No IV
12	ASA	0010 0011	-	D	-	No IV
13	ASA LIGHT	0012	-	D	-	No IV

Use OTP bits to disable DES SCTE41, DES SCTE52, CSA2, CSA3, CPCM LSA and DES

For mode 3 and mode 7 with 128 bits IV, retrieve the IV from key table IV slots with the flag as IV.

Host deposits the IVs into key table slot 0 to 63.

### 9.1.4 TS Output

Save TS/PES/ES data into DDR memory by scatter gather memory list.

Support up to 128 buffers, support byte alignment data address and byte length buffers.

### 9.1.4.1 Descriptor

**Table 9-4 TS data read/write descriptor**

bytes	7	6	5	4	3	2	1	0
0	byte length [23:0]							
1								
2								
3			loop	eoc	irq	Byte length [26:24]		
4	address 32 bits							
5								
6								
7								

Each descriptor is 64 bits, prepared by host and saved in DDR memory, the descriptors are stored in continuous memory address, the last descriptor with end of chain field “eoc” set to 1, or set the loop to 1 to loop back to the beginning of descriptor chain.

Memory access supports byte alignment memory read/write, the address can be any byte alignment and length of memory can be any number of bytes.

Maximum single descriptor memory size is  $2^{27}-1 = 128M-1$  bytes.

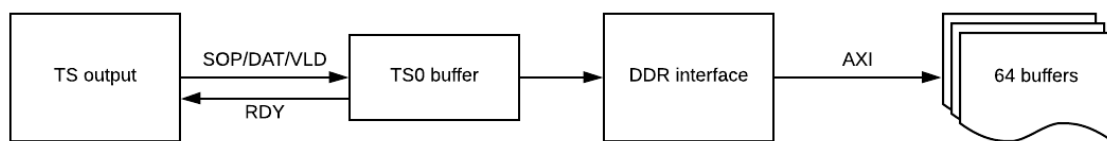
Loop : next descriptor is from beginning, support descriptor loop, send IRQ out when loop back.

Eoc : end of chain, done, send IRQ out

Irq : send IRQ when this descriptor is done.

The DDR memory interface uses 128 bits data bus, the TS crypto module should read/write fraction of 16 bytes first, then process number of 16 byte and finally with a fraction of 16 bytes.

**Figure 9-4 TS output DDR interface**



### 9.1.4.2 Output Format

Table 9-5 TS output options

Format	Program	Usage	Details	Secure Level
TS	One	PVR	One program include: Video PID Audio PIDs Subtitle PIDs PCR_PID EMM/ECM PIDs	Encrypted: 0 Clear content: key level
TS	Two+	PVR	Multiple programs	Encrypted: 0 Clear content: key level
TS, PES, ES, PCR registers, PTS/DTS buffer	One	View	Video buffer Audio buffers Subtitle buffers PCR APB registers PTS/DTS buffer	Video ES: key level Audio ES: key level Subtitle: 0 PCR register: 0 PTS/DTS: 0
TS		PAT, EMM, ECM	One SID per buffer	0

### 9.1.4.3 TS DMA

32 DMA read channels and 128 write channels

Table 9-6 Read channel data structure

Address	Name	Details	Note
0x0	Ready	[0] semaphore	Read 1 to own read channel Write 1 to release read channel
0x4	Debug	Check semaphore	Debug
0x8	RCH_CFG	[28] done [27] enable [26] pause [25] clear [23:16] packet length [15:0] read length	Read length: number of bytes per each round robin
0xC	RCH_ADR	Descriptor address	First descriptor address
0x10	RCH_LEN	Total bytes	
0x14	RCH_RD_LEN	number of bytes read	
0x18	RCH_PTR	Current memory address	



**Table 9-7 Write channel data structure**

Address	Name	Details	Note
0x0	Ready	[0] semaphore	Read 1 to own read channel Write 1 to release read channel
0x4	Debug	Check semaphore	Debug
0x8	WCH_CFG	[25] clear [24] unlimited	
0xC	WCH_ADR	Descriptor address	First descriptor address
0x10	WCH_LEN	Total bytes	
0x14	WCH_RD_LEN	Number of bytes written	
0x18	WCH_PTR	Current memory address	

Read TS stream from DDR memory

Support TS stream packet size 188 or 192, when packet size is 192, the extra 4 bytes between TS packet is removed before sent to TS descrambler.

- Acquire the read channel by read register Ready 0x0 and it is 1.
- Prepare the data descriptor in DDR memory.
- Program the first descriptor address to register RCH\_ADR 0xC.
- Total number of bytes to register RCH\_LEN 0x10.
- Program RCH\_CFG register with packet length, read length and “enable” to 1.
- When done, release Ready by write register 0x0 with 1.

Use similar method to program write channel.

TS DMA APB base address:

APB PADDR[16:14] = 1, APB range 16K bytes

BASE = SECURE\_BASE + 0x4000

**Table 9-8 read/write channel APB address**

APB address	Registers	Note
BASE + RCH_idx * 0x20	32 RCH registers RCH_idx = 0, 1, ..., 31	Refer to read channel registers
BASE + 0x1000 + WCH_idx * 0x20	128 WCH registers WCH_idx = 0, 1, ..., 127	Refer to write channel registers
BASE + 0x2000 + Ctrl_register		

Table 9-9 read/write control registers

Address	R/W	Name	Details	Note
BASE + 0x2000	R/W	RCH_INT_MASK	RCH = 0 ~ 31	1 : Mask RCH/WCH INT 0 : Send INT to ACPU
BASE + 0x2004	R/W	WCH_INT_MASK0	WCH = 0 ~ 31	
BASE + 0x2008	R/W	WCH_INT_MASK1	WCH = 32 ~ 63	
BASE + 0x200c	R/W	WCH_INT_MASK2	WCH = 64 ~ 95	
BASE + 0x2010	R/W	WCH_INT_MASK3	WCH = 96 ~ 127	
BASE + 0x2014	R/W	CLEAR_W_BATCH0		
BASE + 0x2018	R/W	CLEAR_W_BATCH1		
BASE + 0x201c	R/W	CLEAR_W_BATCH2		
BASE + 0x2020	R/W	CLEAR_W_BATCH3		
BASE + 0x2024	R/W	CLEAR_RCH	RCH = 0 ~ 31	1 : Clear RCH/ WCH status
BASE + 0x2028	R/W	CLEAR_WCH0	WCH = 0 ~ 31	
BASE + 0x202c	R/W	CLEAR_WCH1	WCH = 32 ~ 63	
BASE + 0x2030	R/W	CLEAR_WCH2	WCH = 64 ~ 95	
BASE + 0x2034	R/W	CLEAR_WCH3	WCH = 96 ~ 127	
BASE + 0x2038	R	RCH_ACTIVE	RCH = 0 ~ 31	1 : RCH/WCH active
BASE + 0x203c	R	WCH_ACTIVE0	WCH = 0 ~ 31	
BASE + 0x2040	R	WCH_ACTIVE1	WCH = 32 ~ 63	
BASE + 0x2044	R	WCH_ACTIVE2	WCH = 64 ~ 95	
BASE + 0x2048	R	WCH_ACTIVE3	WCH = 96 ~ 127	
BASE + 0x204C	R	RCH_DONE	RCH = 0 ~ 31	1 : RCH descriptor EOC WCH write RESP received
BASE + 0x2050	R	WCH_DONE0	WCH = 0 ~ 31	
BASE + 0x2054	R	WCH_DONE1	WCH = 32 ~ 63	
BASE + 0x2058	R	WCH_DONE2	WCH = 64 ~ 95	
BASE + 0x205c	R	WCH_DONE3	WCH = 96 ~ 127	
BASE + 0x2060	R	RCH_ERR	RCH = 0 ~ 31	No more RCH descriptor
BASE + 0x2064	R	RCH_LEN_ERR	RCH = 0 ~ 31	RCH descriptor length 0
BASE + 0x2068	R	WCH_ERR0	WCH = 0 ~ 31	WCH descriptor length 0
BASE + 0x206C	R	WCH_ERR1	WCH = 32 ~ 63	
BASE + 0x2070	R	WCH_ERR2	WCH = 64 ~ 95	
BASE + 0x2074	R	WCH_ERR3	WCH = 96 ~ 127	
BASE + 0x2078		DMA_BATCH_END0		
BASE + 0x207C		DMA_BATCH_END0		
BASE + 0x2080		DMA_BATCH_END0		

Address	R/W	Name	Details	Note
BASE + 0x2084		DMA_BATCH_END0		
BASE + 0x2088	R	WCH_EOC_DONE0	WCH = 0 ~ 31	WCH descriptor EOC before RESP is received
BASE + 0x208C	R	WCH_EOC_DONE1	WCH = 32 ~ 63	
BASE + 0x2090	R	WCH_EOC_DONE2	WCH = 64 ~ 95	
BASE + 0x2094	R	WCH_EOC_DONE3	WCH = 96 ~ 127	
BASE + 0x2098		UPDT_PKT_SYNC		
BASE + 0x209C		RCH_CFG		
BASE + 0x20A0		WCH_CFG		
BASE + 0x20A4		MEM_PD_CRTL	Memory PD control	
BASE + 0x20A8		DMA_BUS_CFG		
BASE + 0x20AC		DMA_GMW_CFG		
BASE + 0x20B0		DMA_GMR_CFG		

## 9.1.5 HDMIRX

### 9.1.5.1 HDMIRX TOP-Level Registers

#### 9.1.5.1.1 Register Access

To access Top-level normal registers, use type `uint32_t`,

To access Top-Level EDID RAM, use type `uint8_t`:

```
#define HDMIRX_TOP_OFFSET0xfe398000
#define HDMIRX_TOP_EDID_ADDR0_S0x1000
#define HDMIRX_TOP_EDID_ADDR2_E0x15ff

void hdmirx_wr_only_TOP (uint32_t addr, uint32_t data)
{
    if ((addr >= HDMIRX_TOP_EDID_ADDR0_S) && (addr <= HDMIRX_TOP_EDID_ADDR2_E)) {
        *((volatile uint8_t*)(HDMIRX_TOP_OFFSET+addr)) = (uint8_t)(data&0xff);
    } else {
        *((volatile uint32_t*)(HDMIRX_TOP_OFFSET+addr)) = (data);
    }
}

uint32_t hdmirx_rd_TOP (uint8_t int_ext, uint32_t addr)
{
    uint32_t data;
    if ((addr >= HDMIRX_TOP_EDID_ADDR0_S) && (addr <= HDMIRX_TOP_EDID_ADDR2_E)) {
        data = (uint32_t)*((volatile uint8_t*)(HDMIRX_TOP_OFFSET+addr));
    } else {
        data = *((volatile uint32_t*)(HDMIRX_TOP_OFFSET+addr));
    }
    return (data);
}
```

#### 9.1.5.1.2 Register Description

##### Register Address

For below registers the base address is 0xfe398000.

Each register final address = BASE + address \* 4.

The following lists describe the mapping between each register and its address.

Address	Register	RW	Function
0x000<<2	HDMIRX_TOP_SW_RESET	RW	Software reset sub-modules.
0x001<<2	HDMIRX_TOP_CLK_CNTL	RW	Clock gating and inversion.
0x002<<2	HDMIRX_TOP_HPD_PWR5V	RW	HPD control and PWR5V status.
0x003<<2	HDMIRX_TOP_PORT_SEL	RW	RX input port selection.
0x004<<2	HDMIRX_TOP_EDID_GEN_CNTL	RW	EDID control.
0x005<<2	HDMIRX_TOP_EDID_ADDR_CEC	RW	EDID's CEC bytes auto-replace.
0x006<<2	HDMIRX_TOP_EDID_DATA_CEC_PORT01	RW	EDID's CEC bytes auto-replace.
0x007<<2	HDMIRX_TOP_EDID_DATA_CEC_PORT2	RW	EDID's CEC bytes auto-replace.
0x008<<2	HDMIRX_TOP_EDID_GEN_STAT_A	RW	PortA EDID status.
0x009<<2	HDMIRX_TOP_INTR_MASKN	RW	Interrupt mask.
0x00A<<2	HDMIRX_TOP_INTR_STAT	RW	Interrupt status.
0x00B<<2	HDMIRX_TOP_INTR_STAT_CLR	RW	Interrupt clear.
0x00C<<2	HDMIRX_TOP_VID_CNTL	RW	Video processing control.
0x00D<<2	HDMIRX_TOP_VID_STAT	R	Video processing status.
0x00E<<2	HDMIRX_TOP_ACR_CNTL_STAT	RW	ACR control and status.
0x011<<2	HDMIRX_TOP_METER_HDMI_CNTL	RW	tmds_clk meter control.
0x012<<2	HDMIRX_TOP_METER_HDMI_STAT	R	tmds_clk meter status.
0x013<<2	HDMIRX_TOP_VID_CNTL2	RW	Video processing control.
0x015<<2	HDMIRX_TOP_EDID_RAM_OVR0	RW	EDID RAM override byte0 address.
0x016<<2	HDMIRX_TOP_EDID_RAM_OVR0_DATA	RW	EDID RAM override byte0 data.
0x017<<2	HDMIRX_TOP_EDID_RAM_OVR1	RW	EDID RAM override byte1 address.
0x018<<2	HDMIRX_TOP_EDID_RAM_OVR1_DATA	RW	EDID RAM override byte1 data.
0x019<<2	HDMIRX_TOP_EDID_RAM_OVR2	RW	EDID RAM override byte2 address.
0x01A<<2	HDMIRX_TOP_EDID_RAM_OVR2_DATA	RW	EDID RAM override byte2 data.
0x01B<<2	HDMIRX_TOP_EDID_RAM_OVR3	RW	EDID RAM override byte3 address.
0x01C<<2	HDMIRX_TOP_EDID_RAM_OVR3_DATA	RW	EDID RAM override byte3 data.
0x01D<<2	HDMIRX_TOP_EDID_RAM_OVR4	RW	EDID RAM override byte4 address.

Address	Register	RW	Function
0x01E<<2	HDMIRX_TOP_EDID_RAM_OVR4_DATA	RW	EDID RAM override byte4 data.
0x01F<<2	HDMIRX_TOP_EDID_RAM_OVR5	RW	EDID RAM override byte5 address.
0x020<<2	HDMIRX_TOP_EDID_RAM_OVR5_DATA	RW	EDID RAM override byte5 data.
0x021<<2	HDMIRX_TOP_EDID_RAM_OVR6	RW	EDID RAM override byte6 address.
0x022<<2	HDMIRX_TOP_EDID_RAM_OVR6_DATA	RW	EDID RAM override byte6 data.
0x023<<2	HDMIRX_TOP_EDID_RAM_OVR7	RW	EDID RAM override byte7 address.
0x024<<2	HDMIRX_TOP_EDID_RAM_OVR7_DATA	RW	EDID RAM override byte7 data.
0x025<<2	HDMIRX_TOP_EDID_GEN_STAT_B	RW	PortB EDID status.
0x026<<2	HDMIRX_TOP_EDID_GEN_STAT_C	RW	PortC EDID status.
0x028<<2	HDMIRX_TOP_CHAN_SWITCH_0	RW	Data channel input control.
0x02B<<2	HDMIRX_TOP_TMDS_ALIGN_STAT	R	Data channel alignment status.
0x02C<<2	HDMIRX_TOP_INFILTER_HDCP	RW	HDCP and SCDC input glitch filter control.
0x02D<<2	HDMIRX_TOP_INFILTER_I2C_0	RW	DDC SCL/SDA input glitch filter control for port 0.
0x02E<<2	HDMIRX_TOP_INFILTER_I2C_1	RW	DDC SCL/SDA input glitch filter control for port 1.
0x02F<<2	HDMIRX_TOP_INFILTER_I2C_2	RW	DDC SCL/SDA input glitch filter control for port 2.
0x033<<2	HDMIRX_TOP_PRBS_GEN	RW	PRBS generator control.
0x034<<2	HDMIRX_TOP_PRBS_ANA_0	RW	PRBS analyzer control.
0x035<<2	HDMIRX_TOP_PRBS_ANA_1	RW	PRBS analyzer control.
0x036<<2	HDMIRX_TOP_PRBS_ANA_STAT	R	PRBS analyzer status.
0x037<<2	HDMIRX_TOP_PRBS_ANA_BER_CH0	R	PRBS analyzer BER count for data channel 0.
0x038<<2	HDMIRX_TOP_PRBS_ANA_BER_CH1	R	PRBS analyzer BER count for data channel 1.
0x039<<2	HDMIRX_TOP_PRBS_ANA_BER_CH2	R	PRBS analyzer BER count for data channel 2.
0x03A<<2	HDMIRX_TOP_METER_CABLE_CNTL	RW	cable_clk meter control.
0x03B<<2	HDMIRX_TOP_METER_CABLE_STAT	R	cable_clk meter status.

Address	Register	RW	Function
0x03C<<2	HDMIRX_TOP_CHAN_SWITCH_1	RW	Data channel input control.
0x040<<2	HDMIRX_TOP_AUDPLL_LOCK_FILTER	RW	Aud_pll lock input filter.
0x041<<2	HDMIRX_TOP_CHAN01_ERRCNT	R	Character error counter for data channel 0 and 1.
0x042<<2	HDMIRX_TOP_CHAN2_ERRCNT	R	Character error counter for data channel 2.
0x043<<2	HDMIRX_TOP_ACR_CNTL2	RW	ACR control.
0x044<<2	HDMIRX_TOP_ACR_N_STAT	R	Read back N value to AUD_PLL.
0x045<<2	HDMIRX_TOP_ACR_CTS_STAT	R	Read back CTS value to AUD_PLL.
0x046<<2	HDMIRX_TOP_AUDMEAS_CTRL	RW	Audio clock measure control.
0x047<<2	HDMIRX_TOP_AUDMEAS_CYCLES_M1	RW	Audio clock measure period length.
0x048<<2	HDMIRX_TOP_AUDMEAS_INTR_MASKN	RW	Audio clock measure interrupt mask.
0x049<<2	HDMIRX_TOP_AUDMEAS_INTR_STAT	RW	Audio clock measure interrupt status.
0x04A<<2	HDMIRX_TOP_AUDMEAS_REF_CYCLES_STAT0	R	Audio clock measure result.
0x04B<<2	HDMIRX_TOP_AUDMEAS_REF_CYCLES_STAT1	R	Audio clock measure result.
0x050<<2	HDMIRX_TOP_SHIFT_PTTN_012	RW	Shift pattern BIST.
0x051<<2	HDMIRX_TOP_SHIFT_PTTN_345	RW	Shift pattern BIST.
0x052<<2	HDMIRX_TOP_SHIFT_PTTN_678	RW	Shift pattern BIST.
0x053<<2	HDMIRX_TOP_SHIFT_PTTN_9	RW	Shift pattern BIST.
0x054<<2	HDMIRX_TOP_SHFT_ANA_CNTL	RW	Shift pattern BIST.
0x055<<2	HDMIRX_TOP_SHFT_ANA_STAT	R	Shift pattern BIST.
0x056<<2	HDMIRX_TOP_SHFT_PTTN_RCV_012	R	Shift pattern BIST.
0x057<<2	HDMIRX_TOP_SHFT_PTTN_RCV_345	R	Shift pattern BIST.
0x058<<2	HDMIRX_TOP_SHFT_PTTN_RCV_678	R	Shift pattern BIST.
0x059<<2	HDMIRX_TOP_SHFT_PTTN_RCV_9	R	Shift pattern BIST.
0x06E<<2	HDMIRX_TOP_NSEC_SCRATCH	RW	Scratch register for non-secure access.

Address	Register	RW	Function
0x06F<<2	HDMIRX_TOP_SEC_SCRATCH	RW	Scratch register for secure access.
0x070<<2	HDMIRX_TOP_EMP_DDR_START_A	RW	DDR pointer A to store EMP.
0x071<<2	HDMIRX_TOP_EMP_DDR_START_B	RW	DDR pointer B to store EMP.
0x072<<2	HDMIRX_TOP_EMP_DDR_FILTER	RW	Packet filter to go to DDR.
0x073<<2	HDMIRX_TOP_EMP_CNTL_0	RW	EMP to DDR control.
0x074<<2	HDMIRX_TOP_EMP_CNTL_1	RW	EMP to DDR control.
0x075<<2	HDMIRX_TOP_EMP_CNTMAX	RW	Maximum number of packets to store in DDR per frame.
0x076<<2	HDMIRX_TOP_EMP_ERR_STAT	RW	EMP error status.
0x077<<2	HDMIRX_TOP_EMP_RCV_CNT_CUR	R	Current number of packet received.
0x078<<2	HDMIRX_TOP_EMP_RCV_CNT_BUF	R	Buffered number of packet received.
0x079<<2	HDMIRX_TOP_EMP_DDR_ADDR_CUR	R	Current DDR pointer for EMP.
0x07A<<2	HDMIRX_TOP_EMP_DDR_PTR_S_BUF	R	Buffered DDR pointer for EMP.
0x07B<<2	HDMIRX_TOP_EMP_STAT_0	R	EMP status.
0x07C<<2	HDMIRX_TOP_EMP_STAT_1	R	EMP status.
0x080<<2	HDMIRX_TOP_PHYIF_CNTL0	RW	MISC PHY signal control
0x081<<2	HDMIRX_TOP_FIX_DISABLE_0	RW	Fix IP internal domain crossing bug.
0x084<<2	HDMIRX_TOP_MISC_STAT0	R	MISC status
0x090<<2	HDMIRX_TOP_OVID_OVERRIDE0	RW	Override hdmirx video output.
0x091<<2	HDMIRX_TOP_OVID_OVERRIDE1	RW	Override hdmirx video output.
0x0a0<<2	HDMIRX_TOP_SECURE_INDEX	RW	Secure core register bank start index.
0x0a1<<2	HDMIRX_TOP_SECURE_DATA	RW	Secure core register bank content.
0x0a2<<2	HDMIRX_TOP_SECURE_MODE	RW	Control if core register can be accessed.
0x1000-0x11ff	HDMIRX_TOP_EDID_ADDR0_S- HDMIRX_TOP_EDID_ADDR0_E	RW	For Host to program port0's EDID RAM.

Address	Register	RW	Function
0x1200-0x13ff	HDMIRX_TOP_EDID_ADDR1_S- HDMIRX_TOP_EDID_ADDR1_E	RW	For Host to program port1's EDID RAM.
0x1400-0x15ff	HDMIRX_TOP_EDID_ADDR2_S- HDMIRX_TOP_EDID_ADDR2_E	RW	For Host to program port2's EDID RAM.

## Register Description

**Table 9-10 HDMIRX\_TOP\_SW\_RESET**

Bit	R/W	Default	Description
31:16	R	0	Reserved
15	RW	1	sw_reset_audmeas: to reset audio clock measure module. 0=Release from reset; 1=Apply reset.
13	RW	1	sw_reset_emp: to reset EMP module. 0=Release from reset; 1=Apply reset.
11	RW	1	sw_reset_fit: to reset DDC input glitch filter. 0=Release from reset; 1=Apply reset.
8	RW	1	sw_reset_bist: to reset BIST module. 0=Release from reset; 1=Apply reset.
7	RW	1	sw_reset_chan: to reset channel_switch module. 0=Release from reset; 1=Apply reset.
6	RW	1	sw_reset_cable_meter: to reset cable_clk meter. 0=Release from reset; 1=Apply reset.
5	RW	1	sw_reset_hdmi_meter: to reset hdmi_clk meter. 0=Release from reset; 1=Apply reset.
4	RW	1	sw_reset_acr: to reset ACR module. 0=Release from reset; 1=Apply reset.
3	RW	1	sw_reset_vid: to reset video processing path. 0=Release from reset; 1=Apply reset.
2	RW	1	sw_reset_intr: to reset interrupt block. 0=Release from reset; 1=Apply reset.
1	RW	1	sw_reset_edid: to reset EDID interface. 0=Release from reset; 1=Apply reset.
0	RW	1	sw_reset_core: To reset RX Controller IP. 0=Release from reset; 1=Apply reset.



**Table 9-11 HDMIRX\_TOP\_CLK\_CNTL**

Bit	R/W	Default	Description
31	RW	0	free_clk_en: 0= Enable clock gating for power saving; 1= Disable clock gating, enable free-run clock.
30:18	R	0	Reserved
15	RW	0	hbr_spdif_en: 1= Select HBR audio through SPDIF not I2S.
14:13	RW	0	Reserved
8	RW	0	tmads_ch2_clk_inv: 1= Invert tmads_ch2_clk.
7	RW	0	tmads_ch1_clk_inv: 1= Invert tmads_ch1_clk.
6	RW	0	tmads_ch0_clk_inv: 1= Invert tmads_ch0_clk.
5	RW	0	pll4x_cfg
4	RW	0	force_pll4x: 1=Force pll4x_en value to be pll4x_cfg. 0=Use auto detect.
3	RW	0	phy_clk_inv: 1= Invert phy_clk.
2:0	R	0	Reserved

**Table 9-12 HDMIRX\_TOP\_HPD\_PWR5V**

Bit	R/W	Default	Description
31:23	R	0	Reserved
22	R	0	stat_5v_portC: Status of 5v power for port C.
21	R	0	stat_5v_portB: Status of 5v power for port B.
20	R	0	stat_5v_portA: Status of 5v power for port A.
19	R	0	Reserved
18	R	0	stat_hpd_portC: Status of IP's own HPD value for port C.
17	R	0	stat_hpd_portB: Status of IP's own HPD value for port B.
16	R	0	stat_hpd_portA: Status of IP's own HPD value for port A.
15:6	R	0	Reserved
5	RW	0	invert_hpd: 1=Invert the HPD value before final output; 0=No invert.
4	RW	1	force_hpd: 1=Select hpd_config as HPD value; 0=Select IP's own value as HPD.

Bit	R/W	Default	Description
3	R	0	Reserved
2:0	RW	0	hpd_config: HPD values when force_hpd=1, bit0 for port A, bit1 for port B, bit2 for port C.

**Table 9-13 HDMIRX\_TOP\_PORT\_SEL**

Bit	R/W	Default	Description
31:5	R	0	Reserved
4	RW	0	multi_edid: 1=Enable all ports available for EDID transaction; 0=Only the selected port can do EDID.
2:0	RW	0	port_sel: HDMI port select for EDID. 0001=Select port A; 0010=Select port B; 0100=Select port C.

**Table 9-14 HDMIRX\_TOP\_EDID\_GEN\_CNTL**

Bit	R/W	Default	Description
31:16	RW	0	Reserved
15:13	RW	0	i2c_filter_enable[2:0]. One bit for each port. 1=Enable glitch filter on SCL and SDA; 0=No glitch filter, same as M6TV.
12	R	0	Reserved
11	RW	0	auto_cec_enable: Control whether to auto-replace the port-related CEC information within EDID info. 1=Auto-replace data byte with edid_byte0_cec_portA/B/C/D, if the byte's location matches edid_addr0_cec, and port matches A/B/C/D; and auto-replace data byte with edid_byte1_cec_portA/B/C/D, if the byte's location matches edid_addr1_cec, and port matches A/B/C/D; 0=Send EDID data exactly as filled in RAM.
10	RW	0	scl_stretch_trigger_config: 1=Manual trigger for clock stretch, applicable when force_scl_stretch_trigger=1. 0=No trigger.
9	RW	0	force_scl_stretch_trigger: Select the source to trigger start clock-stretch. 1=Manual trigger of scl_stretch_trigger_config=1; 0=Auto trigger on every 2 EDID blocks sent.
8	RW	0	scl_stretch_enable: 1=Enable I2C clock-stretch feature to suspend I2C when Host need time to fill new EDID data into RAM during I2C transaction; 0=Disable I2C clock-stretch feature.
7:0	RW	0	clk_divide_m1: Divide system clock down for EDID I/F clock, no need to divide down, but can divide down if for power saving purpose.

**Table 9-15 HDMIRX\_TOP\_EDID\_ADDR\_CEC**

Bit	R/W	Default	Description
31:16	RW	0x99	edid_addr1_cec: byte1 address for auto-replacing EDID's CEC info.
15:0	RW	0x98	edid_addr0_cec: byte0 address for auto-replacing EDID's CEC info.

**Table 9-16 HDMIRX\_TOP\_EDID\_DATA\_CEC\_PORT01**

Bit	R/W	Default	Description
31:24	RW	0	edid_byte1_cec_portB: byte1 data for auto-replacing EDID's CEC info.
23:16	RW	0	edid_byte0_cec_portB: byte0 data for auto-replacing EDID's CEC info.
15:8	RW	0	edid_byte1_cec_portA: byte1 data for auto-replacing EDID's CEC info.
7:0	RW	0	edid_byte0_cec_portA: byte0 data for auto-replacing EDID's CEC info.

**Table 9-17 HDMIRX\_TOP\_EDID\_DATA\_CEC\_PORT2**

Bit	R/W	Default	Description
15:8	RW	0	edid_byte1_cec_portC: byte1 data for auto-replacing EDID's CEC info.
7:0	RW	0	edid_byte0_cec_portC: byte0 data for auto-replacing EDID's CEC info.

**Table 9-18 HDMIRX\_TOP\_EDID\_GEN\_STAT\_A**

EDID status for portA.

Bit	R/W	Default	Description
31:25	R	0	Reserved
24:22	R	0	i2c_state: Status of I2C transaction. 0=IDLE; 1=SEND_BYTE; 2=GET_ACK; 3=READ_BYTE; 4=PUT_ACK.
21:20	R	0	slave_state: Status of I2C slave. 0=IDLE; 1=OFFSET, read offset address; 2=SEGMENT, read segment pointer; 3=SEND, sending data.
19:17	R	0	Reserved
16	RW	0	edid_addr_intr_stat_clr: This bit read back 1 after 2-block EDID data were sent, and there are further block to be sent. Write this bit 1 to clear this bit.
15:0	R	0	edid_addr: Status of current EDID address.

**Table 9-19 HDMIRX\_TOP\_INTR\_MASKN**

Interrupt MASKN, one bit per interrupt source. 0= Disable interrupt source; 1= Enable interrupt source.

Bit	R/W	Default	Description
31:30	R	0	Reserved
29	RW	0	hdmirx_sqofclk_fall
28	RW	0	hdmirx_sqofclk_rise
27	RW	0	de_rise_irq: DE rise edge.
26	RW	0	last_emp_done: For EMP-to-DDR mode, this bit means last EMP packet received; For TMDS-to-DDR mode, this bit means last TMDS character received.
25	RW	0	emp_field_done: EMP receive done for last field. Note: EMP field end at DE rise.
24	RW	0	Reserved
23	RW	0	meter_stable_chg_cable: cable_clk measure stable/unstable status change
22:20	RW	0	Reserved
19	RW	0	edid_addr_intr for the port C
18	RW	0	edid_addr_intr for the port B
17	RW	0	edid_addr_intr for the port A
16	RW	0	hdcp_enc_state_fall
15	RW	0	hdcp_enc_state_rise
14	RW	0	hdcp_auth_start_fall
13	RW	0	hdcp_auth_start_rise
12	RW	0	meter_stable_chg_hdmi: tmds_clk measure stable/unstable status change
11	RW	0	vid_colour_depth_chg: video data color_depth change
10	RW	0	vid_fmt_chg: video data format change
9	RW	0	Reserved
8	RW	0	Port C 5v fall
7	RW	0	Port B 5v fall
6	RW	0	Port A 5v fall
5	RW	0	Port C 5v rise
4	RW	0	Port B 5v rise
3	RW	0	Port A 5v rise
2	RW	0	RX PHY digital interrupt.
1	RW	0	RX Controller pwd interrupt.
0	RW	0	RX Controller aon interrupt.

**Table 9-20 HDMIRX\_TOP\_INTR\_STAT**

Interrupt status. For each bit of bit[29:3], write 1 to manually set the interrupt bit, read back the interrupt status.

Bit	R/W	Default	Description
31:30	RW	0	Reserved
29	RW	0	hdmirx_sqofclk_fall
28	RW	0	hdmirx_sqofclk_rise
27	RW	0	de_rise_irq: DE rise edge.
26	RW	0	last_emp_done: For EMP-to-DDR mode, this bit means last EMP packet received; For TMDS-to-DDR mode, this bit means last TMDS character received.
25	RW	0	emp_field_done: EMP receive done for last field. Note: EMP field end at DE rise.
24	RW	0	Reserved
23	RW	0	meter_stable_chg_cable: cable_clk measure stable/unstable status change
22:20	RW	0	Reserved
19	RW	0	edid_addr_intr for the port C
18	RW	0	edid_addr_intr for the port B
17	RW	0	edid_addr_intr for the port A
16	RW	0	hdcp_enc_state_fall
15	RW	0	hdcp_enc_state_rise
14	RW	0	hdcp_auth_start_fall
13	RW	0	hdcp_auth_start_rise
12	RW	0	meter_stable_chg_hdmi: tmds_clk measure stable/unstable status change
11	RW	0	vid_colour_depth_chg: video data color_depth change
10	RW	0	vid_fmt_chg: video data format change
9	RW	0	Reserved
8	RW	0	Port C 5v fall
7	RW	0	Port B 5v fall
6	RW	0	Port A 5v fall
5	RW	0	Port C 5v rise
4	RW	0	Port B 5v rise
3	RW	0	Port A 5v rise
2	RW	0	RX PHY digital interrupt.
1	RW	0	RX Controller pwr interrupt.
0	RW	0	RX Controller aon interrupt.

**Table 9-21 HDMIRX\_TOP\_INTR\_STAT\_CLR**

Interrupt status clear. For each bit[29:3], write 1 to clear the interrupt bit.

Bit	R/W	Default	Description
31:30	R	0	Reserved
29	W	0	hdmirx_sqofclk_fall
28	W	0	hdmirx_sqofclk_rise
27	W	0	de_rise_irq: DE rise edge.
26	W	0	last_emp_done: For EMP-to-DDR mode, this bit means last EMP packet received; For TMDS-to-DDR mode, this bit means last TMDS character received.
25	W	0	emp_field_done: EMP receive done for last field. Note: EMP field end at DE rise.
24	W	0	Reserved
23	W	0	meter_stable_chg_cable: cable_clk measure stable/unstable status change
22:20	W	0	Reserved
19	W	0	edid_addr_intr for the port C
18	W	0	edid_addr_intr for the port B
17	W	0	edid_addr_intr for the port A
16	W	0	hdcp_enc_state_fall
15	W	0	hdcp_enc_state_rise
14	W	0	hdcp_auth_start_fall
13	W	0	hdcp_auth_start_rise
12	W	0	meter_stable_chg_hdmi: tmds_clk measure stable/unstable status change
11	W	0	vid_colour_depth_chg: video data color_depth change
10	W	0	vid_fmt_chg: video data format change
9	W	0	Reserved
8	W	0	Port C 5v fall
7	W	0	Port B 5v fall
6	W	0	Port A 5v fall
5	W	0	Port C 5v rise
4	W	0	Port B 5v rise
3	W	0	Port A 5v rise
2		0	RX PHY digital interrupt.
1		0	RX Controller pwd interrupt.
0		0	RX Controller aon interrupt.

Table 9-22 HDMIRX\_TOP\_VID\_CNTL

Bit	R/W	Default	Description
31:29	RW	0	Reserved
28	RW	0	vs_timing: 0=input VS rising edge to generate output VS pulse; 1=input VS falling edge to generate output VS pulse.

Bit	R/W	Default	Description
27	RW	0	hs_timing: 0=input HS rising edge to generate output HS pulse; 1=input HS falling edge to generate output HS pulse.
26:24	RW	0	data_map: Re-map component data to different position. 0={vid2, vid1, vid0} 1={vid1, vid2, vid0} 2={vid1, vid0, vid2} 3={vid0, vid1, vid2} 4={vid0, vid2, vid1} 5={vid2, vid0, vid1}
23:20	RW	0	Reserved
19	RW	0	mode_422: 1=Average neighbor pixels to interpolate to 444; 0=Repeat previous pixel to interpolate to 444.
18	RW	0	Reserved
17	RW	0	dbv422_mode: 1=For DolbyVision 422 mode, pass through data to downstream video path without converting to 444.
11	RW	0	vid_fmt_override. 0=Use automatic detected vid_fmt; 1=Use vid_fmt_val.
10:8	RW	0	vid_fmt_val. 0=RGB; 1=YUV422; 2=YUV444; 3=YUV420.
7	RW	0	pixel_repeat_override. Always set to 0. 0=Use automatic detected pixel_repeat; 1=Use pixel_repeat_cfg.
6:3	RW	0	pixel_repeat_cfg
2	RW	0	color_depth_override. 0=Use automatic detected color_depth; 1=Use color_depth_cfg.
1:0	RW	0	color_depth_cfg. 0=8-bit; 1=10-bit; 2=12-bit; 3=16-bit.

Table 9-23 HDMIRX\_TOP\_VID\_STAT

Bit	R/W	Default	Description
31:24	R	0	Reserved
23:20	R	0	vid_fmt: input video data format. 0=RGB; 1=YUV422; 2=YUV444; 3=YUV420.

Bit	R/W	Default	Description
19:16	R	0	vid_colour_depth: input video data color_depth. 0-3=Reserved; 4=24-bit per pixel; 5=30-bit; 6=36-bit; 7=48-bit; 8-15=Reserved.
15:0	R	0	window_start_pix: Applicable to 420 mode only. The measure input data line's position from HS rise to DE rise, minus cntl_last420l_rdwin_auto.

Table 9-24 HDMIRX\_TOP\_ACR\_CNTL\_STAT

Bit	R/W	Default	Description
28:27	R	0	Reserved
15:13	RW	0	n_cts_update_width: 000=hdmirx_n_cts_update pulse width is 1 tmds_clk cycle; 001=2-cycle; 011=3-cycle; 111=4-cycle.
12	RW	0	n_cts_update_on_pll4x: 1=aud_pll4x_en change will result in a hdmirx_n_cts_update pulse; 0=No hdmirx_n_cts_update pulse on aud_pll4x_en change.
11	RW	0	n_cts_update_man: Write 1 to manually generate an hdmirx_n_cts_update pulse. This bit self-clears.
10	RW	0	pll_tmode_cntl: Set to 1 for production testing audio ACR PLL. Normal working mode set to 0.

Table 9-25 HDMIRX\_TOP\_METER\_HDMI\_CNTL

Bit	R/W	Default	Description
31:28	RW	0x1	meas_tolerance
27:24	RW	0	Reserved
23:0	RW	0x80	ref_cycles

Table 9-26 HDMIRX\_TOP\_METER\_HDMI\_STAT

Bit	R/W	Default	Description
31	R	0	meas_stable
30:24	R	0	Reserved
23:0	R	0	meas_cycles. $\text{Freq}(\text{tmds\_clk}) = \text{Freq}(\text{ref\_clk}) * \text{meas\_cycles} / \text{ref\_cycles}$



**Table 9-27 HDMIRX\_TOP\_VID\_CNTL2**

Bit	R/W	Default	Description
31:21	RW	0	Reserved
20	RW	0	last420l_rdwin_mode: Control when to start read from FIFO for outputting 420 even or last line. 0=End Of Active Video by detecting no DE cntl_last420l_rdwin_auto + window_start_pix clock cycles after go_line. 1=End of Active Video after cntl_last420l_rdwin_manual active line passed.
19:8	RW	0	last420l_rdwin_manual
7:0	RW	0xa	last420l_rdwin_auto

**Table 9-28 HDMIRX\_TOP\_EDID\_RAM\_OVR0**

Bit	R/W	Default	Description
31:19	R	0	Reserved
18:16	RW	0	edid_ram_ovr0_en: For each bit, 1=Enable override the data read from EDID RAM address edid_ram_ovr0_addr, with the data defined in HDMIRX_EDID_RAM_OVR0_DATA. Bit[0] for portA, bit[1] for portB, bit[2] for portC.
15:0	RW	0	edid_ram_ovr0_addr

**Table 9-29 HDMIRX\_TOP\_EDID\_RAM\_OVR0\_DATA**

Bit	R/W	Default	Description
23:16	RW	0	edid_ram_ovr0_data for portC
15:8	RW	0	edid_ram_ovr0_data for portB
7:0	RW	0	edid_ram_ovr0_data for portA

**Table 9-30 HDMIRX\_TOP\_EDID\_RAM\_OVR1**

Bit	R/W	Default	Description
31:19	R	0	Reserved
18:16	RW	0	edid_ram_ovr1_en: For each bit, 1=Enable override the data read from EDID RAM address edid_ram_ovr1_addr, with the data defined in HDMIRX_EDID_RAM_OVR1_DATA. Bit[0] for portA, bit[1] for portB, bit[2] for portC.
15:0	RW	0	edid_ram_ovr1_addr

**Table 9-31 HDMIRX\_TOP\_EDID\_RAM\_OVR1\_DATA**

Bit	R/W	Default	Description
23:16	RW	0	edid_ram_ovr1_data for portC
15:8	RW	0	edid_ram_ovr1_data for portB
7:0	RW	0	edid_ram_ovr1_data for portA

**Table 9-32 HDMIRX\_TOP\_EDID\_RAM\_OVR2**

Bit	R/W	Default	Description
31:19	R	0	Reserved
18:16	RW	0	edid_ram_ovr2_en: For each bit, 1=Enable override the data read from EDID RAM address edid_ram_ovr2_addr, with the data defined in HDMIRX_EDID_RAM_OVR2_DATA. Bit[0] for portA, bit[1] for portB, bit[2] for portC.
15:0	RW	0	edid_ram_ovr2_addr

**Table 9-33 HDMIRX\_TOP\_EDID\_RAM\_OVR2\_DATA**

Bit	R/W	Default	Description
23:16	RW	0	edid_ram_ovr2_data for portC
15:8	RW	0	edid_ram_ovr2_data for portB
7:0	RW	0	edid_ram_ovr2_data for portA

**Table 9-34 HDMIRX\_TOP\_EDID\_RAM\_OVR3**

Bit	R/W	Default	Description
31:19	R	0	Reserved
18:16	RW	0	edid_ram_ovr3_en: For each bit, 1=Enable override the data read from EDID RAM address edid_ram_ovr3_addr, with the data defined in HDMIRX_EDID_RAM_OVR3_DATA. Bit[0] for portA, bit[1] for portB, bit[2] for portC.
15:0	RW	0	edid_ram_ovr3_addr

**Table 9-35 HDMIRX\_TOP\_EDID\_RAM\_OVR3\_DATA**

Bit	R/W	Default	Description
23:16	RW	0	edid_ram_ovr3_data for portC
15:8	RW	0	edid_ram_ovr3_data for portB
7:0	RW	0	edid_ram_ovr3_data for portA

**Table 9-36 HDMIRX\_TOP\_EDID\_RAM\_OVR4**

Bit	R/W	Default	Description
31:19	R	0	Reserved
18:16	RW	0	edid_ram_ovr4_en: For each bit, 1=Enable override the data read from EDID RAM address edid_ram_ovr4_addr, with the data defined in HDMIRX_EDID_RAM_OVR4_DATA. Bit[0] for portA, bit[1] for portB, bit[2] for portC.
15:0	RW	0	edid_ram_ovr4_addr

**Table 9-37 HDMIRX\_TOP\_EDID\_RAM\_OVR4\_DATA**

Bit	R/W	Default	Description
23:16	RW	0	edid_ram_ovr4_data for portC
15:8	RW	0	edid_ram_ovr4_data for portB
7:0	RW	0	edid_ram_ovr4_data for portA

**Table 9-38 HDMIRX\_TOP\_EDID\_RAM\_OVR5**

Bit	R/W	Default	Description
31:19	R	0	Reserved
18:16	RW	0	edid_ram_ovr5_en: For each bit, 1=Enable override the data read from EDID RAM address edid_ram_ovr5_addr, with the data defined in HDMIRX_EDID_RAM_OVR5_DATA. Bit[0] for portA, bit[1] for portB, bit[2] for portC.
15:0	RW	0	edid_ram_ovr5_addr

**Table 9-39 HDMIRX\_TOP\_EDID\_RAM\_OVR5\_DATA**

Bit	R/W	Default	Description
23:16	RW	0	edid_ram_ovr5_data for portC
15:8	RW	0	edid_ram_ovr5_data for portB
7:0	RW	0	edid_ram_ovr5_data for portA

**Table 9-40 HDMIRX\_TOP\_EDID\_RAM\_OVR6**

Bit	R/W	Default	Description
31:19	R	0	Reserved
18:16	RW	0	edid_ram_ovr6_en: For each bit, 1=Enable override the data read from EDID RAM address edid_ram_ovr6_addr, with the data defined in HDMIRX_EDID_RAM_OVR6_DATA. Bit[0] for portA, bit[1] for portB, bit[2] for portC.
15:0	RW	0	edid_ram_ovr6_addr

**Table 9-41 HDMIRX\_TOP\_EDID\_RAM\_OVR6\_DATA**

Bit	R/W	Default	Description
23:16	RW	0	edid_ram_ovr6_data for portC
15:8	RW	0	edid_ram_ovr6_data for portB
7:0	RW	0	edid_ram_ovr6_data for portA

**Table 9-42 HDMIRX\_TOP\_EDID\_RAM\_OVR7**

Bit	R/W	Default	Description
31:19	R	0	Reserved
18:16	RW	0	edid_ram_ovr7_en: For each bit, 1=Enable override the data read from EDID RAM address edid_ram_ovr7_addr, with the data defined in HDMIRX_EDID_RAM_OVR7_DATA. Bit[0] for portA, bit[1] for portB, bit[2] for portC.
15:0	RW	0	edid_ram_ovr7_addr

**Table 9-43 HDMIRX\_TOP\_EDID\_RAM\_OVR7\_DATA**

Bit	R/W	Default	Description
23:16	RW	0	edid_ram_ovr7_data for portC
15:8	RW	0	edid_ram_ovr7_data for portB
7:0	RW	0	edid_ram_ovr7_data for portA

**Table 9-44 HDMIRX\_TOP\_EDID\_GEN\_STAT\_B**

EDID status for portB.

Bit	R/W	Default	Description
31:25	R	0	Reserved
24:22	R	0	i2c_state: Status of I2C transaction. 0=IDLE; 1=SEND_BYTE; 2=GET_ACK; 3=READ_BYTE; 4=PUT_ACK.
21:20	R	0	slave_state: Status of I2C slave. 0=IDLE; 1=OFFSET, read offset address; 2=SEGMENT, read segment pointer; 3=SEND, sending data.
19:17	R	0	Reserved
16	RW	0	edid_addr_intr_stat_clr: This bit read back 1 after 2-block EDID data were sent, and there are further block to be sent. Write this bit 1 to clear this bit.
15:0	R	0	edid_addr: Status of current EDID address.

**Table 9-45 HDMIRX\_TOP\_EDID\_GEN\_STAT\_C**

EDID status for portC.

Bit	R/W	Default	Description
31:25	R	0	Reserved
24:22	R	0	i2c_state: Status of I2C transaction. 0=IDLE; 1=SEND_BYTE; 2=GET_ACK;

Bit	R/W	Default	Description
			3=READ_BYTE; 4=PUT_ACK.
21:20	R	0	slave_state: Status of I2C slave. 0=IDLE; 1=OFFSET, read offset address; 2=SEGMENT, read segment pointer; 3=SEND, sending data.
19:17	R	0	Reserved
16	RW	0	edid_addr_intr_stat_clr: This bit read back 1 after 2-block EDID data were sent, and there are further block to be sent. Write this bit 1 to clear this bit.
15:0	R	0	edid_addr: Status of current EDID address.

Table 9-46 HDMIRX\_TOP\_CHAN\_SWITCH\_0

Bit	R/W	Default	Description
29:28	RW	0	source_chan2: 0=select chan0 input; 1=select chan1 input; 2=select chan2 input.
27:26	RW	0	source_chan1: 0=select chan0 input; 1=select chan1 input; 2=select chan2 input.
25:24	RW	0	source_chan0: 0=select chan0 input; 1=select chan1 input; 2=select chan2 input.
22:20	RW	0	skew_chan2: 0=Delay data by 1 clock cycle; 1=Delay by 2 cycles; ...; 7=Delay by 8.
18:16	RW	0	skew_chan1: 0=Delay data by 1 clock cycle; 1=Delay by 2 cycles; ...; 7=Delay by 8.
14:12	RW	0	skew_chan0: 0=Delay data by 1 clock cycle; 1=Delay by 2 cycles; ...; 7=Delay by 8.
10	RW	0	bitswap_chan2: 1=Reverse input data endian.
9	RW	0	bitswap_chan1: 1=Reverse input data endian.
8	RW	0	bitswap_chan0: 1=Reverse input data endian.
6	RW	0	polarity_chan2: 1=Invert input data polarity.
5	RW	0	polarity_chan1: 1=Invert input data polarity.
4	RW	0	polarity_chan0: 1=Invert input data polarity.

Bit	R/W	Default	Description
3	RW	0	charswap_2: 1=Swap the 10-bit word position within the 20-bit.
2	RW	0	charswap_1: 1=Swap the 10-bit word position within the 20-bit.
1	RW	0	charswap_0: 1=Swap the 10-bit word position within the 20-bit.
0	RW	0	enable: 1=enable data channels.

Table 9-47 HDMIRX\_TOP\_TMDS\_ALIGN\_STAT

Bit	R/W	Default	Description
0	R	0	tmads_align_status 0=Not aligned; 1=aligned.

Table 9-48 HDMIRX\_TOP\_INFILTER\_HDCP

Bit	R/W	Default	Description
31:29	RW	0	For HDCP/SCDC SDA infilter: filter internal clock divider. Refer to SCL infilter.
28:16	RW	0	For HDCP/SCDC SDA infilter: sampling clock divider. Refer to SCL infilter.
15: 13	RW	0	For HDCP/SCDC SCL infilter: filter internal clock divider. 0=No divide; 1=Divide by 2; 2=Divide by 3; ... 7=Divide by 8.
12:0	RW	0	For HDCP/SCDC SCL infilter: sampling clock divider. 0=No divide; 1=Divide the filter sampling clock by 2; 2=Divide the filter sampling clock by 3; ... 8191=Divide the filter sampling clock by 8192;

Table 9-49 HDMIRX\_TOP\_INFILTER\_I2C\_0/1/2

Bit	R/W	Default	Description
31:29	RW	0	For EDID SDA infilter: filter internal clock divider. Refer to HDCP infilter.
28:16	RW	0	For EDID SDA infilter: sampling clock divider. Refer to HDCP infilter.
15: 13	RW	0	For EDID SCL infilter: filter internal clock divider. Refer to HDCP infilter.
12:0	RW	0	For EDID SCL infilter: sampling clock divider. Refer to HDCP infilter.

**Table 9-50 HDMIRX\_TOP\_PRBS\_GEN**

Bit	R/W	Default	Description
13:10	RW	0	pattern_max: maximum pattern words. 0=Use pattern word 0 only; 1=Use pattern word 0~1; 2=Use pattern word 0~2; ... 9=Use pattern 0~9. 10~15=Reserved.
9	RW	0	shift_pattern_gen_enable: 0=Disable shift pattern generator; 1=Enable shift pattern generator.
8	RW	0	bist_loopback: 0=Loopback PRBS data to PHY; 1=Loopback original PHY data to PHY.
7:5	RW	3	decoupl_thresh_prbs: Read start threshold for decouple RIFO at PRBS generator.
4:3	RW	0	prbs_mode: 0=PRBS11; 1=PRBS15; 2=PRBS7; 3=PRBS31.
2:1	RW	0	prbs_width: 0=output 0; 1=output 8-bit pattern; 2=output 1-bit pattern; 3=output 10-bit pattern.
0	RW	0	prbs_enable

**Table 9-51 HDMIRX\_TOP\_PRBS\_ANA\_0**

Bit	R/W	Default	Description
23:22	RW	0	chan2_prbs_mode: 0=PRBS11; 1=PRBS15; 2=PRBS7; 3=PRBS31.
21:20	RW	0	chan2_prbs_width: 0=idle; 1=8-bit pattern; 2=reserved; 3=10-bit pattern.
19	RW	0	chan2_clr_BER: 1=Clear BER result.
18	RW	0	chan2_freeze_BER: 1=Freeze BER result so it can be sample.
17	RW	0	chan2_inverse: 1=Reverse bit order/endian.
15:14	RW	0	chan1_prbs_mode: 0=PRBS11; 1=PRBS15; 2=PRBS7; 3=PRBS31.
13:12	RW	0	chan1_prbs_width: 0=idle; 1=8-bit pattern; 2=reserved; 3=10-bit pattern.
11	RW	0	chan1_clr_BER: 1=Clear BER result.
10	RW	0	chan1_freeze_BER: 1=Freeze BER result so it can be sample.
9	RW	0	chan1_inverse: 1=Reverse bit order/endian.
7:6	RW	0	chan0_prbs_mode: 0=PRBS11; 1=PRBS15; 2=PRBS7; 3=PRBS31.
5:4	RW	0	chan0_prbs_width: 0=idle; 1=8-bit pattern; 2=reserved; 3=10-bit pattern.
3	RW	0	chan0_clr_BER: 1=Clear BER result.

Bit	R/W	Default	Description
2	RW	0	chan0_freeze_BER. 1:Freeze BER result so it can be sample.
1	RW	0	chan0_inverse: 1=Reverse bit order/endian.

Table 9-52 HDMIRX\_TOP\_PRBS\_ANA\_1

Bit	R/W	Default	Description
19:8	RW	0	time_window: Define the number of clocks after which the lock signal goes high if error number is smaller than error_threshold.
7:0	RW	0	err_threshold: Allowed number of errors.

Table 9-53 HDMIRX\_TOP\_PRBS\_ANA\_STAT

Bit	R/W	Default	Description
5	R	0	chan2_pattern_not_ok
4	R	0	chan2_lock
3	R	0	chan1_pattern_not_ok
2	R	0	chan1_lock
1	R	0	chan0_pattern_not_ok
0	R	0	chan0_lock

Table 9-54 HDMIRX\_TOP\_PRBS\_ANA\_BER\_CH0

Bit	R/W	Default	Description
19:0	R	0	chan0_BER_result

Table 9-55 HDMIRX\_TOP\_PRBS\_ANA\_BER\_CH1

Bit	R/W	Default	Description
19:0	R	0	chan1_BER_result

Table 9-56 HDMIRX\_TOP\_PRBS\_ANA\_BER\_CH2

Bit	R/W	Default	Description
19:0	R	0	chan2_BER_result



**Table 9-57 HDMIRX\_TOP\_METER\_CABLE\_CNTL**

Bit	R/W	Default	Description
31:28	RW	0x1	meas_tolerance: If difference between the consecutive measure results is greater than meas_tolerance, then stat_meas_stable=0; Otherwise stat_meas_stable=1.
23:0	RW	0x80	ref_cycles: Number of meter_clk cycles as one measure duration.

**Table 9-58 HDMIRX\_TOP\_METER\_CABLE\_STAT**

Bit	R/W	Default	Description
31	R	0	meas_stable: 1=If difference between the consecutive measure results is greater than meas_tolerance.
23:0	R	0	meas_cycles: Number of cable_clk cycles counted during one measure duration. $\text{Freq}(\text{cable\_clk}) = \text{Freq}(\text{ref\_clk}) * \text{meas\_cycles} / \text{ref\_cycles}$

**Table 9-59 HDMIRX\_TOP\_CHAN\_SWITCH\_1**

Bit	R/W	Default	Description
4	RW	0	valid_always: 0=use tmdsvalid signal from FIFO to gate data receiving; 1=Regardless tmdsvalid signal level, always receive data form PHY.
3:0	RW	7	decoup_thresh_chan: Read start threshold for decouple FIFO at channel input.

**Table 9-60 HDMIRX\_TOP\_AUDPLL\_LOCK\_FILTER**

Bit	R/W	Default	Description
7:0	RW	0	audpll_lock_filter value N: filter out glitch $\leq N$

**Table 9-61 HDMIRX\_TOP\_CHAN01\_ERRCNT**

Bit	R/W	Default	Description
30:16	R	0	erdet_ch1: shadow IP's periodical error counter for channel 1.
14:0	R	0	erdet_ch0: shadow IP's periodical error counter for channel 0.

**Table 9-62 HDMIRX\_TOP\_CHAN2\_ERRCNT**

Bit	R/W	Default	Description
14:0	R	0	erdet_ch0: shadow IP's periodical error counter for channel 2.

**Table 9-63 HDMIRX\_TOP\_ACR\_CNTL2**

Bit	R/W	Default	Description
31:5	R	0	Reserved
4	RW	0	n_cts_auto_mode: 0=Auto update on every ACR packet, regardless N&CTS value change or not; 1=Auto update on ACR packet's N or CTS value change, so may not update on every ACR.
3	RW	0	n_cts_update_inv: 1=Invert hdmirx_n_cts_update output polarity.
2:0	RW	0	n_cts_update_del: 0=hdmirx_n_cts_update pulse is aligned with N&CTS value change; 1=hdmirx_n_cts_update pulse is 1 cycle behind N&CTS value change; ... 7=hdmirx_n_cts_update pulse is 7 cycle behind N&CTS value change.

**Table 9-64 HDMIRX\_TOP\_ACR\_N\_STAT**

Bit	R/W	Default	Description
19:0	R	0	Records N value to AUD_PLL.

**Table 9-65 HDMIRX\_TOP\_ACR\_CTS\_STAT**

Bit	R/W	Default	Description
19:0	R	0	Records CTS value to AUD_PLL.

**Table 9-66 HDMIRX\_TOP\_AUDMEAS\_CTRL**

Bit	R/W	Default	Description
31:16	RW	0	meas_tolerance. Clk is considered stable, if the difference between two consecutive measurements is less than meas_tolerance. Applicable meas_mode=0, periodic measure.
2	RW	0	meas_mode. 0=Periodic measure; 1=Accumulate time stamp.
1	RW	0	enable. 0=Disable; 1=Enable.
0	RW	0	sw_reset. 0=No reset; 1=Reset.

**Table 9-67 HDMIRX\_TOP\_AUDMEAS\_CYCLES\_M1**

Bit	R/W	Default	Description
31:0	RW	0	Set (meas_clk_cycles_m1+1) of meas_clk cycles as one measure period, to count the number of ref_clk cycles.

**Table 9-68 HDMIRX\_TOP\_AUDMEAS\_INTR\_MASKN**

Bit	R/W	Default	Description
3:0	RW	0	Interrupt src mask, one bit per interrupt src. 0=disable the src to interrupt; 1=enable the src to interrupt. Bit[3]: meas_stable fall Bit[2]: meas_stable rise Bit[1]: ref_counter overflow Bit[0]: one measure period completes

**Table 9-69 HDMIRX\_TOP\_AUDMEAS\_INTR\_STAT**

Bit	R/W	Default	Description
3:0	RW	0	Read back interrupt status, one bit per interrupt src. For each bit, write 1 to clear the interrupt. Bit[3]: meas_stable fall Bit[2]: meas_stable rise Bit[1]: ref_counter overflow Bit[0]: one measure period completes

**Table 9-70 HDMIRX\_TOP\_AUDMEAS\_REF\_CYCLES\_STAT0**

Bit	R/W	Default	Description
31:0	R	0	ref_clk_cycles_m1[31:0]. ref_clk_cycles_m1[47:0]+1 is the number of ref_clk cycles during one measure period (meas_mode=0), or during N measure period (meas_mode=1).

**Table 9-71 HDMIRX\_TOP\_AUDMEAS\_REF\_CYCLES\_STAT1**

Bit	R/W	Default	Description
17	R	0	meas_stable. 1=meas_clk is stable; 0=not stable. Applicable only to meas_mode=0.
16	R	0	ref_clk_cycles_ovl. 1=ref_clk_cycles_m1 wrapped over.
15:0	R	0	ref_clk_cycles_m1[47:32].

**Table 9-72 HDMIRX\_TOP\_SHIFT\_PTTN\_012**

Bit	R/W	Default	Description
29:20	RW	0	Shift pattern word 2.
19:10	RW	0	Shift pattern word 1.
9:0	RW	0	Shift pattern word 0.

**Table 9-73 HDMIRX\_TOP\_SHIFT\_PTTN\_345**

Bit	R/W	Default	Description
29:20	RW	0	Shift pattern word 5.
19:10	RW	0	Shift pattern word 4.
9:0	RW	0	Shift pattern word 3.

**Table 9-74 HDMIRX\_TOP\_SHIFT\_PTTN\_678**

Bit	R/W	Default	Description
29:20	RW	0	Shift pattern word 8.
19:10	RW	0	Shift pattern word 7.
9:0	RW	0	Shift pattern word 6.

**Table 9-75 HDMIRX\_TOP\_SHIFT\_PTTN\_9**

Bit	R/W	Default	Description
9:0	RW	0	Shift pattern word 9.

**Table 9-76 HDMIRX\_TOP\_SHFT\_ANA\_CNTL**

Bit	R/W	Default	Description
15:4	RW	0	time_window: Defines the number of cycles of error-free before the lock signal goes high.
3	RW	0	snapshot_in. 1=Take a snapshot of received pattern, that can be read back through HDMIRX_TOP_SHFT_PTTN_RCV_***.
2	RW	0	freeze_ber. 1=Freeze the data into ber_meter register so it can be read.
1	RW	0	shift_pattern_clr_ber_meter. 1=Clear BER counter.
0	RW	0	shift_pattern_ana_enable. 0=Disable shift pattern analyzer; 1=Enable shift pattern analyzer.

**Table 9-77 HDMIRX\_TOP\_SHFT\_ANA\_STAT**

Bit	R/W	Default	Description
23:4	R	0	stat_shft_ana_ber: BER meter counter output. Buffered by freeze_ber, so safe for reg read.
1	R	0	shift_ana_pattern_Nok: Status of the current comparison, valid after shift_ana_pattern_lock =1. 0=Received pattern match; 1=Received pattern NOT match
0	R	0	shift_ana_pattern_lock: Sticky signal, =1 after the received pattern has matched for at least for time_window cycles.

**Table 9-78 HDMIRX\_TOP\_SHFT\_PTTN\_RCV\_012**

Bit	R/W	Default	Description
29:20	R	0	Received shift pattern word 2. (After taking a snapshot, and maybe bit-shifted.)
19:10	R	0	Received shift pattern word 1. (After taking a snapshot, and maybe bit-shifted.)
9:0	R	0	Received shift pattern word 0. (After taking a snapshot, and maybe bit-shifted.)

**Table 9-79 HDMIRX\_TOP\_SHFT\_PTTN\_RCV\_345**

Bit	R/W	Default	Description
29:20	R	0	Received shift pattern word 5. (After taking a snapshot, and maybe bit-shifted.)
19:10	R	0	Received shift pattern word 4. (After taking a snapshot, and maybe bit-shifted.)
9:0	R	0	Received shift pattern word 3. (After taking a snapshot, and maybe bit-shifted.)

**Table 9-80 HDMIRX\_TOP\_SHFT\_PTTN\_RCV\_678**

Bit	R/W	Default	Description
29:20	R	0	Received shift pattern word 8. (After taking a snapshot, and maybe bit-shifted.)
19:10	R	0	Received shift pattern word 7. (After taking a snapshot, and maybe bit-shifted.)
9:0	R	0	Received shift pattern word 6. (After taking a snapshot, and maybe bit-shifted.)

**Table 9-81 HDMIRX\_TOP\_SHFT\_PTTN\_RCV\_9**

Bit	R/W	Default	Description
9:0	R	0	Received shift pattern word 9. (After taking a snapshot, and maybe bit-shifted.)

**Table 9-82 HDMIRX\_TOP\_NSEC\_SCRATCH**

Bit	R/W	Default	Description
31:0	RW	0	Scratch register that can be used for either secure or non-secure reg access.

**Table 9-83 HDMIRX\_TOP\_SEC\_SCRATCH**

Bit	R/W	Default	Description
31:0	RW	0	Scratch register that can be used for secure reg access only.

**Table 9-84 HDMIRX\_TOP\_EMP\_DDR\_START\_A**

Bit	R/W	Default	Description
31:0	RW	0	DDR pointer A[33:2] for storing EMP packets or TMDS. 16-byte aligned, so low 2-bit is ignored. For EMP-to-DDR, the pointer is alternated between A and B, frame by frame. For TMDS-to-DDR, the pointer is fixed at A.

**Table 9-85 HDMIRX\_TOP\_EMP\_DDR\_START\_B**

Bit	R/W	Default	Description
31:0	RW	0	DDR pointer B[33:2] for storing EMP packets. 16-byte aligned, so low 2-bit is ignored. For EMP-to-DDR, the pointer is alternated between A and B, frame by frame. Not applicable for TMDS-to-DDR.

**Table 9-86 HDMIRX\_TOP\_EMP\_DDR\_FILTER**

Applicable only to EMP-to-DDR mode.

Bit	R/W	Default	Description
24	RW	0	ddr_store_other. 1=Enable store other packets that is neither NULL nor any that has already been described in bit[22:0].
23	RW	0	Reserved.
22	RW	0	ddr_store_drm. 1=Enable store Dynamic Range and Mastering InfoFrame.
21	RW	0	ddr_store_ntscvbi. 1=Enable store NTSC VBI InfoFrame.
20	RW	0	ddr_store_mpegs. 1=Enable store MPEG Source InfoFrame.
19	RW	0	ddr_store_aif. 1=Enable store Audio InfoFrame.
18	RW	0	ddr_store_spd. 1=Enable store Source Product Description InfoFrame.
17	RW	0	ddr_store_avi. 1=Enable store Auxiliary Video InfoFrame.
16	RW	0	ddr_store_vsi. 1=Enable store Vendor Specific InfoFrame.
15	RW	0	ddr_store_emp. 1=Enable store Extended Metadata packet.
14	RW	0	ddr_store_obm. 1=Enable store One bit Multi-stream audio sample packet.
13	RW	0	ddr_store_mas. 1=Enable store Multi-Stream Audio sample packet.
12	RW	0	ddr_store_amp. 1=Enable store Audio Metadata packet.
11	RW	0	ddr_store_ob3d. 1=Enable store One Bit 3D audio sample packet.
10	RW	0	ddr_store_auds3d. 1=Enable store 3D Audio Sample packet.
9	RW	0	ddr_store_gmd. 1=Enable store Gamut MetaData packet.
8	RW	0	ddr_store_hbr. 1=Enable store High-BitRate audio stream packet.
7	RW	0	ddr_store_dst. 1=Enable store DST audio packet.
6	RW	0	ddr_store_oba. 1=Enable store One Bit Audio sample packet.
5	RW	0	ddr_store_isrc2. 1=Enable store ISRC2 packet.
4	RW	0	ddr_store_isrc1. 1=Enable store ISRC1 packet.
3	RW	0	ddr_store_acp. 1=Enable store Audio Content Protection packet.
2	RW	0	ddr_store_gcp. 1=Enable store General Control packet.
1	RW	0	ddr_store_auds. 1=Enable store Audio Sample packet.
0	RW	0	ddr_store_acr. 1=Enable store Audio Clock Regeneration packet.

**Table 9-87 HDMIRX\_TOP\_EMP\_CNTL\_0**

Bit	R/W	Default	Description
23:16	RW	0	hs_beat_rate. 0=Break-up DDR write burst on every Hsync rise; N=Break-up DDR write burst on the (N+1)th Hsync rise. Applicable only for EMP-to-DDR mode.
15	RW	0	pkt_clean_mode.

Bit	R/W	Default	Description
			0=Store packet regardless of BCH error; 1=Only store packet that is clean of BCH error. Applicable only for EMP-to-DDR mode.
14	RW	0	buffer_info_mode. 0=Buffer received-packet-count and DDR-pointer-in-use on EMP_FIELD_DONE; 1=Buffer on the last EMP packet. Applicable only for EMP-to-DDR mode.
13	RW	0	reset_on_de. 1=Reset on DE rise. Applicable only for EMP-to-DDR mode.
12	RW	0	brst_end_on_last_emp. 1=Break-up DDR write burst at the last EMP packet. 0=Do not break-up. The burst will end at 16x128-bit, or at programmable Hsync interval or at DE rise. Note: Set this bit to 0 for TMDS-to-DDR mode.
11:2	RW	0	de_rise_delay. 0=Generate DE rise on actual DE rise; N=Generate DE rise (N+1) tmds_clk cycles after actual DE rise. Applicable only for EMP-to-DDR mode.
1:0	RW	0	Endian for DDR storage. 0=Little-endian. 3=Big-endian per 8-byte.

Table 9-88 HDMIRX\_TOP\_EMP\_CNTL\_1

Bit	R/W	Default	Description
1	RW	0	ddr_mode. 0=Store packet, EMP-to-DDR; 1=Store tmds character from PHY, TMDS-to-DDR.
0	RW	0	ddr_en. 1=Enable storing either packet or tmds to DDR.

Table 9-89 HDMIRX\_TOP\_EMP\_CNTMAX

Bit	R/W	Default	Description
31:0	RW	0	For EMP-to-DDR, maximum number of packets to store per frame; For TMDS-to-DDR, maximum number of 3x10-b tmds characters to store.

Table 9-90 HDMIRX\_TOP\_EMP\_ERR\_STAT

Bit	R/W	Default	Description
9	R	0	packet_num_exceeds_max
8	R	0	axi_overdue
7	R	0	bfifo_underflow
6	R	0	bfifo_overflow
5	R	0	axi_bresp error
4	R	0	Extra axi_bresp
3	R	0	wfifo_underflow

Bit	R/W	Default	Description
2	R	0	wfifo_overflow
1	R	0	awfifo_underflow
0	R	0	awfifo_overflow

Table 9-91 HDMIRX\_TOP\_EMP\_RCV\_CNT\_CUR

Bit	R/W	Default	Description
31:0	R	0	packet_rcv_cnt_cur. Current received packet count. Applicable only for EMP-to-DDR mode..

Table 9-92 HDMIRX\_TOP\_EMP\_RCV\_CNT\_BUF

Bit	R/W	Default	Description
31:0	R	0	packet_rcv_cnt_buf. Received packet count, buffered on either LAST_EMP_DONE or EMP_FIELD_DONE. Applicable only for EMP-to-DDR mode.

Table 9-93 HDMIRX\_TOP\_EMP\_DDR\_ADDR\_CUR

Bit	R/W	Default	Description
31:0	R	0	Current DDR pointer[33:2]. Low 2-bit always 0.

Table 9-94 HDMIRX\_TOP\_EMP\_DDR\_PTR\_S\_BUF

Bit	R/W	Default	Description
31:0	R	0	DDR start pointer[33:2] buffered on either LAST_EMP_DONE or EMP_FIELD_DONE. Applicable only for EMP-to-DDR mode.

Table 9-95 HDMIRX\_TOP\_EMP\_STAT\_0

Bit	R/W	Default	Description
31	R	0	odd_field
26:18	R	0	wfifo_count
17:16	R	0	axi_w_state. 0=IDLE; 1=WAIT_READY_L; 2=WAIT_READY_H.
10:2	R	0	awfifo_cnt
1:0	R	0	axi_aw_state. 0=IDLE; 1=WAIT_READY; 2=WAIT_W.

Table 9-96 HDMIRX\_TOP\_EMP\_STAT\_1

Bit	R/W	Default	Description
15:0	R	0	aw_pending_cnt



**Table 9-97 HDMIRX\_TOP\_PHYIF\_CNTL0**

Bit	R/W	Default	Description
15:0	RW	0	ckdt_sel. 0/1=Select hdmirx_sqofclk as clock detect signal to PHY digital; 2=Select hdmirx_tmdsvalid as clock detect signal to PHY digital; 3=Select hdmirx_pll_lock as clock detect signal to PHY digital.

**Table 9-98 HDMIRX\_TOP\_FIX\_DISABLE\_0**

Bit	R/W	Default	Description
31:0	RW	0	fix_disable[31:0]. Each bit to control whether to fix a internal time crossing bug. 0=Enable fix; 1=Go back to original behavior.

**Table 9-99 HDMIRX\_TOP\_MISC\_STAT0**

Bit	R/W	Default	Description
9	R	0	hdmirx_pll_lock status from PHY
8	R	0	hdmirx_tmdsvalid status from PHY
2	R	0	Hsync polarity
1	R	0	Vsync polarity
0	R	0	PHY.sqofclk status

**Table 9-100 HDMIRX\_TOP\_OVID\_OVERRIDE0**

Bit	R/W	Default	Description
31	R	0	vid_comp_override 1=Override vid_comp0/1/2 output to VDIN with manual values. For production test.
29:20	RW	0	vid_comp2_manual[9:0] to VDIN
19:10	RW	0	vid_comp1_manual[9:0] to VDIN
9:0	RW	0	vid_comp0_manual[9:0] to VDIN

**Table 9-101 HDMIRX\_TOP\_OVID\_OVERRIDE1**

Bit	R/W	Default	Description
21	RW	0	hdcp_enc_state to VDIN
20	RW	0	VS to VDIN
19	RW	0	HS to VDIN
18	RW	0	Field to VDIN
17	RW	0	DE to VDIN
16	RW	0	dpix to VDIN
9:0	RW	0	vid_comp2_manual[19:10] to VDIN

**Table 9-102 HDMIRX\_TOP\_SECURE\_INDEX**

Bit	R/W	Default	Description
4:0	RW	0	Start pointer of the 32-deep secure core register table

**Table 9-103 HDMIRX\_TOP\_SECURE\_DATA**

Bit	R/W	Default	Description
15:0	RW	0	The core register address value to be programmed into the 32-deep table.

**Table 9-104 HDMIRX\_TOP\_SECURE\_MODE**

Bit	R/W	Default	Description
1	RW	0	cor_reg_access_en: 0=Cannot access Controller core registers; 1=Enable access Controller core register.

**Table 9-105 HDMIRX\_TOP\_EDID\_ADDR0\_S - HDMIRX\_TOP\_EDID\_ADDR0\_E(0x1000-0x11ff)**

0x1000 for port0' s EDID byte 0, 0x1001 for port0' s EDID byte1, and so on. Store up to 512-byte EDID data.

Bit	R/W	Default	Description
7:0	RW	0	Port0's EDID byte.

**Table 9-106 HDMIRX\_TOP\_EDID\_ADDR1\_S - HDMIRX\_TOP\_EDID\_ADDR1\_E(0x1200-0x13ff)**

0x1200 for port1' s EDID byte 0, 0x1201 for port1' s EDID byte1, and so on. Store up to 512-byte EDID data.

Bit	R/W	Default	Description
7:0	RW	0	Port1's EDID byte.

**Table 9-107 HDMIRX\_TOP\_EDID\_ADDR2\_S - HDMIRX\_TOP\_EDID\_ADDR2\_E(0x1400-0x15ff)**

0x1400 for port2' s EDID byte 0, 0x1401 for port2' s EDID byte1, and so on. Store up to 512-byte EDID data.

Bit	R/W	Default	Description
7:0	RW	0	Port2's EDID byte.

### 9.1.5.1.3 Operating Procedure

#### **EMP-To-DDR: Store Extended Metadata Packet to DDR**

1. EMPs are transmitted during the video blanking periods. DE rise edge signals the end of video blanking period, where the packets should all have been received, and the content of the packets should apply to the first active video pixel following the video blank.

2. Enable emp\_field\_done interrupt by setting HDMIRX\_TOP\_INTR\_MASKN bit[25]=1.
3. Program the following DDR start address pointers A and B, the DDR start pointer will alternate between A and B frame-by-frame. The address must be at least 16-byte aligned, meaning the low 4-bit will be ignored.
  - HDMIRX\_TOP\_EMP\_DDR\_START\_A programs address[33:2] where low 2-bit=0, hence address[3:0]=0 for 16-byte aligned.
  - HDMIRX\_TOP\_EMP\_DDR\_START\_B programs address[33:2] where low 2-bit=0, hence address[3:0]=0 for 16-byte aligned.
4. Program HDMIRX\_TOP\_EMP\_DDR\_FILTER, to select the packet type(s) to be stored in DDR.
5. Program HDMIRX\_TOP\_EMP\_CNTMAX to limit the number packets to be stored in DDR. E.g., if this register is set to 5, only the first 5 type-selected packets are allow to go to DDR.
6. Program HDMIRX\_TOP\_EMP\_CNTL\_0 appropriately, for detail refer to register description section 2.88. Recommended values for the following bit fields:
  - [23:16] hs\_beat\_rate=0xf
  - [14] buffer\_info\_mode=0
  - [13] reset\_on\_de=1
  - [12] burst\_end\_on\_last\_emp=1
  - [11:2] de\_rise\_delay=0
  - [1:0] Endian = 0
7. Write HDMIRX\_TOP\_EMP\_CNTL\_1 bit[1] ddr\_mode=0 first, to select the function mode to be EMP-to-DDR.

Then write HDMIRX\_TOP\_EMP\_CNTL\_1 bit[0] ddr\_en=1, to enable the function.

8. On emp\_field\_done interrupt, read the following registers to determine where and how many packets in DDR:

HDMIRX\_TOP\_EMP\_DDR\_PTR\_S\_BUF – The DDR start pointer[33:2] to read the stored packets

HDMIRX\_TOP\_EMP\_RCV\_CNT\_BUF – The number of packets stored in DDR

9. The format of the packet data in DDR memory (little-endian) is as below: ----- 1st received Packet

```

Addr_S+0:Packet Header HB0 (E.g: EMP=0x7F, DRM=0x87)
Addr_S+1:Packet Header HB1 (E.g: EMP=0x80/0x00/0x40, DRM=InfoFrame_version)
Addr_S+2:Packet Header HB2 (E.g: EMP=Sequence_index, DRM=InfoFrame_length)
Addr_S+3:Packet Header PB0 (E.g: EMP=Payload byte0, DRM=Checksum byte)
Addr_S+4:Packet Header PB1 (E.g: EMP=Payload byte1, DRM=Byte1)
Addr_S+5:Packet Header PB2 (E.g: EMP=Payload byte2, DRM=Byte2)
.....
Addr_S+30:Packet Header PB27 (E.g: EMP=Payload byte27, DRM=Byte27)
Addr_S+31:0x00
----- 2nd received Packet
Addr_S+32:Packet Header HB0
.....
Addr_S+63:0x00
----- 3rd received Packet
.....
.....

```

```

----- N-th/Last received Packet
Addr_S+32*(N-1):Packet Header HB0
.....
Addr_S+32*N-1:0x00
-----

```



### Note

- *Addr\_S* is the start address of a DDR location to store the packet data, must be 16-byte aligned. If use LPDDR4 which is most efficient at 64-byte aligned access, then it is better that the *Addr\_S* is 64-byte aligned, for memory bandwidth efficiency.
- *N* is the total number of packets that were received in a Frame. Need  $32*N$  byte memory space.
- All types of packets are supported.

## TMDS-To-DDR: Store TMDS characters to DDR

1. This function stores a pre-defined number of 3-channel TMDS characters from PHY, into DDR. It is for debugging only, and can not co-exist with EMP-to-DDR function. If EMP-to-DDR is enabled, then you cannot do TMDS-to-DDR, and vice-versa.
2. If previously EMP-to-DDR is enabled, disable it by write `HDMIRX_TOP_EMP_CNTL_1 bit[0] ddr_en=0`.
3. Wait until `HDMIRX_TOP_EMP_STAT_0 bit[30:0]` reads back 0, and `HDMIRX_TOP_EMP_STAT_1` reads back 0.
4. Check no error in `HDMIRX_TOP_EMP_ERR_STAT`.
5. Enable `last_emp_done` interrupt by setting `HDMIRX_TOP_INTR_MASKN bit[26]=1`.
6. Program `HDMIRX_TOP_EMP_DDR_START_A` as the DDR start address[33:2] to store TMDS.
7. Program `HDMIRX_TOP_EMP_CNTMAX` to limit the number 3x10-bit TMDS to be stored in DDR. E.g., if this register is set to  $8-1=7$ , then 8 of 3x10-bit TMDS characters will be stored to DDR.

(`HDMIRX_TOP_EMP_CNTMAX + 1`) must be integer multiplier of 8.

8. Program `HDMIRX_TOP_EMP_CNTL_0` appropriately, for detail refer to the register description.

Recommended values for the following bit fields:

- [23:16] `hs_beat_rate`. N/A for TMDS-to-DDR.
- [14] `buffer_info_mode`. N/A for TMDS-to-DDR.
- [13] `reset_on_de`. N/A for TMDS-to-DDR.
- [12] `burst_end_on_last_emp=0`. **Note: Must be 0 for TMDS-to-DDR.**
- [11:2] `de_rise_delay`. N/A for TMDS-to-DDR.
- [1:0] `Endian = 0`
- Write `HDMIRX_TOP_EMP_CNTL_1 bit[1] ddr_mode=1` first, to select the function mode to be TMDS-to-DDR.

Then write `HDMIRX_TOP_EMP_CNTL_1 bit[0] ddr_en=1`, to enable recording TMDS.

- On `last_emp_done` interrupt, write `HDMIRX_TOP_EMP_CNTL_1 bit[0] ddr_en=0`. Dump DDR from address `HDMIRX_TOP_EMP_DDR_START_A`, of  $(\text{HDMIRX\_TOP\_EMP\_CNTMAX} + 1)*4$  bytes.
- The format of TMDS data in DDR memory (little-endian) is as below:----- 1st

3x10-bit TMDS

```

{Addr_S+3,+2,+1,+0}:{2'h0, tmds_ch2[9:0], tmds_ch1[9:0], tmds_ch0[9:0]}
----- 2nd 3x10-bit TMDS
{Addr_S+7,+6,+5,+4}:{2'h0, tmds_ch2[9:0], tmds_ch1[9:0], tmds_ch0[9:0]}
----- 3rd 3x10-bit TMDS
.....
.....
----- N-th/Last 3x10-bit TMDS
Addr_S+32*(N-1):Packet Header HBO
.....
Addr_S+32*N-1:0x00
{Addr_S+(N-1)*4+3 ,+(N-1)*4+2, ,+(N-1)*4+1,+(N-1)*4}:
{2'h0, tmds_ch2[9:0], tmds_ch1[9:0], tmds_ch0[9:0]}
-----

```



### Note

- *Addr\_S* is the start address of a DDR location to store the packet data, must be 16-byte aligned. If use LPDDR4 which is most efficient at 64-byte aligned access, then it is better that the *Addr\_S* is 64-byte aligned, for memory bandwidth efficiency.
- *N* is the total number of 3x10-bit TMDS that were stored in DDR. Need  $4*N$  byte memory space.

## Audio Clock Measure: Periodic measure (meas\_mode=0)

This function measures the frequency of either Audio PLL clk or internal digital ACR clk.

1. Select either Audio PLL clk or digital ACR clk to measure: HDMIRX\_TOP\_ACR\_CNTL\_STAT bit[1] – 0=Audio PLL clk; 1=digital ACR clk.
2. Program ref\_clk to 24MHz xtal clock for measuring audio clk. HHI\_HDMIRX\_METER\_CLK\_CNTL
  - [6:0] clk\_div = 0
  - [8] clk\_en = 1
  - [10:9] clk\_sel = 0, to select xtal clock 24MHz.
3. Initialization: HDMIRX\_TOP\_AUDMEAS\_CTRL
  - [0] sw\_reset = 1
  - [1] enable = 1
  - [2] meas\_mode = 0
  - [31:16] meas\_tolerance = appropriate value, e.g. 2 for Aud PLL, 500 for ACR.

HDMIRX\_TOP\_AUDMEAS\_CYCLES\_M1 = meas\_cycles\_m1[31:0]. Set to appropriate value, e.g. 65535.

4. Start measure: HDMIRX\_TOP\_AUDMEAS\_CTRL

[0] sw\_reset= 0

5. One measure period is  $T(\text{audio clk}) * (\text{meas\_cycles\_m1}+1)$ . From initialization, wait about two periods, because the very first period maybe an incomplete measure.
6. Read back the result periodically.
  - HDMIRX\_TOP\_AUDMEAS\_REF\_CYCLES\_STAT0 – ref\_cycles\_m1[31:0]
  - HDMIRX\_TOP\_AUDMEAS\_REF\_CYCLES\_STAT1
  - [15:0] ref\_cycles\_m1[47:32]

- [16] ref\_cycles\_overflow. N/A for periodic measure.
  - [17] meas\_stable.
  - [31:24] timestamp\_num. N/A for periodic measure.
7. The audio clock frequency is:  $\text{Freq}(\text{audio\_clk}) = \text{Freq}(\text{ref\_clk}) * (\text{meas\_cycles\_m1} + 1) / (\text{ref\_cycles\_m1} + 1)$

If meas\_stable=0, it means that, between the two measure periods, the difference of ref\_cycles\_m1 exceeds meas\_tolerance.

$\text{ABS}(\text{last ref\_cycles\_m1} - \text{current ref\_cycles\_m1}) > \text{meas\_tolerance}$ .

### **Audio Clock Measure: Timestamp (meas\_mode=1)**

This function generates audio clock timestamps.

1. Select either Audio PLL clk or digital ACR clk to measure: HDMIRX\_TOP\_ACR\_CNTL\_STAT bit[1] – 0=Audio PLL clk; 1=digital ACR clk.
2. Program ref\_clk to 24MHz xtal clock for measuring audio clk.

HMI\_HDMIRX\_METER\_CLK\_CNTL

- [6:0] clk\_div = 0
  - [8] clk\_en = 1
  - [10:9] clk\_sel = 0, to select xtal clock 24MHz.
3. Initialization: HDMIRX\_TOP\_AUDMEAS\_CTRL
- [0] sw\_reset = 1
  - [1] enable = 1
  - [2] meas\_mode = 1
  - [31:16] meas\_tolerance. N/A for timestamp mode.

HDMIRX\_TOP\_AUDMEAS\_CYCLES\_M1 = meas\_cycles\_m1[31:0]. Set to appropriate value, e.g. 65535.

4.

Start measure: HDMIRX\_TOP\_AUDMEAS\_CTRL

[0] sw\_reset = 0

5. One measure period is  $T(\text{audio clk}) * (\text{meas\_cycles\_m1} + 1)$ . From initialization, wait about two periods, because the very first period maybe an incomplete measure.
- 6.

Read back the result periodically.

HDMIRX\_TOP\_AUDMEAS\_REF\_CYCLES\_STAT0 – ref\_cycles\_m1[31:0]

HDMIRX\_TOP\_AUDMEAS\_REF\_CYCLES\_STAT1

- [15:0] ref\_cycles\_m1[47:32]
- [16] ref\_cycles\_overflow
- [17] meas\_stable. N/A for timestamp mode.
- [31:24] timestamp\_num

7. timestamp\_num and ref\_cycles\_m1 reflects the time elapsed since the beginning of measurement. timestamp\_num shows the number of measure periods elapsed since measure begins; (ref\_cycles\_m1+1) shows the total number of ref\_clk cycles since measure begins.
8. To assist timestamp checking, AUDMEAS interrupt can be enabled to generate one interrupt per measure period. Refer to reg HDMIRX\_TOP\_AUDMEAS\_INTR\_MASKN/STAT.

### **Protect a selection of core registers to be secure mode access only**

1. After reset, none of the IP registers are accessible, regardless secure mode or not.
2. Via secure mode, program HDMIRX\_TOP\_SECURE\_INDEX to an appropriate value, e.g. = 0
3. Repetitively use secure write to HDMIRX\_TOP\_SECURE\_DATA with the list of IP register addresses that you'd like to set to be secure-access only. Only up to 32 IP registers can be selected.
4. Then write to 1 to HDMIRX\_TOP\_SECURE\_MODE, to enable IP register access.

### **9.1.5.2 HDMIRX PHY Registers**

#### **Register Address**

For below registers the base address is 0xfe39c000.

Each register final address = BASE + address \* 4.

The following lists describe the mapping between each register and its address.

- HDMIRX\_APLL\_CNTL0 0xfe39c000
- HDMIRX\_APLL\_CNTL1 0xfe39c004
- HDMIRX\_APLL\_CNTL2 0xfe39c008
- HDMIRX\_APLL\_CNTL3 0xfe39c00c
- HDMIRX\_APLL\_CNTL4 0xfe39c010
- HDMIRX\_PHY\_MISC0 0xfe39c014
- HDMIRX\_PHY\_MISC1 0xfe39c018
- HDMIRX\_PHY\_MISC2 0xfe39c01c
- HDMIRX\_PHY\_MISC3 0xfe39c020
- HDMIRX\_PHY\_DCHA\_CNTL0 0xfe39c024
- HDMIRX\_PHY\_DCHA\_CNTL1 0xfe39c028
- HDMIRX\_PHY\_DCHA\_CNTL2 0xfe39c02c
- HDMIRX\_PHY\_DCHA\_CNTL3 0xfe39c030
- HDMIRX\_PHY\_DCHD\_CNTL0 0xfe39c034
- HDMIRX\_PHY\_DCHD\_CNTL1 0xfe39c038
- HDMIRX\_PHY\_DCHD\_CNTL2 0xfe39c03c
- HDMIRX\_PHY\_DCHD\_CNTL3 0xfe39c040
- HDMIRX\_PHY\_DCHD\_CNTL4 0xfe39c044
- HDMIRX\_PHY\_MISC\_STAT 0xfe39c048
- HDMIRX\_PHY\_DCHD\_STAT 0xfe39c04c
- HDMIRX\_AUD\_PLL\_CNTL 0xfe39c080
- HDMIRX\_AUD\_PLL\_CNTL2 0xfe39c084
- HDMIRX\_AUD\_PLL\_CNTL3 0xfe39c088
- HDMIRX\_AUD\_PLL\_CNTL\_I 0xfe39c08c
- HDMIRX\_AUD\_PLL4X\_CNTL 0xfe39c090
- HDMIRX\_PHY\_PROD\_TEST0 0xfe39c0c0
- HDMIRX\_PHY\_PROD\_TEST1 0xfe39c0c4

- HDMIRX\_EARCTX\_CNTL0 0xfe39c100
- HDMIRX\_EARCTX\_CNTL1 0xfe39c104
- HDMIRX\_ARC\_CNTL 0xfe39c108

## Register Description

**Table 9-108 HDMIRX\_APLL\_CNTL0 0x00**

Bit(s)	R/W	Default	Description
[31]	R	0	apll_lock
[30]	R	0	apll_afc_done
[29]	R/W	1	apll_reset
[28]	R/W	0	apll_en
[27]	R/W	0	apll_vco_div_sel
[26]	R/W	0	apll_afc_start
[14:10]	R/W	0	apll_prediv_sel
[7:0]	R/W	0	apll_fbdiv

**Table 9-109 HDMIRX\_APLL\_CNTL1 0x01**

Bit(s)	R/W	Default	Description
[20]	R/W	0	apll_sdm_en
[18:0]	R/W	0	apll_sdm_frac

**Table 9-110 HDMIRX\_APLL\_CNTL2 0x02**

Bit(s)	R/W	Default	Description
[31:28]	R/W	0	apll_ssc_dep_sel
[26:24]	R/W	0	apll_ssc_fref_sel
[23:22]	R/W	0	apll_ssc_mode
[21:20]	R/W	0	apll_ssc_offset
[19:16]	R/W	0	apll_str_m
[15:0]	R/W	0	apll_reserve

**Table 9-111 HDMIRX\_APLL\_CNTL3 0x03**

Bit(s)	R/W	Default	Description
[31]	R/W	0	apll_afc_bypass_en
[29:28]	R/W	0	apll_afc_hold_t
[26:20]	R/W	0	apll_afc_in
[19]	R/W	0	apll_afc_nt
[18:17]	R/W	0	apll_afc_div



Bit(s)	R/W	Default	Description
[16]	R/W	0	apll_bias_lpf_en
[15:12]	R/W	0	apll_cp_icap
[11:8]	R/W	0	apll_cp_ires
[5:4]	R/W	0	apll_cpi
[2:0]	R/W	0	apll_hdmi_aud_cntl

**Table 9-112 HDMIRX\_APLL\_CNTL4 0x04**

Bit(s)	R/W	Default	Description
[27:24]	R/W	0	apll_hdmi_cntl
[23]	R/W	0	apll_vctrl_mon_en
[21:20]	R/W	0	apll_lpf_cap
[19:16]	R/W	0	apll_lpf_capadj
[13:12]	R/W	0	apll_lpf_res
[11]	R/W	0	apll_lpf_sf
[10]	R/W	0	apll_lvr_od_en
[9]	R/W	0	apll_refclk_mon_en
[8]	R/W	0	apll_fbclk_mon_en

**Table 9-113 HDMIRX\_PHY\_MISC0 0x05**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_misc[31:0]

**Table 9-114 HDMIRX\_PHY\_MISC1 0x06**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_misc[63:32]

**Table 9-115 HDMIRX\_PHY\_MISC2 0x07**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_misc[95:64]

**Table 9-116 HDMIRX\_PHY\_MISC3 0x08**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_misc[127:96]

**Table 9-117 HDMIRX\_PHY\_DCHA\_CNTL0 0x09**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_dcha_afe[31:0]

**Table 9-118 HDMIRX\_PHY\_DCHA\_CNTL1 0x0a**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_dcha_dfe[31:0]

**Table 9-119 HDMIRX\_PHY\_DCHA\_CNTL2 0x0b**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_dcha_ctrl[31:0]

**Table 9-120 HDMIRX\_PHY\_DCHA\_CNTL3 0x0c**

Bit(s)	R/W	Default	Description
[15:0]	R/W	0	hdmirx_dcha_pi[31:0]

**Table 9-121 HDMIRX\_PHY\_DCHD\_CNTL0 0x0d**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_dchd_cdr[31:0]

**Table 9-122 HDMIRX\_PHY\_DCHD\_CNTL1 0x0e**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_dchd_byp[31:0]

**Table 9-123 HDMIRX\_PHY\_DCHD\_CNTL2 0x0f**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_dchd_dfe[31:0]

**Table 9-124 HDMIRX\_PHY\_DCHD\_CNTL3 0x10**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_dchd_taps[31:0]

**Table 9-125 HDMIRX\_PHY\_DCHD\_CNTL4 0x11**

Bit(s)	R/W	Default	Description
[31:0]	R/W	0	hdmirx_dchd_eye[31:0]

**Table 9-126 HDMIRX\_PHY\_MISC\_STAT 0x12**

Bit(s)	R/W	Default	Description
[10:1]	R	0	reserved
[0]	R	0	reserved

**Table 9-127 HDMIRX\_PHY\_DCHD\_STAT 0x13**

Bit(s)	R/W	Default	Description
[23:16]	R	0	sta_dchd2
[15:8]	R	0	sta_dchd1
[7:0]	R	0	sta_dchd0

**HDMIRX\_AUD\_PLL\_CNTL 0x20**

Original HIU\_AUD\_PLL\_CNTL

**HDMIRX\_AUD\_PLL\_CNTL2 0x21**

Original HIU\_AUD\_PLL\_CNTL2

**HDMIRX\_AUD\_PLL\_CNTL3 0x22**

Original HIU\_AUD\_PLL\_CNTL3

**HDMIRX\_AUD\_PLL\_CNTL3 0x22**

Original HIU\_AUD\_PLL\_CNTL3

**HDMIRX\_AUD\_PLL\_CNTL\_I 0x23**

Original HIU\_AUD\_PLL\_CNTL\_I

**HDMIRX\_AUD\_PLL4X\_CNTL 0x24**

Original HIU\_AUD\_PLL4X\_CNTL

**Table 9-128 HDMIRX\_PHY\_PROD\_TEST0 0x30**

Bit(s)	R/W	Default	Description
[31:1]	R	0	prod_test_reg0

**Table 9-129 HDMIRX\_PHY\_PROD\_TEST1 0x31**

Bit(s)	R/W	Default	Description
[31:1]	R	0	prod_test_reg1

**Table 9-130 HDMIRX\_EARCTX\_CNTL0 0x40**

Bit(s)	R/W	Default	Description
[31:1]	R	0	prod_test_reg1

HDMIRX\_EARCTX\_CNTL1 0x41  
HDMIRX\_ARC\_CNTL 0x42

## 9.1.6 CEC Registers

### 9.1.6.1 CEC Top-level Registers

#### Register Address

The following lists describe the mapping between each register and its address.

Address	Register	RW	Function
0xfe044000	CECA_GEN_CNTL	RW	CECA general control.
0xfe044004	CECA_RW_REG	RW	CECA internal register access.
0xfe044008	CECA_INTR_MASKN	RW	CECA interrupt mask.
0xfe04400c	CECA_INTR_CLR	RW	CECA interrupt clear.
0xfe044010	CECA_INTR_STAT	RW	CECA interrupt status.
0xfe044040	CECB_GEN_CNTL	RW	CECB general control.
0xfe044044	CECB_RW_REG	RW	CECB internal register access.
0xfe044048	CECB_INTR_MASKN	RW	CECB interrupt mask.
0xfe04404c	CECB_INTR_CLR	RW	CECB interrupt clear.
0xfe044050	CECB_INTR_STAT	RW	CECB interrupt status.

#### Register Description

Table 9-131 CECA\_GEN\_CNTL

Bit	R/W	Default	Description
14:12	RW	0	Filter_del. For glitch-filtering CEC line, ignore signal change pulse width < filter_del * T(filter_tick) * 3.
9:8	RW	0	Filter_tick_sel: Select which periodical pulse for glitch-filtering CEC line signal. 0=Use T(xtal)*3=125ns; 1=Use once-per-1us pulse; 2=Use once-per-10us pulse; 3=Use once-per-100us pulse.
4	RW	0	Together with CECB_GEN_CNTL bit[4], to select whether CEC_A/B to share the same pin mux. {CECA_GEN_CNTL[4], CECB_GEN_CNTL[4]}: {0,0} = A and B separate, original behavior. Use this if pin-mux selects either A-only or B-only. {1,0} = Use this if pin-mux selects A, but also wants to use B through the same pin. {0,1} = Use this if pin-mux selects B, but also wants to use A through the same pin. {1,1} = This value is not allowed.
2:1	RW	0	Clk_cntl. 0=Disable clock; 1=Enable gated clock(normal working mode); 2=Enable free-run clock (debug mode).
0	RW	0	Sw_reset. 1=Apply reset; 0=No reset.

**Table 9-132 CECA\_RW\_REG**

Bit	R/W	Default	Description
31:24	R	0	CECA internal register data read-back value.
23	R	0	Cec_reg_busy. 1=Current register access is busy, do not start another register access yet; 0=Idle, safe to start a new register access.
16	RW	0	CECA internal register access mode. 1=Write; 0=Read.
15:8	RW	0	CECA internal register data write value.
7:0	RW	0	CECA internal register address offset. Refer to CECA reg spec.

**Table 9-133 CECA\_INTR\_MASKN**

Bit	R/W	Default	Description
2:0	RW	0	Interrupt mask. One bit for each interrupt source. Each bit: 0=Mask interrupt; 1=Enable interrupt. Note: For interrupt source description refer to CECA_INTR_STAT.

**Table 9-134 CECA\_INTR\_CLR**

Bit	R/W	Default	Description
2:0	RW	0	Interrupt clear. One bit for each interrupt source. Each bit: Write 1 to clear the respective interrupt status bit. These bits are written to 1 and automatically set back to 0. Note: For interrupt source description refer to CECA_INTR_STAT.

**Table 9-135 CECA\_INTR\_STAT**

Bit	R/W	Default	Description
2:0	R	0	Interrupt status. One bit for each interrupt source. Bit 0: HOST_ACK. 1=CECA internal register access action done. Bit 1: CEC_TX_INT. 1=CECA TX status update. Bit 2: CEC_RX_INT. 1=CECA RX status update.

**Table 9-136 CECB\_GEN\_CNTL**

Bit	R/W	Default	Description
14:12	RW	0	Filter_del. For glitch-filtering CEC line, ignore signal change pulse width < filter_del * T(filter_tick) * 3.
9:8	RW	0	Filter_tick_sel: Select which periodical pulse for glitch-filtering CEC line signal. 0=Use T(xtal)*3=125ns; 1=Use once-per-1us pulse; 2=Use once-per-10us pulse; 3=Use once-per-100us pulse.
4	RW	0	Refer to CECA_GEN_CNTL bit[4].
3	RW	0	Sysclk_en. 0=Disable system clk; 1=Enable system clk.

Bit	R/W	Default	Description
2:1	RW	0	Clk_cntl. 0=Disable clock; 1=Enable gated clock(normal working mode); 2=Enable free-run clock (debug mode).
0	RW	0	Sw_reset. 1=Apply reset; 0=No reset.

Table 9-137 CECB\_RW\_REG

Bit	R/W	Default	Description
31:24	R	0	CECB internal register data read-back value.
16	RW	0	CECB internal register access mode. 1=Write; 0=Read.
15:8	RW	0	CECB internal register data write value.
7:0	RW	0	CECB internal register address offset. Refer to CECB reg spec.

Table 9-138 CECB\_INTR\_MASKN

Bit	R/W	Default	Description
6:0	RW	0	Interrupt mask. One bit for each interrupt source. Each bit: 0=Mask interrupt; 1=Enable interrupt. Note: For interrupt source description refer to CECB_INTR_STAT.

Table 9-139 CECB\_INTR\_CLR

Bit	R/W	Default	Description
6:0	RW	0	Interrupt clear. One bit for each interrupt source. Each bit: Write 1 to clear the respective interrupt status bit. These bits are written to 1 and automatically set back to 0. Note: For interrupt source description refer to CECB_INTR_STAT.

Table 9-140 CECB\_INTR\_STAT

Bit	R/W	Default	Description
6:0	R	0	Interrupt status. One bit for each interrupt source. Bit 0: CECB Done interrupt flag status. Bit 1: CECB End of Message interrupt flag status. Bit 2: CECB Not Acknowledge interrupt flag status. Bit 3: CECB Arbitration Loss interrupt flag status. Bit 4: CECB Initiator Error interrupt flag status. Bit 5: CECB Follower Error interrupt flag status. Bit 6: CECB Wake-up interrupt flag status.

## 9.1.6.2 CECA Registers

### Register Access

Accessing CEC\_A internal registers is by indirectly method through a register.

```
#define CECA_RW_REG (0xfe044000 + (0x001 << 2))
```

```

void ceca_wr_only_reg (uint32_t addr, uint32_t data)
{
    uint32_t  data32;

    data32 = 0;
    data32 |= (1 << 16); // [16]   cec_reg_wr
    data32 |= (data << 8); // [15:8] cec_reg_wrdata
    data32 |= (addr << 0); // [7:0]  cec_reg_addr
    Wr(CECA_RW_REG, data32);
} /* cec_wr_only_reg */

uint32_t cec_rd_reg (uint32_t addr)
{
    uint32_t  data32;

    data32 = 0;
    data32 |= (0 << 16); // [16]   cec_reg_wr
    data32 |= (0 << 8); // [15:8] cec_reg_wrdata
    data32 |= (addr << 0); // [7:0]  cec_reg_addr
    Wr(CECA_RW_REG, data32);

    data32 = ((Rd(CECA_RW_REG)) >> 24) & 0xff;

    return (data32);
} /* cec_rd_reg */

```

## Register Address

Addr	Name	RW	Function
0x00	TX_MSG_HEADER	RW	TX message Header byte.
0x01	TX_MSG_OPCODE	RW	TX message Op-code.
0x02	TX_MSG_OP1	RW	TX message 1 <sup>st</sup> operand.
0x03	TX_MSG_OP2	RW	TX message 2 <sup>nd</sup> operand.
0x04	TX_MSG_OP3	RW	TX message 3 <sup>rd</sup> operand.
0x05	TX_MSG_OP4	RW	TX message 4 <sup>th</sup> operand.
0x06	TX_MSG_OP5	RW	TX message 5 <sup>th</sup> operand.
0x07	TX_MSG_OP6	RW	TX message 6 <sup>th</sup> operand.
0x08	TX_MSG_OP7	RW	TX message 7 <sup>th</sup> operand.
0x09	TX_MSG_OP8	RW	TX message 8 <sup>th</sup> operand.
0x0A	TX_MSG_OP9	RW	TX message 9 <sup>th</sup> operand.
0x0B	TX_MSG_OP10	RW	TX message 10 <sup>th</sup> operand.
0x0C	TX_MSG_OP11	RW	TX message 11 <sup>th</sup> operand.
0x0D	TX_MSG_OP12	RW	TX message 12 <sup>th</sup> operand.
0x0E	TX_MSG_OP13	RW	TX message 13 <sup>th</sup> operand.
0x0F	TX_MSG_OP14	RW	TX message 14 <sup>th</sup> operand.
0x10	TX_MSG_LENGTH	RW	TX message length.
0x11	TX_MSG_CMD	RW	TX message command.
0x12-0x13	Reserved	RW	
0x14	RX_MSG_CMD	RW	Command for reading RX msg.
0x15	RX_CLEAR_BUF	RW	Clear RX msg buffer.
0x16	LOGICAL_ADDR_LO	RW	Device logical address valid[7:0].

Addr	Name	RW	Function
0x17	LOGICAL_ADDR_HI	RW	Device logical address valid[15:8].
0x18-0x1A	Reserved	RW	
0x1B	CLOCK_DIV_H	RW	MSB of the clock divider.
0x1C	CLOCK_DIV_L	RW	LSB of the clock divider.
0x1D-0x1F	Reserved	RW	
0x20	QUIESCENT_25MS_BIT7_0	RW	CEC bit timing parameter.
0x21	QUIESCENT_25MS_BIT11_8	RW	CEC bit timing parameter.
0x22	STARTBITMINL2H_3MS5_BIT7_0	RW	CEC bit timing parameter.
0x23	STARTBITMINL2H_3MS5_BIT8	RW	CEC bit timing parameter.
0x24	STARTBITMAXL2H_3MS9_BIT7_0	RW	CEC bit timing parameter.
0x25	STARTBITMAXL2H_3MS9_BIT8	RW	CEC bit timing parameter.
0x26	STARTBITMINH_0MS6_BIT7_0	RW	CEC bit timing parameter.
0x27	STARTBITMINH_0MS6_BIT8	RW	CEC bit timing parameter.
0x28	STARTBITMAXH_1MS0_BIT7_0	RW	CEC bit timing parameter.
0x29	STARTBITMAXH_1MS0_BIT8	RW	CEC bit timing parameter.
0x2A	STARTBITMINTOTAL_4MS3_BIT7_0	RW	CEC bit timing parameter.
0x2B	STARTBITMINTOTAL_4MS3_BIT9_8	RW	CEC bit timing parameter.
0x2C	STARTBITMAXTOTAL_4MS7_BIT7_0	RW	CEC bit timing parameter.
0x2D	STARTBITMAXTOTAL_4MS7_BIT9_8	RW	CEC bit timing parameter.
0x2E	LOGIC1MINL2H_0MS4_BIT7_0	RW	CEC bit timing parameter.
0x2F	LOGIC1MINL2H_0MS4_BIT8	RW	CEC bit timing parameter.
0x30	LOGIC1MAXL2H_0MS8_BIT7_0	RW	CEC bit timing parameter.
0x31	LOGIC1MAXL2H_0MS8_BIT8	RW	CEC bit timing parameter.
0x32	LOGIC0MINL2H_1MS3_BIT7_0	RW	CEC bit timing parameter.
0x33	LOGIC0MINL2H_1MS3_BIT8	RW	CEC bit timing parameter.
0x34	LOGIC0MAXL2H_1MS7_BIT7_0	RW	CEC bit timing parameter.
0x35	LOGIC0MAXL2H_1MS7_BIT8	RW	CEC bit timing parameter.
0x36	LOGICMINTOTAL_2MS05_BIT7_0	RW	CEC bit timing parameter.
0x37	LOGICMINTOTAL_2MS05_BIT9_8	RW	CEC bit timing parameter.
0x38	LOGICMAXHIGH_2MS8_BIT7_0	RW	CEC bit timing parameter.
0x39	LOGICMAXHIGH_2MS8_BIT8	RW	CEC bit timing parameter.



Addr	Name	RW	Function
0x3A	LOGICERRLOW_3MS4_BIT7_0	RW	CEC bit timing parameter.
0x3B	LOGICERRLOW_3MS4_BIT8	RW	CEC bit timing parameter.
0x3C	NOMSMPPPOINT_1MS05	RW	CEC bit timing parameter.
0x3D	Reserved	RW	
0x3E	DELCNTR_LOGICERR	RW	CEC bit timing parameter.
0x3F	Reserved	RW	
0x40	TXTIME_17MS_BIT7_0	RW	CEC bit timing parameter.
0x41	TXTIME_17MS_BIT15_8	RW	CEC bit timing parameter.
0x42	TXTIME_2BIT_BIT7_0	RW	CEC bit timing parameter.
0x43	TXTIME_2BIT_BIT15_8	RW	CEC bit timing parameter.
0x44	TXTIME_4BIT_BIT7_0	RW	CEC bit timing parameter.
0x45	TXTIME_4BIT_BIT15_8	RW	CEC bit timing parameter.
0x46	STARTBITNOML2H_3MS7_BIT7_0	RW	CEC bit timing parameter.
0x47	STARTBITNOML2H_3MS7_BIT8	RW	CEC bit timing parameter.
0x48	STARTBITNOMH_0MS8_BIT7_0	RW	CEC bit timing parameter.
0x49	STARTBITNOMH_0MS8_BIT8	RW	CEC bit timing parameter.
0x4A	LOGIC1NOML2H_0MS6_BIT7_0	RW	CEC bit timing parameter.
0x4B	LOGIC1NOML2H_0MS6_BIT8	RW	CEC bit timing parameter.
0x4C	LOGIC0NOML2H_1MS5_BIT7_0	RW	CEC bit timing parameter.
0x4D	LOGIC0NOML2H_1MS5_BIT8	RW	CEC bit timing parameter.
0x4E	LOGIC1NOMH_1MS8_BIT7_0	RW	CEC bit timing parameter.
0x4F	LOGIC1NOMH_1MS8_BIT8	RW	CEC bit timing parameter.
0x50	LOGIC0NOMH_0MS9_BIT7_0	RW	CEC bit timing parameter.
0x51	LOGIC0NOMH_0MS9_BIT8	RW	CEC bit timing parameter.
0x52	LOGICERRLOW_3MS6_BIT7_0	RW	CEC bit timing parameter.
0x53	LOGICERRLOW_3MS6_BIT8	RW	CEC bit timing parameter.
0x54	CHKCONTENTION_0MS1	RW	CEC bit timing parameter.
0x55	Reserved	RW	
0x56	PREPARENXTBIT_0MS05_BIT7_0	RW	CEC bit timing parameter.
0x57	PREPARENXTBIT_0MS05_BIT8	RW	CEC bit timing parameter.
0x58	NOMSMPPACKPOINT_0MS45	RW	CEC bit timing parameter.
0x59	Reserved	RW	
0x5A	ACK0NOML2H_1MS5_BIT7_0	RW	CEC bit timing parameter.

Addr	Name	RW	Function
0x5B	ACK0NOML2H_1MS5_BIT8	RW	CEC bit timing parameter.
0x5C-0x5F	Reserved	RW	
0x60	BUGFIX_DISABLE_0	RW	Bug fix control.
0x61	BUGFIX_DISABLE_1	RW	Bug fix control.
0x62-0x7F	Reserved	RW	
0x80	RX_MSG_HEADER	R	RX message Header byte.
0x81	RX_MSG_OPCODE	R	RX message Op-code.
0x82	RX_MSG_OP1	R	RX message 1 <sup>st</sup> operand.
0x83	RX_MSG_OP2	R	RX message 2 <sup>nd</sup> operand.
0x84	RX_MSG_OP3	R	RX message 3 <sup>rd</sup> operand.
0x85	RX_MSG_OP4	R	RX message 4 <sup>th</sup> operand.
0x86	RX_MSG_OP5	R	RX message 5 <sup>th</sup> operand.
0x87	RX_MSG_OP6	R	RX message 6 <sup>th</sup> operand.
0x88	RX_MSG_OP7	R	RX message 7 <sup>th</sup> operand.
0x89	RX_MSG_OP8	R	RX message 8 <sup>th</sup> operand.
0x8A	RX_MSG_OP9	R	RX message 9 <sup>th</sup> operand.
0x8B	RX_MSG_OP10	R	RX message 10 <sup>th</sup> operand.
0x8C	RX_MSG_OP11	R	RX message 11 <sup>th</sup> operand.
0x8D	RX_MSG_OP12	R	RX message 12 <sup>th</sup> operand.
0x8E	RX_MSG_OP13	R	RX message 13 <sup>th</sup> operand.
0x8F	RX_MSG_OP14	R	RX message 14 <sup>th</sup> operand.
0x90	RX_MSG_LENGTH	R	Length in bytes of RX message.
0x91	RX_MSG_STATUS	R	Status of the received message.
0x92	RX_NUM_MSG	R	Number of received messages.
0x93	TX_MSG_STATUS	R	Status of the transmitted message.
0x94-0x9F	Reserved	R	
0xA0	STAT_0_0	R	Internal status register.
0xA1	STAT_0_1	R	Internal status register.
0xA2	STAT_0_2	R	Internal status register.
0xA3	STAT_0_3	R	Internal status register.
0xA4	STAT_1_0	R	Internal status register.
0xA5	STAT_1_1	R	Internal status register.
0xA6	STAT_1_2	R	Internal status register.

## Register Description

**Table 9-141 TX\_MSG\_HEADER**

Bit	R/W	Default	Description
7:4	RW	0	Initiator Address.
3:0	RW	0	Follower Address.

**Table 9-142 TX\_MSG\_OPCODE**

Bit	R/W	Default	Description
7:0	RW	0	Op-code

**Table 9-143 TX\_MSG\_OP1 – TX\_MSG\_OP14**

Bit	R/W	Default	Description
7:0	RW	0	1st–14th operand.

**Table 9-144 TX\_MSG\_LENTH**

Bit	R/W	Default	Description
3:0	RW	0	Tx msg length in bytes, excluding Header byte.

**Table 9-145 TX\_MSG\_CMD**

Bit	R/W	Default	Description
1:0	RW	0	Command for transmitting a message. 0=TX_NO_OP. No transaction. 1=TX_REQ_CURRENT. Transmit message in buffer. 2=TX_ABORT. Abort transmit. 3=Same as TX_REQ_CURRENT.

**Table 9-146 RX\_MSG\_CMD**

Bit	R/W	Default	Description
1:0	RW	0	Command for reading a received message. 0=RX_NO_OP. No transaction. 1=RX_ACK_CURRENT. Read message in buffer. 2=RX_DISABLE. Disable receiving message. 3=Same as RX_ACK_CURRENT.

**Table 9-147 RX\_CLEAR\_BUF**

Bit	R/W	Default	Description
0	RW	0	Clear message in Rx buffer.

**Table 9-148 LOGICAL\_ADDR\_LO**

Bit	R/W	Default	Description
7	RW	0	1=Enable receiving message destined to logical address 7.
6	RW	0	1=Enable receiving message destined to logical address 6.
5	RW	0	1=Enable receiving message destined to logical address 5.
4	RW	0	1=Enable receiving message destined to logical address 4.
3	RW	0	1=Enable receiving message destined to logical address 3.
2	RW	0	1=Enable receiving message destined to logical address 2.
1	RW	0	1=Enable receiving message destined to logical address 1.
0	RW	0	1=Enable receiving message destined to logical address 0.

**Table 9-149 LOGICAL\_ADDR\_HI**

Bit	R/W	Default	Description
7	RW	1	1=Enable receiving message destined to logical address 15.
6	RW	0	1=Enable receiving message destined to logical address 14.
5	RW	0	1=Enable receiving message destined to logical address 13.
4	RW	0	1=Enable receiving message destined to logical address 12.
3	RW	0	1=Enable receiving message destined to logical address 11.
2	RW	0	1=Enable receiving message destined to logical address 10.
1	RW	0	1=Enable receiving message destined to logical address 9.
0	RW	0	1=Enable receiving message destined to logical address 8.

**Table 9-150 CLOCK\_DIV\_H – CLOCK\_DIV\_L**

If CEC works in original behavior, meaning the internal beat rate is once per 0.01ms, then for example: Input clock is 24MHz, then set {CLOCK\_DIV\_H, CLOCK\_DIV\_L} = 240-1 = 239, so that the internal beat rate is once per 0.01ms. If CEC works under input clock = 32768Hz, no clock division, CLOK\_DIV\_H/L=0.

Bit	R/W	Default	Description
7:0	RW	0	MSB/LSB of the clock divider.

**Table 9-151 QUIESCENT\_25MS\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x33	Quiescent_25ms[7:0]

**Table 9-152 QUIESCENT\_25MS\_BIT11\_8**

Bit	R/W	Default	Description
3:0	RW	0x3	Quiescent_25ms[11:8]

**Table 9-153 STARTBITMINL2H\_3MS5\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x70	StartBitMinL2H_3ms5[7:0]

**Table 9-154 STARTBITMINL2H\_3MS5\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	StartBitMinL2H_3ms5[8]

**Table 9-155 STARTBITMAXL2H\_3MS9\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x7c	StartBitMaxL2H_3ms9[7:0]

**Table 9-156 STARTBITMAXL2H\_3MS9\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	StartBitMaxL2H_3ms9[8]

**Table 9-157 STARTBITMINH\_0MS6\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x11	StartBitMinH_0ms6[7:0]

**Table 9-158 STARTBITMINH\_0MS6\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	StartBitMinH_0ms6[8]

**Table 9-159 STARTBITMAXH\_1MS0\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x1d	StartBitMaxH_1ms0[7:0]

**Table 9-160 STARTBITMAXH\_1MS0\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	StartBitMaxH_1ms0[8]

**Table 9-161 STARTBITMINTOTAL\_4MS3\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x8a	StartBitMinTotal_4ms3[7:0]

**Table 9-162 STARTBITMINTOTAL\_4MS3\_BIT9\_8**

Bit	R/W	Default	Description
1:0	RW	0x0	StartBitMinTotal_4ms3[9:8]

**Table 9-163 STARTBITMAXTOTAL\_4MS7\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x96	StartBitMaxTotal_4ms7[7:0]

**Table 9-164 STARTBITMAXTOTAL\_4MS7\_BIT9\_8**

Bit	R/W	Default	Description
1:0	RW	0x0	StartBitMaxTotal_4ms7[9:8]

**Table 9-165 LOGIC1MINL2H\_0MS4\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x0a	Logic1MinL2H_0ms4[7:0]

**Table 9-166 LOGIC1MINL2H\_0MS4\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	Logic1MinL2H_0ms4[8]

**Table 9-167 LOGIC1MAXL2H\_0MS8\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x16	Logic1MaxL2H_0ms8[7:0]

**Table 9-168 LOGIC1MAXL2H\_0MS8\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	Logic1MaxL2H_0ms8[8]

**Table 9-169 LOGIC0MINL2H\_1MS3\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x27	Logic0MinL2H_1ms3[7:0]

**Table 9-170 LOGIC0MINL2H\_1MS3\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	Logic0MinL2H_1ms3[8]

**Table 9-171 LOGIC0MAXL2H\_1MS7\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x34	Logic0MaxL2H_1ms7[7:0]

**Table 9-172 LOGIC0MAXL2H\_1MS7\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	Logic0MaxL2H_1ms7[8]

**Table 9-173 LOGICMINTOTAL\_2MS05\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x40	LogicMinTotal_2ms05[7:0]

**Table 9-174 LOGICMINTOTAL\_2MS05\_BIT9\_8**

Bit	R/W	Default	Description
1:0	RW	0x0	LogicMinTotal_2ms05[9:8]

**Table 9-175 LOGICMAXHIGH\_2MS8\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x59	LogicMaxHigh_2ms8[7:0]

**Table 9-176 LOGICMAXHIGH\_2MS8\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	LogicMaxHigh_2ms8[8]

**Table 9-177 LOGICERRLOW\_3MS4\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x6c	LogicErrLow_3ms4[7:0]

**Table 9-178 LOGICERRLOW\_3MS4\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	LogicErrLow_3ms4[8]

**Table 9-179 NOMSMPPPOINT\_1MS05**

Bit	R/W	Default	Description
7:0	RW	0x1f	NomSmpPoint_1ms05[7:0]

**Table 9-180 DELCNTR\_LOGICERR**

Bit	R/W	Default	Description
6:4	RW	0x3	DelCntr_LogicErrHigh[2:0]
2:0	RW	0x5	DelCntr_LogicErrLow[2:0]

**Table 9-181 TXTIME\_17MS\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x2c	TxTime_17ms[7:0]

**Table 9-182 TXTIME\_17MS\_BIT15\_8**

Bit	R/W	Default	Description
7:0	RW	0x02	TxTime_17ms[15:8]

**Table 9-183 TXTIME\_2BIT\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0xa2	TxTime_2Bit[7:0]

**Table 9-184 TXTIME\_2BIT\_BIT15\_8**

Bit	R/W	Default	Description
7:0	RW	0x0	TxTime_2Bit[15:8]

**Table 9-185 TXTIME\_4BIT\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x3f	TxTime_4Bit[7:0]



**Table 9-186 TXTIME\_4BIT\_BIT15\_8**

Bit	R/W	Default	Description
7:0	RW	0x01	TxTime_4Bit[15:8]

**Table 9-187 STARTBITNOML2H\_3MS7\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x78	StartBitNomL2H_3ms7[7:0]

**Table 9-188 STARTBITNOML2H\_3MS7\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	StartBitNomL2H_3ms7[8]

**Table 9-189 STARTBITNOMH\_0MS8\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x19	StartBitNomH_0ms8[7:0]

**Table 9-190 STARTBITNOMH\_0MS8\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	StartBitNomH_0ms8[8]

**Table 9-191 LOGIC1NOML2H\_0MS6\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x13	Logic1NomL2H_0ms6[7:0]

**Table 9-192 LOGIC1NOML2H\_0MS6\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	Logic1NomL2H_0ms6[8]

**Table 9-193 LOGIC0NOML2H\_1MS5\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x30	Logic0NomL2H_1ms5[7:0]

**Table 9-194 LOGIC0NOML2H\_1MS5\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	Logic0NomL2H_1ms5[8]

**Table 9-195 LOGIC1NOMH\_1MS8\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x3a	Logic1NomH_1ms8[7:0]

**Table 9-196 LOGIC1NOMH\_1MS8\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	Logic1NomH_1ms8[8]

**Table 9-197 LOGIC0NOMH\_0MS9\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x1d	Logic0NomH_0ms9[7:0]

**Table 9-198 LOGIC0NOMH\_0MS9\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	Logic0NomH_0ms9[8]

**Table 9-199 LOGICERRLOW\_3MS6\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x75	LogicErrLow_3ms6[7:0]

**Table 9-200 LOGICERRLOW\_3MS6\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	LogicErrLow_3ms6[8]

**Table 9-201 CHKCONTENTION\_0MS1**

Bit	R/W	Default	Description
3:0	RW	0x3	ChkContention_0ms1[7:0]

**Table 9-202 PREPARENXTBIT\_0MS05\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x05	PrepareNxtBit_0ms05[7:0]

**Table 9-203 PREPARENXTBIT\_0MS05\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	PrepareNxtBit_0ms05[8]

**Table 9-204 NOMSMPACKPOINT\_0MS45**

Bit	R/W	Default	Description
7:0	RW	0x0f	NomSmpAckPoint_0ms45[7:0]

**Table 9-205 ACK0NOML2H\_1MS5\_BIT7\_0**

Bit	R/W	Default	Description
7:0	RW	0x2c	Ack0NomL2H_1ms5[7:0]

**Table 9-206 ACK0NOML2H\_1MS5\_BIT8**

Bit	R/W	Default	Description
0	RW	0x0	Ack0NomL2H_1ms5[8]

**Table 9-207 BUGFIX\_DISABLE\_0**

Bit	R/W	Default	Description
7	RW	0	Bug H: In special case it send a msg with initiator_addr=follower_addr, it should not consider itself a follower, as it should know itself being the sender. 0=Apply the fix. 1=Disable the fix. Original behavior.
6	RW	0	Bug G: At start bit, cec_rx_monitor's should also check cec_data_out when deciding whether the current transaction it is initiated by itself or not. 0=Apply the fix. 1=Disable the fix. Original behavior.
5	RW	0	Bug G: cec_tx_frame_composer only update broadcast_msg during IDLE, would cause it to be incorrect if the follower_adr is updated outside the IDLE period. 0=Apply the fix. 1=Disable the fix. Original behavior.
4	RW	1	Bug E: Line error handling. Before it knows it is a follower, it does not detect line error. The probably is correct behavior. But I make it an option to detect line error, even before it knows to be a follower. However, this bit is default to be 1, as it is likely not to be a bug. 0=Apply the fix. 1=Disable the fix. Original behavior.
3	RW	0	Bug D: cec_tx_signal_free_time_ctrl use (bit_start&last_bit) to start counting for observing free time rule. But bit_start is too wide, (it basically is the inversion of the cec_data_in with the exception of start_bit). And last_bit straddles from the previous bit's sampling point to current bit's sampling point. If the previous CEC bit is logic0, then bit_start for this previous CEC bit is extra wide that it also makes (bit_start&last_bit) condition to true, it shifts the free time counter start point, thus making free time in-accurate. 0=Apply the fix. 1=Disable the fix. Original behavior.
2	RW	0	Bug C: cec_tx_timing_ctrl's status signal msg_tx. If previous transmit has failed, the status should be TX_ERROR. However, a transaction on CEC line will

Bit	R/W	Default	Description
			change this status to TX_DONE, even if the transaction does not involve our CEC. So if we don't read this status fast enough, we may not safely know whether the last transmit was successful or not. 0=Apply the fix. 1=Disable the fix. Original behavior.
1	RW	0	Bug B: cec_rx_ser2par's couldn't differentiate the two scenarios as below: 1. a new message following previous message that it is not a follower; 2. subsequent message bytes within a message that it is a follower. So for scenario 1, it treats it as subsequent bytes of the previous message, so it will not update follower address field, therefore won't recognize that itself is to be the follower, so it will not ACK to the sender.. 0=Apply the fix. 1=Disable the fix. Original behavior.
0	RW	0	Bug A: cec_rx_monitor's should also check cec_data_out when deciding whether the current transaction it is initiated by itself or not. 0=Apply the fix. 1=Disable the fix. Original behavior.

Table 9-208 BUGFIX\_DISABLE\_1

Bit	R/W	Default	Description
7:0	RW	0x0	Reserved.

Table 9-209 RX\_MSG\_HEADER

Bit	R/W	Default	Description
7:4	R	0x0	RX msg Initiator address.
3:0	R	0x0	RX msg Follower address.

Table 9-210 RX\_MSG\_OPCODE

Bit	R/W	Default	Description
7:0	R	0x0	RX msg Op-code.

Table 9-211 RX\_MSG\_OP1 - RX\_MSG\_OP14

Bit	R/W	Default	Description
7:0	R	0x0	RX msg 1st – 14th operand.

Table 9-212 RX\_MSG\_LENGTH

Bit	R/W	Default	Description
3:0	R	0x0	Length in bytes of Rx msg. Excluding Header byte.

**Table 9-213 RX\_MSG\_STATUS**

Bit	R/W	Default	Description
1:0	R	0x0	Status of the received message. 0=RX_IDLE. No transaction. 2=RX_DONE. Message has been received successfully. 1,3=Reserved.

**Table 9-214 RX\_NUM\_MSG**

Bit	R/W	Default	Description
3:0	R	0x0	Number of received messages in Rx buffer.

**Table 9-215 TX\_MSG\_STATUS**

Bit	R/W	Default	Description
1:0	R	0x0	Status of the transmitted message. 0=TX_IDLE. No transaction. 1=TX_BUSY. Transmitter is busy. 2=TX_DONE. Message has been successfully transmitted. 3=TX_ERROR. Message has been transmitted with error.

**Table 9-216 STAT\_0\_0**

Bit	R/W	Default	Description
7	R	0x0	rx_msg_status[0]. rx_msg_status[1:0]: 0=RX_IDLE. No transaction. 2=RX_DONE. Message has been received successfully. 1,3=Reserved.
6:4	R	0x0	cec_ifc_rx_state. 0=IDLE. Idle state when reset. 1=WAIT4DATA_VALID. Send Rx buffer write enable. 2=SEND_WRITE_BUF. Send Rx buffer write enable. 3=WAIT4NXT_BYTE. Wait for the next byte to be received. 4=WAIT4SEND_NXT_BYTE. Wait for Tx_frame_composer to request for the next byte. 5=SEND_NXT_BYTE. Send the next byte of the message. 6=WAIT4END_OF_MSG. Wait for the whole message to be received. 7=WAIT4BUF_NOT_FULL. Wait for the Rx buffer to become not full.
3:2	R	0x0	rx_msg_rd_state. 0=IDLE_RD. Idle state when reset and when buffer empty. 1=READ_CUR_MSG. The host is reading the current message. 2=WAIT4NXT_READ. Wait for the next read. 3=Reserved.
1:0	R	0x0	rx_msg_wr_state. 0=IDLE_WR. Idle state when reset and when buffer empty. 1=WRITE_CUR_BYTE. Write to current buffer. 2=WAIT4NXT_WRITE. Wait for the next write. 3=WAIT4NXT_MSG. Current message was received, wait for next message.

Table 9-217 STAT\_0\_1

Bit	R/W	Default	Description
7	R	0x0	cec_ifc_tx_state[0]. cec_ifc_tx_state[2:0]: 0=IDLE. Idle state when reset. 1=SEND_ACK_GET_MSG. Send acknowledge to get_msg signal. 2=GET_HEADER. Get message header. 3=WAIT4SEND_NXT_BYTE. Wait for Tx_frame_composer to request for the next byte. 4=SEND_NXT_BYTE. Send the next byte of the message. 5=SEND_DIN_ENB. Send din_enb signal. 6=WAIT4SEND_LAST_BYTE. Send next FIFO read enable. 7=RETRANSMIT_MSG. Wait for retransmit_message low.
6:5	R	0x0	msg_status_tx_cec. 0=TX_IDLE. No transaction. 1=Reserved. 3=TX_DONE. Message has been successfully transmitted. 4=TX_ERROR. Message has been transmitted with error.
4	R	0x0	tx_msg_buf_state. 0=IDLE. Idle state when reset and when buffer empty. 1=TX_ON_GOING. The data is kept in the buffer during the transmission.
3:1	R	0x0	host_ifc_rx_state. 0=IDLE. Idle state when reset. 1=WAIT4_HOST_ACK. A message was written in the Rx buffer and the host has to acknowledge reading it. 2=WAIT4_HOST_NO_OP. Wait for RX_NO_OP from the host before reading the next message. 3=WAIT4NOT_RX_DONE. Wait for msg_stat_rx_buf to be not RX_DONE. 4=WAIT4ACK_RX_DIS. Wait for CEC interface to acknowledge the receive disable.
0	R	0x0	rx_msg_status[1].

Table 9-218 STAT\_0\_2

Bit	R/W	Default	Description
7	R	0x0	frame_compose_state[0]. frame_compose_state[3:0]: 0=IDLE. Idle state when reset. 1=SEND_REQUEST. Send request to the arbiter for transmission of a new frame. 2=SEND_START_BIT. If Tx granted tell TX_timing controller to send Start Bit. 3=WAIT4SEND_FIRST_BIT. Wait for signal send_nxt_bit from the TX_timing controller. 4=SEND_DIN_7. Send bit 7 of din bus. 5=SEND_DIN_6. Send bit 6 of din bus. 6=SEND_DIN_5. Send bit 5 of din bus. 7=SEND_DIN_4. Send bit 4 of din bus. 8=SEND_DIN_3. Send bit 3 of din bus. 9=SEND_DIN_2. Send bit 2 of din bus. 10=SEND_DIN_1. Send bit 1 of din bus. 11=SEND_DIN_0. Send bit 0 of din bus. 12=EOM_TRANSMISSION. End of Message is transmitted. 13=WAIT4ACK_SEND. Wait for ACK to be sent.
6:4	R	0x0	host_ifc_tx_state. 0=IDLE. Idle state when reset. 1=SEND_SAMP_DATA. Assert samp_data signal. 2=SEND_GET_MSG. Send get_msg to CEC interface. 3=WAIT4RD_NXT_BYTE_1. Increase rd_addr every read_nxt_byte pulse and go to WAIT4RD_NXT_BYTE_0. 4=WAIT4RD_NXT_BYTE_0. If rd_addr_r = msg_length go to WAIT4TX_DONE. 5=WAIT4TX_DONE. Wait for TX_DONE and go to IDLE.

Bit	R/W	Default	Description
			6=SEND_ABORT. Send abort to CEC interface. 7=CHECK_TX_NO_OP. Wait for TX_NO_OP command from the host.
3:2	R	0x0	tx_msg_status. 0=TX_IDLE. No transaction. 1=TX_BUSY. Transmitter is busy. 2=TX_DONE. Message has been successfully transmitted. 3=TX_ERROR. Message has been transmitted with error.
1:0	R	0x0	cec_ifc_tx_state[2:1].

Table 9-219 STAT\_0\_3

Bit	R/W	Default	Description
7:6	R	0x0	signal_free_time_state. 0=IDLE. Idle state when reset. 1=FREE_TIME_MEASURE. Start measuring the Signal Free Time. 2=FREE_TIME_MEASURE_END. Wait for end_free_time_req signal.
5:3	R	0x0	tx_arbiter_state. 0=IDLE. Idle state when reset. 1=CHECK_SIGNAL_FREE_TIME. Send request to the arbiter for transmission of a new frame. 2=GRANT_TRANSMISSION. If Tx granted tell TX_timing controller to send Start Bit. 3=ABORT_TRANSMISSION. If there is a contention on CEC bus abort transmission. 4=END_ARBITRATION. Current device got control of the CEC bus as an initiator.
2:0	R	0x0	frame_compose_state[3:1].

Table 9-220 STAT\_1\_0

Bit	R/W	Default	Description
7:6	R	0x0	ack_send_state[1:0]. ack_send_state[2:0]: 0=IDLE_ACK. Idle state when reset. 1=SEND_ACK_LOW. Put the CEC bus on low impedance for a '0', ACK transmission. 2=SEND_ACK_HIGH. Put the CEC bus on high impedance for a '0', ACK transmission. 3=CHECK_LINE_ERROR. Check send_line_error signal again. 4=SEND_LINE_ERROR. Put the CEC bus on low impedance for 3.6 ms to signalize that a CEC line error occurred.
5:2	R	0x0	tx_timing_ctrl_state. 0=IDLE. Idle state when reset. 1=WAIT_FOR_GRANT. Wait for grant from the arbiter. 2=SEND_START_BIT_LOW. If Tx granted send Start Bit low. 3=SEND_START_BIT_HIGH. Start Bit high. 4=SEND_1_LOW. Put the CEC bus on low impedance for a '1' transmission. 5=SEND_0_LOW. Put the CEC bus on low impedance for a '0' transmission. 6=SEND_1_HIGH. Put the CEC bus on high impedance for a '1' transmission. 7=SEND_0_HIGH. Put the CEC bus on high impedance for a '0' transmission. 8=CHECK_ACK. Check acknowledge. 9=DATA_BLOCK_OK. Header/Data block was acknowledged, receive the next one. 10=REQUEST_RETRANSMIT. A valid frame was lost and therefore it should be re-transmitted. 11=WAIT4TX_NO_OP. Wait for host to send tx_cmd = TX_NO_OP.
1:0	R	0x0	msg_tx. 0=TX_IDLE. No transaction. 1=Reserved. 2=TX_DONE. Message has been successfully transmitted. 3=TX_ERROR. Message has been transmitted with error.

Table 9-221 STAT\_1\_1

Bit	R/W	Default	Description
7:6	R	0x0	rx_ser2par_state[1:0]. rx_ser2par_state[3:0]: 0=IDLE. Idle state when reset. 1=RECEIVE_EIGHTH_BIT. Receive the eighth data bit. 2=RECEIVE_SEVENTH_BIT. Receive the seventh data bit. 3=RECEIVE_SIXTH_BIT. Receive the sixth data bit. 4=RECEIVE_FIFTH_BIT. Receive the fifth data bit. 5=RECEIVE_FOURTH_BIT. Receive the fourth data bit. 6=RECEIVE_THIRD_BIT. Receive the third data bit. 7=RECEIVE_SECOND_BIT. Receive the second data bit. 8=RECEIVE_FIRST_BIT. Receive the first data bit. 9=RECEIVE_EOM_BIT. Receive the EOM bit. 10=RECEIVE_ACK_BIT. Receive the ACK bit.
5:4	R	0x0	msg_rx. 0=RX_IDLE. No transaction. 1=RX_BUSY. Receiver is busy. 2=RX_DONE. Message has been received successfully. 3=RX_ERROR. Message has been received with error.



Bit	R/W	Default	Description
3:1	R	0x0	rx_monitor_state. 0=IDLE. Idle state when reset. 1=CHECK_START_BIT_LOW. Check the start bit low period. 2=CHECK_START_BIT_HIGH. Check the start bit high period. 3=CHECK_BIT_LOW. Check data bit low period. 4=CHECK_BIT_HIGH. Check data bit high period. 5=DELAY_OUTPUT_SIGNAL. The del_cntr counter creates a desired delay for an output signal.
0	R	0x0	ack_send_state[2].

Table 9-222 STAT\_1\_2

Bit	R/W	Default	Description
1:0	R	0x0	rx_ser2par_state[3:2].

### CEC Bit Timing Parameters

The following table lists the appropriate values to apply to the timing parameters for two scenarios, respectively.

Scenario 1, original behavior, when the internal beat rate is once per 0.01ms;

Scenario 2, when the input clock = internal beat rate = 32768Hz.

The timing parameter values apply to scenario 2 – 32768Hz by default.

Table 9-223 CEC Bit Timing Parameter Setting

Name	Scen1 100KHz	Scen2 32768Hz	Description
Quiescent_25ms[11:0]	0x9C4	0x333	No more activity, >=25 ms
StartBitMinL2H_3ms5[8:0]	0x15E	0x070	Earliest time for Start Bit L->H, 3.5 ms
StartBitMaxL2H_3ms9 [8:0]	0x186	0x07C	Latest time for Start Bit L->H, 3.9 ms
StartBitMinH_0ms6[8:0]	0x03C	0x011	Min Start Bit High pulse width, 0.6 ms
StartBitMaxH_1ms0[8:0]	0x064	0x01D	Max start Bit High pulse width, 1.0 ms
StartBitMinTotal_4ms3 [9:0]	0x1AE	0x08A	Min Start Bit total time, 4.3 ms
StartBitMaxTotal_4ms7 [9:0]	0x1D6	0x096	Max Start Bit total time, 4.7 ms
Logic1MinL2H_0ms4[8:0]	0x028	0x00A	Earliest time for logic1 L->H, T1=0.4 ms
Logic1MaxL2H_0ms8[8:0]	0x050	0x016	Latest time for logic1 L->H, T2=0.8 ms
Logic0MinL2H_1ms3[8:0]	0x082	0x027	Earliest time for logic0 L->H, T5=1.3 ms
Logic0MaxL2H_1ms7[8:0]	0x0AA	0x034	Latest time for logic0 L->H, T6=1.7 ms
LogicMinTotal_2ms05[9:0]	0x0CD	0x040	Earliest time for next bit, T7=2.05 ms
LogicMaxHigh_2ms8[8:0]	0x118	0x059	Go to IDLE if High for >=2.8 ms
LogicErrLow_3ms4[8:0]	0x154	0x06C	Assert Low time error when active, >=3.4 ms

Name	Scen1 100KHz	Scen2 32768Hz	Description
NomSmpPoint_1ms05 [7:0]	0x69	0x1F	Nominal sample time, 1.05 ms
DelCntr_LogicErrLow[2:0]	0x5	0x5	Initial value for del_cntr on LogicErrLow.
DelCntr_LogicErrHigh[2:0]	0x3	0x3	Initial value for del_cntr on LogicErrHigh.
TxTime_17ms[15:0]	0x06A4	0x022C	17 ms
TxTime_2Bit[15:0]	0x01F0	0x00A2	Time for 7-5 data bit, 4.96 ms
TxTime_4Bit[15:0]	0x03D0	0x013F	Time for 7-3 data bit, 9.76 ms
StartBitNomL2H_3ms7 [8:0]	0x172	0x078	Nominal time for Start Bit L->H, 3.7 ms
StartBitNomH_0ms8[8:0]	0x050	0x019	Nominal Start Bit High pulse width, 0.8 ms
Logic1NomL2H_0ms6[8:0]	0x03C	0x013	Nominal time for logic1 L->H, 0.6 ms
Logic0NomL2H_1ms5[8:0]	0x096	0x030	Nominal time for logic0 L->H, 1.5 ms
Logic1NomH_1ms8[8:0]	0x0B4	0x03A	Nominal Logic1 High pulse width, 1.8 ms
Logic0NomH_0ms9[8:0]	0x05A	0x01D	Nominal Logic0 High pulse width, 0.9 ms
LogicErrLow_3ms6[8:0]	0x168	0x075	Assert Low time error when active, >=3.6 ms
ChkContention_0ms1[3:0]	0xA	0x3	Check for bus contention 0.1ms=100us after H
PrepareNxtBit_0ms05[8:0]	0x005	0x005	Prepare next bit, 0.05ms before end of prev H
NomSmpAckPoint_0ms45 [7:0]	0x2D	0x0F	Nominal sample time 1.05ms=0.6(Logi- c1NomL2H_0ms6)+0.45
Ack0NomL2H_1ms5[8:0]	0x096	0x02C	Nominal time for Ack0 L->H, 1.5 ms

### 9.1.6.3 CECB Registers

#### Register Access

Accessing CEC\_B internal registers is by indirectly method through a register.

```
#define CECB_RW_REG (0xfe044000 + (0x011 << 2))
```

```
void cecb_wr_only_reg (uint32_t addr, uint32_t data)
```

```
{
    uint32_t data32;

    data32 = 0;
    data32 |= (1 << 16); // [16] cec_reg_wr
    data32 |= (data << 8); // [15:8] cec_reg_wrdata
    data32 |= (addr << 0); // [7:0] cec_reg_addr
    Wr(CECB_RW_REG, data32);
} /* cecb_wr_only_reg */
```

```
uint32_t cecb_rd_reg (uint32_t addr)
```

```
{
    uint32_t data32;

    data32 = 0;
```

```

data32 |= (0 << 16); // [16] cec_reg_wr
data32 |= (0 << 8); // [15:8] cec_reg_wrdata
data32 |= (addr << 0); // [7:0] cec_reg_addr
Wr(CECB_RW_REG, data32);

data32 = ((Rd(CECB_RW_REG)) >> 24) & 0xff;

return (data32);
} /* cecb_rd_reg */

```

## Register Address

Addr	Name	RW	Function
0x00	CEC_CTRL	RW	CEC control.
0x01	CEC_CTRL2	RW	CEC control.
0x02	CEC_INTR_MASK	RW	Interrupt mask.
0x05	CEC_LADD_LOW	RW	Logical address.
0x06	CEC_LADD_HIGH	RW	Logical address.
0x07	CEC_TX_CNT	RW	Tx byte count.
0x08	CEC_RX_CNT	R	Rx byte count.
0x09	CEC_STAT0	R	Internal status.
0x10	CEC_TX_DATA00	RW	TX byte 0.
0x11	CEC_TX_DATA01	RW	TX byte 1.
0x12	CEC_TX_DATA02	RW	TX byte 2.
0x13	CEC_TX_DATA03	RW	TX byte 3.
0x14	CEC_TX_DATA04	RW	TX byte 4.
0x15	CEC_TX_DATA05	RW	TX byte 5.
0x16	CEC_TX_DATA06	RW	TX byte 6.
0x17	CEC_TX_DATA07	RW	TX byte 7.
0x18	CEC_TX_DATA08	RW	TX byte 8.
0x19	CEC_TX_DATA09	RW	TX byte 9.
0x1A	CEC_TX_DATA10	RW	TX byte 10.
0x1B	CEC_TX_DATA11	RW	TX byte 11.
0x1C	CEC_TX_DATA12	RW	TX byte 12.
0x1D	CEC_TX_DATA13	RW	TX byte 13.
0x1E	CEC_TX_DATA14	RW	TX byte 14.
0x1F	CEC_TX_DATA15	RW	TX byte 15.
0x20	CEC_RX_DATA00	R	RX byte 0.
0x21	CEC_RX_DATA01	R	RX byte 1.
0x22	CEC_RX_DATA02	R	RX byte 2.
0x23	CEC_RX_DATA03	R	RX byte 3.
0x24	CEC_RX_DATA04	R	RX byte 4.

Addr	Name	RW	Function
0x25	CEC_RX_DATA05	R	RX byte 5.
0x26	CEC_RX_DATA06	R	RX byte 6.
0x27	CEC_RX_DATA07	R	RX byte 7.
0x28	CEC_RX_DATA08	R	RX byte 8.
0x29	CEC_RX_DATA09	R	RX byte 9.
0x2A	CEC_RX_DATA10	R	RX byte 10.
0x2B	CEC_RX_DATA11	R	RX byte 11.
0x2C	CEC_RX_DATA12	R	RX byte 12.
0x2D	CEC_RX_DATA13	R	RX byte 13.
0x2E	CEC_RX_DATA14	R	RX byte 14.
0x2F	CEC_RX_DATA15	R	RX byte 15.
0x30	CEC_LOCK_BUF	RW	CEC lock register.
0x31	CEC_WAKEUPCTRL	RW	Wakeup status.

## Register Description

**Table 9-224 CEC\_CTRL2**

Bit	R/W	Default	Description
6	RW	0	curb_err_init. Bug fix: Do not signal initiator error, when it's myself who pulled down the line when functioning as a follower.
5	RW	0	en_chk_sbitlow. Bug fix – “Receive messages are ignored and not acknowledge” – to resolve bus contention during Start bit, when Tx is sending while Rx is receiving,
4:0	RW	0	rise_del_max. To support CEC line slow rise delay. 0=Support rise delay up to 1*30.5us; 1=Support rise delay up to 2*30.5us; ... N=Support rise delay up to (N+1)*30.5us.

Table 9-225 CEC\_STAT0

Bit	R/W	Default	Description
6:4	RW	0	p2s_state. 0=TIDLE; 1=SEND_SBIT; Send Start bit. Apply to initiator only. 2=SEND_DBIT; Send Data bit. Apply to initiator only. 3=SEND_EOM; Send End-Of-Message bit. Apply to initiator only. 4=SEND_ACK; ACK bit period. Apply to initiator only. 5=FBACK_ACK; Send ACK bit. Apply to follower only. 6=FBACK_RX_ERR. Signal line error. Apply to follower only.
3:0	R	0	s2p_status. 0x0=IDLE; 0x1=SBITLOWER; Start Bit's Low impedance period. 0x2=SBH; Start Bit's High impedance period. 0x5=L1LOWER; Data bit (logic 0 or 1), the period before one can decide if it's logic 0 or 1. 0x6=SMP1; Detected the data bit is logic 1. 0x7=SMP0; Detected the data bit is logic 0. 0x8=L0H; Logic 0 bit's High impedance period. 0x9=ERRLMIN; Detecting line error. 0xE=ERRLMAX. Line error detected.

## 9.2 Video Output

### 9.2.1 Overview

This section describes the SoC's VPU sub-module, including RDMA sub-module, VIU sub-module, HDMITX sub-module, and MIPI\_DSI sub-module.

### 9.2.2 VPU

VPU is display process unit, the main function is to receive data from decoder/ddr/hdmirx etc, then process the source data in order to get the high-quality video picture, and finally send out the video to the screen by HDMITX etc.

### 9.2.3 Register Description

#### 9.2.3.1 VPU Registers

##### Register Address

- VPU\_CRC\_CTRL 0xff009c04
- VPU\_RO\_CRC0 0xff009c08
- VPU\_RO\_CRC1 0xff009c0c
- VPU\_RO\_CRC2 0xff009c10
- VPU\_RO\_CRC3 0xff009c14
- VPU\_RO\_CRC4 0xff009c18
- VPU\_RO\_CRC5 0xff009c1c
- VPU\_RO\_CRC6 0xff009c20
- VPU\_INTF\_CTRL 0xff009c28
- VPU\_APB\_PROT\_CTRL 0xff009c2c
- VPU\_ENC\_ERROR 0xff009c30

- VPU\_VDIN\_PRE\_ARB\_CTRL0xff009c50
- VPU\_VDISP\_PRE\_ARB\_CTRL 0xff009c54
- VPU\_VPUARB2\_PRE\_ARB\_CTRL 0xff009c58
- VPU\_VIU\_VENC\_MUX\_CTRL0xff009c68
- VPU\_HDMI\_SETTING 0xff009c6c
- ENCI\_INFO\_READ 0xff009c70
- ENCP\_INFO\_READ 0xff009c74
- ENCT\_INFO\_READ 0xff009c78
- ENCL\_INFO\_READ 0xff009c7c
- VPU\_SW\_RESET 0xff009c80
- VPU\_CLK\_GATE 0xff009c8c
- VPU\_MEM\_PD\_REG0 0xff009c94
- VPU\_MEM\_PD\_REG1 0xff009c98
- VPU\_HDMI\_DATA\_OVR0xff009c9c
- VPU\_VPU\_PWM\_V0 0xff007380
- VPU\_VPU\_PWM\_V1 0xff007384
- VPU\_VPU\_PWM\_V2 0xff007388
- VPU\_VPU\_PWM\_V3 0xff00738c
- VPU\_VPU\_PWM\_H0 0xff007390
- VPU\_VPU\_PWM\_H1 0xff007394
- VPU\_VPU\_PWM\_H2 0xff007398
- VPU\_VPU\_PWM\_H3 0xff00739c
- VPU\_VPU\_3D\_SYNC1 0xff0073a0
- VPU\_VPU\_3D\_SYNC2 0xff0073a4
- VPU\_HDMI\_FMT\_CTRL0xff009d0c
- VPU\_VDIN\_ASYNC\_HOLD\_CTRL 0xff009d10
- VPU\_VDISP\_ASYNC\_HOLD\_CTRL0xff009d14
- VPU\_VPUARB2\_ASYNC\_HOLD\_CTRL 0xff009d18
- VPU\_ARB\_URG\_CTRL 0xff009d1c
- VPU\_422TO444\_RST 0xff009d28
- VPU\_422TO444\_CTRL0 0xff009d2c
- VPU\_422TO444\_CTRL1 0xff009d30
- VPU\_VIU\_ASYNC\_MASK 0xff009e04
- VPU\_VDIN\_MISC\_CTRL 0xff009e08
- VPU\_VIU\_VDIN\_IF\_MUX\_CTRL 0xff009e0c
- VPU\_VIU2VDIN1\_HDN\_CTRL 0xff009e10
- VPU\_VENCX\_CLK\_CTRL 0xff009e14
- VPU\_RDARB\_MODE\_L1C1 0xff009e40
- VPU\_RDARB\_REQEN\_SLV\_L1C1 0xff009e44
- VPU\_RDARB\_WEIGH0\_SLV\_L1C10xff009e48
- VPU\_RDARB\_WEIGH1\_SLV\_L1C10xff009e4c
- VPU\_WRARB\_MODE\_L1C1 0xff009e50
- VPU\_WRARB\_REQEN\_SLV\_L1C1 0xff009e54
- VPU\_WRARB\_WEIGH0\_SLV\_L1C10xff009e58

- VPU\_WRARB\_WEIGH1\_SLV\_L1C10xff009e5c
- VPU\_RDWR\_ARB\_STATUS\_L1C1 0xff009e60
- VPU\_RDARB\_MODE\_L1C2 0xff009e64
- VPU\_RDARB\_REQEN\_SLV\_L1C2 0xff009e68
- VPU\_RDARB\_WEIGH0\_SLV\_L1C20xff009e6c
- VPU\_RDWR\_ARB\_STATUS\_L1C2 0xff009e70
- VPU\_RDARB\_MODE\_L2C1 0xff009e74
- VPU\_RDARB\_REQEN\_SLV\_L2C1 0xff009e78
- VPU\_RDARB\_WEIGH0\_SLV\_L2C10xff009e7c
- VPU\_RDARB\_WEIGH1\_SLV\_L2C10xff009e80
- VPU\_RDWR\_ARB\_STATUS\_L2C1 0xff009e84
- VPU\_WRARB\_MODE\_L2C1 0xff009e88
- VPU\_WRARB\_REQEN\_SLV\_L2C1 0xff009e8c
- VPU\_WRARB\_WEIGH0\_SLV\_L2C10xff009e90
- VPU\_ASYNC\_RD\_MODE0 0xff009e94
- VPU\_ASYNC\_RD\_MODE1 0xff009e98
- VPU\_ASYNC\_RD\_MODE2 0xff009e9c
- VPU\_ASYNC\_RD\_MODE3 0xff009ea0
- VPU\_ASYNC\_RD\_MODE4 0xff009ea4
- VPU\_ASYNC\_WR\_MODE0 0xff009ea8
- VPU\_ASYNC\_WR\_MODE1 0xff009eac
- VPU\_ASYNC\_WR\_MODE2 0xff009eb0
- VPU\_ASYNC\_STAT 0xff009eb4
- VPU\_HDMI\_DITH\_01\_04 0xff009fc0
- VPU\_HDMI\_DITH\_01\_15 0xff009fc4
- VPU\_HDMI\_DITH\_01\_26 0xff009fc8
- VPU\_HDMI\_DITH\_01\_37 0xff009fcc
- VPU\_HDMI\_DITH\_10\_04 0xff009fd0
- VPU\_HDMI\_DITH\_10\_15 0xff009fd4
- VPU\_HDMI\_DITH\_10\_26 0xff009fd8
- VPU\_HDMI\_DITH\_10\_37 0xff009fdc
- VPU\_HDMI\_DITH\_11\_04 0xff009fe0
- VPU\_HDMI\_DITH\_11\_15 0xff009fe4
- VPU\_HDMI\_DITH\_11\_26 0xff009fe8
- VPU\_HDMI\_DITH\_11\_37 0xff009fec
- VPU\_HDMI\_DITH\_CNTL 0xff009ff0
- VPU\_VENCL\_DITH\_CTRL 0xff0073c0
- VPU\_VENCL\_DITH\_LUT\_1 0xff0073c4
- VPU\_VENCL\_DITH\_LUT\_2 0xff0073c8
- VPU\_VENCL\_DITH\_LUT\_3 0xff0073cc
- VPU\_VENCL\_DITH\_LUT\_4 0xff0073d0
- VPU\_VENCL\_DITH\_LUT\_5 0xff0073d4
- VPU\_VENCL\_DITH\_LUT\_6 0xff0073d8
- VPU\_VENCL\_DITH\_LUT\_7 0xff0073dc

- VPU\_VENCL\_DITH\_LUT\_8 0xff0073e0
- VPU\_VENCL\_DITH\_LUT\_9 0xff0073e4
- VPU\_VENCL\_DITH\_LUT\_100 0xff0073e8
- VPU\_VENCL\_DITH\_LUT\_110 0xff0073ec
- VPU\_VENCL\_DITH\_LUT\_120 0xff0073f0

## Register Description

**Table 9-226 VPU\_CRC\_CTRL**

Bit(s)	R/W	Default	Description
10	R/W	0	vpu axi read channel 2 bist enable
9	R/W	0	vpu axi read channel 1 bist enable
8	R/W	0	vpu axi read channel 0 bist enable
4	R/W	0	vpu axi write channel 1 crc start
3	R/W	0	vpu axi write channel 0 crc start
2	R/W	0	vpu axi read channel 2 crc start
1	R/W	0	vpu axi read channel 1 crc start
0	R/W	0	vpu axi read channel 0 crc start

**Table 9-227 VPP\_RO\_CRC0**

Bit(s)	R/W	Default	Description
31-0	R	0	vpu axi read 0 crc

**Table 9-228 VPP\_RO\_CRC1**

Bit(s)	R/W	Default	Description
31-0	R	0	vpu axi read 1 crc

**Table 9-229 VPP\_RO\_CRC2**

Bit(s)	R/W	Default	Description
31-0	R	0	vpu axi read 2 crc

**Table 9-230 VPP\_RO\_CRC3**

Bit(s)	R/W	Default	Description
31-0	R	0	vpu axi write 0 cmd crc

**Table 9-231 VPP\_RO\_CRC4**

Bit(s)	R/W	Default	Description
31-0	R	0	vpu axi write 0 data crc



**Table 9-232 VPP\_RO\_CRC5**

Bit(s)	R/W	Default	Description
31-0	R	0	vpu axi write 1 cmd crc

**Table 9-233 VPP\_RO\_CRC6**

Bit(s)	R/W	Default	Description
31-0	R	0	vpu axi write 1 data crc

**Table 9-234 VPP\_INTF\_CTRL**

Bit(s)	R/W	Default	Description
18	R/W	1	vpu axi write 1 enable
17	R/W	0	vpu axi write 1 gate clock auto
16	R/W	0	vpu axi write 1 disable clock
14	R/W	1	vpu axi write 0 enable
13	R/W	0	vpu axi write 0 gate clock auto
12	R/W	0	vpu axi write 0 disable clock
10	R/W	1	vpu axi read 2 enable
9	R/W	0	vpu axi read 2 gate clock auto
8	R/W	0	vpu axi read 2 disable clock
6	R/W	1	vpu axi read 1 enable
5	R/W	0	vpu axi read 1 gate clock auto
4	R/W	0	vpu axi read 1 disable clock
2	R/W	1	vpu axi read 0 enable
1	R/W	0	vpu axi read 0 gate clock auto
0	R/W	0	vpu axi read 0 disable clock

**Table 9-235 VPP\_APB\_PROT\_CTRL**

Bit(s)	R/W	Default	Description
21-2	R/W	0	hold number
1	R/W	0	protect enable
0	R/W	0	soft reset

**Table 9-236 VPP\_ENC\_ERROR**

Bit(s)	R/W	Default	Description
31-24	R	0	RO enci error count
23-16	R	0	RO encp error count
15-8	R	0	RO encl error count
2	R/W	1	encl afifo error count enable
1	R/W	1	encp afifo error count enable
0	R/W	1	enci afifo error count enable

**Table 9-237 VPU\_VDIN\_PRE\_ARB\_CTRL**

Bit(s)	R/W	Default	Description
31	R/W	0	VDIN_PREARB_SOFT_RESET: 1=SW reset vdin_mmc_pre_arb.
30-18	R	0	Reserved.
17	R/W	1	VDIN_ACG_EN: 1=Enable auto-clock gating of vdin_mmc_pre_arb.
16	R/W	0	VDIN_DISABLE_CLK: 1=Disalbe the clock to vdin_mmc_pre_arb.
15-0	R/W	0xffff	VDIN_REQ_EN: vdin_mmc_pre_arb request enable.

**Table 9-238 VPU\_VDISP\_PRE\_ARB\_CTRL**

Bit(s)	R/W	Default	Description
31	R/W	0	VDISP_PREARB_SOFT_RESET: 1=SW reset vdisp_mmc_pre_arb.
30-18	R	0	Reserved.
17	R/W	1	VDISP_ACG_EN: 1=Enable auto-clock gating of vdisp_mmc_pre_arb.
16	R/W	0	VDISP_DISABLE_CLK: 1=Disalbe the clock to vdisp_mmc_pre_arb.
15-0	R/W	0xffff	VDISP_REQ_EN: vdisp_mmc_pre_arb request enable.

**Table 9-239 VPU\_VPUARB2\_PRE\_ARB\_CTRL**

Bit(s)	R/W	Default	Description
31	R/W	0	VPUARB2_PREARB_SOFT_RESET: 1=SW reset vpuarb2_mmc_pre_arb.
30-18	R	0	Reserved.
17	R/W	1	VPUARB2_ACG_EN: 1=Enable auto-clock gating of vpuarb2_mmc_pre_arb.
16	R/W	0	VPUARB2_DISABLE_CLK: 1=Disalbe the clock to vpuarb2_mmc_pre_arb.
15-0	R/W	0xffff	VPUARB2_REQ_EN: vpuarb2_mmc_pre_arb request enable.

Table 9-240 VPU\_VIU\_VENC\_MUX\_CTRL

Bit(s)	R/W	Default	Description
20	R/W	0	Viu_sync_mode_en 1:Viu_sync_mode_en enable// ENCL/P/I vsync send to both viu1 and viu2 Bit[1:0] select data fbetween viu1 and ENCL/P/I Bit[5:4] select ctrl singal(vsync) between ENCL/P/I and viu1. Bit[3:2] select data and vsync between viu2 and ENCL/P/I 0:Viu_sync_mode_en disable// Bit[1:0] select data and vsync fbetween viu1 and ENCL/P/I/Bit[3:2] select data and vsync between viu2 and ENCL/P/I/Bit[5:4] unuse
17-16	W	0	RASP DPI CLOCK SEL : 00 : cph1 01 : cph2 10/11 : cph3
15:6	W	0	reserved
5-4			VIU1_SEL_VENC_CTRL: Select which one of the encl/P/T/L control singal that Viu1 connects to when viu_sync_model enable.unuse when viu_sync_model disable. 0: ENCL 1: ENCI 2: ENCP 3: ENCT
3-2	R/W	0	VIU2_SEL_VENC: Select which one of the encl/P/T/L that Viu2 connects to (both vsync and data). 0: ENCL 1: ENCI 2: ENCP 3: ENCT
1-0	R/W	0	VIU1_SEL_VENC: Select which one of the encl/P/T/L that Viu1 connects to. On-ly select data when viu_sync_model enable,select both data and vsync when viu_sync_model disable, 0: ENCL 1: ENCI 2: ENCP 3: ENCT

Table 9-241 VPU\_HDMI\_SETTING

Bit(s)	R/W	Default	Description
15-12	R/W	0	RD_RATE: Read rate to the async FIFO between VENC and HDMI. 0: One read every rd_clk 1: One read every 2 rd_clk 2: One read every 3 rd_clk ... 15: One read every 16 rd_clk
11-8	R/W	0	WR_RATE: Write rate to the async FIFO between VENC and HDMI. 0: One write every wr_clk 1: One write every 2 wr_clk 2: One write every 3 wr_clk ... 15: One write every 16 wr_clk
7-5	R/W	0	DATA_COMP_MAP: Input data is CrYCr(BRG), map the output data to desired format: 0: output CrYCb (BRG) 1: output YCbCr (RGB) 2: output YCrCb (RBG) 3: output CbCrY (GBR) 4: output CbYCr (GRB) 5: output CrCbY (BGR) 6,7: Reserved

Bit(s)	R/W	Default	Description
4	R/W	0	INV_DVI_CLK: If true, invert the polarity of clock output to external DVI interface. (NOT internal HDMI).
3	R/W	0	INV_VSYNC: If true, invert the polarity of VSYNC input from VENC
2	R/W	0	INV_HSYNC: If true, invert the polarity of HSYNC input from VENC
1-0	R/W	0	SRC_SEL: Select which HDMI source from between ENCI and ENCP. 2'b00: Disable HDMI source 2'b01: Select ENCI data to HDMI 2'b10: Select ENCP data to HDMI

**Table 9-242 ENCI\_INFO\_READ**

Bit(s)	R/W	Default	Description
31-29	R	0	Current ENCI field status.
28-25	R	0	Reserved
24-16	R	0	Current ENCI line counter status.
15-11	R	0	Reserved
10-0	R	0	Current ENCI pixel counter status.

**Table 9-243 ENCP\_INFO\_READ**

Bit(s)	R/W	Default	Description
31-29	R	0	Current ENCP field status.
28-16	R	0	Current ENCP line counter status.
15-13	R	0	Reserved
12-0	R	0	Current ENCP pixel counter status.

**Table 9-244 ENCT\_INFO\_READ**

Bit(s)	R/W	Default	Description
31-29	R	0	Current ENCT field status.
28-16	R	0	Current ENCT line counter status.
15-13	R	0	Reserved
12-0	R	0	Current ENCT pixel counter status.

**Table 9-245 ENCL\_INFO\_READ**

Bit(s)	R/W	Default	Description
31-29	R	0	Current ENCL field status.
28-16	R	0	Current ENCL line counter status.
15-13	R	0	Reserved
12-0	R	0	Current ENCL pixel counter status.

Table 9-246 VPU\_SW\_RESET

Bit(s)	R/W	Default	Description
3	R/W	0	vpuarb2_mmc_arb_rst_n
2	R/W	0	vdisp_mmc_arb_rst_n
1	R/W	0	vdin_mmc_arb_rst_n
0	R/W	0	viu_rst_n

Table 9-247 VPU\_D2D3\_MMC\_CTRL

Bit(s)	R/W	Default	Description
30	R/W	0	d2d3_depr_req_sel, 0:vdisp_pre_arb, 1: vpuarb2_pre_arb
27-22	R/W	0x3f	d2d3_depr_brst_num
21-16	R/W	0x2d	d2d3_depr_id
14	R/W	0x0	d2d3_depwr_req_sel, 0: vdin_pre_arb, 1: vdisp_pre_arb
11-6	R/W	0x3f	d2d3_depwr_brst_num
5-0	R/W	0x2e	d2d3_depwr_id

Table 9-248 VPU\_CONT\_MMC\_CTRL

Bit(s)	R/W	Default	Description
30	R/W	0x0	mtn_contrd_req_sel, 0:vdisp_post_arb, 1: vpuarb2_pre_arb
27-22	R/W	0x3f	mtn_contrd_brst_num
21-16	R/W	0x2b	mtn_contrd_id
14	R/W	0x0	mtn_contwr_req_sel, 0: vdisp_post_arb, 1: vpuarb2_pre_arb
11-6	R/W	0x3f	mtn_contwr_brst_num
5-0	R/W	0x2c	mtn_contwr_id

Table 9-249 VPU\_CLK\_GATE

Bit(s)	R/W	Default	Description
30-18	R/W	0x0	Reserved
17	R/W	0x1	Clk_b control register no latch
16	R/W	0x1	Clk_vib enable
15	R/W	0x1	Gvapbclk enable
14	R/W	0x1	Clk mpeg vlock enable
13	R/W	0x1	Reserved
12	R/W	0x1	Venc_dac_process_clk enable
11	R/W	0x1	Venc_i_top enable
10	R/W	0x1	Venci_int enable

Bit(s)	R/W	Default	Description
9:8	R/W	0x11	Clk_vib latch sync source select 00: di2ldim_go_field, 01: post_frame_rst;10: pre_frame_rst, 11:viu_go_field
7	R/W	0x1	Reserved
6	R/W	0x1	Vpu_misc_clk enable
5	R/W	0x1	Venc_l_top enable
4	R/W	0x1	Venc_l_int enable
3	R/W	0x1	Venc_p_int enable
2	R/W	0x1	Reserved
1	R/W	0x1	Vi_top clock enable
0	R/W	0x1	Venc_p_top enable

Table 9-250 VPU\_HDMI\_DATA\_OVR

Bit(s)	R/W	Default	Description
31	R/W	0	DATA_OVR_EN: Control if override HDMI input data with DATA_OVR[29:0], for display e.g. black or blue screen. 0: No override 1: Enable override
30	R	0	Security override
29-0	R/W	0	DATA_OVR: programmable pixel data value for override.

Table 9-251 VPU\_VPU\_PWM\_V0

Bit(s)	R/W	Default	Description
31	R/W	0	reg_vpu_pwm_inv, 1: invert the pwm signal, active low
30-29	R/W	0	reg_vpu_pwm_src_sel, 00: encl, enct, encp
28-16	R/W	0	reg_vpu_pwm_v_end0
15-14	R/W	0	reg_vpu_pwm_setting_latch_mode
13	R/W	1	reg_vpu_pwm_vs_inv
12-0	R/W	0	reg_vpu_pwm_v_start0

Table 9-252 VPU\_VPU\_PWM\_V1

Bit(s)	R/W	Default	Description
28-16	R/W	0	reg_vpu_pwm_v_end1
12-0	R/W	0	reg_vpu_pwm_v_start1

**Table 9-253 VPU\_VPU\_PWM\_V2**

Bit(s)	R/W	Default	Description
28-16	R/W	0	reg_vpu_pwm_v_end2
12-0	R/W	0	reg_vpu_pwm_v_start2

**Table 9-254 VPU\_VPU\_PWM\_V3**

Bit(s)	R/W	Default	Description
28-16	R/W	0	reg_vpu_pwm_v_end3
12-0	R/W	0	reg_vpu_pwm_v_start3

**Table 9-255 VPU\_VPU\_PWM\_H0**

Bit(s)	R/W	Default	Description
28-16	R/W	0	reg_vpu_pwm_h_end0
12-0	R/W	0	reg_vpu_pwm_h_start0

**Table 9-256 VPU\_VPU\_PWM\_H1**

Bit(s)	R/W	Default	Description
28-16	R/W	0	reg_vpu_pwm_h_end1
12-0	R/W	0	reg_vpu_pwm_h_start1

**Table 9-257 VPU\_VPU\_PWM\_H2**

Bit(s)	R/W	Default	Description
28-16	R/W	0	reg_vpu_pwm_h_end2
12-0	R/W	0	reg_vpu_pwm_h_start2

**Table 9-258 VPU\_VPU\_PWM\_H3**

Bit(s)	R/W	Default	Description
28-16	R/W	0	reg_vpu_pwm_h_end3
12-0	R/W	0	reg_vpu_pwm_h_start3

**Table 9-259 VPU\_VPU\_3D\_SYNC1**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_3dsync_enable, 1: enable 3d sync output
30	R/W	0	3dsync setting vsync latch
29	R/W	0	3dsync go high field polarity: 1, go high while field[0]=1

Bit(s)	R/W	Default	Description
28-16	R/W	0	reg_3dsync_v_end0
15	R/W	0	3dsync out inv
14	R/W	0	3dsync vbo out inv
13	R/W	1	Vbo 3d en, to v by one, 3d enable
12-0	R/W	0	reg_3dsync_v_start0

**Table 9-260 VPU\_VPU\_3D\_SYNC2**

Bit(s)	R/W	Default	Description
31	R/W	0	Reg_3dsync_field_bit_sel: 1. Keep 3dsync not changed for two fields, i.e. L-L-R-R-L-L-R-R (11001100) 0. Change 3dsync every field i.e. L-R-L-R (1010)
28-16	R/W	0	reg_3dsync_h_end
12-0	R/W	0	reg_3dsync_h_start

**Table 9-261 VPU\_HDMI\_FMT\_CTRL**

12 bit to 10 bit dither control register. 10 bit to 8 bit see VPU\_HDMI\_DITH\_CNTL

Bit(s)	R/W	Default	Description
21-19	R/W	0	frame count offset for B
18-16	R/W	0	frame count offset for G
15	R/W	0	hcnt hold when de valid
14	R/W	0	RGB frame count separate
13	R/W	0	dith4x4 : frame random enable
12	R/W	0	dith4x4 enable
11	R/W	0	tunnel enable for DOLBY
10	R/W	0	rounding enable
9-6	R/W	0	Cntl_hdmi_dith10 :
5	R/W	0	Cntl_hdmi_dith_md:
4	R/W	0	Cntl_hdmi_dith_en: dither 10-b to 8-b enable
3-2	R/W	0	Cntl_chroma_dnsmpl: Chroma down sample mode when convert to 422 or 420. 0 = use pixel 0; 1 = use pixel 1; 2 = use average;
1-0	R/W	0	Cntl_hdmi_vid_fmt: Control whether to convert ENCP's 444 data to 422 or 420 0 = No conversion; 1 = Convert to 422; 2 = Convert to 420;



**Table 9-262 VPU\_VDIN\_ASYNC\_HOLD\_CTRL**

Bit(s)	R/W	Default	Description
31-24	R/W	'h18	Wr_hold_num
23-16	R/W	'h10	Wr_rel_num
15-8	R/W	'h18	Rd_hold_num
7-0	R/W	'h10	Rd_rel_num

**Table 9-263 VPU\_VDISP\_ASYNC\_HOLD\_CTRL**

Bit(s)	R/W	Default	Description
31-24	R/W	'h18	Wr_hold_num
23-16	R/W	'h10	Wr_rel_num
15-8	R/W	'h18	Rd_hold_num
7-0	R/W	'h10	Rd_rel_num

**Table 9-264 VPU\_VPUARB2\_ASYNC\_HOLD\_CTRL**

Bit(s)	R/W	Default	Description
31-24	R/W	'h18	Wr_hold_num
23-16	R/W	'h10	Wr_rel_num
15-8	R/W	'h18	Rd_hold_num
7-0	R/W	'h10	Rd_rel_num

**Table 9-265 VPU\_ARB\_URG\_CTRL**

Bit(s)	R/W	Default	Description
11	R/W	0	Rdma_ddr_reg_busy to vpuarb2_urg_ctrl
10	R/W	0	Rdma_ddr_reg_busy to vdisp_urg_ctrl
9	R/W	0	Rdma_ddr_reg_busy to vdin_urg_ctrl
8	R/W	0	Vdin1_lff_urg_ctrl to vpuarb2_urg_ctrl
7	R/W	0	Vdin0_lff_urg_ctrl to vpuarb2_urg_ctrl
6	R/W	0	Vpp_off_urg_ctrl to vpuarb2_urg_ctrl
5	R/W	0	Vdin1_lff_urg_ctrl to vdisp_urg_ctrl
4	R/W	0	Vdin0_lff_urg_ctrl to vdisp_urg_ctrl
3	R/W	0	Vpp_off_urg_ctrl to vdisp_urg_ctrl
2	R/W	0	Vdin1_lff_urg_ctrl to vdin_urg_ctrl
1	R/W	0	Vdin0_lff_urg_ctrl to vdin_urg_ctrl
0	R/W	0	Vpp_off_urg_ctrl to vdin_urg_ctrl

Table 9-266 VPU\_422TO444\_RST

Bit(s)	R/W	Default	Description
3	R/W	0	Change output of viu_12bit422to10bit444_vd1 to 10bits 1:cut high 2 bits 0:cut low 2 bits
2	R/W	0	Change output of viu_12bit422to10bit444_encp to 10bits 1:cut high 2 bits 0:cut low 2 bits
1	R/W	0	Soft rst enable of viu_12bit422to10bit444_vd1 module active high
0	R/W	0	Soft rst enable of viu_12bit422to10bit444_encp module active high

Table 9-267 VPU\_422TO444\_CTRL0

Bit(s)	R/W	Default	Description
31	R/W	0	bypass_mode : active high if this bit set high ,viu_12bit422to10bit444_encp work in bypass mode, dout = din
30	/	/	reversed
29	R/W	0	clip10_mode: active high if this bit set high ,viu_12bit422to10bit444_encp work clip10_mode ,output high 10bits value of din and clip data to 10bit,then expand to 12bits according clip_lend.
28	R/W	0	Clip8_mode: active high if this bit set high ,viu_12bit422to10bit444_encp work clip8_mode ,output high 8bits value of din and clip data to 8bit,then expand to 12bits according clip_lend.
27	R/W	0	clip_lend :active high make output data to 12bits in clip10_mode/clip8_mode/scramble 1;expand bits in high bits 0: ;expand bits low bits
26	R/W	0	scramble_mode :active high viu_12bit422to10bit444_encp module get luma and chroma from 422 format data every pixel ,then reorganize data to 8bits according to reg_tunnel value(this model can change data from 422 to 444??). Luma = din[35:24] Chroma =odd_pixel?din[23:12]:din[11:0]
25	R/W	0	go_field_en 1:rst odd_pixel to 0 when go_filed come
24	R/W	0	go_line_en1:rst rst odd_pixel to 0 when go_line come
23	R/W	0	oft_rst_en 1:soft rst rst odd_pixel to 0
22	R/W	0	de_sel : input data de singal 1:chose encp require singal as de 0:close this module
17:15	R/W	0	reg_tunnel_sel_b1 : select high 4 bits for B from {luma,chroma} ,only work in scramble mode: 0:bit[3:0] 1:bit[7:3] 2:bit[11:8] 3:bit[15:12] 4:bit[19:16] 5:bit[23:20] Default:bit[23:20]
14:12	R/W	0	reg_tunnel_sel_g1 : select high 4 bits for G from {luma,chroma} ,only work in scramble mode: 0:bit[3:0] 1:bit[7:3] 2:bit[11:8] 3:bit[15:12]

Bit(s)	R/W	Default	Description
			4:bit[19:16] 5:bit[23:20] Default:bit[23:20]
11:9	R/W	0	reg_tunnel_sel_r1 : select high 4 bits for R from {luma,chroma} ,only work in scramble mode: 0:bit[3:0] 1:bit[7:3] 2:bit[11:8] 3:bit[15:12] 4:bit[19:16] 5:bit[23:20] Default:bit[23:20]
8:6	R/W	0	reg_tunnel_sel_B0 : select low 4 bits for B from {luma,chroma} ,only work in scramble mode: 0:bit[3:0] 1:bit[7:3] 2:bit[11:8] 3:bit[15:12] 4:bit[19:16] 5:bit[23:20] Default:bit[23:20]
5:3	R/W	0	reg_tunnel_sel_g0 : select low 4 bits for G from {luma,chroma} ,only work in scramble mode: 0:bit[3:0] 1:bit[7:3] 2:bit[11:8] 3:bit[15:12] 4:bit[19:16] 5:bit[23:20] Default:bit[23:20]
2:0	R/W	0	reg_tunnel_sel_r0 : select low 4 bits for R from {luma,chroma} ,only work in scramble mode: 0:bit[3:0] 1:bit[7:3] 2:bit[11:8] 3:bit[15:12] 4:bit[19:16] 5:bit[23:20] Default:bit[23:20]

Table 9-268 VPU\_422TO444\_CTRL1

Bit(s)	R/W	Default	Description
20	R/W	0	software reset
19-18	R/W	0	reg_viu2vdin_dn_ratio: down-scale ratio: 0->no scale; 1-> 1/2; 2->1/4; 3->reserved
17-16	R/W	0	reg_viu2vdinflt_mode: filter mode; 0->no filter; 1->[0 2 2 0]/4; 2->[1 1 1 1]/4; 3->[1 3 3 1]/8
15-14	R/W	0	reserved
13-0	R/W	0	reg_viu2vdin_hsize: source horizontal size

**Table 9-269 VPU\_VIU\_ASYNC\_MASK**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:24	R/W	0	VDIN_STTs_data => VPP
23:20	R/W	0	Nr/di -> Idim
19-18	R/W	0	reg_viu2vdin_dn_ratio: down-scale ratio: 0->no scale; 1-> 1/2; 2->1/4; 3->reserved
11:8	R/W	0	Vpp afbc-> di pre inp
7:4	R/W	0	Mask vd1-> di post input
3:0	R/W	0	Mask di post => VPP_VD1_IN

**Table 9-270 VPU\_VDIN\_MISC\_CTRL**

Bit(s)	R/W	Default	Description
31:26	R/W	0	reserved
25-24	R/W	0	afbce gate clock ctrl
23	R/W	0	afbce_go_field & afbce_go_line select : 0:vdin 1:di
22	R/W	0	vdin_go_field & vdin_go_line select : 0:vdin0 1:vdin1
21	R/W	0	vdin0 output to afbc enc
20	R/W	0	vdin0 output to write mif
19	R/W	0	vdin0 output to mif enable
18	R/W	0	vdin1 output to afbc enc
17	R/W	0	vdin1 output to write mif
16	R/W	0	vdin1 output to mif enable
15	R/W	0	vdin line int enable
14-11	R/W	0	vdin arbiter gate clock ctrl
10	R/W	0	ldim stts data select : 0: vdin0 1:vdin1
9-8	R/W	0	vdin1 write mif fix disable
7-6	R/W	0	vdin0 write mif fix disable
4	R/W	0	Vdin1_wr_rst_n
3	R/W	0	Vdin0_wr_rst_n
2	R/W	0	Nrin_mux_wr_rst_n
1	R/W	0	Vdin1_rst_n
0	R/W	0	Vdin0_rst_n

Table 9-271 VPU\_VIU\_VDIN\_IF\_MUX\_CTRL

Bit(s)	R/W	Default	Description
28-24	R/W	0	//Select VIU to "VDIN0/VDIN1 source 9" data path , must clear it first before changing the path sel //5'b00000= Disable VIU to VDIN path //5'b00001= Enable VIU of ENC_I domain to VDIN //5'b00010= Enable VIU of ENC_P domain to VDIN //5'b00100= Enable VIU of ENC_T domain to VDIN //5'b01000= Enable VIU WriteBack 1 domain to VDIN //5'b10000= Enable VIU WriteBack 2 domain to VDIN
20-16	R/W	0	//Select VIU Clock to "VDIN0/VDIN1 source 9" clk path , must clear it first before changing the path sel //5'b00000= Disable VIU to VDIN clock //5'b00001= Enable clock VIU of ENC_I domain to VDIN //5'b00010= Enable clock VIU of ENC_P domain to VDIN //5'b00100= Enable clock VIU of ENC_T domain to VDIN //5'b01000= Enable clock VIU WriteBack 1 domain to VDIN //5'b10000= Enable clock VIU WriteBack 2 domain to VDIN
12-8	R/W	0	//Select VIU to "VDIN0/VDIN1 source 7" data path , must clear it first before changing the path sel //5'b00000= Disable VIU to VDIN clock //5'b00001= Enable clock VIU of ENC_I domain to VDIN //5'b00010= Enable clock VIU of ENC_P domain to VDIN //5'b00100= Enable clock VIU of ENC_T domain to VDIN //5'b01000= Enable clock VIU WriteBack 1 domain to VDIN //5'b10000= Enable clock VIU WriteBack 2 domain to VDIN
4-0	R/W	0	//Select Clock to "VDIN0/VDIN1 source 7" , must clear it first before changing the path sel //5'b00000= Disable VIU to VDIN clock //5'b00001= Enable clock VIU of ENC_I domain to VDIN //5'b00010= Enable clock VIU of ENC_P domain to VDIN //5'b00100= Enable clock VIU of ENC_T domain to VDIN //5'b01000= Enable clock VIU WriteBack 1 domain to VDIN //5'b10000= Enable clock VIU WriteBack 2 domain to VDIN

Table 9-272 VPU\_VIU2VDIN1\_HDN\_CTRL

Bit(s)	R/W	Default	Description
20	R/W	0	software reset
19-18	R/W	0	reg_viu2vdin2_dn_ratio: down-scale ratio: 0->no scale; 1-> 1/2; 2->1/4; 3->reserved
17-16	R/W	0	reg_viu2vdin2_ft_mode: filter mode; 0->no filter; 1->[0 2 2 0]/4; 2->[1 1 1 1]/4; 3->[1 3 3 1]/8
15-14	R/W	0	reserved
13-0	R/W	0	reg_viu2vdin2_hsize: source horizontal size

Table 9-273 VPU\_VENCX\_CLK\_CTRL

Bit(s)	R/W	Default	Description
2	R/W	0	Enci_afifo_clk: 0: cts_vpu_clk_tm 1: cts_vpu_clkc_tm
1	R/W	0	Encl_afifo_clk: 0: cts_vpu_clk_tm 1: cts_vpu_clkc_tm
0	R/W	0	Encp_afifo_clk: 0: cts_vpu_clk_tm 1: cts_vpu_clkc_tm

**Table 9-274 VPU\_RDARB\_MODE\_L1C1 0x2790**

Bit(s)	R/W	Default	Description
21:16	R/W	0	rdarb_sel : uns, default = 0 , rdarb_sel [0]==0 slave dc0 connect master port0 rdarb_sel[0]==1 slave dc0 connect master port1 rdarb_sel [1]==0 slave dc1 connect master port0 rdarb_sel [1]==1 slave dc1 connect master port1 rdarb_sel [2]==0 slave dc2 connect master port0 rdarb_sel[2]==1 slave dc2 connect master port1 rdarb_sel [3]==0 slave dc3 connect master port0 rdarb_sel[3]==1 slave dc3 connect master port1 rdarb_sel [4]==0 slave dc4 connect master port0 rdarb_sel[4]==1 slave dc4 connect master port1 rdarb_sel [5]==0 slave dc5 connect master port0 rdarb_sel[5]==1 slave dc5 connect master port1
9:8	R/W	0	rdarb_arb_mode : uns, default = 0 , rdarb_arb_mode [0] master port0 arb way, rdarb_arb_mode [1] master port1 arb way,
3:0	R/W	0	rdarb_gate_clk_ctrl : uns, default = 0 , rdarb_gate_clk_ctrl [1:0] master port0 clk gate control rdarb_gate_clk_ctrl [3:2] master port1 clk gate control

**Table 9-275 VPU\_RDARB\_REQEN\_SLV\_L1C1 0x2791**

Bit(s)	R/W	Default	Description
11:0	R/W	0xff	rdarb_dc_req_en : unsigned , default = 12'hfff rdarb_dc_req_en [0]: the slv0 req to mst port0 enable, rdarb_dc_req_en [1]: the slv1 req to mst port0 enable, rdarb_dc_req_en [2]: the slv2 req to mst port0 enable, rdarb_dc_req_en [3]: the slv3 req to mst port0 enable, rdarb_dc_req_en [4]: the slv4 req to mst port0 enable, rdarb_dc_req_en [5]: the slv5 req to mst port0 enable, rdarb_dc_req_en [6]: the slv0 req to mst port1 enable, rdarb_dc_req_en [7]: the slv1 req to mst port1 enable, rdarb_dc_req_en [8]: the slv2 req to mst port1 enable, rdarb_dc_req_en [9]: the slv3 req to mst port1 enable, rdarb_dc_req_en [10]: the slv4 req to mst port1 enable, rdarb_dc_req_en [11]: the slv5 req to mst port1 enable,

**Table 9-276 VPU\_RDARB\_WEIGH0\_SLV\_L1C1 0x2792**

Bit(s)	R/W	Default	Description
29:0	R/W	0	rddc_weigh_sxn : unsigned , default = 0 rddc_weigh_sxn [0*6+:6]: the slv0 req weigh number rddc_weigh_sxn [1*6+:6]: the slv1 req weigh number rddc_weigh_sxn [2*6+:6]: the slv2 req weigh number rddc_weigh_sxn [3*6+:6]: the slv3 req weigh number rddc_weigh_sxn [4*6+:6]: the slv4 req weigh number

**Table 9-277 VPU\_RDARB\_WEIGH1\_SLV\_L1C1 0x2793**

Bit(s)	R/W	Default	Description
5:0	R/W	0	rddc_weigh_sxn : unsigned , default = 0 rddc_weigh_sxn [5*6+:6]: the slv5 req weigh number

**Table 9-278 VPU\_WRARB\_MODE\_L1C1 0x2794**

Bit(s)	R/W	Default	Description
21:16	R/W	0	wrarb_sel : uns, default = 0 , wrarb_sel [0]==0 slave dc0 connect master port0 wrarb_sel[0]==1 slave dc0 connect master port1 wrarb_sel [1]==0 slave dc1 connect master port0 wrarb_sel[1]==1 slave dc1 connect master port1 wrarb_sel [2]==0 slave dc2 connect master port0 wrarb_sel[2]==1 slave dc2 connect master port1 wrarb_sel [3]==0 slave dc3 connect master port0 wrarb_sel[3]==1 slave dc3 connect master port1 wrarb_sel [4]==0 slave dc4 connect master port0 wrarb_sel[4]==1 slave dc4 connect master port1 wrarb_sel [5]==0 slave dc5 connect master port0 wrarb_sel[5]==1 slave dc5 connect master port1
9:8	R/W	0	wrarb_arb_mode : uns, default = 0 , wrarb_arb_mode [0] master port0 arb way, wrarb_arb_mode [1] master port1 arb way,
3:0	R/W	0	wrarb_gate_clk_ctrl : uns, default = 0 , wrarb_gate_clk_ctrl [1:0] master port0 clk gate control wrarb_gate_clk_ctrl [3:2] master port1 clk gate control

**Table 9-279 VPU\_WRARB\_REQEN\_SLV\_L1C1 0x2795**

Bit(s)	R/W	Default	Description
11:0	R/W	0	wrarb_dc_req_en : unsigned , default = 0 wrarb_dc_req_en [0]: the slv0 req to mst port0 enable, wrarb_dc_req_en [1]: the slv1 req to mst port0 enable, wrarb_dc_req_en [2]: the slv2 req to mst port0 enable, wrarb_dc_req_en [3]: the slv3 req to mst port0 enable, wrarb_dc_req_en [4]: the slv4 req to mst port0 enable, wrarb_dc_req_en [5]: the slv5 req to mst port0 enable, wrarb_dc_req_en [0]: the slv0 req to mst port1 enable, wrarb_dc_req_en [1]: the slv1 req to mst port1 enable, wrarb_dc_req_en [2]: the slv2 req to mst port1 enable, wrarb_dc_req_en [3]: the slv3 req to mst port1 enable, wrarb_dc_req_en [4]: the slv4 req to mst port1 enable, wrarb_dc_req_en [5]: the slv5 req to mst port1 enable,

**Table 9-280 VPU\_WRARB\_WEIGH0\_SLV\_L1C1 0x2796**

Bit(s)	R/W	Default	Description
29:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 wrdc_weigh_sxn [0*6+:6]: the slv0 req weigh number wrdc_weigh_sxn [1*6+:6]: the slv1 req weigh number wrdc_weigh_sxn [2*6+:6]: the slv2 req weigh number wrdc_weigh_sxn [3*6+:6]: the slv3 req weigh number wrdc_weigh_sxn [4*6+:6]: the slv4 req weigh number

**Table 9-281 VPU\_WRARB\_WEIGH1\_SLV\_L1C1 0x2797**

Bit(s)	R/W	Default	Description
5:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 wrdc_weigh_sxn [5*6+:6]: the slv5 req weigh number

**Table 9-282 VPU\_RDWR\_ARB\_STATUS\_L1C1 0x2798**

Bit(s)	R/W	Default	Description
3:2	R/W	0	wrarb_arb_busy : unsigned , default = 0
1:0	R/W	0	rdarb_arb_busy : unsigned , default = 0

**Table 9-283 VPU\_RDARB\_MODE\_L1C2 0x2799**

Bit(s)	R/W	Default	Description
20:16	R/W	0	rdarb_sel : uns, default = 0 , rdarb_sel [0]==0 slave dc0 connect master port0 rdarb_sel[0]==1 slave dc0 connect master port1 rdarb_sel [1]==0 slave dc1 connect master port0 rdarb_sel[1]==1 slave dc1 connect master port1 rdarb_sel [2]==0 slave dc2 connect master port0 rdarb_sel[2]==1 slave dc2 connect master port1 rdarb_sel [3]==0 slave dc3 connect master port0 rdarb_sel[3]==1 slave dc3 connect master port1 rdarb_sel [4]==0 slave dc4 connect master port0 rdarb_sel[4]==1 slave dc4 connect master port1
9:8	R/W	0	rdarb_arb_mode : uns, default = 0 , rdarb_arb_mode [0] master port0 arb way, rdarb_arb_mode [1] master port1 arb way,
3:0	R/W	0	rdarb_gate_clk_ctrl : uns, default = 0 , rdarb_gate_clk_ctrl [1:0] master port0 clk gate control rdarb_gate_clk_ctrl [3:2] master port0 clk gate control

**Table 9-284 VPU\_RDARB\_REQEN\_SLV\_L1C2 0x279a**

Bit(s)	R/W	Default	Description
9:0	R/W	0	rdarb_dc_req_en : unsigned , default = 0 rdarb_dc_req_en [0]: the slv0 req to mst port0 enable, rdarb_dc_req_en [1]: the slv1 req to mst port0 enable, rdarb_dc_req_en [2]: the slv2 req to mst port0 enable, rdarb_dc_req_en [3]: the slv3 req to mst port0 enable, rdarb_dc_req_en [4]: the slv4 req to mst port0 enable, rdarb_dc_req_en [5]: the slv0 req to mst port1 enable, rdarb_dc_req_en [6]: the slv1 req to mst port1 enable, rdarb_dc_req_en [7]: the slv2 req to mst port1 enable, rdarb_dc_req_en [8]: the slv3 req to mst port1 enable, rdarb_dc_req_en [9]: the slv4 req to mst port1 enable,

**Table 9-285 VPU\_RDARB\_WEIGHT0\_SLV\_L1C2 0x279b**

Bit(s)	R/W	Default	Description
29:0	R/W	0	rddc_weigh_sxn : unsigned , default = 0 rddc_weigh_sxn [0*6+:6]: the slv0 req weigh number rddc_weigh_sxn [1*6+:6]: the slv1 req weigh number rddc_weigh_sxn [2*6+:6]: the slv2 req weigh number rddc_weigh_sxn [3*6+:6]: the slv3 req weigh number rddc_weigh_sxn [4*6+:6]: the slv4 req weigh number

**Table 9-286 VPU\_RDWR\_ARB\_STATUS\_L1C2 0x279c**

Bit(s)	R/W	Default	Description
1:0	R/W	0	rdarb_arb_busy : unsigned , default = 0



**Table 9-287 VPU\_RDARB\_MODE\_L2C1 0x279d**

Bit(s)	R/W	Default	Description
27:16	R/W	0	rdarb_sel : uns, default = 0 , rdarb_sel [0]==0 slave dc0 connect master port0 rdarb_sel[0]==1 slave dc0 connect master port1 rdarb_sel [1]==0 slave dc1 connect master port0 rdarb_sel[1]==1 slave dc1 connect master port1 rdarb_sel [2]==0 slave dc2 connect master port0 rdarb_sel[2]==1 slave dc2 connect master port1 rdarb_sel [3]==0 slave dc3 connect master port0 rdarb_sel[3]==1 slave dc3 connect master port1 rdarb_sel [4]==0 slave dc4 connect master port0 rdarb_sel[4]==1 slave dc4 connect master port1 rdarb_sel [5]==0 slave dc5 connect master port0 rdarb_sel[5]==1 slave dc5 connect master port1
10:8	R/W	0	rdarb_arb_mode : uns, default = 0 , rdarb_arb_mode [0] master port0 arb way, rdarb_arb_mode [1] master port1 arb way,
5:0	R/W	0	rdarb_gate_clk_ctrl : uns, default = 0 , rdarb_gate_clk_ctrl [1:0] master port0 clk gate control rdarb_gate_clk_ctrl [3:2] master port1 clk gate control rdarb_gate_clk_ctrl [5:4] master port2 clk gate control

**Table 9-288 VPU\_RDARB\_REQEN\_SLV\_L2C1 0x279e**

Bit(s)	R/W	Default	Description
17:0	R/W	0	rdarb_dc_req_en : unsigned , default = 0 rdarb_dc_req_en [0]: the slv0 req to mst port0 enable, rdarb_dc_req_en [1]: the slv1 req to mst port0 enable, rdarb_dc_req_en [2]: the slv2 req to mst port0 enable, rdarb_dc_req_en [3]: the slv3 req to mst port0 enable, rdarb_dc_req_en [4]: the slv4 req to mst port0 enable, rdarb_dc_req_en [5]: the slv5 req to mst port0 enable, rdarb_dc_req_en [0]: the slv0 req to mst port1 enable, rdarb_dc_req_en [1]: the slv1 req to mst port1 enable, rdarb_dc_req_en [2]: the slv2 req to mst port1 enable, rdarb_dc_req_en [3]: the slv3 req to mst port1 enable, rdarb_dc_req_en [4]: the slv4 req to mst port1 enable, rdarb_dc_req_en [5]: the slv5 req to mst port1 enable,

**Table 9-289 VPU\_RDARB\_WEIGH0\_SLV\_L2C1 0x279f**

Bit(s)	R/W	Default	Description
29:0	R/W	0	rddc_weigh_sxn : unsigned , default = 0 rddc_weigh_sxn [0*6+:6]: the slv0 req weigh number rddc_weigh_sxn [1*6+:6]: the slv1 req weigh number rddc_weigh_sxn [2*6+:6]: the slv2 req weigh number rddc_weigh_sxn [3*6+:6]: the slv3 req weigh number rddc_weigh_sxn [4*6+:6]: the slv4 req weigh number

**Table 9-290 VPU\_RDARB\_WEIGH1\_SLV\_L2C1 0x27a0**

Bit(s)	R/W	Default	Description
5:0	R/W	0	rddc_weigh_sxn : unsigned , default = 0 rddc_weigh_sxn [5*6+:6]: the slv5 req weigh number

**Table 9-291 VPU\_RDWR\_ARB\_STATUS\_L2C1 0x27a1**

Bit(s)	R/W	Default	Description
3:2	R/W	0	wrarb_arb_busy : unsigned , default = 0
1:0	R/W	0	rdarb_arb_busy : unsigned , default = 0

Table 9-292 VPU\_WRARB\_MODE\_L2C1 0x27a2

Bit(s)	R/W	Default	Description
19:16	R/W	0	wrarb_sel : uns, default = 0 , wrarb_sel [0]==0 slave dc0 connect master port0 wrarb_sel[0]==1 slave dc0 connect master port1 wrarb_sel [1]==0 slave dc1 connect master port0 wrarb_sel[1]==1 slave dc1 connect master port1 wrarb_sel [2]==0 slave dc2 connect master port0 wrarb_sel[2]==1 slave dc2 connect master port1 wrarb_sel [3]==0 slave dc3 connect master port0 wrarb_sel[3]==1 slave dc3 connect master port1
9:8	R/W	0	wrarb_arb_mode : uns, default = 0 , wrarb_arb_mode [0] master port0 arb way, wrarb_arb_mode [1] master port1 arb way,
3:0	R/W	0	wrarb_gate_clk_ctrl : uns, default = 0 , wrarb_gate_clk_ctrl [1:0] master port0 clk gate control wrarb_gate_clk_ctrl [3:2] master port0 clk gate control

Table 9-293 VPU\_WRARB\_REQEN\_SLV\_L2C1 0x27a3

Bit(s)	R/W	Default	Description
7:0	R/W	0	wrarb_dc_req_en : unsigned , default = 0 wrarb_dc_req_en [0]: the slv0 req to mst port0 enable, wrarb_dc_req_en [1]: the slv1 req to mst port0 enable, wrarb_dc_req_en [2]: the slv2 req to mst port0 enable, wrarb_dc_req_en [3]: the slv3 req to mst port0 enable, wrarb_dc_req_en [0]: the slv0 req to mst port1 enable, wrarb_dc_req_en [1]: the slv1 req to mst port1 enable, wrarb_dc_req_en [2]: the slv2 req to mst port1 enable, wrarb_dc_req_en [3]: the slv3 req to mst port1 enable,

Table 9-294 VPU\_WRARB\_WEIGH0\_SLV\_L2C1 0x27a4

Bit(s)	R/W	Default	Description
23:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 wrdc_weigh_sxn [0*6+:6]: the slv0 req weigh number wrdc_weigh_sxn [1*6+:6]: the slv1 req weigh number wrdc_weigh_sxn [2*6+:6]: the slv2 req weigh number wrdc_weigh_sxn [3*6+:6]: the slv3 req weigh number

Table 9-295 VPU\_ASYNC\_RD\_MODE0 0x27a5

Bit(s)	R/W	Default	Description
18	R/W	0	req_en : unsigned , default = 0 async enable
17:16	R/W	0	clk_gate_ctrl : unsigned , default = 0 async clock gate control
15:12	R/W	4	auto_arugt_weight : unsigned , default = 4
10:9	R/W	0	arugt_sel : unsigned , default = 0 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 00 : use the input argument
8	R/W	0	argument_cfg : unsigned , default = 0 register argument control bit
7:4	R/W	4	rd_hold_num : unsigned , default = 4 hold the read command threshold
3:0	R/W	0	rd_rel_num : unsigned , default = 0 release the read command threshold

**Table 9-296 VPU\_ASYNC\_RD\_MODE1 0x27a6**

Bit(s)	R/W	Default	Description
18	R/W	0	req_en : unsigned , default = 0 async enable
17:16	R/W	0	clk_gate_ctrl : unsigned , default = 0 async clock gate control
15:12	R/W	4	auto_arugt_weight : unsigned , default = 4
10:9	R/W	0	arugt_sel : unsigned , default = 0 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 00 : use the input arguent
8	R/W	0	arguent_cfg : unsigned , default = 0 register arguent control bit
7:4	R/W	4	rd_hold_num : unsigned , default = 4 hold the read command threshold
3:0	R/W	0	rd_rel_num : unsigned , default = 0 release the read command threshold

**Table 9-297 VPU\_ASYNC\_RD\_MODE2 0x27a7**

Bit(s)	R/W	Default	Description
18	R/W	0	req_en : unsigned , default = 0 async enable
17:16	R/W	0	clk_gate_ctrl : unsigned , default = 0 async clock gate control
15:12	R/W	4	auto_arugt_weight : unsigned , default = 4
10:9	R/W	0	arugt_sel : unsigned , default = 0 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 00 : use the input arguent
8	R/W	0	arguent_cfg : unsigned , default = 0 register arguent control bit
7:4	R/W	4	rd_hold_num : unsigned , default = 4 hold the read command threshold
3:0	R/W	0	rd_rel_num : unsigned , default = 0 release the read command threshold

**Table 9-298 VPU\_ASYNC\_RD\_MODE3 0x27a8**

Bit(s)	R/W	Default	Description
18	R/W	0	req_en : unsigned , default = 0 async enable
17:16	R/W	0	clk_gate_ctrl : unsigned , default = 0 async clock gate control
15:12	R/W	4	auto_arugt_weight : unsigned , default = 4
10:9	R/W	0	arugt_sel : unsigned , default = 0 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 00 : use the input arguent
8	R/W	0	arguent_cfg : unsigned , default = 0 register arguent control bit
7:4	R/W	4	rd_hold_num : unsigned , default = 4 hold the read command threshold
3:0	R/W	0	rd_rel_num : unsigned , default = 0 release the read command threshold

**Table 9-299 VPU\_ASYNC\_RD\_MODE4 0x27a9**

Bit(s)	R/W	Default	Description
18	R/W	0	req_en : unsigned , default = 0 async enable
17:16	R/W	0	clk_gate_ctrl : unsigned , default = 0 async clock gate control

Bit(s)	R/W	Default	Description
15:12	R/W	4	auto_arugt_weight : unsigned , default = 4
10:9	R/W	0	arugt_sel : unsigned , default = 0 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 00 : use the input arguent
8	R/W	0	arguent_cfg : unsigned , default = 0 register arguent control bit
7:4	R/W	4	rd_hold_num : unsigned , default = 4 hold the read command threshold
3:0	R/W	0	rd_rel_num : unsigned , default = 0 release the read command threshold

Table 9-300 VPU\_ASYNC\_WR\_MODE0 0x27aa

Bit(s)	R/W	Default	Description
18	R/W	0	req_en : unsigned , default = 0 async enable
17:16	R/W	0	clk_gate_ctrl : unsigned , default = 0 async clock gate control
15:12	R/W	4	auto_arugt_weight : unsigned , default = 4
10:9	R/W	0	arugt_sel : unsigned , default = 0 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 00 : use the input arguent
8	R/W	0	arguent_cfg : unsigned , default = 0 register arguent control bit
7:4	R/W	4	wr_hold_num : unsigned , default = 4 hold the read command threshold
3:0	R/W	0	wr_rel_num : unsigned , default = 0 release the write command threshold

Table 9-301 VPU\_ASYNC\_WR\_MODE1 0x27ab

Bit(s)	R/W	Default	Description
18	R/W	0	req_en : unsigned , default = 0 async enable
17:16	R/W	0	clk_gate_ctrl : unsigned , default = 0 async clock gate control
15:12	R/W	4	auto_arugt_weight : unsigned , default = 4
10:9	R/W	0	arugt_sel : unsigned , default = 0 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 00 : use the input arguent
8	R/W	0	arguent_cfg : unsigned , default = 0 register arguent control bit
7:4	R/W	4	wr_hold_num : unsigned , default = 4 hold the read command threshold
3:0	R/W	0	wr_rel_num : unsigned , default = 0 release the write command threshold

Table 9-302 VPU\_ASYNC\_WR\_MODE2 0x27ac

Bit(s)	R/W	Default	Description
18	R/W	0	req_en : unsigned , default = 0 async enable
17:16	R/W	0	clk_gate_ctrl : unsigned , default = 0 async clock gate control
15:12	R/W	4	auto_arugt_weight : unsigned , default = 4
10:9	R/W	0	arugt_sel : unsigned , default = 0 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 00 : use the input arguent
8	R/W	0	arguent_cfg : unsigned , default = 0 register arguent control bit

Bit(s)	R/W	Default	Description
7:4	R/W	4	wr_hold_num : unsigned , default = 4 hold the read command threshold
3:0	R/W	0	wr_rel_num : unsigned , default = 0 release the write command threshold

**Table 9-303 VPU\_ASYNC\_STAT 0x27ad**

Bit(s)	R/W	Default	Description
18	R/W	0x0	axiwr2_chan_idle : unsigned , RO, axi write channel2 idle state
17	R/W	0x0	axiwr1_chan_idle : unsigned , RO, axi write channel1 idle state
16	R/W	0x0	axiwr0_chan_idle : unsigned , RO, axi write channel0 idle state
4	R/W	0x0	axird4_chan_idle : unsigned , RO, axi read channel4 idle state
3	R/W	0x0	axird3_chan_idle : unsigned , RO, axi read channel3 idle state
2	R/W	0x0	axird2_chan_idle : unsigned , RO, axi read channel2 idle state
1	R/W	0x0	axird1_chan_idle : unsigned , RO, axi read channel1 idle state
0	R/W	0x0	axird0_chan_idle : unsigned , RO, axi read channel0 idle state

**Table 9-304 VPU\_HDMI\_DITH\_01\_04 0x27f0**

Bit(s)	R/W	Default	Description
31-0	R/W	0x8214_1428	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-305 VPU\_HDMI\_DITH\_01\_15 0x27f1**

Bit(s)	R/W	Default	Description
31-0	R/W	0x4128_2841	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-306 VPU\_HDMI\_DITH\_01\_26 0x27f2**

Bit(s)	R/W	Default	Description
31-0	R/W	0x2841_4182	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-307 VPU\_HDMI\_DITH\_01\_37 0x27f3**

Bit(s)	R/W	Default	Description
31-0	R/W	0x1482_8214	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-308 VPU\_HDMI\_DITH\_10\_04 0x27f4**

Bit(s)	R/W	Default	Description
31-0	R/W	0x9669_9696	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-309 VPU\_HDMI\_DITH\_10\_15 0x27f5**

Bit(s)	R/W	Default	Description
31-0	R/W	0x3c3c_6969	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-310 VPU\_HDMI\_DITH\_10\_26 0x27f6**

Bit(s)	R/W	Default	Description
31-0	R/W	0x6996_9696	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-311 VPU\_HDMI\_DITH\_10\_37 0x27f7**

Bit(s)	R/W	Default	Description
31-0	R/W	0xc3c3_6969	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-312 VPU\_HDMI\_DITH\_11\_04 0x27f8**

Bit(s)	R/W	Default	Description
31-0	R/W	0x7deb_ebd7	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-313 VPU\_HDMI\_DITH\_11\_15 0x27f9**

Bit(s)	R/W	Default	Description
31-0	R/W	0xbed7_d7be	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-314 VPU\_HDMI\_DITH\_11\_26 0x27fa**

Bit(s)	R/W	Default	Description
31-0	R/W	0xd7be_be7d	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-315 VPU\_HDMI\_DITH\_11\_37 0x27fb**

Bit(s)	R/W	Default	Description
31-0	R/W	0xeb7d_7deb	dith lut VPU_HDMI_DITH_CNTL[0] : 1 : lut for 10b to 8b0 : lut for 12b to 10b

**Table 9-316 VPU\_HDMI\_DITH\_CNTL**

10b to 8b dither control register. 12b to 10b see VPU\_HDMI\_FMT\_CTRL

Bit(s)	R/W	Default	Description
21-19	R/W	0	frame count offset for B
18-16	R/W	0	frame count offset for G
15	R/W	0	hcnt hold when de valid
14	R/W	0	RGB frame count separate
13	R/W	0	dith4x4 : frame random enable
12	R/W	0	dith4x4 enable
11	R/W	0	tunnel enable for DOLBY
10	R/W	0	rounding enable
9-6	R/W	0	Cntl_hdmi_dith10 :
5	R/W	0	Cntl_hdmi_dith_md:
4	R/W	0	Cntl_hdmi_dith_en: dither 10-b to 8-b enable
3	R/W	0	hsync invert
2	R/W	0	vsync invert
0	R/W	0	dither lut sel :1 : sel 10b to 8b0: sel 12b to 10b

**Table 9-317 VPU\_VENCL\_DITH\_CTRL**

Bit(s)	R/W	Default	Description
19-17	R/W	0	dither 2x2 : frame number sel
16	R/W	0	dither 2x2 : frame number random
15-14	R/W	0	Reserved
13-11	R/W	7	G frame number offset
10-8	R/W	3	B frame number offset
7	R/W	0	Reserved
6	R/W	1	dither 4x4 : G/B frame number = B frame number + offset g/b
5	R/W	0	dither 4x4 : frame number random
4	R/W	1	dither 4x4 : enable
3	R/W	0	Reserved
2	R/W	0	dither md
1	R/W	0	rounding enable
0	R/W	1	dither enable

**Table 9-318 VPU\_VENCL\_DITH\_LUT\_1**

Bit(s)	R/W	Default	Description
31-0	R/W	0x8214_ 1428	dith lut

**Table 9-319 VPU\_VENCL\_DITH\_LUT\_2**

Bit(s)	R/W	Default	Description
31-0	R/W	0x4128_ 2841	dith lut

**Table 9-320 VPU\_VENCL\_DITH\_LUT\_3**

Bit(s)	R/W	Default	Description
31-0	R/W	0x2841_ 4182	dith lut

**Table 9-321 VPU\_VENCL\_DITH\_LUT\_4**

Bit(s)	R/W	Default	Description
31-0	R/W	0x1482_ 8214	dith lut

**Table 9-322 VPU\_VENCL\_DITH\_LUT\_5**

Bit(s)	R/W	Default	Description
31-0	R/W	0x9669_ 9696	dith lut

**Table 9-323 VPU\_VENCL\_DITH\_LUT\_6**

Bit(s)	R/W	Default	Description
31-0	R/W	0x3c3c_ 6969	dith lut

**Table 9-324 VPU\_VENCL\_DITH\_LUT\_7**

Bit(s)	R/W	Default	Description
31-0	R/W	0x6996_ 9696	dith lut

**Table 9-325 VPU\_VENCL\_DITH\_LUT\_8**

Bit(s)	R/W	Default	Description
31-0	R/W	0xc3c3_ 6969	dith lut



**Table 9-326 VPU\_VENCL\_DITH\_LUT\_9**

Bit(s)	R/W	Default	Description
31-0	R/W	0x7deb_ ebd7	dith lut

**Table 9-327 VPU\_VENCL\_DITH\_LUT\_10**

Bit(s)	R/W	Default	Description
31-0	R/W	0xbed7_ d7be	dith lut

**Table 9-328 VPU\_VENCL\_DITH\_LUT\_11**

Bit(s)	R/W	Default	Description
31-0	R/W	0xd7be_ be7d	dith lut

**Table 9-329 VPU\_VENCL\_DITH\_LUT\_12**

Bit(s)	R/W	Default	Description
31-0	R/W	0xeb7d_ 7deb	dith lut

### 9.2.3.2 VPU Video Lock Registers

#### Register Address

The following lists describe the mapping between each register and its address.

- VPU\_VLOCK\_CTRL 0xff00c000
- VPU\_VLOCK\_MISC\_CTRL 0xff00c004
- VPU\_VLOCK\_LOOP0\_ACCUM\_LMT 0xff00c008
- VPU\_VLOCK\_LOOP0\_CTRL0 0xff00c00c
- VPU\_VLOCK\_LOOP1\_CTRL0 0xff00c010
- VPU\_VLOCK\_LOOP1\_IMISSYNC\_MAX 0xff00c014
- VPU\_VLOCK\_LOOP1\_IMISSYNC\_MIN 0xff00c018
- VPU\_VLOCK\_OVWRITE\_ACCUM0 0xff00c01c
- VPU\_VLOCK\_OVWRITE\_ACCUM1 0xff00c020
- VPU\_VLOCK\_OUTPUT0\_CAPT\_LMT 0xff00c024
- VPU\_VLOCK\_OUTPUT0\_PLL\_LMT 0xff00c028
- VPU\_VLOCK\_OUTPUT1\_CAPT\_LMT 0xff00c02c
- VPU\_VLOCK\_OUTPUT1\_PLL\_LMT 0xff00c030
- VPU\_VLOCK\_LOOP1\_PHSDIF\_TGT 0xff00c034
- VPU\_VLOCK\_RO\_LOOP0\_ACCUM 0xff00c038
- VPU\_VLOCK\_RO\_LOOP1\_ACCUM 0xff00c03c
- VPU\_VLOCK\_OROW\_OCOL\_MAX 0xff00c040

- VPU\_VLOCK\_RO\_VS\_I\_DIST 0xff00c044
- VPU\_VLOCK\_RO\_VS\_O\_DIST 0xff00c048
- VPU\_VLOCK\_RO\_LINE\_PIX\_ADJ 0xff00c04c
- VPU\_VLOCK\_RO\_OUTPUT\_00\_01 0xff00c050
- VPU\_VLOCK\_RO\_OUTPUT\_10\_11 0xff00c054
- VPU\_VLOCK\_MX4096 0xff00c058
- VPU\_VLOCK\_STBDET\_WIN0\_WIN1 0xff00c05c
- VPU\_VLOCK\_STBDET\_CLP 0xff00c060
- VPU\_VLOCK\_STBDET\_ABS\_WIN0 0xff00c064
- VPU\_VLOCK\_STBDET\_ABS\_WIN1 0xff00c068
- VPU\_VLOCK\_STBDET\_SGN\_WIN0 0xff00c06c
- VPU\_VLOCK\_STBDET\_SGN\_WIN1 0xff00c070
- VPU\_VLOCK\_ADJ\_EN\_SYNC\_CTRL 0xff00c074
- VPU\_VLOCK\_GCLK\_EN 0xff00c078
- VPU\_VLOCK\_LOOP1\_ACCUM\_LMT 0xff00c07c
- VPU\_VLOCK\_RO\_M\_INT\_FRAC 0xff00c080
- VPU\_VLOCK\_RO\_PH\_DIS 0xff00c084
- VPU\_VLOCK\_RO\_PH\_ERR 0xff00c088
- VPU\_VLOCK\_LOCK\_TH 0xff00c08c
- VPU\_VLOCK\_RO\_LCK\_FRM 0xff00c090
- VPU\_VLOCK\_WIN0\_TH 0xff00c094
- VPU\_VLOCK\_WIN0\_RATIO 0xff00c098
- VPU\_VLOCK\_WIN0\_FILTER\_CNTL 0xff00c09c
- VPU\_VLOCK\_WIN1\_TH 0xff00c0a0
- VPU\_VLOCK\_WIN1\_RATIO 0xff00c0a4
- VPU\_VLOCK\_WIN1\_FILTER\_CNTL 0xff00c0a8
- VPU\_VLOCK\_LOCK\_TH1 0xff00c0ac
- VPU\_VLOCK\_LOOP0\_ERR\_LMT 0xff00c0b0
- VPU\_VLOCK\_LOOP1\_ERR\_LMT 0xff00c0b4
- VPU\_VLOCK\_ERR\_CTRL0 0xff00c0b8
- VPU\_VLOCK\_RO\_VS\_O\_ABS\_CNT\_LSB 0xff00c0bc
- VPU\_VLOCK\_RO\_VS\_I\_ABS\_CNT\_LSB 0xff00c0c0
- VPU\_VLOCK\_RO\_VS\_ABS\_CNT\_MSB 0xff00c0c4

## Register Description

**Table 9-330 VPU\_VLOCK\_CTRL**

Bit(s)	R/W	Default	Description
31	R/W	0x0	Vid_lock_en: 1: enable video lock module
30	R/W	0x0	Reg_adj_enc: enable video lock to adjust encoder
29	R/W	0x0	Adj_pll: enable video lock to adjust PLL
28	R/W	0x0	Mpeg_vs: set this to 1, then 0, this is software controlled mpeg vsync
27-26	R/W	0x0	Output goes to which module: 0: encl, 1: encp, 2: enci

Bit(s)	R/W	Default	Description
25-20	R/W	0x0	Output vsync width extend: make sure the vsync width is extended big enough for vpu_vid_lock_clk to sample
19	R/W	0x0	m frac right shift 1 : right shift 2 bit 0 : no shift
18-16	R/W	0x0	Input Vsync source select: 0: unuse, 1: fromhdmi rx , 2:from tv-decoder, 3: from dvin, 4: from dvin, 5: from 2nd bt656
15			Output vsync invert: 1, invert
14			Input vsync invert: 1, invert
13-8	R/W		Input vsync width extend: make sure the vsync width is extended big enough for vpu_vid_lock_clk to sample
7			Force loop1 err enable: 1. Force error of loop1
6			Force loop0 err enable: 1. Force error of loop0
5			Overwrite accum0 enable
4			Loop0 adjust capture enable
3			Loop0 adjust pll enable
2			Overwrite accum1 enable
1			Loop1 adjust capture enable
0			Loop0 adjust pll enable

Table 9-331 VPU\_VLOCK\_MISC\_CTRL

Bit(s)	R/W	Default	Description
28	R/W	0x0	for interlace : get average of every two input vsync
26-24	R/W	0x0	Adj_capt_pxgroupers, make sure the pixel number in one line of encoder is multiples of 2^pxgroupers
23-16	R/W	0x0	lfrm_cnt_mod: (output vsync freq)/(input vsync_freq * ifrm_cnt_mod) must be integer
15-8	R/W	0x0	Output vsync frequency
7-0	R/W	0x0	Input vsync frequency

Table 9-332 VPU\_VLOCK\_LOOP0\_ACCUM\_LMT

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	LOOP0 accumulator limit

Table 9-333 VPU\_VLOCK\_LOOP0\_CTRL0

Bit(s)	R/W	Default	Description
31-24	R/W	0x0	Loop0 errclip rate
23-20	R/W	0x0	Loop0_adj_pll_rs, right shift of loop0 adjust pll portion
19-12	R/W	0x0	Loop0_adj_pll_gain, u1.7

Bit(s)	R/W	Default	Description
11-8	R/W	0x0	Loop0_adj_capt_rs, right shift of loop0 adjust capture portion
7-0	R/W	0x0	Loop0_adj_capt_gain

**Table 9-334 VPU\_VLOCK\_LOOP0\_CTRL1**

Bit(s)	R/W	Default	Description
23-20	R/W	0x0	Loop1_adj_pll_rs
19-12	R/W	0x0	Loop1_adj_pll_gain
11-8	R/W	0x0	Loop1_adj_capt_rs
7-0	R/W	0x0	Loop1_adj_capt_gain

**Table 9-335 VPU\_VLOCK\_LOOP1\_IMISSYNC\_MAX**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0	Loop1 imissync max, input signal is missed after input vsync counter is larger than this max threshold

**Table 9-336 VPU\_VLOCK\_LOOP1\_IMISSYNC\_MIN**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0	Loop1 imissync min, input signal is missed after input vsync counter is less than this max threshold

**Table 9-337 VPU\_VLOCK\_OVERWRITE\_ACCUM0**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0	Overwrite value of accum0

**Table 9-338 VPU\_VLOCK\_OVERWRITE\_ACCUM1**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0	Overwrite value of accum1

**Table 9-339 VPU\_VLOCK\_OUTPUT0\_CAPT\_LMT**

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	Output0 capture limit

**Table 9-340 VPU\_VLOCK\_OUTPUT0\_PLL\_LMT**

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	Output0 pll limit

**Table 9-341 VPU\_VLOCK\_OUTPUT1\_CAPT\_LMT**

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	Output1 capture limit

**Table 9-342 VPU\_VLOCK\_OUTPUT1\_PLL\_LMT**

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	Output1 pll limit

**Table 9-343 VPU\_VLOCK\_LOOP1\_PHSDFI\_TARGET**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0	Loop1 phase difference target, (input vsync - output vsync) phase distance target

**Table 9-344 VPU\_VLOCK\_RO\_LOOP0\_ACCUM**

Bit(s)	R/W	Default	Description
27-0	R	0x0	Read only, loop0 accum result

**Table 9-345 VPU\_VLOCK\_RO\_LOOP1\_ACCUM**

Bit(s)	R/W	Default	Description
27-0	R	0x0	Read only, loop1 accum result

**Table 9-346 VPU\_VLOCK\_OROW\_OCOL\_MAX**

Bit(s)	R/W	Default	Description
29-16	R/W	0x0	Ocol_max
13-0	R/W	0x0	Orow_max

**Table 9-347 VPU\_VLOCK\_RO\_VS\_I\_D**

Bit(s)	R/W	Default	Description
27-0	R	0x0	Read only, input vsync counter

**Table 9-348 VPU\_VLOCK\_RO\_VS\_O\_D**

Bit(s)	R/W	Default	Description
27-0	R	0x0	Read only, output vsync counter

**Table 9-349 VPU\_VLOCK\_RO\_LINE\_PIX\_ADJ**

Bit(s)	R/W	Default	Description
29-16	R	0x0	Read only, encoder line adjust number
13-0	R	0x0	Read only, encoder pix adjust number

**Table 9-350 VPU\_VLOCK\_RO\_OUTPUT\_00\_01**

Bit(s)	R/W	Default	Description
31-16	R	0x0	Read only, accum0 output 00
15-0	R	0x0	Read only, accum0 output 01

**Table 9-351 VPU\_VLOCK\_RO\_OUTPUT\_10\_11**

Bit(s)	R/W	Default	Description
31-16	R	0x0	Read only, accum1 output 10
15-0	R	0x0	Read only, accum1 output 11

**Table 9-352 VPU\_VLOCK\_MX4096**

Bit(s)	R/W	Default	Description
20-0	R/W	0x0	Mx4096

**Table 9-353 VPU\_VLOCK\_STBDET\_WIN0\_WIN1**

Bit(s)	R/W	Default	Description
15-8	R/W	0x0	Verr_stbdet_win1
7-0	R/W	0x0	Verr_stbdet_win0

**Table 9-354 VPU\_VLOCK\_STBDET\_CLP**

Bit(s)	R/W	Default	Description
15-8	R	0x0	Read only, ro_verr_clp_win1, verr_clp number in win0
7-0	R	0x0	Read only, ro_verr_clp_win0, verr_clp number in win1

**Table 9-355 VPU\_VLOCK\_STBDET\_ABS\_WIN0**

Bit(s)	R/W	Default	Description
23-0	R	0x0	Read only, ro_verr_abs_win0

**Table 9-356 VPU\_VLOCK\_STBDET\_ABS\_WIN1**

Bit(s)	R/W	Default	Description
23-0	R	0x0	Read only, ro_verr_abs_win1

**Table 9-357 VPU\_VLOCK\_STBDET\_SGN\_WIN0**

Bit(s)	R/W	Default	Description
23-0	R	0x0	Read only, ro_verr_sgn_win0

**Table 9-358 VPU\_VLOCK\_STBDET\_SGN\_WIN1**

Bit(s)	R/W	Default	Description
23-0	R	0x0	Read only, ro_verr_sgn_win1

**Table 9-359 VPU\_VLOCK\_ADJ\_EN\_SYNC\_CTRL**

Bit(s)	R/W	Default	Description
31-24	R/W	0x0	PLL adjust enable signal sync ctrl, adj_en_for_pll_end, end counter of adj_en_pll signal fall to 0
23-16	R/W	0x0	PLL adjust enable signal sync ctrl, adj_en_for_pll_start, start counter of adj_en_pll signal go to 1, start must be larger than end
15-8	R/W	0x0	Adj_en_sync_latch_cnt, this is a delay to latch the adj_en signal
7-0	R/W	0x0	Adj_en_ext_cnt, extend the adj_en signal from vid_lock clock domain to pll sample domain, make sure it's wide enough

**Table 9-360 VPU\_VLOCK\_GCLK\_EN**

Bit(s)	R/W	Default	Description
2	R/W	0x0	Ref clock enable
1	R/W	0x0	Vsout clk enable
0	R/W	0x0	Vsin clk enable

**Table 9-361 VPU\_VLOCK\_LOOP1\_ACCUM\_LMT**

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	LOOP1 accumulator limit

**Table 9-362 VPU\_VLOCK\_RO\_M\_INT\_FRAC**

Bit(s)	R/W	Default	Description
29-16	R	0x0	Read only, m_int to PLL
13-0	R	0x0	Read only, m_frac to PLL

**Table 9-363 VPU\_VLOCK\_RO\_PH\_DIS**

Bit(s)	R/W	Default	Description
27-0	R	0x0	phase distance RO

**Table 9-364 VPU\_VLOCK\_RO\_PH\_ERR**

Bit(s)	R/W	Default	Description
27-0	R	0x0	phase error RO

**Table 9-365 VPU\_VLOCK\_LOCK\_TH**

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	phase error threshold

**Table 9-366 VPU\_VLOCK\_LOCK\_TH1**

Bit(s)	R/W	Default	Description
21-0	R/W	0x0	frequency error threshold

**Table 9-367 VPU\_VLOCK\_RO\_LCK\_FRM**

Bit(s)	R/W	Default	Description
31-18	R/W	0x0	reserved
17	R/W	0x0	phase lock flag
16	R/W	0x0	frequency lock flag
15-0	R/W	0x0	ro_lock_frame_count

**Table 9-368 VPU\_VLOCK\_WIN0\_TH**

Bit(s)	R/W	Default	Description
23-0	R/W	0x0	reg_win0_th

**Table 9-369 VPU\_VLOCK\_WIN0\_RATIO**

Bit(s)	R/W	Default	Description
7-0	R/W	0x0	reg_win0_ratio

**Table 9-370 VPU\_VLOCK\_WIN0\_FILTER\_CNTL**

Bit(s)	R/W	Default	Description
20-0	R/W	0x0	reg_win0_lock_filter_cntl



**Table 9-371 VPU\_VLOCK\_WIN1\_TH**

Bit(s)	R/W	Default	Description
23-0	R/W	0x0	reg_win0_th

**Table 9-372 VPU\_VLOCK\_WIN1\_RATIO**

Bit(s)	R/W	Default	Description
7-0	R/W	0x0	reg_win0_ratio

**Table 9-373 VPU\_VLOCK\_WIN1\_FILTER\_CNTL**

Bit(s)	R/W	Default	Description
20-0	R/W	0x0	reg_win0_lock_filter_cntl

### 9.2.3.3 VPU\_AFBCE Registers

#### DI AFBCE Registers

**Table 9-374 DI\_AFBCE\_ENABLE 0x2060**

Bit(s)	R/W	Default	Description
31:20	R/W	0	gclk_ctrl : unsigned , default = 0, gclk_ctrl of afbe submodule : 12'h000 : gatting afbce clock. 12'hfff : free_run afbce clock. 12'h555 : close afbce clock.
19:16	R/W	0	1) afbce_sync_sel : unsigned , default = 0, shadow some size/scope/mode registers in afbce 4'hf : sync active. 4'h0: sync close.
13	R/W	0	enc_rst_mode : unsigned , default = 0, frame rst mode bits: 1:use software reset as frame reset by writing a pulse to register DI_AFBCE_MODE[29]. 0:use vsync as frame reset.
12	R/W	0	enc_en_mode : unsigned , default = 0, DI_AFBCE start mode bits: 1: DI_AFBCE start_up by writing a pulse to register DI_AFBCE_ENABLE[0]. 0: DI_AFBCE start_up several line after vsync,line number setting by DI_AFBCE_MODE[22:16].
8	R/W	0	enc_enable : unsigned , default = 0, 1: open DI_AFBCE module. 0: close DI_AFBCE module.
0	R/W	0	enc_frm_start : DI_AFBCE start_up bits This bits can start afbce by writing high signal when DI_AFBCE_ENABLE[12] equal 1.

**Table 9-375 DI\_AFBCE\_MODE 0x2061**

Bit(s)	R/W	Default	Description
31:29	R/W	0	soft_rst : unsigned, default = 0 , bit[0] async rst ,soft_ware ctrl async rst bit[1] sync rst ,soft_ware ctrl sync rst bit[2] sync rst by vsync
27:26	R/W	0	rev_mode : unsigned, default = 0 , reverse mode Useless in afbce
25:24	R/W	3	mif_urgent : unsigned, default = 3 , head mif and body mif urgent bit[0] head axi write urgent. bit[1] body axi write urgent.
22:16	R/W	4	hold_line_num : unsigned, default = 4, hold_line_num
15:14	R/W	1	burst_mode : unsigned, default = 1, 0:burst1 1:burst2 2:burst4
0	R/W	0	reg_fmt444_comb : unsigned, default = 0, Put 444_8bit data into 422 12bit buffer,if hsize > 1024(di_afbce) Or hsize > 2048 (vdin_afbce),this bit should be high. 0: 444 8bit comb mode close 1: 444 8bit comb mode open

**Table 9-376 DI\_AFBCE\_SIZE\_IN 0x2062**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	hsize : unsigned, default = 1920 , hsize of afbce encoder buffer
12:0	R/W	1080	vsize : unsigned, default = 1080 , vsize of afbce encoder buffer

**Table 9-377 DI\_AFBCE\_BLK\_SIZE\_IN 0x2063**

Bit(s)	R/W	Default	Description
28:16	R/W	60	hblk_size : unsigned, default = 60 , blk horz size out unit: 32*4 block hblk_size of afbce encoder buffer
12:0	R/W	270	vblk_size : unsigned, default = 270, blk vert size out unit: 32*4 block vblk_size of afbce encoder buffer

**Table 9-378 DI\_AFBCE\_HEAD\_BADDR 0x2064**

Bit(s)	R/W	Default	Description
31:0	R/W	0x00	head_baddr : unsigned, default = 32'h00,head_baddr

**Table 9-379 DI\_AFBCE\_MIF\_SIZE 0x2065**

Bit(s)	R/W	Default	Description
29:28	R/W	32	ddr_blk_size : unsigned, default = 32'd1,ddr_blk_size
26:24	R/W	32	cmd_blk_size : unsigned, default = 32'd3,cmd_blk_size

Bit(s)	R/W	Default	Description
20:16	R/W	32	uncmp_size : unsigned, default = 32'd20 ,uncmp_size of a uncompressed 32*4 block uncmp_size = ((16*uncmp_bits*sblk_num+7)>>3 +31)/32 << 1
15:0	R/W	32	mmu_page_size : unsigned, default = 32'd4096,mmu_page_size 4096/8192

**Table 9-380 DI\_AFBCE\_PIXEL\_IN\_HOR\_SCOPE 0x2066**

Bit(s)	R/W	Default	Description
28:16	R/W	1919	enc_win_end_h : unsigned, default = 1919 ,pic scope hor end
12:0	R/W	0	enc_win_bgn_h : unsigned, default = 0 ,pic scope hor bgn

**Table 9-381 DI\_AFBCE\_PIXEL\_IN\_VER\_SCOPE 0x2067**

Bit(s)	R/W	Default	Description
28:16	R/W	1079	enc_win_end_v : unsigned, default = 1079 ,pic scope ver end
12:0	R/W	0	enc_win_bgn_v : unsigned, default = 0 ,pic scope ver bgn

**Table 9-382 DI\_AFBCE\_CONV\_CTRL 0x2068**

Bit(s)	R/W	Default	Description
28:16	R/W	2048	fmt_ybuf_depth : unsigned, default = 2048,fmt_ybuf_depth, fixed
11: 0	R/W	256	lbuf_depth : unsigned, default = 256, lbuf_depth, fixed

**Table 9-383 DI\_AFBCE\_MIF\_HOR\_SCOPE 0x2069**

Bit(s)	R/W	Default	Description
25:16	R/W	0	blk_end_h : unsigned, default = 0 ,blk scope horend
9:0	R/W	59	blk_bgn_h : unsigned, default = 59 ,blk scope horbgn

**Table 9-384 DI\_AFBCE\_MIF\_VER\_SCOPE 0x206a**

Bit(s)	R/W	Default	Description
27:16	R/W	0	blk_end_v : unsigned, default = 0 ,blk scope ver end
11:0	R/W	269	blk_bgn_v : unsigned, default = 269 ,blk scope ver bgn

**Table 9-385 DI\_AFBCE\_STAT1 0x206b**

Bit(s)	R/W	Default	Description
31	R.O	0	ro_frm_end_pulse1 : unsigned, RO,default = 0 ;frame end status
30:0	R.O	0	ro_dbg_top_info1 : unsigned, RO,default = 0 ,ro_dbg_top_info1

Table 9-386 DI\_AFBCE\_STAT2 0x206c

Bit(s)	R/W	Default	Description
30:0	R.O	0	ro_dbg_top_info2 : unsigned, RO,default = 0 ,ro_dbg_top_info2

Table 9-387 DI\_AFBCE\_FORMAT 0x206d

Bit(s)	R/W	Default	Description
9: 8	R/W	2	reg_format_mode : // unsigned , RW, default = 2 data format; 0 :YUV444/RGB, 1 :YUV422, 2 :YUV420,
7: 4	R/W	10	reg_compbits_c : // unsigned , RW, default = 10 chroma bitwidth 8 or 10
3: 0	R/W	10	reg_compbits_y : // unsigned , RW, default = 10 luma bitwidth 8 or 10

Table 9-388 DI\_AFBCE\_MODE\_EN 0x206e

Bit(s)	R/W	Default	Description
25	R/W	0	reg_adpt_interleave_ymode : // unsigned , RW, default = 0 force 0 to disable it: no HW implementation
24	R/W	0	reg_adpt_interleave_cmode : // unsigned , RW, default = 0 force 0 to disable it: not HW implementation
23	R/W	1	reg_adpt_yinterleave_luma_ride : // unsigned , RW, default = 1 vertical interleave piece luma reorder ride; 0: no reorder ride; 1: w/4 as ride
22	R/W	1	reg_adpt_yinterleave_chrm_ride : // unsigned , RW, default = 1 vertical interleave piece chroma reorder ride; 0: no reorder ride; 1: w/2 as ride
21	R/W	1	reg_adpt_xinterleave_luma_ride : // unsigned , RW, default = 1 vertical interleave piece luma reorder ride; 0: no reorder ride; 1: w/4 as ride
20	R/W	1	reg_adpt_xinterleave_chrm_ride : // unsigned , RW, default = 1 vertical interleave piece chroma reorder ride; 0: no reorder ride; 1: w/2 as ride
18	R/W	0	reg_disable_order_mode_i_6 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
17	R/W	0	reg_disable_order_mode_i_5 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
16	R/W	0	reg_disable_order_mode_i_4 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
15	R/W	0	reg_disable_order_mode_i_3 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
14	R/W	0	reg_disable_order_mode_i_2 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
13	R/W	0	reg_disable_order_mode_i_1 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
12	R/W	0	reg_disable_order_mode_i_0 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
10	R/W	0	reg_minval_yenc_en : // unsigned , RW, default = 0 force disable, final decision to remove this ws 1% performance loss
9	R/W	0	reg_16x4block_enable : // unsigned , RW, default = 0 block as mission, but per- mit 16x4 block

Bit(s)	R/W	Default	Description
8	R/W	0	reg_uncompress_split_mode : // unsigned , RW, default = 0 0: no split; 1: split
5	R/W	0	reg_input_padding_uv128 : // unsigned , RW, default = 0 input picture 32x4 block gap mode: 0: pad uv=0; 1: pad uv=128
4	R/W	0	reg_dwds_padding_uv128 : // unsigned , RW, default = 0 downsampled image for double write 32x gap mode: 0: pad uv=0; 1: pad uv=128
3: 1	R/W	0	reg_force_order_mode_value : // unsigned , RW, default = 0 force order mode 0~7
0	R/W	0	reg_force_order_mode_en : // unsigned , RW, default = 0 force order mode enable: 0: no force; 1: forced to force_value

Table 9-389 DI\_AFBCE\_DWSCALAR\_0x206f

Bit(s)	R/W	Default	Description
7: 6	R/W	3	reg_dwscalar_w0 : // unsigned , RW, default = 3 horizontal 1st step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg
5: 4	R/W	0	reg_dwscalar_w1 : // unsigned , RW, default = 0 horizontal 2nd step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg
3: 2	R/W	2	reg_dwscalar_h0 : // unsigned , RW, default = 2 vertical 1st step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg
1: 0	R/W	3	reg_dwscalar_h1 : // unsigned , RW, default = 3 vertical 2nd step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg

Table 9-390 DI\_AFBCE\_DEFCOLOR\_1\_0x2070

Bit(s)	R/W	Default	Description
23:12	R/W	4095	reg_enc_defalutcolor_3 : // unsigned , RW, default = 4095 Picture wise default color value in [Y Cb Cr]
11: 0	R/W	4095	reg_enc_defalutcolor_0 : // unsigned , RW, default = 4095 Picture wise default color value in [Y Cb Cr]

Table 9-391 DI\_AFBCE\_DEFCOLOR\_2\_0x2071

Bit(s)	R/W	Default	Description
23:12	R/W	2048	reg_enc_defalutcolor_2 : // unsigned , RW, default = 2048 wise default color value in [Y Cb Cr]
11: 0	R/W	2048	reg_enc_defalutcolor_1 : // unsigned , RW, default = 2048 wise default color value in [Y Cb Cr]

**Table 9-392 DI\_AFBCE\_QUANT\_ENABLE 0x2072**

Bit(s)	R/W	Default	Description
11	R/W	0	reg_quant_expand_en_1 : // unsigned , RW, default = 0 enable for quantization value expansion
10	R/W	0	reg_quant_expand_en_0 : // unsigned , RW, default = 0 enable for quantization value expansion
9: 8	R/W	0	reg_bcleav_ofst : // signed , RW, default = 0 bcleave ofset to get lower range, especially under lossy, for v1/v2, x=0 is equivalent, default = -1;
4	R/W	0	reg_quant_enable_1 : // unsigned , RW, default = 0 enable for quant to get some lossy
0	R/W	0	reg_quant_enable_0 : // unsigned , RW, default = 0 enable for quant to get some lossy

**Table 9-393 DI\_AFBCE\_IQUANT\_LUT\_1 0x2073**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_iquant_yclut_0_11 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
26:24	R/W	1	reg_iquant_yclut_0_10 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
22:20	R/W	2	reg_iquant_yclut_0_9 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
18:16	R/W	3	reg_iquant_yclut_0_8 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
14:12	R/W	4	reg_iquant_yclut_0_7 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	5	reg_iquant_yclut_0_6 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	5	reg_iquant_yclut_0_5 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	4	reg_iquant_yclut_0_4 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

**Table 9-394 DI\_AFBCE\_IQUANT\_LUT\_2 0x2074**

Bit(s)	R/W	Default	Description
14:12	R/W	3	reg_iquant_yclut_0_3 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	2	reg_iquant_yclut_0_2 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	1	reg_iquant_yclut_0_1 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	0	reg_iquant_yclut_0_0 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

**Table 9-395 DI\_AFBCE\_IQUANT\_LUT\_3 0x2075**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_iquant_yclut_1_11 : // unsigned , RW, default = 0 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
26:24	R/W	1	reg_iquant_yclut_1_10 : // unsigned , RW, default = 1 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
22:20	R/W	2	reg_iquant_yclut_1_9 : // unsigned , RW, default = 2 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
18:16	R/W	3	reg_iquant_yclut_1_8 : // unsigned , RW, default = 3 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
14:12	R/W	4	reg_iquant_yclut_1_7 : // unsigned , RW, default = 4 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
10: 8	R/W	5	reg_iquant_yclut_1_6 : // unsigned , RW, default = 5 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
6: 4	R/W	5	reg_iquant_yclut_1_5 : // unsigned , RW, default = 5 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
2: 0	R/W	4	reg_iquant_yclut_1_4 : // unsigned , RW, default = 4 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$

**Table 9-396 DI\_AFBCE\_IQUANT\_LUT\_4 0x2076**

Bit(s)	R/W	Default	Description
14:12	R/W	3	reg_iquant_yclut_1_3 : // unsigned , RW, default = 3 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
10: 8	R/W	2	reg_iquant_yclut_1_2 : // unsigned , RW, default = 2 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
6: 4	R/W	1	reg_iquant_yclut_1_1 : // unsigned , RW, default = 1 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
2: 0	R/W	0	reg_iquant_yclut_1_0 : // unsigned , RW, default = 0 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$

**Table 9-397 DI\_AFBCE\_RQUANT\_LUT\_1 0x2077**

Bit(s)	R/W	Default	Description
30:28	R/W	5	reg_rquant_yclut_0_11 : // unsigned , RW, default = 5 quantization lut for bctree leavs, $quant=2^{lut}(bc\_leav\_r+1)$ , can be calculated from $iquant\_yclut(fw\_setting)$
26:24	R/W	5	reg_rquant_yclut_0_10 : // unsigned , RW, default = 5
22:20	R/W	4	reg_rquant_yclut_0_9 : // unsigned , RW, default = 4
18:16	R/W	4	reg_rquant_yclut_0_8 : // unsigned , RW, default = 4
14:12	R/W	3	reg_rquant_yclut_0_7 : // unsigned , RW, default = 3
10: 8	R/W	3	reg_rquant_yclut_0_6 : // unsigned , RW, default = 3
6: 4	R/W	2	reg_rquant_yclut_0_5 : // unsigned , RW, default = 2
2: 0	R/W	2	reg_rquant_yclut_0_4 : // unsigned , RW, default = 2

**Table 9-398 DI\_AFBCE\_RQUANT\_LUT\_2 0x2078**

Bit(s)	R/W	Default	Description
14:12	R/W	1	reg_rquant_yclut_0_3 : // unsigned , RW, default = 1
10: 8	R/W	1	reg_rquant_yclut_0_2 : // unsigned , RW, default = 1
6: 4	R/W	0	reg_rquant_yclut_0_1 : // unsigned , RW, default = 0
2: 0	R/W	0	reg_rquant_yclut_0_0 : // unsigned , RW, default = 0

**Table 9-399 DI\_AFBCE\_RQUANT\_LUT\_3 0x2079**

Bit(s)	R/W	Default	Description
30:28	R/W	5	reg_rquant_yclut_1_11 : // unsigned , RW, default = 5 quantization lut for bctree leavs, quant=2^lut(bc_leav_r+1), can be calculated from iquant_yclut(fw_setting)
26:24	R/W	5	reg_rquant_yclut_1_10 : // unsigned , RW, default = 5
22:20	R/W	4	reg_rquant_yclut_1_9 : // unsigned , RW, default = 4
18:16	R/W	4	reg_rquant_yclut_1_8 : // unsigned , RW, default = 4
14:12	R/W	3	reg_rquant_yclut_1_7 : // unsigned , RW, default = 3
10: 8	R/W	3	reg_rquant_yclut_1_6 : // unsigned , RW, default = 3
6: 4	R/W	2	reg_rquant_yclut_1_5 : // unsigned , RW, default = 2
2: 0	R/W	2	reg_rquant_yclut_1_4 : // unsigned , RW, default = 2

**Table 9-400 DI\_AFBCE\_RQUANT\_LUT\_4 0x207a**

Bit(s)	R/W	Default	Description
14:12	R/W	1	reg_rquant_yclut_1_3 : // unsigned , RW, default = 1
10: 8	R/W	1	reg_rquant_yclut_1_2 : // unsigned , RW, default = 1
6: 4	R/W	0	reg_rquant_yclut_1_1 : // unsigned , RW, default = 0
2: 0	R/W	0	reg_rquant_yclut_1_0 : // unsigned , RW, default = 0

**Table 9-401 DI\_AFBCE\_YUV\_FORMAT\_CONV\_MODE 0x207b**

Bit(s)	R/W	Default	Description
6: 4	R/W	0	reg_444to422_mode : // unsigned , RW, default = 0
2: 0	R/W	0	reg_422to420_mode : // unsigned , RW, default = 0

**Table 9-402 DI\_AFBCE\_DUMMY\_DATA 0x207c**

Bit(s)	R/W	Default	Description
29: 0	R/W	0	reg_dummy_data : // unsigned , default = 0 ,bit[9:0] v bit[19:10] U bit[29:20] Y



Table 9-403 DI\_AFBCE\_CLR\_FLAG 0x207d

Bit(s)	R/W	Default	Description
2	R/W	0	frm_din_end_clr : // unsigned, default = 0 ;
1	R/W	0	enc_error_clr : // unsigned, default = 0 ;
0	R/W	0	frm_end_clr : // unsigned, default = 0 ;

Table 9-404 DI\_AFBCE\_STA\_FLAGT 0x207e

Bit(s)	R/W	Default	Description
31:28	R.O	0	reverse
27	R.O	0	ro_init_end_flag; //init phas end
26	R.O	0	ro_din_frm_end_flag; //data din finished
25:24	R.O	0	enc_st 0:idle 1:enc 2:hold 3:wait
23:21	R.O	0	emit_st 0:idle 1:tx_bcrot 2:tx_bc_lev 3:tx_zcd 4:tx_tree 5:tx_uncmp 6:tx_redu 7:tx_check
20:15	R.O	0	ro_cmd2bresp_cnt: // unsigned, RO,default = 0 ;
14:9	R.O	0	ro_enc2body_cnt: // unsigned, RO,default = 0 ;
8:3	R.O	0	ro_cmd2wd_cnt: // unsigned, RO,default = 0 ;
2	R.O	0	ro_wr_abort_flag: // unsigned, RO,default = 0 ;
1	R.O	0	ro_enc_error_flag: // unsigned, RO,default = 0 ;
0	R.O	0	ro_frm_end_flag: // unsigned, RO,default = 0 ;

Table 9-405 DI\_AFBCE\_MMU\_NUM 0x207f

Bit(s)	R/W	Default	Description
15: 0	R.O	0	ro_frm_mmu_num : // unsigned, RO,default = 0 ,mmu addr have been used in a frame

Table 9-406 DI\_AFBCE\_MMU\_RMIF\_CTRL1 0x2080

Bit(s)	R/W	Default	Description
25:24	R/W	0	reg_sync_sel : // unsigned , default = 0, axi canvas id sync with frm rst
23:16	R/W	0	reg_canvas_id : // unsigned , default = 0, axi canvas id num
14:12	R/W	1	reg_cmd_intr_len : // unsigned , default = 1, interrupt send cmd when how many series axi cmd, 0 =12 1=16 2=24 3=32 4=40 5=48 6=56 7=64

Bit(s)	R/W	Default	Description
11:10	R/W	1	reg_cmd_req_size : // unsigned , default = 1, how many room fifo have, then axi send series req, 0=16 1=32 2=24 3=64
9:8	R/W	2	reg_burst_len : // unsigned , default = 2, burst type: 0-single 1-bst2 2-bst4
7	R/W	0	reg_swap_64bit : // unsigned , default = 0, 64bits of 128bit swap enable
6	R/W	0	reg_little_endian : // unsigned , default = 0, big endian enable
5	R/W	0	reg_y_rev : // unsigned , default = 0, vertical reverse enable
4	R/W	0	reg_x_rev : // unsigned , default = 0, horizontal reverse enable
2:0	R/W	3	reg_pack_mode : // unsigned , default = 3, 0:4bit 1:8bit 2:16bit 3:32bit 4:64bit 5:128bit

Table 9-407 DI\_AFBCE\_MMU\_RMIF\_CTRL2 0x2081

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_sw_rst : // unsigned , default = 0,
23:18	R/W	0x0	reg_gclk_ctrl :
16:0	R/W	0	reg_urgent_ctrl : // unsigned , default = 0, urgent control reg : 16 reg_ugt_init : urgent initial value 15 reg_ugt_en : urgent enable 14 reg_ugt_type : 1= wrmif 0= rdmif 7 :4 reg_ugt_top_th: urgent top threshold 3 :0 reg_ugt_bot_th: urgent bottom threshold

Table 9-408 DI\_AFBCE\_MMU\_RMIF\_CTRL3 0x2082

Bit(s)	R/W	Default	Description
16	R/W	1	reg_acc_mode : // unsigned , default = 1,
12:0	R/W	4096	reg_stride : // unsigned , default = 4096,

Table 9-409 DI\_AFBCE\_MMU\_RMIF\_CTRL4 0x2083

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_baddr : // unsigned , default = 0, reg_mmu_baddr

Table 9-410 DI\_AFBCE\_MMU\_RMIF\_SCOPE\_X 0x2084

Bit(s)	R/W	Default	Description
28:16	R/W	4095	reg_x_end : // unsigned , default = 4095, the canvas hor end pixel position
12: 0	R/W	0	reg_x_start : // unsigned , default = 0, the canvas hor start pixel position

Table 9-411 DI\_AFBCE\_MMU\_RMIF\_SCOPE\_Y 0x2085

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_y_end : // unsigned , default = 0, the canvas ver end pixel position
12: 0	R/W	0	reg_y_start : // unsigned , default = 0, the canvas ver start pixel position

**Table 9-412 DI\_AFBCE\_MMU\_RMIF\_RO\_STAT 0x2086**

Bit(s)	R/W	Default	Description
15:0	R/W	0x0	reg_status : // unsigned , reg_status

**Table 9-413 DI\_AFBCE\_PIP\_CTRL 0x208a**

Bit(s)	R/W	Description
2	R/W	reg_enc_align_en : //unsigned , RW, default = 1,
1	R/W	reg_pip_ini_ctrl : //unsigned , RW, default = 0,
0	R/W	reg_pip_mode : //unsigned , RW, default = 0,

**Table 9-414 DI\_AFBCE\_ROT\_CTRL 0x208b**

Bit(s)	R/W	Default	Description
4	R.O	0	reg_rot_en : //unsigned , RW, default = 0, rotation enable
3:0	R/W	8	reg_vstep : //unsigned , RW, default = 8, rotation vstep ,setting according rotation shrink mode

**Table 9-415 DI\_AFBCE\_DIMM\_CTRL 0x208c**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_dimm_layer_en : //unsigned , RW, default = 0, dimm_layer enable signal
29:0	R/W	0x0008-0200	reg_dimm_data : //unsigned , RW, default = 29'h00080200, dimm_layer data

**Table 9-416 DI\_AFBCE\_BND\_DEC\_MISC 0x208d**

Bit(s)	R/W	Default	Description
27:26	R/W	0	bnd_dec_rev_mode : //unsigned , RW, default = 0 only pip mode use those bits, ususaly don't need configure
25:24	R/W	3	bnd_dec_mif_urgent : //unsigned , RW, default = 3 only pip mode use those bits, ususaly don't need configure
23:22	R/W	2	bnd_dec_burst_len : //unsigned , RW, default = 2 only pip mode use those bits, ususaly don't need configure
21:20	R/W	1	bnd_dec_ddr_blk_size : //unsigned , RW, default = 1 only pip mode use those bits, ususaly don't need configure
18:16	R/W	3	bnd_dec_cmd_blk_size : //unsigned , RW, default = 3 only pip mode use those bits, ususaly don't need configure
14	R/W	0	bnd_dec_blk_mem_mode : //unsigned , RW, default = 0 only pip mode use those bits, ususaly don't need configure
13	R/W	1	bnd_dec_addr_link_en : //unsigned , RW, default = 1 only pip mode use those bits, ususaly don't need configure

Bit(s)	R/W	Default	Description
12	R/W	0	bnd_dec_always_body_rden : //unsigned , RW,default = 0 only pip mode use those bits,usualy don't need configure
11:0	R/W	128	bnd_dec_mif_lbuf_depth : //unsigned , RW,default = 128 only pip mode use those bits,usualy don't need configure

## DI AFBCE1 Registers

Table 9-417 DI\_AFBCE1\_ENABLE 0x20c0

Bit(s)	R/W	Default	Description
31:20	R/W	0	gclk_ctrl : unsigned , default = 0, gclk_ctrl of afbe submodule : 12'h000 : gatting afbce clock. 12'hfff : free_run afbce clock. 12'h555 : close afbce clock.
19:16	R/W	0	2) afbce_sync_sel : unsigned , default = 0, shadow some size/scope/mode registers in afbce 4'hf : sync active. 4'h0: sync close.
13	R/W	0	enc_rst_mode : unsigned , default = 0, frame rst mode bits: 1:use software reset as frame reset by writing a pulse to register DI_AFBCE1_MODE[29]. 0:use vsync as frame reset.
12	R/W	0	enc_en_mode : unsigned , default = 0, DI_AFBCE1 start mode bits: 1: DI_AFBCE1 start_up by writing a pulse to register DI_AFBCE1_ENABLE[0]. 0: DI_AFBCE1 start_up several line after vsync, line number setting by DI_AFBCE1_MODE[22:16].
8	R/W	0	enc_enable : unsigned , default = 0, 1: open DI_AFBCE1 module. 0: close DI_AFBCE1 module.
0	R/W	0	enc_frm_start : DI_AFBCE1 start_up bits This bits can start afbce by writing high signal when DI_AFBCE1_ENABLE[12] equal 1.

Table 9-418 DI\_AFBCE1\_MODE 0x20c1

Bit(s)	R/W	Default	Description
31:29	R/W	0	soft_rst : unsigned, default = 0 , bit[0] async rst ,soft_ware ctrl async rst bit[1] sync rst ,soft_ware ctrl sync rst bit[2] sync rst by vsync
27:26	R/W	0	rev_mode : unsigned, default = 0 , reverse mode Useless in afbce
25:24	R/W	3	mif_urgent : unsigned, default = 3 , head mif and body mif urgent bit[0] head axi write urgent. bit[1] body axi write urgent.
22:16	R/W	4	hold_line_num : unsigned, default = 4, hold_line_num

Bit(s)	R/W	Default	Description
15:14	R/W	1	burst_mode : unsigned, default = 1, 0:burst1 1:burst2 2:burst4
0	R/W	0	reg_fmt444_comb : unsigned, default = 0, Put 444_8bit data into 422 12bit buffer,if hsize > 1024(DI_AFBCE1) Or hsize > 2048(vdin_afbce),this bit should be high. 0: 444 8bit comb mode close 1: 444 8bit comb mode open

Table 9-419 DI\_AFBCE1\_SIZE\_IN 0x20c2

Bit(s)	R/W	Default	Description
28:16	R/W	1920	hsize : unsigned, default = 1920 , hsize of afbce encoder buffer
12:0	R/W	1080	vsize : unsigned, default = 1080 , vsize of afbce encoder buffer

Table 9-420 DI\_AFBCE1\_BLK\_SIZE\_IN 0x20c3

Bit(s)	R/W	Default	Description
28:16	R/W	60	hblk_size : unsigned, default = 60 , blk horz size out unit: 32*4 block hblk_size of afbce encoder buffer
12:0	R/W	270	vblk_size : unsigned, default = 270, blk vert size out unit: 32*4 block vblk_size of afbce encoder buffer

Table 9-421 DI\_AFBCE1\_HEAD\_BADDR 0x20c4

Bit(s)	R/W	Default	Description
31:0	R/W	0x00	head_baddr : unsigned, default = 32'h00, head_baddr

Table 9-422 DI\_AFBCE1\_MIF\_SIZE 0x20c5

Bit(s)	R/W	Default	Description
29:28	R/W	32	ddr_blk_size : unsigned, default = 32'd1, ddr_blk_size
26:24	R/W	32	cmd_blk_size : unsigned, default = 32'd3, cmd_blk_size
20:16	R/W	32	uncmp_size : unsigned, default = 32'd20 ,uncmp_size of a uncompressed 32*4 block uncmp_size = ((16*uncmp_bits*sblk_num+7)>>3 +31)/32 << 1
15:0	R/W	32	mmu_page_size : unsigned, default = 32'd4096, mmu_page_size 4096/8192

Table 9-423 DI\_AFBCE1\_PIXEL\_IN\_HOR\_SCOPE 0x20c6

Bit(s)	R/W	Default	Description
28:16	R/W	1919	enc_win_end_h : unsigned, default = 1919, pic scope hor end
12:0	R/W	0	enc_win_bgn_h : unsigned, default = 0, pic scope hor bgn

**Table 9-424 DI\_AFBCE1\_PIXEL\_IN\_VER\_SCOPE 0x20c7**

Bit(s)	R/W	Default	Description
28:16	R/W	1079	enc_win_end_v : unsigned, default = 1079 ,pic scope ver end
12:0	R/W	0	enc_win_bgn_v : unsigned, default = 0 ,pic scope ver bgn

**Table 9-425 DI\_AFBCE1\_CONV\_CTRL 0x20c8**

Bit(s)	R/W	Default	Description
28:16	R/W	2048	fmt_ybuf_depth : unsigned, default = 2048,fmt_ybuf_depth,fixed
11: 0	R/W	256	lbuf_depth : unsigned, default = 256, lbuf_depth,fixed

**Table 9-426 DI\_AFBCE1\_MIF\_HOR\_SCOPE 0x20c9**

Bit(s)	R/W	Default	Description
25:16	R/W	0	blk_end_h : unsigned, default = 0 ,blk scope horend
9:0	R/W	59	blk_bgn_h : unsigned, default = 59 ,blk scope horbgn

**Table 9-427 DI\_AFBCE1\_MIF\_VER\_SCOPE 0x20ca**

Bit(s)	R/W	Default	Description
27:16	R/W	0	blk_end_v : unsigned, default = 0 ,blk scope ver end
11:0	R/W	269	blk_bgn_v : unsigned, default = 269 ,blk scope ver bgn

**Table 9-428 DI\_AFBCE1\_STAT1 0x20cb**

Bit(s)	R/W	Default	Description
31	R.O	0	ro_frm_end_pulse1 : unsigned, RO,default = 0 ;frame end status
30:0	R.O	0	ro_dbg_top_info1 : unsigned, RO,default = 0 ,ro_dbg_top_info1

**Table 9-429 DI\_AFBCE1\_STAT2 0x20cc**

Bit(s)	R/W	Default	Description
30:0	R.O	0	ro_dbg_top_info2 : unsigned, RO,default = 0 ,ro_dbg_top_info2

**Table 9-430 DI\_AFBCE1\_FORMAT 0x20cd**

Bit(s)	R/W	Default	Description
9: 8	R/W	2	reg_format_mode : // unsigned , RW, default = 2 data format; 0 :YUV444/RGB, 1 :YUV422, 2 :YUV420,
7: 4	R/W	10	reg_compbits_c : // unsigned , RW, default = 10 chroma bitwidth 8 or 10
3: 0	R/W	10	reg_compbits_y : // unsigned , RW, default = 10 luma bitwidth 8 or 10

**Table 9-431 DI\_AFBCE1\_MODE\_EN 0x20ce**

Bit(s)	R/W	Default	Description
25	R/W	0	reg_adpt_interleave_ymode : // unsigned , RW, default = 0 force 0 to disable it: no HW implementation
24	R/W	0	reg_adpt_interleave_cmode : // unsigned , RW, default = 0 force 0 to disable it: not HW implementation
23	R/W	1	reg_adpt_yinterleave_luma_ride : // unsigned , RW, default = 1 vertical interleave piece luma reorder ride; 0: no reorder ride; 1: w/4 as ride
22	R/W	1	reg_adpt_yinterleave_chrm_ride : // unsigned , RW, default = 1 vertical interleave piece chroma reorder ride; 0: no reorder ride; 1: w/2 as ride
21	R/W	1	reg_adpt_xinterleave_luma_ride : // unsigned , RW, default = 1 vertical interleave piece luma reorder ride; 0: no reorder ride; 1: w/4 as ride
20	R/W	1	reg_adpt_xinterleave_chrm_ride : // unsigned , RW, default = 1 vertical interleave piece chroma reorder ride; 0: no reorder ride; 1: w/2 as ride
18	R/W	0	reg_disable_order_mode_i_6 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
17	R/W	0	reg_disable_order_mode_i_5 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
16	R/W	0	reg_disable_order_mode_i_4 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
15	R/W	0	reg_disable_order_mode_i_3 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
14	R/W	0	reg_disable_order_mode_i_2 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
13	R/W	0	reg_disable_order_mode_i_1 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
12	R/W	0	reg_disable_order_mode_i_0 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
10	R/W	0	reg_minval_yenc_en : // unsigned , RW, default = 0 force disable, final decision to remove this ws 1% performance loss
9	R/W	0	reg_16x4block_enable : // unsigned , RW, default = 0 block as mission, but per- mit 16x4 block
8	R/W	0	reg_uncompress_split_mode : // unsigned , RW, default = 0 0: no split; 1: split
5	R/W	0	reg_input_padding_uv128 : // unsigned , RW, default = 0 input picture 32x4 block gap mode: 0: pad uv=0; 1: pad uv=128
4	R/W	0	reg_dwds_padding_uv128 : // unsigned , RW, default = 0 down sampled image for double write 32x gap mode: 0: pad uv=0; 1: pad uv=128

Bit(s)	R/W	Default	Description
3: 1	R/W	0	reg_force_order_mode_value : // unsigned , RW, default = 0 force order mode 0~7
0	R/W	0	reg_force_order_mode_en : // unsigned , RW, default = 0 force order mode enable: 0: no force; 1: forced to force_value

Table 9-432 DI\_AFBCE1\_DWSCALAR 0x20cf

Bit(s)	R/W	Default	Description
7: 6	R/W	3	reg_dwscalar_w0 : // unsigned , RW, default = 3 horizontal 1st step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg
5: 4	R/W	0	reg_dwscalar_w1 : // unsigned , RW, default = 0 horizontal 2nd step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg
3: 2	R/W	2	reg_dwscalar_h0 : // unsigned , RW, default = 2 vertical 1st step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg
1: 0	R/W	3	reg_dwscalar_h1 : // unsigned , RW, default = 3 vertical 2nd step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg

Table 9-433 DI\_AFBCE1\_DEFCOLOR\_1 0x20d0

Bit(s)	R/W	Default	Description
23:12	R/W	4095	reg_enc_defalutcolor_3 : // unsigned , RW, default = 4095 Picture wise default color value in [Y Cb Cr]
11: 0	R/W	4095	reg_enc_defalutcolor_0 : // unsigned , RW, default = 4095 Picture wise default color value in [Y Cb Cr]

Table 9-434 DI\_AFBCE1\_DEFCOLOR\_2 0x20d1

Bit(s)	R/W	Default	Description
23:12	R/W	2048	reg_enc_defalutcolor_2 : // unsigned , RW, default = 2048 wise default color value in [Y Cb Cr]
11: 0	R/W	2048	reg_enc_defalutcolor_1 : // unsigned , RW, default = 2048 wise default color value in [Y Cb Cr]

Table 9-435 DI\_AFBCE1\_QUANT\_ENABLE 0x20d2

Bit(s)	R/W	Default	Description
11	R/W	0	reg_quant_expand_en_1 : // unsigned , RW, default = 0 enable for quantization value expansion
10	R/W	0	reg_quant_expand_en_0 : // unsigned , RW, default = 0 enable for quantization value expansion
9: 8	R/W	0	reg_bcleav_ofst : // signed , RW, default = 0 bcleave ofset to get lower range, especially under lossy, for v1/v2, x=0 is equivalent, default = -1;



Bit(s)	R/W	Default	Description
4	R/W	0	reg_quant_enable_1 : // unsigned , RW, default = 0 enable for quant to get some lossy
0	R/W	0	reg_quant_enable_0 : // unsigned , RW, default = 0 enable for quant to get some lossy

Table 9-436 DI\_AFBCE1\_IQUANT\_LUT\_1 0x20d3

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_iquant_yclut_0_11 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
26:24	R/W	1	reg_iquant_yclut_0_10 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
22:20	R/W	2	reg_iquant_yclut_0_9 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
18:16	R/W	3	reg_iquant_yclut_0_8 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
14:12	R/W	4	reg_iquant_yclut_0_7 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	5	reg_iquant_yclut_0_6 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	5	reg_iquant_yclut_0_5 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	4	reg_iquant_yclut_0_4 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

Table 9-437 DI\_AFBCE1\_IQUANT\_LUT\_2 0x20d4

Bit(s)	R/W	Default	Description
14:12	R/W	3	reg_iquant_yclut_0_3 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	2	reg_iquant_yclut_0_2 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	1	reg_iquant_yclut_0_1 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	0	reg_iquant_yclut_0_0 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

Table 9-438 DI\_AFBCE1\_IQUANT\_LUT\_3 0x20d5

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_iquant_yclut_1_11 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
26:24	R/W	1	reg_iquant_yclut_1_10 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
22:20	R/W	2	reg_iquant_yclut_1_9 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

Bit(s)	R/W	Default	Description
18:16	R/W	3	reg_iquant_yclut_1_8 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
14:12	R/W	4	reg_iquant_yclut_1_7 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	5	reg_iquant_yclut_1_6 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	5	reg_iquant_yclut_1_5 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	4	reg_iquant_yclut_1_4 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

Table 9-439 DI\_AFBCE1\_IQUANT\_LUT\_4 0x20d6

Bit(s)	R/W	Default	Description
14:12	R/W	3	reg_iquant_yclut_1_3 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	2	reg_iquant_yclut_1_2 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	1	reg_iquant_yclut_1_1 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	0	reg_iquant_yclut_1_0 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

Table 9-440 DI\_AFBCE1\_RQUANT\_LUT\_1 0x20d7

Bit(s)	R/W	Default	Description
30:28	R/W	5	reg_rquant_yclut_0_11 : // unsigned , RW, default = 5 quantization lut for bctree leavs, quant=2^lut(bc_leav_r+1), can be calculated from iquant_yclut(fw_setting)
26:24	R/W	5	reg_rquant_yclut_0_10 : // unsigned , RW, default = 5
22:20	R/W	4	reg_rquant_yclut_0_9 : // unsigned , RW, default = 4
18:16	R/W	4	reg_rquant_yclut_0_8 : // unsigned , RW, default = 4
14:12	R/W	3	reg_rquant_yclut_0_7 : // unsigned , RW, default = 3
10: 8	R/W	3	reg_rquant_yclut_0_6 : // unsigned , RW, default = 3
6: 4	R/W	2	reg_rquant_yclut_0_5 : // unsigned , RW, default = 2
2: 0	R/W	2	reg_rquant_yclut_0_4 : // unsigned , RW, default = 2

Table 9-441 DI\_AFBCE1\_RQUANT\_LUT\_2 0x20d8

Bit(s)	R/W	Default	Description
14:12	R/W	1	reg_rquant_yclut_0_3 : // unsigned , RW, default = 1
10: 8	R/W	1	reg_rquant_yclut_0_2 : // unsigned , RW, default = 1
6: 4	R/W	0	reg_rquant_yclut_0_1 : // unsigned , RW, default = 0
2: 0	R/W	0	reg_rquant_yclut_0_0 : // unsigned , RW, default = 0

**Table 9-442 DI\_AFBCE1\_RQUANT\_LUT\_3 0x20d9**

Bit(s)	R/W	Default	Description
30:28	R/W	5	reg_rquant_yclut_1_11 : // unsigned , RW, default = 5 quantization lut for bctree leavs, quant=2^lut(bc_leav_r+1), can be calculated from iquant_yclut(fw_setting)
26:24	R/W	5	reg_rquant_yclut_1_10 : // unsigned , RW, default = 5
22:20	R/W	4	reg_rquant_yclut_1_9 : // unsigned , RW, default = 4
18:16	R/W	4	reg_rquant_yclut_1_8 : // unsigned , RW, default = 4
14:12	R/W	3	reg_rquant_yclut_1_7 : // unsigned , RW, default = 3
10: 8	R/W	3	reg_rquant_yclut_1_6 : // unsigned , RW, default = 3
6: 4	R/W	2	reg_rquant_yclut_1_5 : // unsigned , RW, default = 2
2: 0	R/W	2	reg_rquant_yclut_1_4 : // unsigned , RW, default = 2

**Table 9-443 DI\_AFBCE1\_RQUANT\_LUT\_4 0x20da**

Bit(s)	R/W	Default	Description
14:12	R/W	1	reg_rquant_yclut_1_3 : // unsigned , RW, default = 1
10: 8	R/W	1	reg_rquant_yclut_1_2 : // unsigned , RW, default = 1
6: 4	R/W	0	reg_rquant_yclut_1_1 : // unsigned , RW, default = 0
2: 0	R/W	0	reg_rquant_yclut_1_0 : // unsigned , RW, default = 0

**Table 9-444 DI\_AFBCE1\_YUV\_FORMAT\_CONV\_MODE 0x20db**

Bit(s)	R/W	Default	Description
6: 4	R/W	0	reg_444to422_mode : // unsigned , RW, default = 0
2: 0	R/W	0	reg_422to420_mode : // unsigned , RW, default = 0

**Table 9-445 DI\_AFBCE1\_DUMMY\_DATA 0x20dc**

Bit(s)	R/W	Default	Description
29: 0	R/W	0	reg_dummy_data : // unsigned , default = 0 ,bit[9:0] v bit[19:10] U bit[29:20] Y

**Table 9-446 DI\_AFBCE1\_CLR\_FLAG 0x20dd**

Bit(s)	R/W	Default	Description
2	R/W	0	frm_din_end_clr : // unsigned, default = 0 ;
1	R/W	0	enc_error_clr : // unsigned, default = 0 ;
0	R/W	0	frm_end_clr : // unsigned, default = 0 ;

Table 9-447 DI\_AFBCE1\_STA\_FLAGT 0x20de

Bit(s)	R/W	Default	Description
31:28	R.O	0	reverse
27	R.O	0	ro_init_end_flag://init phas end
26	R.O	0	ro_din_frm_end_flag://data din finished
25:24	R.O	0	enc_st 0:idle 1:enc 2:hold 3:wait
23:21	R.O	0	emit_st 0:idle 1:tx_bcrot 2:tx_bc_lev 3:tx_zcd 4:tx_tree 5:tx_uncmp 6:tx_redu 7:tx_check
20:15	R.O	0	ro_cmd2bresp_cnt: // unsigned, RO,default = 0 ;
14:9	R.O	0	ro_enc2body_cnt: // unsigned, RO,default = 0 ;
8:3	R.O	0	ro_cmd2wd_cnt: // unsigned, RO,default = 0 ;
2	R.O	0	ro_wr_abort_flag: // unsigned, RO,default = 0 ;
1	R.O	0	ro_enc_error_flag: // unsigned, RO,default = 0 ;
0	R.O	0	ro_frm_end_flag: // unsigned, RO,default = 0 ;

Table 9-448 DI\_AFBCE1\_MMU\_NUM 0x20df

Bit(s)	R/W	Default	Description
15: 0	R.O	0	ro_frm_mmu_num : // unsigned, RO, default = 0 ,mmu addr have been used in a frame

Table 9-449 DI\_AFBCE1\_MMU\_RMIF\_CTRL1 0x20e0

Bit(s)	R/W	Default	Description
25:24	R/W	0	reg_sync_sel : // unsigned , default = 0, axi canvas id sync with frm rst
23:16	R/W	0	reg_canvas_id : // unsigned , default = 0, axi canvas id num
14:12	R/W	1	reg_cmd_intr_len : // unsigned , default = 1, interrupt send cmd when how many series axi cmd, 0 =12 1=16 2=24 3=32 4=40 5=48 6=56 7=64
11:10	R/W	1	reg_cmd_req_size : // unsigned , default = 1, how many room fifo have, then axi send series req, 0=16 1=32 2=24 3=64
9:8	R/W	2	reg_burst_len : // unsigned , default = 2, burst type: 0-single 1-bst2 2-bst4
7	R/W	0	reg_swap_64bit : // unsigned , default = 0, 64bits of 128bit swap enable
6	R/W	0	reg_little_endian : // unsigned , default = 0, big endian enable
5	R/W	0	reg_y_rev : // unsigned , default = 0, vertical reverse enable

Bit(s)	R/W	Default	Description
4	R/W	0	reg_x_rev : // unsigned , default = 0, horizontal reverse enable
2:0	R/W	3	reg_pack_mode : // unsigned , default = 3, 0:4bit 1:8bit 2:16bit 3:32bit 4:64bit 5:128bit

Table 9-450 DI\_AFBCE1\_MMU\_RMIF\_CTRL2 0x20e1

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_sw_rst : // unsigned , default = 0,
23:18	R/W	0x0	reg_gclk_ctrl :
16:0	R/W	0	reg_urgent_ctrl : // unsigned , default = 0, urgent control reg : 16 reg_ugt_init : urgent initial value 15 reg_ugt_en : urgent enable 14 reg_ugt_type : 1= wrmif 0= rdmif 7 :4 reg_ugt_top_th: urgent top threshold 3 :0 reg_ugt_bot_th: urgent bottom threshold

Table 9-451 DI\_AFBCE1\_MMU\_RMIF\_CTRL3 0x20e2

Bit(s)	R/W	Default	Description
16	R/W	1	reg_acc_mode : // unsigned , default = 1,
12:0	R/W	4096	reg_stride : // unsigned , default = 4096,

Table 9-452 DI\_AFBCE1\_MMU\_RMIF\_CTRL4 0x20e3

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_baddr : // unsigned , default = 0, reg_mmu_baddr

Table 9-453 DI\_AFBCE1\_MMU\_RMIF\_SCOPE\_X 0x20e4

Bit(s)	R/W	Default	Description
28:16	R/W	4095	reg_x_end : // unsigned , default = 4095, the canvas hor end pixel position
12: 0	R/W	0	reg_x_start : // unsigned , default = 0, the canvas hor start pixel position

Table 9-454 DI\_AFBCE1\_MMU\_RMIF\_SCOPE\_Y 0x20e5

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_y_end : // unsigned , default = 0, the canvas ver end pixel position
12: 0	R/W	0	reg_y_start : // unsigned , default = 0, the canvas ver start pixel position

Table 9-455 DI\_AFBCE1\_MMU\_RMIF\_RO\_STAT 0x20e6

Bit(s)	R/W	Default	Description
15:0	R/W	0x0	reg_status : // unsigned , reg_status

**Table 9-456 DI\_AFBCE1\_PIP\_CTRL 0x20ea**

Bit(s)	R/W	Default	Description
2	R/W		reg_enc_align_en : //unsigned , RW,default = 1,
1	R/W		reg_pip_ini_ctrl : //unsigned , RW,default = 0,
0	R/W		reg_pip_mode : //unsigned , RW,default = 0,

**Table 9-457 DI\_AFBCE1\_ROT\_CTRL 0x20eb**

Bit(s)	R/W	Default	Description
4	R.O	0	reg_rot_en : //unsigned , RW,default = 0, rotation enable
3:0	R/W	8	reg_vstep : //unsigned , RW,default = 8, rotation vstep ,setting acorrding rotation shrink mode

**Table 9-458 DI\_AFBCE1\_DIMM\_CTRL 0x20ec**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_dimm_layer_en : //unsigned , RW,default = 0,dimm_layer enable signal
29:0	R/W	0x0008-0200	reg_dimm_data : //unsigned , RW,default = 29'h00080200,dimm_layer data

**Table 9-459 DI\_AFBCE1\_BND\_DEC\_MISC 0x20ed**

Bit(s)	R/W	Default	Description
27:26	R/W	0	bnd_dec_rev_mode : //unsigned , RW,default = 0 only pip mode use those bits, ususaly don't need configure
25:24	R/W	3	bnd_dec_mif_urgent : //unsigned , RW,default = 3 only pip mode use those bits, ususaly don't need configure
23:22	R/W	2	bnd_dec_burst_len : //unsigned , RW,default = 2 only pip mode use those bits, ususaly don't need configure
21:20	R/W	1	bnd_dec_ddr_blk_size : //unsigned , RW,default = 1 only pip mode use those bits,ususaly don't need configure
18:16	R/W	3	bnd_dec_cmd_blk_size : //unsigned , RW,default = 3 only pip mode use those bits,ususaly don't need configure
14	R/W	0	bnd_dec_blk_mem_mode : //unsigned , RW,default = 0 only pip mode use those bits,ususaly don't need configure
13	R/W	1	bnd_dec_addr_link_en : //unsigned , RW,default = 1 only pip mode use those bits,ususaly don't need configure
12	R/W	0	bnd_dec_always_body_rden : //unsigned , RW,default = 0 only pip mode use those bits,ususaly don't need configure
11:0	R/W	128	bnd_dec_mif_lbuf_depth : //unsigned , RW,default = 128 only pip mode use those bits,ususaly don't need configure

### 9.2.3.4 VIU Top-Level Registers

#### Register Address

- VIU\_SW\_RESET 0xff006804
- VIU\_SW\_RESET0 0xff006808
- VIU\_MISC\_CTRL0 0xff006818
- VIU\_MISC\_CTRL1 0xff00681c
- VIUB\_SW\_RESET 0xff008004
- VIUB\_SW\_RESET0 0xff008008
- DI\_AFBCE\_CTRL 0xff00800c
- VIUB\_MISC\_CTRL0 0xff008018
- VIUB\_GCLK\_CTRL0 0xff00801c
- VIUB\_GCLK\_CTRL2 0xff008024
- VIUB\_GCLK\_CTRL3 0xff008028
- DI\_DBG\_CTRL 0xff00802c
- DI\_DBG\_CTRL1 0xff008030
- DI\_DBG\_SRDY\_INF 0xff008034
- DI\_DBG\_RRDY\_INF 0xff008038

#### Register Description

**Table 9-460 VIU\_SW\_RESET 0x1A01**

Bit(s)	R/W	Default	Description
31	R/W	0	Osd1 afbcd reset
30	R/W	0	hist_spl reset
29	R/W	0	Ldim stts reset
8	R/W	0	Vd2 Dos afbcd reset
7	R/W	0	vpp_reset
6	R/W	0	di_dsr1to2_reset
5	R/W	0	vd2_fmt_reset
4	R/W	0	vd2_reset
3	R/W	0	vd1_fmt_reset
2	R/W	0	vd1_reset
1	R/W	0	osd2_reset
0	R/W	0	osd1_reset

**Table 9-461 VIU\_SW\_RESET0 0x1A02**

Bit(s)	R/W	Default	Description
2	R/W	0	Vd1 Dos afbcd reset

**Table 9-462 VIU\_MISC\_CTRL0 0x1a06**

Bit(s)	R/W	Default	Description
17	R/W	0	Vdin0_wr_out_ctrl : 0: nr_inp to vdin 1: vdin wr dout
16	R/W	0	Afbc_inp_sel : 0: mif to INP 1: afbc to INP
16	R/W	0	di_mif0_en: vd1(afbc) to di post(if0) enable
8	R/W	0	vsync_int_ctrl : default = 0
6:5	R/W	2	0: close mif data 1: mif to tbf_downscale: 2: mif to nr , 3: mif to tbf_downscale & NR
4	R/W		Vpp_di_mif0_sel 0:buf0_data from di_mif0 1: buf0_data from din_post_din
0	R/W	0	scan_reg : default = 0

**Table 9-463 VIU\_MISC\_CTRL1 0x1A07**

Bit(s)	R/W	Default	Description
27-22	R/W	0	afbc gate clk ctrl
15:14	R/W	0	Mali afbcd clock gate control
12	R/W	0	Osd1 axi bus select 1 : select mali afbcd 0 : normal osd1
11:8	R/W	0	di_mad_en: di post to vpp enable
7-2	R/W	0	Afbcd2 Clock gate control
1	R/W	0	1 : connect dos afbcd2 to vpp vd1, 0 : connect mif to vpp vd1
0	R/W	0	1 : Dos afbcd2 output to di ; 0 : dos afbcd2 output to vpp

**Table 9-464 VIUB\_SW\_RESET 0x2001**

Bit(s)	R/W	Default	Description
17	R/W	0	Vd_afbcd m5 reset
16	R/W	0	Vd_afbcd m4 reset
15	R/W	0	Vd_afbcd m3 reset
14	R/W	0	Vd_afbcd m2 reset
13	R/W	0	Vd_afbcd m1 reset
12	R/W	0	Vd_afbcd m0 reset
4	R/W	0	Nr top reset
3	R/W	0	Di axi register reset
2	R/W	0	Di axi reset
0	R/W	0	Di top reset



**Table 9-465 DI\_AFBCE\_CTRL 0x2003**

Bit(s)	R/W	Default	Description
31	R/W	0	inp_2vdin_sel : 1: inp mif to VDIN

**Table 9-466 VIUB\_MISC\_CTRL0 0x2006**

Bit(s)	R/W	Default	Description
17	R/W	0	input2pre enable: 1->di inp data from vdin0 0->di inp data from inp_afbc(see bit16)
16	R/W	0	AFBC_INP_SEL: 1->di inp_afbc data from afbc 0->di inp_afbc data from inp mif(DDR)
6:5	R/W	0	0:mask input 1:buf_data ->nr_dsacle 2: buf_data->NR 3: buf_data->nr_dsacle
4	R/W	0	0:post_din to buf 1: di_mif0 -> buf
3-2	R/W	0	Fix_disable:dein_wr_mif
1-0	R/W	0	Fix_disable: di_nr_wr_mif

**Table 9-467 VIUB\_GCLK\_CTRL0 0x2007**

Bit(s)	R/W	Default	Description
17:16	R/W	0	di arbiter gate clock
15	R/W	0	Di_gate_all,for old di
14	R/W	1	Di_no_clk_gate,for old di
9	R/W	0	Mad post clock enable,from mad clock
8	R/W	0	Mad pre clock enable,from mad clock
7:1	R/W		reserved

**Table 9-468 VIUB\_GCLK\_CTRL2 0x2009**

Bit(s)	R/W	Default	Description
31-14	R/W	0	reserved
19-18	R/W	0	XLR
17-16	R/W	0	CUE blend
15-14	R/W	0	tbf
13-12	R/W	0	mcdi clock gate ctrl 2'b00: gate clock ,2'b01: close clock,2'b1x: always clock
11:10	R/W	0	Nr blend clock gate ctrl 2'b00: gate clock ,2'b01: close clock,2'b1x: always clock
9-8	R/W	0	Dnr clock gate ctrl 2'b00: gate clock ,2'b01: close clock,2'b1x: always clock
7-6	R/W	0	nning clock gate ctrl 2'b00: gate clock ,2'b01: close clock,2'b1x: always clock
5-4	R/W	0	Mtn det clock gate ctrl 2'b00: gate clock ,2'b01: close clock,2'b1x: always clock
3-2	R/W	0	pd clock gate ctrl 2'b00: gate clock ,2'b01: close clock,2'b1x: always clock
1-0	R/W	0	Nr clock gate ctrl 2'b00: gate clock ,2'b01: close clock,2'b1x: always clock

**Table 9-469 VIUB\_GCLK\_CTRL3 0x200a**

Bit(s)	R/W	Default	Description
31-6	R/W	0	Reserved
5-4	R/W	0	Ei clock gate ctrl 2'b00: gate clock ,2'b01: close clock,2'b1x: always clock
3-2	R/W	0	Ei_0 clock gate ctrl 2'b00: gate clock ,2'b01: close clock,2'b1x: always clock
1-0	R/W	0	Di blend clock gate ctrl 2'b00: gate clock ,2'b01: close clock,2'b1x: always clock

**Table 9-470 DI\_DBG\_CTRL 0x200B**

Bit(s)	R/W	Default	Description
31:7	R/W	0	Reserved
6	R/W	0	go_field_sel : 1: post_frame_rst 0: pre_frame_rst
5:0	R/W	0	Debug_sel : 0:diwr_out_mif 1: mtn1_din 2:mtn1_dout 3:nrdm_bld 4:dnr_dm 5:dnr_din 6:ds_dout 7:ds_din 8:deband_din 9:me_mtn 10:mc_info_mif 11:me_din_pipe 12:buf1_dout 13:mem2post 14:cont_buf 15:nr_wr_out 16:mtn_din 17:me_din 18:nr_din 19:dnr_out 20:nr_out 21:contp_mif 22:chan2 23:di_inp_mif 24:di_mem_mif 25:contp2_mif 26:contp_mif 27:chan2_mif 28:di_inp 29:di_mem 30:mcinfo_pipe 31:mad_dout 32:deint_pipe 33:mtnp_mif 34:blend_1_dout 35:blend_buf1_din 36:direct_mux 37:ei_fifo_dout 38:di_buf0 39:mif_mtnp 40:buf2_pipe 41:buf1_pipe 42:mcvec_pipe 43:buf0_pipe

**Table 9-471 DI\_DBG\_CTRL1 0x200C**

Bit(s)	R/W	Default	Description
31:16	R/W	0	Rrdy_to_srdy_max_num
15:0	R/W	0	Srdy_to_rrdy_max_num

**Table 9-472 DI\_DBG\_SRDY\_INF 0x200D**

Bit(s)	R/W	Default	Description
31:16	RO	0	wait_rrdy_bmax_dbg_cnt
15:0	RO	0	Srdy_to_rrdy_dbg_cnt

**Table 9-473 DI\_DBG\_RRDY\_INF 0x200E**

Bit(s)	R/W	Default	Description
31:16	RO	0	Wait_srdy_bmax_dbg_cnt
15:0	RO	0	Rrdy_to_srdy_dbg_cnt

### 9.2.3.5 DI\_AXI\_ARB Registers

#### Register Address

- DI\_RDARB\_MODE\_L1C1 0xff008140
- DI\_RDARB\_REQEN\_SLV\_L1C1 0xff008144
- DI\_RDARB\_WEIGH0\_SLV\_L1C1 0xff008148
- DI\_RDARB\_WEIGH1\_SLV\_L1C1 0xff00814c
- DI\_WRARB\_MODE\_L1C1 0xff008150
- DI\_WRARB\_REQEN\_SLV\_L1C1 0xff008154
- DI\_WRARB\_WEIGH0\_SLV\_L1C1 0xff008158
- DI\_WRARB\_WEIGH1\_SLV\_L1C1 0xff00815c
- DI\_RDWR\_ARB\_STATUS\_L1C1 0xff008160
- DI\_ARB\_DBG\_CTRL\_L1C1 0xff008164
- DI\_ARB\_DBG\_STAT\_L1C1 0xff008168
- DI\_RDARB\_UGT\_L1C1 0xff00816c
- DI\_RDARB\_LIMT0\_L1C1 0xff008170
- DI\_WRARB\_UGT\_L1C1 0xff008174
- DI\_SUB\_RDARB\_MODE 0xff00df00
- DI\_SUB\_RDARB\_REQEN\_SLV 0xff00df04
- DI\_SUB\_RDARB\_WEIGH0\_SLV 0xff00df08
- DI\_SUB\_RDARB\_WEIGH1\_SLV 0xff00df0c
- DI\_SUB\_RDARB\_UGT 0xff00df10
- DI\_SUB\_RDARB\_LIMT0 0xff00df14
- DI\_SUB\_WRARB\_MODE 0xff00df18
- DI\_SUB\_WRARB\_REQEN\_SLV 0xff00df1c

- DI\_SUB\_WRARB\_WEIGH0\_SLV 0xff00df20
- DI\_SUB\_WRARB\_WEIGH1\_SLV 0xff00df24
- DI\_SUB\_WRARB\_UGT 0xff00df28
- DI\_SUB\_RDWR\_ARB\_STATUS 0xff00df2c
- DI\_SUB\_ARB\_DBG\_CTRL 0xff00df30
- DI\_SUB\_ARB\_DBG\_STAT 0xff00df34
- CONTRD\_CTRL1 0xff00df40
- CONTRD\_CTRL2 0xff00df44
- CONTRD\_SCOPE\_X 0xff00df48
- CONTRD\_SCOPE\_Y 0xff00df4c
- CONTRD\_RO\_STAT 0xff00df50
- CONT2RD\_CTRL1 0xff00df54
- CONT2RD\_CTRL2 0xff00df58
- CONT2RD\_SCOPE\_X 0xff00df5c
- CONT2RD\_SCOPE\_Y 0xff00df60
- CONT2RD\_RO\_STAT 0xff00df64
- MTNRD\_CTRL1 0xff00df68
- MTNRD\_CTRL2 0xff00df6c
- MTNRD\_SCOPE\_X 0xff00df70
- MTNRD\_SCOPE\_Y 0xff00df74
- MTNRD\_RO\_STAT 0xff00df78
- MCVECRD\_CTRL1 0xff00df7c
- MCVECRD\_CTRL2 0xff00df80
- MCVECRD\_SCOPE\_X 0xff00df84
- MCVECRD\_SCOPE\_Y 0xff00df88
- MCVECRD\_RO\_STAT 0xff00df8c
- MCINFRD\_CTRL1 0xff00df90
- MCINFRD\_CTRL2 0xff00df94
- MCINFRD\_SCOPE\_X 0xff00df98
- MCINFRD\_SCOPE\_Y 0xff00df9c
- MCINFRD\_RO\_STAT 0xff00dfa0
- CONTWR\_X 0xff00dfa4
- CONTWR\_Y 0xff00dfa8
- CONTWR\_CTRL 0xff00dfac
- CONTWR\_CAN\_SIZE 0xff00dfb0
- MTNWR\_X 0xff00dfb4
- MTNWR\_Y 0xff00dfb8
- MTNWR\_CTRL 0xff00dfbc
- MTNWR\_CAN\_SIZE 0xff00dfc0
- MCVECWR\_X 0xff00dfc4
- MCVECWR\_Y 0xff00dfc8
- MCVECWR\_CTRL 0xff00dfcc
- MCVECWR\_CAN\_SIZE 0xff00dfd0
- MCINFWR\_X 0xff00dfd4

- MCINFWR\_Y 0xff00dfd8
- MCINFWR\_CTRL 0xff00dfdc
- MCINFWR\_CAN\_SIZE 0xff00dfe0

## Register Description

**Table 9-474 DI\_RDARB\_MODE\_L1C1 0x2050**

Bit(s)	R/W	Default	Description
21:16	R/W	0	rdarb_sel : uns, default = 0 , rdarb_sel [0]==0 slave dc0 connect master port0 rdarb_sel[0]==1 slave dc0 connect master port1 rdarb_sel [1]==0 slave dc1 connect master port0 rdarb_sel[1]==1 slave dc1 connect master port1 rdarb_sel [2]==0 slave dc2 connect master port0 rdarb_sel[2]==1 slave dc2 connect master port1 rdarb_sel [3]==0 slave dc3 connect master port0 rdarb_sel[3]==1 slave dc3 connect master port1 rdarb_sel [4]==0 slave dc4 connect master port0 rdarb_sel [4]==1 slave dc4 connect master port1 rdarb_sel [5]==0 slave dc5 connect master port0 rdarb_sel[5]==1 slave dc5 connect master port1
9:8	R/W	0	rdarb_arb_mode : uns, default = 0 , rdarb_arb_mode [0] master port0 arb way, rdarb_arb_mode [1] master port1 arb way,
3:0	R/W	0	rdarb_gate_clk_ctrl : uns, default = 0 , rdarb_gate_clk_ctrl [1:0] master port0 clk gate control rdarb_gate_clk_ctrl [3:2] master port1 clk gate control

**Table 9-475 DI\_RDARB\_REQEN\_SLV\_L1C1 0x2051**

Bit(s)	R/W	Default	Description
11:0	R/W	0xffff	rdarb_dc_req_en : unsigned , default = 12'hfff rdarb_dc_req_en [0]: the slv0 req to mst port0 enable, rdarb_dc_req_en [1]: the slv1 req to mst port0 enable, rdarb_dc_req_en [2]: the slv2 req to mst port0 enable, rdarb_dc_req_en [3]: the slv3 req to mst port0 enable, rdarb_dc_req_en [4]: the slv4 req to mst port0 enable, rdarb_dc_req_en [5]: the slv5 req to mst port0 enable, rdarb_dc_req_en [6]: the slv0 req to mst port1 enable, rdarb_dc_req_en [7]: the slv1 req to mst port1 enable, rdarb_dc_req_en [8]: the slv2 req to mst port1 enable, rdarb_dc_req_en [9]: the slv3 req to mst port1 enable, rdarb_dc_req_en [10]: the slv4 req to mst port1 enable, rdarb_dc_req_en [11]: the slv5 req to mst port1 enable,

**Table 9-476 DI\_RDARB\_WEIGH0\_SLV\_L1C1 0x2052**

Bit(s)	R/W	Default	Description
29:0	R/W	0	rddc_weigh_sxn : unsigned , default = 0 rddc_weigh_sxn [0*6+:6]: the slv0 req weigh number rddc_weigh_sxn [1*6+:6]: the slv1 req weigh number rddc_weigh_sxn [2*6+:6]: the slv2 req weigh number rddc_weigh_sxn [3*6+:6]: the slv3 req weigh number rddc_weigh_sxn [4*6+:6]: the slv4 req weigh number

**Table 9-477 DI\_RDARB\_WEIGH1\_SLV\_L1C1 0x2053**

Bit(s)	R/W	Default	Description
5:0	R/W	0	rddc_weigh_sxn : unsigned , default = 0 rddc_weigh_sxn [5*6+:6]: the slv5 req weigh number

Table 9-478 DI\_WRARB\_MODE\_L1C1 0x2054

Bit(s)	R/W	Default	Description
21:16	R/W	0	wrarb_sel : uns, default = 0 , wrarb_sel [0]==0 slave dc0 connect master port0 wrarb_sel[0]==1 slave dc0 connect master port1 wrarb_sel [1]==0 slave dc1 connect master port0 wrarb_sel[1]==1 slave dc1 connect master port1 wrarb_sel [2]==0 slave dc2 connect master port0 wrarb_sel[2]==1 slave dc2 connect master port1 wrarb_sel [3]==0 slave dc3 connect master port0 wrarb_sel[3]==1 slave dc3 connect master port1 wrarb_sel [4]==0 slave dc4 connect master port0 wrarb_sel[4]==1 slave dc4 connect master port1 wrarb_sel [5]==0 slave dc5 connect master port0 wrarb_sel[5]==1 slave dc5 connect master port1
9:8	R/W	0	wrarb_arb_mode : uns, default = 0 , wrarb_arb_mode [0] master port0 arb way, wrarb_arb_mode [1] master port1 arb way,
3:0	R/W	0	wrarb_gate_clk_ctrl : uns, default = 0 , wrarb_gate_clk_ctrl [1:0] master port0 clk gate control wrarb_gate_clk_ctrl [3:2] master port1 clk gate control

Table 9-479 DI\_WRARB\_REQEN\_SLV\_L1C1 0x2055

Bit(s)	R/W	Default	Description
11:0	R/W	0	wrarb_dc_req_en : unsigned , default = 0 wrarb_dc_req_en [0]: the slv0 req to mst port0 enable, wrarb_dc_req_en [1]: the slv1 req to mst port0 enable, wrarb_dc_req_en [2]: the slv2 req to mst port0 enable, wrarb_dc_req_en [3]: the slv3 req to mst port0 enable, wrarb_dc_req_en [4]: the slv4 req to mst port0 enable, wrarb_dc_req_en [5]: the slv5 req to mst port0 enable, wrarb_dc_req_en [0]: the slv0 req to mst port1 enable, wrarb_dc_req_en [1]: the slv1 req to mst port1 enable, wrarb_dc_req_en [2]: the slv2 req to mst port1 enable, wrarb_dc_req_en [3]: the slv3 req to mst port1 enable, wrarb_dc_req_en [4]: the slv4 req to mst port1 enable, wrarb_dc_req_en [5]: the slv5 req to mst port1 enable,

Table 9-480 DI\_WRARB\_WEIGH0\_SLV\_L1C1 0x2056

Bit(s)	R/W	Default	Description
29:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 wrdc_weigh_sxn [0*6+:6]: the slv0 req weigh number wrdc_weigh_sxn [1*6+:6]: the slv1 req weigh number wrdc_weigh_sxn [2*6+:6]: the slv2 req weigh number wrdc_weigh_sxn [3*6+:6]: the slv3 req weigh number wrdc_weigh_sxn [4*6+:6]: the slv4 req weigh number

Table 9-481 DI\_WRARB\_WEIGH1\_SLV\_L1C1 0x2057

Bit(s)	R/W	Default	Description
5:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 wrdc_weigh_sxn [5*6+:6]: the slv5 req weigh number

Table 9-482 DI\_RDWR\_ARB\_STATUS\_L1C1 0x2058

Bit(s)	R/W	Default	Description
3:2	R/W	0	wrarb_arb_busy : unsigned , default = 0
1:0	R/W	0	rdarb_arb_busy : unsigned , default = 0

**Table 9-483 DI\_ARB\_DBG\_CTRL\_L1C1 0x2059**

Bit(s)	R/W	Default	Description
31:0	R/W	8	det_cmd_ctrl : unsigned , default = 8

**Table 9-484 DI\_ARB\_DBG\_STAT\_L1C1 0x205a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	det_dbg_stat : unsigned , default = 0

**Table 9-485 DI\_RDARB\_UGT\_L1C1 0x205b**

Bit(s)	R/W	Default	Description
15:0	R/W	0x0	rdarb_ugt_basic : unsigned , default = {8{2'b1}};

**Table 9-486 DI\_RDARB\_LIMT0\_L1C1 0x205c**

Bit(s)	R/W	Default	Description
31:0	R/W	0x0	rdarb_req_limt_num : unsigned , default = {2{16'h3f3f}};

**Table 9-487 DI\_WRARB\_UGT\_L1C1 0x205d**

Bit(s)	R/W	Default	Description
11:0	R/W	0	wrarb_ugt_basic : unsigned , default = 0

**Table 9-488 DI\_SUB\_RDARB\_MODE 0x37c0**

Bit(s)	R/W	Default	Description
23:16	R/W	0	rdarb_sel : uns, default = 0 , rdarb_sel [0]==0 slave dc0 connect master port0 rdarb_sel[0]==1 slave dc0 connect master port1 rdarb_sel [1]==0 slave dc1 connect master port0 rdarb_sel[1]==1 slave dc1 connect master port1 rdarb_sel [2]==0 slave dc2 connect master port0 rdarb_sel[2]==1 slave dc2 connect master port1 rdarb_sel [3]==0 slave dc3 connect master port0 rdarb_sel[3]==1 slave dc3 connect master port1 rdarb_sel [4]==0 slave dc4 connect master port0 rdarb_sel [4]==1 slave dc4 connect master port1 rdarb_sel [5]==0 slave dc5 connect master port0 rdarb_sel[5]==1 slave dc5 connect master port1 rdarb_sel [6]==0 slave dc5 connect master port0 rdarb_sel[6]==1 slave dc6 connect master port1 rdarb_sel [7]==0 slave dc5 connect master port0 rdarb_sel[7]==1 slave dc7 connect master port1
9:8	R/W	0	rdarb_arb_mode : uns, default = 0 , rdarb_arb_mode [0] master port0 arb way, rdarb_arb_mode [1] master port1 arb way,
3:0	R/W	0	rdarb_gate_clk_ctrl : uns, default = 0 , rdarb_gate_clk_ctrl [1:0] master port0 clk gate control rdarb_gate_clk_ctrl [3:2] master port1 clk gate control

**Table 9-489 DI\_SUB\_RDARB\_REQEN\_SLV 0x37c1**

Bit(s)	R/W	Default	Description
15:0	R/W	0xffff	rdarb_dc_req_en : uns, default = 16'hffff , slv0~slv7 enable to mst.

Table 9-490 DI\_SUB\_RDARB\_WEIGHT0\_SLV 0x37c2

Bit(s)	R/W	Default	Description
29:0	R/W	0	rddc_weigh_sxn : unsigned , default = 0 rddc_weigh_sxn [0*6+:6]: the slv0 req weigh number rddc_weigh_sxn [1*6+:6]: the slv1 req weigh number rddc_weigh_sxn [2*6+:6]: the slv2 req weigh number rddc_weigh_sxn [3*6+:6]: the slv3 req weigh number rddc_weigh_sxn [4*6+:6]: the slv4 req weigh number

Table 9-491 DI\_SUB\_RDARB\_WEIGHT1\_SLV 0x37c3

Bit(s)	R/W	Default	Description
17:0	R/W	0	rddc_weigh_sxn : unsigned , default = 0 rddc_weigh_sxn [0*6+:6]: the slv6 req weigh number rddc_weigh_sxn [1*6+:6]: the slv7 req weigh number rddc_weigh_sxn [2*6+:6]: the slv8 req weigh number

Table 9-492 DI\_SUB\_RDARB\_UGT 0x37c4

Bit(s)	R/W	Default	Description
15:0	R/W	0x0	rdarb_ugt_basic : unsigned , default = {8{2'h1}} rdarb_ugt_basic [0*1+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [1*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [2*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [3*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [4*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [5*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [6*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [7*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen

Table 9-493 DI\_SUB\_RDARB\_LIMT0 0x37c5

Bit(s)	R/W	Default	Description
31:0	R/W	0x0	rdarb_req_limt_num : unsigned , default = {2{16'h3f3f}}

Table 9-494 DI\_SUB\_WRARB\_MODE 0x37c6

Bit(s)	R/W	Default	Description
21:16	R/W	0	wrarb_sel : uns, default = 0 , wrarb_sel [0]==0 slave dc0 connect master port0 wrarb_sel[0]==1 slave dc0 connect master port1 wrarb_sel [1]==0 slave dc1 connect master port0 wrarb_sel[1]==1 slave dc1 connect master port1 wrarb_sel [2]==0 slave dc2 connect master port0 wrarb_sel[2]==1 slave dc2 connect master port1 wrarb_sel [3]==0 slave dc3 connect master port0 wrarb_sel[3]==1 slave dc3 connect master port1 wrarb_sel [4]==0 slave dc4 connect master port0 wrarb_sel[4]==1 slave dc4 connect master port1 wrarb_sel [5]==0 slave dc5 connect master port0 wrarb_sel[5]==1 slave dc5 connect master port1
8	R/W	0	wrarb_arb_mode : uns, default = 0 , wrarb_arb_mode[0] master port0 arb way,
1:0	R/W	0	wrarb_gate_clk_ctrl : uns, default = 0 , wrarb_gate_clk_ctrl[1:0] master port0 clk gate control



Table 9-495 DI\_SUB\_WRARB\_REQEN\_SLV 0x37c7

Bit(s)	R/W	Default	Description
5:0	R/W	0	wrarb_dc_req_en : unsigned , default = 0 wrarb_dc_req_en [0]: the slv0 req to mst port0 enable, wrarb_dc_req_en [1]: the slv1 req to mst port0 enable, wrarb_dc_req_en [2]: the slv2 req to mst port0 enable, wrarb_dc_req_en [0]: the slv0 req to mst port1 enable, wrarb_dc_req_en [1]: the slv1 req to mst port1 enable, wrarb_dc_req_en [2]: the slv2 req to mst port1 enable,

Table 9-496 DI\_SUB\_WRARB\_WEIGH0\_SLV 0x37c8

Bit(s)	R/W	Default	Description
29:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 wrdc_weigh_sxn [0*6+:6]: the slv0 req weigh number wrdc_weigh_sxn [1*6+:6]: the slv1 req weigh number wrdc_weigh_sxn [2*6+:6]: the slv2 req weigh number wrdc_weigh_sxn [3*6+:6]: the slv3 req weigh number wrdc_weigh_sxn [4*6+:6]: the slv4 req weigh number

Table 9-497 DI\_SUB\_WRARB\_WEIGH1\_SLV 0x37c9

Bit(s)	R/W	Default	Description
5:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 the slv5 req weigh number

Table 9-498 DI\_SUB\_WRARB\_UGT 0x37ca

Bit(s)	R/W	Default	Description
11:0	R/W	0x0	rdarb_ugt_basic : unsigned , default = {8'2'h1} rdarb_ugt_basic [0*1+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [1*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [2*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [3*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [4*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen rdarb_ugt_basic [5*2+:2]: 00 : use auto fifo arugt generate the output arugt. 01 : use the register bit control 10 : use the input arguen

Table 9-499 DI\_SUB\_RDWR\_ARB\_STATUS 0x37cb

Bit(s)	R/W	Default	Description
2	R.O	0	ro_wrarb_arb_busy : unsigned , default = 0
1	R/W	0x0	reserve :
0	R.O	0	ro_rdarb_arb_busy : unsigned , default = 0

Table 9-500 DI\_SUB\_ARB\_DBG\_CTRL 0x37cc

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_det_cmd_ctrl : unsigned , default = 0

Table 9-501 DI\_SUB\_ARB\_DBG\_STAT 0x37cd

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_det_dbg_stat : unsigned , default = 0

### 9.2.3.6 DI\_SUB\_MIF Registers

#### Register Address

- CONTRD\_CTRL1 0xff00df40
- CONTRD\_CTRL2 0xff00df44
- CONTRD\_SCOPE\_X 0xff00df48
- CONTRD\_SCOPE\_Y 0xff00df4c
- CONTRD\_RO\_STAT 0xff00df50
- CONT2RD\_CTRL1 0xff00df54
- CONT2RD\_CTRL2 0xff00df58
- CONT2RD\_SCOPE\_X 0xff00df5c
- CONT2RD\_SCOPE\_Y 0xff00df60
- CONT2RD\_RO\_STAT 0xff00df64
- MTNRD\_CTRL1 0xff00df68
- MTNRD\_CTRL2 0xff00df6c
- MTNRD\_SCOPE\_X 0xff00df70
- MTNRD\_SCOPE\_Y 0xff00df74
- MTNRD\_RO\_STAT 0xff00df78
- MCVECRD\_CTRL1 0xff00df7c
- MCVECRD\_CTRL2 0xff00df80
- MCVECRD\_SCOPE\_X 0xff00df84
- MCVECRD\_SCOPE\_Y 0xff00df88
- MCVECRD\_RO\_STAT 0xff00df8c
- MCINFRD\_CTRL1 0xff00df90
- MCINFRD\_CTRL2 0xff00df94
- MCINFRD\_SCOPE\_X 0xff00df98
- MCINFRD\_SCOPE\_Y 0xff00df9c
- MCINFRD\_RO\_STAT 0xff00dfa0
- CONTWR\_X 0xff00dfa4
- CONTWR\_Y 0xff00dfa8
- CONTWR\_CTRL 0xff00dfac
- CONTWR\_CAN\_SIZE 0xff00dfb0
- MTNWR\_X 0xff00dfb4
- MTNWR\_Y 0xff00dfb8
- MTNWR\_CTRL 0xff00dfbc
- MTNWR\_CAN\_SIZE 0xff00dfc0
- MCVECWR\_X 0xff00dfc4
- MCVECWR\_Y 0xff00dfc8

- MCVECWR\_CTRL 0xff0dfcc
- MCVECWR\_CAN\_SIZE 0xff0dfd0
- MCINFWR\_X 0xff0dfd4
- MCINFWR\_Y 0xff0dfd8
- MCINFWR\_CTRL 0xff0dfdc
- MCINFWR\_CAN\_SIZE 0xff0dfe0
- NRDSWR\_X 0xff0dfe4
- NRDSWR\_Y 0xff0dfe8
- NRDSWR\_CTRL 0xff0dfec
- NRDSWR\_CAN\_SIZE 0xff0dff0
- CONTRD\_BADDR 0xff0dca4
- CONT2RD\_BADDR 0xff0dca8
- MTNRD\_BADDR 0xff0dcac
- MCVECRD\_BADDR 0xff0dcb0
- MCINFRD\_BADDR 0xff0dcb4
- CONTWR\_BADDR 0xff0dcd0
- CONTWR\_STRIDE 0xff0dcd4
- MTNWR\_BADDR 0xff0dcd8
- MTNWR\_STRIDE 0xff0dcdc
- MCVECWR\_BADDR 0xff0dcb8
- MCVECWR\_STRIDE 0xff0dcbc
- MCINFWR\_BADDR 0xff0df38
- MCINFWR\_STRIDE 0xff0df3c
- NRDSWR\_BADDR 0xff0dff4
- NRDSWR\_STRIDE 0xff0dff8

## Register Description

**Table 9-502 CONTRD\_CTRL1 0x37d0**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_sw_rst : unsigned, default = 0
29:26	R/W	0	Reg_gclk_ctrl[3:0]
25:24	R/W	0	reg_sync_sel : unsigned, default = 0
23:16	R/W	0	reg_canvas_id : unsigned, default = 0
14:12	R/W	1	reg_cmd_intr_len : unsigned, default = 1
11:10	R/W	1	reg_cmd_req_size : unsigned, default = 1
9:8	R/W	2	reg_burst_len : unsigned, default = 2
7	R/W	0	reg_swap_64bit : unsigned, default = 0
6	R/W	0	reg_little_endian : unsigned, default = 0
5	R/W	0	reg_y_rev : unsigned, default = 0
4	R/W	0	reg_x_rev : unsigned, default = 0

Bit(s)	R/W	Default	Description
3	R/W	1	Cntl acc mode (linear ddr mode)
2:0	R/W	1	reg_pack_mode : unsigned, default = 1

**Table 9-503 CONTRD\_CTRL2 0x37d1**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_gclk_ctrl[5:4] : unsigned, default = 0
23	R/W	0	urgent
22	R/W	0	auto_urgent_en
21	R/W	0	urgent_wr
20:17	R/W	0	up_th
16:13	R/W	0	dn_th
12:0	R/W	1920	Stride (128 bits)

**Table 9-504 CONTRD\_SCOPE\_X 0x37d2**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_x_end : unsigned, default = 0
12:0	R/W	0	reg_x_start : unsigned, default = 0

**Table 9-505 CONTRD\_SCOPE\_Y 0x37d3**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_y_end : unsigned, default = 0
12:0	R/W	0	reg_y_start : unsigned, default = 0

**Table 9-506 CONTRD\_RO\_STAT 0x37d4**

Bit(s)	R/W	Default	Description
15:0	R.O	0	ro_reg_status : unsigned, default = 0

**Table 9-507 CONT2RD\_CTRL1 0x37d5**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_sw_rst : unsigned, default = 0
29:26	R/W	0	Reg_gclk_ctrl[3:0]
25:24	R/W	0	reg_sync_sel : unsigned, default = 0
23:16	R/W	0	reg_canvas_id : unsigned, default = 0
14:12	R/W	1	reg_cmd_intr_len : unsigned, default = 1

Bit(s)	R/W	Default	Description
11:10	R/W	1	reg_cmd_req_size : unsigned, default = 1
9:8	R/W	2	reg_burst_len : unsigned, default = 2
7	R/W	0	reg_swap_64bit : unsigned, default = 0
6	R/W	0	reg_little_endian : unsigned, default = 0
5	R/W	0	reg_y_rev : unsigned, default = 0
4	R/W	0	reg_x_rev : unsigned, default = 0
3	R/W	1	Cntl acc mode (linear ddr mode)
2:0	R/W	1	reg_pack_mode : unsigned, default = 1

Table 9-508 CONT2RD\_CTRL2 0x37d6

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_gclk_ctrl[5:4] : unsigned, default = 0
23	R/W	0	urgent
22	R/W	0	auto_urgent_en
21	R/W	0	urgent_wr
20:17	R/W	0	up_th
16:13	R/W	0	dn_th
12:0	R/W	1920	Stride (128 bits)

Table 9-509 CONT2RD\_SCOPE\_X 0x37d7

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_x_end : unsigned, default = 0
12:0	R/W	0	reg_x_start : unsigned, default = 0

Table 9-510 CONT2RD\_SCOPE\_Y 0x37d8

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_y_end : unsigned, default = 0
12:0	R/W	0	reg_y_start : unsigned, default = 0

Table 9-511 CONT2RD\_RO\_STAT 0x37d9

Bit(s)	R/W	Default	Description
15:0	R.O	0	ro_reg_status : unsigned, default = 0

**Table 9-512 MTNRD\_CTRL1 0x37da**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_sw_rst : unsigned, default = 0
29:26	R/W	0	Reg_gclk_ctrl[3:0]
25:24	R/W	0	reg_sync_sel : unsigned, default = 0
23:16	R/W	0	reg_canvas_id : unsigned, default = 0
14:12	R/W	1	reg_cmd_intr_len : unsigned, default = 1
11:10	R/W	1	reg_cmd_req_size : unsigned, default = 1
9:8	R/W	2	reg_burst_len : unsigned, default = 2
7	R/W	0	reg_swap_64bit : unsigned, default = 0
6	R/W	0	reg_little_endian : unsigned, default = 0
5	R/W	0	reg_y_rev : unsigned, default = 0
4	R/W	0	reg_x_rev : unsigned, default = 0
3	R/W	1	Cntl acc mode (linear ddr mode)
2:0	R/W	1	reg_pack_mode : unsigned, default = 1

**Table 9-513 MTNRD\_CTRL2 0x37db**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_gclk_ctrl[5:4] : unsigned, default = 0
23	R/W	0	urgent
22	R/W	0	auto_urgent_en
21	R/W	0	urgent_wr
20:17	R/W	0	up_th
16:13	R/W	0	dn_th
12:0	R/W	1920	Stride (128 bits)

**Table 9-514 MTNRD\_SCOPE\_X 0x37dc**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_x_end : unsigned, default = 0
12:0	R/W	0	reg_x_start : unsigned, default = 0

**Table 9-515 MTNRD\_SCOPE\_Y 0x37dd**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_y_end : unsigned, default = 0
12:0	R/W	0	reg_y_start : unsigned, default = 0

**Table 9-516 MTNRD\_RO\_STAT 0x37de**

Bit(s)	R/W	Default	Description
15:0	R.O	0	ro_reg_status : unsigned, default = 0

**Table 9-517 MCVECRD\_CTRL1 0x37df**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_sw_rst : unsigned, default = 0
29:26	R/W	0	Reg_gclk_ctrl[3:0]
25:24	R/W	0	reg_sync_sel : unsigned, default = 0
23:16	R/W	0	reg_canvas_id : unsigned, default = 0
14:12	R/W	1	reg_cmd_intr_len : unsigned, default = 1
11:10	R/W	1	reg_cmd_req_size : unsigned, default = 1
9:8	R/W	2	reg_burst_len : unsigned, default = 2
7	R/W	0	reg_swap_64bit : unsigned, default = 0
6	R/W	0	reg_little_endian : unsigned, default = 0
5	R/W	0	reg_y_rev : unsigned, default = 0
4	R/W	0	reg_x_rev : unsigned, default = 0
3	R/W	1	Cntl acc mode (linear ddr mode)
2:0	R/W	1	reg_pack_mode : unsigned, default = 1

**Table 9-518 MCVECRD\_CTRL2 0x37e0**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_gclk_ctrl[5:4] : unsigned, default = 0
23	R/W	0	urgent
22	R/W	0	auto_urgent_en
21	R/W	0	urgent_wr
20:17	R/W	0	up_th
16:13	R/W	0	dn_th
12:0	R/W	1920	Stride (128 bits)

**Table 9-519 MCVECRD\_SCOPE\_X 0x37e1**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_x_end : unsigned, default = 0
12:0	R/W	0	reg_x_start : unsigned, default = 0

**Table 9-520 MCVECRD\_SCOPE\_Y 0x37e2**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_y_end : unsigned, default = 0
12:0	R/W	0	reg_y_start : unsigned, default = 0

**Table 9-521 MCVECRD\_RO\_STAT 0x37e3**

Bit(s)	R/W	Default	Description
15:0	R.O	0	ro_reg_status : unsigned, default = 0

**Table 9-522 MCINFRD\_CTRL1 0x37e4**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_sw_rst : unsigned, default = 0
29:26	R/W	0	Reg_gclk_ctrl[3:0]
25:24	R/W	0	reg_sync_sel : unsigned, default = 0
23:16	R/W	0	reg_canvas_id : unsigned, default = 0
14:12	R/W	1	reg_cmd_intr_len : unsigned, default = 1
11:10	R/W	1	reg_cmd_req_size : unsigned, default = 1
9:8	R/W	2	reg_burst_len : unsigned, default = 2
7	R/W	0	reg_swap_64bit : unsigned, default = 0
6	R/W	0	reg_little_endian : unsigned, default = 0
5	R/W	0	reg_y_rev : unsigned, default = 0
4	R/W	0	reg_x_rev : unsigned, default = 0
3	R/W	1	Cntl acc mode (linear ddr mode)
2:0	R/W	1	reg_pack_mode : unsigned, default = 1

**Table 9-523 MCINFRD\_CTRL2 0x37e5**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_gclk_ctrl[5:4] : unsigned, default = 0
23	R/W	0	urgent
22	R/W	0	auto_urgent_en
21	R/W	0	urgent_wr
20:17	R/W	0	up_th
16:13	R/W	0	dn_th
12:0	R/W	1920	Stride (128 bits)



**Table 9-524 MCINFRD\_SCOPE\_X 0x37e6**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_x_end : unsigned, default = 0
12:0	R/W	0	reg_x_start : unsigned, default = 0

**Table 9-525 MCINFRD\_SCOPE\_Y 0x37e7**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_y_end : unsigned, default = 0
12:0	R/W	0	reg_y_start : unsigned, default = 0

**Table 9-526 MCINFRD\_RO\_STAT 0x37e8**

Bit(s)	R/W	Default	Description
15:0	R.O	0	ro_reg_status : unsigned, default = 0

**Table 9-527 CONTWR\_X 0x37e9**

Bit(s)	R/W	Default	Description
31:30	R/W	2	burst_len : unsigned, default = 2
29	R/W	0	rev_x : unsigned, default = 0
28:16	R/W	0	start_x : unsigned, default = 0
12:0	R/W	2	end_x : unsigned, default = 2cf

**Table 9-528 CONTWR\_Y 0x37ea**

Bit(s)	R/W	Default	Description
31:30	R/W	0	canvas_id : unsigned, default = 0
29	R/W	0	rev_y : unsigned, default = 0
28:16	R/W	0	start_y : unsigned, default = 0
12:0	R/W	0	end_y : unsigned, default = 0x1df

**Table 9-529 CONTWR\_CTRL 0x37eb**

Bit(s)	R/W	Default	Description
31:16	R/W	0	urgent_ctrl : unsigned, default = 0
15	R/W	0	force_wvalid : unsigned, default = 0
14	R/W	0	canvas_syncen : unsigned, default = 0
13	R/W	1	canvas_wr : unsigned, default = 1
12	R/W	0	req_en : unsigned, default = 0
10	R/W	0	clr_wrrsp : unsigned, default = 0

Bit(s)	R/W	Default	Description
8	R/W	0	urgent : unsigned, default = 0
7:0	R/W	0	canvas_index : unsigned, default = 0

**Table 9-530 CONTWR\_CAN\_SIZE 0x37ec**

Bit(s)	R/W	Default	Description
30:29	R/W	0	reg_rst : unsigned, default = 0
28:16	R/W	0	hsizem1 : unsigned, default = 0x2cf
14	R/W	0	reg_reset : unsigned, default = 0
13	R/W	0	little_endian : unsigned, default = 0
12:0	R/W	0	vsizem1 : unsigned, default = 0x1df

**Table 9-531 MTNWR\_X 0x37ed**

Bit(s)	R/W	Default	Description
31:30	R/W	2	burst_len : unsigned, default = 2
29	R/W	0	rev_x : unsigned, default = 0
28:16	R/W	0	start_x : unsigned, default = 0
12:0	R/W	2	end_x : unsigned, default = 2cf

**Table 9-532 MTNWR\_Y 0x37ee**

Bit(s)	R/W	Default	Description
31:30	R/W	0	canvas_id : unsigned, default = 0
29	R/W	0	rev_y : unsigned, default = 0
28:16	R/W	0	start_y : unsigned, default = 0
12:0	R/W	0	end_y : unsigned, default = 0x1df

**Table 9-533 MTNWR\_CTRL 0x37ef**

Bit(s)	R/W	Default	Description
31:16	R/W	0	urgent_ctrl : unsigned, default = 0
15	R/W	0	force_wvalid : unsigned, default = 0
14	R/W	0	canvas_syncen : unsigned, default = 0
13	R/W	1	canvas_wr : unsigned, default = 1
12	R/W	0	req_en : unsigned, default = 0
10	R/W	0	clr_wrrsp : unsigned, default = 0
8	R/W	0	urgent : unsigned, default = 0
7:0	R/W	0	canvas_index : unsigned, default = 0

**Table 9-534 MTNWR\_CAN\_SIZE 0x37f0**

Bit(s)	R/W	Default	Description
30:29	R/W	0	reg_rst : unsigned, default = 0
28:16	R/W	0	hsizem1 : unsigned, default = 0x2cf
14	R/W	0	reg_reset : unsigned, default = 0
13	R/W	0	little_endian : unsigned, default = 0
12:0	R/W	0	vsizem1 : unsigned, default = 0x1df

**Table 9-535 MCVECWR\_X 0x37f1**

Bit(s)	R/W	Default	Description
31:30	R/W	2	burst_len : unsigned, default = 2
29	R/W	0	rev_x : unsigned, default = 0
28:16	R/W	0	start_x : unsigned, default = 0
12:0	R/W	2	end_x : unsigned, default = 2cf

**Table 9-536 MCVECWR\_Y 0x37f2**

Bit(s)	R/W	Default	Description
31:30	R/W	0	canvas_id : unsigned, default = 0
29	R/W	0	rev_y : unsigned, default = 0
28:16	R/W	0	start_y : unsigned, default = 0
12:0	R/W	0	end_y : unsigned, default = 0x1df

**Table 9-537 MCVECWR\_CTRL 0x37f3**

Bit(s)	R/W	Default	Description
31:16	R/W	0	urgent_ctrl : unsigned, default = 0
15	R/W	0	force_wvalid : unsigned, default = 0
14	R/W	0	canvas_syncen : unsigned, default = 0
13	R/W	1	canvas_wr : unsigned, default = 1
12	R/W	0	req_en : unsigned, default = 0
10	R/W	0	clr_wrrsp : unsigned, default = 0
8	R/W	0	urgent : unsigned, default = 0
7:0	R/W	0	canvas_index : unsigned, default = 0

**Table 9-538 MCVECWR\_CAN\_SIZE 0x37f4**

Bit(s)	R/W	Default	Description
30:29	R/W	0	reg_rst : unsigned, default = 0
28:16	R/W	0	hsizem1 : unsigned, default = 0x2cf
14	R/W	0	reg_reset : unsigned, default = 0
13	R/W	0	little_endian : unsigned, default = 0
12:0	R/W	0	vsizem1 : unsigned, default = 0x1df

**Table 9-539 MCINFWR\_X 0x37f5**

Bit(s)	R/W	Default	Description
31:30	R/W	2	burst_len : unsigned, default = 2
29	R/W	0	rev_x : unsigned, default = 0
28:16	R/W	0	start_x : unsigned, default = 0
12:0	R/W	2	end_x : unsigned, default = 2cf

**Table 9-540 MCINFWR\_Y 0x37f6**

Bit(s)	R/W	Default	Description
31:30	R/W	0	canvas_id : unsigned, default = 0
29	R/W	0	rev_y : unsigned, default = 0
28:16	R/W	0	start_y : unsigned, default = 0
12:0	R/W	0	end_y : unsigned, default = 0x1df

**Table 9-541 MCINFWR\_CTRL 0x37f7**

Bit(s)	R/W	Default	Description
31:16	R/W	0	urgent_ctrl : unsigned, default = 0
15	R/W	0	force_wvalid : unsigned, default = 0
14	R/W	0	canvas_syncen : unsigned, default = 0
13	R/W	1	canvas_wr : unsigned, default = 1
12	R/W	0	req_en : unsigned, default = 0
10	R/W	0	clr_wrrsp : unsigned, default = 0
8	R/W	0	urgent : unsigned, default = 0
7:0	R/W	0	canvas_index : unsigned, default = 0

**Table 9-542 MCINFWR\_CAN\_SIZE 0x37f8**

Bit(s)	R/W	Default	Description
30:29	R/W	0	reg_rst : unsigned, default = 0
28:16	R/W	0	hsizem1 : unsigned, default = 0x2cf
14	R/W	0	reg_reset : unsigned, default = 0
13	R/W	0	little_endian : unsigned, default = 0
12:0	R/W	0	vsizem1 : unsigned, default = 0x1df

**Table 9-543 CONTRD\_BADDR 0x3729**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Base address (ddr physical address >>4)

**Table 9-544 CONT2RD\_BADDR 0x372a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Base address (ddr physical address >>4)

**Table 9-545 MTNRD\_BADDR 0x372b**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Base address (ddr physical address >>4)

**Table 9-546 MCVECRD\_BADDR 0x372c**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Base address (ddr physical address >>4)

**Table 9-547 CONTWR\_BADDR 0x3734**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Base address (ddr physical address >>4)

**Table 9-548 CONTWR\_STRIDE 0x3735**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Stride ( 128 bits )

**Table 9-549 MTNWR\_BADDR 0x3736**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Base address (ddr physical address >>4)

**Table 9-550 MTNWR\_STRIDE 0x3737**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Stride ( 128 bits )

**Table 9-551 MCVECWR\_BADDR 0x372e**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Base address (ddr physical address >>4)

**Table 9-552 MCVECWR\_STRIDE 0x372f**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Stride ( 128 bits )

**Table 9-553 MCINFWR\_BADDR 0x37ce**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Base address (ddr physical address >>4)

**Table 9-554 MCINFWR\_STRIDE 0x37cf**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Stride ( 128 bits )

**Table 9-555 NRDSWR\_BADDR 0x37fd**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Base address (ddr physical address >>4)

**Table 9-556 NRDSWR\_STRIDE 0x37fe**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Stride ( 128 bits )

### 9.2.3.7 De-Interlace mad Registers

#### Register Address

- DI\_TOP\_PRE\_CTRL 0xff005f14
- DI\_TOP\_POST\_CTRL 0xff005f18
- DI\_PRE\_GL\_CTRL 0xff005f1c
- DI\_PRE\_GL\_THD 0xff005f20
- DI\_POST\_GL\_CTRL 0xff005f24

- DI\_POST\_GL\_THD 0xff005f28
- DI\_RO\_PRE\_DBG 0xff005f2c
- DI\_RO\_POST\_DBG 0xff005f30
- DI\_TOP\_CTRL 0xff005f34
- DI\_AFBCD\_GCLK0 0xff005f38
- DI\_AFBCD\_GCLK1 0xff005f3c
- DI\_RDMIF\_DEPTH0 0xff005f40
- DI\_RDMIF\_DEPTH1 0xff005f44
- DI\_RDMIF\_DEPTH2 0xff005f48
- DI\_TOP\_CTRL1 0xff005f4c
- DI\_AFBCE0\_HOLD\_CTRL 0xff005f50
- DI\_AFBCE1\_HOLD\_CTRL 0xff005f54
- DI\_PRE\_CTRL 0xff005c00
- DI\_POST\_CTRL 0xff005c04
- DI\_POST\_SIZE 0xff005c08
- DI\_PRE\_SIZE 0xff005c0c
- DI\_EI\_CTRL0 0xff005c10
- DI\_EI\_CTRL1 0xff005c14
- DI\_EI\_CTRL2 0xff005c18
- DI\_NR\_CTRL1 0xff005c20
- DI\_NR\_CTRL2 0xff005c24
- DI\_MTN\_CTRL1 0xff005c30
- DI\_BLEND\_CTRL 0xff005c34
- DI\_ARB\_CTRL 0xff005c3c
- DI\_BLEND\_REG0\_X 0xff005c40
- DI\_EI\_CTRL4 0xff005c68
- DI\_EI\_CTRL5 0xff005c6c
- DI\_EI\_CTRL6 0xff005c70
- DI\_EI\_CTRL7 0xff005c74
- DI\_EI\_CTRL8 0xff005c78
- DI\_EI\_CTRL9 0xff005c7c
- DI\_MC\_REG0\_X 0xff005c80
- DI\_MC\_32LVL0 0xff005ca8
- DI\_MC\_32LVL1 0xff005cac
- DI\_MC\_22LVL0 0xff005cb0
- DI\_MC\_22LVL1 0xff005cb4
- DI\_MC\_22LVL2 0xff005cb8
- DI\_MC\_CTRL 0xff005cbc
- DI\_INTR\_CTRL 0xff005cc0
- DI\_INFO\_ADDR 0xff005cc4
- DI\_INFO\_DATA 0xff005cc8
- DI\_PRE\_HOLD 0xff005ccc
- DI\_MTN\_1\_CTRL1 0xff005d00
- DI\_MTN\_1\_CTRL2 0xff005d04

- DI\_MTN\_1\_CTRL3 0xff005d08
- DI\_MTN\_1\_CTRL4 0xff005d0c
- DI\_MTN\_1\_CTRL5 0xff005d10
- DI\_MTN\_1\_CTRL6 0xff005ea4
- DI\_MTN\_1\_CTRL7 0xff005ea8
- DI\_MTN\_1\_CTRL8 0xff005eac
- DI\_MTN\_1\_CTRL9 0xff005eb0
- DI\_MTN\_1\_CTRL10 0xff005eb4
- DI\_MTN\_1\_CTRL11 0xff005eb8
- DI\_MTN\_1\_CTRL12 0xff005ebc
- DI\_RO\_CRC\_NRWR 0xff005f00
- DI\_RO\_CRC\_MTNWR 0xff005f04
- DI\_RO\_CRC\_DEINT 0xff005f08
- DI\_CRC\_CHK0 0xff005f0c
- DI\_RO\_CRC\_NRWR 0xff005f00
- DI\_RO\_CRC\_MTNWR 0xff005f04
- DI\_RO\_CRC\_DEINT 0xff005f08

## Register Description

**Table 9-557 DI\_TOP\_PRE\_CTRL 0x17c5**

Bit(s)	R/W	Default	Description
31:30	W/R	0	Pre_frm_sel : 0: frame reset by register, 0x1700 bit31 1: pre-post link ( if post_frm_sel==0, pre-post-viu link ) 2: pre viu link, use viu vsync as frame reset 3: pre vdin link, use vdin vsync as frame reset
29:27	W/R	0	Rdmi line buffer bypass: bit0: 1:di inp mif line buffer bypass Bit1: 1:di chan2 mif line buffer bypass Bit2: 1:di mem mif line buffer bypass
26:24	W/R	0	Afbcd0_frm_st_sel: no use
23:22	W/R	0	bit23:afbce done flag clear: clear di afbce write done flag, pulse bit22:Wrmif1 done flag clear: clear di wrmif write done flag, pulse
21:20	W/R	0	Wr_mif fix disable pre
19	W/R	0	Pre_bypass_en : 1:pre process bypass enable (example: bit19:12=0x81, di inp direct to nr write)
18:16			Pre_bypass_mode: 0: nr write select byp_din0 1: nr write select byp_din1 2: nr write select byp_din2
14:12			Pre_bypass_sel: Bit0: 1: byp_din0 select di inp Bit1: 1: byp_din1 select di chan2 Bit2: 1: byp_din2 select di mem
11	W/R	0	Reg_afbce0_awugt: 1:nr write afbce urgent enable
10	W/R	0	Nr ch0 en : 1: nr write enable(afbce or normal mif) 0: nr write disable
1:0	W/R	0	Nrwr_path_sel: bit0: normal mif path enable bit1: afbc enc enable



Table 9-558 DI\_TOP\_POST\_CTRL 0x17c6

Bit(s)	R/W	Default	Description
31:30	W/R	0	Post_frm_sel : 0: post viu link, use viu vsync as frame reset 1: frame reset by register , 0x1701 bit31 2: pre-post link, use post frame reset as pre-post total reset (default use pre-frame reset)
29:27	W/R	0	Rdmi line buffer bypass: Bit0: 1:di if1 mif line buffer bypass Bit1: 1:di if0 mif line buffer bypass Bit2: 1:di if2 mif line buffer bypass
26:24	W/R	0	Afbcd1_frm_st_sel: no use
23:22	W/R	0	bit23:afbce done flag clear: clear di write done flag,pulse bit22:Wrmif1 done flag clear: clear di write done flag,pulse
21:20	W/R	0	Wr_mif Fix disable post
19	W/R	0	Post_bypass_en : 1:post process bypass enable (example: bit19:12=0x81, di if0 direct to di write)
18:16			Post_bypass_mode: 0: di write select byp_din0 1: di write select byp_din1 2: di write select byp_din2
14:12			Post_bypass_sel: Bit0: 1: byp_din0 select di if0 Bit1: 1: byp_din1 select di if1 Bit2: 1: byp_din2 select di if2
11	W/R	0	Reg_afbce1_awugt: 1:di write afbce urgent enable
10	W/R	0	Wrmif_afbce_rdugt: afbce read urgent enable
1:0	W/R	0	diwr_path_sel: bit0: normal mif path enable bit1: afbc enc enable

Table 9-559 DI\_TOP\_CTRL 0x17cd

Bit(s)	R/W	Default	Description
21:4	W/R	{5,4,3,2,-1,0}	Afbcd 6mux6_sel: reorder di read mif (afbcd or normal mif)
0	W/R		Vpp path sel : 0: di post output to VPP_VD1 1: di pre output to VPP_VD1

Table 9-560 DI\_TOP\_CTRL1 0x17d3

Bit(s)	R/W	Default	Description
31:28	W/R		Afbc enc gate clock control bit31:30: afbce1 gating control bit29:28: afbce0 gating control
27	W/R		Wrmif shrink switch 0:shrink in di path 1.shrink in nr path

Bit(s)	R/W	Default	Description
26:22	W/R		Soft reset: Bit4: wrmif arb reset Bit3 : nr afbc enc reset Bit2 : nr wr mif reset Bit1 : di afbc enc reset Bit0 : di wr mif reset
3	W/R		Wrmif ram switch : 1 :di can write 4k afbce 0: nr can write 4k afbce
2	W/R		Wrmif nr 4k: 1:nr/di write 4K size afbc 0:nr 1080p + di 1080 afbce
0	W/R		Wrmif_path_switch: 0: no switch 1: nr write and di write path switch□only can switch in initial state

Table 9-561 DI\_AFBCE0\_HOLD\_CTRL 0x17d4

Bit(s)	R/W	Default	Description
31	W/R	0	Hold enable
23:12	W/R	0	Hold count
11:0	W/R	0	Pass count

Table 9-562 DI\_AFBCE1\_HOLD\_CTRL 0x17d5

Bit(s)	R/W	Default	Description
31	W/R	0	Hold enable
23:12	W/R	0	Hold count
11:0	W/R	0	Pass count

Table 9-563 DI\_RDMIF\_DEPTH0 0x17d0

Bit(s)	R/W	Default	Description
27:16	W/R	512	Di chan2 rdmif line buffer depth
11:0	W/R	512	Di inp rdmif line buffer depth

Table 9-564 DI\_RDMIF\_DEPTH1 0x17d1

Bit(s)	R/W	Default	Description
27:16	W/R	512	Di if1 rdmif line buffer depth
11:0	W/R	512	Di mem rdmif line buffer depth

**Table 9-565 DI\_RDMIF\_DEPTH2 0x17d2**

Bit(s)	R/W	Default	Description
27:16	W/R	512	Di if2 rdmif line buffer depth
11:0	W/R	512	Di if0 rdmif line buffer depth

**Table 9-566 DI\_PRE\_GL\_CTRL 0x20ab**

Bit(s)	R/W	Default	Description
31	W/R	0	pre count enable
30	W/R	0	pre count reset
29:16	W/R	0x20	total line number for pre count
15	W	0	pre mif manual start
13:0	W/R	0xc	the line number of pre frame reset

**Table 9-567 DI\_PRE\_GL\_THD 0x20ac**

Bit(s)	R/W	Default	Description
21:16	W/R	10	DI PRE hold line number
15:0	W/R	1920	H total pixel number for pre count

**Table 9-568 DI\_POST\_GL\_CTRL 0x20ad**

Bit(s)	R/W	Default	Description
31	W/R	0	post count enable
30	W/R	0	post count reset
29:16	W/R	0x20	total line number for post count
15	W	0	post mif manual start
13:0	W/R	0xc	the line number of post frame reset

**Table 9-569 DI\_POST\_GL\_THD 0x20ae**

Bit(s)	R/W	Default	Description
21:16	W/R	10	DI POST hold line number
15:0	W/R	1920	H total pixel number for post count

**Table 9-570 DI\_PRE\_CTRL 0x1700**

Bit(s)	R/W	Default	Description
31	W/R		cbus_pre_frame_rst
30	W/R		cbus_pre_soft_rst
29	W/R		pre_field_num

Bit(s)	R/W	Default	Description
27:26	W/R		mode_444c422
25	W/R		di_cont_read_en
24:23	W/R		mode_422c444
21	W/R		pre field num for nr
20	W/R		pre field num for pulldown
19	W/R		pre field num for mcdi
17	W/R		reg_me_autoen
16	W/R		reg_me_en
12:11	W/R	0x2	Reg_tfbf_en : bit0:tfbf enable bit1:normal path enable important!!!
10	W/R		nr_wr_by
9	W/R		di_buf2_en: chan2 line buffer enable , P1 field
8	W/R		Di buf3 en: mem line buffer enable, P2 field
7	W/R		Pd_mtn_swap
6	W/R		mtn after nr
5	W/R		hist_check_en
4	W/R		check_after_nr
3	W/R		check222p_en
2	W/R		check322p_en
1	W/R		mtn_en
0	W/R		nr_en

Table 9-571 DI\_POST\_CTRL 0x1701

Bit(s)	R/W	Default	Description
31	W/R		cbus_post_frame_rst
30	W/R		cbus_post_soft_rst
29	W/R		post_field_num
14	W/R		Reg_link_nxt_sel: pre-post link 0: current line 1:next line
13:12	W/R		Reg_link_cur_sel: pre-post link 0: current line 1:next line
11	W/R		di_post_repeat
10	W/R		di_post_drop_1st
6	W/R		Reg_post_mb_en
5	W/R		di_vpp_out_en
4	W/R		di_wr_bk_en
3	W/R		di_mux_en

Bit(s)	R/W	Default	Description
2	W/R		di_ei_en
1	W/R		Di_blend_en
0	W/R		di_post_en : 1:di post process enable 0:di post disable

**Table 9-572 DI\_POST\_SIZE 0x1702**

Bit(s)	R/W	Default	Description
31:29	W/R	0	diwr_field_mode
28:16	W/R	0	vsize1post
12:0	W/R	0	hsize1post

**Table 9-573 DI\_PRE\_SIZE 0x1703**

Bit(s)	R/W	Default	Description
31:29	W/R	0	nrwr_field_mode
28:16	W/R	0	vsize1pre
12:0	W/R	0	hsize1pre

**Table 9-574 DI\_TOP\_CTRL 0x17cd**

Bit(s)	R/W	Default	Description
21:4	W/R	{5,4,3,2,-1,0}	Afbc 6mux6_sel: reorder di read mif (afbc or normal mif)
0	W/R		Vpp path sel : 0: di post output to VPP_VD1 1: di pre output to VPP_VD1

**Table 9-575 DI\_TOP\_CTRL1 0x17d3**

Bit(s)	R/W	Default	Description
31:28	W/R		Afbc enc gate clock control
27	W/R		Wrmif shrink switch
26:22	W/R		Soft reset: bit4: wrmif arb reset Bit3 : nr afbc enc reset Bit2 : nr wr mif reset Bit1 : di afbc enc reset Bit0 : di wr mif reset
3	W/R		Wrmif ram switch : 1 di write 4k 0: nr write 4k
2	W/R		Wrmif nr 4k: 1:nr write 4K size
0	W/R		Wrmif_path_switch: 0: no switch 1: nr write and di write switch

**Table 9-576 DI\_AFBCE0\_HOLD\_CTRL 0x17d4**

Bit(s)	R/W	Default	Description
31	W/R	0	Hold enable
23:12	W/R	0	Hold count
11:0	W/R	0	Pass count

**Table 9-577 DI\_AFBCE1\_HOLD\_CTRL 0x17d5**

Bit(s)	R/W	Default	Description
31	W/R	0	Hold enable
23:12	W/R	0	Hold count
11:0	W/R	0	Pass count

**Table 9-578 DI\_RDMIF\_DEPTH0 0x17d0**

Bit(s)	R/W	Default	Description
27:16	W/R	512	Di chan2 rdmif line buffer depth
11:0	W/R	512	Di inp rdmif line buffer depth

**Table 9-579 DI\_RDMIF\_DEPTH1 0x17d1**

Bit(s)	R/W	Default	Description
27:16	W/R	512	Di if1 rdmif line buffer depth
11:0	W/R	512	Di mem rdmif line buffer depth

**Table 9-580 DI\_RDMIF\_DEPTH2 0x17d2**

Bit(s)	R/W	Default	Description
27:16	W/R	512	Di if2 rdmif line buffer depth
11:0	W/R	512	Di if0 rdmif line buffer depth

**Table 9-581 DI\_EI\_CTRL0 0x1704**

Bit(s)	R/W	Default	Description
23:16	W/R		ei0_filter[2:+] abs_diff_left>filter && ...right>filter && ...top>filter && ...bot>filter -> filter
15:8	W/R		ei0_threshold[2:+]
3	W/R		ei0_vertical
2	W/R		ei0_bpscf2
1	W/R		ei0_bpsfar1

**Table 9-582 DI\_EI\_CTRL1 0x1705**

Bit(s)	R/W	Default	Description
31:24	W/R		ei0_diff
23:16	W/R		ei0_angle45
15:8	W/R		ei0_peak
7:0	W/R		ei0_cross

**Table 9-583 DI\_EI\_CTRL2 0x1706**

Bit(s)	R/W	Default	Description
31:24	W/R		ei0_close2
23:16	W/R		ei0_close1
15:8	W/R		ei0_far2
7:0	W/R		ei0_far1

**Table 9-584 DI\_NR\_CTRL0 0x1707**

Bit(s)	R/W	Default	Description
26	W/R		nr_cue_en
25	W/R		nr2_en

**Table 9-585 DI\_NR\_CTRL1 0x1708**

Bit(s)	R/W	Default	Description
31:30	W/R		mot_p1txtcore_mode
29:24	W/R		mot_p1txtcore_clmt
21:16	W/R		mot_p1txtcore_ylmt
15:8	W/R		mot_p1txtcore_crate
7:0	W/R		mot_p1txtcore_yrate

**Table 9-586 DI\_NR\_CTRL2 0x1709**

Bit(s)	R/W	Default	Description
31-30	W/R	0	text_chk_mode ( tm2_revb : DI_MTN_CTRL 0x170b, bit21:20 )
29-24	W/R	0	mot_curtxtcore_clmt
21-16	W/R	0	mot_curtxtcore_ylmt
15-8	W/R	0	mot_curtxtcore_crate
7-0	W/R	0	mot_curtxtcore_yrate

**Table 9-587 DI\_CANVAS\_URGENT0 0x170a**

Bit(s)	R/W	Default	Description
26	R/W	0	di write mif bvalid_sel: 1. Bvalid_signal from bus, 0: bytes_wr handshakes
25	R/W	0	di write mif burst last sel: 1. All kind of burst last signal include ext_data_last. 0. Used the normal burst last signal
24:16	W/R	0	Di write mif urgent ctrl
9	R/W	0	nr write mif bvalid_sel: 1. Bvalid_signal from bus, 0: bytes_wr handshakes
8	R/W	0	nr write mif burst last sel: 1. All kind of burst last signal include ext_data_last. 0. Used the normal burst last signal
7:0	W/R	0	Nr write mif urgent ctrl

**Table 9-588 DI\_MTN\_CTRL 0x170b**

Bit(s)	R/W	Default	Description
31	W/R	0	reg_mtn_1_en if 0x170b[0]=1 (DI_MTN_CTRL)
30	W/R	0	reg_mtn_init if 0x170b[0]=1 (DI_MTN_CTRL)
29	W/R	0	reg_di2nr_txt_en if 0x170b[0]=1 (DI_MTN_CTRL)
28	W/R	0	reg_di2nr_txt_mode if 0x170b[0]=1 (DI_MTN_CTRL)
27:24	W/R	0	reg_mtn_def if 0x170b[0]=1 (DI_MTN_CTRL)
23: 0	W/R	0	reserved

**Table 9-589 DI\_MTN\_CTRL1 0x170c**

Bit(s)	R/W	Default	Description
11:8	W/R		mtn_paramnthd
7:0	W/R		mtn_parafitthd

**Table 9-590 DI\_BLEND\_CTRL 0x170d**

Bit(s)	R/W	Default	Description
31	W/R		blend_1_en
30	W/R		blend_mtn_lpf
28	W/R		post_mb_en
21:20	W/R		blend_top_mode 00: mtn, 01: weave mode, 10: bob mode, 11 : blend mode
19	W/R		blend_reg3_enable
18	W/R		blend_reg2_enable
17	W/R		blend_reg1_enable
16	W/R		blend_reg0_enable
15:14	W/R		blend_reg3_mode
13:12	W/R		blend_reg2_mode



Bit(s)	R/W	Default	Description
11:10	W/R		blend_reg1_mode
9:8	W/R		blend_reg0_mode

**Table 9-591 DI\_ARB\_CTRL 0x170f**

Bit(s)	R/W	Default	Description
31:26	W/R	0x20	Di_arb_thd1
25:20	W/R	0x20	Di_arb_thd0
19	W/R	0	Di_arb_tid_mode
18	W/R	0	Di_arb_arb_mode
17	W/R	0	Di_arb_acg_en
16	W/R	0	Di_arb_disable_clk
15:0	W/R	0	Di_arb_req_en

**Table 9-592 DI\_BLEND\_REG0\_X 0x1710**

Bit(s)	R/W	Default	Description
28:16	W/R		blend_reg0_startx
12:0	W/R		blend_reg0_endx

**Table 9-593 DI\_EI\_CTRL4 0x171a**

Bit(s)	R/W	Default	Description
29	W/R	0	reg_ei_caldr_t_amblike2_biasvertical
28:24	W/R	21	reg_ei_caldr_t_addxla2list_drtmax
23	W/R	0	N/A
22:20	W/R	1	reg_ei_caldr_t_addxla2list_signm0th
19	W/R	1	reg_ei_caldr_t_addxla2list_mode
18:16	W/R	3	reg_ei_signm_sad_cor_rate
15:12	W/R	3	reg_ei_signm_sadi_cor_rate
11:6	W/R	2	reg_ei_signm_sadi_cor_ofst
5:0	W/R	4	reg_ei_signm_sad_ofst

**Table 9-594 DI\_EI\_CTRL5 0x171b**

Bit(s)	R/W	Default	Description
30:28	W/R	5	reg_ei_caldr_t_cnflctchk_frverthrd
27	W/R	0	N/A
26:24	W/R	2	reg_ei_caldr_t_cnflctchk_mg

Bit(s)	R/W	Default	Description
23:22	W/R	1	reg_ei_caldr_t_cnflctchk_ws
21	W/R	1	reg_ei_caldr_t_cnflctchk_en
20	W/R	1	reg_ei_caldr_t_verfr_c_final_en
19	W/R	0	reg_ei_caldr_t_verfr_c_retimflt_en
18:16	W/R	3	reg_ei_caldr_t_verfr_c_eithratemth
15	W/R	0	reg_ei_caldr_t_verfr_c_retiming_en
14:12	W/R	2	reg_ei_caldr_t_verfr_c_bothratemth
11:9	W/R	0	reg_ei_caldr_t_ver_thr_d
8:4	W/R	4	reg_ei_caldr_t_addxla2list_drtmin
3:0	W/R	15	reg_ei_caldr_t_addxla2list_drtlimit

Table 9-595 DI\_EI\_CTRL6 0x171c

Bit(s)	R/W	Default	Description
31:24	W/R	80	reg_ei_caldr_t_abext_sad12thhig
23:16	W/R	35	reg_ei_caldr_t_abext_sad00thlow
15:8	W/R	28	reg_ei_caldr_t_abext_sad12thlow
6:4	W/R	1	reg_ei_caldr_t_abext_ratemth
2:0	W/R	5	reg_ei_caldr_t_abext_drtthrd

Table 9-596 DI\_EI\_CTRL7 0x171d

Bit(s)	R/W	Default	Description
29	W/R	1	reg_ei_caldr_t_xlanopeak_codien
28:24	W/R	15	reg_ei_caldr_t_xlanopeak_drtmax
23	W/R	1	reg_ei_caldr_t_xlanopeak_en
22:20	W/R	3	reg_ei_caldr_t_abext_monotrnd_alpha
19:18	W/R	1	reg_ei_caldr_t_abext_mononum12_thr_d
17:16	W/R	1	reg_ei_caldr_t_abext_mononum00_thr_d
15:12	W/R	6	reg_ei_caldr_t_abext_sad00rate
11:8	W/R	6	reg_ei_caldr_t_abext_sad12rate
7:0	W/R	80	reg_ei_caldr_t_abext_sad00thhig

Table 9-597 DI\_EI\_CTRL8 0x171e

Bit(s)	R/W	Default	Description
30:28	W/R	2	reg_ei_assign_headtail_magin
26:24	W/R	3	reg_ei_retime_lastcurpncnftchk_mode

Bit(s)	R/W	Default	Description
22:21	W/R	0	reg_ei_retime_lastcurpncnftchk_drth
13:11	W/R	3	reg_ei_caldr_t_amblike2_drting
10:8	W/R	1	reg_ei_caldr_t_amblike2_valmg
7:4	W/R	10	reg_ei_caldr_t_amblike2_alpha
3:0	W/R	4	reg_ei_caldr_t_amblike2_drth

Table 9-598 DI\_EI\_CTRL9 0x171f

Bit(s)	R/W	Default	Description
31:28	W/R	7	reg_ei_caldr_t_hcnfcheck_frcvert_xla_th3
27	W/R	1	reg_ei_caldr_t_hcnfcheck_frcvert_xla_en
26:24	W/R	4	reg_ei_caldr_t_conf_drth
23:20	W/R	11	reg_ei_caldr_t_conf_absdrth
19:18	W/R	2	reg_ei_caldr_t_abcheck_mode1
17:16	W/R	1	reg_ei_caldr_t_abcheck_mode0
15:12	W/R	11	reg_ei_caldr_t_abcheck_drth1
11:8	W/R	11	reg_ei_caldr_t_abcheck_drth0
6:4	W/R	3	reg_ei_caldr_t_abpnchk1_th
1	W/R	1	reg_ei_caldr_t_abpnchk1_en
0	W/R	1	reg_ei_caldr_t_abpnchk0_en

Table 9-599 DI\_EI\_CTRL10 0x1793

Bit(s)	R/W	Default	Description
31:28	W/R	0	reg_ei_caldr_t_hstrgchk_drth
27:24	W/R	8	reg_ei_caldr_t_hstrgchk_frcverthr
23:20	W/R	4	reg_ei_caldr_t_hstrgchk_mg
19	W/R	0	reg_ei_caldr_t_hstrgchk_1sidnul
18	W/R	0	reg_ei_caldr_t_hstrgchk_excpcnf
17:16	W/R	2	reg_ei_caldr_t_hstrgchk_ws
15	W/R	1	reg_ei_caldr_t_hstrgchk_en
14:13	W/R	2	reg_ei_caldr_t_hpncheck_mode
12	W/R	0	reg_ei_caldr_t_hpncheck_mute
11:9	W/R	3	reg_ei_caldr_t_hcnfcheck_mg2
8:6	W/R	2	reg_ei_caldr_t_hcnfcheck_mg1
5:4	W/R	2	reg_ei_caldr_t_hcnfcheck_mode
3:0	W/R	9	reg_ei_caldr_t_hcnfcheck_frcvert_xla_th5

**Table 9-600 DI\_EI\_CTRL11 0x179e**

Bit(s)	R/W	Default	Description
30:29	W/R	2	reg_ei_amb_detect_mode
28:24	W/R	8	reg_ei_amb_detect_winth
23:21	W/R	3	reg_ei_amb_decide_rppth
20:19	W/R	1	reg_ei_retime_lastmappncnftchk_drth
18:16	W/R	2	reg_ei_retime_lastmappncnftchk_mode
15:14	W/R	2	reg_ei_retime_lastmapvertfcchk_mode
13:12	W/R	3	reg_ei_retime_lastvertfcchk_mode
11:8	W/R	0	reg_ei_retime_lastpnchk_drth
6	W/R	1	reg_ei_retime_lastpnchk_en
5:4	W/R	3	reg_ei_retime_mode
3	W/R	1	reg_ei_retime_last_en
2	W/R		reg_ei_retime_ab_en
1	W/R	1	reg_ei_caldrt_hstrvertfcchk_en
0	W/R	0	reg_ei_caldrt_hstrrgchk_mode

**Table 9-601 DI\_EI\_CTRL12 0x179f**

Bit(s)	R/W	Default	Description
31:28	W/R	13	reg_ei_drtdelay2_lmt
27:26	W/R	2	reg_ei_drtdelay2_notver_lrwin
25:24	W/R	3	reg_ei_drtdelay_mode
23	W/R	0	reg_ei_drtdelay2_mode
22:20	W/R	0	reg_ei_assign_xla_signm0th
19	W/R	1	reg_ei_assign_pkbiasvert_en
18	W/R	1	reg_ei_assign_xla_en
17:16	W/R	0	reg_ei_assign_xla_mode
15:12	W/R	2	reg_ei_assign_nlfilter_magin
11:8	W/R	5	reg_ei_localsearch_maxrange
7:4	W/R	0	reg_ei_xla_drth
3:0	W/R	3	reg_ei_flatmsad_thrd

**Table 9-602 DI\_EI\_CTRL13 0x17a8**

Bit(s)	R/W	Default	Description
27:24	W/R	15	reg_ei_int_drt2x_chrdrt_limit
23:20	W/R	0	reg_ei_int_drt16x_core
19:16	W/R	2	reg_ei_int_drtdelay2_notver_cancv
15:8	W/R	20	reg_ei_int_drtdelay2_notver_sadth
7:0	W/R	20	reg_ei_int_drtdelay2_vlddrt_sadth

**Table 9-603 DI\_EI\_DRT\_CTRL 0x2028**

Bit(s)	R/W	Default	Description
31	W/R	0	reg_rectg_en: Low angle enable.
30	W/R	0	reg_recbld_en: New and old drt blend enable.
29:28	W/R	2	reg_rectg_ws : window sideto calculate the reference direction: 0:1x1; 1:1x3; 2:1x5; 3:1x7
27			reserved
26:24	W/R	2	reg_abq_margin : top and bottom curve trend quantization margin of noise for direction assignments.
23			reserved
22:20	W/R	3	reg_trend_mg : the Margin of the top/bot trend.
19:16	W/R	1	reg_int_d16xc1 : Coring to drtf.
15:14			reserved
13:8	W/R	40	reg_int_chlmt1: Limit to drtf(16x) for chroma angle
7			reserved
6:4	W/R	5	reg_nscheck_thr:check whether the pixels id noise or not.
3			reserved
2:0	W/R	7	reg_horsl_ws: window side to check the existent number of low angle drt, if the number<the value, drt=raw drt.

**Table 9-604 DI\_EI\_DRT\_PIXTH 0x2029**

Bit(s)	R/W	Default	Description
31□24	W/R	22	reg_min_pix: the threshold of min pix of photos, <threshold the pix do not participate in the monotonic trend calculation.
23:16	W/R	203	reg_max_pix:the threshold of max pix of photos, >threshold the pix do not participate in the monotonic trend calculation.
15:8	W/R	50	reg_dmaxmin_thr: the max pixel and min pixel difference is larger than the value the trend existent.
7:0	W/R	30	reg_dmaxmin_thr:mi: the max pixel and min pixel difference is smaller than the value the trend non-existent.

**Table 9-605 DI\_EI\_DRT\_CORRPIXTH 0x202a**

Bit(s)	R/W	Default	Description
31□24	W/R	40	reg_newcorrpix_maxthrd: the new low angle drt sad threshold.
23:16	W/R	60	reg_corrpix_diffthrd: the top and bottom pixel difference is larger than the value, the case may be ultra-low angle.
15:8	W/R	10	reg_corrpix_minthrd: the difference of top and bottom pixel is smaller than the value, the drt may be raw drt.
7:0	W/R	20	reg_corrpix_maxthrd: the difference of top and bottom pixel is larger than the value, the drt may be raw drt.

**Table 9-606 DI\_EI\_DRT\_RECTG\_WAVE 0x202b**

Bit(s)	R/W	Default	Description
31:29			reserved
28:24	W/R	0	reg_max_pixwave: the wave of the max pix threshold, prevent min pix close to reg_max_pix caused the number between max and min pix zeros.
23:21			reserved
20:16	W/R	15	reg_pix_wave: the wave of the max and min pix, the max pixel smaller than the value or the min pixel larger than the value, may be the ultra-low angle case.
15:14			reserved
13:8	W/R	40	reg_maxdrt_thrd: the threshold of the low angle max drt.
7:0	W/R	20	reg_wave_thrd:in bilateral cases tow pixel difference is smaller than the value, the trend between the tow pixel not change.

**Table 9-607 DI\_EI\_DRT\_PIX\_DIFFTH 0x202c**

Bit(s)	R/W	Default	Description
31□24	W/R	32	reg_newraw_thrd: the old drt and new drt transition threshold.
23:16	W/R	10	reg_tb_max_thrd: the threshold of top and bottom max or min pixel.
15:8	W/R	20	reg_diffpix_thrd: Max-Min<the value, the trend is non-existent.
7:6			reserved
5:0	W/R	5	reg_bilt_trendnumt: in bilateral cases the difference between the top and bottom pixel number of the monotonic trend smaller than the value is low angle.

**Table 9-608 DI\_EI\_DRT\_UNBITREND\_TH 0x202d**

Bit(s)	R/W	Default	Description
31:29			reserved
28:24	W/R	10	reg_trend_numt:in bilateral cases the pixel number of the monotonic trend larger than the value is low angle.
23:21			reserved
20:16	W/R	4	reg_bilt_trendnumt:in bilateral cases the pixel number of the trend larger than the value is low angle.

Bit(s)	R/W	Default	Description
15:13			reserved
12:8	W/R	7	reg_unil_trendnumt: in unilateral cases the difference between the top and bottom pixel number of the monotonic trend smaller than the value is low angle.
7:5			reserved
4:0	W/R	10	reg_trend_num: in unilateral cases the pixel number of the trend larger than the value is ultra-low angle.

**Table 9-609 DI\_EI\_XWIN0 0x1798**

Bit(s)	R/W	Default	Description
27:16	W/R		ei_xend0
11:0	W/R		ei_xstart0

**Table 9-610 DI\_MC\_REG0\_X 0x1720**

Bit(s)	R/W	Default	Description
27:16	W/R		mc_reg0_start_x
11:0	W/R		mc_reg0_end_x

**Table 9-611 DI\_MC\_32LVL0 0x172a**

Bit(s)	R/W	Default	Description
31:24	W/R		mc_reg2_32lvl
23:16	W/R		mc_reg1_32lvl
15:8	W/R		mc_reg0_32lvl
7:0	W/R		field_32lvl

**Table 9-612 DI\_MC\_32LVL1 0x172b**

Bit(s)	R/W	Default	Description
15:8	W/R		mc_reg3_32lvl
7:0	W/R		mc_reg4_32lvl

**Table 9-613 DI\_MC\_22LVL0 0x172c**

Bit(s)	R/W	Default	Description
31:16	W/R		mc_reg0_22lvl
15:0	W/R		field_22lvl

**Table 9-614 DI\_MC\_22LVL1 0x172d**

Bit(s)	R/W	Default	Description
31:16	W/R		mc_reg2_22lvl
15:0	W/R		mc_reg1_22lvl

**Table 9-615 DI\_MC\_22LVL2 0x172e**

Bit(s)	R/W	Default	Description
31:16	W/R		mc_reg4_22lvl
15:0	W/R		mc_reg3_22lvl

**Table 9-616 DI\_MC\_CTRL 0x172f**

Bit(s)	R/W	Default	Description
4	W/R		mc_reg4_en
3	W/R		mc_reg3_en
2	W/R		mc_reg2_en
1	W/R		mc_reg1_en
0	W/R		mc_reg0_en

**Table 9-617 DI\_INTR\_CTRL 0x1730**

Bit(s)	R/W	Default	Description
31	W/R	0	Deint_irq_mode==0: the interrupt flag will be erected when predi_int or postdi_int is generated Deint_irq_mode==1: when the interrupt source of any module in DI is generated without mask operation, the interrupt flag will be erected.
30:26	W/R	0	reserved
25	w/R	0	NrDownscale_int_mask
24	w/R	0	Det3d_int_mask
23	w/R	0	Mcinfowr_int_mask
22	w/R	0	Mcvecwr_int_mask
21	w/R	0	Medi_int_mask
20	w/R	0	Contwr_int_mask
19	w/R	0	Hist_int_mask
18	w/R	0	Diwr_int_mask
17	w/R	0	Mtn_wr_int_mask
16	w/R	0	Nrwr_int_mask
15:10	W/R	0	reserved
9	R		Nrdownscale_done



Bit(s)	R/W	Default	Description
8	R		Det3d_done
7	R		Mcinfowr_done (not valid in GX)
6	R		Mcvecwr_done (not valid in GX)
5	R		Medi_done(not valid in GX)
4	R		Contwr_done
3	R		Hist_done
2	R		diwr_done
1	R		Mtnwr_done
0	R		Nrwr_done

DI\_INFO\_ADDR 0x1731

**Table 9-618 Addr\_0**

Bit(s)	R/W	Default	Description
31:0	R		Field_32p , sum of difference between n-2 and n

**Table 9-619 Addr\_1**

Bit(s)	R/W	Default	Description
31:24	R		Field_32max, maximum difference between n-2 and n
23:0	R		Field_32num, numbers of pixels difference > threshold

**Table 9-620 Addr\_2**

Bit(s)	R/W	Default	Description
31:0	R		Field_22p, sum of difference between temporal and vertical difference

**Table 9-621 Addr\_3**

Bit(s)	R/W	Default	Description
15:0	R		Field_22max , maximum difference between temporal and verticaldifference

**Table 9-622 Addr\_4**

Bit(s)	R/W	Default	Description
23:0	R		Field_22num, pixel sum which difference > threshold

**Table 9-623 Addr\_5**

Bit(s)	R/W	Default	Description
31:0	R		Luma sum

**Table 9-624 Addr\_6**

Bit(s)	R/W	Default	Description
31:0	R		Difference of 32, sum in area 0

**Table 9-625 Addr\_7**

Bit(s)	R/W	Default	Description
31:0	R		Difference of 32, sum in area 1

**Table 9-626 Addr\_8**

Bit(s)	R/W	Default	Description
31:0	R		Difference of 32, sum in area 2

**Table 9-627 Addr\_9**

Bit(s)	R/W	Default	Description
31:0	R		Difference of 32, sum in area 3

**Table 9-628 Addr\_10**

Bit(s)	R/W	Default	Description
31:0	R		Difference of 32, sum in area 4

**Table 9-629 Addr\_11**

Bit(s)	R/W	Default	Description
31:0	R		Difference of 22, sum in area 0

**Table 9-630 Addr\_12**

Bit(s)	R/W	Default	Description
31:0	R		Difference of 22, sum in area 1

**Table 9-631 Addr\_13**

Bit(s)	R/W	Default	Description
31:0	R		Difference of 22, sum in area 2

**Table 9-632 Addr\_14**

Bit(s)	R/W	Default	Description
31:0	R		Difference of 22, sum in area 3

**Table 9-633 Addr\_15**

Bit(s)	R/W	Default	Description
31:0	R		Difference of 22, sum in area 4

**Table 9-634 Addr\_16**

Bit(s)	R/W	Default	Description
31:0	R		luma, sum in area 0

**Table 9-635 Addr\_17**

Bit(s)	R/W	Default	Description
31:0	R		luma, sum in area 1

**Table 9-636 Addr\_18**

Bit(s)	R/W	Default	Description
31:0	R		luma, sum in area 2

**Table 9-637 Addr\_19**

Bit(s)	R/W	Default	Description
31:0	R		luma, sum in area 3

**Table 9-638 Addr\_20**

Bit(s)	R/W	Default	Description
31:0	R		luma, sum in area 4

**Table 9-639 Addr\_21**

Bit(s)	R/W	Default	Description
31:24	R		Field_32max, maximum difference between n-2 and n in area0
23:0	R		Field_32num, numbers of pixels difference > threshold in area0

**Table 9-640 Addr\_22**

Bit(s)	R/W	Default	Description
31:24	R		Field_32max, maximum difference between n-2 and n in area1
23:0	R		Field_32num, numbers of pixels difference > threshold in area1

**Table 9-641 Addr\_23**

Bit(s)	R/W	Default	Description
31:24	R		Field_32max, maximum difference between n-2 and n in area2
23:0	R		Field_32num, numbers of pixels difference > threshold in area2

**Table 9-642 Addr\_24**

Bit(s)	R/W	Default	Description
31:24	R		Field_32max, maximum difference between n-2 and n in area3
23:0	R		Field_32num, numbers of pixels difference > threshold in area3

**Table 9-643 Addr\_25**

Bit(s)	R/W	Default	Description
31:24	R		Field_32max, maximum difference between n-2 and n in area4
23:0	R		Field_32num, numbers of pixels difference > threshold in area4

**Table 9-644 Addr\_26**

Bit(s)	R/W	Default	Description
31:20	R		Field_22max/16, in area 0
19:0	R		Field_22 num/16, in area 0

**Table 9-645 Addr\_27**

Bit(s)	R/W	Default	Description
31:20	R		Field_22max/16, in area 1
19:0	R		Field_22 num/16, in area 1

**Table 9-646 Addr\_28**

Bit(s)	R/W	Default	Description
31:20	R		Field_22max/16, in area 2
19:0	R		Field_22 num/16, in area 2

**Table 9-647 Addr\_29**

Bit(s)	R/W	Default	Description
31:20	R		Field_22max/16, in area 3
19:0	R		Field_22 num/16, in area 3

**Table 9-648 Addr\_30**

Bit(s)	R/W	Default	Description
31:20	R		Field_22max/16, in area 4
19:0	R		Field_22 num/16, in area 4

**Table 9-649 DI\_PRE\_HOLD 0x1733**

Bit(s)	R/W	Default	Description
31	R/W	0	cntl_pre_hold_enable
27:16	R/W	0	cntl_pre_hold_count
11:0	R/W	0	cntl_pre_pass_count

**Table 9-650 DI\_MTN\_1\_CTRL1 0x1740**

Bit(s)	R/W	Default	Description
31-30	W/R	0	text_chk_mode ( tm2_revb : DI_MTN_CTRL 0x170b, bit21:20 )
29-24	W/R	0	mot_curtxtcore_clmt
21-16	W/R	0	mot_curtxtcore_ylmt
15-8	W/R	0	mot_curtxtcore_crate
7-0	W/R	0	mot_curtxtcore_yrate

**Table 9-651 DI\_MTN\_1\_CTRL2 0x1741**

Bit(s)	R/W	Default	Description
31:24	W/R	26	reg_DI_m1b_core_Ykinter
23:16	W/R	26	reg_DI_m1b_core_Ckinter
15:8	W/R	58	reg_DI_m1b_core_Ykintra
7:0	W/R	98	reg_DI_m1b_core_Ckintra

**Table 9-652 DI\_MTN\_1\_CTRL3 0x1742**

Bit(s)	R/W	Default	Description
31:24	W/R	21	reg_DI_m1b_thr_2Yrate
23:16	W/R	32	reg_DI_m1b_thr_2Crate

Bit(s)	R/W	Default	Description
15: 8	W/R	10	reg_DI_m1b_core_mxcmbY
7: 0	W/R	10	reg_DI_m1b_core_mxcmbC

**Table 9-653 DI\_MTN\_1\_CTRL4 0x1743**

Bit(s)	R/W	Default	Description
31:24	W/R	1	reg_DI_m1b_coreY
23:16	W/R	0	reg_DI_m1b_coreC
15: 8	W/R	8	reg_DI_m1b_thrd_min
7: 0	W/R	128	reg_DI_m1b_thrd_max

**Table 9-654 DI\_MTN\_1\_CTRL5 0x1744**

Bit(s)	R/W	Default	Description
31:27	W/R	7	reg_DI_m1b_pp_extnd_num
27:24	W/R	4	reg_DI_m1b_pp_errord_num
21:20	W/R	0	Re_di2nr_txt_mode 0:average of top/bot 1:max; 2:  a+c-2b ;
15: 8	W/R	13	reg_DI_mot_core_Ykinter
7: 0	W/R	13	reg_DI_mot_core_Ckinter

**Table 9-655 DI\_MTN\_1\_CTRL6 0x17a9**

Bit(s)	R/W	Default	Description
31:24	W/R	13	reg_DI_mot_core_Ykintra
23:16	W/R	90	reg_DI_mot_core_Ckintra
15: 8	W/R	21	reg_DI_mot_cor_2Yrate
7: 0	W/R	32	reg_DI_mot_cor_2Crate

**Table 9-656 DI\_MTN\_1\_CTRL7 0x17aa**

Bit(s)	R/W	Default	Description
31:24	W/R	10	reg_DI_mot_core_mxcmbY
23:16	W/R	10	reg_DI_mot_core_mxcmbC
15: 8	W/R	2	reg_DI_mot_coreY
7: 0	W/R	1	reg_DI_mot_coreC

**Table 9-657 DI\_MTN\_1\_CTRL8 0x17ab**

Bit(s)	R/W	Default	Description
31:24	W/R	26	reg_DI_fmot_core_Ykinter
23:16	W/R	26	reg_DI_fmot_core_Ckinter
15: 8	W/R	38	reg_DI_fmot_core_Ykintra
7: 0	W/R	98	reg_DI_fmot_core_Ckintra

**Table 9-658 DI\_MTN\_1\_CTRL9 0x17ac**

Bit(s)	R/W	Default	Description
31:24	W/R	13	reg_DI_fmot_cor_2Yrate
23:16	W/R	32	reg_DI_fmot_cor_2Crate
15: 8	W/R	3	reg_DI_fmot_coreY
7: 0	W/R	2	reg_DI_fmot_coreC

**Table 9-659 DI\_MTN\_1\_CTRL10 0x17ad**

Bit(s)	R/W	Default	Description
27:24	W/R	2	reg_DI_m1b_suremot_num_fld0
19:16	W/R	2	reg_DI_m1b_surestl_num_fld0
11: 8	W/R	6	reg_DI_m1b_suremot_num_fld1
3: 0	W/R	6	reg_DI_m1b_surestl_num_fld1

**Table 9-660 DI\_MTN\_1\_CTRL11 0x17ae**

Bit(s)	R/W	Default	Description
27:24	W/R	5	reg_DI_m1b_suremot_evn_th
20:16	W/R	8	reg_DI_m1b_suremot_odd_th
11: 8	W/R	3	reg_DI_m1b_surestl_evn_th
6	W/R	0	reg_DI_m1b_suremot_fast_en
5	W/R	0	reg_DI_m1b_surestl_fast_en
4: 0	W/R	4	reg_DI_m1b_surestl_odd_th

**Table 9-661 DI\_MTN\_1\_CTRL12 0x17af**

Bit(s)	R/W	Default	Description
31:24	W/R	64	reg_DI_mot_norm_gain
17:16	W/R	2	reg_DI_mot_alpha_lpf
15: 8	W/R	10	reg_DI_m1b_surestl_thrd
4: 0	W/R	4	reg_DI_mot_surestl_gain

**Table 9-662 DI\_CRC\_CHK0 0x17c3**

Bit(s)	R/W	Default	Description
31	W	0	di pre crc check start (pulse)
30	W	0	Di post crc check start (pulse)
29	W	0	motion crc check start (pulse)
2	R/W	0	MTN output crc check enable
1	R/W	0	DI output crc check enable
0	R/W	0	Nr output crc check enable

**Table 9-663 DI\_RO\_CRC\_NRWR 0x17c0**

Bit(s)	R/W	Default	Description
31-0	W	0	nrwr crc output (lock by pre frame rst)

**Table 9-664 DI\_RO\_CRC\_MTNWR 0x17c2**

Bit(s)	R/W	Default	Description
31-0	W	0	mtnwr crc output (lock by pre frame rst)

**Table 9-665 DI\_RO\_CRC\_DEINT 0x17c1**

Bit(s)	R/W	Default	Description
31-0	W	0	di post crc output (lock by pre frame rst)

### 9.2.3.8 VIU AFBCD\_VDMIF\_FGAIN Registers

**Table 9-666 Base Address Summary**

Module	Sub module	Base address
VD1 read mif	rdmif	0x4800
	afbc_dec	0x4840
	film grain	0x4870
VD2 read mif	rdmif	0x4880
	afbc_dec	0x48c0
	film grain	0x48f0
DI read mif 0	rdmif	0x5400
	afbc_dec	0x5440



Module	Sub module	Base address
	film grain	0x5470
DI read mif 1	rdmif	0x5480
	afbc_dec	0x54c0
	film grain	0x54f0
DI read mif 2	rdmif	0x5500
	afbc_dec	0x5540
	film grain	0x5570
DI read mif 3	rdmif	0x5580
	afbc_dec	0x55c0
	film grain	0x55f0
DI read mif 4	rdmif	0x5600
	afbc_dec	0x5640
	film grain	0x5670
DI read mif 5	rdmif	0x5680
	afbc_dec	0x56c0
	film grain	0x56f0

## VD1 Path vd\_rmem\_if0 Registers

Table 9-667 VD1\_IF0\_GEN\_REG 0x4800

Bit(s)	R/W	Default	Description
31	R/W	0	ENABLE_FREE_CLK. 0: Gated clock for power saving 1: Free-running clock to drive logic
30	R/W	0	SW_RESET: Write 1 to this bit to generate a pulse to reset everything except registers.
29	R/W	0	RESET_ON_GO_FIELD: Define whether to reset state machines on go_field pulse. 0: No reset on go_field 1: go_field reset everything except registers
28	R/W	0	URGENT_CHROMA: Set urgent level for chroma fifo request from DDR. 0: Non urgent 1: Urgent
27	R/W	0	URGENT_LUMA: Set urgent level for luma fifo request from DDR. 0: Non urgent 1: Urgent
26	R/W	0	Chroma_end_at_last_line: For chroma line, similar to luma_end_at_last_line, as below. Not used if data are stored together in one canvas.
25	R/W	0	Luma_end_at_last_line: Control whether continue outputting luma line past last line. 0: Repeat the last line or dummy pixels, after past the last line 1: Stop outputting data, once past the last line.

Bit(s)	R/W	Default	Description
24-19	R/W	4	Hold_lines: After go_field, the number of lines to hold before the module is enabled.
18	R/W	0	LAST_LINE: This bit controls whether we simply repeat the last line or we push dummy pixels. '1' tells the state-machines to repeat the last line using the dummy pixels defined in the register below. '0' indicates that the state-machine should re-read the last line of real data.
17	R	0	Busy status of the state-machines. '1' = busy, '0' = idle
16	R/W	0	DEMUX_MODE: 0 = 4:2:2, 1 = RGB (24-bit). This value is used to control the demuxing logic when the picture is stored together. When a picture is stored together, the data is read into a single FIFO (the Y FIFO) and must be demultiplexed into the "drain" outputs. In the case of 4:2:2 the data is assumed to be stored in memory in 16-bit chunks: <YCb><YCr><YCb><YCr>,... the Y, Cb and Cr 8-bit values are pulled from the single Y-FIFO and sent out in pairs.  This value is only valid when the picture is stored together. If the picture is separated into different canvases, then this bit field is ignored.
15-14	R/W	0	BYTES_PER_PIXEL: This value is used to determine how many bytes are associated with each pixel. 0: This value should be used if the image is stored separately (e.g. RGB or Y, Cb, Cr). 1: This value should be used if the data is 4:2:2 data stored together. In this case each pixel, YCb or YCr, is 16-bits (two bytes). 2: This value should be used if the RGB (24-bit) data is stored together. 3: reserved for future use (alpha RGB).
13-12	R/W	0	DDR_BURST_SIZE_CR: This value is used to control the DDR burst request size for the Cr FIFO. 0: Maximum burst = 24 64-bit values 1: Maximum burst = 32 64-bit values 2: Maximum burst = 48 64-bit values 3: Maximum burst = 64 64-bit values
11-10	R/W	0	DDR_BURST_SIZE_CB: This value is used to control the DDR burst request size for the Cb FIFO. 0: Maximum burst = 24 64-bit values 1: Maximum burst = 32 64-bit values 2: Maximum burst = 48 64-bit values 3: Maximum burst = 64 64-bit values
9-8	R/W	0	DDR_BURST_SIZE_Y: This value is used to control the DDR burst request size for the Y FIFO. 0: Maximum burst = 24 64-bit values 1: Maximum burst = 32 64-bit values 2: Maximum burst = 48 64-bit values 3: Maximum burst = 64 64-bit values
7	R/W	0	MANUAL_START_FRAME: non-latching bit that can be used to simulate the go_field signal for simulation.
6	R/W	0	CHRO_RPT_LASTL_CTRL: This bit controls whether to allow VPP's chroma-repeat request. 0: Chroma-repeat pulses from VPP are ignored 1: Chroma-repeat pulses from VPP are used.
5	R/W	0	Unused
4	R/W	0	LITTLE_ENDIAN: This bit defines the endianness of the memory data . 0: Pixel data are stored big-endian in memory 1: Pixel data are stored little-endian in memory
3	R/W	0	Chroma_hz_avg: For chroma line output control, similar to luma_hz_avg, as below. Not used if data are stored together in one canvas.
2	R/W	0	Luma_hz_avg: Enable output half amount of data per line to save bandwidth. 0: Output every pixel per line 1: Output half line, each data averaged between every 2 pixels

Bit(s)	R/W	Default	Description
			Note: For 4:2:2 mode data stored together in one canvas, only do averaging over luma data.
1	R/W	0	SEPARATE_EN: Set this bit to 1 if the image is in separate canvas locations.
0	R/W	0	ENABLE: This bit is set to 1 to enable the FIFOs and other logic. This bit can be set to 0 to cleanup and put the logic into an IDLE state.

Table 9-668 VD1\_IF0\_CANVAS0 – Picture 0 0x4801

Bit(s)	R/W	Default	Description
31-24	R/W	0	unused
23-16	R/W	0	CANVAS0_ADDR2: Canvas table address for picture 0 for component 2 (Cr FIFO). This value is ignored when the picture is stored together
15-8	R/W	0	CANVAS0_ADDR1: Canvas table address for picture 0 for component 1 (Cb FIFO). This value is ignored when the picture is stored together
7-0	R/W	0	CANVAS0_ADDR0: Canvas table address for picture 0 for component 0 (Y FIFO).

Table 9-669 VD1\_IF0\_CANVAS1 – Picture 1 0x4802

Bit(s)	R/W	Default	Description
31-24	R/W	0	unused
23-16	R/W	0	CANVAS1_ADDR2: Canvas table address for picture 1 for component 2 (Cr FIFO). This value is ignored when the picture is stored together
15-8	R/W	0	CANVAS1_ADDR1: Canvas table address for picture 1 for component 1 (Cb FIFO). This value is ignored when the picture is stored together
7-0	R/W	0	CANVAS1_ADDR0: Canvas table address for picture 1 for component 0 (Y FIFO).

Table 9-670 VD1\_IF0\_LUMA\_X0 – Picture 0 0x4803

Bit(s)	R/W	Default	Description
31	R/W	0	Unused
30-16	R/W	0	LUMA_X_END0: Picture 0, luma X end value
15	R/W	0	Unused
14-0	R/W	0	LUMA_X_START0: Picture 0, luma X start value

Table 9-671 VD1\_IF0\_LUMA\_Y0 – Picture 0 0x4804

Bit(s)	R/W	Default	Description
31-29	R/W	0	Unused
28-16	R/W	0	LUMA_Y_END0: Picture 0, luma Y end value
15-13	R/W	0	Unused
12-0	R/W	0	LUMA_Y_START0: Picture 0, luma Y start value

**Table 9-672 VD1\_IF0\_CHROMA\_X0 – Picture 0 0x4805**

Bit(s)	R/W	Default	Description
31	R/W	0	Unused
30-16	R/W	0	CHROMA_X_END0: Picture 0, chroma X end value. This value is only used when the picture is not stored together.
15	R/W	0	Unused
14-0	R/W	0	CHROMA_X_START0: Picture 0, chroma X start value. This value is only used when the picture is not stored together.

**Table 9-673 VD1\_IF0\_CHROMA\_Y0 – Picture 0 0x4806**

Bit(s)	R/W	Default	Description
31-29	R/W	0	Unused
28-16	R/W	0	CHROMA_Y_END0: Picture 0, chroma Y end value. This value is only used when the picture is not stored together.
15-13	R/W	0	Unused
12-0	R/W	0	CHROMA_Y_START0: Picture 0, chroma Y start value. This value is only used when the picture is not stored together.

**Table 9-674 VD1\_IF0\_LUMA\_X1 – Picture 1 0x4807**

Bit(s)	R/W	Default	Description
31	R/W	0	Unused
30-16	R/W	0	LUMA_X_END1: Picture 1, luma X end value
15	R/W	0	Unused
14-0	R/W	0	LUMA_X_START1: Picture 1, luma X start value

**Table 9-675 VD1\_IF0\_LUMA\_Y1 – Picture 1 0x4808**

Bit(s)	R/W	Default	Description
31-29	R/W	0	Unused
28-16	R/W	0	LUMA_Y_END1: Picture 1, luma Y end value
15-13	R/W	0	Unused
12-0	R/W	0	LUMA_Y_START1: Picture 1, luma Y start value

**Table 9-676 VD1\_IF0\_CHROMA\_X1 – Picture 1 0x4809**

Bit(s)	R/W	Default	Description
31	R/W	0	Unused
30-16	R/W	0	CHROMA_X_END1: Picture 1, chroma X end value. This value is only used when the picture is not stored together.

Bit(s)	R/W	Default	Description
15	R/W	0	Unused
14-0	R/W	0	CHROMA_X_START1: Picture 1, chroma X start value. This value is only used when the picture is not stored together.

Table 9-677 VD1\_IF0\_CHROMA\_Y1 – Picture 1 0x480A

Bit(s)	R/W	Default	Description
31-29	R/W	0	Unused
28-16	R/W	0	CHROMA_Y_END1: Picture 1, chroma Y end value. This value is only used when the picture is not stored together.
15-13	R/W	0	Unused
12-0	R/W	0	CHROMA_Y_START1: Picture 1, chroma Y start value. This value is only used when the picture is not stored together.

Table 9-678 VD1\_IF0\_REPEAT\_LOOP – Pictures 0 and 1 0x480B

Bit(s)	R/W	Default	Description
31-24	R/W	0	CHROMA_RPT_LOOP1: Repeat loop for Picture 1. Bits[6:4] = start loop pointer, bits [2:0] = end loop pointer. Bits [7] and [3] are ignored.
23-16	R/W	0	LUMA_RPT_LOOP1: Repeat loop for Picture 1. Bits[6:4] = start loop pointer, bits [2:0] = end loop pointer. Bits [7] and [3] are ignored.
15-8	R/W	0	CHROMA_RPT_LOOP0: Repeat loop for Picture 0. Bits[6:4] = start loop pointer, bits [2:0] = end loop pointer. Bits [7] and [3] are ignored.
7-0	R/W	0	LUMA_RPT_LOOP0: Repeat loop for Picture 0. Bits[6:4] = start loop pointer, bits [2:0] = end loop pointer. Bits [7] and [3] are ignored.

Table 9-679 VD1\_IF0\_LUMA0\_RPT\_PAT – Picture 0 LUMA repeat pattern 0x480C

Bit(s)	R/W	Default	Description
31-0	R/W	0	Luma repeat/skip pattern for picture 0

Bits	Pattern Index	Pattern description
31-28	7	Repeat/skip pattern: Bit[3] = 0 indicates repeat. Bit[3] = 1 indicates either skip, or output this line and then skip. How to interpret this bit depends on the value of the previous pattern's Bit[3]. If previous Bit[3]=0, then skip; If previous Bit[3]=1, then output this line and then skip. Bits[2:0] indicate the skip / repeat count. Blow is an example of consecutive patterns, the start line is line 0: {0010} Repeat this line (line 0) two more times for a total of three line reads. Proceed to next line (line 1). {0000} Don't repeat this line (line 1). This line will be read just once. Proceed to next line (line 2). {1000} Skip one line (line 2) to get to the next line (line 3). The skip implies that the next line (line 3) should be read at least once. {1011} Read this line (line 3) once, and then skip the next four lines to get to the next line (line 8). The skip implies that the next line (line 8) should be read at least once. {0100} Repeat this line (line 8) four more times for a total of five line read. Proceed to next line (line 9). {1001} Skip two lines to get to the next line (line 11). The skip implies that the next line (line 11) should be read at least once.
27-24	6	See pattern definition above.

Bits	Pattern Index	Pattern description
23-20	5	See pattern definition above.
19-16	4	See pattern definition above.
15-12	3	See pattern definition above.
11-8	2	See pattern definition above.
7-4	1	See pattern definition above.
3-0	0	See pattern definition above.

**Table 9-680 VD1\_IF0\_CHROMA0\_RPT\_PAT – Picture 0 CHROMA repeat pattern 0x480D**

Bit(s)	R/W	Default	Description
31-0	R/W	0	Chroma repeat/skip pattern for picture 0. See picture 0 luma pattern for description. This value is only used when the picture is not stored together.

**Table 9-681 VD1\_IF0\_LUMA1\_RPT\_PAT – Picture 1 LUMA repeat pattern 0x480E**

Bit(s)	R/W	Default	Description
31-0	R/W	0	Luma repeat/skip pattern for picture 1. See picture 0 luma pattern for description.

**Table 9-682 VD1\_IF0\_CHROMA1\_RPT\_PAT – Picture 1 CHROMA repeat pattern 0x480F**

Bit(s)	R/W	Default	Description
31-0	R/W	0	Chroma repeat/skip pattern for picture 1. See picture 0 luma pattern for description. This value is only used when the picture is not stored together.

**Table 9-683 VD1\_IF0\_LUMA\_PSEL – Picture 0 and 1's LUMA 0x4810**

Bit(s)	R/W	Default	Description
31-28	R/W	0	unused
27-26	R/W	0	Luma_psel_mode: controls whether it's single-picture or two-picture mode. {00} Only picture 0 is used. Ignore settings defined in Luma_psel_last_line, Luma_psel_pattern and Luma_psel_loop. {01} Only picture 1 is used. Ignore settings defined in Luma_psel_last_line, Luma_psel_pattern and Luma_psel_loop. {1x} Two-picture mode.
25-24	R/W	0	Luma_psel_last_line: select which picture's last line to output, during repeat last line mode. Bit[0]=0, when picture 0 past the last line, use picture 0's last line during repeat last line mode; Bit[0]=1, when picture 0 past the last line, use picture 1's last line during repeat last line mode; Bit[1]=0, when picture 1 past the last line, use picture 0's last line during repeat last line mode; Bit[1]=1, when picture 1 past the last line, use picture 1's last line during repeat last line mode.
23-8	R/W	0	Luma_psel_pattern. If the value of the bit pointed by the loop pointer is 0, output picture 0's luma line, if the bit value is 1, output picture 1's luma line.

Bit(s)	R/W	Default	Description
7-4	R/W	0	Luma_psel_loop start pointer.
3-0	R/W	0	Luma_psel_loop end pointer.

**Table 9-684 VD1\_IF0\_CHROMA\_PSEL – Picture 0 and 1's CHROMA 0x4811**

Bit(s)	R/W	Default	Description
31-28	R/W	0	unused
27-26	R/W	0	Chroma_psel_mode: see luma_psel_mode. This value is only used when the picture is not stored together.
25-24	R/W	0	Chroma_psel_last_line: See luma_psel_last_line. This value is only used when the picture is not stored together.
23-8	R/W	0	Chroma_psel_pattern. If the value of the bit pointed by the loop pointer is 0, output picture 0's chroma line, if the bit value is 1, output picture 1's chroma line. This value is only used when the picture is not stored together.
7-4	R/W	0	Chroma_psel_loop start pointer. This value is only used when the picture is not stored together.
3-0	R/W	0	Chroma_psel_loop end pointer. This value is only used when the picture is not stored together.

**Table 9-685 VD1\_IF0\_DUMMY\_PIXEL 0x4812**

Bit(s)	R/W	Default	Description
31-24	R/W	0x00	Y or R dummy pixel value ,8bit
23-16	R/W	0x80	Cb or G dummy pixel value ,8bit
15-8	R/W	0x80	Cr or B dummy pixel value ,8bit
7-0	R/W	0	unused

VD1\_RANGE\_MAP\_Y 0x481A

VD1\_RANGE\_MAP\_CB 0x481B

**Table 9-686 VD1\_RANGE\_MAP\_CR 0x481C**

Output data range conversion function:

$$Y[n] = \text{clip} ( \text{Round} ( ( Y[n] + \text{DIN\_OFFSET} ) * \text{RANGE\_MAP\_COEF} ) / ( 1 \ll \text{RANGE\_MAP\_SR} ) ) + \text{DOUT\_OFFSET} );$$

To perform VC-1 range reduction, set the following:

DIN\_OFFSET = 0x180 = -128;

RANGE\_MAP\_COEF = RANGE\_MAPY + 9

RANGE\_MAP\_SR = 3

DOUT\_OFFSET = 0x080 = 128

To get the equivalent function:

$$Y[n] = \text{clip}(((Y[n]-128) * (\text{RANGE\_MAPY} + 9) + 4) \gg 3) + 128);$$

Bit(s)	R/W	Default	Description
31-23	R/W	0	DIN_OFFSET
22-15	R/W	0	RANGE_MAP_COEF
14	R/W	0	unused
13-10	R/W	0	RANGE_MAP_SR
9-1	R/W	0	DOUT_OFFSET
0	R/W	0	RANGE_MAP_EN

Table 9-687 VD1\_IF0\_GEN\_REG2 0x481D

Bit(s)	R/W	Default	Description
31-2	R/W	0	unused
1-0	R/W	0	COLOR_MAP: Define color map for NV12 or NV21 mode. Only applicable when VD1_IF0_GEN_REG.SEPARATE_EN = 1. 0: NOT NV12 or NV21; 1: NV12 (CbCr); 2: NV21 (CrCb).

Table 9-688 VD1\_IF0\_GEN\_REG3 0x4816

Bit(s)	R/W	Default	Description
11-10	R/W	0	cntl_dbg_mode
9-8	R/W	0	cntl_bits_mode : 0->8bit 1->10bit 422 2->10bit 444
6-4	R/W	3	cntl_blk_len
2-1	R/W	1	cntl_burst_len
0	R/W	1	cntl_64bit_rev

Table 9-689 VIU\_VD1\_FMT\_CTRL 0x4818

Bit(s)	R/W	Default	Description
31	R/W	0	gate_clk_en. 0=No clock gating, free-running; 1=Enable clock gating for power saving.
30	R/W	0	soft_rst. If true, reset formatters.
29	R/W	0	unused
28	R/W	0	if true, horizontal formatter use repeating to generate pixel, otherwise use bilinear interpolation
27-24	R/W	0	horizontal formatter initial phase
23	R/W	0	horizontal formatter repeat pixel 0 enable
22-21	R/W	0	horizontal Y/C ratio, 00: 1:1, 01: 2:1, 10: 4:1
20	R/W	0	horizontal formatter enable
19	R/W	0	if true, always use phase0 while vertical formater, meaning always repeat data, no interpolation
18	R/W	0	if true, disable vertical formatter chroma repeat last line



Bit(s)	R/W	Default	Description
17	R/W	0	vertical formatter doesn't need repeat line on phase0, 1: enable, 0: disable
16	R/W	0	vertical formatter repeat line 0 enable
15-12	R/W	0	vertical formatter skip line num at the beginning
11-8	R/W	0	vertical formatter initial phase
7-1	R/W	0	vertical formatter phase step (3.4)
0	R/W	0	vertical formatter enable

**Table 9-690 VIU\_VD1\_FMT\_W 0x4819**

Bit(s)	R/W	Default	Description
27-16	R/W	0	horizontal formatter width
11-0	R/W	0	vertical formatter width

VD2 vd\_rmem\_if0 is same as vd1 vd\_rmem\_if0, BASE\_addr is from 0x4800 to 0x4880

DI vd\_rmem\_if0 is same as vd1 vd\_rmem\_if0, BASE\_addr is from 0x4800 to 0x5400-0x5680

## VPU AFBC Registers

**Table 9-691 AFBCDM\_ENABLE 0x4840**

Bit(s)	R/W	Default	Description
28:23	R/W	0	reg_gclk_ctrl_core : unsigned, default = 0 if true ,disable clock, otherwise enable clock
22	R/W	0	reg_fmt_size_sw_mode : unsigned, default = 0, 0:hw mode 1:sw mode for format size
21	R/W	1	reg_addr_link_en : unsigned, default = 1, 1:enable
20	R/W	0	reg_fmt444_comb : unsigned, default = 0, 0: 444 8bit uncomb
19	R/W	0	reg_dos_uncomp_mode : unsigned , default = 0
18:16	R/W	4	soft_rst : unsigned , default = 4
13:12	R/W	1	ddr_blk_size : unsigned , default = 1
11:9	R/W	3	cmd_blk_size : unsigend , default = 3
8	R/W	0	dec_enable : unsigned , default = 0
1	R/W	1	head_len_sel : unsigned , default = 1
0	R/W	0	dec_frm_start : unsigned , default = 0

**Table 9-692 AFBCDM\_MODE 0x4841**

Bit(s)	R/W	Default	Description
29	R/W	0	ddr_sz_mode : uns, default = 0 , 0: fixed block ddr size 1 : unfixed block ddr size;
28	R/W	0	blk_mem_mode : uns, default = 0 , 0: fixed 16x128 size; 1 : fixed 12x128 size
27:26	R/W	0	rev_mode : uns, default = 0 , reverse mode

Bit(s)	R/W	Default	Description
25:24	R/W	3	mif_urgent : uns, default = 3 , info mif and data mif urgent
22:16	R/W	0	hold_line_num : uns, default = 0 ,
15:14	R/W	2	burst_len : uns, default = 2, 0:burst1 1:burst2 2:burst4
13:8	R/W	0	compbits_yuv : uns, default = 0 , bit 1:0,: y component bitwidth : 00-8bit 01-9bit 10-10bit 11-12bit bit 3:2,: u component bitwidth : 00-8bit 01-9bit 10-10bit 11-12bit bit 5:4,: v component bitwidth : 00-8bit 01-9bit 10-10bit 11-12bit
7:6	R/W	0	vert_skip_y : uns, default = 0 , luma vert skip mode : 00-y0y1, 01-y0, 10-y1, 11-(y0+y1)/2
5:4	R/W	0	horz_skip_y : uns, default = 0 , luma horz skip mode : 00-y0y1, 01-y0, 10-y1, 11-(y0+y1)/2
3:2	R/W	0	vert_skip_uv : uns, default = 0 , chroma vert skip mode : 00-y0y1, 01-y0, 10-y1, 11-(y0+y1)/2
1:0	R/W	0	horz_skip_uv : uns, default = 0 , chroma horz skip mode : 00-y0y1, 01-y0, 10-y1, 11-(y0+y1)/2

Table 9-693 AFBCDM\_SIZE\_IN 0x4842

Bit(s)	R/W	Default	Description
28:16	R/W	1920	hsize_in : uns, default = 1920 , pic horz size in unit: pixel
12:0	R/W	1080	vsize_in : uns, default = 1080 , pic vert size in unit: pixel

Table 9-694 AFBCDM\_DEC\_DEF\_COLOR 0x4843

Bit(s)	R/W	Default	Description
29:20	R/W	256	def_color_y : uns, default = 256, afbc dec y default setting value
19:10	R/W	128	def_color_u : uns, default = 128, afbc dec u default setting value
9: 0	R/W	128	def_color_v : uns, default = 128, afbc dec v default setting value

Table 9-695 AFBCDM\_CONV\_CTRL 0x4844

Bit(s)	R/W	Default	Description
13:12	R/W	2	fmt_mode : uns, default = 2, 0:yuv444 1:yuv422 2:yuv420
11: 0	R/W	256	conv_lbuf_len : uns, default = 256, unit=16 pixel need to set = 2^n

Table 9-696 AFBCDM\_LBUF\_DEPTH 0x4845

Bit(s)	R/W	Default	Description
27:16	R/W	128	dec_lbuf_depth : uns, default = 128; // unit= 8 pixel
11:0	R/W	128	mif_lbuf_depth : uns, default = 128;

**Table 9-697 AFBCDM\_HEAD\_BADDR 0x4846**

Bit(s)	R/W	Default	Description
31:0	R/W	0x0	mif_info_baddr : uns, default = 32'h0;

**Table 9-698 AFBCDM\_BODY\_BADDR 0x4847**

Bit(s)	R/W	Default	Description
31:0	R/W	0x0001_0000	mif_data_baddr : uns, default = 32'h0001_0000;

**Table 9-699 AFBCDM\_SIZE\_OUT 0x4848**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	hszie_out: uns, default = 1920 ; // unit: 1 pixel
12:0	R/W	1080	Vsize_out : uns, default = 1080 ; // unit: 1 pixel

**Table 9-700 AFBCDM\_OUT\_YSCOPE 0x4849**

Bit(s)	R/W	Default	Description
28:16	R/W	0	out_vert_bgn : uns, default = 0 ; // unit: 1 pixel
12:0	R/W	1079	out_vert_end : uns, default = 1079 ; // unit: 1 pixel

**Table 9-701 AFBCDM\_STAT 0x484a**

Bit(s)	R/W	Default	Description
31:1	RO	0	ro_dbg_top_info : uns, default = 0
0	R/W	0	frm_end_stat : uns, default = 0 frame end status

**Table 9-702 AFBCDM\_VD\_CFMT\_CTRL 0x484b**

Bit(s)	R/W	Default	Description
31	R/W	0	cfmt_gclk_bit_dis : uns, default = 0 ; // if true, disable clock, otherwise enable clock
30	R/W	0	cfmt_soft_rst_bit : uns, default = 0 ; // soft rst bit
28	R/W	0	chfmt_rpt_pix : uns, default = 0 ; // if true, horizontal formatter use repeating to generate pixel, otherwise use bilinear interpolation
27:24	R/W	0	chfmt_ini_phase : uns, default = 0 ; // horizontal formatter initial phase
23	R/W	0	chfmt_rpt_p0_en : uns, default = 0 ; // horizontal formatter repeat pixel 0 enable
22:21	R/W	0	chfmt_yc_ratio : uns, default = 0 ; // horizontal Y/C ratio, 00: 1:1, 01: 2:1, 10: 4:1
20	R/W	0	chfmt_en : uns, default = 0 ; // horizontal formatter enable
19	R/W	0	cvfmt_phase0_always_en : uns, default = 0 ; //if true, always use phase0 while vertical formater, meaning always repeat data, no interpolation

Bit(s)	R/W	Default	Description
18	R/W	0	cvfmt_rpt_last_dis : uns, default = 0 ; //if true, disable vertical formatter chroma repeat last line
17	R/W	0	cvfmt_phase0_nrpt_en : uns, default = 0 ; //vertical formatter dont need repeat line on phase0, 1: enable, 0: disable
16	R/W	0	cvfmt_rpt_line0_en : uns, default = 0 ; //vertical formatter repeat line 0 enable
15:12	R/W	0	cvfmt_skip_line_num : uns, default = 0 ; //vertical formatter skip line num at the beginning
11:8	R/W	0	cvfmt_ini_phase : uns, default = 0 ; //vertical formatter initial phase
7:1	R/W	0	cvfmt_phase_step : uns, default = 0 ; //vertical formatter phase step (3.4)
0	R/W	0	cvfmt_en : uns, default = 0 ; //vertical formatter enable

Table 9-703 AFBCDM\_VD\_CFMT\_W 0x484c

Bit(s)	R/W	Default	Description
28:16	R/W	0x0	cfmt_w : uns, default = 0 ; //horizontal : formatter width
12:0	R/W	0x0	cvfmt_w : uns, default = 0 ; //vertical : formatter width

Table 9-704 AFBCDM\_MIF\_HOR\_SCOPE 0x484d

Bit(s)	R/W	Default	Description
25:16	R/W	0	mif_blk_bgn_h : uns, default = 0 ; // unit: 32 pixel/block hor
9: 0	R/W	59	mif_blk_end_h : uns, default = 59 ; // unit: 32 pixel/block hor

Table 9-705 AFBCDM\_MIF\_VER\_SCOPE 0x484e

Bit(s)	R/W	Default	Description
27:16	R/W	0	mif_blk_bgn_v : uns, default = 0 ; // unit: 32 pixel/block ver
11: 0	R/W	269	mif_blk_end_v : uns, default = 269 ; // unit: 32 pixel/block ver

Table 9-706 AFBCDM\_PIXEL\_HOR\_SCOPE 0x484f

Bit(s)	R/W	Default	Description
28:16	R/W	0	dec_pixel_bgn_h : uns, default = 0 ; // unit: pixel
12: 0	R/W	1919	dec_pixel_end_h : uns, default = 1919 ; // unit: pixel

Table 9-707 AFBCDM\_PIXEL\_VER\_SCOPE 0x4850

Bit(s)	R/W	Default	Description
28:16	R/W	0	dec_pixel_bgn_v : uns, default = 0 ; // unit: pixel
12: 0	R/W	1079	dec_pixel_end_v : uns, default = 1079 ; // unit: pixel

**Table 9-708 AFBCDM\_VD\_CFMT\_H 0x4851**

Bit(s)	R/W	Default	Description
12:0	R/W	142	cfmt : uns, default =142;// vertical : formatter height

**Table 9-709 AFBCDM\_IQUANT\_ENABLE 0x4852**

Bit(s)	R/W	Default	Description
11	R/W	0x0	reg_quant_expand_en_1 : //unsigned, RW, enable for quantization value expansion
10	R/W	0x0	reg_quant_expand_en_0 : //unsigned, RW, enable for quantization value expansion
9: 8	R/W	-1	reg_bcleav_ofst : //signed , RW, default = 0 bcleave ofset to get lower range, especially under lossy, for v1/v2, x=0 is equivalent, default = -1;
4	R/W	0	reg_quant_enable_1 : // unsigned , RW, default = 0 enable for quant to get some lossy
0	R/W	0	reg_quant_enable_0 : // unsigned , RW, default = 0 enable for quant to get some lossy

**Table 9-710 AFBCDM\_IQUANT\_LUT\_1 0x4853**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_iquant_yclut_0_11 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
26:24	R/W	1	reg_iquant_yclut_0_10 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
22:20	R/W	2	reg_iquant_yclut_0_9 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
18:16	R/W	3	reg_iquant_yclut_0_8 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
14:12	R/W	4	reg_iquant_yclut_0_7 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	5	reg_iquant_yclut_0_6 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	5	reg_iquant_yclut_0_5 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	4	reg_iquant_yclut_0_4 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

**Table 9-711 AFBCDM\_IQUANT\_LUT\_2 0x4854**

Bit(s)	R/W	Default	Description
14:12	R/W	3	reg_iquant_yclut_0_3 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	2	reg_iquant_yclut_0_2 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

Bit(s)	R/W	Default	Description
6: 4	R/W	1	reg_iquant_yclut_0_1 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	0	reg_iquant_yclut_0_0 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

Table 9-712 AFBCDM\_IQUANT\_LUT\_3 0x4855

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_iquant_yclut_1_11 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
26:24	R/W	1	reg_iquant_yclut_1_10 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
22:20	R/W	2	reg_iquant_yclut_1_9 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
18:16	R/W	3	reg_iquant_yclut_1_8 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
14:12	R/W	4	reg_iquant_yclut_1_7 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	5	reg_iquant_yclut_1_6 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	5	reg_iquant_yclut_1_5 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	4	reg_iquant_yclut_1_4 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

Table 9-713 AFBCDM\_IQUANT\_LUT\_4 0x4856

Bit(s)	R/W	Default	Description
14:12	R/W	3	reg_iquant_yclut_1_3 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	2	reg_iquant_yclut_1_2 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	1	reg_iquant_yclut_1_1 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	0	reg_iquant_yclut_1_0 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

Table 9-714 AFBCDM\_ROT\_CTRL 0x4860

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_rot_ohds2_mode : uns, default = 0 , rot output format down horizontal drop mode, 0:average , 1:use 0, 2: use 1
29:28	R/W	0	reg_rot_ovds2_mode : uns, default = 0 , rot output format down vertical drop mode, 0:average , 1:use 0, 2: use 1
27	R/W	0	reg_pip_mode : uns, default = 0 , 0:dec_source from vdin/dos 1: dec_source from pip

Bit(s)	R/W	Default	Description
26:24	R/W	0	reg_rot_uv_vshrk_drop_mode: uns, default = 0, 0: average (1/2: 1:left 2:right) (1/4:1:[0] 2:p[1] 3:[2] 4:[3] , 5:left_121 6:right_121
22:20	R/W	0	reg_rot_uv_hshrk_drop_mode: uns, default = 0, 0: average (1/2: 1:left 2:right) (1/4:1:[0] 2:p[1] 3:[2] 4:[3] , 5:left_121 6:right_121
19:18	R/W	0	reg_rot_uv_vshrk_ratio: uns, default = 0, 0:no shrink 1:1/2 shrink 2:1/4 shrink
17:16	R/W	0	reg_rot_uv_hshrk_ratio: uns, default = 0, 0:no shrink 1:1/2 shrink 2:1/4 shrink
14:12	R/W	0	reg_rot_y_vshrk_drop_mode : uns,default = 0, 0: average (1/2: 1:left 2:right) (1/4:1:[0] 2:p[1] 3:[2] 4:[3] , 5:left_121 6:right_121
10:8	R/W	0	reg_rot_y_hshrk_drop_mode : uns,default = 0, 0: average (1/2: 1:left 2:right) (1/4:1:[0] 2:p[1] 3:[2] 4:[3] , 5:left_121 6:right_121
7:6	R/W	0	reg_rot_y_vshrk_ratio:uns, default =0, 0:disable 1: 1/2 v-shrink 2:1/4 v-shrink
5:4	R/W	0	reg_rot_y_hshrk_ratio:uns, default =0, 0:disable 1: 1/2 h-shrink 2:1/4 h-shrink
3:2	R/W	0	reg_rot_uv422_drop_mode :uns, default = 0, 0:average two pixel 1:keep left 2:keep right
1	R/W	0	reg_rot_uv422_omode : uns,default = 0, When rot input is fmt 422, 0:output_uv422 1:output uv420
0	R/W	0	reg_rot_enable : uns, default = 0, rotation enable

Table 9-715 AFBCDM\_ROT\_SCOPE 0x4861

Bit(s)	R/W	Default	Description
25:20	R/W	0	reg_rot_debug_probe : uns, default= 0; bits [2:0] for y, bits [5:3] for uv 0: iblk_size, 1:oblk_size, 2:iblk_cnt, 3:oblk_cnt, 4:hsize_in, 5:vsize_in, 6:vstep
19	R/W	0	reg_rot_dout_ds_mode_sw: uns, default= 0, 0: use hardware mode, 1: use software mode
18:17	R/W	0	reg_dout_ds_mode:uns, default = 0 ,rot output format downscale mode : 0 : h_downscale, 1: v_downscale
16	R/W	1	reg_ifmt_force444: uns, default = 1 : rot input fmt force as 444
15:14	R/W	0	reg_rot_ofmt_mode: uns, default = 0, rot output format mode
13:12	R/W	0	reg_rot_combits_out_y: uns, default = 0 ,rot output combit y
11:10	R/W	0	reg_rot_combits_out_uv: uns, default = 0 ,rot output combit uv
9:8	R/W	0	reg_rot_wrbgn_v: uns, default = 0, rot pic vertical size window begin pixel
4:0	R/W	0	reg_rot_wrbgn_h: uns ,default = 0, rot pic horizontal size window begin pixel

Table 9-716 AFBCDM\_RPLC\_CTRL 0x4862

Bit(s)	R/W	Default	Description
31	R/W	0	reg_rplc_inter_corr_en: uns, default = 0 //pip replace inter_frame edge correct enable
30	R/W	0	reg_rplc_dummy_corr_en: uns, default = 0 //pip replace outside of real-pipframe correct enable
29	R/W	1	reg_rplc_bypas: uns, default = 1, //pip replace module bypass

Bit(s)	R/W	Default	Description
28:16	R/W	1080	reg_rplc_vsize_in: uns, default = 1080
12:0	R/W	1920	reg_rplc_hsize_in: uns, default = 1920

**Table 9-717 AFBCDM\_RPLC\_PICEN 0x4863**

Bit(s)	R/W	Default	Description
27:16	R/W	0	reg_rplc_def_color_y: uns, default = 0 //pip replace def_color_y
15:0	R/W	0xffff	Reg_rplc_pic_enable: uns, default =16'hffff, //pip_replace pip_picure enable

**Table 9-718 AFBCDM\_RPLC\_DEFCOL 0x4864**

Bit(s)	R/W	Default	Description
23:12	R/W	0	reg_rplc_def_color_v: uns, default =0, //pip replace def_color_v
11:0	R/W	0	reg_rplc_def_color_u: uns, default =0, //pip replace def_color_u

**Table 9-719 AFBCDM\_RPLC\_SCPXN\_ADDR 0x4865**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_rplc_scpxn_addr: uns, default =0, //pip replace scopes of 16 windows addr

**Table 9-720 AFBCDM\_RPLC\_SCPXN\_DATA 0x4866**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_rplc_scpxn_data: uns, default = 0, //pip_replace scopes of 16 windows data

**Table 9-721 AFBCDM\_ROT\_RO\_STAT 0x4867**

Bit(s)	R/W	Default	Description
31:0	R/O	0	ro_rot_debug: uns, RO ,default = 0, rot some status

VD2\_AFBC dec is same as vd1 AFBC dec , BASE\_addr is from 0x4840 to 0x48c0

DI AFBC dec is same as vd1 AFBC dec , BASE\_addr is from 0x4840 to 0x5440-0x56c0

## VPU FILM GRAIN Registers

**Table 9-722 FGRAIN\_CTRL 0x4870**

Bit(s)	R/W	Default	Description
25:24	R/W	0	reg_sync_ctrl: [0] 1 to shaddow reg_fgrain_glb_en; [1] to shaddow reg_fgrain_loc_en.
22	R/W	0	reg_dma_st_clr: write 1 to clear DMA error status, then write to 0
21	R/W	0	reg_hold4dma_scale:1 to wait DMA scale data ready before accept input data
20	R/W	0	reg_hold4dma_tbl: 1 to wait DMA grain table data ready before accept input data



Bit(s)	R/W	Default	Description
19	R/W	0	reg_cin_uv_swap: 1 to swap U/V input
18	R/W	0	reg_cin_rev: 1 to reverse the U/V input order
17	R/W	0	reg_yin_rev: 1 to reverse the Y input order
16	R/W	1	reg_fgrain_ext_imode: 0 to indicate the input data is (<<2) in 8bit mode
15	R/W	0	reg_use_par_apply_fgrain: 1 to use parameter "apply_fgrain" from DMA table
14	R/W	0	reg_fgrain_last_in_mode □ 1 to keep film grain noise generator though the input is finished for none-afbc mode
13	R/W	0	reg_fgrain_use_sat4bp: 1 to use SW max/min for output saturation and not (1<<bitdepth)-1/0 when film grain bypass.
12	R/W	1	reg_apply_c_mode: following C code
11	R/W	1	reg_fgrain_tbl_sign_mode: 0 to indicate signed bit is not extended in 8bit mode
10	R/W	1	reg_fgrain_tbl_ext_mode: 0 to indicate the grain table value is (<<2) in 8bit mode
9: 8	R/W	2	reg_fmt_mode: 0:444; 1:422; 2:420; 3:reserved. Only 420 have affect, others mode will bypass film grain
7: 6	R/W	1	reg_comp_bits: 0:8bits; 1:10bits, others reserved.
5	R/W	0	reg_rev_mode □ 1 for vertical scan reverse
4	R/W	0	reg_rev_mode □ 1 for horizontal scan reverse
2	R/W	1	reg_block_mode: 0: none-afbc mode; 1 afbc mode
1	R/W	0	reg_fgrain_loc_en: frame-based fgrain enable
0	R/W	0	reg_fgrain_glb_en: global-based fgrain enable

Table 9-723 FGRAIN\_WIN\_H 0x4871

Bit(s)	R/W	Default	Description
31:16	R/W	3812	reg_win_end_h: horizontal end position of window to do film grain: LSB 5bit should be 0 for afbc mode. For example, if horizontal size is 3840 with no window cut, it will set to (3840-32)
15: 0	R/W	0	reg_win_bgn_h: horizontal start position of window to do film grain: LSB 5bit should be 0 for afbc mode

Table 9-724 FGRAIN\_WIN\_V 0x4872

Bit(s)	R/W	Default	Description
31:16	R/W	2156	reg_win_end_v: vertical end position of window to do film grain: LSB 2bit should be 0 for afbc mode. For example, if vertical size is 2160 with no window cut, it will set to (2160-4)
15: 0	R/W	0	reg_win_bgn_v: vertical start position of window to do film grain: LSB 2bit should be 0 for afbc mode

**Table 9-725 FGRAIN\_SW\_Y\_RANGE 0x4873**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_fgrain_sw_yrange, 1 to use fgrain_ymin/ymax for saturation. Otherwise use 16/235 when parameter: clip_to_restricted_range =1, 0/255 when clip_to_restricted_range=0.
25:16	R/W	1023	reg_fgrain_ymax: max value for SW saturation
9: 0	R/W	0	reg_fgrain_ymin: min value for SW saturation

**Table 9-726 FGRAIN\_SW\_C\_RANGE 0x4874**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_fgrain_sw_crange, 1 to use fgrain_cmin/cmax for saturation. Otherwise use 16/240 when parameter: clip_to_restricted_range =1, 0/255 when clip_to_restricted_range=0.
25:16	R/W	1023	reg_fgrain_cmax: max value for SW saturation
9: 0	R/W	0	reg_fgrain_cmin: min value for SW saturation

**Table 9-727 FGRAIN\_GCLK\_CTRL\_0 0x4875**

Bit(s)	R/W	Default	Description
31 :0	R/W	0	reg_fgrain_gclk_ctrl0: Gate clock control.

**Table 9-728 FGRAIN\_GCLK\_CTRL\_1 0x4876**

Bit(s)	R/W	Default	Description
31 :0	R/W	0	reg_fgrain_gclk_ctrl1: Gate clock control.

**Table 9-729 FGRAIN\_GCLK\_CTRL\_2 0x4877**

Bit(s)	R/W	Default	Description
31 :0	R/W	0	reg_fgrain_gclk_ctrl2: Gate clock control.

**Table 9-730 FGRAIN\_PARAM\_ADDR 0x4878**

Bit(s)	R/W	Default	Description
31 :0	R/W	0	Parameter table index. it will increase automatically when read FGRAIN_PARAM_DATA

**Table 9-731 FGRAIN\_PARAM\_DATA 0x4879**

Bit(s)	R/W	Default	Description
31 :0	RO	0	Parameter table data Index=0: mode parameter is ready scale parameter is ready noise table parameter is ready Index=1: Parameter: apply_grain Parameter: overlap Parameter: chroma_scaling_from_luma Parameter: clip_to_restricted_range Parameter: mc_identity [7:5] Parameter: data depth [11:8] Parameter: y_num_points [15:12] Parameter: cb_num_points [19:16] Parameter: cr_num_points Index=2: [15:0] Parameter: rand seed Index=3: [0] DMA status: 1 to indicate DMA transfer is too slow, finish after input data arrived

VD2 FGRAIN is same as vd1 FGRAIN , BASE\_addr is from 0x4870 to 0x48f0

DI FGRAIN is same as vd1 FGRAIN , BASE\_addr is from 0x4870 to 0x5440-0x56f0

### 9.2.3.9 NR2 Registers

#### Register Address

- DET3D\_MOTN\_CFG 0xff005cd0
- DET3D\_CB\_CFG 0xff005cd4
- DET3D\_SPLT\_CFG 0xff005cd8
- DET3D\_HV\_MUTE 0xff005cdc
- DET3D\_MAT\_STA\_P1M1 0xff005ce0
- DET3D\_MAT\_STA\_P1TH 0xff005ce4
- DET3D\_MAT\_STA\_M1TH 0xff005ce8
- DET3D\_MAT\_STA\_RSFT 0xff005cec
- DET3D\_MAT\_SYMTC\_TH 0xff005cf0
- DET3D\_RO\_DET\_CB\_HOR 0xff005cf4
- DET3D\_RO\_DET\_CB\_VER 0xff005cf8
- DET3D\_RO\_SPLT\_HT 0xff005cfc
- NR2\_MET\_NM\_CTRL 0xff005d14
- NR2\_MET\_NM\_YCTRL 0xff005d18
- NR2\_MET\_NM\_CCTRL 0xff005d1c
- NR2\_MET\_NM\_TNR 0xff005d20
- NR2\_MET\_NMFRM\_TNR\_YLEV 0xff005d24
- NR2\_MET\_NMFRM\_TNR\_YCNT 0xff005d28
- NR2\_MET\_NMFRM\_TNR\_CLEV 0xff005d2c
- NR2\_MET\_NMFRM\_TNR\_CCNT 0xff005d30
- NR2\_3DEN\_MODE 0xff005d34

- NR2\_SW\_EN 0xff005d3c
- NR2\_FRM\_SIZE 0xff005d40
- NR2\_SNR\_SAD\_CFG 0xff005d44
- NR2\_MATNR\_SNR\_OS 0xff005d48
- NR2\_MATNR\_SNR\_NRM\_CFG 0xff005d4c
- NR2\_MATNR\_SNR\_NRM\_GAIN 0xff005d50
- NR2\_MATNR\_SNR\_LPF\_CFG 0xff005d54
- NR2\_MATNR\_SNR\_USF\_GAIN 0xff005d58
- NR2\_MATNR\_SNR\_EDGE2B 0xff005d5c
- NR2\_MATNR\_BETA\_EGAIN 0xff005d60
- NR2\_MATNR\_BETA\_BRT 0xff005d64
- NR2\_MATNR\_XBETA\_CFG 0xff005d68
- NR2\_MATNR\_YBETA\_SCL 0xff005d6c
- NR2\_MATNR\_CBETA\_SCL 0xff005d70
- NR2\_SNR\_MASK 0xff005d74
- NR2\_SAD2NORM\_LUT0 0xff005d78
- NR2\_SAD2NORM\_LUT1 0xff005d7c
- NR2\_SAD2NORM\_LUT2 0xff005d80
- NR2\_SAD2NORM\_LUT3 0xff005d84
- NR2\_EDGE2BETA\_LUT0 0xff005d88
- NR2\_EDGE2BETA\_LUT1 0xff005d8c
- NR2\_EDGE2BETA\_LUT2 0xff005d90
- NR2\_EDGE2BETA\_LUT3 0xff005d94
- NR2\_MOTION2BETA\_LUT0 0xff005d98
- NR2\_MOTION2BETA\_LUT1 0xff005d9c
- NR2\_MOTION2BETA\_LUT2 0xff005da0
- NR2\_MOTION2BETA\_LUT3 0xff005da4
- NR2\_IIR\_CTRL 0xff005d38
- NR2\_MATNR\_MTN\_CRTL 0xff005da8
- NR2\_MATNR\_MTN\_CRTL2 0xff005dac
- NR2\_MATNR\_MTN\_COR 0xff005db0
- NR2\_MATNR\_MTN\_GAIN 0xff005db4
- NR2\_MATNR\_DEGHOST 0xff005db8
- NR2\_MATNR\_ALPHALP\_LUT0 0xff005dbc
- NR2\_MATNR\_ALPHALP\_LUT1 0xff005dc0
- NR2\_MATNR\_ALPHALP\_LUT2 0xff005dc4
- NR2\_MATNR\_ALPHALP\_LUT3 0xff005dc8
- NR2\_MATNR\_ALPHAHP\_LUT0 0xff005dcc
- NR2\_MATNR\_ALPHAHP\_LUT1 0xff005dd0
- NR2\_MATNR\_ALPHAHP\_LUT2 0xff005dd4
- NR2\_MATNR\_ALPHAHP\_LUT3 0xff005dd8
- NR2\_MATNR\_MTNB\_BRT 0xff005ddc
- NR2\_CUE\_MODE 0xff005de0
- NR2\_CUE\_CON\_MOT\_TH 0xff005de4

- NR2\_CUE\_CON\_DIF0 0xff005de8
- NR2\_CUE\_CON\_DIF1 0xff005dec
- NR2\_CUE\_CON\_DIF2 0xff005df0
- NR2\_CUE\_CON\_DIF3 0xff005df4
- NR2\_CUE\_PRG\_DIF 0xff005df8
- NR2\_CONV\_MODE 0xff005dfc
- DET3D\_RO\_SPLT\_HB 0xff005e00
- DET3D\_RO\_SPLT\_VL 0xff005e04
- DET3D\_RO\_SPLT\_VR 0xff005e08
- DET3D\_RO\_MAT\_LUMA\_LR 0xff005e0c
- DET3D\_RO\_MAT\_LUMA\_TB 0xff005e10
- DET3D\_RO\_MAT\_CHRU\_LR 0xff005e14
- DET3D\_RO\_MAT\_CHRU\_TB 0xff005e18
- DET3D\_RO\_MAT\_CHRV\_LR 0xff005e1c
- DET3D\_RO\_MAT\_CHRV\_TB 0xff005e20
- DET3D\_RO\_MAT\_HEDG\_LR 0xff005e24
- DET3D\_RO\_MAT\_HEDG\_TB 0xff005e28
- DET3D\_RO\_MAT\_VEDG\_LR 0xff005e2c
- DET3D\_RO\_MAT\_VEDG\_TB 0xff005e30
- DET3D\_RO\_MAT\_MOTN\_LR 0xff005e34
- DET3D\_RO\_MAT\_MOTN\_TB 0xff005e38
- DET3D\_RO\_FRM\_MOTN 0xff005e3c
- DET3D\_RAMRD\_ADDR\_PORT 0xff005e68
- DET3D\_RAMRD\_DATA\_PORT 0xff005e6c
- NR2\_CFR\_PARA\_CFG0 0xff005e70
- NR2\_CFR\_PARA\_CFG1 0xff005e74
- NR3\_MODE 0xff00bfc0
- NR3\_COOP\_PARA 0xff00bfc4
- NR3\_CNOOP\_GAIN 0xff00bfc8
- NR3\_YMOT\_PARA 0xff00bfcc
- NR3\_CMOT\_PARA 0xff00bfd0
- NR3\_SUREMOT\_YGAIN 0xff00bfd4
- NR3\_SUREMOT\_CGAIN 0xff00bfd8

## Register Description

**Table 9-732 DET3D\_MOTN\_CFG 0x1734**

Bit(s)	R/W	Default	Description
16	R/W	0	reg_det3d_intr_en : Det3d interrupt enable
9:8	R/W	0	reg_Det3D_Motion_Mode : U2 Different mode for Motion Calculation of Luma and Chroma: 0 : MotY, 1: $(2 * \text{MotY} + (\text{MotU} + \text{MotV})) / 4$ ; 2: $\text{Max}(\text{MotY}, \text{MotU}, \text{MotV})$ ; 3: $\text{Max}(\text{MotY}, (\text{MotU} + \text{MotV}) / 2)$
7:4	R/W	0	reg_Det3D_Motion_Core_Rate : U4 K Rate to Edge (HV) details for coring of Motion Calculations, normalized to 32
3:0	R/W	0	reg_Det3D_Motion_Core_Thrd : U4 2X: static coring value for Motion Detection.

**Table 9-733 DET3D\_CB\_CFG 0x1735**

Bit(s)	R/W	Default	Description
7:4	R/W	0	reg_Det3D_ChessBd_HV_ofst : U4, Noise immune offset for Horizontal or vertical combing detection.
3:0	R/W	0	reg_Det3D_ChessBd_NHV_ofst : U4, Noise immune offset for NON-Horizontal or vertical combing detection.

**Table 9-734 DET3D\_SPLT\_CFG 0x1736**

Bit(s)	R/W	Default	Description
7:4	R/W	0x0	reg_Det3D_SplitValid_ratio : U4, Ratio between max_value and the avg_value of the edge mapping for split line valid detection. The smaller of this value, the easier of the split line detected.
3:0	R/W	0x0	reg_Det3D_AvgIdx_ratio : U4, Ratio to the avg_value of the edge mapping for split line position estimation. The smaller of this value, the more samples will be added to the estimation.

**Table 9-735 DET3D\_HV\_MUTE 0x1737**

Bit(s)	R/W	Default	Description
23:20	R/W	0x0	reg_Det3D_Edge_Ver_Mute : U4 X2: Horizontal pixels to be mute from H/V Edge calculation Top and Bottom border part.
19:16	R/W	0x0	reg_Det3D_Edge_Hor_Mute : U4 X2: Horizontal pixels to be mute from H/V Edge calculation Left and right border part.
15:12	R/W	0x0	reg_Det3D_ChessBd_Ver_Mute : U4 X2: Horizontal pixels to be mute from ChessBoard statistics calculation in middle part
11:8	R/W	0x0	reg_Det3D_ChessBd_Hor_Mute : U4 X2: Horizontal pixels to be mute from ChessBoard statistics calculation in middle part
7:4	R/W	0x0	reg_Det3D_STA8X8_Ver_Mute : U4 1X: Vertical pixels to be mute from 8x8 statistics calculation in each block.
3:0	R/W	0x0	reg_Det3D_STA8X8_Hor_Mute : U4 1X: Horizontal pixels to be mute from 8x8 statistics calculation in each block.

**Table 9-736 DET3D\_MAT\_STA\_P1M1 0x1738**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_Det3D_STA8X8_P1_K0_R8 : U8 SAD to SAI ratio to decide P1, normalized to 256 (0.8)
23:16	R/W	0x0	reg_Det3D_STA8X8_P1_K1_R7 : U8 SAD to ENG ratio to decide P1, normalized to 128 (0.5)
15:8	R/W	0x0	reg_Det3D_STA8X8_M1_K0_R6 : U8 SAD to SAI ratio to decide M1, normalized to 64 (1.1)
7:0	R/W	0x0	reg_Det3D_STA8X8_M1_K1_R6 : U8 SAD to ENG ratio to decide M1, normalized to 64 (0.8)

**Table 9-737 DET3D\_MAT\_STA\_P1TH 0x1739**

Bit(s)	R/W	Default	Description
23:16	R/W	0x0	reg_Det3D_STAYUV_P1_TH_L4 : U8 SAD to ENG Thrd offset to decide P1, X16 (100)
15:8	R/W	0x0	reg_Det3D_STAEDG_P1_TH_L4 : U8 SAD to ENG Thrd offset to decide P1, X16 (80)
7:0	R/W	0x0	reg_Det3D_STAMOT_P1_TH_L4 : U8 SAD to ENG Thrd offset to decide P1, X16 (48)

**Table 9-738 DET3D\_MAT\_STA\_M1TH 0x173a**

Bit(s)	R/W	Default	Description
23:16	R/W	0x0	reg_Det3D_STAYUV_M1_TH_L4 : U8 SAD to ENG Thrd offset to decide M1, X16 (100)
15:8	R/W	0x0	reg_Det3D_STAEDG_M1_TH_L4 : U8 SAD to ENG Thrd offset to decide M1, X16 (80)
7:0	R/W	0x0	reg_Det3D_STAMOT_M1_TH_L4 : U8 SAD to ENG Thrd offset to decide M1, X16 (64)

**Table 9-739 DET3D\_MAT\_STA\_RSFT 0x173b**

Bit(s)	R/W	Default	Description
5:4	R/W	0x0	reg_Det3D_STAYUV_RSHFT : U2 YUV statistics SAD and SAI calculation result right shift bits to accommodate the 12bits clipping: 0 : mainly for images <=720x480: 1: mainly for images <=1366x768: 2: mainly for images <=1920X1080: 2; 3: other higher resolutions
3:2	R/W	0x0	reg_Det3D_STAEDG_RSHFT : U2 Horizontal and Vertical Edge Statistics SAD and SAI calculation result right shift bits to accommodate the 12bits clipping: 0 : mainly for images <=720x480: 1: mainly for images <=1366x768: 2: mainly for images <=1920X1080: 2; 3: other higher resolutions
1:0	R/W	0x0	reg_Det3D_STAMOT_RSHFT : U2 Motion SAD and SAI calculation result right shift bits to accommodate the 12bits clipping: 0 : mainly for images <=720x480: 1: mainly for images <=1366x768: 2: mainly for images <=1920X1080: 2; 3: other higher resolutions

**Table 9-740 DET3D\_MAT\_SYMTC\_TH 0x173c**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_Det3D_STALUM_symtc_Th : U8 threshold to decide if the Luma statistics is TB or LR symmetric.
23:16	R/W	0x0	reg_Det3D_STACHR_symtc_Th : U8 threshold to decide if the Chroma (UV) statistics is TB or LR symmetric.
15:8	R/W	0x0	reg_Det3D_STAEDG_symtc_Th : U8 threshold to decide if the Horizontal and Vertical Edge statistics is TB or LR symmetric.
7:0	R/W	0x0	reg_Det3D_STAMOT_symtc_Th : U8 threshold to decide if the Motion statistics is TB or LR symmetric.

**Table 9-741 DET3D\_RO\_DET\_CB\_HOR 0x173d**

Bit(s)	R/W	Default	Description
31:16	R.O	0x0	RO_Det3D_ChessBd_NHor_value : U16 X64: number of Pixels of Horizontally Surely NOT matching Chessboard pattern.
15:0	R.O	0x0	RO_Det3D_ChessBd_Hor_value : U16 X64: number of Pixels of Horizontally Surely matching Chessboard pattern.

**Table 9-742 DET3D\_RO\_DET\_CB\_VER 0x173e**

Bit(s)	R/W	Default	Description
31:16	R.O	0x0	RO_Det3D_ChessBd_NVer_value : U16 X64: number of Pixels of Vertically Surely NOT matching Chessboard pattern.
15:0	R.O	0x0	RO_Det3D_ChessBd_Ver_value : U16 X64: number of Pixels of Vertically Surely matching Chessboard pattern.

**Table 9-743 DET3D\_RO\_SPLT\_HT 0x173f**

Bit(s)	R/W	Default	Description
24	R.O	0x0	RO_Det3D_Split_HT_valid : U1 horizontal LR split border detected valid signal for top half picture
20:16	R.O	0x0	RO_Det3D_Split_HT_pxnum : U5 number of pixels included for the LR split position estimation for top half picture
9:0	R.O	0x0	RO_Det3D_Split_HT_idxX4 : S10 X4: horizontal pixel shifts of LR split position to the (ColMax/2) for top half picture

**Table 9-744 NR2\_MET\_NM\_CTRL 0x1745**

Bit(s)	R/W	Default	Description
28	R/W	0x0	reg_NM_reset : Reset to the status of the Loop filter.
27:24	R/W	0x0	reg_NM_calc_length : Length mode of the Noise measurement sample number for statistics. 0 : 256 samples; 1: 512 samples; 2: 1024 samples; iX: 2^(8+x) samples
23:20	R/W	0x0	reg_NM_inc_step : Loop filter input gain increase step.
19:16	R/W	0x0	reg_NM_dec_step : Loop filter input gain decrease step.
15:8	R/W	0x0	reg_NM_YHPmot_thrd : Luma channel HP portion motion for condition of pixels included in Luma Noise measurement.
7:0	R/W	0x0	reg_NM_CHPmot_thrd : Chroma channel HP portion motion for condition of pixels included in Chroma Noise measurement.

**Table 9-745 NR2\_MET\_NM\_YCTRL 0x1746**

Bit(s)	R/W	Default	Description
31:28	R/W	0x0	reg_NM_YPLL_target : Target rate of NM_Ynoise_thrd to mean of the Luma Noise
27:24	R/W	0x0	reg_NM_YLPmot_thrd : Luma channel LP portion motion for condition of pixels included in Luma Noise measurement.



Bit(s)	R/W	Default	Description
23:16	R/W	0x0	reg_NM_YHPmot_thr_min : Minimum threshold for Luma channel HP portion motion to decide whether the pixel will be included in Luma noise measurement.
15:8	R/W	0x0	reg_NM_YHPmot_thr_max : Maximum threshold for Luma channel HP portion motion to decide whether the pixel will be included in Luma noise measurement.
7:0	R/W	0x0	reg_NM_Ylock_rate : Rate to decide whether the Luma noise measurement is lock or not.

**Table 9-746 NR2\_MET\_NM\_CTRL 0x1747**

Bit(s)	R/W	Default	Description
31:28	R/W	0x0	reg_NM_CPLL_target : Target rate of NM_Cnoise_thr to mean of the Chroma Noise
27:24	R/W	0x0	reg_NM_CLPmot_thr : Chroma channel LP portion motion for condition of pixels included in Chroma Noise measurement.
23:16	R/W	0x0	reg_NM_CHPmot_thr_min : Minimum threshold for Chroma channel HP portion motion to decide whether the pixel will be included in Chroma noise measurement.
15:8	R/W	0x0	reg_NM_CHPmot_thr_max : Maximum threshold for Chroma channel HP portion motion to decide whether the pixel will be included in Chroma noise measurement.
7:0	R/W	0x0	reg_NM_Clock_rate : Rate to decide whether the Chroma noise measurement is lock or not;

**Table 9-747 NR2\_MET\_NM\_TNR 0x1748**

Bit(s)	R/W	Default	Description
25	R.O	0x0	ro_NM_TNR_Ylock : Read-only register to tell ifLuma channel noise measurement is locked or not.
24	R.O	0x0	ro_NM_TNR_Clock : Read-only register to tell if Chroma channel noise measurement is locked or not.
23:12	R.O	0x0	ro_NM_TNR_Ylevel : Read-only register to give Luma channel noise level. It was 16x of pixel difference in 8 bits of YHPmot.
11:0	R.O	0x0	ro_NM_TNR_Clevel : Read-only register to give Chroma channel noise level. It was 16x of pixel difference in 8 bits of CHPmot.

**Table 9-748 NR2\_MET\_NMFRM\_TNR\_YLEV 0x1749**

Bit(s)	R/W	Default	Description
28:0	R.O	0x0	ro_NMFrm_TNR_Ylevel : Frame based Read-only register to give Luma channel noise level within one frame/field.

**Table 9-749 NR2\_MET\_NMFRM\_TNR\_YCNT 0x174a**

Bit(s)	R/W	Default	Description
23:0	R.O	0x0	ro_NMFrm_TNR_Ycount : Number of Luma channel pixels included in Frame/Field based noise level measurement.

**Table 9-750 NR2\_MET\_NMFRM\_TNR\_CLEV 0x174b**

Bit(s)	R/W	Default	Description
28:0	R.O	0x0	ro_NMFRm_TNR_Clevel : Frame based Read-only register to give Chroma channel noise level within one frame/field.

**Table 9-751 NR2\_MET\_NMFRM\_TNR\_CCNT 0x174c**

Bit(s)	R/W	Default	Description
23:0	R.O	0x0	ro_NMFRm_TNR_Ccount : Number of Chroma channel pixels included in Frame/Field based noise level measurement.

**Table 9-752 NR2\_3DEN\_MODE 0x174d**

Bit(s)	R/W	Default	Description
6:4	R/W	0x0	Blend_3dnr_en_r :
2:0	R/W	0x0	Blend_3dnr_en_l :

**Table 9-753 NR2\_IIR\_CTRL 0x174e**

Bit(s)	R/W	Default	Description
15:14	R/W	0x0	reg_LP_IIR_8bit_mode : LP IIR membitwidth mode:0: 10bits will be store in memory;1: 9bits will be store in memory; 2 : 8bits will be store in memory;3: 7bits will be store in memory;
13:12	R/W	0x0	reg_LP_IIR_mute_mode : Mode for the LP IIR mute,
11:8	R/W	0x0	reg_LP_IIR_mute_thrd : Threshold of LP IIR mute to avoid ghost:
7:6	R/W	0x0	reg_HP_IIR_8bit_mode : IIR membitwidth mode:0: 10bits will be store in memory;1: 9bits will be store in memory; 2 : 8bits will be store in memory;3: 7bits will be store in memory;
5:4	R/W	0x0	reg_HP_IIR_mute_mode : Mode for theLP IIR mute
3:0	R/W	0x0	reg_HP_IIR_mute_thrd : Threshold of HP IIR mute to avoid ghost

**Table 9-754 NR2\_SNR\_SAD\_CFG 0x1751**

Bit(s)	R/W	Default	Description
12	R/W	0x1	reg_MATNR_SNR_SAD_CenRPL : U1, Enable signal for Current pixel position SAD to be replaced by SAD_min.0: do not replace Current pixel position SAD by SAD_min;1: do replacements
11:8	R/W	0x3	reg_MATNR_SNR_SAD_coring : Coring value of the intra-frame SAD. sum = (sum - reg_MATNR_SNR_SAD_coring);sum = (sum<0) ? 0: (sum>255)? 255: sum;
6:5	R/W	0x1	reg_MATNR_SNR_SAD_WinMod : Unsigned, Intra-frame SAD matching window mode:0: 1x1; 1: [1 1 1] 2: [1 2 1]; 3: [1 2 2 1];
4:0	R/W	0x1	Sad_coef_num : Sad coefficient

**Table 9-755 NR2\_MATNR\_SNR\_OS 0x1752**

Bit(s)	R/W	Default	Description
7:4	R/W	0x8	reg_MATNR_SNR_COS : SNR Filter overshoot control margin for UV channel (X2 to u10 scale)
3:0	R/W	0xd	reg_MATNR_SNR_YOS : SNR Filter overshoot control margin for luma channel (X2 to u10 scale)

**Table 9-756 NR2\_MATNR\_SNR\_NRM\_CFG 0x1753**

Bit(s)	R/W	Default	Description
23:16	R/W	0x40	reg_MATNR_SNR_NRM_ofst : Edge based SNR boosting normalization offset to SAD_max ;
15:8	R/W	0xff	reg_MATNR_SNR_NRM_max : Edge based SNR boosting normalization Max value
7:0	R/W	0x0	reg_MATNR_SNR_NRM_min : Edge based SNR boosting normalization Min value

**Table 9-757 NR2\_MATNR\_SNR\_NRM\_GAIN 0x1754**

Bit(s)	R/W	Default	Description
15:8	R/W	0x0	reg_MATNR_SNR_NRM_Cgain : Edge based SNR boosting normalization Gain for Chrm channel (norm 32 as 1)
7:0	R/W	0x20	reg_MATNR_SNR_NRM_Ygain : Edge based SNR boosting normalization Gain for Luma channel (norm 32 as 1)

**Table 9-758 NR2\_MATNR\_SNR\_LPF\_CFG 0x1755**

Bit(s)	R/W	Default	Description
23:16	R/W	0xc	reg_MATNR_SNR_LPF_SADmaxTH : U8, Threshold to SADmax to use TNRLPF to replace SNRLPF. i.e.if (SAD_max<reg_MATNR_SNR_LPF_SADmaxTH) SNRLPF_yuv[k] = TNRLPF_yuv[k];
13:11	R/W	0x2	reg_MATNR_SNR_LPF_Cmode : LPF based SNR filtering mode on CHRM channel: 0 : gradient LPF [1 1]/2, 1: gradient LPF [2 1 1]/4; 2: gradient LPF [3 3 2]/8; 3: gradient LPF [5 4 4 3]/16; 4 : TNRLPF; 5 : CurLPF3x3_yuv[]; 6: CurLPF3o3_yuv [] 7: CurLPF3x5_yuv[]
10: 8	R/W	0x2	reg_MATNR_SNR_LPF_Ymode : LPF based SNR filtering mode on LUMA channel: 0 : gradient LPF //Bit [1 1]/2, 1: gradient LPF [2 1 1]/4; 2: gradient LPF [3 3 2]/8;3: gradient LPF [5 4 4 3]/16; 4 : TNRLPF; 5 : CurLPF3x3_yuv[]; 6: CurLPF3o3_yuv[] 7: CurLPF3x5_yuv[]
7:4	R/W	0x6	reg_MATNR_SNR_LPF_SADmin3TH : Offset threshold to SAD_min to Discard SAD_min3 corresponding pixel in LPF SNR filtering. (X8 to u8 scale)
3:0	R/W	0x4	reg_MATNR_SNR_LPF_SADmin2TH : Offset threshold to SAD_min to Discard SAD_min2 corresponding pixel in LPF SNR filtering. (X8 to u8 scale)

**Table 9-759 NR2\_MATNR\_SNR\_USF\_GAIN 0x1756**

Bit(s)	R/W	Default	Description
15:8	R/W	0x0	reg_MATNR_SNR_USF_Cgain : Un-sharp (HP) compensate back Chrm portion gain, (norm 64 as 1)
7:0	R/W	0x0	reg_MATNR_SNR_USF_Ygain : Un-sharp (HP) compensate back Luma portion gain, (norm 64 as 1)

**Table 9-760 NR2\_MATNR\_SNR\_EDGE2B 0x1757**

Bit(s)	R/W	Default	Description
15:8	R/W	0x80	reg_MATNR_SNR_Edge2Beta_ofst : U8, Offset for Beta based on Edge.
7:0	R/W	0x10	reg_MATNR_SNR_Edge2Beta_gain : U8. Gain to SAD_min for Beta based on Edge. (norm 16 as 1)

**Table 9-761 NR2\_MATNR\_BETA\_EGAIN 0x1758**

Bit(s)	R/W	Default	Description
15:8	R/W	0x20	reg_MATNR_CBeta_Egain : U8, Gain to Edge based Beta for Chrm channel. (normalized to 32 as 1)
7:0	R/W	0x20	reg_MATNR_YBeta_Egain : U8, Gain to Edge based Beta for Luma channel. (normalized to 32 as 1)

**Table 9-762 NR2\_MATNR\_BETA\_BRT 0x1759**

Bit(s)	R/W	Default	Description
31:28	R/W	0x0	reg_MATNR_beta_BRT_limt_hi : U4, Beta adjustment based on Brightness high side Limit. (X16 to u8 scale)
27:24	R/W	0x0	reg_MATNR_beta_BRT_slop_hi : U4, Beta adjustment based on Brightness high side slope. Normalized to 16 as 1
23:16	R/W	0xa0	reg_MATNR_beta_BRT_thrd_hi : U8, Beta adjustment based on Brightness high threshold.(u8 scale)
15:12	R/W	0x6	reg_MATNR_beta_BRT_limt_lo : U4, Beta adjustment based on Brightness low side Limit. (X16 to u8 scale)
11:8	R/W	0x6	reg_MATNR_beta_BRT_slop_lo : U4, Beta adjustment based on Brightness low side slope. Normalized to 16 as 1
7:0	R/W	0x64	reg_MATNR_beta_BRT_thrd_lo : U8, Beta adjustment based on Brightness low threshold.(u8 scale)

**Table 9-763 NR2\_MATNR\_XBETA\_CFG 0x175a**

Bit(s)	R/W	Default	Description
19:18	R/W	0x0	reg_MATNR_CBeta_use_mode : U2, Beta options (mux) from beta_motion and beta_edge for Chrm channel;
17:16	R/W	0x0	reg_MATNR_YBeta_use_mode : U2, Beta options (mux) from beta_motion and beta_edge for Luma channel;

Bit(s)	R/W	Default	Description
15: 8	R/W	0x0	reg_MATNR_CBeta_Ofst : U8, Offset to Beta for Chrm channel.(after beta_edge and beta_motion mux)
7: 0	R/W	0x0	reg_MATNR_YBeta_Ofst : U8, Offset to Beta for Luma channel.(after beta_edge and beta_motion mux)

**Table 9-764 NR2\_MATNR\_YBETA\_SCL 0x175b**

Bit(s)	R/W	Default	Description
31:24	R/W	0x3c	reg_MATNR_YBeta_scale_min : U8, Final step Beta scale low limit for Luma channel;
23:16	R/W	0xff	reg_MATNR_YBeta_scale_max : U8, Final step Beta scale high limit for Luma channe;
15: 8	R/W	0x20	reg_MATNR_YBeta_scale_gain : U8, Final step Beta scale Gain for Luma channel (normalized 32 to 1);
7 : 0	R/W	0x0	reg_MATNR_YBeta_scale_ofst : S8, Final step Beta scale offset for Luma channel ;

**Table 9-765 NR2\_MATNR\_CBETA\_SCL 0x175c**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_CBeta_scale_min : Final step Beta scale low limit for Chrm channel.Similar to Y
23:16	R/W	0xff	reg_MATNR_CBeta_scale_max : U8, Final step Beta scale high limit for Chrm channel.Similar to Y
15: 8	R/W	0x20	reg_MATNR_CBeta_scale_gain : U8, Final step Beta scale Gain for Chrm channel Similar to Y
7: 0	R/W	0x0	reg_MATNR_CBeta_scale_ofst : S8, Final step Beta scale offset for Chrm channel Similar to Y

**Table 9-766 NR2\_SNR\_MASK 0x175d**

Bit(s)	R/W	Default	Description
20:0	R/W	0x0	SAD_MSK : Valid signal in the 3x7 SAD surface

**Table 9-767 NR2\_SAD2NORM\_LUT0 0x175e**

Bit(s)	R/W	Default	Description
31:24	R/W	0x72	reg_MATNR_SAD2Norm_LUT_3 : SAD convert normal LUT node 3
23:16	R/W	0x92	reg_MATNR_SAD2Norm_LUT_2 : SAD convert normal LUT node 2
15: 8	R/W	0xab	reg_MATNR_SAD2Norm_LUT_1 : SAD convert normal LUT node 1
7: 0	R/W	0xcd	reg_MATNR_SAD2Norm_LUT_0 : SAD convert normal LUT node 0

**Table 9-768 NR2\_SAD2NORM\_LUT1 0x175f**

Bit(s)	R/W	Default	Description
31:24	R/W	0x1c	reg_MATNR_SAD2Norm_LUT_7 : SAD convert normal LUT node 7
23:16	R/W	0x23	reg_MATNR_SAD2Norm_LUT_6 : SAD convert normal LUT node 6
15: 8	R/W	0x31	reg_MATNR_SAD2Norm_LUT_5 : SAD convert normal LUT node 5
7: 0	R/W	0x4f	reg_MATNR_SAD2Norm_LUT_4 : SAD convert normal LUT node 4

**Table 9-769 NR2\_SAD2NORM\_LUT2 0x1760**

Bit(s)	R/W	Default	Description
31:24	R/W	0xf	reg_MATNR_SAD2Norm_LUT_11 : SAD convert normal LUT node 11
23:16	R/W	0x11	reg_MATNR_SAD2Norm_LUT_10 : SAD convert normal LUT node 10
15: 8	R/W	0x13	reg_MATNR_SAD2Norm_LUT_9 : SAD convert normal LUT node 9
7: 0	R/W	0x17	reg_MATNR_SAD2Norm_LUT_8 : SAD convert normal LUT node 8

**Table 9-770 NR2\_SAD2NORM\_LUT3 0x1761**

Bit(s)	R/W	Default	Description
31:24	R/W	0x8	reg_MATNR_SAD2Norm_LUT_15 : SAD convert normal LUT node 15
23:16	R/W	0x9	reg_MATNR_SAD2Norm_LUT_14 : SAD convert normal LUT node 14
15:8	R/W	0xa	reg_MATNR_SAD2Norm_LUT_13 : SAD convert normal LUT node 13
7:0	R/W	0xc	reg_MATNR_SAD2Norm_LUT_12 : SAD convert normal LUT node 12

**Table 9-771 NR2\_EDGE2BETA\_LUT0 0x1762**

Bit(s)	R/W	Default	Description
31:24	R/W	0x80	reg_MATNR_Edge2Beta_LUT_3 : Edge convert beta LUT node 3
23:16	R/W	0xa0	reg_MATNR_Edge2Beta_LUT_2 : Edge convert beta LUT node 2
15: 8	R/W	0xe0	reg_MATNR_Edge2Beta_LUT_1 : Edge convert beta LUT node 1
7: 0	R/W	0xff	reg_MATNR_Edge2Beta_LUT_0 : Edge convert beta LUT node 0

**Table 9-772 NR2\_EDGE2BETA\_LUT1 0x1763**

Bit(s)	R/W	Default	Description
31:24	R/W	0x4	reg_MATNR_Edge2Beta_LUT_7 : Edge convert beta LUT node 7
23:16	R/W	0x10	reg_MATNR_Edge2Beta_LUT_6 : Edge convert beta LUT node 6
15: 8	R/W	0x20	reg_MATNR_Edge2Beta_LUT_5 : Edge convert beta LUT node 5
7: 0	R/W	0x50	reg_MATNR_Edge2Beta_LUT_4 : Edge convert beta LUT node 4

**Table 9-773 NR2\_EDGE2BETA\_LUT2 0x1a64**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_Edge2Beta_LUT_11 : Edge convert beta LUT node 11
23:16	R/W	0x0	reg_MATNR_Edge2Beta_LUT_10 : Edge convert beta LUT node 10
15: 8	R/W	0x0	reg_MATNR_Edge2Beta_LUT_9 : Edge convert beta LUT node 9
7: 0	R/W	0x2	reg_MATNR_Edge2Beta_LUT_8 : Edge convert beta LUT node 8

**Table 9-774 NR2\_EDGE2BETA\_LUT3 0x1765**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_Edge2Beta_LUT_15 : Edge convert beta LUT node 15
23:16	R/W	0x0	reg_MATNR_Edge2Beta_LUT_14 : Edge convert beta LUT node 14
15: 8	R/W	0x0	reg_MATNR_Edge2Beta_LUT_13 : Edge convert beta LUT node 13
7: 0	R/W	0x0	reg_MATNR_Edge2Beta_LUT_12 : Edge convert beta LUT node 12

**Table 9-775 NR2\_MOTION2BETA\_LUT0 0x1766**

Bit(s)	R/W	Default	Description
31:24	R/W	0x20	reg_MATNR_Mot2Beta_LUT_3 : Motion convert beta LUT node 3
23:16	R/W	0x10	reg_MATNR_Mot2Beta_LUT_2 : Motion convert beta LUT node 2
15: 8	R/W	0x4	reg_MATNR_Mot2Beta_LUT_1 : Motion convert beta LUT node 1
7: 0	R/W	0x0	reg_MATNR_Mot2Beta_LUT_0 : Motion convert beta LUT node 0

**Table 9-776 NR2\_MOTION2BETA\_LUT1 0x1767**

Bit(s)	R/W	Default	Description
31:24	R/W	0xc4	reg_MATNR_Mot2Beta_LUT_7 : Motion convert beta LUT node 7
23:16	R/W	0x80	reg_MATNR_Mot2Beta_LUT_6 : Motion convert beta LUT node 6
15: 8	R/W	0x40	reg_MATNR_Mot2Beta_LUT_5 : Motion convert beta LUT node 5
7: 0	R/W	0x30	reg_MATNR_Mot2Beta_LUT_4 : Motion convert beta LUT node 4

**Table 9-777 NR2\_MOTION2BETA\_LUT2 0x1768**

Bit(s)	R/W	Default	Description
31:24	R/W	0xff	reg_MATNR_Mot2Beta_LUT_11 : Motion convert beta LUT node 11
23:16	R/W	0xff	reg_MATNR_Mot2Beta_LUT_10 : Motion convert beta LUT node 10
15: 8	R/W	0xf0	reg_MATNR_Mot2Beta_LUT_9 : Motion convert beta LUT node 9
7: 0	R/W	0xe0	reg_MATNR_Mot2Beta_LUT_8 : Motion convert beta LUT node 8

**Table 9-778 NR2\_MOTION2BETA\_LUT3 0x1769**

Bit(s)	R/W	Default	Description
31:24	R/W	0xff	reg_MATNR_Mot2Beta_LUT_15 : Motion convert beta LUT node 15
23:16	R/W	0xff	reg_MATNR_Mot2Beta_LUT_14 : Motion convert beta LUT node 14
15: 8	R/W	0xff	reg_MATNR_Mot2Beta_LUT_13 : Motion convert beta LUT node 13
7: 0	R/W	0xff	reg_MATNR_Mot2Beta_LUT_12 : Motion convert beta LUT node 12

**Table 9-779 NR2\_MATNR\_MTN\_CTRL 0x176a**

Bit(s)	R/W	Default	Description
25:24	R/W	0x0	reg_MATNR_Vmtn_use_mode : Motion_yuvV channel motion selection mode:0: Vmot;1:Ymot/2 + (Umot+Vmot)/4; 2:Ymot/2 + max(Umot,Vmot)/2; 3: max(Ymot, Umot, Vmot)
21:20	R/W	0x0	reg_MATNR_Umtn_use_mode : Motion_yuvU channel motion selection mode:0: Umot;1:Ymot/2 + (Umot+Vmot)/4; 2:Ymot/2 + max(Umot,Vmot)/2; 3: max(Ymot, Umot, Vmot)
17:16	R/W	0x0	reg_MATNR_Ymtn_use_mode : Motion_yuvLuma channel motion selection mode:0: Ymot, 1: Ymot/2 + (Umot+Vmot)/4; 2: Ymot/2 + max(Umot,Vmot)/2; 3: max(Ymot,Umot, Vmot)
13:12	R/W	0x1	reg_MATNR_mtn_txt_mode : Texture detection mode for adaptive coring of HP motion
9: 8	R/W	0x1	reg_MATNR_mtn_cor_mode : Coring selection mode based on texture detection;
6: 4	R/W	0x8	reg_MATNR_mtn_hpf_mode : video mode of current and previous frame/field for MotHPF_yuv[k] calculation:
2: 0	R/W	0x6	reg_MATNR_mtn_lpf_mode : LPF video mode of current and previous frame/field for MotLPF_yuv[k] calculation:

**Table 9-780 NR2\_MATNR\_MTN\_CTRL2 0x176b**

Bit(s)	R/W	Default	Description
18:16	R/W	0x6	reg_MATNR_iir_BS_Ymode : IIR TNR filter Band split filter mode for Luma LPF result generation (Cur and Prev);
15: 8	R/W	0x40	reg_MATNR_mtnb_alpLP_Cgain : Scale of motion_brthp_uv to motion_brtp_uv, normalized to 32 as 1
7: 0	R/W	0x40	reg_MATNR_mtnb_alpLP_Ygain : Scale of motion_brthp_y to motion_brtp_y, normalized to 32 as 1

**Table 9-781 NR2\_MATNR\_MTN\_COR 0x176c**

Bit(s)	R/W	Default	Description
15:12	R/W	0x3	reg_MATNR_mtn_cor_Cofst : Coring Offset for Chroma Motion.
11: 8	R/W	0x3	reg_MATNR_mtn_cor_Cgain : Gain to texture based coring for Chroma Motion. Normalized to 16 as 1
7: 4	R/W	0x3	reg_MATNR_mtn_cor_Yofst : Coring Offset for Luma Motion.
3: 0	R/W	0x3	reg_MATNR_mtn_cor_Ygain : Gain to texture based coring for Luma Motion. Normalized to 16 as 1



**Table 9-782 NR2\_MATNR\_MTN\_GAIN 0x176d**

Bit(s)	R/W	Default	Description
31:24	R/W	0x40	reg_MATNR_mtn_hp_Cgain : Gain to MotHPF_yuv[k] Chrm channel for motion calculation, normalized to 64 as 1
23:16	R/W	0x40	reg_MATNR_mtn_hp_Ygain : Gain to MotHPF_yuv[k] Luma channel for motion calculation, normalized to 64 as 1
15: 8	R/W	0x40	reg_MATNR_mtn_lp_Cgain : Gain to MotLPF_yuv[k] Chrm channel for motion calculation, normalized to 32 as 1
7: 0	R/W	0x40	reg_MATNR_mtn_lp_Ygain : Gain to MotLPF_yuv[k] Luma channel for motion calculation, normalized to 32 as 1

**Table 9-783 NR2\_MATNR\_DEGHOST 0x176e**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_matnr_deghost_mode : // unsigned , default = 0 0:old_deghost; 1:soft_denoise & strong_deghost; 2:strong_denoise & soft_deghost; 3:strong_denoise & strong_deghost
24:20	R/W	4	reg_matnr_deghost_ygain : // unsigned , default = 4
16:12	R/W	4	reg_matnr_deghost_cgain : // unsigned , default = 4
8	R/W	1	reg_matnr_deghost_en : // unsigned , default = 1 0: disable; 1: enable Enable signal for DeGhost function:0: disable; 1: enable
7: 4	R/W	3	reg_matnr_deghost_cos : // unsigned , default = 3 DeGhost Overshoot margin for UV channel, (X2 to u10 scale)
3: 0	R/W	3	reg_matnr_deghost_yos : // unsigned , default = 3 DeGhost Overshoot margin for Luma channel, (X2 to u10 scale)

**Table 9-784 NR2\_MATNR\_ALPHALP\_LUT0 0x176f**

Bit(s)	R/W	Default	Description
31:24	R/W	0x40	reg_MATNR_AlphaLP_LUT_3 : Matnr low-pass filter alpha LUT node 3
23:16	R/W	0x80	reg_MATNR_AlphaLP_LUT_2 : Matnr low-pass filter alpha LUT node 2
15: 8	R/W	0x80	reg_MATNR_AlphaLP_LUT_1 : Matnr low-pass filter alpha LUT node 1
7: 0	R/W	0x80	reg_MATNR_AlphaLP_LUT_0 : Matnr low-pass filter alpha LUT node 0

**Table 9-785 NR2\_MATNR\_ALPHALP\_LUT1 0x1770**

Bit(s)	R/W	Default	Description
31:24	R/W	0xff	reg_MATNR_AlphaLP_LUT_7 : Matnr low-pass filter alpha LUT node 7
23:16	R/W	0x80	reg_MATNR_AlphaLP_LUT_6 : Matnr low-pass filter alpha LUT node 6
15: 8	R/W	0x50	reg_MATNR_AlphaLP_LUT_5 : Matnr low-pass filter alpha LUT node 5
7: 0	R/W	0x40	reg_MATNR_AlphaLP_LUT_4 : Matnr low-pass filter alpha LUT node 4

**Table 9-786 NR2\_MATNR\_ALPHA\_LP\_LUT2 0x1771**

Bit(s)	R/W	Default	Description
31:24	R/W	0xff	reg_MATNR_AlphaLP_LUT_11 : Matnr low-pass filter alpha LUT node 11
23:16	R/W	0xff	reg_MATNR_AlphaLP_LUT_10 : Matnr low-pass filter alpha LUT node 10
15: 8	R/W	0xff	reg_MATNR_AlphaLP_LUT_9 : Matnr low-pass filter alpha LUT node 9
7: 0	R/W	0xff	reg_MATNR_AlphaLP_LUT_8 : Matnr low-pass filter alpha LUT node 8

**Table 9-787 NR2\_MATNR\_ALPHA\_LP\_LUT3 0x1772**

Bit(s)	R/W	Default	Description
31:24	R/W	0xff	reg_MATNR_AlphaLP_LUT_15 : Matnr low-pass filter alpha LUT node 15
23:16	R/W	0xff	reg_MATNR_AlphaLP_LUT_14 : Matnr low-pass filter alpha LUT node 14
15: 8	R/W	0xff	reg_MATNR_AlphaLP_LUT_13 : Matnr low-pass filter alpha LUT node 13
7: 0	R/W	0xff	reg_MATNR_AlphaLP_LUT_12 : Matnr low-pass filter alpha LUT node 12

**Table 9-788 NR2\_MATNR\_ALPHA\_HP\_LUT0 0x1773**

Bit(s)	R/W	Default	Description
31:24	R/W	0x40	reg_MATNR_AlphaHP_LUT_3 : Matnr high-pass filter alpha LUT node 3
23:16	R/W	0x80	reg_MATNR_AlphaHP_LUT_2 : Matnr high-pass filter alpha LUT node 2
15: 8	R/W	0x80	reg_MATNR_AlphaHP_LUT_1 : Matnr high-pass filter alpha LUT node 1
7: 0	R/W	0x80	reg_MATNR_AlphaHP_LUT_0 : Matnr high-pass filter alpha LUT node 0

**Table 9-789 NR2\_MATNR\_ALPHA\_HP\_LUT1 0x1774**

Bit(s)	R/W	Default	Description
31:24	R/W	0xff	reg_MATNR_AlphaHP_LUT_7 : Matnr high-pass filter alpha LUT node 7
23:16	R/W	0x80	reg_MATNR_AlphaHP_LUT_6 : Matnr high-pass filter alpha LUT node 6
15: 8	R/W	0x50	reg_MATNR_AlphaHP_LUT_5 : Matnr high-pass filter alpha LUT node 5
7: 0	R/W	0x40	reg_MATNR_AlphaHP_LUT_4 : Matnr high-pass filter alpha LUT node 4

**Table 9-790 NR2\_MATNR\_ALPHA\_HP\_LUT2 0x1775**

Bit(s)	R/W	Default	Description
31:24	R/W	0xff	reg_MATNR_AlphaHP_LUT_11 : Matnr high-pass filter alpha LUT node 11
23:16	R/W	0xff	reg_MATNR_AlphaHP_LUT_10 : Matnr high-pass filter alpha LUT node 10
15: 8	R/W	0xff	reg_MATNR_AlphaHP_LUT_9 : Matnr high-pass filter alpha LUT node 9
7: 0	R/W	0xff	reg_MATNR_AlphaHP_LUT_8 : Matnr high-pass filter alpha LUT node 8

**Table 9-791 NR2\_MATNR\_ALPHAHP\_LUT3 0x1776**

Bit(s)	R/W	Default	Description
31:24	R/W	0xff	reg_MATNR_AlphaHP_LUT_15 : Matnr high-pass filter alpha LUT node 15
23:16	R/W	0xff	reg_MATNR_AlphaHP_LUT_14 : Matnr high-pass filter alpha LUT node 14
15: 8	R/W	0xff	reg_MATNR_AlphaHP_LUT_13 : Matnr high-pass filter alpha LUT node 13
7: 0	R/W	0xff	reg_MATNR_AlphaHP_LUT_12 : Matnr high-pass filter alpha LUT node 12

**Table 9-792 NR2\_MATNR\_MTNB\_BRT 0x1777**

Bit(s)	R/W	Default	Description
31:28	R/W	0x0	reg_MATNR_mtnb_BRT_limt_hi : Motion adjustment based on Brightness high side Limit. (X16 to u8 scale)
27:24	R/W	0x0	reg_MATNR_mtnb_BRT_slop_hi : Motion adjustment based on Brightness high side slope. Normalized to 16 as 1
23:16	R/W	0xa0	reg_MATNR_mtnb_BRT_thr_d_hi : Motion adjustment based on Brightness high threshold.(u8 scale)
15:12	R/W	0x6	reg_MATNR_mtnb_BRT_limt_lo : Motion adjustment based on Brightness low side Limit. (X16 to u8 scale)
11: 8	R/W	0x6	reg_MATNR_mtnb_BRT_slop_lo : Motion adjustment based on Brightness low side slope. Normalized to 16 as 1
7: 0	R/W	0x64	reg_MATNR_mtnb_BRT_thr_d_lo : Motion adjustment based on Brightness low threshold.(u8 scale)

**Table 9-793 NR2\_CUE\_MODE 0x1778**

Bit(s)	R/W	Default	Description
31:20	R/W	0x0	Reserved
19	R/W	0x0	Cue2_isabv_org_invert1
18	R/W	0x1	Cue2_valid_condition
17:16	R/W	0x1	Cue2_orgline_ft_sel
15:12	R/W	0x4	Cue2_orgline_ft_alph
11	R/W	0x0	Cue2_isabv_org_invert
10	R/W	0x0	Cue2_iscur_org_invert
9	R/W	0x0	Cue_enable_r : Cue right half frame enable
8	R/W	0x0	Cue_enable_l : Cue left half frame enable
6:4	R/W	0x0	reg_CUE_CON_RPLC_mode : U3, CUE pixel chroma replace mode;
3:0	R/W	0x0	reg_CUE_CHRM_FLT_mode : U4, CUE improvement filter mode,

**Table 9-794 NR2\_CUE\_CON\_MOT\_TH 0x1779**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_CUE_CON_Cmot_thrd2 : U8, Motion Detection threshold of up/down two rows, Chroma channel in Chroma Up-sampling Error (CUE) Detection (tighter).
23:16	R/W	0x0	reg_CUE_CON_Ymot_thrd2 : U8, Motion Detection threshold of up/mid/down three rows, Luma channel in Chroma Up-sampling Error (CUE) Detection (tighter).
15: 8	R/W	0x0	reg_CUE_CON_Cmot_thrd : U8, Motion Detection threshold of up/down two rows, Chroma channel in Chroma Up-sampling Error (CUE) Detection.
7: 0	R/W	0x0	reg_CUE_CON_Ymot_thrd : U8, Motion Detection threshold of up/mid/down three rows, Luma channel in Chroma Up-sampling Error (CUE) Detection.

**Table 9-795 NR2\_CUE\_CON\_DIF0 0x177a**

Bit(s)	R/W	Default	Description
15:8	R/W	0x0	reg_CUE_CON_difP1_thrd : U8, P1 field Intra-Field top/below line chroma difference threshold,
7:0	R/W	0x0	reg_CUE_CON_difCur_thrd : U8, Current Field/Frame Intra-Field up/down line chroma difference threshold,

**Table 9-796 NR2\_CUE\_CON\_DIF1 0x177b**

Bit(s)	R/W	Default	Description
19:16	R/W	0x0	reg_CUE_CON_rate0 : U4, The Krate to decide CUE by relationship between CUE_difIG and CUE_difEG
15: 8	R/W	0x0	reg_CUE_CON_difEG_thrd : U8, Threshold to the difference between current Field/Frame middle line to down line color channel(CUE_difEG).
7: 0	R/W	0x0	reg_CUE_CON_difIG_thrd : U8, Threshold to the difference between P1 field top line to current Field/Frame down line color channel (CUE_difIG).

**Table 9-797 NR2\_CUE\_CON\_DIF2 0x177c**

Bit(s)	R/W	Default	Description
19:16	R/W	0x0	reg_CUE_CON_rate1 : U4, The Krate to decide CUE by relationship between CUE_difnC and CUE_difEC
15: 8	R/W	0x0	reg_CUE_CON_difEC_thrd : U8, Threshold to the difference between current Field/Frame middle line to up line color channel(CUE_difEC).
7: 0	R/W	0x0	reg_CUE_CON_difnC_thrd : U8, Threshold to the difference between P1 field bot line to current Field/Frame up line color channel (CUE_difnC).

**Table 9-798 NR2\_CUE\_CON\_DIF3 0x177d**

Bit(s)	R/W	Default	Description
19:16	R/W	0x0	reg_CUE_CON_rate2 : U4, The Krate to decide CUE by relationship between CUE_difP1 and CUE_difEP1
15: 8	R/W	0x0	reg_CUE_CON_difEP1_thrd : U8, Inter-Field top/below line to current field/frame middle line chroma difference (CUE_difEP1) threshold.
7: 0	R/W	0x0	reg_CUE_CON_difP1_thrd2 : U8, P1 field Intra-Field top/below line chroma difference threshold (tighter),

**Table 9-799 NR2\_CUE\_PRG\_DIF 0x177e**

Bit(s)	R/W	Default	Description
20	R/W	0x0	reg_CUE_PRG_Enable : Enable bit for progressive video CUE detection.If interlace input video,
19:16	R/W	0x0	reg_CUE_PRG_rate : U3, The Krate to decide CUE by relationship between CUE_difCur and (CUE_difEC+CUE_difEG)
15: 8	R/W	0x0	reg_CUE_PRG_difCEG_thrd : U8, Current Frame Intra-Field up-mid and mid-down line chroma difference threshold for progressive video CUE detection,
7: 0	R/W	0x0	reg_CUE_PRG_difCur_thrd : U8, Current Frame Intra-Field up/down line chroma difference threshold,

**Table 9-800 NR2\_CONV\_MODE 0x177f**

Bit(s)	R/W	Default	Description
3:2	R/W	0x0	Conv_c444_mode : The format convert mode about 422 to 444 when data read out line buffer
1:0	R/W	0x0	Conv_c422_mode : the format convert mode about 444 to 422 when data write to line buffer

**Table 9-801 DET3D\_RO\_SPLIT\_HB 0x1780**

Bit(s)	R/W	Default	Description
24	R.O	0x0	RO_Det3D_Split_HB_valid : U1 horizontal LR split border detected valid signal for top half picture
20:16	R.O	0x0	RO_Det3D_Split_HB_pxnum : U5 number of pixels included for the LR split position estimation for top half picture
9: 0	R.O	0x0	RO_Det3D_Split_HB_idxX4 : S10 X4: horizontal pixel shifts of LR split position to the (ColMax/2) for top half picture

**Table 9-802 DET3D\_RO\_SPLIT\_VL 0x1781**

Bit(s)	R/W	Default	Description
24	R.O	0x0	RO_Det3D_Split_VL_valid : U1 horizontal LR split border detected valid signal for top half picture
20:16	R.O	0x0	RO_Det3D_Split_VL_pxnum : U5 number of pixels included for the LR split position estimation for top half picture
9: 0	R.O	0x0	RO_Det3D_Split_VL_idxX4 : S10 X4: horizontal pixel shifts of LR split position to the (ColMax/2) for top half picture

**Table 9-803 DET3D\_RO\_SPLIT\_VR 0x1782**

Bit(s)	R/W	Default	Description
24	R.O	0x0	RO_Det3D_Split_VR_valid : U1 horizontal LR split border detected valid signal for top half picture
20:16	R.O	0x0	RO_Det3D_Split_VR_pxnum : U5 number of pixels included for the LR split position estimation for top half picture
9: 0	R.O	0x0	RO_Det3D_Split_VR_idxX4 : S10 X4: horizontal pixel shifts of LR split position to the (ColMax/2) for top half picture

**Table 9-804 DET3D\_RO\_MAT\_LUMA\_LR 0x1783**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_Luma_LR_score : S2*8 LUMA statistics left right decision score for each band (8bands vertically), it can be -1/0/1:-1: most likely not LR symmetric 0: not sure 1: most likely LR symmetric
7:0	R.O	0x0	RO_Luma_LR_symtc : U1*8 Luma statistics left right pure symmetric for each band (8bands vertically), it can be 0/1: 0: not sure 1: most likely LR is pure symmetric
4:0	R.O	0x0	RO_Luma_LR_sum : S5 Total score of 8x8 Luma statistics for LR like decision, the larger this score, the more confidence that this is a LR 3D video. It is sum of RO_Luma_LR_score[0~7]

**Table 9-805 DET3D\_RO\_MAT\_LUMA\_TB 0x1784**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_Luma_TB_score : S2*8 LUMA statistics Top/Bottom decision score for each band (8bands Horizontally),
7:0	R.O	0x0	RO_Luma_TB_symtc : Luma statistics Top/Bottom pure symmetric for each band (8bands Horizontally),
4:0	R.O	0x0	RO_Luma_TB_sum : Total score of 8x8 Luma statistics for TB like decision,

**Table 9-806 DET3D\_RO\_MAT\_CHRU\_LR 0x1785**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_ChrU_LR_score : S2*8 LUMA statistics left right decision score for each band (8bands vertically),
7:0	R.O	0x0	RO_ChrU_LR_symtc : CHRU statistics left right pure symmetric for each band (8bands vertically),
4:0	R.O	0x0	RO_ChrU_LR_sum : Total score of 8x8 ChrU statistics for LR like decision,

**Table 9-807 DET3D\_RO\_MAT\_CHRU\_TB 0x1786**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_ChrU_TB_score : S2*8 CHRU statistics Top/Bottom decision score for each band (8bands Horizontally)
7:0	R.O	0x0	RO_ChrU_TB_symtc : CHRU statistics Top/Bottom pure symmetric for each band (8bands Horizontally)
4:0	R.O	0x0	RO_ChrU_TB_sum : Total score of 8x8 ChrU statistics for TB like decision

**Table 9-808 DET3D\_RO\_MAT\_CHRV\_LR 0x1787**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_ChrV_LR_score : S2*8 CHRU statistics left right decision score for each band (8bands vertically)
7:0	R.O	0x0	RO_ChrV_LR_symtc : CHRV statistics left right pure symmetric for each band (8bands vertically)
4:0	R.O	0x0	RO_ChrV_LR_sum : Total score of 8x8 ChrV statistics for LR like decision

**Table 9-809 DET3D\_RO\_MAT\_CHRV\_TB 0x1788**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_ChrV_TB_score : CHRV statistics Top/Bottom decision score for each band (8bands Horizontally)
7:0	R.O	0x0	RO_ChrV_TB_symtc : CHRV statistics Top/Bottom pure symmetric for each band (8bands Horizontally)
4:0	R.O	0x0	RO_ChrV_TB_sum : Total score of 8x8 ChrV statistics for TB like decision

**Table 9-810 DET3D\_RO\_MAT\_HEDG\_LR 0x1789**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_Hedg_LR_score : Horizontal Edge statistics left right decision score for each band (8bands vertically)
7:0	R.O	0x0	RO_Hedg_LR_symtc : Horizontal Edge statistics left right pure symmetric for each band (8bands vertically)
4:0	R.O	0x0	RO_Hedg_LR_sum : Total score of 8x8 Hedg statistics for LR like decision

**Table 9-811 DET3D\_RO\_MAT\_HEDG\_TB 0x178a**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_Hedg_TB_score : Horizontal Edge statistics Top/Bottom decision score for each band (8bands Horizontally)
7:0	R.O	0x0	RO_Hedg_TB_symtc : Horizontal Edge statistics Top/Bottom pure symmetric for each band (8bands Horizontally)
4:0	R.O	0x0	RO_Hedg_TB_sum : Total score of 8x8 Hedg statistics for TB like decision

**Table 9-812 DET3D\_RO\_MAT\_VEDG\_LR 0x178b**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_Vedg_LR_score : Vertical Edge statistics left right decision score for each band (8bands vertically)
7:0	R.O	0x0	RO_Vedg_LR_symtc : Vertical Edge statistics left right pure symmetric for each band (8bands vertically)
4:0	R.O	0x0	RO_Vedg_LR_sum : Total score of 8x8 Vedg statistics for LR like decision

**Table 9-813 DET3D\_RO\_MAT\_VEDG\_TB 0x178c**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_Vedg_TB_score : Vertical Edge statistics Top/Bottom decision score for each band (8bands Horizontally)
7:0	R.O	0x0	RO_Vedg_TB_symtc : Vertical Edge statistics Top/Bottom pure symmetric for each band (8bands Horizontally)
4:0	R.O	0x0	RO_Vedg_TB_sum : Total score of 8x8 Vedg statistics for TB like decision

**Table 9-814 DET3D\_RO\_MAT\_MOTN\_LR 0x178d**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_Motn_LR_score : Motion statistics left right decision score for each band (8bands vertically)
7:0	R.O	0x0	RO_Motn_LR_symtc : Motion statistics left right pure symmetric for each band (8bands vertically)
4:0	R.O	0x0	RO_Motn_LR_sum : Total score of 8x8 Motion statistics for LR like decision

**Table 9-815 DET3D\_RO\_MAT\_MOTN\_TB 0x178e**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_Motn_TB_score : Motion statistics Top/Bottom decision score for each band (8bands Horizontally)
7:0	R.O	0x0	RO_Motn_TB_symtc : Motion statistics Top/Bottom pure symmetric for each band (8bands Horizontally)
4:0	R.O	0x0	RO_Motn_TB_sum : Total score of 8x8 Motion statistics for TB like decision



**Table 9-816 DET3D\_RO\_FRM\_MOTN 0x178f**

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_Det3D_Frame_Motion : U16 frame based motion value sum for still image decision in FW. mat ram read enter addr

DET3D\_RAMRD\_ADDR\_PORT 0x179a

DET3D\_RAMRD\_DATA\_PORT 0x179b

**Table 9-817 NR2\_CFR\_PARA\_CFG0 0x179c**

Bit(s)	R/W	Default	Description
8	R/W	0x0	reg_CFR_CurDif_luma_mode : Current Field Top/Bottom line Luma difference calculation mode
7:6	R/W	0x0	reg_MACFR_frm_phase : U2 This will be a field based phase register that need to be set by FW phase to phase: this will be calculated based on dbdr_phase of the specific line of this frame. u1 : dbdr_phase=1, center line is DB in current line; dbdr_phase=2, center line is Dr in current line;
5:4	R/W	0x0	reg_CFR_CurDif_tran_mode : U2 Current Field Top/Bot line Luma/Chroma transition level calculation mode,
3:2	R/W	0x0	reg_CFR_alpha_mode : U2 Alpha selection mode for CFR block from curAlp and motAlp i.e. 0: motAlp; 1: (motAlp+curAlp)/2; 2: min(motAlp,curAlp); 3: max(motAlp,curAlp);
1:0	R/W	0x0	reg_CFR_Motion_Luma_mode : U2 LumaMotion Calculation mode for MA-CFR. 0: top/bot Lumma motion; 1: middle Luma Motion 2: top/bot + middle motion; 3: max(top/tot motion, middle motion)

**Table 9-818 NR2\_CFR\_PARA\_CFG1 0x179d**

Bit(s)	R/W	Default	Description
23:16	R/W	0x0	reg_CFR_alpha_gain : gain to map muxed curAlp and motAlp to alpha that will be used for final blending.
15: 8	R/W	0x0	reg_CFR_Motion_ofst : Offset to Motion to calculate the motAlp, e.g:motAlp=reg_CFR_Motion_ofst- Motion;This register can be seen as the level of motion that we consider it at moving.
7: 0	R/W	0x0	reg_CFR_CurDif_gain : gain to CurDif to map to alpha, normalized to 32;

**Table 9-819 NR3\_MODE 0x2ff0**

Bit(s)	R/W	Default	Description
5	R/W	0x0	reg_3dnr_nr3_vtxt_mode ;
4	R/W	0x0	reg_3dnr_nr3_cbyy_ignor_coop ; ; // u1: ignore coop condition for cbyy motion decision
3	R/W	0x0	reg_3dnr_nr3_ybyc_ignor_cnoop ; ; // u1: ignore cnoop condition for ybyc motion decision
2:0	R/W	0x3	reg_3dnr_nr3_suremot_txt_mode ; ; // u3: 0: cur, 1:p2; 2: (cur+p2)/2; 3/up: min (cur,p2)

**Table 9-820 NR3\_COOP\_PARA 0x2ff1**

Bit(s)	R/W	Default	Description
21:20	R/W	0x2	reg_3dnr_nr3_coop_mode : ; // u2 0 original pixel 1: [1 2 1]/4 lpf; 2: [1 2 2 1]/8; 3: 3x3 lpf
19:16	R/W	0x8	reg_3dnr_nr3_coop_ratio : ; // u4 cur and p2 color oop decision ratio: (avg1<(MAX(sat0,sat2)*ratio/8 + ofst));
15:8	R/W	0x0	reg_3dnr_nr3_coop_ofset : ; // s8 cur and p2 color oop decision ofst: (avg1<(MAX(sat0,sat2)*ratio/8 + ofst));
7:0	R/W	0x0	reg_3dnr_nr3_coop_sat_thrd : ; // u8 cur and p2 color oop decision min(sat0, sat1) threshold;

**Table 9-821 NR3\_CNOOP\_GAIN 0x2ff2**

Bit(s)	R/W	Default	Description
23:20	R/W	0x8	reg_3dnr_nr3_cnoop_ratio0 : ; // u4 cur and p2 color noop decision ratio0: (avg1<(MAX(sat0,sat2)*ratio0/8 + ofst0));
19:16	R/W	0x8	reg_3dnr_nr3_cnoop_ratio1 : ; // u4 cur and p2 color noop decision ratio1: (dif1<(MIN(sat0,sat2)*ratio1/8 + ofst1));
15:8	R/W	0x19	reg_3dnr_nr3_cnoop_ofset0 : ; // s8 cur and p2 color noop decision ofset0: (avg1<(MAX(sat0,sat2)*ratio0/8 + ofst0));
7:0	R/W	0x0	reg_3dnr_nr3_cnoop_ofset1 : ; // s8 cur and p2 color noop decision ofset1: (dif1<(MIN(sat0,sat2)*ratio1/8 + ofst1));

**Table 9-822 NR3\_YMOT\_PARA 0x2ff3**

Bit(s)	R/W	Default	Description
19	R/W	0x1	reg_3dnr_nr3_ymot_only_en : ; // u1: enable signal for ignor chroma motion: (ytxt& coop)
18	R/W	0x1	reg_3dnr_nr3_ymot_only_cmtmode : ; // u1: 0: cmot=ymot; 1: cmot = MIN(ymot, cmot)
17:16	R/W	0x0	reg_3dnr_nr3_ymot_only_txtmode : ; // u2: 0, min(txt0,txt2); 1, max(txt0,txt2);2, (txt0+txt2)/2; 3: sat(txt0, txt2)
15:8	R/W	0xa	reg_3dnr_nr3_ymot_only_txtthrd : ; // u8: threshold to luma texture to decide use ymot only
7:0	R/W	0x1e	reg_3dnr_nr3_ymot_only_motthrd : ; // u8: threshold to luma motion to decide use ymot only

**Table 9-823 NR3\_CMOT\_PARA 0x2ff4**

Bit(s)	R/W	Default	Description
19	R/W	0x1	reg_3dnr_nr3_cmot_only_en : ; // u1: enable signal for ignor luma motion: (ctxt &cnoop)
18	R/W	0x0	reg_3dnr_nr3_cmot_only_ytmotmode : ; // u1: 0: ymot=cmot+ymot/4; 1: ymot = MIN(ymot, cmot)
17:16	R/W	0x0	reg_3dnr_nr3_cmot_only_txtmode : ; // u2: 0, min(txt0,txt2); 1, max(txt0,txt2);2, (txt0+txt2)/2; 3: sat(txt0, txt2)

Bit(s)	R/W	Default	Description
15:8	R/W	0x14	reg_3dnr_nr3_cmot_only_txtthrd : ; // u8: threshold to chroma texture to decide use cmot only
7:0	R/W	0xf	reg_3dnr_nr3_cmot_only_motthrd : ; // u8: threshold to chroma motion to decide use cmot only

Table 9-824 NR3\_SUREMOT\_YGAIN 0x2ff5

Bit(s)	R/W	Default	Description
31:24	R/W	0x10	reg_3dnr_nr3_suremot_dec_yrate : ; // u8: (norm 16)lpfMot>(dec_rate*txt +ofst) then force lpfMot*frg_gain+frg_ofset
23:16	R/W	0xc	reg_3dnr_nr3_suremot_dec_yofst : ; // u8: lpfMot>(dec_rate*txt +ofst) then force lpfMot*frg_gain+frg_ofset
15:8	R/W	0x40	reg_3dnr_nr3_suremot_frc_ygain : ; // u8: (norm 8)lpfMot>(dec_rate*txt +ofst) then force lpfMot*frg_gain+frg_ofset
7:0	R/W	0x14	reg_3dnr_nr3_suremot_frc_yofst : ; // u8: lpfMot>(dec_rate*txt +ofst) then force lpfMot*frg_gain+frg_ofset

Table 9-825 NR3\_SUREMOT\_CGAIN 0x2ff6

Bit(s)	R/W	Default	Description
31:24	R/W	0x22	reg_3dnr_nr3_suremot_dec_crate : ; // u8: (norm 16)lpfMot>(dec_rate*txt +ofst) then force lpfMot*frg_gain+frg_ofset
23:16	R/W	0x26	reg_3dnr_nr3_suremot_dec_cofst : ; // u8: lpfMot>(dec_rate*txt +ofst) then force lpfMot*frg_gain+frg_ofset
15:8	R/W	0x40	reg_3dnr_nr3_suremot_frc_cgain : ; // u8: (norm 8)lpfMot>(dec_rate*txt +ofst) then force lpfMot*frg_gain+frg_ofset
7:0	R/W	0x14	reg_3dnr_nr3_suremot_frc_cofst : ; // u8: lpfMot>(dec_rate*txt +ofst) then force lpfMot*frg_gain+frg_ofset

### 9.2.3.10 LBUF Registers

Table 9-826 LBUF\_TOP\_CTRL 0xff00bffc

Bit(s)	R/W	Default	Description
25:20	R/W	6'd0	gate clk control of line buf . LBUF_TOP_CTRL[25:24] is the clk control of current line linebuffer. LBUF_TOP_CTRL[23:22] is the clk control of previous one line.linebuffer, LBUF_TOP_CTRL[21:20] is the clk control of previous two line linebuffer.
17	R/W	1'b1	lbuf_fmt444_mode; format of data store in linebuf ,high mean store 444 data into linebuf
16	R/W	1'b1	lbuf_line5_mode;Store 5 line or 3 lines in linebuf ,high means 5 lines
12:0	R/W	13'd342	pre_lbuf_size: size of linebuf

### 9.2.3.11 DI\_SCALE Registers

#### Register Address

- DI\_SCO\_FIFO\_CTRL 0xff00dd38
- DI\_SC\_TOP\_CTRL 0xff00dd3c
- DI\_SC\_DUMMY\_DATA 0xff00dd40
- DI\_SC\_LINE\_IN\_LENGTH 0xff00dd44
- DI\_SC\_PIC\_IN\_HEIGHT 0xff00dd48
- DI\_SC\_COEF\_IDX 0xff00dd4c
- DI\_SC\_COEF 0xff00dd50
- DI\_VSC\_REGION12\_STARTP 0xff00dd54
- DI\_VSC\_REGION34\_STARTP 0xff00dd58
- DI\_VSC\_REGION4\_ENDP 0xff00dd5c
- DI\_VSC\_START\_PHASE\_STEP 0xff00dd60
- DI\_VSC\_REGION0\_PHASE\_SLOPE 0xff00dd64
- DI\_VSC\_REGION1\_PHASE\_SLOPE 0xff00dd68
- DI\_VSC\_REGION3\_PHASE\_SLOPE 0xff00dd6c
- DI\_VSC\_REGION4\_PHASE\_SLOPE 0xff00dd70
- DI\_VSC\_PHASE\_CTRL 0xff00dd74
- DI\_VSC\_INI\_PHASE 0xff00dd78
- DI\_HSC\_REGION12\_STARTP 0xff00dd80
- DI\_HSC\_REGION34\_STARTP 0xff00dd84
- DI\_HSC\_REGION4\_ENDP 0xff00dd88
- DI\_HSC\_START\_PHASE\_STEP 0xff00dd8c
- DI\_HSC\_REGION0\_PHASE\_SLOPE 0xff00dd90
- DI\_HSC\_REGION1\_PHASE\_SLOPE 0xff00dd94
- DI\_HSC\_REGION3\_PHASE\_SLOPE 0xff00dd98
- DI\_HSC\_REGION4\_PHASE\_SLOPE 0xff00dd9c
- DI\_HSC\_PHASE\_CTRL 0xff00dda0
- DI\_SC\_MISC 0xff00dda4
- DI\_HSC\_PHASE\_CTRL1 0xff00dda8
- DI\_HSC\_INI\_PAT\_CTRL 0xff00ddac
- DI\_SC\_GCLK\_CTRL 0xff00ddb0
- DI\_SC\_HOLD\_LINE 0xff00ddb4
- DI\_HDR\_IN\_HSIZE 0xff00ddb8
- DI\_HDR\_IN\_VSIZE 0xff00ddbc
- DI\_HDR2\_CTRL 0xff00ddc0
- DI\_HDR2\_CLK\_GATE 0xff00ddc4
- DI\_HDR2\_MATRIXI\_COEF00\_01 0xff00ddc8
- DI\_HDR2\_MATRIXI\_COEF02\_10 0xff00ddcc
- DI\_HDR2\_MATRIXI\_COEF11\_12 0xff00ddd0
- DI\_HDR2\_MATRIXI\_COEF20\_21 0xff00ddd4
- DI\_HDR2\_MATRIXI\_COEF22 0xff00ddd8
- DI\_HDR2\_MATRIXI\_COEF30\_31 0xff00ddd8

- DI\_HDR2\_MATRIXI\_COEF32\_40 0xff00dde0
- DI\_HDR2\_MATRIXI\_COEF41\_42 0xff00dde4
- DI\_HDR2\_MATRIXI\_OFFSET0\_1 0xff00dde8
- DI\_HDR2\_MATRIXI\_OFFSET2 0xff00ddec
- DI\_HDR2\_MATRIXI\_PRE\_OFFSET0\_1 0xff00ddf0
- DI\_HDR2\_MATRIXI\_PRE\_OFFSET2 0xff00ddf4
- DI\_HDR2\_MATRIXO\_COEF00\_01 0xff00ddf8
- DI\_HDR2\_MATRIXO\_COEF02\_10 0xff00ddfc
- DI\_HDR2\_MATRIXO\_COEF11\_12 0xff00de00
- DI\_HDR2\_MATRIXO\_COEF20\_21 0xff00de04
- DI\_HDR2\_MATRIXO\_COEF22 0xff00de08
- DI\_HDR2\_MATRIXO\_COEF30\_31 0xff00de0c
- DI\_HDR2\_MATRIXO\_COEF32\_40 0xff00de10
- DI\_HDR2\_MATRIXO\_COEF41\_42 0xff00de14
- DI\_HDR2\_MATRIXO\_OFFSET0\_1 0xff00de18
- DI\_HDR2\_MATRIXO\_OFFSET2 0xff00de1c
- DI\_HDR2\_MATRIXO\_PRE\_OFFSET0\_1 0xff00de20
- DI\_HDR2\_MATRIXO\_PRE\_OFFSET2 0xff00de24
- DI\_HDR2\_MATRIXI\_CLIP 0xff00de28
- DI\_HDR2\_MATRIXO\_CLIP 0xff00de2c
- DI\_HDR2\_CGAIN\_OFFT 0xff00de30
- DI\_EOTF\_LUT\_ADDR\_PORT 0xff00de38
- DI\_EOTF\_LUT\_DATA\_PORT 0xff00de3c
- DI\_OETF\_LUT\_ADDR\_PORT 0xff00de40
- DI\_OETF\_LUT\_DATA\_PORT 0xff00de44
- DI\_CGAIN\_LUT\_ADDR\_PORT 0xff00de48
- DI\_CGAIN\_LUT\_DATA\_PORT 0xff00de4c
- DI\_HDR2\_CGAIN\_COEF0 0xff00de50
- DI\_HDR2\_CGAIN\_COEF1 0xff00de54
- DI\_OGAIN\_LUT\_ADDR\_PORT 0xff00de58
- DI\_OGAIN\_LUT\_DATA\_PORT 0xff00de5c
- DI\_HDR2\_ADPS\_CTRL 0xff00de60
- DI\_HDR2\_ADPS\_ALPHA0 0xff00de64
- DI\_HDR2\_ADPS\_ALPHA1 0xff00de68
- DI\_HDR2\_ADPS\_BETA0 0xff00de6c
- DI\_HDR2\_ADPS\_BETA1 0xff00de70
- DI\_HDR2\_ADPS\_BETA2 0xff00de74
- DI\_HDR2\_ADPS\_COEF0 0xff00de78
- DI\_HDR2\_ADPS\_COEF1 0xff00de7c
- DI\_HDR2\_GMUT\_CTRL 0xff00de80
- DI\_HDR2\_GMUT\_COEF0 0xff00de84
- DI\_HDR2\_GMUT\_COEF1 0xff00de88
- DI\_HDR2\_GMUT\_COEF2 0xff00de8c
- DI\_HDR2\_GMUT\_COEF3 0xff00de90

- DI\_HDR2\_GMUT\_COEF4 0xff00de94
- DI\_HDR2\_PIPE\_CTRL1 0xff00de98
- DI\_HDR2\_PIPE\_CTRL2 0xff00de9c
- DI\_HDR2\_PIPE\_CTRL3 0xff00dea0
- DI\_HDR2\_PROC\_WIN1 0xff00dea4
- DI\_HDR2\_PROC\_WIN2 0xff00dea8
- DI\_HDR2\_MATRIXI\_EN\_CTRL 0xff00deac
- DI\_HDR2\_MATRIXO\_EN\_CTRL 0xff00deb0

## Register Description

**Table 9-827 DI\_SCO\_FIFO\_CTRL 0x374e**

Bit(s)	R/W	Default	Description
28:0	R/W	0x0	sco_fifo_ctrl

**Table 9-828 DI\_SC\_TOP\_CTRL 0x374f**

Bit(s)	R/W	Default	Description
31	R.O	0	prog_interlace : no use // unsigned , default = 0x0
30	R/W	0	path_sel : 1 : di pre scaler for nr inp 0: di post scaler // unsigned , default = 0x0
29	R/W	0	go_field_sel : 1 : di pre go field 0: di post go filed // unsigned , default = 0x0
28	R/W	0	sw_resets : // unsigned , default = 0x0
27	R/W	0	pps_dummy_data_mode : 1: use low 8 bits 0: use high 8 bits // unsigned , default = 0x0
26	R/W	0	field_inv : field reverse// unsigned , default = 0x0
25	R/W	0	reg_field : no use// unsigned , default = 0x0
5:4	R/W	0	hdr_gclk_ctrl : 01 : no cbus clock for hdr other: free cbus clock for hdr // unsigned , default = 0x0
1	R/W	0	reg_gclk_ctrl : 1 : free clock for di scaler register 0: auto gate for di scalercbus // unsigned , default = 0x0

**Table 9-829 DI\_SC\_DUMMY\_DATA 0x3750**

Bit(s)	R/W	Default	Description
29:20	R/W	0	VD1_SC_Y : // unsigned , default = 0x10,dummy data used in the VD1 scaler,according VPP_DOLBY_CTRL[17] 1:set 8bit value 2:set 10bit value
19:10	R/W	0	VD1_SC_CB : // unsigned , default = 0x80,dummy data used in the VD1 scaler,according VPP_DOLBY_CTRL[17] 1:set 8bit value 2:set 10bit value
9 :0	R/W	0	VD1_SC_CR : // unsigned , default = 0x80,dummy data used in the VD1 scaler,according VPP_DOLBY_CTRL[17] 1:set 8bit value 2:set 10bit value

**Table 9-830 DI\_SC\_LINE\_IN\_LENGTH 0x3751**

Bit(s)	R/W	Default	Description
13:0	R/W	14	line_in_length : // unsigned , default = 14'd1920,VD1 scaler input hsize

**Table 9-831 DI\_SC\_PIC\_IN\_HEIGHT 0x3752**

Bit(s)	R/W	Default	Description
12:0	R/W	0x1fff	line_in_height : // unsigned , default = 13'h1fff,VD1 scaler input vsize

**Table 9-832 DI\_SC\_COEF\_IDX 0x3753**

Bit(s)	R/W	Default	Description
15	R/W	0	index_inc : // unsigned , default = 0x0 ,index increment, if bit9 = 1 then (0: index increase 1, 1: index increase 2) else (index increase 2)
14	R/W	0	rd_cbus_coef_en : // unsigned , default = 0x0 ,1: read coef through cbus enable, just for debug purpose in case when we wanna check the coef in ram in correct or not
13	R/W	0	vf_sep_coef_en : // unsigned , default = 0x0 ,if true, vertical separated coef enable
9	R/W	0	high_reso_en : // unsigned , default = 0x0 ,if true, use 9bit resolution coef, other use 8bit resolution coef
8:7	R/W	0	type_index : // unsigned , default = 0x0 ,type of index, 00: vertical coef, 01: vertical chroma coef, 10: horizontal coef, 11: reserved
6:0	R/W	0	coef_index : // unsigned , default = 0x0 ,coef index

**Table 9-833 DI\_SC\_COEF 0x3754**

Bit(s)	R/W	Default	Description
31:24	R/W	0	coef0 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter
23:16	R/W	0	coef1 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter
15:8	R/W	0	coef2 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter
7 :0	R/W	0	coef3 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter

**Table 9-834 DI\_VSC\_REGION12\_STARTP 0x3755**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region1_startp : //unsigned , default = 0 ,region1 startp
12:0	R/W	0	region2_startp : //unsigned , default = 0 ,region2 startp

**Table 9-835 DI\_VSC\_REGION34\_STARTP 0x3756**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region3_startp : //unsigned , default = 0x0438,region3 startp
12:0	R/W	0	region4_startp : //unsigned , default = 0x0438,region4 startp

**Table 9-836 DI\_VSC\_REGION4\_ENDP 0x3757**

Bit(s)	R/W	Default	Description
12:0	R/W	13	region4_endp : //unsigned , default = 13'd1079 ,region4 endp

**Table 9-837 DI\_VSC\_START\_PHASE\_STEP 0x3758**

Bit(s)	R/W	Default	Description
27:24	R/W	1	integer_part : //unsigned , default = 1,vertical start phase step, (source/dest)* (2^24),integer part of step
23:0	R/W	0	fraction_part : //unsigned , default = 0,vertical start phase step, (source/dest)* (2^24),fraction part of step

**Table 9-838 DI\_VSC\_REGION0\_PHASE\_SLOPE 0x3759**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,vertical scaler region0 phase slope, region0 phase slope

**Table 9-839 DI\_VSC\_REGION1\_PHASE\_SLOPE 0x375a**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region1_phase_slope : //signed , default = 0,region1 phase slope

**Table 9-840 DI\_VSC\_REGION3\_PHASE\_SLOPE 0x375b**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region3_phase_slope : //signed , default = 0,region3 phase slope

**Table 9-841 DI\_VSC\_REGION4\_PHASE\_SLOPE 0x375c**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region4_phase_slope : //signed , default = 0,region4 phase slope

**Table 9-842 DI\_VSC\_PHASE\_CTRL 0x375d**

Bit(s)	R/W	Default	Description
18:17	R/W	0	vsc_double_line_mode : //unsigned , default = 0, double line mode, input/output line width of vscler becomes 2X, so only 2 line buffer in this case, use for 3D line by line interleave scaling bit1 true, double the input width and half input height, bit0 true, change line buffer 2 lines instead of 4 lines
16	R.O	0	prog_interlace : //unsigned , default = 0,0: progressive output, 1: interlace output
15	R/W	0	vsc_bot_l0_out_en : //unsigned , default = 0,vertical scaler output line0 in advance or not for bottom field
14:13	R/W	1	vsc_bot_rpt_l0_num : //unsigned , default = 1,vertical scaler initial repeat line0 number for bottom field



Bit(s)	R/W	Default	Description
11:8	R/W	4	vsc_bot_ini_rcv_num : //unsigned , default = 4,vertical scaler initial receiving number for bottom field
7	R/W	0	vsc_top_l0_out_en : //unsigned , default = 0,vertical scaler output line0 in advance or not for top field
6:5	R/W	1	vsc_top_rpt_l0_num : //unsigned , default = 1,vertical scaler initial repeat line0 number for top field
3:0	R/W	4	vsc_top_ini_rcv_num : //unsigned , default = 4,vertical scaler initial receiving number for top field

**Table 9-843 DI\_VSC\_INI\_PHASE 0x375e**

Bit(s)	R/W	Default	Description
31:16	R/W	0	vertical scaler field initial phase for bottom field
15:0	R/W	0	vertical scaler field initial phase for top field

**Table 9-844 DI\_HSC\_REGION12\_STARTP 0x3760**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region1_startp : //unsigned , default = 0,region1 startp
12:0	R/W	0	region2_startp : //unsigned , default = 0,region2 startp

**Table 9-845 DI\_HSC\_REGION34\_STARTP 0x3761**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region3 : startp //unsigned , default = 0x780,region3 startp
12:0	R/W	0	region4 : startp //unsigned , default = 0x780,region4 startp

**Table 9-846 DI\_HSC\_REGION4\_ENDP 0x3762**

Bit(s)	R/W	Default	Description
12:0	R/W	13	region4 : startp //unsigned , default = 13'd1919,region4 startp

**Table 9-847 DI\_HSC\_START\_PHASE\_STEP 0x3763**

Bit(s)	R/W	Default	Description
27:24	R/W	1	integer_part : //unsigned , default = 1, integer part of step
23:0	R/W	0	fraction_part : //unsigned , default = 0, fraction part of step

**Table 9-848 DI\_HSC\_REGION0\_PHASE\_SLOPE 0x3764**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region0 phase slope

**Table 9-849 DI\_HSC\_REGION1\_PHASE\_SLOPE 0x3765**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region1 phase slope

**Table 9-850 DI\_HSC\_REGION3\_PHASE\_SLOPE 0x3766**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region3 phase slope

**Table 9-851 DI\_HSC\_REGION4\_PHASE\_SLOPE 0x3767**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region4 phase slope

**Table 9-852 DI\_HSC\_PHASE\_CTRL 0x3768**

Bit(s)	R/W	Default	Description
22:21	R/W	1	hsc_rpt_p0_num0 : //unsigned , default = 1 ,horizontal scaler initial repeat pixel0 number0
19:16	R/W	4	hsc_ini_rcv_num0 : //unsigned , default = 4 ,horizontal scaler initial receiving number0
15:0	R/W	0	hsc_ini_phase0 : //unsigned , default = 0 ,horizontal scaler top field initial phase0

**Table 9-853 DI\_SC\_MISC 0x3769**

Bit(s)	R/W	Default	Description
22	R/W	0	hsc_len_div2_en : //unsigned , default = 0 ,if true, divide VSC line length 2 as the HSC input length, othwise VSC length length is the same as the VSC line length just for special usage, more flexibility
21	R/W	0	lbuf_mode : //unsigned , default = 0 ,if true, prevsc uses lin buffer, otherwise prevsc does not use line buffer, it should be same as prevsc_en
20	R/W	0	prehsc_en : //unsigned , default = 0 ,prehsc_en
19	R/W	0	prevsc_en : //unsigned , default = 0 ,prevsc_en
18	R/W	0	vsc_en : //unsigned , default = 0 ,vsc_en
17	R/W	0	hsc_en : //unsigned , default = 0 ,hsc_en
16	R/W	0	sc_top_en : //unsigned , default = 0 ,scale_top_en
15	R/W	0	sc_vd_en : //unsigned , default = 0 ,video1 scale out enable
12	R/W	1	hsc_nonlinear_4region_en : //unsigned , default = 1 ,if true, region0,region4 are nonlinear regions, otherwise they are not scaling regions, for horizontal scaler
10:8	R/W	0	hsc_bank_length : //unsigned , default = 0 ,horizontal scaler bank length
5	R/W	4	vsc_phase_field_mode : //unsigned , default = 4 ,vertical scaler phase field mode, if true, disable the opposite parity line output, more bandwidth needed if output 1080i

Bit(s)	R/W	Default	Description
4	R/W	0	vsc_nonlinear_4region_en : //unsigned , default = 0 ,if true, region0,region4 are nonlinear regions, otherwise they are not scaling regions, for vertical scaler
2:0	R/W	4	vsc_bank_length : //unsigned , default = 4 ,vertical scaler bank length

Table 9-854 DI\_HSC\_PHASE\_CTRL1 0x376a

Bit(s)	R/W	Default	Description
22:21	R/W	1	hsc_rpt_p0_num0 : //unsigned , default = 1 ,horizontal scaler initial repeat pixel0 number0
19:16	R/W	4	hsc_ini_rcv_num0 : //unsigned , default = 4 ,horizontal scaler initial receiving number0
15:0	R/W	0	hsc_ini_phase0 : //unsigned , default = 0 ,horizontal scaler top field initial phase0

Table 9-855 VPP\_SC\_MISC 0x1D19

Bit(s)	R/W	Default	Description
22	R/W	0	hsc_len_div2_en : //unsigned , default = 0 ,if true, divide VSC line length 2 as the HSC input length, otherwise VSC length length is the same as the VSC line length just for special usage, more flexibility
21	R/W	0	lbuf_mode : //unsigned , default = 0 ,if true, prevsc uses lin buffer, otherwise prevsc does not use line buffer, it should be same as prevsc_en
20	R/W	0	prehsc_en : //unsigned , default = 0 ,prehsc_en
19	R/W	0	prevsc_en : //unsigned , default = 0 ,prevsc_en
18	R/W	0	vsc_en : //unsigned , default = 0 ,vsc_en
17	R/W	0	hsc_en : //unsigned , default = 0 ,hsc_en
16	R/W	0	sc_top_en : //unsigned , default = 0 ,scale_top_en
15	R/W	0	sc_vd_en : //unsigned , default = 0 ,video1 scale out enable
12	R/W	1	hsc_nonlinear_4region_en : //unsigned , default = 1 ,if true, region0,region4 are nonlinear regions, otherwise they are not scaling regions, for horizontal scaler
10:8	R/W	0	hsc_bank_length : //unsigned , default = 0 ,horizontal scaler bank length
5	R/W	4	vsc_phase_field_mode : //unsigned , default = 4 ,vertical scaler phase field mode, if true, disable the opposite parity line output, more bandwidth needed if output 1080i
4	R/W	0	vsc_nonlinear_4region_en : //unsigned , default = 0 ,if true, region0,region4 are nonlinear regions, otherwise they are not scaling regions, for vertical scaler
2:0	R/W	4	vsc_bank_length : //unsigned , default = 4 ,vertical scaler bank length

Table 9-856 DI\_HSC\_INI\_PAT\_CTRL 0x376b

Bit(s)	R/W	Default	Description
31:24	R/W	0	prehsc_pattern : //unsigned , default = 0 ,prehsc pattern, each pattern 1 bit, from lsb -> msb
22:20	R/W	0	prehsc_pat_star : //unsigned , default = 0 ,prehsc pattern start

Bit(s)	R/W	Default	Description
18:16	R/W	0	prehsc_pat_end : //unsigned , default = 0, prehsc pattern end
15:8	R/W	0	hsc_pattern : //unsigned , default = 0, hsc pattern, each pattern 1 bit, from lsb -> msb
6:4	R/W	0	hsc_pat_start : //unsigned , default = 0, hsc pattern start
2:0	R/W	0	hsc_pat_end : //unsigned , default = 0, hsc pattern end

**Table 9-857 DI\_SC\_GCLK\_CTRL 0x376c**

Bit(s)	R/W	Default	Description
15:0	R/W	0	vpp_sc_gclk_ctrl : //unsigned , default = 0,

**Table 9-858 DI\_SC\_HOLD\_LINE 0x376d**

Bit(s)	R/W	Default	Description
31:0	R/W	0	sc_hold_line : //unsigned , default = 0,

**Table 9-859 DI\_HDR\_IN\_HSIZE 0x376e**

Bit(s)	R/W	Default	Description
12:0	R/W	0	hdr input h size

**Table 9-860 DI\_HDR\_IN\_VSIZE 0x376f**

Bit(s)	R/W	Default	Description
12:0	R/W	0	hdr input v size

**Table 9-861 DI\_HDR2\_CTRL 0x3770**

Bit(s)	R/W	Default	Description
20:18	R/W	0	reg_din_swap : // unsigned , default = 0
17	R/W	0	reg_out_fmt : // unsigned , default = 0
16	R/W	0	reg_only_mat : // unsigned , default = 0
13	R/W	0	reg_VDIN0_HDR2_top_en : // unsigned , default = 0
12	R/W	1	reg_cgain_mode : // unsigned , default = 1
7: 6	R/W	1	reg_gmut_mode : // unsigned , default = 1
5	R/W	0	reg_in_shift : // unsigned , default = 0
4	R/W	1	reg_in_fmt : // unsigned , default = 1
3	R/W	1	reg_eo_enable : // unsigned , default = 1
2	R/W	1	reg_oe_enable : // unsigned , default = 1

Bit(s)	R/W	Default	Description
1	R/W	1	reg_ogain_enable : // unsigned , default = 1
0	R/W	1	reg_cgain_enable : // unsigned , default = 1

Table 9-862 DI\_HDR2\_CLK\_GATE 0x3771

Bit(s)	R/W	Default	Description
31:30	R/W	0	clk_tm : gate clock ctrl (main clock) // unsigned , default = 0
29:28	R/W	0	output : matrix clock gate ctrl // unsigned , default = 0
25:24	R/W	0	input : matrix clock gate ctrl // unsigned , default = 0
23:22	R/W	0	hdr : top cbus clock gate ctrl // unsigned , default = 0
21:20	R/W	0	eotf : cbus clock gate ctrl // unsigned , default = 0
19:18	R/W	0	oetf : cbus clock gate ctrl // unsigned , default = 0
17:16	R/W	0	gamma : mult cbus clock gate ctrl // unsigned , default = 0
15:14	R/W	0	adaptive : cbus scaler clock gate ctrl // unsigned , default = 0
13:12	R/W	0	cgain : cbus clock gate ctrl // unsigned , default = 0
11:10	R/W	0	eotf : clock gate ctrl // unsigned , default = 0
9:8	R/W	0	oetf : clock gate ctrl // unsigned , default = 0
7:6	R/W	0	gamma : mult clock gate ctrl // unsigned , default = 0
5:4	R/W	0	adaptive : scaler clock gate ctrl // unsigned , default = 0
3:2	R/W	0	uv : gain clock gate ctrl // unsigned , default = 0
1:0	R/W	0	cgain : clock gate ctrl // unsigned , default = 0

Table 9-863 DI\_HDR2\_MATRIXI\_COEF00\_01 0x3772

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

Table 9-864 DI\_HDR2\_MATRIXI\_COEF02\_10 0x3773

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

Table 9-865 DI\_HDR2\_MATRIXI\_COEF11\_12 0x3774

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-866 DI\_HDR2\_MATRIXI\_COEF20\_21 0x3775**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-867 DI\_HDR2\_MATRIXI\_COEF22 0x3776**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-868 DI\_HDR2\_MATRIXI\_COEF30\_31 0x3777**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-869 DI\_HDR2\_MATRIXI\_COEF32\_40 0x3778**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-870 DI\_HDR2\_MATRIXI\_COEF41\_42 0x3779**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-871 DI\_HDR2\_MATRIXI\_OFFSET0\_1 0x377A**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-872 DI\_HDR2\_MATRIXI\_OFFSET2 0x377B**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-873 DI\_HDR2\_MATRIXI\_PRE\_OFFSET0\_1 0x377C**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-874 DI\_HDR2\_MATRIXI\_PRE\_OFFSET2 0x377D**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-875 DI\_HDR2\_MATRIXO\_COEF00\_01 0x377E**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

**Table 9-876 DI\_HDR2\_MATRIXO\_COEF02\_10 0x377F**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

**Table 9-877 DI\_HDR2\_MATRIXO\_COEF11\_12 0x3780**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-878 DI\_HDR2\_MATRIXO\_COEF20\_21 0x3781**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-879 DI\_HDR2\_MATRIXO\_COEF22 0x3782**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-880 DI\_HDR2\_MATRIXO\_COEF30\_31 0x3783**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-881 DI\_HDR2\_MATRIXO\_COEF32\_40 0x3784**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-882 DI\_HDR2\_MATRIXO\_COEF41\_42 0x3785**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-883 DI\_HDR2\_MATRIXO\_OFFSET0\_1 0x3786**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-884 DI\_HDR2\_MATRIXO\_OFFSET2 0x3787**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-885 DI\_HDR2\_MATRIXO\_PRE\_OFFSET0\_1 0x3788**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-886 DI\_HDR2\_MATRIXO\_PRE\_OFFSET2 0x3789**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0



**Table 9-887 DI\_HDR2\_MATRIXI\_CLIP 0x378A**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-888 DI\_HDR2\_MATRIXO\_CLIP 0x378B**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-889 DI\_HDR2\_CGAIN\_OFFT 0x378C**

Bit(s)	R/W	Default	Description
26:16	R/W	0	reg_cgain_offt2 : // signed , default = 0
10:0	R/W	0	reg_cgain_offt1 : // signed , default = 0

**Table 9-890 DI\_EOTF\_LUT\_ADDR\_PORT 0x378E**

Bit(s)	R/W	Default	Description
7:0	R/W	0	eotf_lut_addr : // unsigned , default = 0

**Table 9-891 DI\_EOTF\_LUT\_DATA\_PORT 0x378F**

Bit(s)	R/W	Default	Description
19:0	R/W	0	eotf_lut_data : // unsigned , default = 0

**Table 9-892 DI\_OETF\_LUT\_ADDR\_PORT 0x3790**

Bit(s)	R/W	Default	Description
7:0	R/W	0	oetf_lut_addr : // unsigned , default = 0

**Table 9-893 DI\_OETF\_LUT\_DATA\_PORT 0x3791**

Bit(s)	R/W	Default	Description
11:0	R/W	0	oetf_lut_data : // unsigned , default = 0

**Table 9-894 DI\_CGAIN\_LUT\_ADDR\_PORT 0x3792**

Bit(s)	R/W	Default	Description
7:0	R/W	0	cgain_lut_addr : // unsigned , default = 0

**Table 9-895 DI\_CGAIN\_LUT\_DATA\_PORT 0x3793**

Bit(s)	R/W	Default	Description
11:0	R/W	0	cgain_lut_data : // unsigned , default = 0

**Table 9-896 DI\_HDR2\_CGAIN\_COEF0 0x3794**

Bit(s)	R/W	Default	Description
27:16	R/W	0	reg_cgain_coef1 : // unsigned , default = 0
11:0	R/W	0	reg_cgain_coef0 : // unsigned , default = 0

**Table 9-897 DI\_HDR2\_CGAIN\_COEF1 0x3795**

Bit(s)	R/W	Default	Description
11:0	R/W	0	reg_cgain_coef2 : // unsigned , default = 0

**Table 9-898 DI\_OGAIN\_LUT\_ADDR\_PORT 0x3796**

Bit(s)	R/W	Default	Description
7:0	R/W	0	ogain_lut_addr : // unsigned , default = 0

**Table 9-899 DI\_OGAIN\_LUT\_DATA\_PORT 0x3797**

Bit(s)	R/W	Default	Description
11:0	R/W	0	ogain_lut_data : // unsigned , default = 0

**Table 9-900 DI\_HDR2\_ADPS\_CTRL 0x3798**

Bit(s)	R/W	Default	Description
6	R/W	1	reg_adpscl_bypass2 : // unsigned , default = 1
5	R/W	1	reg_adpscl_bypass1 : // unsigned , default = 1
4	R/W	1	reg_adpscl_bypass0 : // unsigned , default = 1
1:0	R/W	1	reg_adpscl_mode : // unsigned , default = 1

**Table 9-901 DI\_HDR2\_ADPS\_ALPHA0 0x3799**

Bit(s)	R/W	Default	Description
29:16	R/W	0x1000	reg_adpscl_alpha1 : // unsigned , default = 0x1000
13:0	R/W	0x1000	reg_adpscl_alpha0 : // unsigned , default = 0x1000

**Table 9-902 DI\_HDR2\_ADPS\_ALPHA1 0x379A**

Bit(s)	R/W	Default	Description
27:24	R/W	0xc	reg_adpscl_shift0 : // unsigned , default = 0xc
23:20	R/W	0xc	reg_adpscl_shift1 : // unsigned , default = 0xc
19:16	R/W	0xc	reg_adpscl_shift2 : // unsigned , default = 0xc
13:0	R/W	0x1000	reg_adpscl_alpha2 : // unsigned , default = 0x1000

**Table 9-903 DI\_HDR2\_ADPS\_BETA0 0x379B**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta0_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta0 : // unsigned , default = 0xfc000

**Table 9-904 DI\_HDR2\_ADPS\_BETA1 0x379C**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta1_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta1 : // unsigned , default = 0xfc000

**Table 9-905 DI\_HDR2\_ADPS\_BETA2 0x379D**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta2_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta2 : // unsigned , default = 0xfc000

**Table 9-906 DI\_HDR2\_ADPS\_COEF0 0x379E**

Bit(s)	R/W	Default	Description
27:16	R/W	460	reg_adpscl_ys_coef1 : // unsigned , default = 460
11:0	R/W	1188	reg_adpscl_ys_coef0 : // unsigned , default = 1188

**Table 9-907 DI\_HDR2\_ADPS\_COEF1 0x379F**

Bit(s)	R/W	Default	Description
11:0	R/W	104	reg_adpscl_ys_coef2 : // unsigned , default = 104

**Table 9-908 DI\_HDR2\_GMUT\_CTRL 0x37A0**

Bit(s)	R/W	Default	Description
3:0	R/W	14	reg_gmut_shift : // unsigned , default = 14

**Table 9-909 DI\_HDR2\_GMUT\_COEF0 0x37A1**

Bit(s)	R/W	Default	Description
31:16	R/W	674	reg_gmut_coef01 : // unsigned , default = 674
15:0	R/W	1285	reg_gmut_coef00 : // unsigned , default = 1285

**Table 9-910 DI\_HDR2\_GMUT\_COEF1 0x37A2**

Bit(s)	R/W	Default	Description
31:16	R/W	142	reg_gmut_coef10 : // unsigned , default = 142
15:0	R/W	89	reg_gmut_coef02 : // unsigned , default = 89

**Table 9-911 DI\_HDR2\_GMUT\_COEF2 0x37A3**

Bit(s)	R/W	Default	Description
31:16	R/W	23	reg_gmut_coef12 : // unsigned , default = 23
15:0	R/W	1883	reg_gmut_coef11 : // unsigned , default = 1883

**Table 9-912 DI\_HDR2\_GMUT\_COEF3 0x37A4**

Bit(s)	R/W	Default	Description
31:16	R/W	180	reg_gmut_coef21 : // unsigned , default = 180
15:0	R/W	34	reg_gmut_coef20 : // unsigned , default = 34

**Table 9-913 DI\_HDR2\_GMUT\_COEF4 0x37A5**

Bit(s)	R/W	Default	Description
15:0	R/W	1834	reg_gmut_coef22 : // unsigned , default = 1834

**Table 9-914 DI\_HDR2\_PIPE\_CTRL1 0x37A6**

Bit(s)	R/W	Default	Description
31:24	R/W	4	vblank_num_oetf : // unsigned , default = 4
23:16	R/W	4	hblank_num_oetf : // unsigned , default = 4
15:8	R/W	10	vblank_num_eotf : // unsigned , default = 10
7:0	R/W	10	hblank_num_eotf : // unsigned , default = 10

**Table 9-915 DI\_HDR2\_PIPE\_CTRL2 0x37A7**

Bit(s)	R/W	Default	Description
31:24	R/W	10	vblank_num_cgain : // unsigned , default = 10
23:16	R/W	10	hblank_num_cgain : // unsigned , default = 10
15:8	R/W	11	vblank_num_gmut : // unsigned , default = 11
7:0	R/W	11	hblank_num_gmut : // unsigned , default = 11

**Table 9-916 DI\_HDR2\_PIPE\_CTRL3 0x37A8**

Bit(s)	R/W	Default	Description
31:24	R/W	22	vblank_num_adps : // unsigned , default = 22
23:16	R/W	2	hblank_num_adps : // unsigned , default = 2
15:8	R/W	4	vblank_num_uv : // unsigned , default = 4
7:0	R/W	4	hblank_num_uv : // unsigned , default = 4

**Table 9-917 DI\_HDR2\_PROC\_WIN1 0x37A9**

Bit(s)	R/W	Default	Description
28:16	R/W	0	proc_win_h_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_h_st : // unsigned , default = 0

**Table 9-918 DI\_HDR2\_PROC\_WIN2 0x37AA**

Bit(s)	R/W	Default	Description
31	R/W	0	proc_win_gmut_en : // unsigned , default = 0
30	R/W	0	proc_win_adps_en : // unsigned , default = 0
29	R/W	0	proc_win_cgain_en : // unsigned , default = 0
28:16	R/W	0	proc_win_v_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_v_st : // unsigned , default = 0

**Table 9-919 DI\_HDR2\_MATRIXI\_EN\_CTRL 0x37AB**

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

**Table 9-920 DI\_HDR2\_MATRIXO\_EN\_CTRL 0x37AC**

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

### 9.2.3.12 NR\_SCALE Registers

#### Register Address

- NRDSWR\_X 0xff00dfe4
- NRDSWR\_Y 0xff00dfe8
- NRDSWR\_CTRL 0xff00dfec
- NRDSWR\_CAN\_SIZE 0xff00dff0
- NR\_DS\_BUF\_SIZE 0xff00dd00
- NR\_DS\_CTRL 0xff00dd04
- NR\_DS\_OFFSET 0xff00dd08
- NR\_DS\_BLD\_COEF 0xff00dd0c

#### Register Description

**Table 9-921 NRDSWR\_X 0x37f9**

Bit(s)	R/W	Default	Description
31:30	R/W	2	burst_len : unsigned , default = 2
29	R/W	0	rev_x : unsigned , default = 0
28:16	R/W	0	start_x : unsigned , default = 0
12:0	R/W	2	end_x : unsigned , default = 2cf

**Table 9-922 NRDSWR\_Y 0x37fa**

Bit(s)	R/W	Default	Description
31:30	R/W	0	canvas_id : unsigned , default = 0
29	R/W	0	rev_y : unsigned , default = 0
28:16	R/W	0	start_y : unsigned , default = 0
12:0	R/W	0	end_y : unsigned , default = 0x1df

**Table 9-923 NRDSWR\_CTRL 0x37fb**

Bit(s)	R/W	Default	Description
31:16	R/W	0	urgent_ctrl : unsigned , default = 0
15	R/W	0	force_wvalid : unsigned , default = 0
14	R/W	0	canvas_syncen : unsigned , default = 0

Bit(s)	R/W	Default	Description
13	R/W	1	canvas_wr : unsigned , default = 1
12	R/W	0	req_en : unsigned , default = 0
10	R/W	0	clr_wrrsp : unsigned , default = 0
8	R/W	0	urgent : unsigned , default = 0
7:0	R/W	0	canvas_index : unsigned , default = 0

**Table 9-924 NRDSWR\_CAN\_SIZE 0x37fc**

Bit(s)	R/W	Default	Description
30:29	R/W	0	reg_rst : unsigned , default = 0
28:16	R/W	0	hsizem1 : unsigned , default = 0x2cf
14	R/W	0	reg_reset : unsigned , default = 0
13	R/W	0	little_endian : unsigned , default = 0
12:0	R/W	0	vsizem1 : unsigned , default = 0x1df

**Table 9-925 NR\_DS\_BUF\_SIZE 0x3740**

Bit(s)	R/W	Default	Description
31:24	R.O	96	dsbuf_rowmax : // unsigned , default = 96
23:16	R/W	128	dsbuf_colmax : // unsigned , default = 128
15: 8	R.O	128	dsbuf_owrow : // unsigned , default = 128
7: 0	R/W	128	dsbuf_ocol : // unsigned , default = 128

**Table 9-926 NR\_DS\_CTRL 0x3741**

Bit(s)	R/W	Default	Description
30	R/W	0	Nr_ds_enable
29:24	R/W	8	reg_h_step : // unsigned , default = 8 rand lut0
21:16	R/W	8	reg_v_step : // unsigned , default = 8 rand lut0
14:12	R/W	4	reg_haa_sel : // unsigned , default = 4
10: 8	R/W	4	reg_vaa_sel : // unsigned , default = 4
6: 4	R/W	1	reg_use_hphase : // unsigned , default = 1
0	R/W	0	reg_yuv_bldmode : // unsigned , default = 0

**Table 9-927 NR\_DS\_OFFSET 0x3742**

Bit(s)	R/W	Default	Description
25:16	R/W	0	reg_h_ofst : // signed , default = 0
9: 0	R/W	0	reg_v_ofst : // signed , default = 0

Table 9-928 NR\_DS\_BLD\_COEF 0x3743

Bit(s)	R/W	Default	Description
23:16	R/W	128	reg_yuv_bldcoef2 : // unsigned , default = 128
15: 8	R/W	64	reg_yuv_bldcoef1 : // unsigned , default = 64
7: 0	R/W	64	reg_yuv_bldcoef0 : // unsigned , default = 64

### 9.2.3.13 MCDI Registers

#### Register Address

- MCDI\_HV\_SIZEIN 0xff00bc00
- MCDI\_HV\_BLKSIZEIN 0xff00bc04
- MCDI\_BLKTOTAL 0xff00bc08
- MCDI\_MOTINEN 0xff00bc0c
- MCDI\_CTRL\_MODE 0xff00bc10
- MCDI\_UNI\_MVDST 0xff00bc14
- MCDI\_BI\_MVDST 0xff00bc18
- MCDI\_SAD\_GAIN 0xff00bc1c
- MCDI\_TXT\_THD 0xff00bc20
- MCDI\_FLT\_MODESEL 0xff00bc24
- MCDI\_CHK\_EDGE\_THD 0xff00bc28
- MCDI\_CHK\_EDGE\_GAIN\_OFFST 0xff00bc2c
- MCDI\_LMV\_RT 0xff00bc30
- MCDI\_LMV\_GAINTHD 0xff00bc34
- MCDI\_RPTMV\_THD0 0xff00bc38
- MCDI\_RPTMV\_THD1 0xff00bc3c
- MCDI\_RPTMV\_THD2 0xff00bc40
- MCDI\_RPTMV\_SAD 0xff00bc44
- MCDI\_RPTMV\_FLG 0xff00bc48
- MCDI\_RPTMV\_GAIN 0xff00bc4c
- MCDI\_GMV\_RT 0xff00bc50
- MCDI\_GMV\_GAIN 0xff00bc54
- MCDI\_HOR\_SADOFST 0xff00bc58
- MCDI\_REF\_MV\_NUM 0xff00bc5c
- MCDI\_REF\_BADW\_THD\_GAIN 0xff00bc60
- MCDI\_REF\_BADW\_SUM\_GAIN 0xff00bc64
- MCDI\_REF\_BS\_THD\_GAIN 0xff00bc68
- MCDI\_REF\_ERR\_GAIN0 0xff00bc6c
- MCDI\_REF\_ERR\_GAIN1 0xff00bc70
- MCDI\_REF\_ERR\_FRQ\_CHK 0xff00bc74
- MCDI\_QME\_LPF\_MSK 0xff00bc78



- MCDI\_REL\_DIF\_THD\_02 0xff00bc7c
- MCDI\_REL\_DIF\_THD\_34 0xff00bc80
- MCDI\_REL\_BADW\_GAIN\_OFFST\_01 0xff00bc84
- MCDI\_REL\_BADW\_GAIN\_OFFST\_23 0xff00bc88
- MCDI\_REL\_BADW\_THD\_GAIN\_OFFST 0xff00bc8c
- MCDI\_REL\_BADW\_THD\_MIN\_MAX 0xff00bc90
- MCDI\_REL\_SAD\_GAIN\_OFFST\_01 0xff00bc94
- MCDI\_REL\_SAD\_GAIN\_OFFST\_23 0xff00bc98
- MCDI\_REL\_SAD\_THD\_GAIN\_OFFST 0xff00bc9c
- MCDI\_REL\_SAD\_THD\_MIN\_MAX 0xff00bca0
- MCDI\_REL\_DET\_GAIN\_00 0xff00bca4
- MCDI\_REL\_DET\_GAIN\_01 0xff00bca8
- MCDI\_REL\_DET\_GAIN\_10 0xff00bcac
- MCDI\_REL\_DET\_GAIN\_11 0xff00bcb0
- MCDI\_REL\_DET\_GAIN\_20 0xff00bcb4
- MCDI\_REL\_DET\_GAIN\_21 0xff00bcb8
- MCDI\_REL\_DET\_GMV\_DIF\_CHK 0xff00bcbc
- MCDI\_REL\_DET\_LMV\_DIF\_CHK 0xff00bcc0
- MCDI\_REL\_DET\_FRQ\_CHK 0xff00bcc4
- MCDI\_REL\_DET\_PD22\_CHK 0xff00bcc8
- MCDI\_REL\_DET\_RPT\_CHK\_ROW 0xff00bccc
- MCDI\_REL\_DET\_RPT\_CHK\_GAIN\_QMV 0xff00bcd0
- MCDI\_REL\_DET\_RPT\_CHK\_THD\_0 0xff00bcd4
- MCDI\_REL\_DET\_RPT\_CHK\_THD\_1 0xff00bcd8
- MCDI\_REL\_DET\_LPF\_DIF\_THD 0xff00bcdc
- MCDI\_REL\_DET\_LPF\_MSK\_00\_03 0xff00bce0
- MCDI\_REL\_DET\_LPF\_MSK\_04\_12 0xff00bce4
- MCDI\_REL\_DET\_LPF\_MSK\_13\_21 0xff00bce8
- MCDI\_REL\_DET\_LPF\_MSK\_22\_30 0xff00bcec
- MCDI\_REL\_DET\_LPF\_MSK\_31\_34 0xff00bcf0
- MCDI\_REL\_DET\_MIN 0xff00bcf4
- MCDI\_REL\_DET\_LUT\_0\_3 0xff00bcf8
- MCDI\_REL\_DET\_LUT\_4\_7 0xff00bcfc
- MCDI\_REL\_DET\_LUT\_8\_11 0xff00bd00
- MCDI\_REL\_DET\_LUT\_12\_15 0xff00bd04
- MCDI\_REL\_DET\_COL\_CFD\_THD 0xff00bd08
- MCDI\_REL\_DET\_COL\_CFD\_AVG\_LUMA 0xff00bd0c
- MCDI\_REL\_DET\_BAD\_THD\_0 0xff00bd10
- MCDI\_REL\_DET\_BAD\_THD\_1 0xff00bd14
- MCDI\_PD22\_CHK\_THD 0xff00bd18
- MCDI\_PD22\_CHK\_GAIN\_OFFST\_0 0xff00bd1c
- MCDI\_PD22\_CHK\_GAIN\_OFFST\_1 0xff00bd20
- MCDI\_LMV\_LOCK\_CNT\_THD\_GAIN 0xff00bd24
- MCDI\_LMV\_LOCK\_ABS\_DIF\_THD 0xff00bd28

- MCDI\_LMV\_LOCK\_ROW 0xff00bd2c
- MCDI\_LMV\_LOCK\_RT\_MODE 0xff00bd30
- MCDI\_GMV\_LOCK\_CNT\_THD\_GAIN 0xff00bd34
- MCDI\_GMV\_LOCK\_ABS\_DIF\_THD 0xff00bd38
- MCDI\_HIGH\_VERT\_FRQ\_DIF\_THD 0xff00bd3c
- MCDI\_HIGH\_VERT\_FRQ\_DIF\_DIF\_THD 0xff00bd40
- MCDI\_HIGH\_VERT\_FRQ\_RT\_GAIN 0xff00bd44
- MCDI\_MOTION\_PARADOX\_THD 0xff00bd48
- MCDI\_MOTION\_PARADOX\_RT 0xff00bd4c
- MCDI\_MOTION\_REF\_THD 0xff00bd50
- MCDI\_REL\_COL\_REF\_RT 0xff00bd54
- MCDI\_PD22\_CHK\_THD\_RT 0xff00bd58
- MCDI\_CHAR\_DET\_DIF\_THD 0xff00bd5c
- MCDI\_CHAR\_DET\_CNT\_THD 0xff00bd60
- MCDI\_PD\_22\_CHK\_WND0\_X 0xff00bd64
- MCDI\_PD\_22\_CHK\_WND0\_Y 0xff00bd68
- MCDI\_PD\_22\_CHK\_WND1\_X 0xff00bd6c
- MCDI\_PD\_22\_CHK\_WND1\_Y 0xff00bd70
- MCDI\_PD\_22\_CHK\_FRC\_LMV 0xff00bd74
- MCDI\_PD\_22\_CHK\_FLG\_CNT 0xff00bd78
- MCDI\_RO\_FLD\_PD\_22\_PRE\_CNT1 0xff00bf28
- MCDI\_RO\_FLD\_PD\_22\_FOR\_CNT1 0xff00bf2c
- MCDI\_RO\_FLD\_PD\_22\_FLT\_CNT1 0xff00bf30
- MCDI\_RO\_FLD\_PD\_22\_PRE\_CNT2 0xff00bf34
- MCDI\_RO\_FLD\_PD\_22\_FOR\_CNT2 0xff00bf38
- MCDI\_RO\_FLD\_PD\_22\_FLT\_CNT2 0xff00bf3c
- MCDI\_FIELD\_MV 0xff00bd80
- MCDI\_FIELD\_HVF\_PRDX\_CNT 0xff00bd84
- MCDI\_FIELD\_LUMA\_AVG\_SUM\_0 0xff00bd88
- MCDI\_FIELD\_LUMA\_AVG\_SUM\_1 0xff00bd8c
- MCDI\_YCBCR\_BLEND\_CRTL 0xff00bd90
- MCDI\_MCVFCWR\_CANVAS\_SIZE 0xff00bd94
- MCDI\_MCVFCRD\_CANVAS\_SIZE 0xff00bd98
- MCDI\_MCINFOWR\_CANVAS\_SIZE 0xff00bd9c
- MCDI\_MCINFORD\_CANVAS\_SIZE 0xff00bda0
- MCDI\_MCVFCWR\_X 0xff00be48
- MCDI\_MCVFCWR\_Y 0xff00be4c
- MCDI\_MCVFCWR\_CTRL 0xff00be50
- MCDI\_MCVFCRD\_X 0xff00be54
- MCDI\_MCVFCRD\_Y 0xff00be58
- MCDI\_MCVFCRD\_CTRL 0xff00be5c
- MCDI\_MCINFOWR\_X 0xff00be60
- MCDI\_MCINFOWR\_Y 0xff00be64
- MCDI\_MCINFOWR\_CTRL 0xff00be68

- MCDI\_MCINFORD\_X 0xff00be6c
- MCDI\_MCINFORD\_Y 0xff00be70
- MCDI\_MCINFORD\_CTRL 0xff00be74
- MCDI\_LMVLCKSTEXT\_0 0xff00bda4
- MCDI\_LMVLCKSTEXT\_1 0xff00bda8
- MCDI\_LMVLCKEEXT\_0 0xff00bdac
- MCDI\_LMVLCKEEXT\_1 0xff00bdb0
- MCDI\_MC\_CRTL 0xff00bdc0
- MCDI\_MC\_LPF\_MSK\_0 0xff00bdc4
- MCDI\_MC\_LPF\_MSK\_1 0xff00bdc8
- MCDI\_MC\_LPF\_MSK\_2 0xff00bdcc
- MCDI\_MC\_LPF\_MSK\_3 0xff00bdd0
- MCDI\_MC\_LPF\_MSK\_4 0xff00bdd4
- MCDI\_MC\_REL\_GAIN\_OFFST\_0 0xff00bdd8
- MCDI\_MC\_REL\_GAIN\_OFFST\_1 0xff00bddc
- MCDI\_MC\_COL\_CFD\_0 0xff00bde0
- MCDI\_MC\_COL\_CFD\_1 0xff00bde4
- MCDI\_MC\_COL\_CFD\_2 0xff00bde8
- MCDI\_MC\_COL\_CFD\_3 0xff00bdec
- MCDI\_MC\_COL\_CFD\_4 0xff00bdf0
- MCDI\_MC\_COL\_CFD\_5 0xff00bdf4
- MCDI\_MC\_COL\_CFD\_6 0xff00bdf8
- MCDI\_MC\_COL\_CFD\_7 0xff00bdfc
- MCDI\_MC\_COL\_CFD\_8 0xff00be00
- MCDI\_MC\_COL\_CFD\_9 0xff00be04
- MCDI\_MC\_COL\_CFD\_10 0xff00be08
- MCDI\_MC\_COL\_CFD\_11 0xff00be0c
- MCDI\_MC\_COL\_CFD\_12 0xff00be10
- MCDI\_MC\_COL\_CFD\_13 0xff00be14
- MCDI\_MC\_COL\_CFD\_14 0xff00be18
- MCDI\_MC\_COL\_CFD\_15 0xff00be1c
- MCDI\_MC\_COL\_CFD\_16 0xff00be20
- MCDI\_MC\_COL\_CFD\_17 0xff00be24
- MCDI\_MC\_COL\_CFD\_18 0xff00be28
- MCDI\_MC\_COL\_CFD\_19 0xff00be2c
- MCDI\_MC\_COL\_CFD\_20 0xff00be30
- MCDI\_MC\_COL\_CFD\_21 0xff00be34
- MCDI\_MC\_COL\_CFD\_22 0xff00be38
- MCDI\_MC\_COL\_CFD\_23 0xff00be3c
- MCDI\_MC\_COL\_CFD\_24 0xff00be40
- MCDI\_MC\_COL\_CFD\_25 0xff00be44
- MCDI\_RO\_FLD\_LUMA\_AVG\_SUM 0xff00be80
- MCDI\_RO\_GMV\_VLD\_CNT 0xff00be84
- MCDI\_RO\_RPT\_FLG\_CNT 0xff00be88

- MCDI\_RO\_FLD\_BAD\_SAD\_CNT 0xff00be8c
- MCDI\_RO\_FLD\_BAD\_BADW\_CNT 0xff00be90
- MCDI\_RO\_FLD\_BAD\_REL\_CNT 0xff00be94
- MCDI\_RO\_FLD\_MTN\_CNT 0xff00be98
- MCDI\_RO\_FLD\_VLD\_CNT 0xff00be9c
- MCDI\_RO\_FLD\_PD\_22\_PRE\_CNT 0xff00bea0
- MCDI\_RO\_FLD\_PD\_22\_FOR\_CNT 0xff00bea4
- MCDI\_RO\_FLD\_PD\_22\_FLT\_CNT 0xff00bea8
- MCDI\_RO\_HIGH\_VERT\_FRQ\_FLG 0xff00beac
- MCDI\_RO\_GMV\_LOCK\_FLG 0xff00beb0
- MCDI\_RO\_RPT\_MV 0xff00beb4
- MCDI\_RO\_MOTION\_PARADOX\_FLG 0xff00beb8
- MCDI\_RO\_PD\_22\_FLG 0xff00bebc
- MCDI\_RO\_COL\_CFD\_0 0xff00bec0
- MCDI\_RO\_COL\_CFD\_1 0xff00bec4
- MCDI\_RO\_COL\_CFD\_2 0xff00bec8
- MCDI\_RO\_COL\_CFD\_3 0xff00becc
- MCDI\_RO\_COL\_CFD\_4 0xff00bed0
- MCDI\_RO\_COL\_CFD\_5 0xff00bed4
- MCDI\_RO\_COL\_CFD\_6 0xff00bed8
- MCDI\_RO\_COL\_CFD\_7 0xff00bedc
- MCDI\_RO\_COL\_CFD\_8 0xff00bee0
- MCDI\_RO\_COL\_CFD\_9 0xff00bee4
- MCDI\_RO\_COL\_CFD\_10 0xff00bee8
- MCDI\_RO\_COL\_CFD\_11 0xff00beec
- MCDI\_RO\_COL\_CFD\_12 0xff00bef0
- MCDI\_RO\_COL\_CFD\_13 0xff00bef4
- MCDI\_RO\_COL\_CFD\_14 0xff00bef8
- MCDI\_RO\_COL\_CFD\_15 0xff00befc
- MCDI\_RO\_COL\_CFD\_16 0xff00bf00
- MCDI\_RO\_COL\_CFD\_17 0xff00bf04
- MCDI\_RO\_COL\_CFD\_18 0xff00bf08
- MCDI\_RO\_COL\_CFD\_19 0xff00bf0c
- MCDI\_RO\_COL\_CFD\_20 0xff00bf10
- MCDI\_RO\_COL\_CFD\_21 0xff00bf14
- MCDI\_RO\_COL\_CFD\_22 0xff00bf18
- MCDI\_RO\_COL\_CFD\_23 0xff00bf1c
- MCDI\_RO\_COL\_CFD\_24 0xff00bf20
- MCDI\_RO\_COL\_CFD\_25 0xff00bf24
- MCDI\_RO\_FLD\_PD\_22\_PRE\_CNT1 0xff00bf28
- MCDI\_RO\_FLD\_PD\_22\_FOR\_CNT1 0xff00bf2c
- MCDI\_RO\_FLD\_PD\_22\_FLT\_CNT1 0xff00bf30
- MCDI\_RO\_FLD\_PD\_22\_PRE\_CNT2 0xff00bf34
- MCDI\_RO\_FLD\_PD\_22\_FOR\_CNT2 0xff00bf38

- MCDI\_RO\_FLD\_PD\_22\_FLT\_CNT2 0xff00bf3c

## Register Description

**Table 9-929 MCDI\_HV\_SIZEIN 0x2f00**

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28-16	R/W	1024	reg_mcdi_hsize image horizontal size (number of cols) default=1024
15-13	R/W		reserved
12-0	R/W	1024	reg_mcdi_vsize image vertical size (number of rows) default=1024

**Table 9-930 MCDI\_HV\_BLKSIZEIN 0x2f01**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_mcdi_vrev default = 0
30	R/W	0	reg_mcdi_hrev default = 0
29-28	R/W		reserved
27-16	R/W	1024	reg_mcdi_blkhsz image horizontal blk size (number of cols) default=1024
15-13	R/W		reserved
11-0	R/W	1024	reg_mcdi_blkvsize image vertical blk size (number of rows) default=1024

**Table 9-931 MCDI\_BLKTOTAL 0x2f02**

Bit(s)	R/W	Default	Description
31-24	R/W		reserved
23-0	R/W	0	reg_mcdi_blktotal

**Table 9-932 MCDI\_MOTINEN 0x2f03**

Bit(s)	R/W	Default	Description
31-2	R/W		reserved
1	R/W	1	reg_mcdi_motionrefen. enable motion refinement of MA, default = 1
0	R/W	1	reg_mcdi_motionparadoxen. enable motion paradox detection, default = 1

**Table 9-933 MCDI\_CTRL\_MODE 0x2f04**

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28	R/W	0	mc info read enable
27-26	R/W	2	reg_mcdi_lmvlocken 0:disable, 1: use max Lmv, 2: use no-zero Lmv, lmv lock enable mode, default = 2
25	R/W	1	reg_mcdi_reldetrptchken 0-unable; 1: enableenable repeat pattern check (not repeat mv detection) in rel det part, default = 1

Bit(s)	R/W	Default	Description
24	R/W	1	reg_mcdi_reldetgmvpd22chken 0-unable; 1: enable enable pull-down 22 mode check in gmV lock mode for rel det, default = 1
23	R/W	1	reg_mcdi_pd22chken 0-unable; 1: enable enable pull-down 22 mode check (lock) function, default = 1
22	R/W	1	reg_mcdi_reldetlpfen 0-unable; 1: enable enable det value lpf, default = 1
21	R/W	1	reg_mcdi_reldetlmvpd22chken 0-unable; 1: enable enable pull-down 22 mode check in lmv lock mode for rel det, default = 1
20	R/W	1	reg_mcdi_reldetlmvdifchken 0-unable; 1: enable enable lmv dif check in lmv lock mode for rel det, default = 1
19	R/W	1	reg_mcdi_reldetgmvdifchken 0-unable; 1: enable enable lmv dif check in lmv lock mode for rel det, default = 1
18	R/W	1	reg_mcdi_reldetpd22chken 0-unable; 1: enable enable pull-down 22 mode check for rel det refinement, default = 1
17	R/W	1	reg_mcdi_reldetfrqchken 0-unable; 1: enable enable mv frequency check in rel det, default = 1
16	R/W		reg_mcdi_qmeen 0-unable; 1: enable enable quarter motion estimation, default = 1
15	R/W	1	reg_mcdi_refrptmven 0-unable; 1: enable use repeat mv in refinement, default = 1
14	R/W	1	reg_mcdi_refgmven 0-unable; 1: enable use gmV in refinement, default = 1
13	R/W	1	reg_mcdi_reflmven 0-unable; 1: enable use lmvS in refinement, default = 1
12	R/W	1	reg_mcdi_refnmven 0-unable; 1: enable use neighboring mvS in refinement, default = 1
11	R/W		reserved
10	R/W	1	reg_mcdi_referrfqchken 0-unable; 1: enable enable mv frequency check while finding min err in ref, default = 1
9	R/W	1	reg_mcdi_refen 0-unable; 1: enable enable mv refinement, default = 1
8	R/W	1	reg_mcdi_horlineen 0-unable; 1: enable enable horizontal lines detection by sad map, default = 1
7	R/W	1	reg_mcdi_highvertfrqdeten 0-unable; 1: enable enable high vertical frequency pattern detection, default = 1
6	R/W	1	reg_mcdi_gmvlocken 0-unable; 1: enable enable gmV lock mode, default = 1
5	R/W	1	reg_mcdi_rptmven 0-unable; 1: enable enable repeat pattern detection, default = 1
4	R/W	1	reg_mcdi_gmven 0-unable; 1: enable enable global motion estimation, default = 1
3	R/W	1	reg_mcdi_lmven 0-unable; 1: enable enable line mv estimation for hme, default = 1
2	R/W	1	reg_mcdi_chkedgeen 0-unable; 1: enable enable check edge function, default = 1
1	R/W	1	reg_mcdi_txtdeten 0-unable; 1: enable enable texture detection, default = 1

**Table 9-934 MCDI\_UNI\_MVDST 0x2f05**

Bit(s)	R/W	Default	Description
31-20	R/W		reserved
19-17	R/W	1	reg_mcdi_unimvdstabsseg0 segment0 for uni-mv abs, default = 1
16-12	R/W	15	reg_mcdi_unimvdstabsseg1 segment1 for uni-mv abs, default = 15
11-8	R/W	2	reg_mcdi_unimvdstabsdifgain0 2/2, gain0 of uni-mv abs dif for segment0, normalized 2 to '1', default = 2
7-5	R/W	2	reg_mcdi_unimvdstabsdifgain1 2/2, gain1 of uni-mv abs dif for segment1, normalized 2 to '1', default = 2
4-2	R/W	2	reg_mcdi_unimvdstabsdifgain2 2/2, gain2 of uni-mv abs dif beyond segment1, normalized 2 to '1', default = 2
1-0	R/W	0	reg_mcdi_unimvdstsgnshft shift for neighboring distance of uni-mv, default = 0

**Table 9-935 MCDI\_BI\_MVDST 0x2f06**

Bit(s)	R/W	Default	Description
31-20	R/W		reserved
19-17	R/W	1	reg_mcdi_bimvdstabsseg0 segment0 for bi-mv abs, default = 1
16-12	R/W	9	reg_mcdi_bimvdstabsseg1 segment1 for bi-mv abs, default = 9
11-8	R/W	6	reg_mcdi_bimvdstabsdifgain0 6/2, gain0 of bi-mv abs dif for segment0, normalized 2 to '1', default = 6
7-5	R/W	3	reg_mcdi_bimvdstabsdifgain1 3/2, gain1 of bi-mvabs dif for segment1, normalized 2 to '1', default = 3
4-2	R/W	2	reg_mcdi_bimvdstabsdifgain2 2/2, gain2 of bi-mvabs dif beyond segment1, normalized 2 to '1', default = 2
1-0	R/W	0	reg_mcdi_bimvdstsgnshft shift for neighboring distance of bi-mv, default = 0

**Table 9-936 MCDI\_SAD\_GAIN 0x2f07**

Bit(s)	R/W	Default	Description
31-19	R/W		reserved
18-17	R/W	3	reg_mcdi_unisadcorepxlgain uni-sad core pixels gain, default = 3
16	R/W	0	reg_mcdi_unisadcorepxlnormen enable uni-sad core pixels normalization, default = 0
15-11	R/W		reserved
10-9	R/W	3	reg_mcdi_bisadcorepxlgain bi-sad core pixels gain, default = 3
8	R/W	1	reg_mcdi_bisadcorepxlnormen enable bi-sad core pixels normalization, default = 1
7-3	R/W		reserved
2-1	R/W	3	reg_mcdi_biqsadcorepxlgain bi-qsad core pixels gain, default = 3
0	R/W	1	reg_mcdi_biqsadcorepxlnormen enable bi-qsad core pixels normalization, default = 1

**Table 9-937 MCDI\_TXT\_THD 0x2f08**

Bit(s)	R/W	Default	Description
31-24	R/W		reserved
23-16	R/W	24	reg_mcdi_txtminmaxdifthd, min max dif threshold ( $\geq$ ) for texture detection, default = 24
15-8	R/W	9	reg_mcdi_txtmeandifthd, mean dif threshold ( $<$ ) for texture detection, default = 9
7-3	R/W		reserved
2-0	R/W	2	reg_mcdi_txtdetthd, texture detecting threshold, 0~4, default = 2

**Table 9-938 MCDI\_FLT\_MODESEL 0x2f09**

Bit(s)	R/W	Default	Description
31	R/W		reserved
30-28	R/W	1	reg_mcdi_fltorlineselmode mode for horizontal line detecting flat calculation, default = 1, same as below
27	R/W		reserved
26-24	R/W	4	reg_mcdi_fltgmvselmode mode for gmv flat calculation, default = 4, same as below
23	R/W		reserved
22-20	R/W	2	reg_mcdi_fltadselmode mode for sad flat calculation, default = 2, same as below
19	R/W		reserved
18-16	R/W	3	reg_mcdi_fltbadwselmode mode for badw flat calculation, default = 3, same as below
15	R/W		reserved
14-12	R/W	4	reg_mcdi_fltrptmvselmode mode for repeat mv flat calculation, default = 4, same as below
11	R/W		reserved
10-8	R/W	4	reg_mcdi_fltbadrelselmode mode for bad rel flat calculation, default = 4, same as below
7	R/W		reserved
6-4	R/W	2	reg_mcdi_fltcolcfdselmode mode for col cfd flat calculation, default = 2, same as below
3	R/W		reserved
2-0	R/W	2	reg_mcdi_fltpd22chksselmode mode for pd22 check flat calculation, default = 2, 0:cur dif h, 1: cur dif v, 2: pre dif h, 3: pre dif v, 4: cur flt, 5: pre flt, 6: cur+pre, 7: max all(cur,pre)

**Table 9-939 MCDI\_CHK\_EDGE\_THD 0x2f0a**

Bit(s)	R/W	Default	Description
23-28	R/W		reserved.
27-24	R/W	1	reg_mcdi_chkgedgedifsadthd. thd ( $\leq$ ) for sad dif check, 0~8, default = 1
23-16	R/W		reserved.



Bit(s)	R/W	Default	Description
15-12	R/W	15	reg_mcdi_chkedgemaxedgethd. max drt of edge, default = 15
11-8	R/W	2	reg_mcdi_chkedgeminedgethd. min drt of edge, default = 2
7	R/W		reserved.
6-0	R/W	14	reg_mcdi_chkedgevdifthd. thd for vertical dif in check edge, default = 14

Table 9-940 MCDI\_CHK\_EDGE\_GAIN\_OFFST 0x2f0b

Bit(s)	R/W	Default	Description
31-24	R/W		reserved.
23-20	R/W	4	reg_mcdi_chkedgedifthd1. thd1 for edge dif check (<=), default = 4
19-16	R/W	15	reg_mcdi_chkedgedifthd0. thd0 for edge dif check (>=), default = 15
-15	R/W		reserved.
14-10	R/W	24	reg_mcdi_chkedgechklen. total check length for edge check, 1~24 (>0), default = 24
9-8	R/W	1	reg_mcdi_chkedgeedgesel. final edge select mode, 0: original start edge, 1: lpf start edge, 2: original start+end edge, 3: lpf start+end edge, default = 1
7-3	R/W	4	reg_mcdi_chkedgesaddstgain. distance gain for sad calc while getting edges, default = 4
2	R/W		reg_mcdi_chkedgechkmode. edge used in check mode, 0- original edge, 1: lpf edge, default = 1
1	R/W		reg_mcdi_chkedgestartedge. edge mode for start edge, 0- original edge, 1: lpf edge, default = 0
0	R/W	0	reg_mcdi_chkedgeedgelpf. edge lpf mode, 0-[0,2,4,2,0], 1:[1,2,2,2,1], default = 0

Table 9-941 MCDI\_LMV\_RT 0x2f0c

Bit(s)	R/W	Default	Description
31-24	R/W	32	Reg_mcdi_lmv_vx_gain
14-12	R/W		reg_mcdi_lmvvalidmode valid mode for lmv calc., 100b: use char det, 010b: use flt, 001b: use hori flg
11-10	R/W	1	reg_mcdi_lmvgainmvmode four modes of mv selection for lmv weight calucluation, default = 1
// // lst(x-1	R/W		x,x+1); 1- cur(x-4,x-3), lst(x,x+1); 2: cur(x-5,x-4,x-3), lst(x-1,x,x+1,x+2,x+3); 3: cur(x-6,x-5,x-4,x-3), lst(x-1,x,x+1,x+2);
9	R/W	0	reg_mcdi_lmvinitmode initial lmv at first row of input field, 0- initial value = 0; 1: initial = 32 (invalid), default = 0
8	R/W		reserved
7-4	R/W	5	reg_mcdi_lmvrt0 ratio of max mv, default = 5
3-0	R/W	5	reg_mcdi_lmvrt1 ratio of second max mv, default = 5

**Table 9-942 MCDI\_LMV\_GAINTHD 0x2f0d**

Bit(s)	R/W	Default	Description
31-24	R/W	96	reg_mcdi_lmvsxmaxgain max gain of lmv weight, default = 96
23	R/W		reserved
22-20	R/W	1	reg_mcdi_lmvdifthd0 dif threshold 0 (<) for small lmv, default = 1
19-17	R/W	2	reg_mcdi_lmvdifthd1 dif threshold 1 (<) for median lmv, default = 2
16-14	R/W	3	reg_mcdi_lmvdifthd2 dif threshold 2 (<) for large lmv, default = 3
13-8	R/W	20	reg_mcdi_lmvsnumlmt least/limit number of (total number - max0), default = 20
7-0	R/W	9	reg_mcdi_lmvsfltthd flt cnt thd (<) for lmv, default = 9

**Table 9-943 MCDI\_RPTMV\_THD0 0x2f0e**

Bit(s)	R/W	Default	Description
31-25	R/W	64	reg_mcdi_rptmvslpthd2 slope thd (>=) between i and i+3/i-3 (i+4/i-4), default = 64
24-20	R/W	4	reg_mcdi_rptmvslpthd1 slope thd (>=) between i and i+2/i-2, default = 4
19-10	R/W	300	reg_mcdi_rptmvampthd2 amplitude thd (>=) between max and min, when count cycles, default = 300
9-0	R/W	400	reg_mcdi_rptmvampthd1 amplitude thd (>=) between average of max and min, default = 400

**Table 9-944 MCDI\_RPTMV\_THD1 0x2f0f**

Bit(s)	R/W	Default	Description
31-28	R/W		reserved
27-25	R/W	2	reg_mcdi_rptmvcycnthd thd (>=) of total cycles count, default = 2
24-21	R/W	3	reg_mcdi_rptmvcycdifthd dif thd (<) of cycles length, default = 3
20-18	R/W	1	reg_mcdi_rptmvcycvldthd thd (>) of valid cycles number, default = 1
17-15	R/W	2	reg_mcdi_rptmvhalfcycminthd min length thd (>=) of half cycle, default = 2
14-11	R/W	5	reg_mcdi_rptmvhalfcycdifthd neighboring half cycle length dif thd (<), default = 5
10-8	R/W	2	reg_mcdi_rptmvminmaxcnthd least number of valid max and min, default = 2
7-5	R/W	2	reg_mcdi_rptmvcycminthd min length thd (>=) of cycles, default = 2
4-0	R/W	17	reg_mcdi_rptmvcycmaxthd max length thd (<) of cycles, default = 17

**Table 9-945 MCDI\_RPTMV\_THD2 0x2f10**

Bit(s)	R/W	Default	Description
31-24	R/W		reserved
23-16	R/W	8	reg_mcdi_rptmvhdifthd0 higher hdif thd (>=) (vertical edge) for rpt detection, default = 8

Bit(s)	R/W	Default	Description
15-8	R/W	4	reg_mcdi_rptmvhdifthd1 hdif thd ( $\geq$ ) (slope edge) for rpt detection, default = 4
7-0	R/W	1	reg_mcdi_rptmvvdifthd vdif thd ( $\geq$ ) (slope edge) for rpt detection, default = 1

Table 9-946 MCDI\_RPTMV\_SAD 0x2f11

Bit(s)	R/W	Default	Description
31-26	R/W		reserved
25-16	R/W	336	reg_mcdi_rptmvsaddifthdgain 7x3x(16/16), gain for sad dif thd in rpt mv detection, 0~672, normalized 16 as '1', default = 336
15-10	R/W		reserved
9-0	R/W	16	reg_mcdi_rptmvsaddifthdoffst offset for sad dif thd in rpt mv detection, -512~511, default = 16

Table 9-947 MCDI\_RPTMV\_FLG 0x2f12

Bit(s)	R/W	Default	Description
31-18	R/W		reserved
17-16	R/W	2	reg_mcdi_rptmvmode select mode of mvs for repeat motion estimation, 0: hmv, 1: qmv/2, 2 or 3: qmv/4, default = 2
15-8	R/W	64	reg_mcdi_rptmvflgcntthd thd ( $\geq$ ) of min count number for rptmv of whole field, for rptmv estimation, default = 64
7-5	R/W		reserved
4-0	R/W		reg_mcdi_rptmvflgcntrt 4/32, ratio for repeat mv flag count, normalized 32 as '1', set 31 to 32,

Table 9-948 MCDI\_RPTMV\_GAIN 0x2f13

Bit(s)	R/W	Default	Description
31-24	R/W	96	reg_mcdi_rptmvlftgain up repeat mv gain for hme, default = 96
23-16	R/W	32	reg_mcdi_rptmvuplftgain up left repeat mv gain for hme, default = 32
15-8	R/W	64	reg_mcdi_rptmvupgain up repeat mv gain for hme, default = 64
7-0	R/W	32	reg_mcdi_rptmvuprightgain up right repeat mv gain for hme, default = 32

Table 9-949 MCDI\_GMV\_RT 0x2f14

Bit(s)	R/W	Default	Description
31	R/W		reserved
30-24	R/W	32	reg_mcdi_gmvmtnr0 ratio 0 for motion senario, set 127 to 128, normalized 128 as '1', default =32
23	R/W		reserved
22-16	R/W	56	reg_mcdi_gmvmtnr1 ratio 1 for motion senario, set 127 to 128, normalized 128 as '1', default = 56

Bit(s)	R/W	Default	Description
15	R/W		reserved
14-8	R/W	56	reg_mcdi_gmvstlrt0 ratio 0 for still senario, set 127 to 128, normalized 128 as '1', default = 56
7	R/W		reserved
6-0	R/W	80	reg_mcdi_gmvstlrt1 ratio 1 for still senario, set 127 to 128, normalized 128 as '1', default = 80

**Table 9-950 MCDI\_GMV\_GAIN 0x2f15**

Bit(s)	R/W	Default	Description
31-25	R/W	100	reg_mcdi_gmvzeromvlockrt0 ratio 0 for locking zero mv, set 127 to 128, normalized 128 as '1', default = 100
24-18	R/W	112	reg_mcdi_gmvzeromvlockrt1 ratio 1 for locking zero mv, set 127 to 128, normalized 128 as '1', default = 112
17-16	R/W	3	reg_mcdi_gmvvalidmode valid mode for gmv calc., 10b: use flt, 01b: use hori flg, default = 3
15-8	R/W	0	reg_mcdi_gmvvxgain gmv's vx gain when gmv locked for hme, default = 0
7-0	R/W	3	reg_mcdi_gmvflthd flat thd (<) for gmv calc. default = 3

**Table 9-951 MCDI\_HOR\_SADOFST 0x2f16**

Bit(s)	R/W	Default	Description
31-25	R/W		reserved
24-16	R/W	21	reg_mcdi_horsaddifthdgain $21 * 1/8$ , gain/divisor for sad dif threshold in hor line detection, normalized 8 as '1', default = 21
15-8	R/W	0	reg_mcdi_horsaddifthdoffst offset for sad dif threshold in hor line detection, -128~127, default = 0
7-0	R/W	24	reg_mcdi_horvdifthd threshold ( $\geq$ ) of vertical dif of next block for horizontal line detection, default = 24

**Table 9-952 MCDI\_REF\_MV\_NUM 0x2f17**

Bit(s)	R/W	Default	Description
31-2	R/W		reserved
1-0	R/W	0	reg_mcdi_refmcmode. motion compensated mode used in refinement, 0: pre, 1: next, 2: (pre+next)/2, default = 0

**Table 9-953 MCDI\_REF\_BADW\_THD\_GAIN 0x2f18**

Bit(s)	R/W	Default	Description
31-28	R/W		reserved
27-24	R/W	6	reg_mcdi_refbadwcnt2gain. gain for badwv count num==3, default = 6
23-20	R/W	3	reg_mcdi_refbadwcnt1gain. gain for badwv count num==2, default = 3
19-16	R/W	1	reg_mcdi_refbadwcnt0gain. gain for badwv count num==1, default = 1

Bit(s)	R/W	Default	Description
15-12	R/W	4	reg_mcdi_refbadwthd3. threshold 3 for detect badweave with largest average luma, default = 4
11-8	R/W	3	reg_mcdi_refbadwthd2. threshold 2 for detect badweave with third smallest average luma, default = 3
7-4	R/W	2	reg_mcdi_refbadwthd1. threshold 1 for detect badweave with second smallest average luma, default = 2
3-0	R/W	1	reg_mcdi_refbadwthd0. threshold 0 for detect badweave with smallest average luma, default = 1

Table 9-954 MCDI\_REF\_BADW\_SUM\_GAIN 0x2f19

Bit(s)	R/W	Default	Description
31-13	R/W		reserved
12-8	R/W	8	reg_mcdi_refbadwsumgain0. sum gain for r channel, 0~16, default = 8
7-5	R/W		reserved
4	R/W	0	reg_mcdi_refbadwcalcmode. mode for badw calculation, 0-sum, 1:max, default = 0
3-0	R/W		reserved

Table 9-955 MCDI\_REF\_BS\_THD\_GAIN 0x2f1a

Bit(s)	R/W	Default	Description
31-28	R/W	2	reg_mcdi_refbsudgain1. up & down block strength gain1, normalized to 8 as '1', default = 2
27-24	R/W	4	reg_mcdi_refbsudgain0. up & down block strength gain0, normalized to 8 as '1', default = 4
23-19	R/W		reserved
18-16	R/W	0	reg_mcdi_refbslftgain. left block strength gain, default = 0
15-13	R/W		reserved
12-8	R/W	16	reg_mcdi_refbsthd1. threshold 1 for detect block strength in refinement, default = 16
7-5	R/W		reserved
4-0	R/W	8	reg_mcdi_refbsthd0. threshold 0 for detect block strength in refinement, default = 8

Table 9-956 MCDI\_REF\_ERR\_GAIN0 0x2f1b

Bit(s)	R/W	Default	Description
31	R/W		reserved
30-24	R/W	48	reg_mcdi_referrnbrdstgain. neighoring mv distances gain for err calc. in ref, normalized to 8 as '1', default = 48
23-20	R/W		reserved
19-16	R/W	4	reg_mcdi_referrbsgain. bs gain for err calc. in ref, normalized to 8 as '1', default = 4

Bit(s)	R/W	Default	Description
15	R/W		reserved
14-8	R/W	64	reg_mcdi_referrbadwgain. badw gain for err calc. in ref, normalized to 8 as '1', default = 64
7-4	R/W		reserved
3-0	R/W	4	reg_mcdi_referrsadgain. sad gain for err calc. in ref, normalized to 8 as '1', default = 4

Table 9-957 MCDI\_REF\_ERR\_GAIN1 0x2f1c

Bit(s)	R/W	Default	Description
31-20	R/W		reserved
19-16	R/W	4	reg_mcdi_referrchkegegain. check edge gain for err calc. in ref, normalized to 8 as '1', default = 4
15-12	R/W		reserved
11-8	R/W	0	reg_mcdi_referrlmvgain. (locked) lmv gain for err calc. in ref, normalized to 8 as '1', default = 0
7-4	R/W		reserved
3-0	R/W	0	reg_mcdi_referrgmvgain. (locked) gmvgain for err calc. in ref, normalized to 8 as '1', default = 0

Table 9-958 MCDI\_REF\_ERR\_FRQ\_CHK 0x2f1d

Bit(s)	R/W	Default	Description
31-28	R/W		reserved
27-24	R/W	10	reg_mcdi_referrfrqgain. gain for mv frequency, normalized to 4 as '1', default = 10
23-21	R/W		reserved
20-16	R/W	31	reg_mcdi_referrfrqmax. max gain for mv frequency check, default = 31
15	R/W		reserved
14-12	R/W	3	reg_mcdi_ref_errfrqmvdifthd2. mv dif threshold 2 (<) for mv frequency check, default = 3
11	R/W		reserved
10-8	R/W	2	reg_mcdi_ref_errfrqmvdifthd1. mv dif threshold 1 (<) for mv frequency check, default = 2
7	R/W		reserved
6-4	R/W	1	reg_mcdi_ref_errfrqmvdifthd0. mv dif threshold 0 (<) for mv frequency check, default = 1
3-0	R/W		reserved

**Table 9-959 MCDI\_QME\_LPF\_MSK 0x2f1e**

Bit(s)	R/W	Default	Description
31-28	R/W		reserved
27-24	R/W	7	reg_mcdi_qmechkedgelpfmsk0. lpf mask0 for chk edge in qme, 0~8, msk1 = (8-msk0), normalized to 8 as '1', default = 7
23-20	R/W		reserved
19-16	R/W	7	reg_mcdi_qmebslpfmsk0. lpf mask0 for bs in qme, 0~8, msk1 = (8-msk0), normalized to 8 as '1', default = 7
15-12	R/W		reserved
11-8	R/W	7	reg_mcdi_qmebadwlpfmsk0. lpf mask0 for badw in qme, 0~8, msk1 = (8-msk0), normalized to 8 as '1', default = 7
7-4	R/W		reserved
3-0	R/W	7	reg_mcdi_qmesadlpfmsk0. lpf mask0 for sad in qme, 0~8, msk1 = (8-msk0), normalized to 8 as '1', default = 7

**Table 9-960 MCDI\_REL\_DIF\_THD\_02 0x2f1f**

Bit(s)	R/W	Default	Description
31-24	R/W		reserved.
23-16	R/W	9	reg_mcdi_reldifhd2. thd (<) for (hdif+vdif), default = 9
15-8	R/W	5	reg_mcdi_reldifhd1. thd (<) for (vdif), default = 5
7-0	R/W	48	reg_mcdi_reldifhd0. thd (>=) for (hdif-vdif), default = 48

**Table 9-961 MCDI\_REL\_DIF\_THD\_34 0x2f20**

Bit(s)	R/W	Default	Description
31-16	R/W		reserved.
15-8	R/W	255	reg_mcdi_reldifhd4. thd (<) for (hdif), default = 255
7-0	R/W	48	reg_mcdi_reldifhd3. thd (>=) for (vdif-hdif), default = 48

**Table 9-962 MCDI\_REL\_BADW\_GAIN\_OFFST\_01 0x2f21**

Bit(s)	R/W	Default	Description
31-24	R/W	0	reg_mcdi_relbadwoffst1. offset for badw adj, for flat block, -128~127, default = 0
23-16	R/W	128	reg_mcdi_relbadwgain1. gain for badw adj, for flat block, default = 128
15-8	R/W	0	reg_mcdi_relbadwoffst0. offset for badw adj, for vertical block, -128~127, default = 0
7-0	R/W	160	reg_mcdi_relbadwgain0. gain for badw adj, for vertical block, default = 160

**Table 9-963 MCDI\_REL\_BADW\_GAIN\_OFFST\_23 0x2f22**

Bit(s)	R/W	Default	Description
31-24	R/W	0	reg_mcdi_relbawoffst3. offset for badw adj, for other block, -128~127, default = 0
23-16	R/W	48	reg_mcdi_relbawgain3. gain for badw adj, for other block, default = 48
15-8	R/W	0	reg_mcdi_relbawoffst2. offset for badw adj, for horizontal block, -128~127, default = 0
7-0	R/W	48	reg_mcdi_relbawgain2. gain for badw adj, for horizontal block, default = 48

**Table 9-964 MCDI\_REL\_BADW\_THD\_GAIN\_OFFST 0x2f23**

Bit(s)	R/W	Default	Description
31-23	R/W		reserved.
22-16	R/W	0	reg_mcdi_relbawoffst. offset for badw thd adj, -64~63, default = 0
15-8	R/W		reserved.
7-0	R/W	16	reg_mcdi_relbawthdgain. gain0 for badw thd adj, normalized to 16 as '1', default = 16

**Table 9-965 MCDI\_REL\_BADW\_THD\_MIN\_MAX 0x2f24**

Bit(s)	R/W	Default	Description
31-18	R/W		reserved.
17-8	R/W	256	reg_mcdi_relbawthdmax. max for badw thd adj, default = 256
7-0	R/W	16	reg_mcdi_relbawthdmin. min for badw thd adj, default = 16

**Table 9-966 MCDI\_REL\_SAD\_GAIN\_OFFST\_01 0x2f25**

Bit(s)	R/W	Default	Description
31-24	R/W	0	reg_mcdi_relsadoffst1. offset for sad adj, for flat block, -128~127, default = 0
23-20	R/W		reserved.
19-16	R/W	8	reg_mcdi_relsadgain1. gain for sad adj, for flat block, normalized to 8 as '1', default = 8
15-8	R/W	0	reg_mcdi_relsadoffst0. offset for sad adj, for vertical block, -128~127, default = 0
7-4	R/W		reserved.
3-0	R/W	6	reg_mcdi_relsadgain0. gain for sad adj, for vertical block, normalized to 8 as '1', default = 6

**Table 9-967 MCDI\_REL\_SAD\_GAIN\_OFFST\_23 0x2f26**

Bit(s)	R/W	Default	Description
31-24	R/W	0	reg_mcdi_relsadoffst3. offset for sad adj, for other block, -128~127, default = 0
23-20	R/W		reserved.



Bit(s)	R/W	Default	Description
19-16	R/W	8	reg_mcdi_relsadgain3. gain for sad adj, for other block, normalized to 8 as '1', default = 8
15-8	R/W	0	reg_mcdi_relsadoffst2. offset for sad adj, for horizontal block, -128~127, default = 0
7-4	R/W		reserved.
3-0	R/W	12	reg_mcdi_relsadgain2. gain for sad adj, for horizontal block, normalized to 8 as '1', default = 12

Table 9-968 MCDI\_REL\_SAD\_THD\_GAIN\_OFFST 0x2f27

Bit(s)	R/W	Default	Description
31-24	R/W		reserved.
23-16	R/W	0	reg_mcdi_relsadoffst. offset for sad thd adj, -128~127, default = 0
15-10	R/W		reserved.
9-0	R/W	42	reg_mcdi_relsadthdgain. gain for sad thd adj, $21 \times 2 / 16$ , normalized to 16 as '1', default = 42

Table 9-969 MCDI\_REL\_SAD\_THD\_MIN\_MAX 0x2f28

Bit(s)	R/W	Default	Description
31-27	R/W		reserved.
26-16	R/W	672	reg_mcdi_relsadthdmax. max for sad thd adj, $21 \times 32$ , default = 672
15-9	R/W		reserved.
8-0	R/W	42	reg_mcdi_relsadthdmin. min for sad thd adj, $21 \times 2$ , default = 42

Table 9-970 MCDI\_REL\_DET\_GAIN\_00 0x2f29

Bit(s)	R/W	Default	Description
31-21	R/W		reserved.
20-16	R/W	8	reg_mcdi_reldetbsgain0. gain0 (gmv locked) for bs, for det. calc. normalized to 16 as '1', default = 8
15-14	R/W		reserved.
13-8	R/W	12	reg_mcdi_reldetbadwgain0. gain0 (gmv locked) for badw, for det. calc. normalized to 16 as '1', default = 12
7-5	R/W		reserved.
4-0	R/W	8	reg_mcdi_reldetsadgain0. gain0 (gmv locked) for qsad, for det. calc. normalized to 16 as '1', default = 8

**Table 9-971 MCDI\_REL\_DET\_GAIN\_01 0x2f2a**

Bit(s)	R/W	Default	Description
31-14	R/W		reserved.
12-8	R/W	2	reg_mcdi_reldetchkedggain0. gain0 (gmv locked) for chk_edge, for det. calc. normalized to 16 as '1', default = 2
7	R/W		reserved.
6-0	R/W	24	reg_mcdi_reldetnbrdstgain0. gain0 (gmv locked) for neighoring dist, for det. calc. normalized to 16 as '1', default = 24

**Table 9-972 MCDI\_REL\_DET\_GAIN\_10 0x2f2b**

Bit(s)	R/W	Default	Description
31-21	R/W		reserved.
20-16	R/W	0	reg_mcdi_reldetbsgain1. gain1 (lmv locked) for bs, for det. calc. normalized to 16 as '1', default = 0
15-14	R/W		reserved.
13-8	R/W	8	reg_mcdi_reldetbadwgain1. gain1 (lmv locked) for badw, for det. calc. normalized to 16 as '1', default = 8
7-5	R/W		reserved.
4-0	R/W	8	reg_mcdi_reldetsadgain1. gain1 (lmv locked) for qsad, for det. calc. normalized to 16 as '1', default = 8

**Table 9-973 MCDI\_REL\_DET\_GAIN\_11 0x2f2c**

Bit(s)	R/W	Default	Description
31-14	R/W		reserved.
12-8	R/W	0	reg_mcdi_reldetchkedggain1. gain1 (lmv locked) for chk_edge, for det. calc. normalized to 16 as '1', default = 0
7	R/W		reserved.
6-0	R/W	24	reg_mcdi_reldetnbrdstgain1. gain1 (lmv locked) for neighoring dist, for det. calc. normalized to 16 as '1', default = 24

**Table 9-974 MCDI\_REL\_DET\_GAIN\_20 0x2f2d**

Bit(s)	R/W	Default	Description
31-21	R/W		reserved.
20-16	R/W	12	reg_mcdi_reldetbsgain2. gain2 (no locked) for bs, for det. calc. normalized to 16 as '1', default = 12
15-14	R/W		reserved.
13-8	R/W	32	reg_mcdi_reldetbadwgain2. gain2 (no locked) for badw, for det. calc. normalized to 16 as '1', default = 32
7-5	R/W		reserved.
4-0	R/W	16	reg_mcdi_reldetsadgain2. gain2 (no locked) for qsad, for det. calc. normalized to 16 as '1', default = 16

Table 9-975 MCDI\_REL\_DET\_GAIN\_21 0x2f2e

Bit(s)	R/W	Default	Description
31-26	R/W		reserved
25-16	R/W	0	reg_mcdi_reldetoffst. offset for rel calculation, for det. calc. -512~511, default = 0
15-14	R/W		reserved.
12-8	R/W	10	reg_mcdi_reldetchkedgegain2. gain2 (no locked) for chk_edge, for det. calc. normalized to 16 as '1', default = 10
7	R/W		reserved.
6-0	R/W	32	reg_mcdi_reldetnbrdstgain2. gain2 (no locked) for neighoring dist, for det. calc. normalized to 16 as '1', default = 32

Table 9-976 MCDI\_REL\_DET\_GMV\_DIF\_CHK 0x2f2f

Bit(s)	R/W	Default	Description
31-24	R/W		reserved.
23-16	R/W	0	reg_mcdi_reldetgmvlthd. flat thd ( $\geq$ ) for gmV lock decision, default = 0
15	R/W		reserved.
14-12	R/W	3	reg_mcdi_reldetgmvdifthd. dif thd ( $\geq$ ) for current mv different from gmV for gmV dif check, actually used in Lmv lock check, default = 3
11	R/W		reserved.
10-8	R/W	1	reg_mcdi_reldetgmvdifmin. min mv dif for gmV dif check, default = 1, note: dif between reg_mcdi_rel_det_gmv_dif_max and reg_mcdi_rel_det_gmv_dif_min should be; 0,1,3,7, not work for others
7-4	R/W	4	reg_mcdi_reldetgmvdifmax. max mv dif for gmV dif check, default = 4
3-1	R/W		reserved
0	R/W	0	reg_mcdi_reldetgmvdifmvmode. mv mode used for gmV dif check, 0- use refmv, 1: use qmv, default = 0

Table 9-977 MCDI\_REL\_DET\_LMV\_DIF\_CHK 0x2f30

Bit(s)	R/W	Default	Description
31-24	R/W		reserved.
23-16	R/W	12	reg_mcdi_reldetlmvflthd. flat thd ( $\geq$ ) for lmv lock decision, default = 12
15-14	R/W		reserved.
13-12	R/W	1	reg_mcdi_reldetlmvlockchkmode. lmv lock check mode, 0:cur Lmv, 1: cur & (last   next), 2: last & cur & next Lmv, default = 1
11	R/W		reserved.
10-8	R/W	1	reg_mcdi_reldetlmvdifmin. min mv dif for lmv dif check, default = 1, note: dif between reg_mcdi_rel_det_lmv_dif_max and reg_mcdi_rel_det_lmv_dif_min should be; 0,1,3,7, not work for others
7-4	R/W	4	reg_mcdi_reldetlmvdifmax. max mv dif for lmv dif check, default = 4

Bit(s)	R/W	Default	Description
3-1	R/W		reserved
0	R/W	0	reg_mcdi_reldetlmvdfmvmode. mv mode used for lmv dif check, 0- use refmv, 1: use qmv, default = 0

Table 9-978 MCDI\_REL\_DET\_FRQ\_CHK 0x2f31

Bit(s)	R/W	Default	Description
31-12	R/W		reserved.
11-8	R/W	10	reg_mcdi_reldetfrqgain. gain for frequency check, normalized to 4 as '1', default = 10
7-5	R/W		reserved
4-0	R/W	31	reg_mcdi_reldetfrqmax. max value for frequency check, default = 31

Table 9-979 MCDI\_REL\_DET\_PD22\_CHK 0x2f32

Bit(s)	R/W	Default	Description
31-18	R/W		reserved.
30-21	R/W	512	reg_mcdi_reldetpd22chkoffst1. offset for pd22 check happened, default = 512
20-16	R/W	12	reg_mcdi_reldetpd22chkgain1. gain for pd22 check happened, normalized to 8 as '1', default = 12
14-5	R/W	512	reg_mcdi_reldetpd22chkoffst0. offset for pd22 check happened, default = 512
4-0	R/W	12	reg_mcdi_reldetpd22chkgain0. gain for pd22 check happened, normalized to 8 as '1', default = 12

Table 9-980 MCDI\_REL\_DET\_RPT\_CHK\_ROW 0x2f33

Bit(s)	R/W	Default	Description
31-27	R/W		reserved
26-16	R/W	2047	reg_mcdi_reldetrptchkendrow. end row (<) number for repeat check, default = 2047
15-11	R/W		reserved
10-0	R/W	0	reg_mcdi_reldetrptchkstartrow. start row (>=) number for repeat check, default = 0

Table 9-981 MCDI\_REL\_DET\_RPT\_CHK\_GAIN\_QMV 0x2f34

Bit(s)	R/W	Default	Description
31-30	R/W		reserved
29-24	R/W	15	reg_mcdi_reldetrptchkqmvmax. max thd (<) of abs qmv for repeat check, default = 15, note that quarter mv's range is -63~63
23-22	R/W		reserved
21-16	R/W	10	reg_mcdi_reldetrptchkqmvmin. min thd (>=) of abs qmv for repeat check, default = 10, note that quarter mv's range is -63~63

Bit(s)	R/W	Default	Description
15	R/W		reserved/
14-4	R/W	512	reg_mcdi_reldetrptchkoffst. offset for repeat check, default = 512
3-0	R/W	4	reg_mcdi_reldetrptchkgain. gain for repeat check, normalized to 8 as '1', default = 4

Table 9-982 MCDI\_REL\_DET\_RPT\_CHK\_THD\_0 0x2f35

Bit(s)	R/W	Default	Description
31-24	R/W		reserved
23-16	R/W	255	reg_mcdi_reldetrptchkzerosadthd. zero sad thd (<) for repeat check, default = 255
15-14	R/W		reserved.
13-8	R/W	16	reg_mcdi_reldetrptchkzerobadwthd. zero badw thd (>=) for repeat check, default = 16
7-4	R/W		reserved
3-0	R/W	5	reg_mcdi_reldetrptchkfrqdifthd. frequency dif thd (<) for repeat check, 0~10, default = 5

Table 9-983 MCDI\_REL\_DET\_RPT\_CHK\_THD\_1 0x2f36

Bit(s)	R/W	Default	Description
31-16	R/W		reserved
15-8	R/W	16	reg_mcdi_reldetrptchkvdifthd. vertical dif thd (<) for repeat check, default = 16
7-0	R/W	16	reg_mcdi_reldetrptchkhdifthd. horizontal dif thd (>=) for repeat check, default = 16

Table 9-984 MCDI\_REL\_DET\_LPF\_DIF\_THD 0x2f37

Bit(s)	R/W	Default	Description
31-24	R/W	9	reg_mcdi_reldetlpdifthd3. hdif thd (<) for lpf selection of horizontal block, default = 9
23-16	R/W	48	reg_mcdi_reldetlpdifthd2. vdif-hdif thd (>=) for lpf selection of horizontal block, default = 48
15-8	R/W	9	reg_mcdi_reldetlpdifthd1. vdif thd (<) for lpf selection of vertical block, default = 9
7-0	R/W	48	reg_mcdi_reldetlpdifthd0. hdif-vdif thd (>=) for lpf selection of vertical block, default = 48

Table 9-985 MCDI\_REL\_DET\_LPF\_MSK\_00\_03 0x2f38

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28-24	R/W	1	reg_mcdi_reldetlpfmsk03. det lpf mask03 for gmv/lmv locked mode, 0~16, default = 1

Bit(s)	R/W	Default	Description
23-21	R/W		reserved
20-16	R/W	1	reg_mcdi_reldetlpfmsk02. det lpf mask02 for gmv/lmv locked mode, 0~16, default = 1
15-13	R/W		reserved
12-8	R/W	5	reg_mcdi_reldetlpfmsk01. det lpf mask01 for gmv/lmv locked mode, 0~16, default = 5
7-5	R/W		reserved
4-0	R/W	8	reg_mcdi_reldetlpfmsk00. det lpf mask00 for gmv/lmv locked mode, 0~16, default = 8

Table 9-986 MCDI\_REL\_DET\_LPF\_MSK\_04\_12 0x2f39

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28-24	R/W	0	reg_mcdi_reldetlpfmsk12. det lpf mask12 for vertical blocks, 0~16, default = 0
23-21	R/W		reserved
20-16	R/W	0	reg_mcdi_reldetlpfmsk11. det lpf mask11 for vertical blocks, 0~16, default = 0
15-13	R/W		reserved
12-8	R/W	16	reg_mcdi_reldetlpfmsk10. det lpf mask10 for vertical blocks, 0~16, default = 16
7-5	R/W		reserved
4-0	R/W	1	reg_mcdi_reldetlpfmsk04. det lpf mask04 for gmv/lmv locked mode, 0~16, default = 1

Table 9-987 MCDI\_REL\_DET\_LPF\_MSK\_13\_21 0x2f3a

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28-24	R/W	6	reg_mcdi_reldetlpfmsk21. det lpf mask21 for horizontal blocks, 0~16, default = 6
23-21	R/W		reserved
20-16	R/W	8	reg_mcdi_reldetlpfmsk20. det lpf mask20 for horizontal blocks, 0~16, default = 8
15-13	R/W		reserved
12-8	R/W	0	reg_mcdi_reldetlpfmsk14. det lpf mask14 for vertical blocks, 0~16, default = 0
7-5	R/W		reserved
4-0	R/W	0	reg_mcdi_reldetlpfmsk13. det lpf mask13 for vertical blocks, 0~16, default = 0

Table 9-988 MCDI\_REL\_DET\_LPF\_MSK\_22\_30 0x2f3b

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28-24	R/W	16	reg_mcdi_reldetlpfmsk30. det lpf mask30 for other blocks, 0~16, default = 16

Bit(s)	R/W	Default	Description
23-21	R/W		reserved
20-16	R/W	1	reg_mcdi_reldetlpfmsk24. det lpf mask24 for horizontal blocks, 0~16, default = 1
15-13	R/W		reserved
12-8	R/W	0	reg_mcdi_reldetlpfmsk23. det lpf mask23 for horizontal blocks, 0~16, default = 0
7-5	R/W		reserved
4-0	R/W	1	reg_mcdi_reldetlpfmsk22. det lpf mask22 for horizontal blocks, 0~16, default = 1

Table 9-989 MCDI\_REL\_DET\_LPF\_MSK\_31\_34 0x2f3c

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28-24	R/W	0	reg_mcdi_reldetlpfmsk34. det lpf mask34 for other blocks, 0~16, default = 0
23-21	R/W		reserved
20-16	R/W	0	reg_mcdi_reldetlpfmsk33. det lpf mask33 for other blocks, 0~16, default = 0
15-13	R/W		reserved
12-8	R/W	0	reg_mcdi_reldetlpfmsk32. det lpf mask32 for other blocks, 0~16, default = 0
7-5	R/W		reserved
4-0	R/W	0	reg_mcdi_reldetlpfmsk31. det lpf mask31 for other blocks, 0~16, default = 0

Table 9-990 MCDI\_REL\_DET\_MIN 0x2f3d

Bit(s)	R/W	Default	Description
31-7	R/W		reserved
6-0	R/W	16	reg_mcdi_reldetmin. min of detected value, default = 16

Table 9-991 MCDI\_REL\_DET\_LUT\_0\_3 0x2f3e

Bit(s)	R/W	Default	Description
31-24	R/W	8	reg_mcdi_reldetmaplut3. default = 8
23-16	R/W	4	reg_mcdi_reldetmaplut2. default = 4
15-8	R/W	2	reg_mcdi_reldetmaplut1. default = 2
7-0	R/W	0	reg_mcdi_reldetmaplut0. default = 0

Table 9-992 MCDI\_REL\_DET\_LUT\_4\_7 0x2f3f

Bit(s)	R/W	Default	Description
31-24	R/W	64	reg_mcdi_reldetmaplut7. default = 64
23-16	R/W	48	reg_mcdi_reldetmaplut6. default = 48

Bit(s)	R/W	Default	Description
15-8	R/W	32	reg_mcdi_reldetmaplut5. default = 32
7-0	R/W	16	reg_mcdi_reldetmaplut4. default = 16

Table 9-993 MCDI\_REL\_DET\_LUT\_8\_11 0x2f40

Bit(s)	R/W	Default	Description
31-24	R/W	160	reg_mcdi_reldetmaplut11. default = 160
23-16	R/W	128	reg_mcdi_reldetmaplut10. default = 128
15-8	R/W	96	reg_mcdi_reldetmaplut9. default = 96
7-0	R/W	80	reg_mcdi_reldetmaplut8. default = 80

Table 9-994 MCDI\_REL\_DET\_LUT\_12\_15 0x2f41

Bit(s)	R/W	Default	Description
31-24	R/W	255	reg_mcdi_reldetmaplut15. default = 255
23-16	R/W	240	reg_mcdi_reldetmaplut14. default = 240
15-8	R/W	224	reg_mcdi_reldetmaplut13. default = 224
7-0	R/W	192	reg_mcdi_reldetmaplut12. default = 192

Table 9-995 MCDI\_REL\_DET\_COL\_CFD\_THD 0x2f42

Bit(s)	R/W	Default	Description
31-24	R/W	5	reg_mcdi_reldetcolcfdfthd. thd for flat smaller than (<) of column cofidence, default = 5
23-16	R/W	160	reg_mcdi_reldetcolcfdthd1. thd for rel larger than (>=) in rel calc. mode col confidence without gmv locking, default = 160
15-8	R/W	100	reg_mcdi_reldetcolcfdthd0. thd for rel larger than (>=) in rel calc. mode col confidence when gmv locked, default = 100
7-2	R/W	16	reg_mcdi_reldetcolcfdbadwthd. thd for badw larger than (>=) in qbadw calc. mode of column cofidence, default = 16
1	R/W		reserved
0	R/W	0	reg_mcdi_reldetcolcfdcacmode. calc. mode for column cofidence, 0- use rel, 1: use qbadw, default = 0

Table 9-996 MCDI\_REL\_DET\_COL\_CFD\_AVG\_LUMA 0x2f43

Bit(s)	R/W	Default	Description
31-24	R/W	235	reg_mcdi_reldetcolcfdavgmin1. avg luma min1 (>=) for column cofidence, valid between 16~235, default = 235
23-16	R/W	235	reg_mcdi_reldetcolcfdavgmax1. avg luma max1 (<) for column cofidence, valid between 16~235, default = 235



Bit(s)	R/W	Default	Description
15-8	R/W	16	reg_mcdi_reldetcolcfavgmin0. avg luma min0 ( $\geq$ ) for column confidence, valid between 16~235, default = 16
7-0	R/W	21	reg_mcdi_reldetcolcfavgmax0. avg luma max0 ( $<$ ) for column confidence, valid between 16~235, default = 21

**Table 9-997 MCDI\_REL\_DET\_BAD\_THD\_0 0x2f44**

Bit(s)	R/W	Default	Description
31-16	R/W		reserved
15-8	R/W	120	reg_mcdi_reldetbadsadthd. thd ( $\geq$ ) for bad sad, default = 120 (480/4)
7-6	R/W		reserved
5-0	R/W	12	reg_mcdi_reldetbadbadwthd. thd ( $\geq$ ) for bad badw, 0~42, default = 12

**Table 9-998 MCDI\_REL\_DET\_BAD\_THD\_1 0x2f45**

Bit(s)	R/W	Default	Description
31-24	R/W		reserved
23-16	R/W	4	reg_mcdi_reldetbadrelflthd. thd ( $\geq$ ) of flat for bad rel detection, default = 4
15-8	R/W	160	reg_mcdi_reldetbadrelthd1. thd ( $\geq$ ) for bad rel without gmv/lmv locked, default = 160
7-0	R/W	120	reg_mcdi_reldetbadrelthd0. thd ( $\geq$ ) for bad rel with gmv/lmv locked, default = 120

**Table 9-999 MCDI\_PD22\_CHK\_THD 0x2f46**

Bit(s)	R/W	Default	Description
31-25	R/W		reserved
24-16	R/W	64	reg_mcdi_pd22chksaddifthd. sad dif thd ( $\geq$ ) for (pd22chksad - qsad) for pd22 check, default = 64
15-14	R/W		reserved
13-8	R/W	2	reg_mcdi_pd22chkqmvthd. thd ( $\geq$ ) of abs qmv for pd22 check, default = 2
7-0	R/W	4	reg_mcdi_pd22chkflthd. thd ( $\geq$ ) of flat for pd22 check, default = 4

**Table 9-1000 MCDI\_PD22\_CHK\_GAIN\_OFFST\_0 0x2f47**

Bit(s)	R/W	Default	Description
31-24	R/W	0	reg_mcdi_pd22chkedgeoffst0. offset0 of pd22chkedge from right film22 phase, -128~127, default = 0
23-21	R/W		reserved
20-16	R/W	16	reg_mcdi_pd22chkedgegain0. gain0 of pd22chkedge from right film22 phase, normalized to 16 as '1', default = 16
15-12	R/W		reserved

Bit(s)	R/W	Default	Description
11-8	R/W	0	reg_mcdi_pd22chkbadwoffst0. offset0 of pd22chkbadw from right film22 phase, -8~7, default = 0
7-5	R/W		reserved
4-0	R/W	8	reg_mcdi_pd22chkbadwgain0. gain0 of pd22chkbadw from right film22 phase, normalized to 16 as '1', default = 8

Table 9-1001 MCDI\_PD22\_CHK\_GAIN\_OFFST\_1 0x2f48

Bit(s)	R/W	Default	Description
31-24	R/W	0	reg_mcdi_pd22chkedgeoffst1. offset1 of pd22chkedge from right film22 phase, -128~127, default = 0
23-21	R/W		reserved
20-16	R/W	16	reg_mcdi_pd22chkedgegain1. gain1 of pd22chkedge from right film22 phase, normalized to 16 as '1', default = 16
15-12	R/W		reserved
11-8	R/W	0	reg_mcdi_pd22chkbadwoffst1. offset1 of pd22chkbadw from right film22 phase, -8~7, default = 0
7-5	R/W		reserved
4-0	R/W	12	reg_mcdi_pd22chkbadwgain1. gain1 of pd22chkbadw from right film22 phase, normalized to 16 as '1', default = 12

Table 9-1002 MCDI\_LMV\_LOCK\_CNT\_THD\_GAIN 0x2f49

Bit(s)	R/W	Default	Description
31-20	R/W		reserved
19-16	R/W	6	reg_mcdi_lmvlckcntmax. max lmv lock count number, default = 6
15-12	R/W	0	reg_mcdi_lmvlckcntoffst. offset for lmv lock count, -8~7, default = 0
11-8	R/W	8	reg_mcdi_lmvlckcntgain. gain for lmv lock count, normalized 8 as '1', 15 is set to 16, default = 8
7-5	R/W		reserved
4-0	R/W	4	reg_mcdi_lmvlckcntthd. lmv count thd (>=) before be locked, 1~31, default = 4

Table 9-1003 MCDI\_LMV\_LOCK\_ABS\_DIF\_THD 0x2f4a

Bit(s)	R/W	Default	Description
31-27	R/W		reserved
26-24	R/W	1	reg_mcdi_lmvlckdifthd2. lmv dif thd for third part, before locked, default = 1
23	R/W		reserved
22-20	R/W	1	reg_mcdi_lmvlckdifthd1. lmv dif thd for second part, before locked, default = 1
19	R/W		reserved
18-16	R/W	1	reg_mcdi_lmvlckdifthd0. lmv dif thd for first part, before locked, default = 1

Bit(s)	R/W	Default	Description
15-13	R/W		reserved
12-8	R/W	24	reg_mcdi_lmvlckabsmax. max abs (<) of lmv to be locked, default = 24
7-5	R/W		reserved
4-0	R/W	1	reg_mcdi_lmvlckabsmin. min abs (>=) of lmv to be locked, default = 1

Table 9-1004 MCDI\_LMV\_LOCK\_ROW 0x2f4b

Bit(s)	R/W	Default	Description
31-27	R/W		reserved
26-16	R/W	2047	reg_mcdi_lmvlckendrow. end row (<) for lmv lock, default = 2047
15-11	R/W		reserved
10-0	R/W	0	reg_mcdi_lmvlckstartrow. start row (>=) for lmv lock, default = 0

Table 9-1005 MCDI\_LMV\_LOCK\_RT\_MODE 0x2f4c

Bit(s)	R/W	Default	Description
31-27	R/W		reserved
26-24	R/W	2	reg_mcdi_lmvlckextmode. extend lines for lmv lock check, check how many lines for lmv locking, default = 2
23-16	R/W	32	reg_mcdi_lmvlckfltcntrt. ratio of flt cnt for lock check, normalized 256 as '1', 255 is set to 256, default = 32
15-8	R/W	48	reg_mcdi_lmvlcklmcntrt1. ratio when use non-zero lmv for lock check, normalized 256 as '1', 255 is set to 256, default = 48
7-0	R/W	106	reg_mcdi_lmvlcklmcntrt0. ratio when use max lmv for lock check, normalized 256 as '1', 255 is set to 256, default = 106

Table 9-1006 MCDI\_GMV\_LOCK\_CNT\_THD\_GAIN 0x2f4d

Bit(s)	R/W	Default	Description
31-20	R/W		reserved
19-16	R/W	6	reg_mcdi_gmvlockcntmax. max gmv lock count number, default = 6
15-12	R/W	0	reg_mcdi_gmvlockcntoffst. offset for gmv lock count, -8~7, default = 0
11-8	R/W	8	reg_mcdi_gmvlockcntgain. gain for gmv lock count, normalized 8 as '1', 15 is set to 16, default = 8
7-5	R/W		reserved
4-0	R/W	4	reg_mcdi_gmvlockcntthd. gmv count thd (>=) before be locked, 1~31, default = 4

**Table 9-1007 MCDI\_GMV\_LOCK\_ABS\_DIF\_THD 0x2f4e**

Bit(s)	R/W	Default	Description
31-27	R/W		reserved
26-24	R/W	3	reg_mcdi_gmvlockdifthd2. gmv dif thd for third part, before locked, default = 3
23	R/W		reserved
22-20	R/W	2	reg_mcdi_gmvlockdifthd1. gmv dif thd for second part, before locked, default = 2
19	R/W		reserved
18-16	R/W	1	reg_mcdi_gmvlockdifthd0. gmv dif thd for first part, before locked, default = 1
15-13	R/W		reserved
12-8	R/W	15	reg_mcdi_gmvlockabsmax. max abs of gmv to be locked, default = 15
7-5	R/W		reserved
4-0	R/W	1	reg_mcdi_gmvlockabsmin. min abs of gmv to be locked, default = 1

**Table 9-1008 MCDI\_HIGH\_VERT\_FRQ\_DIF\_THD 0x2f4f**

Bit(s)	R/W	Default	Description
31-0	R/W	103680	reg_mcdi_highvertfrqfldavgdifthd. high_vert_frq field average luma dif thd (>=), $3 * \text{Blk\_Width} * \text{Blk\_Height}$ , set by software, default = 103680

**Table 9-1009 MCDI\_HIGH\_VERT\_FRQ\_DIF\_DIF\_THD 0x2f50**

Bit(s)	R/W	Default	Description
31-0	R/W	103680	reg_mcdi_highvertfrqfldavgdifdifthd. high_vert_frq field average luma dif's dif thd (<), $3 * \text{Blk\_Width} * \text{Blk\_Height}$ , set by software, default = 103680

**Table 9-1010 MCDI\_HIGH\_VERT\_FRQ\_RT\_GAIN 0x2f51**

Bit(s)	R/W	Default	Description
31-20	R/W		reserved
19-16	R/W	4	reg_mcdi_highvertfrqcntthd. high_vert_frq count thd (>=) before locked, 1~31, default = 4
15-8	R/W	24	reg_mcdi_highvertfrqbadsadrt. ratio for high_vert_frq bad sad count, normalized 256 as '1', 255 is set to 256, default = 24
7-0	R/W	130	reg_mcdi_highvertfrqbadbadwrt. ratio for high_vert_frq badw count, normalized 256 as '1', 255 is set to 256, default = 130

**Table 9-1011 MCDI\_MOTION\_PARADOX\_THD 0x2f52**

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28-24	R/W	4	reg_mcdi_motionparadoxcntthd. motion paradox count thd (>=) before locked, 1~31, default = 4
23-22	R/W		reserved

Bit(s)	R/W	Default	Description
21-16	R/W	32	reg_mcdi_motionparadoxgmvthd. abs gmv thd (<) of motion paradox, 0~32, note that 32 means invalid gmv, be careful, default = 32
15-0	R/W		reserved

Table 9-1012 MCDI\_MOTION\_PARADOX\_RT 0x2f53

Bit(s)	R/W	Default	Description
31-24	R/W		reserved
23-16	R/W	24	reg_mcdi_motionparadoxbadsadrt. ratio for field bad sad count of motion paradox, normalized 256 as '1', 255 is set to 256, default = 24
15-8	R/W	120	reg_mcdi_motionparadoxbadrelrt. ratio for field bad reliability count of motion paradox, normalized 256 as '1', 255 is set to 256, default = 120
7-0	R/W	218	reg_mcdi_motionparadoxmntnrt. ratio for field motion count of motion paradox, normalized 256 as '1', 255 is set to 256, default = 218

Table 9-1013 MCDI\_MOTION\_REF\_THD 0x2f54

Bit(s)	R/W	Default	Description
31-24	R/W		reserved
23-20	R/W	15	reg_mcdi_motionrefoffst. motion ref additive offset, default = 15
19-16	R/W	8	reg_mcdi_motionrefgain. motion ref gain, normalized 8 as '1', default = 8
15-13	R/W		reserved
12-8	R/W	1	reg_mcdi_motionrefrptmvthd. abs thd (>=) of rpt mv (0~31, 32 means invalid) for motion ref, default = 1
7-2	R/W	2	reg_mcdi_motionrefqmvthd. min thd (>=) of abs qmv for motion ref, note that quarter mv's range is -63~63, default = 2
1-0	R/W	1	reg_mcdi_motionreflplfmode. Mv and (8 x repeat flg) 's lpf mode of motion refinement, 0: no lpf, 1: [1 2 1], 2: [1 2 2 2 1], default = 1

Table 9-1014 MCDI\_REL\_COL\_REF\_RT 0x2f55

Bit(s)	R/W	Default	Description
31-8	R/W		reserved
7-0	R/W	135	reg_mcdi_relcolrefrt. ratio for column cofidence level against column number, for refinement, default = 135

Table 9-1015 MCDI\_PD22\_CHK\_THD\_RT 0x2f56

Bit(s)	R/W	Default	Description
31-27	R/W		reserved
26-16	R/W	1	reg_mcdi_pd22chkfltcntrt. ratio for flat count of field pulldown 22 check, normalized 2048 as '1', 2047 is set to 2048, default = 1
15-8	R/W	100	reg_mcdi_pd22chkcncntrt. ratio of pulldown 22 check count, normalized 256 as '1', 255 is set to 256, default = 100

Bit(s)	R/W	Default	Description
7-5	R/W		reserved
4-0	R/W	4	reg_mcdi_pd22chkcntthd. thd (>=) for pd22 count before locked, 1~31, default = 4

Table 9-1016 MCDI\_CHAR\_DET\_DIF\_THD 0x2f57

Bit(s)	R/W	Default	Description
31-24	R/W		reserved
23-16	R/W	64	reg_mcdi_chardetminmaxdifthd. thd (>=) for dif between min and max value, default = 64
15-8	R/W	17	reg_mcdi_chardetmaxdifthd. thd (<) for dif between max value, default = 17
7-0	R/W	17	reg_mcdi_chardetmindifthd. thd (<) for dif between min value, default = 17

Table 9-1017 MCDI\_CHAR\_DET\_CNT\_THD 0x2f58

Bit(s)	R/W	Default	Description
31-21	R/W		reserved
20-16	R/W	18	reg_mcdi_chardettotcntthd. thd (>=) for total count, 0~21, default = 18
15-13	R/W		reserved
12-8	R/W	1	reg_mcdi_chardetmaxcntthd. thd (>=) for max count, 0~21, default = 1
7-5	R/W		reserved
4-0	R/W	1	reg_mcdi_chardetmincntthd. thd (>=) for min count, 0~21, default = 1

Table 9-1018 MCDI\_PD\_22\_CHK\_WND0\_X 0x2f59

Bit(s)	R/W	Default	Description
28-16	R/W	719	reg_mcdi_pd22chkwnd0_x1
12-0	R/W	0	reg_mcdi_pd22chkwnd0_x0

Table 9-1019 MCDI\_PD\_22\_CHK\_WND0\_Y 0x2f5a

Bit(s)	R/W	Default	Description
28-16	R/W	39	reg_mcdi_pd22chkwnd0_y1
12-0	R/W	0	reg_mcdi_pd22chkwnd0_y0

Table 9-1020 MCDI\_PD\_22\_CHK\_WND1\_X 0x2f5b

Bit(s)	R/W	Default	Description
28-16	R/W	719	reg_mcdi_pd22chkwnd1_x1
12-0	R/W	0	reg_mcdi_pd22chkwnd1_x0

**Table 9-1021 MCDI\_PD\_22\_CHK\_WND1\_Y 0x2f5c**

Bit(s)	R/W	Default	Description
28-16	R/W	199	reg_mcdi_pd22chkwnd1_y1
12-0	R/W	40	reg_mcdi_pd22chkwnd1_y0

**Table 9-1022 MCDI\_PD\_22\_CHK\_FRC\_LMV 0x2f5d**

Bit(s)	R/W	Default	Description
10	R/W	1	reg_mcdi_pd22chklmvchk2
9	R/W	0	reg_mcdi_pd22chklmvchk1
8	R/W	0	reg_mcdi_pd22chklmvchk0
6	R/W	0	reg_mcdi_pd22chkfrcpd2
5	R/W	0	reg_mcdi_pd22chkfrcpd1
4	R/W	0	reg_mcdi_pd22chkfrcpd0
2	R/W	1	reg_mcdi_pd22chkfrcvof2
1	R/W	0	reg_mcdi_pd22chkfrcvof1
0	R/W	0	reg_mcdi_pd22chkfrcvof0

**Table 9-1023 MCDI\_PD\_22\_CHK\_FRC\_LMV 0x2f5e**

Bit(s)	R/W	Default	Description
26	R/W	0	reg_mcdi_pd22chkflg2
25	R/W	0	reg_mcdi_pd22chkflg1
24	R/W	0	reg_mcdi_pd22chkflg
23-16	R/W	1	reg_mcdi_pd22chkcnt2
15-8	R/W	0	reg_mcdi_pd22chkcnt1
7-0	R/W	0	reg_mcdi_pd22chkcnt

**Table 9-1024 MCDI\_FIELD\_MV 0x2f60**

Bit(s)	R/W	Default	Description
23-16	R/W		reg_mcdi_fieldgmvcnt
14	R/W		reg_mcdi_fieldgmvlock
13-8	R/W		reg_mcdi_fieldrptmv. last field rpt mv
7-6	R/W		reserved
5-0	R/W		reg_mcdi_fieldgmv. last field gmv

**Table 9-1025 MCDI\_FIELD\_HVF\_PRDX\_CNT 0x2f61**

Bit(s)	R/W	Default	Description
31-24	R/W		reg_mcdi_motionparadoxcnt.
23-17	R/W		reserved
16	R/W		reg_mcdi_motionparadoxflg.
15-8	R/W		reg_mcdi_highvertfrqcnt.
7-4	R/W		reserved
3-2	R/W		reg_mcdi_highvertfrqphase.
1	R/W		reserved
0	R/W		reg_mcdi_highvertfrqflg.

**Table 9-1026 MCDI\_FIELD\_LUMA\_AVG\_SUM\_0 0x2f62**

Bit(s)	R/W	Default	Description
31-0	R/W		reg_mcdi fld_luma_avg_sum0.

**Table 9-1027 MCDI\_FIELD\_LUMA\_AVG\_SUM\_1 0x2f63**

Bit(s)	R/W	Default	Description
31-0	R/W		reg_mcdi fld_luma_avg_sum1.

**Table 9-1028 MCDI\_YCBCR\_BLEND\_CTRL 0x2f64**

Bit(s)	R/W	Default	Description
31-16	R/W		reserved
15-8	R/W	0	reg_mcdi_ycbcrblendgain. ycbcr blending gain for cbc in ycbcr. default = 0
7-2	R/W		reserved.
1-0	R/W	2	reg_mcdi_ycbcrblendmode. 0:y+cmb(cb,cr), 1:med(r,g,b), 2:max(r,g,b), default = 2

**Table 9-1029 MCDI\_LMVLCKSTEXT\_0 0x2f69**

Bit(s)	R/W	Default	Description
27-16	R/W		reg_mcdi_lmvlckstext_1
11-0	R/W		reg_mcdi_lmvlckstext_0

**Table 9-1030 MCDI\_LMVLCKSTEXT\_1 0x2f6a**

Bit(s)	R/W	Default	Description
31	R/W	1	reg_mcdi_refnewmode
30	R/W	1	reg_mcdi_reflmlck_en



Bit(s)	R/W	Default	Description
29:28	R/W	0	reg_mcdi_lmvlckupdw_ext
27:16	R/W	8	reg_mcdi_lmvlckmin
11:0	R/W	0	reg_mcdi_lmvlckstext_2

Table 9-1031 MCDI\_LMVLCKEEXT\_0 0x2f6b

Bit(s)	R/W	Default	Description
27:16	R/W	0	reg_mcdi_lmvlckedext_1
11:0	R/W	0	reg_mcdi_lmvlckedext_0

Table 9-1032 MCDI\_LMVLCKEEXT\_1 0x2f6c

Bit(s)	R/W	Default	Description
11:0	R/W	0	reg_mcdi_lmvlckedext_2

Table 9-1033 MCDI\_MC\_CRTL 0x2f70

Bit(s)	R/W	Default	Description
31-20	R/W	0	reserved
19	R/W	0	reg buf1 enable (if1)
18	R/W	0	reg buf2 enable (if2)
17	R/W	0	reg mv invert
16	R/W	0	mcvec force 0
15	R/W	0	buf2 always en
14-12	R/W	0	reg_mcdi_mcvec_offset: 0: disable 1: 1 pixel offset of mcvec 2: 2 pixel offset of mcvec 3: 3 pixel offset of mcvec 4: 4 pixel offset of mcvec
11	R/w	0	reg_di_weave_both_side
10	R/W	0	reg_mcdi_mc_uv_en: mc for uv if needed, else use ma of uv
9-8	R/W	1	reg_mcdi_mcpreflg. flag to use previous field for MC, 0-forward field, 1: previous field, 2-use forward & previous. default = 1
7	R/W	1	reg_mcdi_mcrelrefbycolcfden. enable rel refinement by column cofidence in mc blending, default = 1
6-5	R/W	0	reg_mcdi_mclpfen. enable mc pixles/rel lpf, 0:disable, 1: lpf rel, 2: lpf mc pxls, 3: lpf both rel and mc pxls, default = 0
4-2	R/W	0	reg_mcdi_mcdebugmode. enable mc debug mode, 0:disable, 1: split left/right, 2: split top/bottom, 3: debug mv, 4: debug rel, default = 0
1-0	R/W	1	reg_mcdi_mcen. mcdi enable mode, 0:disable, 1: blend with ma, 2: full mc, default = 1

**Table 9-1034 MCDI\_MC\_LPF\_MSK\_0 0x2f71**

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28	R		blend output ready
27	R		mcvec read input ready
26	R		mtn read input ready, same as bit24
25	R		ei dout ready ( if0 )
24	R		mtn read input ready
23	R		if2 input ready
22	R		if1 input ready
21	R		blend input ready
20-16	R/W	0	reg_mcdi_mclpfmsk02. mc lpf coef. 2 for pixel 0 of current block, normalized 16 as '1', default = 0
15-13	R/W		reserved
12-8	R/W	9	reg_mcdi_mclpfmsk01. mc lpf coef. 1 for pixel 0 of current block, normalized 16 as '1', default = 9
7-5	R/W		reserved
4-0	R/W	7	reg_mcdi_mclpfmsk00. mc lpf coef. 0 for pixel 0 of current block, normalized 16 as '1', default = 7

**Table 9-1035 MCDI\_MC\_LPF\_MSK\_1 0x2f72**

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28-21	R		debug info
20-16	R/W	0	reg_mcdi_mclpfmsk12. mc lpf coef. 2 for pixel 1 of current block, 0~16, normalized 16 as '1', default = 0
15-13	R/W		reserved
12-8	R/W	11	reg_mcdi_mclpfmsk11. mc lpf coef. 1 for pixel 1 of current block, 0~16, normalized 16 as '1', default = 11
7-5	R/W		reserved
4-0	R/W	5	reg_mcdi_mclpfmsk10. mc lpf coef. 0 for pixel 1 of current block, 0~16, normalized 16 as '1', default = 5

**Table 9-1036 MCDI\_MC\_LPF\_MSK\_2 0x2f73**

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28-21	R		debug info
20-16	R/W	1	reg_mcdi_mclpfmsk22. mc lpf coef. 2 for pixel 2 of current block, 0~16, normalized 16 as '1', default = 1
15-13	R/W		reserved

Bit(s)	R/W	Default	Description
12-8	R/W	14	reg_mcdi_mclpfmsk21. mc lpf coef. 1 for pixel 2 of current block, 0~16, normalized 16 as '1', default = 14
7-5	R/W		reserved
4-0	R/W	1	reg_mcdi_mclpfmsk20. mc lpf coef. 0 for pixel 2 of current block, 0~16, normalized 16 as '1', default = 1

Table 9-1037 MCDI\_MC\_LPF\_MSK\_3 0x2f74

Bit(s)	R/W	Default	Description
31-29	R/W		reserved
28-21	R		debug info
20-16	R/W	5	reg_mcdi_mclpfmsk32. mc lpf coef. 2 for pixel 3 of current block, 0~16, normalized 16 as '1', default = 5
15-13	R/W		reserved
12-8	R/W	11	reg_mcdi_mclpfmsk31. mc lpf coef. 1 for pixel 3 of current block, 0~16, normalized 16 as '1', default = 11
7-5	R/W		reserved
4-0	R/W	0	reg_mcdi_mclpfmsk30. mc lpf coef. 0 for pixel 3 of current block, 0~16, normalized 16 as '1', default = 0

Table 9-1038 MCDI\_MC\_LPF\_MSK\_4 0x2f75

Bit(s)	R/W	Default	Description
31-21	R/W		reserved
20-16	R/W	7	reg_mcdi_mclpfmsk42. mc lpf coef. 2 for pixel 4 of current block, 0~16, normalized 16 as '1', default = 7
15-13	R/W		reserved
12-8	R/W	9	reg_mcdi_mclpfmsk41. mc lpf coef. 1 for pixel 4 of current block, 0~16, normalized 16 as '1', default = 9
7-5	R/W		reserved
4-0	R/W	0	reg_mcdi_mclpfmsk40. mc lpf coef. 0 for pixel 4 of current block, 0~16, normalized 16 as '1', default = 0

Table 9-1039 MCDI\_MC\_REL\_GAIN\_OFFST\_0 0x2f76

Bit(s)	R/W	Default	Description
31-26	R/W		reserved
25	R/W	0	reg_mcdi_mcmotionparadoxflg. flag of motion paradox, initial with 0 and read from software, default = 0
24	R/W	0	reg_mcdi_mchighvertfrqflg. flag of high vert frq, initial with 0 and read from software, default = 0
23-16	R/W	128	reg_mcdi_mcmotionparadoxoffst. offset (r+ offset) for rel (MC blending coef.) refinement if motion paradox detected before MC blending before MC blending, default = 128

Bit(s)	R/W	Default	Description
15-12	R/W		reserved
11-8	R/W	8	reg_mcdi_mcmotionparadoxgain. gain for rel (MC blending coef.) refinement if motion paradox detected before MC blending, normalized 8 as '1', set 15 to 16, default = 8
7-4	R/W	15	reg_mcdi_mchighvertfrqoffst. minus offset (alpha - offset) for motion (MA blending coef.) refinement if high vertical frequency detected before MA blending, default = 15
3-0	R/W	8	reg_mcdi_mchighvertfrqgain. gain for motion (MA blending coef.) refinement if high vertical frequency detected before MA blending, normalized 8 as '1', set 15 to 16, default = 8

Table 9-1040 MCDI\_MC\_REL\_GAIN\_OFFST\_1 0x2f77

Bit(s)	R/W	Default	Description
31-24	R/W	255	reg_mcdi_mcoutofboundrayoffst. offset (rel + offset) for rel (MC blending coef.) refinement if MC pointed out of boundray before MC blending before MC blending, default = 255
23-20	R/W		reserved
19-16	R/W	8	reg_mcdi_mcoutofboundraygain. gain for rel (MC blending coef.) refinement if MC pointed out of boundray before MC blending, normalized 8 as '1', set 15 to 16, default = 8
15-8	R/W	255	reg_mcdi_mcrelrefbycolcfdoffst. offset (rel + offset) for rel (MC blending coef.) refinement if motion paradox detected before MC blending before MC blending, default = 255
7-4	R/W		reserved.
3-0	R/W	8	reg_mcdi_mcrelrefbycolcfdgain. gain for rel (MC blending coef.) refinement if column cofidence failed before MC blending, normalized 8 as '1', set 15 to 16, default = 8

Table 9-1041 MCDI\_MC\_COL\_CFD\_0 0x2f78

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_0. column cofidence value 0 read from software. initial = 0

Table 9-1042 MCDI\_MC\_COL\_CFD\_1 0x2f79

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_1. column cofidence value 1 read from software. initial = 0

Table 9-1043 MCDI\_MC\_COL\_CFD\_2 0x2f7a

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_2. column cofidence value 2 read from software. initial = 0

**Table 9-1044 MCDI\_MC\_COL\_CFD\_3 0x2f7b**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_3. column cofidence value 3 read from software. initial = 0

**Table 9-1045 MCDI\_MC\_COL\_CFD\_4 0x2f7c**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 4 read from software. initial = 0

**Table 9-1046 MCDI\_MC\_COL\_CFD\_5 0x2f7d**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 5 read from software. initial = 0

**Table 9-1047 MCDI\_MC\_COL\_CFD\_6 0x2f7e**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 6 read from software. initial = 0

**Table 9-1048 MCDI\_MC\_COL\_CFD\_7 0x2f7f**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 7 read from software. initial = 0

**Table 9-1049 MCDI\_MC\_COL\_CFD\_8 0x2f80**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 8 read from software. initial = 0

**Table 9-1050 MCDI\_MC\_COL\_CFD\_9 0x2f81**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 9 read from software. initial = 0

**Table 9-1051 MCDI\_MC\_COL\_CFD\_10 0x2f82**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 10 read from software. initial = 0

**Table 9-1052 MCDI\_MC\_COL\_CFD\_11 0x2f83**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 11 read from software. initial = 0

**Table 9-1053 MCDI\_MC\_COL\_CFD\_12 0x2f84**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 12 read from software. initial = 0

**Table 9-1054 MCDI\_MC\_COL\_CFD\_13 0x2f85**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 13 read from software. initial = 0

**Table 9-1055 MCDI\_MC\_COL\_CFD\_14 0x2f86**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 14 read from software. initial = 0

**Table 9-1056 MCDI\_MC\_COL\_CFD\_15 0x2f87**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 15 read from software. initial = 0

**Table 9-1057 MCDI\_MC\_COL\_CFD\_16 0x2f88**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 16 read from software. initial = 0

**Table 9-1058 MCDI\_MC\_COL\_CFD\_17 0x2f89**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 17 read from software. initial = 0

**Table 9-1059 MCDI\_MC\_COL\_CFD\_18 0x2f8a**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 18 read from software. initial = 0

**Table 9-1060 MCDI\_MC\_COL\_CFD\_19 0x2f8b**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 19 read from software. initial = 0

**Table 9-1061 MCDI\_MC\_COL\_CFD\_20 0x2f8c**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column confidence value 20 read from software. initial = 0

**Table 9-1062 MCDI\_MC\_COL\_CFD\_21 0x2f8d**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column confidence value 21 read from software. initial = 0

**Table 9-1063 MCDI\_MC\_COL\_CFD\_22 0x2f8e**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column confidence value 22 read from software. initial = 0

**Table 9-1064 MCDI\_MC\_COL\_CFD\_23 0x2f8f**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column confidence value 23 read from software. initial = 0

**Table 9-1065 MCDI\_MC\_COL\_CFD\_24 0x2f90**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column confidence value 24 read from software. initial = 0

**Table 9-1066 MCDI\_MC\_COL\_CFD\_25 0x2f91**

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column confidence value 25 read from software. initial = 0

**Table 9-1067 MCDI\_RO\_FLD\_LUMA\_AVG\_SUM 0x2fa0**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldlumaavgsum. block's luma avg sum of current filed (block based). initial = 0

**Table 9-1068 MCDI\_RO\_GMV\_VLD\_CNT 0x2fa1**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_gmvvldcnt. valid gmvs count of pre one filed (block based). initial = 0

**Table 9-1069 MCDI\_RO\_RPT\_FLG\_CNT 0x2fa2**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_rptflgcnt. repeat mv's count of pre one filed (block based). initial = 0

**Table 9-1070 MCDI\_RO\_FLD\_BAD\_SAD\_CNT 0x2fa3**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldbadsadcnt. bad sad count of whole pre one field (block based). initial = 0

**Table 9-1071 MCDI\_RO\_FLD\_BAD\_BADW\_CNT 0x2fa4**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldbdbadbadwcnt. bad badw count of whole pre one field (block based). initial = 0

**Table 9-1072 MCDI\_RO\_FLD\_BAD\_REL\_CNT 0x2fa5**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldbdbadrelcnt. bad rel count of whole pre one field (block based). initial = 0

**Table 9-1073 MCDI\_RO\_FLD\_MTN\_CNT 0x2fa6**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldmtncnt. motion count of whole pre one field (pixel based). initial = 0

**Table 9-1074 MCDI\_RO\_FLD\_VLD\_CNT 0x2fa7**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldvldcnt. valid motion count of whole pre one field (pixel based). initial = 0

**Table 9-1075 MCDI\_RO\_FLD\_PD\_22\_PRE\_CNT 0x2fa8**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldpd22precnt. previous pd22 check count of whole pre one field (block based). initial = 0

**Table 9-1076 MCDI\_RO\_FLD\_PD\_22\_FOR\_CNT 0x2fa9**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldpd22forcnt. forward pd22 check count of whole pre one field (block based). initial = 0

**Table 9-1077 MCDI\_RO\_FLD\_PD\_22\_FLT\_CNT 0x2faa**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldpd22fltcnt. flat count (for pd22 check) of whole pre one field (block based). initial = 0



**Table 9-1078 MCDI\_RO\_HIGH\_VERT\_FRQ\_FLG 0x2fab**

Bit(s)	R/W	Default	Description
31-16	R		reserved.
15-8	R		ro_mcdi_highvertfrqcnt. high vertical frequency count till prevoius one field. initial = 0
7-3	R		reserved.
2-1	R		ro_mcdi_highvertfrqphase. high vertical frequency phase of prevoius one field. initial = 2
0	R		ro_mcdi_highvertfrqflg. high vertical frequency flag of prevoius one field. initial = 0

**Table 9-1079 MCDI\_RO\_GMV\_LOCK\_FLG 0x2fac**

Bit(s)	R/W	Default	Description
31-16	R		reserved.
15-8	R		ro_mcdi_gmvlockcnt. global mv lock count till prevoius one field. initial = 0
7-2	R		ro_mcdi_gmv. global mv of prevoius one field. -31~31, initial = 32 (invalid value)
1	R		ro_mcdi_zerogmvlockflg. zero global mv lock flag of prevoius one field. initial = 0
0	R		ro_mcdi_gmvlockflg. global mv lock flag of prevoius one field. initial = 0

**Table 9-1080 MCDI\_RO\_RPT\_MV 0x2fad**

Bit(s)	R/W	Default	Description
5-0	R		ro_mcdi_rptmv. repeate mv of prevoius one field. -31~31, initial = 32 (invalid value)

**Table 9-1081 MCDI\_RO\_MOTION\_PARADOX\_FLG 0x2fae**

Bit(s)	R/W	Default	Description
31-16	R		reserved.
15-8	R		ro_mcdi_motionparadoxcnt. motion paradox count till prevoius one field. initial = 0
7-1	R		reserved.
0	R		ro_mcdi_motionparadoxflg. motion paradox flag of prevoius one field. initial = 0

**Table 9-1082 MCDI\_RO\_PD\_22\_FLG 0x2faf**

Bit(s)	R/W	Default	Description
31-16	R		reserved.
26	R	0	ro_mcdi_pd22flg2. pull down 22 flag of prevoius one field. initial = 0
25	R	0	ro_mcdi_pd22flg1. pull down 22 flag of prevoius one field. initial = 0
24	R	0	ro_mcdi_pd22flg0. pull down 22 flag of prevoius one field. initial = 0

Bit(s)	R/W	Default	Description
23-16	R	0	ro_mcdi_pd22cnt2. pull down 22 count till prevoius one field. initial = 0
15-8	R	0	ro_mcdi_pd22cnt1. pull down 22 count till prevoius one field. initial = 0
7-0	R	0	ro_mcdi_pd22cnt0. pull down 22 count till prevoius one field. initial = 0

**Table 9-1083 MCDI\_RO\_COL\_CFD\_0 0x2fb0**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_0. column cofidence value 0. initial = 0

**Table 9-1084 MCDI\_RO\_COL\_CFD\_1 0x2fb1**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_1. column cofidence value 1. initial = 0

**Table 9-1085 MCDI\_RO\_COL\_CFD\_2 0x2fb2**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_2. column cofidence value 2. initial = 0

**Table 9-1086 MCDI\_RO\_COL\_CFD\_3 0x2fb3**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_3. column cofidence value 3. initial = 0

**Table 9-1087 MCDI\_RO\_COL\_CFD\_4 0x2fb4**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_4. column cofidence value 4. initial = 0

**Table 9-1088 MCDI\_RO\_COL\_CFD\_5 0x2fb5**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_5. column cofidence value 5. initial = 0

**Table 9-1089 MCDI\_RO\_COL\_CFD\_6 0x2fb6**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_6. column cofidence value 6. initial = 0

**Table 9-1090 MCDI\_RO\_COL\_CFD\_7 0x2fb7**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_7. column cofidence value 7. initial = 0

**Table 9-1091 MCDI\_RO\_COL\_CFD\_8 0x2fb8**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_8. column cofidence value 8. initial = 0

**Table 9-1092 MCDI\_RO\_COL\_CFD\_9 0x2fb9**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_9. column cofidence value 9. initial = 0

**Table 9-1093 MCDI\_RO\_COL\_CFD\_10 0x2fba**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_10. column cofidence value 10. initial = 0

**Table 9-1094 MCDI\_RO\_COL\_CFD\_11 0x2fbb**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_11. column cofidence value 11. initial = 0

**Table 9-1095 MCDI\_RO\_COL\_CFD\_12 0x2fbc**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_12. column cofidence value 12. initial = 0

**Table 9-1096 MCDI\_RO\_COL\_CFD\_13 0x2fbd**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_13. column cofidence value 13. initial = 0

**Table 9-1097 MCDI\_RO\_COL\_CFD\_14 0x2fbe**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_14. column cofidence value 14. initial = 0

**Table 9-1098 MCDI\_RO\_COL\_CFD\_15 0x2fbf**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_15. column cofidence value 15. initial = 0

**Table 9-1099 MCDI\_RO\_COL\_CFD\_16 0x2fc0**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_16. column cofidence value 16. initial = 0

**Table 9-1100 MCDI\_RO\_COL\_CFD\_17 0x2fc1**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_17. column cofidence value 17. initial = 0

**Table 9-1101 MCDI\_RO\_COL\_CFD\_18 0x2fc2**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_18. column cofidence value 18. initial = 0

**Table 9-1102 MCDI\_RO\_COL\_CFD\_19 0x2fc3**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_19. column cofidence value 19. initial = 0

**Table 9-1103 MCDI\_RO\_COL\_CFD\_20 0x2fc4**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_20. column cofidence value 20. initial = 0

**Table 9-1104 MCDI\_RO\_COL\_CFD\_21 0x2fc5**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_21. column cofidence value 21. initial = 0

**Table 9-1105 MCDI\_RO\_COL\_CFD\_22 0x2fc6**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_22. column cofidence value 22. initial = 0

**Table 9-1106 MCDI\_RO\_COL\_CFD\_23 0x2fc7**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_23. column cofidence value 23. initial = 0

**Table 9-1107 MCDI\_RO\_COL\_CFD\_24 0x2fc8**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_24. column confidence value 24. initial = 0

**Table 9-1108 MCDI\_RO\_COL\_CFD\_25 0x2fc9**

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_25. column confidence value 25. initial = 0

**Table 9-1109 MCDI\_RO\_FLD\_PD\_22\_PRE\_CNT1 0x2fca**

Bit(s)	R/W	Default	Description
31-0	R		previous pd22 check count of whole pre one field(block based). initial = 0

**Table 9-1110 MCDI\_RO\_FLD\_PD\_22\_POR\_CNT1 0x2fcb**

Bit(s)	R/W	Default	Description
31-0	R		forward pd22 check count of whole pre one field(block based). initial = 0

**Table 9-1111 MCDI\_RO\_FLD\_PD\_22\_FLT\_CNT1 0x2fcc**

Bit(s)	R/W	Default	Description
31-0	R		flat count(for pd22 check) of whole pre one field(block based). initial = 0

**Table 9-1112 MCDI\_RO\_FLD\_PD\_22\_PRE\_CNT2 0x2fcd**

Bit(s)	R/W	Default	Description
31-0	R		previous pd22 check count of whole pre one field(block based). initial = 0

**Table 9-1113 MCDI\_RO\_FLD\_PD\_22\_POR\_CNT2 0x2fce**

Bit(s)	R/W	Default	Description
31-0	R		forward pd22 check count of whole pre one field(block based). initial = 0

**Table 9-1114 MCDI\_RO\_FLD\_PD\_22\_FLT\_CNT2 0x2fcf**

Bit(s)	R/W	Default	Description
31-0	R		flat count(for pd22 check) of whole pre one field(block based). initial = 0

## 9.2.3.14 PULLDOWN Registers

### Register Address

- DIPD\_COMB\_CTRL0 0xff00bf40
- DIPD\_COMB\_CTRL1 0xff00bf44
- DIPD\_COMB\_CTRL2 0xff00bf48
- DIPD\_COMB\_CTRL3 0xff00bf4c
- DIPD\_COMB\_CTRL4 0xff00bf50
- DIPD\_COMB\_CTRL5 0xff00bf54
- DIPD\_RO\_COMB\_0 0xff00bf58
- DIPD\_RO\_COMB\_1 0xff00bf5c
- DIPD\_RO\_COMB\_2 0xff00bf60
- DIPD\_RO\_COMB\_3 0xff00bf64
- DIPD\_RO\_COMB\_4 0xff00bf68
- DIPD\_RO\_COMB\_5 0xff00bf6c
- DIPD\_RO\_COMB\_6 0xff00bf70
- DIPD\_RO\_COMB\_7 0xff00bf74
- DIPD\_RO\_COMB\_8 0xff00bf78
- DIPD\_RO\_COMB\_9 0xff00bf7c
- DIPD\_RO\_COMB\_10 0xff00bf80
- DIPD\_RO\_COMB\_11 0xff00bf84
- DIPD\_RO\_COMB\_12 0xff00bf88
- DIPD\_RO\_COMB\_13 0xff00bf8c
- DIPD\_RO\_COMB\_14 0xff00bf90
- DIPD\_RO\_COMB\_15 0xff00bf94
- DIPD\_RO\_COMB\_16 0xff00bf98
- DIPD\_RO\_COMB\_17 0xff00bf9c
- DIPD\_RO\_COMB\_18 0xff00bfa0
- DIPD\_RO\_COMB\_19 0xff00bfa4
- DIPD\_RO\_COMB\_20 0xff00bfa8
- DIPD\_COMB\_CTRL6 0xff00bfac
- DI\_PD\_GRAD\_CTRL 0xff005f80
- DI\_PD\_GRAD\_TH\_P 0xff005f84
- DI\_PD\_GRAD\_TH\_N 0xff005f88
- DI\_PD\_GRAD\_GAIN\_P 0xff005f8c
- DI\_PD\_GRAD\_GAIN\_N 0xff005f90
- DI\_PD\_RO\_SUM\_P\_WIN0 0xff005f94
- DI\_PD\_RO\_SUM\_P\_WIN1 0xff005f98
- DI\_PD\_RO\_SUM\_P\_WIN2 0xff005f9c
- DI\_PD\_RO\_SUM\_P\_WIN3 0xff005fa0
- DI\_PD\_RO\_SUM\_P\_WIN4 0xff005fa4
- DI\_PD\_RO\_SUM\_N\_WIN0 0xff005fa8
- DI\_PD\_RO\_SUM\_N\_WIN1 0xff005fac
- DI\_PD\_RO\_SUM\_N\_WIN2 0xff005fb0

- DI\_PD\_RO\_SUM\_N\_WIN3 0xff005fb4
- DI\_PD\_RO\_SUM\_N\_WIN4 0xff005fb8
- DI\_PD\_RO\_CNT\_P\_WIN0 0xff005fbc
- DI\_PD\_RO\_CNT\_P\_WIN1 0xff005fc0
- DI\_PD\_RO\_CNT\_P\_WIN2 0xff005fc4
- DI\_PD\_RO\_CNT\_P\_WIN3 0xff005fc8
- DI\_PD\_RO\_CNT\_P\_WIN4 0xff005fcc
- DI\_PD\_RO\_CNT\_N\_WIN0 0xff005fd0
- DI\_PD\_RO\_CNT\_N\_WIN1 0xff005fd4
- DI\_PD\_RO\_CNT\_N\_WIN2 0xff005fd8
- DI\_PD\_RO\_CNT\_N\_WIN3 0xff005fdc
- DI\_PD\_RO\_CNT\_N\_WIN4 0xff005fe0
- DI\_PD\_RO\_SUM\_P 0xff005fe4
- DI\_PD\_RO\_SUM\_N 0xff005fe8
- DI\_PD\_RO\_CNT\_P 0xff005fec
- DI\_PD\_RO\_CNT\_N 0xff005ff0

## Register Description

**Table 9-1115 DIPD\_COMB\_CTRL0 0x2fd0**

Bit(s)	R/W	Default	Description
31-24	W		Cmb_v_dif_min
23-16	W		Cmb_v_dif_max
15-8	W		Cmb_crg_min
7-0	W		Cmb_crg_max

**Table 9-1116 DIPD\_COMB\_CTRL1 0x2fd1**

Bit(s)	R/W	Default	Description
31	W		Pd_check_en
29-24	W		Cmb_wv_min3
21-16	W		Cmb_wv_min2
13-8	W		Cmb_wv_min1
5-0	W		Cmb_wv_min0

**Table 9-1117 DIPD\_COMB\_CTRL2 0x2fd2**

Bit(s)	R/W	Default	Description
31-28	W		Cmb_wnd_cnt1
25-20	W		Ccnt_cmmin1
19-16	W		Ccnt_mtmin

Bit(s)	R/W	Default	Description
13-8	W		Ccnt_cmmin
5-0	W		Cmb_wv_min4

Table 9-1118 DIPD\_COMB\_CTRL3 0x2fd3

Bit(s)	R/W	Default	Description
31	W		Cmb32spcl
17-12	W		Cmb_wnd_mthd
11-4	W		Cmb_abs_nocmb
3-0	W		Cnt_minlen

Table 9-1119 DIPD\_COMB\_CTRL4 0x2fd4

Bit(s)	R/W	Default	Description
30	W		FIm_stamtn_en
29-28	W		In_horflt
27-20	W		Alpha
19-16	W		Rhtran_ctmtd
15-8	W		Htran_mnth1
7-0	W		Htran_mnth0

Table 9-1120 DIPD\_COMB\_CTRL5 0x2fd5

Bit(s)	R/W	Default	Description
31-24	W		Fld_mindif
23-16	W		Frm_mindif
13-8	W		FIm_smp_mtn_cnt
7-0	W		FIm_smp_mtn_thd

Table 9-1121 DIPD\_RO\_COMB\_0 0x2fd6

Bit(s)	R/W	Default	Description
31-0	R		frmdif

Table 9-1122 DIPD\_RO\_COMB\_1 0x2fd7

Bit(s)	R/W	Default	Description
31-0	R		Frmdif0



**Table 9-1123 DIPD\_RO\_COMB\_2 0x2fd8**

Bit(s)	R/W	Default	Description
31-0	R		Frmdif1

**Table 9-1124 DIPD\_RO\_COMB\_3 0x2fd9**

Bit(s)	R/W	Default	Description
31-0	R		Frmdif2

**Table 9-1125 DIPD\_RO\_COMB\_4 0x2fda**

Bit(s)	R/W	Default	Description
31-0	R		Frmdif3

**Table 9-1126 DIPD\_RO\_COMB\_5 0x2fdb**

Bit(s)	R/W	Default	Description
31-0	R		Frmdif4

**Table 9-1127 DIPD\_RO\_COMB\_6 0x2fdc**

Bit(s)	R/W	Default	Description
31-0	R		flddif

**Table 9-1128 DIPD\_RO\_COMB\_7 0x2fdd**

Bit(s)	R/W	Default	Description
31-0	R		Flddif0

**Table 9-1129 DIPD\_RO\_COMB\_8 0x2fde**

Bit(s)	R/W	Default	Description
31-0	R		Flddif1

**Table 9-1130 DIPD\_RO\_COMB\_9 0x2fdf**

Bit(s)	R/W	Default	Description
31-0	R		Flddif2

**Table 9-1131 DIPD\_RO\_COMB\_10 0x2fe0**

Bit(s)	R/W	Default	Description
31-0	R		Flddif3

**Table 9-1132 DIPD\_RO\_COMB\_11 0x2fe1**

Bit(s)	R/W	Default	Description
31-0	R		Flddif4

**Table 9-1133 DIPD\_RO\_COMB\_12 0x2fe2**

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt0

**Table 9-1134 DIPD\_RO\_COMB\_13 0x2fe3**

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt1

**Table 9-1135 DIPD\_RO\_COMB\_14 0x2fe4**

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt2

**Table 9-1136 DIPD\_RO\_COMB\_15 0x2fe5**

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt3

**Table 9-1137 DIPD\_RO\_COMB\_16 0x2fe6**

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt4

**Table 9-1138 DIPD\_RO\_COMB\_17 0x2fe7**

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt5

**Table 9-1139 DIPD\_RO\_COMB\_18 0x2fe8**

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt6

**Table 9-1140 DIPD\_RO\_COMB\_19 0x2fe9**

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt7

**Table 9-1141 DIPD\_RO\_COMB\_20 0x2fea**

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt8

**Table 9-1142 DIPD\_COMB\_CTRL5 0x2fd5**

Bit(s)	R/W	Default	Description
31-24	W		Fld_mindif
23-16	W		Frm_mindif
13-8	W		Flm_smp_mtn_cnt
7-0	W		Flm_smp_mtn_thd

**Table 9-1143 DIPD\_COMB\_CTRL6 0x2feb**

Bit(s)	R/W	Default	Description
21	Rw	0	Reg_edit_sel
20	Rw	1	Reg_horflt_en
16-12	Rw	7	Reg_combseglen
9-4	Rw	6	Reg_trancombrat
3-0	Rw	4	Reg_combsegmin

**Table 9-1144 DI\_PD\_GRAD\_CTRL 0x17e0**

Bit(s)	R/W	Default	Description
11:4	R/W	255	Reg_fd_min_grad_th
3	R/W	1	Reg_fd_min_grad_th_adjust_en
2	R/W	1	Reg_fd_err_grad_bf_ft_en
1	R/W	0	Reg_fd_comb_field_sel
0	R/W	0	Reg_fd_iscur_top_inver

**Table 9-1145 DI\_PD\_GRAD\_TH\_P 0x17e1**

Bit(s)	R/W	Default	Description
31-24	R/W	2	Reg_fd_min_grad_th_p0
23-16	R/W	4	Reg_fd_min_grad_th_p1
15-8	R/W	8	Reg_fd_min_grad_th_p2
7-0	R/W	12	Reg_fd_min_grad_th_p3

**Table 9-1146 DI\_PD\_GRAD\_TH\_N 0x17e2**

Bit(s)	R/W	Default	Description
31-24	R/W	4	Reg_fd_min_grad_th_n0
23-16	R/W	8	Reg_fd_min_grad_th_n1
15-8	R/W	16	Reg_fd_min_grad_th_n2
7-0	R/W	32	Reg_fd_min_grad_th_n3

**Table 9-1147 DI\_PD\_GRAD\_GAIN\_P 0x17e3**

Bit(s)	R/W	Default	Description
28-24	R/W	16	Reg_fd_min_grad_gain_p0
20-16	R/W	16	Reg_fd_min_grad_gain_p1
12-8	R/W	16	Reg_fd_min_grad_gain_p2
4-0	R/W	16	Reg_fd_min_grad_gain_p3

**Table 9-1148 DI\_PD\_GRAD\_GAIN\_N 0x17e4**

Bit(s)	R/W	Default	Description
28-24	R/W	12	Reg_fd_min_grad_gain_n0
20-16	R/W	12	Reg_fd_min_grad_gain_n1
12-8	R/W	14	Reg_fd_min_grad_gain_n2
4-0	R/W	15	Reg_fd_min_grad_gain_n3

**Table 9-1149 DI\_PD\_RO\_SUM\_P\_WIN0 0x17e5**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_p0

**Table 9-1150 DI\_PD\_RO\_SUM\_P\_WIN1 0x17e6**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_p1

**Table 9-1151 DI\_PD\_RO\_SUM\_P\_WIN2 0x17e7**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_p2

**Table 9-1152 DI\_PD\_RO\_SUM\_P\_WIN3 0x17e8**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_p3

**Table 9-1153 DI\_PD\_RO\_SUM\_P\_WIN4 0x17e9**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_p4

**Table 9-1154 DI\_PD\_RO\_SUM\_N\_WIN0 0x17ea**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_n0

**Table 9-1155 DI\_PD\_RO\_SUM\_N\_WIN1 0x17eb**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_n1

**Table 9-1156 DI\_PD\_RO\_SUM\_N\_WIN2 0x17ec**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_n2

**Table 9-1157 DI\_PD\_RO\_SUM\_N\_WIN3 0x17ed**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_n3

**Table 9-1158 DI\_PD\_RO\_SUM\_N\_WIN4 0x17ee**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_n4

**Table 9-1159 DI\_PD\_RO\_CNT\_P\_WIN0 0x17ef**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_p0

**Table 9-1160 DI\_PD\_RO\_CNT\_P\_WIN1 0x17f0**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_p1

**Table 9-1161 DI\_PD\_RO\_CNT\_P\_WIN2 0x17f1**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_p2

**Table 9-1162 DI\_PD\_RO\_CNT\_P\_WIN3 0x17f2**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_p3

**Table 9-1163 DI\_PD\_RO\_CNT\_P\_WIN4 0x17f3**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_p4

**Table 9-1164 DI\_PD\_RO\_CNT\_N\_WIN0 0x17f4**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_n0

**Table 9-1165 DI\_PD\_RO\_CNT\_N\_WIN1 0x17f5**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_n1

**Table 9-1166 DI\_PD\_RO\_CNT\_N\_WIN2 0x17f6**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_n2

**Table 9-1167 DI\_PD\_RO\_CNT\_N\_WIN3 0x17f7**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_n3

**Table 9-1168 DI\_PD\_RO\_CNT\_N\_WIN4 0x17f8**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_n4

**Table 9-1169 DI\_PD\_RO\_SUM\_P 0x17f9**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_p_glb

**Table 9-1170 DI\_PD\_RO\_SUM\_N 0x17fa**

Bit(s)	R/W	Default	Description
31-0	R		Ro_fd_err_grad_sum_n_glb

**Table 9-1171 DI\_PD\_RO\_CNT\_P 0x17fb**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_p_glb

**Table 9-1172 DI\_PD\_RO\_CNT\_N 0x17fc**

Bit(s)	R/W	Default	Description
20-0	R		Ro_fd_err_grad_cnt_n_glb

### 9.2.3.15 DNR Registers

#### Register Address

- DNR\_CTRL 0xff00b400
- DNR\_HVSIZE 0xff00b404
- DNR\_DBLK\_BLANK\_NUM 0xff00b408
- DNR\_BLK\_OFFST 0xff00b40c
- DNR\_GBS 0xff00b410
- DNR\_HBOFFST\_STAT 0xff00b414
- DNR\_VBOFFST\_STAT 0xff00b418
- DNR\_GBS\_STAT 0xff00b41c
- DNR\_STAT\_X\_START\_END 0xff00b420
- DNR\_STAT\_Y\_START\_END 0xff00b424
- DNR\_LUMA 0xff00b428
- DNR\_DB\_YEDGE\_THD 0xff00b42c
- DNR\_DB\_CEDGE\_THD 0xff00b430
- DNR\_DB\_HGAP 0xff00b434
- DNR\_DB\_HBS 0xff00b438
- DNR\_DB\_HACT 0xff00b43c
- DNR\_DB\_YHDELTA\_GAIN 0xff00b440
- DNR\_DB\_YHDELTA2\_GAIN 0xff00b444
- DNR\_DB\_CHDELTA\_GAIN 0xff00b448
- DNR\_DB\_CHDELTA2\_GAIN 0xff00b44c

- DNR\_DB\_YC\_VEDGE\_THD 0xff00b450
- DNR\_DB\_VBS\_MISC 0xff00b454
- DNR\_DB\_YVDELTA\_GAIN 0xff00b458
- DNR\_DB\_CVDELTA\_GAIN 0xff00b45c
- DNR\_RO\_GBS\_STAT\_LR 0xff00b460
- DNR\_RO\_GBS\_STAT\_LL 0xff00b464
- DNR\_RO\_GBS\_STAT\_RR 0xff00b468
- DNR\_RO\_GBS\_STAT\_DIF 0xff00b46c
- DNR\_RO\_GBS\_STAT\_CNT 0xff00b470
- DNR\_RO\_HBOF\_STAT\_CNT\_0 0xff00b474
- DNR\_RO\_HBOF\_STAT\_CNT\_1 0xff00b478
- DNR\_RO\_HBOF\_STAT\_CNT\_2 0xff00b47c
- DNR\_RO\_HBOF\_STAT\_CNT\_3 0xff00b480
- DNR\_RO\_HBOF\_STAT\_CNT\_4 0xff00b484
- DNR\_RO\_HBOF\_STAT\_CNT\_5 0xff00b488
- DNR\_RO\_HBOF\_STAT\_CNT\_6 0xff00b48c
- DNR\_RO\_HBOF\_STAT\_CNT\_7 0xff00b490
- DNR\_RO\_HBOF\_STAT\_CNT\_8 0xff00b494
- DNR\_RO\_HBOF\_STAT\_CNT\_9 0xff00b498
- DNR\_RO\_HBOF\_STAT\_CNT\_10 0xff00b49c
- DNR\_RO\_HBOF\_STAT\_CNT\_11 0xff00b4a0
- DNR\_RO\_HBOF\_STAT\_CNT\_12 0xff00b4a4
- DNR\_RO\_HBOF\_STAT\_CNT\_13 0xff00b4a8
- DNR\_RO\_HBOF\_STAT\_CNT\_14 0xff00b4ac
- DNR\_RO\_HBOF\_STAT\_CNT\_15 0xff00b4b0
- DNR\_RO\_HBOF\_STAT\_CNT\_16 0xff00b4b4
- DNR\_RO\_HBOF\_STAT\_CNT\_17 0xff00b4b8
- DNR\_RO\_HBOF\_STAT\_CNT\_18 0xff00b4bc
- DNR\_RO\_HBOF\_STAT\_CNT\_19 0xff00b4c0
- DNR\_RO\_HBOF\_STAT\_CNT\_20 0xff00b4c4
- DNR\_RO\_HBOF\_STAT\_CNT\_21 0xff00b4c8
- DNR\_RO\_HBOF\_STAT\_CNT\_22 0xff00b4cc
- DNR\_RO\_HBOF\_STAT\_CNT\_23 0xff00b4d0
- DNR\_RO\_HBOF\_STAT\_CNT\_24 0xff00b4d4
- DNR\_RO\_HBOF\_STAT\_CNT\_25 0xff00b4d8
- DNR\_RO\_HBOF\_STAT\_CNT\_26 0xff00b4dc
- DNR\_RO\_HBOF\_STAT\_CNT\_27 0xff00b4e0
- DNR\_RO\_HBOF\_STAT\_CNT\_28 0xff00b4e4
- DNR\_RO\_HBOF\_STAT\_CNT\_29 0xff00b4e8
- DNR\_RO\_HBOF\_STAT\_CNT\_30 0xff00b4ec
- DNR\_RO\_HBOF\_STAT\_CNT\_31 0xff00b4f0
- DNR\_RO\_VBOF\_STAT\_CNT\_0 0xff00b4f4
- DNR\_RO\_VBOF\_STAT\_CNT\_1 0xff00b4f8
- DNR\_RO\_VBOF\_STAT\_CNT\_2 0xff00b4fc



- DNR\_RO\_VBOF\_STAT\_CNT\_3 0xff00b500
- DNR\_RO\_VBOF\_STAT\_CNT\_4 0xff00b504
- DNR\_RO\_VBOF\_STAT\_CNT\_5 0xff00b508
- DNR\_RO\_VBOF\_STAT\_CNT\_6 0xff00b50c
- DNR\_RO\_VBOF\_STAT\_CNT\_7 0xff00b510
- DNR\_RO\_VBOF\_STAT\_CNT\_8 0xff00b514
- DNR\_RO\_VBOF\_STAT\_CNT\_9 0xff00b518
- DNR\_RO\_VBOF\_STAT\_CNT\_10 0xff00b51c
- DNR\_RO\_VBOF\_STAT\_CNT\_11 0xff00b520
- DNR\_RO\_VBOF\_STAT\_CNT\_12 0xff00b524
- DNR\_RO\_VBOF\_STAT\_CNT\_13 0xff00b528
- DNR\_RO\_VBOF\_STAT\_CNT\_14 0xff00b52c
- DNR\_RO\_VBOF\_STAT\_CNT\_15 0xff00b530
- DNR\_RO\_VBOF\_STAT\_CNT\_16 0xff00b534
- DNR\_RO\_VBOF\_STAT\_CNT\_17 0xff00b538
- DNR\_RO\_VBOF\_STAT\_CNT\_18 0xff00b53c
- DNR\_RO\_VBOF\_STAT\_CNT\_19 0xff00b540
- DNR\_RO\_VBOF\_STAT\_CNT\_20 0xff00b544
- DNR\_RO\_VBOF\_STAT\_CNT\_21 0xff00b548
- DNR\_RO\_VBOF\_STAT\_CNT\_22 0xff00b54c
- DNR\_RO\_VBOF\_STAT\_CNT\_23 0xff00b550
- DNR\_RO\_VBOF\_STAT\_CNT\_24 0xff00b554
- DNR\_RO\_VBOF\_STAT\_CNT\_25 0xff00b558
- DNR\_RO\_VBOF\_STAT\_CNT\_26 0xff00b55c
- DNR\_RO\_VBOF\_STAT\_CNT\_27 0xff00b560
- DNR\_RO\_VBOF\_STAT\_CNT\_28 0xff00b564
- DNR\_RO\_VBOF\_STAT\_CNT\_29 0xff00b568
- DNR\_RO\_VBOF\_STAT\_CNT\_30 0xff00b56c
- DNR\_RO\_VBOF\_STAT\_CNT\_31 0xff00b570
- DNR\_DM\_ADP\_EN 0xff00b574
- DNR\_DM\_EDGE\_DIR 0xff00b578
- DNR\_DM\_CTRL 0xff00b580
- DNR\_DM\_NR\_BLND 0xff00b584
- DNR\_DM\_RNG\_THD 0xff00b588
- DNR\_DM\_RNG\_GAIN\_OFST 0xff00b58c
- DNR\_DM\_DIR\_MISC 0xff00b590
- DNR\_DM\_COR\_DIF 0xff00b594
- DNR\_DM\_FLT\_THD 0xff00b598
- DNR\_DM\_VAR\_THD 0xff00b59c
- DNR\_DM\_EDGE\_DIF\_THD 0xff00b5a0
- DNR\_DM\_AVG\_THD 0xff00b5a4
- DNR\_DM\_AVG\_VAR\_DIF\_THD 0xff00b5a8
- DNR\_DM\_VAR\_EDGE\_DIF\_THD2 0xff00b5ac
- DNR\_DM\_DIF\_FLT\_MISC 0xff00b5b0

- DNR\_DM\_SDIF\_LUT0\_2 0xff00b5b4
- DNR\_DM\_SDIF\_LUT3\_5 0xff00b5b8
- DNR\_DM\_SDIF\_LUT6\_8 0xff00b5bc
- DNR\_DM\_LDIF\_LUT0\_2 0xff00b5c0
- DNR\_DM\_LDIF\_LUT3\_5 0xff00b5c4
- DNR\_DM\_LDIF\_LUT6\_8 0xff00b5c8
- DNR\_DM\_DIF2NORM\_LUT0\_2 0xff00b5cc
- DNR\_DM\_DIF2NORM\_LUT3\_5 0xff00b5d0
- DNR\_DM\_DIF2NORM\_LUT6\_8 0xff00b5d4
- DNR\_DM\_GMS\_THD 0xff00b5d8
- DNR\_RO\_DM\_GMS\_STAT\_CNT 0xff00b5dc
- DNR\_RO\_DM\_GMS\_STAT\_MS 0xff00b5e0
- DNR\_DM\_EDGE\_GAIN 0xff00b5e4
- DNR\_DM\_FLG\_BDIF 0xff00b5e8
- DNR\_DM\_GBS\_RORM 0xff00b5ec
- DNR\_DM\_FLG\_LEV 0xff00b5f0
- DNR\_DM\_DIF\_FLG\_TH 0xff00b5f4
- DNR\_DM\_CALP\_GAIN\_OFST 0xff00b5f8
- DECOMB\_DET\_VERT\_CON0 0xff00b600
- DECOMB\_DET\_VERT\_CON1 0xff00b604
- DECOMB\_DET\_EDGE\_CON0 0xff00b608
- DECOMB\_DET\_EDGE\_CON1 0xff00b60c
- DECOMB\_PARA 0xff00b610
- DECOMB\_BLND\_CON0 0xff00b614
- DECOMB\_BLND\_CON1 0xff00b618
- DECOMB\_YC\_THRD 0xff00b61c
- DECOMB\_MTN\_GAIN\_OFST 0xff00b620
- DECOMB\_CMB\_SEL\_GAIN\_OFST 0xff00b624
- DECOMB\_WIND00 0xff00b628
- DECOMB\_WIND01 0xff00b62c
- DECOMB\_WIND10 0xff00b630
- DECOMB\_WIND11 0xff00b634
- DECOMB\_MODE 0xff00b638
- DECOMB\_FRM\_SIZE 0xff00b63c
- DECOMB\_HV\_BLANK 0xff00b640
- NR2\_POLAR3\_MODE 0xff00b660
- NR2\_POLAR3\_THRD 0xff00b664
- NR2\_POLAR3\_PARA0 0xff00b668
- NR2\_POLAR3\_PARA1 0xff00b66c
- NR2\_POLAR3\_CTRL 0xff00b670
- NR2\_RO\_POLAR3\_NUMOFPIX 0xff00b674
- NR2\_RO\_POLAR3\_SMOOTHMV 0xff00b678
- NR2\_RO\_POLAR3\_M1 0xff00b67c
- NR2\_RO\_POLAR3\_P1 0xff00b680

- NR2\_RO\_POLAR3\_M2 0xff00b684
- NR2\_RO\_POLAR3\_P2 0xff00b688
- NR2\_RO\_POLAR3\_32 0xff00b68c
- NR4\_DRT\_CTRL 0xff00b690
- NR4\_DRT\_YSAD\_GAIN 0xff00b694
- NR4\_DRT\_CSAD\_GAIN 0xff00b698
- NR4\_DRT\_SAD\_ALP\_CORE 0xff00b69c
- NR4\_DRT\_ALP\_MINMAX 0xff00b6a0
- NR4\_SNR\_CTRL\_REG 0xff00b6a4
- NR4\_SNR\_ALPHA0\_MAX\_MIN 0xff00b6a8
- NR4\_ALP0C\_ERR2CURV\_LIMIT0 0xff00b6ac
- NR4\_ALP0C\_ERR2CURV\_LIMIT1 0xff00b6b0
- NR4\_ALP0Y\_ERR2CURV\_LIMIT0 0xff00b6b4
- NR4\_ALP0Y\_ERR2CURV\_LIMIT1 0xff00b6b8
- NR4\_SNR\_ALPA1\_RATE\_AND\_OFST 0xff00b6bc
- NR4\_SNR\_ALPHA1\_MAX\_MIN 0xff00b6c0
- NR4\_ALP1C\_ERR2CURV\_LIMIT0 0xff00b6c4
- NR4\_ALP1C\_ERR2CURV\_LIMIT1 0xff00b6c8
- NR4\_ALP1Y\_ERR2CURV\_LIMIT0 0xff00b6cc
- NR4\_ALP1Y\_ERR2CURV\_LIMIT1 0xff00b6d0
- NR4\_MTN\_CTRL 0xff00b6d4
- NR4\_MTN\_REF\_PAR0 0xff00b6d8
- NR4\_MTN\_REF\_PAR1 0xff00b6dc
- NR4\_MCNR\_LUMA\_ENH\_CTRL 0xff00b6e0
- NR4\_MCNR\_LUMA\_STAT\_LIMTX 0xff00b6e4
- NR4\_MCNR\_LUMA\_STAT\_LIMTY 0xff00b6e8
- NR4\_MCNR\_LUMA\_DIF\_CALC 0xff00b6ec
- NR4\_MCNR\_LUMAPRE\_CAL\_PRAM 0xff00b6f0
- NR4\_MCNR\_LUMACUR\_CAL\_PRAM 0xff00b6f4
- NR4\_MCNR\_MV\_CTRL\_REG 0xff00b6f8
- NR4\_MCNR\_MV\_GAIN0 0xff00b6fc
- NR4\_MCNR\_LMV\_PARM 0xff00b700
- NR4\_MCNR\_ALP0\_REG 0xff00b704
- NR4\_MCNR\_ALP1\_AND\_BET0\_REG 0xff00b708
- NR4\_MCNR\_BET1\_AND\_BET2\_REG 0xff00b70c
- NR4\_MCNR\_AC\_DC\_CRTL 0xff00b710
- NR4\_MCNR\_CM\_CTRL0 0xff00b714
- NR4\_MCNR\_CM\_PRAM 0xff00b718
- NR4\_MCNR\_CM\_RSHFT\_ALP0 0xff00b71c
- NR4\_MCNR\_BLUE\_CENT 0xff00b720
- NR4\_MCNR\_BLUE\_GAIN\_PAR0 0xff00b724
- NR4\_MCNR\_BLUE\_GAIN\_PAR1 0xff00b728
- NR4\_MCNR\_CM\_BLUE\_CLIP0 0xff00b72c
- NR4\_MCNR\_CM\_BLUE\_CLIP1 0xff00b730

- NR4\_MCNR\_GREEN\_CENT 0xff00b734
- NR4\_MCNR\_GREEN\_GAIN\_PAR0 0xff00b738
- NR4\_MCNR\_GREEN\_GAIN\_PAR1 0xff00b73c
- NR4\_MCNR\_GREEN\_CLIP0 0xff00b740
- NR4\_MCNR\_GREEN\_CLIP2 0xff00b744
- NR4\_MCNR\_SKIN\_CENT 0xff00b748
- NR4\_MCNR\_SKIN\_GAIN\_PAR0 0xff00b74c
- NR4\_MCNR\_SKIN\_GAIN\_PAR1 0xff00b750
- NR4\_MCNR\_SKIN\_CLIP0 0xff00b754
- NR4\_MCNR\_SKIN\_CLIP1 0xff00b758
- NR4\_MCNR\_ALP1\_GLB\_CTRL 0xff00b75c
- NR4\_MCNR\_DC2NORM\_LUT0 0xff00b760
- NR4\_MCNR\_DC2NORM\_LUT1 0xff00b764
- NR4\_MCNR\_DC2NORM\_LUT2 0xff00b768
- NR4\_MCNR\_AC2NORM\_LUT0 0xff00b76c
- NR4\_MCNR\_AC2NORM\_LUT1 0xff00b770
- NR4\_MCNR\_AC2NORM\_LUT2 0xff00b774
- NR4\_MCNR\_SAD2ALP0\_LUT0 0xff00b778
- NR4\_MCNR\_SAD2ALP0\_LUT1 0xff00b77c
- NR4\_MCNR\_SAD2ALP0\_LUT2 0xff00b780
- NR4\_MCNR\_SAD2ALP0\_LUT3 0xff00b784
- NR4\_MCNR\_SAD2ALP1\_LUT0 0xff00b788
- NR4\_MCNR\_SAD2ALP1\_LUT1 0xff00b78c
- NR4\_MCNR\_SAD2ALP1\_LUT2 0xff00b790
- NR4\_MCNR\_SAD2ALP1\_LUT3 0xff00b794
- NR4\_MCNR\_SAD2BET0\_LUT0 0xff00b798
- NR4\_MCNR\_SAD2BET0\_LUT1 0xff00b79c
- NR4\_MCNR\_SAD2BET0\_LUT2 0xff00b7a0
- NR4\_MCNR\_SAD2BET0\_LUT3 0xff00b7a4
- NR4\_MCNR\_SAD2BET1\_LUT0 0xff00b7a8
- NR4\_MCNR\_SAD2BET1\_LUT1 0xff00b7ac
- NR4\_MCNR\_SAD2BET1\_LUT2 0xff00b7b0
- NR4\_MCNR\_SAD2BET1\_LUT3 0xff00b7b4
- NR4\_MCNR\_SAD2BET2\_LUT0 0xff00b7b8
- NR4\_MCNR\_SAD2BET2\_LUT1 0xff00b7bc
- NR4\_MCNR\_SAD2BET2\_LUT2 0xff00b7c0
- NR4\_MCNR\_SAD2BET2\_LUT3 0xff00b7c4
- NR4\_MCNR\_RO\_U\_SUM 0xff00b7c8
- NR4\_MCNR\_RO\_V\_SUM 0xff00b7cc
- NR4\_MCNR\_RO\_GRDU\_SUM 0xff00b7d0
- NR4\_MCNR\_RO\_GRDV\_SUM 0xff00b7d4
- NR4\_TOP\_CTRL 0xff00b7fc
- NR4\_MCNR\_SAD\_GAIN 0xff00dc00
- NR4\_MCNR\_LPF\_CTRL 0xff00dc04

- NR4\_MCNR\_BLD\_VS3LUT0 0xff00dc08
- NR4\_MCNR\_BLD\_VS3LUT1 0xff00dc0c
- NR4\_MCNR\_BLD\_VS3LUT2 0xff00dc10
- NR4\_MCNR\_BLD\_VS2LUT0 0xff00dc14
- NR4\_MCNR\_BLD\_VS2LUT1 0xff00dc18
- NR4\_COEFBLT\_LUT10 0xff00dc1c
- NR4\_COEFBLT\_LUT11 0xff00dc20
- NR4\_COEFBLT\_LUT12 0xff00dc24
- NR4\_COEFBLT\_LUT20 0xff00dc28
- NR4\_COEFBLT\_LUT21 0xff00dc2c
- NR4\_COEFBLT\_LUT22 0xff00dc30
- NR4\_COEFBLT\_LUT30 0xff00dc34
- NR4\_COEFBLT\_LUT31 0xff00dc38
- NR4\_COEFBLT\_LUT32 0xff00dc3c
- NR4\_COEFBLT\_CONV 0xff00dc40
- NR4\_DBGWIN\_YX0 0xff00dc44
- NR4\_DBGWIN\_YX1 0xff00dc48
- NR4\_NM\_X\_CFG 0xff00dc4c
- NR4\_NM\_Y\_CFG 0xff00dc50
- NR4\_NM\_SAD\_THD 0xff00dc54
- NR4\_MCNR\_BANDSPLIT\_PRAM 0xff00dc58
- NR4\_MCNR\_ALP1\_SGN\_COR 0xff00dc5c
- NR4\_MCNR\_ALP1\_SGN\_PRAM 0xff00dc60
- NR4\_MCNR\_ALP1\_MVX\_LUT1 0xff00dc64
- NR4\_MCNR\_ALP1\_MVX\_LUT2 0xff00dc68
- NR4\_MCNR\_ALP1\_MVX\_LUT3 0xff00dc6c
- NR4\_MCNR\_ALP1\_LP\_PRAM 0xff00dc70
- NR4\_MCNR\_ALP1\_SGN\_LUT1 0xff00dc74
- NR4\_MCNR\_ALP1\_SGN\_LUT2 0xff00dc78
- NR4\_RO\_NM\_SAD\_SUM 0xff00dc7c
- NR4\_RO\_NM\_SAD\_CNT 0xff00dc80
- NR4\_RO\_NM\_VAR\_SUM 0xff00dc84
- NR4\_RO\_NM\_VAR\_SCNT 0xff00dc88
- NR4\_RO\_NM\_VAR\_MIN\_MAX 0xff00dc8c
- NR4\_RO\_NR4\_DBGPIX\_NUM 0xff00dc90
- NR4\_RO\_NR4\_BLDVS2\_SUM 0xff00dc94
- NR4\_BLDVS3\_SUM 0xff00dc98
- NR4\_COEF12\_SUM 0xff00dc9c
- NR4\_COEF123\_SUM 0xff00dca0
- XLR\_CTRL 0xff00dcc0
- XLR\_THRD 0xff00dcc4
- XLR\_HCT\_THRD 0xff00dcc8
- NR\_DB\_FLT\_CTRL 0xff00dce0
- NR\_DB\_FLT\_YC\_THRD 0xff00dce4

- NR\_DB\_FLT\_RANLUT 0xff00dce8
- NR\_DB\_FLT\_PXI\_THRD 0xff00dcec
- NR\_DB\_FLT\_SEED\_Y 0xff00dcf0
- NR\_DB\_FLT\_SEED\_U 0xff00dcf4
- NR\_DB\_FLT\_SEED\_V 0xff00dcf8
- NR\_DB\_FLT\_SEED3 0xff00dcfc
- NRWR\_DBG\_AXI\_CMD\_CNT 0xff008240
- NRWR\_DBG\_AXI\_DAT\_CNT 0xff008244
- DI\_NRWR\_CANVAS 0xff008248
- DI\_NRWR\_URGENT 0xff00824c
- DI\_NRWR\_X 0xff008250
- DI\_NRWR\_Y 0xff008254
- DI\_NRWR\_CTRL 0xff008258
- DI\_NRWR\_SHRK\_CTRL 0xff00825c
- DI\_NRWR\_SHRK\_SIZE 0xff008260
- DI\_NRWR\_CROP\_CTRL 0xff008268
- DI\_NRWR\_CROP\_DIMM\_CTRL 0xff00826c
- DI\_NRWR\_CROP\_SIZE\_IN 0xff008270
- DI\_NRWR\_CROP\_HSCOPE 0xff008274
- DI\_NRWR\_CROP\_VSCOPE 0xff008278
- DIWR\_DBG\_AXI\_CMD\_CNT 0xff0083c0
- DIWR\_DBG\_AXI\_DAT\_CNT 0xff0083c4
- DI\_DIWR\_CANVAS 0xff0083c8
- DI\_DIWR\_URGENT 0xff0083cc
- DI\_DIWR\_X 0xff0083d0
- DI\_DIWR\_Y 0xff0083d4
- DI\_DIWR\_CTRL 0xff0083d8
- DI\_DIWR\_SHRK\_CTRL 0xff0083dc
- DI\_DIWR\_SHRK\_SIZE 0xff0083e0
- DI\_DIWR\_CROP\_CTRL 0xff0083e8
- DI\_DIWR\_CROP\_DIMM\_CTRL 0xff0083ec
- DI\_DIWR\_CROP\_SIZE\_IN 0xff0083f0
- DI\_DIWR\_CROP\_HSCOPE 0xff0083f4
- DI\_DIWR\_CROP\_VSCOPE 0xff0083f8

## Register Description

**Table 9-1173 DNR\_CTRL 0x2d00**

Bit(s)	R/W	Default	Description
31:20	R/W		reserved
19	R/W		reg_dnr_bufctrl, deblock linebuf control, 0 : 3line 1: 5line, default =1
18	R/W		reg_dnr_dm_bufctrl, demosquito linebuf control, 0 : 3line 1: 5line, default =1
16	R/W		reg_dnr_en
15	R/W		reg_dnr_db_vdbstep, vdb step, 0: 4, 1: 8 . unsigned, default = 1

Bit(s)	R/W	Default	Description
14	R/W		reg_dnr_db_vdbprten, vdb protection enable . unsigned, default = 1
13	R/W		reg_dnr_gbs_difen, enable dif (between LR and LL/RR) condition for gbs stat.. unsigned, default = 0
12	R/W		reg_dnr_luma_en, enable ycbcr2luma module . unsigned, default = 1
11:10	R/W		reg_dnr_db_mod, deblocking mode, 0: disable, 1: horizontal deblocking, 2: vertical deblocking, 3: horizontal & vertical deblocking. unsigned, default = 3
9	R/W		reg_dnr_db_chrmn, enable chroma deblocking . unsigned, default = 1
8	R/W		reg_dnr_hvdif_mod, 0: calc. difs by original Y, 1: by new luma. unsigned, default = 1
7	R/W		reserved
6: 4	R/W		reg_dnr_demo_lften, b0: Y b1:U b2:V. unsigned, default = 7
3	R/W		reserved
2: 0	R/W		reg_dnr_demo_rgten, b0: Y b1:U b2:V. unsigned, default = 7

**Table 9-1174 DNR\_HVSIZE 0x2d01**

Bit(s)	R/W	Default	Description
31:29	R/W		reserved
28:16	R/W		reg_dnr_hsize, hsize . unsigned, default = 0
15:13	R/W		reserved
12: 0	R/W		reg_dnr_vsize, vsize . unsigned, default = 0

**Table 9-1175 DNR\_DBLK\_BLANK\_NUM 0x2d02**

Bit(s)	R/W	Default	Description
31:16	R/W		reserved
15: 8	R/W		reg_dblk_hblank_num, deblock hor blank num . unsigned, default = 16
7: 0	R/W		reg_dblk_vblank_num, deblock ver blank num . unsigned, default = 45

**Table 9-1176 DNR\_BLK\_OFFST 0x2d03**

Bit(s)	R/W	Default	Description
31: 7	R/W		reserved
6: 4	R/W		reg_dnr_hbofst, horizontal block offset may provide by software calc.. unsigned, default = 0
3	R/W		reserved
2: 0	R/W		reg_dnr_vbofst, vertical block offset may provide by software calc.. unsigned, default = 0

**Table 9-1177 DNR\_GBS 0x2d04**

Bit(s)	R/W	Default	Description
31: 2	R/W		reserved
1: 0	R/W		reg_dnr_gbs , global block strength may update by software calc.. unsigned, default = 0

**Table 9-1178 DNR\_HBOFFST\_STAT 0x2d05**

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_hbof_difthd, dif threshold ( $\geq$ ) between LR and LL/RR. unsigned, default = 2
23:16	R/W		reg_dnr_hbof_edgethd , edge threshold ( $\leq$ ) for LR . unsigned, default = 32
15: 8	R/W		reg_dnr_hbof_flatthd , flat threshold ( $\geq$ ) for LR . unsigned, default = 0
7	R/W		reserved
6: 4	R/W		reg_dnr_hbof_delta, delta for weighted bin accumulator. unsigned, default = 1
3	R/W		reserved
2: 0	R/W		reg_dnr_hbof_statmod , statistic mode for horizontal block offset, 0: count flags for 8-bin, 1: count LRs for 8-bin, 2: count difs for 8-bin, 3: count weighted flags for 8-bin, 4: count flags for first 32-bin, 5: count LRs for first 32-bin, 6 or 7: count difs for first 32-bin. unsigned, default = 2

**Table 9-1179 DNR\_VBOFFST\_STAT 0x2d06**

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_vbof_difthd, dif threshold ( $\geq$ ) between Up and Dw. unsigned, default = 1
23:16	R/W		reg_dnr_vbof_edgethd , edge threshold ( $\leq$ ) for Up/Dw. unsigned, default = 16
15: 8	R/W		reg_dnr_vbof_flatthd , flat threshold ( $\geq$ ) for Up/Dw. unsigned, default = 0
7	R/W		reserved
6: 4	R/W		reg_dnr_vbof_delta, delta for weighted bin accumulator. unsigned, default = 1
3	R/W		reserved
2: 0	R/W		reg_dnr_vbof_statmod , statistic mode for vertical block offset, 0: count flags for 8-bin, 1: count Ups for 8-bin, 2: count difs for 8-bin, 3: count weighted flags for 8-bin, 4: count flags for first 32-bin, 5: count Ups for first 32-bin, 6 or 7: count difs for first 32-bin. unsigned, default = 2

**Table 9-1180 DNR\_GBS\_STAT 0x2d07**

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_gbs_edgethd, edge threshold ( $\leq$ ) for LR . unsigned, default = 32
23:16	R/W		reg_dnr_gbs_flatthd, flat threshold ( $\geq$ ) for LR . unsigned, default = 0
15: 8	R/W		reg_dnr_gbs_varthd, variation threshold ( $\leq$ ) for Lvar/Rvar. unsigned, default = 16
7: 0	R/W		reg_dnr_gbs_difthd, dif threshold ( $\geq$ ) between LR and LL/RR. unsigned, default = 2



**Table 9-1181 DNR\_STAT\_X\_START\_END 0x2d08**

Bit(s)	R/W	Default	Description
31:30	R/W		reserved
29:16	R/W		reg_dnr_stat_xst . unsigned, default = 24
15:14	R/W		reserved
13: 0	R/W		reg_dnr_stat_xed . unsigned, default = HSIZE - 25

**Table 9-1182 DNR\_STAT\_Y\_START\_END 0x2d09**

Bit(s)	R/W	Default	Description
31:30	R/W		reserved
29:16	R/W		reg_dnr_stat_yst . unsigned, default = 24
15:14	R/W		reserved
13: 0	R/W		reg_dnr_stat_yed . unsigned, default = VSIZE - 25

**Table 9-1183 DNR\_LUMA 0x2d0a**

Bit(s)	R/W	Default	Description
31:27	R/W		reserved
28	R/W		Reg_dnr_luma_mode, ycbcr2luma mode selection, default =0
26:24	R/W		reg_dnr_luma_sqrtshft , left shift for fast squart of chroma, [0, 4]. unsigned, default = 2
23:21	R/W		reserved
20:16	R/W		reg_dnr_luma_sqrtoffst, offset for fast squart of chroma. signed, default = 0
15	R/W		reserved
14:12	R/W		reg_dnr_luma_wcmmod, theta related to warm/cool segment line, 0: 0, 1: 45, 2: 90, 3: 135, 4: 180, 5: 225, 6: 270, 7: 315. . unsigned, default = 3
11: 8	R/W		reg_dnr_luma_cshft, shift for calc. delta part, 0~8, . unsigned, default = 8
7: 6	R/W		reserved
5: 0	R/W		reg_dnr_luma_cgain, final gain for delta part, 32 normalized to "1". unsigned, default = 4

**Table 9-1184 DNR\_DB\_YEDGE\_THD 0x2d0b**

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_db_yedgethd0 , edge threshold0 for luma . unsigned, default = 12
23:16	R/W		reg_dnr_db_yedgethd1 , edge threshold1 for luma . unsigned, default = 15
15: 8	R/W		reg_dnr_db_yedgethd2 , edge threshold2 for luma . unsigned, default = 18
7: 0	R/W		reg_dnr_db_yedgethd3 , edge threshold3 for luma . unsigned, default = 25

**Table 9-1185 DNR\_DB\_CEDGE\_THD 0x2d0c**

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_db_cedgethd0 , edge threshold0 for chroma . unsigned, default = 12
23:16	R/W		reg_dnr_db_cedgethd1 , edge threshold1 for chroma . unsigned, default = 15
15: 8	R/W		reg_dnr_db_cedgethd2 , edge threshold2 for chroma . unsigned, default = 18
7: 0	R/W		reg_dnr_db_cedgethd3 , edge threshold3 for chroma . unsigned, default = 25

**Table 9-1186 DNR\_DB\_HGAP 0x2d0d**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:16	R/W		reg_dnr_db_hgapthd, horizontal gap thd (<=) for very sure blockiness . unsigned, default = 8
15: 8	R/W		reg_dnr_db_hgapdifthd , dif thd between hgap and lft/rgt hdifs. unsigned, default = 1
7: 1	R/W		reserved
0	R/W		reg_dnr_db_hgapmod, horizontal gap calc. mode, 0: just use current col x, 1: find max between (x-1, x, x+1) . unsigned, default = 0

**Table 9-1187 DNR\_DB\_HBS 0x2d0e**

Bit(s)	R/W	Default	Description
31: 6	R/W		reserved
5: 4	R/W		reg_dnr_db_hbsup , horizontal bs up value . unsigned, default = 1
3: 2	R/W		reg_dnr_db_hbsmax , max value of hbs for global control. unsigned, default = 3
1: 0	R/W		reg_dnr_db_hgbsth, gbs thd (>=) for hbs calc. . unsigned, default = 1

**Table 9-1188 DNR\_DB\_HACT 0x2d0f**

Bit(s)	R/W	Default	Description
31:16	R/W		reserved
15: 8	R/W		reg_dnr_db_hactthd0, thd0 of hact, for block classification. unsigned, default = 10
7: 0	R/W		reg_dnr_db_hactthd1, thd1 of hact, for block classification. unsigned, default = 32

**Table 9-1189 DNR\_DB\_YHDELTA\_GAIN 0x2d10**

Bit(s)	R/W	Default	Description
31:27	R/W		reserved
26:24	R/W		reg_dnr_db_yhdeltagain1, (p1-q1) gain for Y's delta calc. when bs=1, normalized 8 as "1" . unsigned, default = 2
23	R/W		reserved

Bit(s)	R/W	Default	Description
22:20	R/W		reg_dnr_db_yhdeltagain2, (p1-q1) gain for Y's delta calc. when bs=2, normalized 8 as "1" . unsigned, default = 0
19	R/W		reserved
18:16	R/W		reg_dnr_db_yhdeltagain3, (p1-q1) gain for Y's delta calc. when bs=3, normalized 8 as "1" . unsigned, default = 0
15	R/W		reserved
14: 8	R/W		reg_dnr_db_yhdeltaadjoffst, offset for adjust Y's hdelta (-64, 63). signed, default = 0
7: 6	R/W		reserved
5: 0	R/W		reg_dnr_db_yhdeltaadjgain , gain for adjust Y's hdelta, normalized 32 as "1" . unsigned, default = 32

Table 9-1190 DNR\_DB\_YHDELTA2\_GAIN 0x2d11

Bit(s)	R/W	Default	Description
31:30	R/W		reserved
29:24	R/W		reg_dnr_db_yhdelta2gain2 , gain for bs=2's adjust Y's hdelta2, normalized 64 as "1" . unsigned, default = 8
23:21	R/W		reserved
20:16	R/W		reg_dnr_db_yhdelta2offst2 , offset for bs=2's adjust Y's hdelta2 (-16, 15). signed, default = 0
15:14	R/W		reserved
13: 8	R/W		reg_dnr_db_yhdelta2gain3 , gain for bs=3's adjust Y's hdelta2, normalized 64 as "1" . unsigned, default = 4
7: 5	R/W		reserved
4: 0	R/W		reg_dnr_db_yhdelta2offst3 , offset for bs=3's adjust Y's hdelta2 (-16, 15). signed, default = 0

Table 9-1191 DNR\_DB\_CHDELTA\_GAIN 0x2d12

Bit(s)	R/W	Default	Description
31:27	R/W		reserved
26:24	R/W		reg_dnr_db_chdeltagain1, (p1-q1) gain for UV's delta calc. when bs=1, normalized 8 as "1". unsigned, default = 2
23	R/W		reserved
22:20	R/W		reg_dnr_db_chdeltagain2, (p1-q1) gain for UV's delta calc. when bs=2, normalized 8 as "1". unsigned, default = 0
19	R/W		reserved
18:16	R/W		reg_dnr_db_chdeltagain3, (p1-q1) gain for UV's delta calc. when bs=3, normalized 8 as "1". unsigned, default = 0
15	R/W		reserved
14: 8	R/W		reg_dnr_db_chdeltaadjoffst, offset for adjust UV's hdelta (-64, 63). signed, default = 0

Bit(s)	R/W	Default	Description
7: 6	R/W		reserved
5: 0	R/W		reg_dnr_db_chdeltaadjgain , gain for adjust UV's hdelta, normalized 32 as "1". unsigned, default = 32

Table 9-1192 DNR\_DB\_CHDELTA2\_GAIN 0x2d13

Bit(s)	R/W	Default	Description
31:30	R/W		reserved
29:24	R/W		reg_dnr_db_chdelta2gain2 , gain for bs=2's adjust UV's hdelta2, normalized 64 as "1" . unsigned, default = 8
23:21	R/W		reserved
20:16	R/W		reg_dnr_db_chdelta2offst2 , offset for bs=2's adjust UV's hdelta2 (-16, 15). signed, default = 0
15:14	R/W		reserved
13: 8	R/W		reg_dnr_db_chdelta2gain3 , gain for bs=2's adjust UV's hdelta2, normalized 64 as "1" . unsigned, default = 4
7: 5	R/W		reserved
4: 0	R/W		reg_dnr_db_chdelta2offst3 , offset for bs=2's adjust UV's hdelta2 (-16, 15). signed, default = 0

Table 9-1193 DNR\_DB\_YC\_VEDGE\_THD 0x2d14

Bit(s)	R/W	Default	Description
31:16	R/W		reserved
15: 8	R/W		reg_dnr_db_yvedgethd , special Y's edge thd for vdb. unsigned, default = 12
7: 0	R/W		reg_dnr_db_cvedgethd , special UV's edge thd for vdb. unsigned, default = 12

Table 9-1194 DNR\_DB\_VBS\_MISC 0x2d15

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_db_vgapthd, vertical gap thd (<=) for very sure blockiness . unsigned, default = 8
23:16	R/W		reg_dnr_db_vactthd, thd of vact, for block classification . unsigned, default = 10
15: 8	R/W		reg_dnr_db_vgapdifthd , dif thd between vgap and vact. unsigned, default = 4
7: 4	R/W		reserved
3: 2	R/W		reg_dnr_db_vbsmax , max value of vbs for global control. unsigned, default = 2
1: 0	R/W		reg_dnr_db_vgbsth, gbs thd (>=) for vbs calc. . unsigned, default = 1

**Table 9-1195 DNR\_DB\_YVDELTA\_GAIN 0x2d16**

Bit(s)	R/W	Default	Description
31:30	R/W		reserved
29:24	R/W		reg_dnr_db_yvdeltaadjgain , gain for adjust Y's vdelta, normalized 32 as "1". unsigned, default = 32
23	R/W		reserved
22:16	R/W		reg_dnr_db_yvdeltaadjfst, offset for adjust Y's vdelta (-64, 63). signed, default = 0
15:14	R/W		reserved
13: 8	R/W		reg_dnr_db_yvdelta2gain, gain for adjust Y's vdelta2, normalized 64 as "1". unsigned, default = 8
7: 5	R/W		reserved
4: 0	R/W		reg_dnr_db_yvdelta2offst , offset for adjust Y's vdelta2 (-16, 15). signed, default = 0

**Table 9-1196 DNR\_DB\_CVDELTA\_GAIN 0x2d17**

Bit(s)	R/W	Default	Description
31:30	R/W		reserved
29:24	R/W		reg_dnr_db_cvdeltaadjgain , gain for adjust UV's vdelta, normalized 32 as "1". unsigned, default = 32
23	R/W		reserved
22:16	R/W		reg_dnr_db_cvdeltaadjfst, offset for adjust UV's vdelta (-64, 63). signed, default = 0
15:14	R/W		reserved
13: 8	R/W		reg_dnr_db_cvdelta2gain, gain for adjust UV's vdelta2, normalized 64 as "1". unsigned, default = 8
7: 5	R/W		reserved
4: 0	R/W		reg_dnr_db_cvdelta2offst , offset for adjust UV's vdelta2 (-16, 15). signed, default = 0

**Table 9-1197 DNR\_RO\_GBS\_STAT\_LR 0x2d19**

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_lr . unsigned, default = 0

**Table 9-1198 DNR\_RO\_GBS\_STAT\_LL 0x2d18**

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_ll . unsigned, default = 0

**Table 9-1199 DNR\_RO\_GBS\_STAT\_RR 0x2d1a**

Bit(s)	R/W	Default	Description
31:0	R		ro_gbs_stat_rr . unsigned, default = 0

**Table 9-1200 DNR\_RO\_GBS\_STAT\_DIF 0x2d1b**

Bit(s)	R/W	Default	Description
31:0	R		ro_gbs_stat_dif . unsigned, default = 0

**Table 9-1201 DNR\_RO\_GBS\_STAT\_CNT 0x2d1c**

Bit(s)	R/W	Default	Description
31:0	R		ro_gbs_stat_cnt . unsigned, default = 0

**Table 9-1202 DNR\_RO\_HBOF\_STAT\_CNT\_0 0x2d1d**

Bit(s)	R/W	Default	Description
31:0	R		ro_hbof_stat_cnt0. unsigned, default = 0

**Table 9-1203 DNR\_RO\_HBOF\_STAT\_CNT\_31 0x2d3c**

Bit(s)	R/W	Default	Description
31:0	R		ro_hbof_stat_cnt31 . unsigned, default = 0

**Table 9-1204 DNR\_RO\_VBOF\_STAT\_CNT\_0 0x2d3d**

Bit(s)	R/W	Default	Description
31:0	R		ro_vbof_stat_cnt0. unsigned, default = 0

**Table 9-1205 DNR\_RO\_VBOF\_STAT\_CNT\_31 0x2d5c**

Bit(s)	R/W	Default	Description
31:0	R		ro_vbof_stat_cnt31. unsigned, default = 0

**Table 9-1206 DNR\_DM\_ADP\_EN 0x2d5d**

Bit(s)	R/W	Default	Description
31:8	R/W		reserved
7	R/W	1	Reg_dnr_scene_change_flag // scence change flag for dnr
6	R/W	1	Reg_dnr_dm_lpf_en // enable lpf fof demosquito filter
5	R/W	1	Reg_dnr_dm_adp_level_en // enable adptive demosquito level
4	R/W	1	Reg_dnr_dm_flag2bdif_en // enable edge flag to blkdif calc

Bit(s)	R/W	Default	Description
3:2	R/W	1	Reg_dnr_dm_edgeiir // edge flag iir mode, 0: cur, 1: max(pre,cur), 2,3: choice by org mtn
1	R/W	1	Reg_dnr_dm_dirdifmod // dif mode for direction calc , 0: abs(dif02), 1: (abs(dif01)+abs(dif21))/2
0	R/W	0	Reg_dnr_dm_sur_dir_mod // sure direction mode for cordif calc

Table 9-1207 DNR\_DM\_EDGE\_DIR 0x2d5e

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:16	R/W	2	Reg_dnr_dm_dirdifcor // dif coring threshold for direction calc
15:8	R/W	160	Reg_dnr_dm_edgewardifhd // edge flag down when cordif is lager than threshold
7:0	R/W	60	Reg_dnr_dm_mtnrt // motion ration for mtn decision

Table 9-1208 DNR\_DM\_CTRL 0x2d60

Bit(s)	R/W	Default	Description
31:14	R/W		reserved
13	R/W	1	Reg_dnr_dm_fedgeflag_en, enable edge flag calc. Of each frame
12	R/W	1	Reg_dnr_dm_fedgeflag_cl, clear frame edge flag if needed
11:10	R/W	1	Reg_dnr_dm_fedgeflg_df, user defined edge when fedge_flg_en=0
9	R/W	1	Reg_dnr_dm_en, enable demosquito function
8	R/W	1	Reg_dnr_dm_chrmn, enable chroma proc. For demosquito
7:6	R/W	3	Reg_dnr_dm_level, demosquito level
5:4	R/W	1	Reg_dnr_dm_leveldw0, level down when gbs is small
3:2	R/W	1	Reg_dnr_dm_level_dw1, level down for no edge/flat blocks
1:0	R/W	0	Reg_dnr_dm_gbsthld, small/large threshold for gbs <=

Table 9-1209 DNR\_DM\_NR\_BLND 0x2d61

Bit(s)	R/W	Default	Description
31:25	R/W		reserved
24	R/W	0	reg_dnr_dm_defalpen, 1 to enable user defined alpha for DM/NR blend
23:16	R/W	0	reg_dnr_dm_defalp , user defined alpha for DM/NR blend
15	R/W		reserved
14:9	R/W	32	reg_dnr_dm_yalpgain , gain for DM/NR alpha, normalized 32 as 1

**Table 9-1210 DNR\_DM\_RNG\_THD 0x2d62**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:16	R/W	2	reg_dnr_dm_rgnminthd
15: 8	R/W	64	reg_dnr_dm_rgnmaxthd

**Table 9-1211 DNR\_DM\_RNG\_GAIN\_OFST 0x2d63**

Bit(s)	R/W	Default	Description
31:14	R/W		reserved
13: 8	R/W	16	reg_dnr_dm_rnggain, normalized 16 as 1

**Table 9-1212 DNR\_DM\_DIR\_MISC 0x2d64**

Bit(s)	R/W	Default	Description
31:30	R/W		reserved
28:24	R/W	0	reg_dnr_dm_diralpgain
23:22	R/W		reserved
21:16	R/W	0	reg_dnr_dm_diralpofst
15:13	R/W		reserved
12: 8	R/W	0	reg_dnr_dm_diralpmin

**Table 9-1213 DNR\_DM\_COR\_DIF 0x2d65**

Bit(s)	R/W	Default	Description
31:6	R/W		reserved

**Table 9-1214 DNR\_DM\_FLT\_THD 0x2d66**

Bit(s)	R/W	Default	Description
31:24	R/W	4	reg_dnr_dm_fitthd00, block flat threshold0 for block average difference when gbs is small
23:16	R/W	6	reg_dnr_dm_fitthd01, block flat threshold1 for block average difference when gbs is small
15: 8	R/W	9	reg_dnr_dm_fitthd10, block flat threshold0 for block average difference when gbs is larger



**Table 9-1215 DNR\_DM\_VAR\_THD 0x2d67**

Bit(s)	R/W	Default	Description
31:24	R/W	2	reg_dnr_dm_varthd00 , block variance threshold0 ( $\geq$ ) when gbs is small
23:16	R/W	15	reg_dnr_dm_varthd01 , block variance threshold1 ( $\leq$ ) when gbs is small
15: 8	R/W	3	reg_dnr_dm_varthd10 , block variance threshold0 ( $\geq$ ) when gbs is larger

**Table 9-1216 DNR\_DM\_EDGE\_DIF\_THD 0x2d68**

Bit(s)	R/W	Default	Description
31:24	R/W	32	reg_dnr_dm_edgethd0 , block edge threshold ( $\leq$ ) when gbs is small
23:16	R/W	48	reg_dnr_dm_edgethd1 , block edge threshold ( $\leq$ ) when gbs is larger
15: 8	R/W	48	reg_dnr_dm_difhd0 , block dif threshold ( $\leq$ ) when gbs is small

**Table 9-1217 DNR\_DM\_AVG\_THD 0x2d69**

Bit(s)	R/W	Default	Description
31:16	R/W		reserved
15: 8	R/W	160	reg_dnr_dm_avgthd0 , block average threshold ( $\geq$ ) when gbs is small

**Table 9-1218 DNR\_DM\_AVG\_VAR\_DIF\_THD 0x2d6a**

Bit(s)	R/W	Default	Description
31:16	R/W		reserved
15: 8	R/W	12	reg_dnr_dm_avgdifhd , block average dif threshold( $<$ ) between cur and up block for flat block

**Table 9-1219 DNR\_DM\_EDGE\_DIF\_THD2 0x2d6b**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:16	R/W	24	reg_dnr_dm_varthd2, block variance threshold ( $\geq$ ) for edge block detect
15: 8	R/W	40	reg_dnr_dm_edgethd2, block edge threshold ( $\geq$ )

**Table 9-1220 DNR\_DM\_DIF\_FLT\_MISC 0x2d6c**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reg_dnr_dm_ldifoob, pre-defined large dif when pixel out of block
27:24	R/W	0	reg_dnr_dm_bdifoob, pre-defined block dif when pixel out of block
23:16	R/W	200	reg_dnr_dm_ftalp, pre-defined alpha for dm and nr blending when block is flat with mos

Bit(s)	R/W	Default	Description
15:12	R/W		reserved
11:8	R/W	12	reg_dnr_dm_ftminbdif, pre-defined min block dif for dm filter when block is flat with mos

Table 9-1221 DNR\_DM\_SDIF\_LUT0\_2 0x2d6d

Bit(s)	R/W	Default	Description
31:21	R/W		reserved
20:16	R/W	16	reg_dnr_dm_sdiflut0, normally 0-16
15:13	R/W		reserved
12:8	R/W	14	reg_dnr_dm_sdiflut1
7:5	R/W		reserved

Table 9-1222 DNR\_DM\_SDIF\_LUT3\_5 0x2d6e

Bit(s)	R/W	Default	Description
31:21	R/W		reserved
20:16	R/W	10	reg_dnr_dm_sdiflut3
15:13	R/W		reserved
12:8	R/W	7	reg_dnr_dm_sdiflut4
7:5	R/W		reserved

Table 9-1223 DNR\_DM\_SDIF\_LUT6\_8 0x2d6f

Bit(s)	R/W	Default	Description
31:21	R/W		reserved
20:16	R/W	3	reg_dnr_dm_sdiflut6
15:13	R/W		reserved
12:8	R/W	1	reg_dnr_dm_sdiflut7
7:5	R/W		reserved

Table 9-1224 DNR\_DM\_LDIF\_LUT0\_2 0x2d70

Bit(s)	R/W	Default	Description
31:21	R/W		reserved
20:16	R/W	0	reg_dnr_dm_ldiflut0
15:13	R/W		reserved
12:8	R/W	4	reg_dnr_dm_ldiflut1
7:5	R/W		reserved

**Table 9-1225 DNR\_DM\_LDIF\_LUT3\_5 0x2d71**

Bit(s)	R/W	Default	Description
31:21	R/W		reserved
20:16	R/W	14	reg_dnr_dm_ldiflut3
15:13	R/W		reserved
12:8	R/W	15	reg_dnr_dm_ldiflut4
7:5	R/W		reserved

**Table 9-1226 DNR\_DM\_LDIF\_LUT6\_8 0x2d72**

Bit(s)	R/W	Default	Description
31:21	R/W		reserved
20:16	R/W	16	reg_dnr_dm_ldiflut6
15:13	R/W		reserved
12:8	R/W	16	reg_dnr_dm_ldiflut7
7:5	R/W		reserved

**Table 9-1227 DNR\_DM\_DIF2NORM\_LUT0\_2 0x2d73**

Bit(s)	R/W	Default	Description
31:21	R/W		reserved
20:16	R/W	16	reg_dnr_dm_dif2normlut0
15:13	R/W		reserved
12:8	R/W	5	reg_dnr_dm_dif2normlut1
7:5	R/W		reserved

**Table 9-1228 DNR\_DM\_DIF2NORM\_LUT3\_5 0x2d74**

Bit(s)	R/W	Default	Description
31:21	R/W		reserved
20:16	R/W	2	reg_dnr_dm_dif2normlut3
15:13	R/W		reserved
12:8	R/W	2	reg_dnr_dm_dif2normlut4
7:5	R/W		reserved

**Table 9-1229 DNR\_DM\_DIF2NORM\_LUT6\_8 0x2d75**

Bit(s)	R/W	Default	Description
31:21	R/W		reserved
20:16	R/W	1	reg_dnr_dm_dif2normlut6

Bit(s)	R/W	Default	Description
15:13	R/W		reserved
12:8	R/W	1	reg_dnr_dm_dif2normlut7
7:5	R/W		reserved

Table 9-1230 DNR\_DM\_GMS\_THD 0x2d76

Bit(s)	R/W	Default	Description
31:16	R/W		reserved
15:8	R/W	0	reg_gms_stat_thd0

Table 9-1231 DNR\_RO\_DM\_GMS\_STST\_CNT 0x2d77

Bit(s)	R/W	Default	Description
31:0	RO		ro_dm_gms_stat_cnt

Table 9-1232 DNR\_RO\_DM\_GMS\_STST\_MS 0x2d78

Bit(s)	R/W	Default	Description
31:0	RO		ro_dm_gms_stat_ms

Table 9-1233 DNR\_DM\_EDGE\_GAIN 0x2d79

Bit(s)	R/W	Default	Description
31:24	R/W	8	reg_dnr_dm_alpedgegain0 : // unsigned , default = 8 , edge based gain for alpha, normalized 16 as "1"
23:16	R/W	16	reg_dnr_dm_alpedgegain1 : // unsigned , default = 16 , edge based gain for alpha, normalized 16 as "1"
15: 8	R/W	24	reg_dnr_dm_alpedgegain2 : // unsigned , default = 24 , edge based gain for alpha, normalized 16 as "1"
7: 0	R/W	32	reg_dnr_dm_alpedgegain3 : // unsigned , default = 32 , edge based gain for alpha, normalized 16 as "1"

Table 9-1234 DNR\_DM\_FLG\_BDIF 0x2d7a

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_dnr_dm_flg2bdif0 : // unsigned , default = 0 , (0-16), edge flg to blkdif calc.
23:16	R/W	96	reg_dnr_dm_flg2bdif1 : // unsigned , default = 96 , (0-16), edge flg to blkdif calc.
15: 8	R/W	160	reg_dnr_dm_flg2bdif2 : // unsigned , default = 160 , (0-16), edge flg to blkdif calc.
7: 0	R/W	192	reg_dnr_dm_flg2bdif3 : // unsigned , default = 192 , (0-16), edge flg to blkdif calc.

**Table 9-1235 DNR\_DM\_GBS\_RORM 0x2d7b**

Bit(s)	R/W	Default	Description
15:12	R/W	0	reg_dnr_dm_gbs4difnorm0 : // unsigned , default = 0 , gbs=0 for dif norm calc.
11: 8	R/W	1	reg_dnr_dm_gbs4difnorm1 : // unsigned , default = 1 , gbs=1 for dif norm calc.
7: 4	R/W	4	reg_dnr_dm_gbs4difnorm2 : // unsigned , default = 4 , gbs=2 for dif norm calc.
3: 0	R/W	6	reg_dnr_dm_gbs4difnorm3 : // unsigned , default = 6 , gbs=3 for dif norm calc.

**Table 9-1236 DNR\_DM\_FLG\_LEV 0x2d7c**

Bit(s)	R/W	Default	Description
13:12	R/W	1	reg_dnr_dm_flg2lev0 : // unsigned , default = 1 , edge flg to filter level calc.
9: 8	R/W	3	reg_dnr_dm_flg2lev1 : // unsigned , default = 3 , edge flg to filter level calc.
5: 4	R/W	3	reg_dnr_dm_flg2lev2 : // unsigned , default = 3 , edge flg to filter level calc.
1: 0	R/W	3	reg_dnr_dm_flg2lev3 : // unsigned , default = 3 , edge flg to filter level calc.

**Table 9-1237 DNR\_DM\_DIF\_FLG\_TH 0x2d7d**

Bit(s)	R/W	Default	Description
15: 8	R/W	128	reg_dnr_dm_dif2flgthd1 : // unsigned , default = 128 , dif to edge flg threshold 1
7: 0	R/W	192	reg_dnr_dm_dif2flgthd2 : // unsigned , default = 192 , dif to edge flg threshold 2

**Table 9-1238 DNR\_DM\_CALP\_GAIN\_OFST 0x2d7e**

Bit(s)	R/W	Default	Description
21:16	R/W	16	reg_dnr_dm_calpgain : // unsigned , default = 16 , chroma gain for nr/dm alpha, normalized 32 as "1"
8: 0	R/W	-64	reg_dnr_dm_calpoffst : // signed , default = -64 , (-255, 255), chroma offset for nr/dm alpha

**Table 9-1239 NR2\_POLAR3\_MODE 0x2d98**

Bit(s)	R/W	Default	Description
19:18	R/W	3	reg_polar3_f02lpf_mod_0 : default = 3 //u2x2: low pass filter mode for field 0 and field2 before polar3 detection; 0 for no lpf, 1: [1 2 1]/4 vert lpf; 2: [1 2 1; 2 4 2; 1 2 1]/16 2d lpf, p1 no hlpf; 2: [1 2 1; 2 4 2; 1 2 1]/16 2d lpf, p1 [1 2 1]/4 hlpf
17:16	R/W	3	reg_polar3_f02lpf_mod_1 : default = 3 //u2x2: low pass filter mode for field 0 and field2 before polar3 detection; 0 for no lpf, 1: [1 2 1]/4 vert lpf; 2: [1 2 1; 2 4 2; 1 2 1]/16 2d lpf, p1 no hlpf; 2: [1 2 1; 2 4 2; 1 2 1]/16 2d lpf, p1 [1 2 1]/4 hlpf
15:8	R/W	5	reg_polar3_dif02_thrd_0 : default = 5 //u8x2: threshold of dif for polar3 detection except for 32 detection, only do polar3 detection on obvious motion, [0] for luma, 1[1] for chroma
7:0	R/W	5	reg_polar3_dif02_thrd_1 : default = 5 //u8x2: threshold of dif for polar3 detection except for 32 detection, only do polar3 detection on obvious motion, [0] for luma, 1[1] for chroma

Table 9-1240 NR2\_POLAR3\_THRD 0x2d99

Bit(s)	R/W	Default	Description
31:24	R/W	30	reg_polar3_txf02_thrd_0 : default = 30 //u8x2: threshold to vertical f0f2 texture, if texture larger than this threshold, will not do the polar3 decision.
23:16	R/W	30	reg_polar3_txf02_thrd_1 : default = 30 //u8x2: threshold to vertical f0f2 texture, if texture larger than this threshold, will not do the polar3 decision.
15:8	R/W	20	reg_polar3_txf1_thrd_0 : default = 20 //u8x2: threshold to vertical f0f2 texture, if texture larger than this threshold, will not do the polar3 decision.
7:0	R/W	20	reg_polar3_txf1_thrd_1 : default = 20 //u8x2: threshold to vertical f0f2 texture, if texture larger than this threshold, will not do the polar3 decision.

Table 9-1241 NR2\_POLAR3\_PARA0 0x2d9a

Bit(s)	R/W	Default	Description
31:28	R/W	6	reg_polar3_rate0_0 : default = 6 //u4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1$ within $((f_0 + f_2) / 2 - \text{delt})$ , $((f_0 + f_2) / 2 + \text{delt})$ , then polar3_smoothmv++;
27:24	R/W	6	reg_polar3_rate0_1 : default = 6 //u4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1$ within $((f_0 + f_2) / 2 - \text{delt})$ , $((f_0 + f_2) / 2 + \text{delt})$ , then polar3_smoothmv++;
23:20	R/W	8	reg_polar3_rate1_0 : default = 8 //u4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < ((f_0 + f_2) / 2 - \text{delt})$ , then polar3_m1++; if $f_1 > ((f_0 + f_2) / 2 + \text{delt})$ , then polar3_p1++;
19:16	R/W	8	reg_polar3_rate1_1 : default = 8 //u4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < ((f_0 + f_2) / 2 - \text{delt})$ , then polar3_m1++; if $f_1 > ((f_0 + f_2) / 2 + \text{delt})$ , then polar3_p1++;
15:12	R/W	2	reg_polar3_rate2_0 : default = 2 //u4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < (f_2 - \text{delt} - \text{ofset2})$ , then polar3_m2++; if $f_1 > ((f_0 + \text{delt} + \text{ofset2})$ , then polar3_p2++;
11:8	R/W	2	reg_polar3_rate2_1 : default = 2 //u4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < (f_2 - \text{delt} - \text{ofset2})$ , then polar3_m2++; if $f_1 > ((f_0 + \text{delt} + \text{ofset2})$ , then polar3_p2++;
7:4	R/W	1	reg_polar3_ofst1_0 : default = 1 //s4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < (f_2 - \text{delt} - \text{ofset2})$ , then polar3_m2++; if $f_1 > ((f_0 + \text{delt} + \text{ofset2})$ , then polar3_p2++;
3:0	R/W	1	reg_polar3_ofst1_1 : default = 1 //s4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < (f_2 - \text{delt} - \text{ofset2})$ , then polar3_m2++; if $f_1 > ((f_0 + \text{delt} + \text{ofset2})$ , then polar3_p2++;

Table 9-1242 NR2\_POLAR3\_PARA1 0x2d9b

Bit(s)	R/W	Default	Description
31:24	R/W	48	reg_polar3_rate3_0 : default = 48 //u8x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < (f_2 - \text{delt} - \text{ofset3})$ or $f_1 > ((f_0 + \text{delt} + \text{ofst3})$ , then polar3_32++;
23:16	R/W	48	reg_polar3_rate3_1 : default = 48 //u8x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < (f_2 - \text{delt} - \text{ofset3})$ or $f_1 > ((f_0 + \text{delt} + \text{ofst3})$ , then polar3_32++;
15:12	R/W	2	reg_polar3_ofst3_0 : default = 2 //s4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < (f_2 - \text{delt} - \text{ofst3})$ or $f_1 > ((f_0 + \text{delt} + \text{ofst3})$ , then polar3_32++;
11:8	R/W	2	reg_polar3_ofst3_1 : default = 2 //s4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < (f_2 - \text{delt} - \text{ofst3})$ or $f_1 > ((f_0 + \text{delt} + \text{ofst3})$ , then polar3_32++;
7:4	R/W	2	reg_polar3_ofst2_0 : default = 2 //s4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < (f_2 - \text{delt} - \text{ofset2})$ , then polar3_m2++; if $f_1 > ((f_0 + \text{delt} + \text{ofset2})$ , then polar3_p2++;
3:0	R/W	2	reg_polar3_ofst2_1 : default = 2 //s4x2: $\text{delt} = \text{rate} * \text{dif02} / 32$ , e.g. $f_2 < f_0$ , if $f_1 < (f_2 - \text{delt} - \text{ofset2})$ , then polar3_m2++; if $f_1 > ((f_0 + \text{delt} + \text{ofset2})$ , then polar3_p2++;

**Table 9-1243 NR2\_POLAR3\_CTRL 0x2d9c**

Bit(s)	R/W	Default	Description
16	R.O	0	reg_polar3_ro_reset : default = 0 //u1: reset signal of the polar3 read only registers
15:8	R/W	10	reg_polar3_h_mute : default = 10 //u8: horizontally pixels to mute for left right sides for polar3 detection;
7:0	R/W	10	reg_polar3_v_mute : default = 10 //u8: horizontally pixels to mute for top and bottom sides for polar3 detection;

**Table 9-1244 NR2\_RO\_POLAR3\_NUMOFPIX 0x2d9d**

Bit(s)	R/W	Default	Description
23:0	R.O	0	ro_polar3_numofpix : default = 0 //u24, number of pixels detected as polar3

**Table 9-1245 NR2\_RO\_POLAR3\_SMOOTHMV 0x2d9e**

Bit(s)	R/W	Default	Description
23:0	R.O	0	ro_polar3_smoothmv : default = 0 //u24, number of pixels with smooth mv, F(t) is close between avg of f(t-1) and f(t+1);

**Table 9-1246 NR2\_RO\_POLAR3\_M1 0x2d9f**

Bit(s)	R/W	Default	Description
23:0	R.O	0	ro_polar3_m1 : default = 0 //u24, number of pixels with F(t) is close to f(t-1) instead of f(t+1), but in between [f(t-1), f(t+1)];

**Table 9-1247 NR2\_RO\_POLAR3\_P1 0x2da0**

Bit(s)	R/W	Default	Description
23:0	R.O	0	ro_polar3_p1 : default = 0 //u24, number of pixels with F(t) is close to f(t+1) instead of f(t-1), but in between [f(t-1), f(t+1)];

**Table 9-1248 NR2\_RO\_POLAR3\_M2 0x2da1**

Bit(s)	R/W	Default	Description
23:0	R.O	0	ro_polar3_m2 : default = 0 //u24, number of pixels with F(t) is close to f(t-1) instead of f(t+1), but out side of (f(t-1), f(t+1));

**Table 9-1249 NR2\_RO\_POLAR3\_P2 0x2da2**

Bit(s)	R/W	Default	Description
23:0	R.O	0	ro_polar3_p2 : default = 0 //u24, number of pixels with F(t) is close to f(t+1) instead of f(t-1), but out side of (f(t-1), f(t+1));

Table 9-1250 NR2\_RO\_POLAR3\_32 0x2da3

Bit(s)	R/W	Default	Description
23:0	R/O	0	ro_polar3_32 : default = 0 //u24, number of pixels with F(t) far from [f(t-1),f(t+1)] and f(t-1) is close to f(t+1);

Table 9-1251 NR4\_DRT\_CTRL 0x2da4

Bit(s)	R/W	Default	Description
31:24	R/W	16	reg_nr4_ydrt_3line_ssd_gain : // unsigned , default = 16 gain to max ssd normalized 16 as '1'
23:16	R/W	16	reg_nr4_ydrt_5line_ssd_gain : // unsigned , default = 16 gain to max ssd normalized 16 as '1'
14:13	R/W	1	reg_nr4_drt_yhsad_mode : // unsigned , default = 1 mode for luma horizontal sad calc., 0: no vertical lpf, 1: vertical [1 2 1], 2 or 3: vertical [1 2 2 2 1] if 5 lines
12:11	R/W	1	reg_nr4_drt_chsad_mode : // unsigned , default = 1 mode for chroma horizontal sad calc., 0: no vertical lpf, 1: vertical [1 2 1], 2 or 3: vertical [1 2 2 2 1] if 5 lines
10	R/W	1	reg_nr4_drt_yhsad_hlpf : // unsigned , default = 1 hlpf for luma hsad of drt calculation, 0: no lpf, 1: with [1 2 1] hlpf
9	R/W	1	reg_nr4_drt_yvsad_hlpf : // unsigned , default = 1 hlpf for luma vsad of drt calculation, 0: no lpf, 1: with [1 2 1] hlpf
8	R/W	1	reg_nr4_drt_ydsad_hlpf : // unsigned , default = 1 hlpf for luma dsad of drt calculation, 0: no lpf, 1: with [1 2 1] hlpf
7	R/W	1	reg_nr4_drt_chsad_hlpf : // unsigned , default = 1 hlpf for chrome hsad of drt calculation, 0: no lpf, 1: with [1 2 1] hlpf
6	R/W	1	reg_nr4_drt_cvsad_hlpf : // unsigned , default = 1 hlpf for chroma vsad of drt calculation, 0: no lpf, 1: with [1 2 1] hlpf
5	R/W	1	reg_nr4_drt_cdsad_hlpf : // unsigned , default = 1 hlpf for chroma dsad of drt calculation, 0: no lpf, 1: with [1 2 1] hlpf
4	R/W	1	reg_nr4_ydrt_dif_mode : // unsigned , default = 1 0:y_dif, 1: y_dif + (u_dif + v_dif)/2
3: 2	R/W	2	reg_nr4_cdrdt_dif_mode : // unsigned , default = 2 0:(u_dif + v_dif), 1: y_dif/4 + (u_dif + v_dif)*3/4, 2:y_dif/2 + (u_dif + v_dif)/2, 3: y_dif (not recommended)
1:0	R/W		reserved

Table 9-1252 NR4\_DRT\_YSAD\_GAIN 0x2da5

Bit(s)	R/W	Default	Description
31:24	R/W	16	reg_nr4_ysad_hrz_gain : // unsigned , default = 16 gain for horizontal sad, 16 normalized to "1"
23:16	R/W	20	reg_nr4_ysad_diag_gain : // unsigned , default = 20 gain for diagonal sad, 16 normalized to "1"
15: 8	R/W	16	reg_nr4_ysad_vrt_gain : // unsigned , default = 16 gain for vertical sad, 16 normalized to "1"
5: 0	R/W	6	reg_nr4_drt_ysad_core_rate : // unsigned , default = 6 rate of coring for sad(theta) - sad(theta+pi/2)*rate/64



**Table 9-1253 NR4\_DRT\_CSAD\_GAIN 0x2da6**

Bit(s)	R/W	Default	Description
31:24	R/W	16	reg_nr4_csad_hrz_gain : // unsigned , default = 16 gain for horizontal sad, 16 normalized to "1"
23:16	R/W	20	reg_nr4_csad_diag_gain : // unsigned , default = 20 gain for diagonal sad, 16 normalized to "1"
15: 8	R/W	16	reg_nr4_csad_vrt_gain : // unsigned , default = 16 gain for vertical sad, 16 normalized to "1"
5: 0	R/W	6	reg_nr4_drt_csad_core_rate : // unsigned , default = 6 rate of coring for sad(theta) - sad(theta+pi/2)*rate/64

**Table 9-1254 NR4\_DRT\_SAD\_ALP\_CORE 0x2da7**

Bit(s)	R/W	Default	Description
23:20	R/W	0	reg_nr4_ydrt_alp_core_rate : // unsigned , default = 0 luma ratio to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err * 64; dft = 0/32
19:16	R/W	0	reg_nr4_cdrt_alp_core_rate : // unsigned , default = 0 chroma ratio to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err * 64; dft = 0/32
13: 8	R/W	10	reg_nr4_ydrt_alp_core_ofst : // unsigned , default = 10 luma offset to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err * 64; dft = 10
5: 0	R/W	10	reg_nr4_cdrt_alp_core_ofst : // unsigned , default = 10 chroma offset to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err * 64; dft = 10

**Table 9-1255 NR4\_DRT\_ALP\_MINMAX 0x2da8**

Bit(s)	R/W	Default	Description
29:24	R/W	0	reg_nr4_ydrt_alp_min : // unsigned , default = 0 luma min value of alpha, dft = 0
21:16	R/W	63	reg_nr4_ydrt_alp_max : // unsigned , default = 63 luma max value of alpha, dft = 63
13: 8	R/W	0	reg_nr4_cdrt_alp_min : // unsigned , default = 0 chroma min value of alpha, dft = 0
5: 0	R/W	63	reg_nr4_cdrt_alp_max : // unsigned , default = 63 chroma max value of alpha, dft = 63

**Table 9-1256 NR4\_SNR\_CTRL\_REG 0x2da9**

Bit(s)	R/W	Default	Description
12	R/W	1	reg_nr4_bet2_sel : // unsigned , default = 1
11: 9	R/W	0	reg_nr4_snr2_sel_mode : // unsigned , default = 0 0: no filter, 1: adpgau, adp_drt_lpf blend; 2: adpgau, drt4_lpf blend; 3: adp_drt_lpf method, 4: drt4_lpf method, 5: adp_drt_lpf //original image blend, 6: drt4_lpf, original image blend, 7: adpgau method; dft=1
8	R/W	1	reg_nr4_snr2_gaulpf_mode : // unsigned , default = 1 0: 3*5 or 5*5 gaussian lpf; 1: 3*3 (window size) gaussian lpf; dft=1
7: 6	R/W	3	reg_nr4_snr2_alpha0_sad_mode : // unsigned , default = 3 0: max_sad*max_ssd; 1: max_sad*max_sad; 2: adp_max_sad*max_ssd; 3: adp_max_sad*adp_max_sad dft=3

Bit(s)	R/W	Default	Description
5: 4	R/W	2	reg_nr4_snr2_alpha1_sad_mode : // unsigned , default = 2 0: max_sad; 1: cross_max_sad; 2 or 3: adp_sad dft=2
1: 0	R/W	3	reg_nr4_snr2_adp_drtlpf_mode : // unsigned , default = 3 0: adp_drtlpf [2 1 1]/4, 1: adp_drtlpf [4 2 1 1]/8; 2: adp_drtlpf [2 2 2 1 1]/8; 3: adp_drtlpf [7 7 7 6 5]/32; dft=3;

Table 9-1257 NR4\_SNR\_ALPHA0\_MAX\_MIN 0x2daa

Bit(s)	R/W	Default	Description
29:23	R/W	127	reg_nr4_snr2_alp0_ymin : // unsigned , default = 127 normalized to 128 as '1'
22:16	R/W	127	reg_nr4_snr2_alp0_ymax : // unsigned , default = 127 normalized to 128 as '1'
13: 7	R/W	127	reg_nr4_snr2_alp0_cmin : // unsigned , default = 127 normalized to 128 as '1'
6: 0	R/W	127	reg_nr4_snr2_alp0_cmax : // unsigned , default = 127 normalized to 128 as '1'

Table 9-1258 NR4\_ALP0C\_ERR2CURV\_LIMIT0 0x2dab

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_nr4_snr2_alp0_minerr_cpar0 : // unsigned , default = 0 threshold0 of curve to map mierr to alp0 for chroma channel, this will be set value of flat region mierr that no need blur.
23:16	R/W	25	reg_nr4_snr2_alp0_minerr_cpar1 : // unsigned , default = 25 threshold1 of curve to map mierr to alp0 for chroma channel, this will be set value of texture region mierr that can not blur.
15: 8	R/W	40	reg_nr4_snr2_alp0_minerr_cpar5 : // unsigned , default = 40 rate0 (for mierr<th0) of curve to map mierr to alp0 for chroma channel. the larger of the value, the deep of the slope. 0~255.
7: 0	R/W	40	reg_nr4_snr2_alp0_minerr_cpar6 : // unsigned , default = 40 rate1 (for mierr>th1) of curve to map mierr to alp0 for chroma channel. the larger of the value, the deep of the slope. 0~255.

Table 9-1259 NR4\_ALP0C\_ERR2CURV\_LIMIT1 0x2dac

Bit(s)	R/W	Default	Description
23:16	R/W	127	reg_nr4_snr2_alp0_minerr_cpar2 : // unsigned , default = 127 level limit(for mierr<th0) of curve to map mierr to alp0 for chroma channel, that we can do for flat region. 0~255.
15: 8	R/W	0	reg_nr4_snr2_alp0_minerr_cpar3 : // unsigned , default = 0 level limit(for th0<mierr<th1) of curve to map mierr to alp0 for chroma channel, that we can do for misc region. 0~255.
7: 0	R/W	127	reg_nr4_snr2_alp0_minerr_cpar4 : // unsigned , default = 127 level limit(for mierr>th1) of curve to map mierr to alp0 for chroma channel, that we can do for texture region. 0~255.

**Table 9-1260 NR4\_ALP0Y\_ERR2CURV\_LIMIT0 0x2dad**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_nr4_snr2_alp0_minerr_ypar0 : // unsigned , default = 0 threshold0 of curve to map mierr to alp0 for luma channel, this will be set value of flat region mierr that no need blur. 0~255.
23:16	R/W	25	reg_nr4_snr2_alp0_minerr_ypar1 : // unsigned , default = 25 threshold1 of curve to map mierr to alp0 for luma channel, this will be set value of texture region mierr that can not blur.
15: 8	R/W	40	reg_nr4_snr2_alp0_minerr_ypar5 : // unsigned , default = 40 rate0 (for mierr<th0) of curve to map mierr to alp0 for luma channel. the larger of the value, the deep of the slope. 0~255.
7: 0	R/W	40	reg_nr4_snr2_alp0_minerr_ypar6 : // unsigned , default = 40 rate1 (for mierr>th1) of curve to map mierr to alp0 for luma channel. the larger of the value, the deep of the slope. 0~255.

**Table 9-1261 NR4\_ALP0Y\_ERR2CURV\_LIMIT1 0x2dae**

Bit(s)	R/W	Default	Description
23:16	R/W	127	reg_nr4_snr2_alp0_minerr_ypar2 : // unsigned , default = 127 level limit(for mierr<th0) of curve to map mierr to alp0 for luma channel, set to alp0 that we can do for flat region. 0~255.
15: 8	R/W	0	reg_nr4_snr2_alp0_minerr_ypar3 : // unsigned , default = 0 level limit(for th0<mierr<th1) of curve to map mierr to alp0 for luma channel, alp0 that we can do for misc region. 0~255.
7: 0	R/W	127	reg_nr4_snr2_alp0_minerr_ypar4 : // unsigned , default = 127 level limit(for mierr>th1) of curve to map mierr to alp0 for luma channel, alp0 that we can do for texture region. 0~255.

**Table 9-1262 NR4\_SNR\_ALPA1\_RATE\_AND\_OFST 0x2daf**

Bit(s)	R/W	Default	Description
23:18	R/W	0	reg_nr4_snr2_alp1_ycore_rate : // unsigned , default = 0 normalized 64 as "1"
17:12	R/W	0	reg_nr4_snr2_alp1_ccore_rate : // unsigned , default = 0 normalized 64 as "1"
11: 6	R/W	3	reg_nr4_snr2_alp1_ycore_ofst : // signed , default = 3 normalized 64 as "1"
5: 0	R/W	3	reg_nr4_snr2_alp1_ccore_ofst : // signed , default = 3 normalized 64 as "1"

**Table 9-1263 NR4\_SNR\_ALPHA1\_MAX\_MIN 0x2db0**

Bit(s)	R/W	Default	Description
23:18	R/W	0	reg_nr4_snr2_alp1_ymin : // unsigned , default = 0 normalized to 64 as '1'
17:12	R/W	63	reg_nr4_snr2_alp1_ymax : // unsigned , default = 63 normalized to 64 as '1'
11: 6	R/W	0	reg_nr4_snr2_alp1_cmin : // unsigned , default = 0 normalized to 64 as '1'
5: 0	R/W	63	reg_nr4_snr2_alp1_cmax : // unsigned , default = 63 normalized to 64 as '1'

**Table 9-1264 NR4\_ALP1C\_ERR2CURV\_LIMIT0 0x2db1**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_nr4_snr2_alp1_minerr_cpar0 : // unsigned , default = 0 anel, this will be set value of flat region mierr that no need directional NR. 0~255.
23:16	R/W	24	reg_nr4_snr2_alp1_minerr_cpar1 : // unsigned , default = 24 hannel,this will be set value of texture region mierr that can not do directional NR. 0~255.
15: 8	R/W	0	reg_nr4_snr2_alp1_minerr_cpar5 : // unsigned , default = 0 a/chroma channel. the larger of the value, the deep of the slope.
7: 0	R/W	20	reg_nr4_snr2_alp1_minerr_cpar6 : // unsigned , default = 20 a/chroma channel. the larger of the value, the deep of the slope. 0~255

**Table 9-1265 NR4\_ALP1C\_ERR2CURV\_LIMIT1 0x2db2**

Bit(s)	R/W	Default	Description
23:16	R/W	0	reg_nr4_snr2_alp1_minerr_cpar2 : // unsigned , default = 0 will be set to alp1 that we can do for flat region. 0~255.
15: 8	R/W	16	reg_nr4_snr2_alp1_minerr_cpar3 : // unsigned , default = 16 this will be set to alp1 that we can do for misc region. 0~255.
7: 0	R/W	63	reg_nr4_snr2_alp1_minerr_cpar4 : // unsigned , default = 63 will be set to alp1 that we can do for texture region. 0~255.255 before

**Table 9-1266 NR4\_ALP1Y\_ERR2CURV\_LIMIT0 0x2db3**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_nr4_snr2_alp1_minerr_ypar0 : // unsigned , default = 0 thra/chroma channel, this will be set value of flat region mierr that no need directional NR. 0~255.
23:16	R/W	24	reg_nr4_snr2_alp1_minerr_ypar1 : // unsigned , default = 24 thra/chroma channel,this will be set value of texture region mierr that can not do directional NR. 0~255.
15: 8	R/W	0	reg_nr4_snr2_alp1_minerr_ypar5 : // unsigned , default = 0 ratlp1 for luma/chroma channel. the larger of the value, the deep of the slope.
7: 0	R/W	20	reg_nr4_snr2_alp1_minerr_ypar6 : // unsigned , default = 20 ratlp1 for luma/chroma channel. the larger of the value, the deep of the slope. 0~255

**Table 9-1267 NR4\_ALP1Y\_ERR2CURV\_LIMIT1 0x2db4**

Bit(s)	R/W	Default	Description
23:16	R/W	0	reg_nr4_snr2_alp1_minerr_ypar2 : // unsigned , default = 0 lev to alp1 for luma/chroma channel, this will be set to alp1 that we can do for flat region. 0~255.
15: 8	R/W	16	reg_nr4_snr2_alp1_minerr_ypar3 : // unsigned , default = 16 lev to alp1 for luma/chroma channel, this will be set to alp1 that we can do for misc region. 0~255.
7: 0	R/W	63	reg_nr4_snr2_alp1_minerr_ypar4 : // unsigned , default = 63 lev to alp1 for luma/chroma channel, this will be set to alp1 that we can do for texture region. 0~255.255 before

**Table 9-1268 NR4\_MTN\_CTRL 0x2db5**

Bit(s)	R/W	Default	Description
1	R/W	1	reg_nr4_mtn_ref_en : // unsigned , default = 1 enable motion refinement, dft = 1
0	R/W	0	reg_nr4_mtn_ref_bet_sel : // unsigned , default = 0 beta selection mode for motion refinement, 0: beta1, 1: beta2, dft = 0

**Table 9-1269 NR4\_MTN\_REF\_PAR0 0x2db6**

Bit(s)	R/W	Default	Description
31:24	R/W	24	reg_nr4_mtn_ref_par0 : // unsigned , default = 24 par0 for beta to gain, dft =
23:16	R/W	60	reg_nr4_mtn_ref_par1 : // unsigned , default = 60 par1 for beta to gain, dft =
15: 8	R/W	4	reg_nr4_mtn_ref_par2 : // unsigned , default = 4 par2 for beta to gain, dft =
7: 0	R/W	32	reg_nr4_mtn_ref_par3 : // unsigned , default = 32 par3 for beta to gain, dft =

**Table 9-1270 NR4\_MTN\_REF\_PAR1 0x2db7**

Bit(s)	R/W	Default	Description
23:16	R/W	128	reg_nr4_mtn_ref_par4 : // unsigned , default = 128 par4 for beta to gain, dft =
15: 8	R/W	40	reg_nr4_mtn_ref_par5 : // unsigned , default = 40 par5 for beta to gain, dft =
7: 0	R/W	20	reg_nr4_mtn_ref_par6 : // unsigned , default = 20 par6 for beta to gain, dft =

**Table 9-1271 NR4\_MCNR\_LUMA\_ENH\_CTRL 0x2db8**

Bit(s)	R/W	Default	Description
3	R/W	1	reg_nr4_luma_plus_en : // unsigned , default = 1 enable luma enhancement, dft = 1
2	R/W	1	reg_nr4_luma_plus_wt_mode : // unsigned , default = 1 luma weight calc mode, 0:sqrt(1+x^2), 1: 1+abs(x), dft = 0
1: 0	R/W	1	reg_nr4_luma_plus_orient_mode : // unsigned , default = 1 0: only use previous orient for pre and cur luma plus, 1: 0: only use current orient for pre and cur luma plus

**Table 9-1272 NR4\_MCNR\_LUMA\_STAT\_LIMTX 0x2db9**

Bit(s)	R/W	Default	Description
29:16	R/W	8	reg_nr4_luma_plus_xst : // unsigned , default = 8 start for luma plus statistic, dft = 8
13: 0	R/W	711	reg_nr4_luma_plus_xed : // unsigned , default = 711 end for luma plus statistic, dft = HSIZE-8-1;

**Table 9-1273 NR4\_MCNR\_LUMA\_STAT\_LIMTY 0x2dba**

Bit(s)	R/W	Default	Description
29:16	R/W	8	reg_nr4_luma_plus_yst : // unsigned , default = 8 start for luma plus statistic, dft = 8
13: 0	R/W	231	reg_nr4_luma_plus_yed : // unsigned , default = 231 end for luma plus statistic, dft = VSIZE-8-1

**Table 9-1274 NR4\_MCNR\_LUMA\_DIF\_CALC 0x2dbb**

Bit(s)	R/W	Default	Description
29:24	R/W	8	reg_nr4_luma_plus_ugain : // unsigned , default = 8 U's gain for luma enhancement, 16 normalized as '1'
21:16	R/W	8	reg_nr4_luma_plus_vgain : // unsigned , default = 8 V's gain for luma enhancement, 16 normalized as '1'
15: 8	R/W	2	reg_nr4_luma_plus_ycor_thd : // unsigned , default = 2 Y coring threshold for difference calc., dft = 0
7: 0	R/W	0	reg_nr4_luma_plus_ccor_thd : // unsigned , default = 0 C coring threshold for difference calc., dft = 0

**Table 9-1275 NR4\_MCNR\_LUMAPRE\_CAL\_PRAM 0x2dbc**

Bit(s)	R/W	Default	Description
25:24	R/W	0	reg_nr4_pre_u_orient : // signed , default = 0 orientation of previous U, initial to 0, and will be updated by software
17:16	R/W	0	reg_nr4_pre_v_orient : // signed , default = 0 orientation of previous V, initial to 0, and will be updated by software
15: 8	R/W	0	reg_nr4_pre_u_mean : // unsigned , default = 0 mean of previous U, initial to 0, and will be updated by software
7: 0	R/W	0	reg_nr4_pre_v_mean : // unsigned , default = 0 mean of previousV, initial to 0, and will be updated by software

**Table 9-1276 NR4\_MCNR\_LUMACUR\_CAL\_PRAM 0x2dbd**

Bit(s)	R/W	Default	Description
25:24	R/W	0	reg_nr4_cur_u_orient : // signed , default = 0 orientation of current U, initial to 0, and will be updated by software
17:16	R/W	0	reg_nr4_cur_v_orient : // signed , default = 0 orientation of current V, initial to 0, and will be updated by software
15: 8	R/W	0	reg_nr4_cur_u_mean : // unsigned , default = 0 mean of current U, initial to 0, and will be updated by software
7: 0	R/W	0	reg_nr4_cur_v_mean : // unsigned , default = 0 mean of current, initial to 0, and will be updated by software

**Table 9-1277 NR4\_MCNR\_MV\_CTRL\_REG 0x2dbe**

Bit(s)	R/W	Default	Description
13:12	R/W	2	reg_nr4_sad_bitw : // unsigned , default = 2 sad bit width (8 + x) before clip to u8, dft = 1
11: 4	R/W	64	reg_nr4_glb_gain : // unsigned , default = 64 global gain calc. by software, 64 is normalized as '1'
3: 0	R/W	8	reg_nr4_mv_err_rsft : // unsigned , default = 8 right shift for mv err calc., dft = 9

**Table 9-1278 NR4\_MCNR\_MV\_GAIN0 0x2dbf**

Bit(s)	R/W	Default	Description
31:28	R/W	1	reg_nr4_lftmvx_gain : // unsigned , default = 1 left mvx gain for err calc., dft = 1
27:24	R/W	1	reg_nr4_lftmvy_gain : // unsigned , default = 1 left mvy gain for err calc., dft = 1
23:20	R/W	5	reg_nr4_zmvx_gain : // unsigned , default = 5 zero mvx gain for err calc., dft = 2
19:16	R/W	5	reg_nr4_zmvy_gain : // unsigned , default = 5 zero mvy gain for err calc., dft = 4
15:12	R/W	2	reg_nr4_lmvy0_gain : // unsigned , default = 2 line mvx0 gain for err calc., dft = 1
11: 8	R/W	2	reg_nr4_lmvy1_gain : // unsigned , default = 2 line mvx1 gain for err calc., dft = 1
7: 4	R/W	2	reg_nr4_lmvy0_gain : // unsigned , default = 2 line mvy0 gain for err calc., dft = 1
3: 0	R/W	2	reg_nr4_lmvy1_gain : // unsigned , default = 2 line mvy1 gain for err calc., dft = 1

**Table 9-1279 NR4\_MCNR\_LMV\_PARM 0x2dc0**

Bit(s)	R/W	Default	Description
31:28	R/W	3	reg_nr4_lmvy_rt0 : // unsigned , default = 3 ratio of max lmv
27:24	R/W	3	reg_nr4_lmvy_rt1 : // unsigned , default = 3 ratio of second max lmv
21:16	R/W	16	reg_nr4_lmvy_num_lmt0 : // unsigned , default = 16 lmv0 least/limit number of (total number - zero_bin)
13: 8	R/W	8	reg_nr4_lmvy_num_lmt1 : // unsigned , default = 8 lmv1 least/limit number of (total number - zero_bin - max0)
1: 0	R/W	1	reg_nr4_max_sad_rng : // unsigned , default = 1 search range of max2 sad in small region, dft = 1

**Table 9-1280 NR4\_MCNR\_ALP0\_REG 0x2dc1**

Bit(s)	R/W	Default	Description
25	R/W	1	reg_nr4_alp0_fail_chk : // unsigned , default = 1 enable check for alp0 fail status
24	R/W	1	reg_nr4_bet0_coef_ref_en : // unsigned , default = 1 bet1 refinement by coef_blt
23:16	R/W	255	reg_nr4_alp0_posad_gain : // unsigned , default = 255 the sad (norm) gain for pixel pointed by MV;
9: 8	R/W	0	reg_nr4_alp0_norm_mode : // unsigned , default = 0 alp0 select sad norm mode, 0: disable, 1: enable dc norm, 2: enable ac norm, 3: enable both (dc/ac) norm, dft = 3
5: 0	R/W	16	reg_nr4_alp0_norm_gain : // unsigned , default = 16 alp0 gain for sad norm, '32' as '1', dft = 1

Table 9-1281 NR4\_MCNR\_ALP1\_AND\_BET0\_REG 0x2dc2

Bit(s)	R/W	Default	Description
25:24	R/W	3	reg_nr4_alp1_norm_mode : // unsigned , default = 3 alp1 select sad norm mode, 0: disable, 1: enable dc norm, 2: enable ac norm, 3: enable both (dc/ac) norm, dft = 3
21:16	R/W	3	reg_nr4_alp1_norm_gain : // unsigned , default = 3 alp1 gain for sad norm, '32' as '1', dft = 1
9: 8	R/W	3	reg_nr4_bet0_norm_mode : // unsigned , default = 3 bet0 select sad norm mode, 0: disable, 1: enable dc norm, 2: enable ac norm, 3: enable both (dc/ac) norm, dft = 3
5: 0	R/W	8	reg_nr4_bet0_norm_gain : // unsigned , default = 8 bet0 gain for sad norm, '32' as '1', dft = 1

Table 9-1282 NR4\_MCNR\_BET1\_AND\_BET2\_REG 0x2dc3

Bit(s)	R/W	Default	Description
25:24	R/W	3	reg_nr4_bet1_norm_mode : // unsigned , default = 3 bet1 select sad norm mode, 0: disable, 1: enable dc norm, 2: enable ac norm, 3: enable both (dc/ac) norm, dft = 3
21:16	R/W	8	reg_nr4_bet1_norm_gain : // unsigned , default = 8 bet1 gain for sad norm, '32' as '1', dft = 1
9: 8	R/W	0	reg_nr4_bet2_norm_mode : // unsigned , default = 0 bet2 select sad norm mode, 0: disable, 1: enable dc norm, 2: enable ac norm, 3: enable both (dc/ac) norm, dft = 3
5: 0	R/W	16	reg_nr4_bet2_norm_gain : // unsigned , default = 16 bet2 gain for sad norm, '32' as '1', dft = 1

Table 9-1283 NR4\_MCNR\_AC\_DC\_CTRL 0x2dc4

Bit(s)	R/W	Default	Description
11	R/W	1	reg_nr4_dc_mode : // unsigned , default = 1 mode for dc selection, 0: Y_lpf, 1: Y_lpf + (U_lpf+V_lpf)/2,
10	R/W	1	reg_nr4_ac_mode : // unsigned , default = 1 mode for ac selection, 0: Y_abs_dif, 1: Y_abs_dif + (U_abs_dif + V_abs_dif)/2
9	R/W	0	reg_nr4_dc_sel : // unsigned , default = 0 selection mode for dc value, 0: 3x5, 1: 5x5, dft = 1
8	R/W	0	reg_nr4_ac_sel : // unsigned , default = 0 selection mode for ac value, 0: 3x5, 1: 5x5, dft = 1
6: 4	R/W	2	reg_nr4_dc_shft : // unsigned , default = 2 right shift for dc value, dft = 2
2: 0	R/W	0	reg_nr4_ac_shft : // unsigned , default = 0 right shift for ac value, dft = 2



Table 9-1284 NR4\_MCNR\_CM\_CTRL0 0x2dc5

Bit(s)	R/W	Default	Description
28	R/W	0	reg_nr4_cm_skin_prc_bet0 : // unsigned , default = 0 enable skin tone processing for mcnr bet0 calc., dft = 1
27:26	R/W	1	reg_nr4_cm_chrm_sel : // unsigned , default = 1 chrome selection for color match, 0: 1x1, 1: 3X3LPF, 2: 3x5LPF, 3: 5x5LPF for 5lines, 3x5LPF for 3lines, dft = 3
25:24	R/W	1	reg_nr4_cm_luma_sel : // unsigned , default = 1 luma selection for color match, 0: 1x1, 1: 3X3LPF, 2: 3x5LPF, 3: 5x5LPF for 5lines, 3x5LPF for 3lines, dft = 3
23:21	R/W	3	reg_nr4_cm_skin_rshft_bet0 : // unsigned , default = 3 right shift for bet0's skin color gains, dft = 3
20	R/W	1	reg_nr4_cm_var_sel : // unsigned , default = 1 variation selection for color match, 0: 3x5, 1: 5x5 for 5lines, 3x5 for 3lines, dft = 1
19	R/W	1	reg_nr4_cm_green_prc_bet0 : // unsigned , default = 1 enable green processing for mcnr bet0 calc., dft = 1
18:16	R/W	4	reg_nr4_cm_green_rshft_bet0 : // unsigned , default = 4 right shift for bet0's green color gains, dft = 4
15:14	R/W	2	reg_nr4_prefilt_mod : // unsigned , default = 2 pre filter mode in mcnr, 0: mv pointed pixel, 1: bilater filter
13:12	R/W	1	reg_nr4_alp1_mode : // unsigned , default = 1 mode for alpha1's sad selection, 0: max sad, 1: three min sads, 2: min sad, 3: co sad
9: 8	R/W	0	reg_nr4_bet0_mode : // unsigned , default = 0 mode for bet0's sad selection, 0: max sad, 1: three min sads, 2: min sad, 3: co sad, else: (co sad) - (min sad)
5: 4	R/W	2	reg_nr4_bet1_mode : // unsigned , default = 2 mode for bet1's sad selection, 0: max sad, 1: three min sads, 2: min sad, 3: co sad, else: (co sad) - (min sad)
1: 0	R/W	1	reg_nr4_bet2_mode : // unsigned , default = 1 mode for bet2's sad selection, 0: max sad, 1: three min sads, 2: min sad, 3: co sad, else: (co sad) - (min sad)

Table 9-1285 NR4\_MCNR\_CM\_PRAM 0x2dc6

Bit(s)	R/W	Default	Description
29	R/W	1	reg_nr4_cm_blue_prc_alp0 : // unsigned , default = 1 enable blue processing for mcnr alpha0 calc., dft = 1
28	R/W	1	reg_nr4_cm_blue_prc_alp1 : // unsigned , default = 1 enable blue processing for mcnr alpha1 calc., dft = 1
27	R/W	1	reg_nr4_cm_skin_prc_alp0 : // unsigned , default = 1 enable skin tone processing for mcnr alpha0 calc., dft = 1
26	R/W	1	reg_nr4_cm_green_prc_alp0 : // unsigned , default = 1 enable green processing for mcnr alpha0 clac., dft = 1
25	R/W	1	reg_nr4_cm_skin_prc_alp1 : // unsigned , default = 1 enable skin tone processing for mcnr alpha0 calc., dft = 1
24	R/W	1	reg_nr4_cm_green_prc_alp1 : // unsigned , default = 1 enable green processing for mcnr alpha1 clac., dft = 1
23:20	R/W	13	reg_nr4_cm_blue_hue_st : // unsigned , default = 13 hue start of blue, dft =
19:16	R/W	15	reg_nr4_cm_blue_hue_ed : // unsigned , default = 15 hue end of blue, dft =
15:12	R/W	7	reg_nr4_cm_green_hue_st : // unsigned , default = 7 hue start of green, dft =
11: 8	R/W	10	reg_nr4_cm_green_hue_ed : // unsigned , default = 10 hue end of green, dft =

Bit(s)	R/W	Default	Description
7: 4	R/W	5	reg_nr4_cm_skin_hue_st : // unsigned , default = 5 hue start of skin, dft =
3: 0	R/W	6	reg_nr4_cm_skin_hue_ed : // unsigned , default = 6 hue end of skin, dft =

Table 9-1286 NR4\_MCNR\_CM\_RSHFT\_ALP0 0x2dc7

Bit(s)	R/W	Default	Description
27:25	R/W	5	reg_nr4_cm_blue_rshft_bet0 : // unsigned , default = 5 right shift for bet0's blue color gains, dft = 5
24	R/W	1	reg_nr4_cm_blue_prc_bet0 : // unsigned , default = 1 enable blue processing for mcnr bet0 calc., dft = 1
22:20	R/W	5	reg_nr4_cm_blue_rshft_alp0 : // unsigned , default = 5 right shift for alpha0/1's blue color gains, dft = 5
18:16	R/W	5	reg_nr4_cm_blue_rshft_alp1 : // unsigned , default = 5 right shift for alpha0/1's blue color gains, dft = 5
14:12	R/W	4	reg_nr4_cm_green_rshft_alp0 : // unsigned , default = 4 right shift for alpha0/1's green color gains, dft = 4
10: 8	R/W	4	reg_nr4_cm_green_rshft_alp1 : // unsigned , default = 4 right shift for alpha0/1's green color gains, dft = 4
6: 4	R/W	3	reg_nr4_cm_skin_rshft_alp0 : // unsigned , default = 3 right shift for alpha0/1's skin color gains, dft = 3
2: 0	R/W	3	reg_nr4_cm_skin_rshft_alp1 : // unsigned , default = 3 right shift for alpha0/1's skin color gains, dft = 3

Table 9-1287 NR4\_MCNR\_BLUE\_CENT 0x2dc8

Bit(s)	R/W	Default	Description
23:16	R/W	157	reg_nr4_cm_blue_centx : // unsigned , default = 157 x coordinate of center of blue, dft =
7: 0	R/W	110	reg_nr4_cm_blue_centy : // unsigned , default = 110 y coordinate of center of blue, dft =

Table 9-1288 NR4\_MCNR\_BLUE\_GAIN\_PAR0 0x2dc9

Bit(s)	R/W	Default	Description
31:24	R/W	32	reg_nr4_cm_blue_gain_par0 : // unsigned , default = 32 par0 for blue gain, dft =
23:16	R/W	255	reg_nr4_cm_blue_gain_par1 : // unsigned , default = 255 par1 for blue gain, dft =
15: 8	R/W	4	reg_nr4_cm_blue_gain_par2 : // unsigned , default = 4 par2 for blue gain, dft =
7: 0	R/W	32	reg_nr4_cm_blue_gain_par3 : // unsigned , default = 32 par3 for blue gain, dft =

**Table 9-1289 NR4\_MCNR\_BLUE\_GAIN\_PAR1 0x2dca**

Bit(s)	R/W	Default	Description
23:16	R/W	32	reg_nr4_cm_blue_gain_par4 : // unsigned , default = 32 par4 for blue gain, dft =
15: 8	R/W	32	reg_nr4_cm_blue_gain_par5 : // unsigned , default = 32 par5 for blue gain, dft =
7: 0	R/W	0	reg_nr4_cm_blue_gain_par6 : // unsigned , default = 0 par6 for blue gain, dft =

**Table 9-1290 NR4\_MCNR\_CM\_BLUE\_CLIP0 0x2dcb**

Bit(s)	R/W	Default	Description
23:16	R/W	40	reg_nr4_cm_blue_luma_min : // unsigned , default = 40 luma min for blue color matching, dft =
7: 0	R/W	180	reg_nr4_cm_blue_luma_max : // unsigned , default = 180 luma max for blue color matching, dft =

**Table 9-1291 NR4\_MCNR\_CM\_BLUE\_CLIP1 0x2dcc**

Bit(s)	R/W	Default	Description
31:24	R/W	5	reg_nr4_cm_blue_sat_min : // unsigned , default = 5 saturation min for blue color matching, dft =
23:16	R/W	255	reg_nr4_cm_blue_sat_max : // unsigned , default = 255 saturation max for blue color matching, dft =
15: 8	R/W	0	reg_nr4_cm_blue_var_min : // unsigned , default = 0 variation min for blue color matching, dft =
7: 0	R/W	12	reg_nr4_cm_blue_var_max : // unsigned , default = 12 variation max for blue color matching, dft =

**Table 9-1292 NR4\_MCNR\_GREEN\_CENT 0x2dcd**

Bit(s)	R/W	Default	Description
23:16	R/W	114	reg_nr4_cm_green_centr : // unsigned , default = 114 x coordinate of center of green, dft =
7: 0	R/W	126	reg_nr4_cm_green_centv : // unsigned , default = 126 y coordinate of center of green, dft =

**Table 9-1293 NR4\_MCNR\_GREEN\_GAIN\_PAR0 0x2dce**

Bit(s)	R/W	Default	Description
31:24	R/W	16	reg_nr4_cm_green_gain_par0 : // unsigned , default = 16 par0 for green gain, dft =
23:16	R/W	255	reg_nr4_cm_green_gain_par1 : // unsigned , default = 255 par1 for green gain, dft =
15: 8	R/W	255	reg_nr4_cm_green_gain_par2 : // unsigned , default = 255 par2 for green gain, dft =
7: 0	R/W	16	reg_nr4_cm_green_gain_par3 : // unsigned , default = 16 par3 for green gain, dft =

**Table 9-1294 NR4\_MCNR\_GREEN\_GAIN\_PAR1 0x2dcf**

Bit(s)	R/W	Default	Description
23:16	R/W	16	reg_nr4_cm_green_gain_par4 : // unsigned , default = 16 par4 for green gain, dft =
15: 8	R/W	128	reg_nr4_cm_green_gain_par5 : // unsigned , default = 128 par5 for green gain, dft =
7: 0	R/W	0	reg_nr4_cm_green_gain_par6 : // unsigned , default = 0 par6 for green gain, dft =

**Table 9-1295 NR4\_MCNR\_GREEN\_CLIP0 0x2dd0**

Bit(s)	R/W	Default	Description
23:16	R/W	40	reg_nr4_cm_green_luma_min : // unsigned , default = 40 luma min for green color matching, dft =
7: 0	R/W	160	reg_nr4_cm_green_luma_max : // unsigned , default = 160 luma max for green color matching, dft =

**Table 9-1296 NR4\_MCNR\_GREEN\_CLIP2 0x2dd1**

Bit(s)	R/W	Default	Description
31:24	R/W	4	reg_nr4_cm_green_sat_min : // unsigned , default = 4 saturation min for green color matching, dft =
23:16	R/W	255	reg_nr4_cm_green_sat_max : // unsigned , default = 255 saturation max for green color matching, dft =
15: 8	R/W	0	reg_nr4_cm_green_var_min : // unsigned , default = 0 variation min for green color matching, dft =
7: 0	R/W	12	reg_nr4_cm_green_var_max : // unsigned , default = 12 variation max for green color matching, dft =

**Table 9-1297 NR4\_MCNR\_SKIN\_CENT 0x2dd2**

Bit(s)	R/W	Default	Description
23:16	R/W	112	reg_nr4_cm_skin_centx : // unsigned , default = 112 x coordinate of center of skin tone, dft =
7: 0	R/W	149	reg_nr4_cm_skin_centy : // unsigned , default = 149 y coordinate of center of skin tone, dft =

**Table 9-1298 NR4\_MCNR\_SKIN\_GAIN\_PAR0 0x2dd3**

Bit(s)	R/W	Default	Description
31:24	R/W	20	reg_nr4_cm_skin_gain_par0 : // unsigned , default = 20 par0 for skin gain, dft =
23:16	R/W	255	reg_nr4_cm_skin_gain_par1 : // unsigned , default = 255 par1 for skin gain, dft =
15: 8	R/W	255	reg_nr4_cm_skin_gain_par2 : // unsigned , default = 255 par2 for skin gain, dft =
7: 0	R/W	8	reg_nr4_cm_skin_gain_par3 : // unsigned , default = 8 par3 for skin gain, dft =

**Table 9-1299 NR4\_MCNR\_SKIN\_GAIN\_PAR1 0x2dd4**

Bit(s)	R/W	Default	Description
23:16	R/W	8	reg_nr4_cm_skin_gain_par4 : // unsigned , default = 8 par4 for skin gain, dft =
15: 8	R/W	128	reg_nr4_cm_skin_gain_par5 : // unsigned , default = 128 par5 for skin gain, dft =
7: 0	R/W	0	reg_nr4_cm_skin_gain_par6 : // unsigned , default = 0 par6 for skin gain, dft =

**Table 9-1300 NR4\_MCNR\_SKIN\_CLIP0 0x2dd5**

Bit(s)	R/W	Default	Description
23:16	R/W	40	reg_nr4_cm_skin_luma_min : // unsigned , default = 40 luma min for skin color matching, dft =
7: 0	R/W	180	reg_nr4_cm_skin_luma_max : // unsigned , default = 180 luma max for skin color matching, dft =

**Table 9-1301 NR4\_MCNR\_SKIN\_CLIP1 0x2dd6**

Bit(s)	R/W	Default	Description
31:24	R/W	5	reg_nr4_cm_skin_sat_min : // unsigned , default = 5 saturation min for skin color matching, dft =
23:16	R/W	255	reg_nr4_cm_skin_sat_max : // unsigned , default = 255 saturation max for skin color matching, dft =
15: 8	R/W	0	reg_nr4_cm_skin_var_min : // unsigned , default = 0 variation min for skin color matching, dft =
7: 0	R/W	12	reg_nr4_cm_skin_var_max : // unsigned , default = 12 variation max for skin color matching, dft =

**Table 9-1302 NR4\_MCNR\_ALP1\_GLB\_CTRL 0x2dd7**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_nr4_alp1_glb_gain_en : // unsigned , default = 0 alp1 adjust by global gain, dft = 1
30:28	R/W	6	reg_nr4_alp1_glb_gain_lsft : // unsigned , default = 6 alp1 left shift before combine with global gain
27	R/W	1	reg_nr4_bet0_glb_gain_en : // unsigned , default = 1 bet0 adjust by global gain, dft = 1
26:24	R/W	6	reg_nr4_bet0_glb_gain_lsft : // unsigned , default = 6 bet1 left shift before combine with global gain
23	R/W	0	reg_nr4_bet1_glb_gain_en : // unsigned , default = 0 bet1 adjust by global gain, dft = 0
22:20	R/W	6	reg_nr4_bet1_glb_gain_lsft : // unsigned , default = 6 bet1 left shift before combine with global gain
19	R/W	1	reg_nr4_bet2_glb_gain_en : // unsigned , default = 1 bet2 adjust by global gain, dft = 1
18:16	R/W	6	reg_nr4_bet2_glb_gain_lsft : // unsigned , default = 6 bet2 left shift before combine with global gain
15	R/W	1	reg_nr4_alp1_ac_en : // unsigned , default = 1 alp1 adjust by ac, dft = 1

Bit(s)	R/W	Default	Description
14:12	R/W	5	reg_nr4_alp1_ac_lsft : // unsigned , default = 5 alp1 left shift before combine with ac
11	R/W	0	reg_nr4_bet0_ac_en : // unsigned , default = 0 bet0 adjust by ac, dft = 1
10: 8	R/W	5	reg_nr4_bet0_ac_lsft : // unsigned , default = 5 bet0 left shift before combine with ac
7	R/W	0	reg_nr4_bet1_ac_en : // unsigned , default = 0 bet1 adjust by ac, dft = 1
6: 4	R/W	5	reg_nr4_bet1_ac_lsft : // unsigned , default = 5 bet1 left shift before combine with ac
3	R/W	0	reg_nr4_bet2_ac_en : // unsigned , default = 0 bet2 adjust by ac, dft = 1
2: 0	R/W	5	reg_nr4_bet2_ac_lsft : // unsigned , default = 5 bet2 left shift before combine with ac

Table 9-1303 NR4\_MCNR\_DC2NORM\_LUT0 0x2dd8

Bit(s)	R/W	Default	Description
28:24	R/W	16	reg_nr4_dc2norm_lut0 : // unsigned , default = 16 normal 0~16, dc to norm for alpha adjust, dft =
20:16	R/W	16	reg_nr4_dc2norm_lut1 : // unsigned , default = 16 normal 0~16, dc to norm for alpha adjust, dft =
12: 8	R/W	16	reg_nr4_dc2norm_lut2 : // unsigned , default = 16 normal 0~16, dc to norm for alpha adjust, dft =
4: 0	R/W	16	reg_nr4_dc2norm_lut3 : // unsigned , default = 16 normal 0~16, dc to norm for alpha adjust, dft =

Table 9-1304 NR4\_MCNR\_DC2NORM\_LUT1 0x2dd9

Bit(s)	R/W	Default	Description
28:24	R/W	16	reg_nr4_dc2norm_lut4 : // unsigned , default = 16 normal 0~16, dc to norm for alpha adjust, dft =
20:16	R/W	16	reg_nr4_dc2norm_lut5 : // unsigned , default = 16 normal 0~16, dc to norm for alpha adjust, dft =
12: 8	R/W	16	reg_nr4_dc2norm_lut6 : // unsigned , default = 16 normal 0~16, dc to norm for alpha adjust, dft =
4: 0	R/W	12	reg_nr4_dc2norm_lut7 : // unsigned , default = 12 normal 0~16, dc to norm for alpha adjust, dft =

Table 9-1305 NR4\_MCNR\_DC2NORM\_LUT2 0x2dda

Bit(s)	R/W	Default	Description
4: 0	R/W	8	reg_nr4_dc2norm_lut8 : // unsigned , default = 8 normal 0~16, dc to norm for alpha adjust, dft =

**Table 9-1306 NR4\_MCNR\_AC2NORM\_LUT0 0x2ddb**

Bit(s)	R/W	Default	Description
28:24	R/W	2	reg_nr4_ac2norm_lut0 : // unsigned , default = 2 normal 0~16, ac to norm for alpha adjust, dft =
20:16	R/W	16	reg_nr4_ac2norm_lut1 : // unsigned , default = 16 normal 0~16, ac to norm for alpha adjust, dft =
12: 8	R/W	16	reg_nr4_ac2norm_lut2 : // unsigned , default = 16 normal 0~16, ac to norm for alpha adjust, dft =
4: 0	R/W	12	reg_nr4_ac2norm_lut3 : // unsigned , default = 12 normal 0~16, ac to norm for alpha adjust, dft =

**Table 9-1307 NR4\_MCNR\_AC2NORM\_LUT1 0x2ddc**

Bit(s)	R/W	Default	Description
28:24	R/W	4	reg_nr4_ac2norm_lut4 : // unsigned , default = 4 normal 0~16, ac to norm for alpha adjust, dft =
20:16	R/W	2	reg_nr4_ac2norm_lut5 : // unsigned , default = 2 normal 0~16, ac to norm for alpha adjust, dft =
12: 8	R/W	1	reg_nr4_ac2norm_lut6 : // unsigned , default = 1 normal 0~16, ac to norm for alpha adjust, dft =
4: 0	R/W	1	reg_nr4_ac2norm_lut7 : // unsigned , default = 1 normal 0~16, ac to norm for alpha adjust, dft =

**Table 9-1308 NR4\_MCNR\_AC2NORM\_LUT2 0x2ddd**

Bit(s)	R/W	Default	Description
4: 0	R/W	1	reg_nr4_ac2norm_lut8 : // unsigned , default = 1 normal 0~16, ac to norm for alpha adjust, dft =

**Table 9-1309 NR4\_MCNR\_SAD2ALP0\_LUT0 0x2dde**

Bit(s)	R/W	Default	Description
31:24	R/W	255	reg_nr4_sad2alp0_lut0 : // unsigned , default = 255 sad to alpha0 for temporal pixel value, dft = 255
23:16	R/W	252	reg_nr4_sad2alp0_lut1 : // unsigned , default = 252 sad to alpha0 for temporal pixel value, dft = 252
15: 8	R/W	249	reg_nr4_sad2alp0_lut2 : // unsigned , default = 249 sad to alpha0 for temporal pixel value, dft = 249
7: 0	R/W	235	reg_nr4_sad2alp0_lut3 : // unsigned , default = 235 sad to alpha0 for temporal pixel value, dft = 70

**Table 9-1310 NR4\_MCNR\_SAD2ALP0\_LUT1 0x2ddf**

Bit(s)	R/W	Default	Description
31:24	R/W	185	reg_nr4_sad2alp0_lut4 : // unsigned , default = 185 sad to alpha0 for temporal pixel value, dft = 12
23:16	R/W	70	reg_nr4_sad2alp0_lut5 : // unsigned , default = 70 sad to alpha0 for temporal pixel value, dft = 1
15: 8	R/W	14	reg_nr4_sad2alp0_lut6 : // unsigned , default = 14 sad to alpha0 for temporal pixel value, dft = 0
7: 0	R/W	1	reg_nr4_sad2alp0_lut7 : // unsigned , default = 1 sad to alpha0 for temporal pixel value, dft = 0

**Table 9-1311 NR4\_MCNR\_SAD2ALP0\_LUT2 0x2de0**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_nr4_sad2alp0_lut8 : // unsigned , default = 0 sad to alpha0 for temporal pixel value, dft = 0
23:16	R/W	0	reg_nr4_sad2alp0_lut9 : // unsigned , default = 0 sad to alpha0 for temporal pixel value, dft = 0
15: 8	R/W	0	reg_nr4_sad2alp0_lut10 : // unsigned , default = 0 sad to alpha0 for temporal pixel value, dft = 0
7: 0	R/W	0	reg_nr4_sad2alp0_lut11 : // unsigned , default = 0 sad to alpha0 for temporal pixel value, dft = 0

**Table 9-1312 NR4\_MCNR\_SAD2ALP0\_LUT3 0x2de1**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_nr4_sad2alp0_lut12 : // unsigned , default = 0 sad to alpha0 for temporal pixel value, dft = 0
23:16	R/W	0	reg_nr4_sad2alp0_lut13 : // unsigned , default = 0 sad to alpha0 for temporal pixel value, dft = 0
15: 8	R/W	0	reg_nr4_sad2alp0_lut14 : // unsigned , default = 0 sad to alpha0 for temporal pixel value, dft = 0
7: 0	R/W	0	reg_nr4_sad2alp0_lut15 : // unsigned , default = 0 sad to alpha0 for temporal pixel value, dft = 0

**Table 9-1313 NR4\_MCNR\_SAD2ALP1\_LUT0 0x2de2**

Bit(s)	R/W	Default	Description
31:24	R/W	192	reg_nr4_sad2alp1_lut0 : // unsigned , default = 192 sad to alpha1 for temporal blending, dft = 128
23:16	R/W	160	reg_nr4_sad2alp1_lut1 : // unsigned , default = 160 sad to alpha1 for temporal blending, dft = 128
15: 8	R/W	128	reg_nr4_sad2alp1_lut2 : // unsigned , default = 128 sad to alpha1 for temporal blending, dft = 128
7: 0	R/W	96	reg_nr4_sad2alp1_lut3 : // unsigned , default = 96 sad to alpha1 for temporal blending, dft = 64



**Table 9-1314 NR4\_MCNR\_SAD2ALP1\_LUT1 0x2de3**

Bit(s)	R/W	Default	Description
31:24	R/W	64	reg_nr4_sad2alp1_lut4 : // unsigned , default = 64 sad to alpha1 for temporal blending, dft = 64
23:16	R/W	32	reg_nr4_sad2alp1_lut5 : // unsigned , default = 32 sad to alpha1 for temporal blending, dft = 128
15: 8	R/W	16	reg_nr4_sad2alp1_lut6 : // unsigned , default = 16 sad to alpha1 for temporal blending, dft = 255
7: 0	R/W	8	reg_nr4_sad2alp1_lut7 : // unsigned , default = 8 sad to alpha1 for temporal blending, dft = 255

**Table 9-1315 NR4\_MCNR\_SAD2ALP1\_LUT2 0x2de4**

Bit(s)	R/W	Default	Description
31:24	R/W	4	reg_nr4_sad2alp1_lut8 : // unsigned , default = 4 sad to alpha1 for temporal blending, dft = 255
23:16	R/W	0	reg_nr4_sad2alp1_lut9 : // unsigned , default = 0 sad to alpha1 for temporal blending, dft = 255
15: 8	R/W	16	reg_nr4_sad2alp1_lut10 : // unsigned , default = 16 sad to alpha1 for temporal blending, dft = 255
7: 0	R/W	64	reg_nr4_sad2alp1_lut11 : // unsigned , default = 64 sad to alpha1 for temporal blending, dft = 255

**Table 9-1316 NR4\_MCNR\_SAD2ALP1\_LUT3 0x2de5**

Bit(s)	R/W	Default	Description
31:24	R/W	96	reg_nr4_sad2alp1_lut12 : // unsigned , default = 96 sad to alpha1 for temporal blending, dft = 255
23:16	R/W	224	reg_nr4_sad2alp1_lut13 : // unsigned , default = 224 sad to alpha1 for temporal blending, dft = 255
15: 8	R/W	255	reg_nr4_sad2alp1_lut14 : // unsigned , default = 255 sad to alpha1 for temporal blending, dft = 255
7: 0	R/W	255	reg_nr4_sad2alp1_lut15 : // unsigned , default = 255 sad to alpha1 for temporal blending, dft = 255

**Table 9-1317 NR4\_MCNR\_SAD2BET0\_LUT0 0x2de6**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_nr4_sad2bet0_lut0 : // unsigned , default = 0 sad to beta0 for tnr and mcnr blending, dft = 0
23:16	R/W	2	reg_nr4_sad2bet0_lut1 : // unsigned , default = 2 sad to beta0 for tnr and mcnr blending, dft = 2
15: 8	R/W	4	reg_nr4_sad2bet0_lut2 : // unsigned , default = 4 sad to beta0 for tnr and mcnr blending, dft = 4
7: 0	R/W	8	reg_nr4_sad2bet0_lut3 : // unsigned , default = 8 sad to beta0 for tnr and mcnr blending, dft = 8

**Table 9-1318 NR4\_MCNR\_SAD2BET0\_LUT1 0x2de7**

Bit(s)	R/W	Default	Description
31:24	R/W	16	reg_nr4_sad2bet0_lut4 : // unsigned , default = 16 sad to beta0 for tnr and mcnr blending, dft = 16
23:16	R/W	32	reg_nr4_sad2bet0_lut5 : // unsigned , default = 32 sad to beta0 for tnr and mcnr blending, dft = 32
15: 8	R/W	48	reg_nr4_sad2bet0_lut6 : // unsigned , default = 48 sad to beta0 for tnr and mcnr blending, dft = 48
7: 0	R/W	64	reg_nr4_sad2bet0_lut7 : // unsigned , default = 64 sad to beta0 for tnr and mcnr blending, dft = 64

**Table 9-1319 NR4\_MCNR\_SAD2BET0\_LUT2 0x2de8**

Bit(s)	R/W	Default	Description
31:24	R/W	80	reg_nr4_sad2bet0_lut8 : // unsigned , default = 80 sad to beta0 for tnr and mcnr blending, dft = 80
23:16	R/W	96	reg_nr4_sad2bet0_lut9 : // unsigned , default = 96 sad to beta0 for tnr and mcnr blending, dft = 96
15: 8	R/W	112	reg_nr4_sad2bet0_lut10 : // unsigned , default = 112 sad to beta0 for tnr and mcnr blending, dft = 112
7: 0	R/W	128	reg_nr4_sad2bet0_lut11 : // unsigned , default = 128 sad to beta0 for tnr and mcnr blending, dft = 128

**Table 9-1320 NR4\_MCNR\_SAD2BET0\_LUT3 0x2de9**

Bit(s)	R/W	Default	Description
31:24	R/W	196	reg_nr4_sad2bet0_lut12 : // unsigned , default = 196 sad to beta0 for tnr and mcnr blending, dft = 160
23:16	R/W	224	reg_nr4_sad2bet0_lut13 : // unsigned , default = 224 sad to beta0 for tnr and mcnr blending, dft = 192
15: 8	R/W	255	reg_nr4_sad2bet0_lut14 : // unsigned , default = 255 sad to beta0 for tnr and mcnr blending, dft = 224
7: 0	R/W	255	reg_nr4_sad2bet0_lut15 : // unsigned , default = 255 sad to beta0 for tnr and mcnr blending, dft = 255

**Table 9-1321 NR4\_MCNR\_SAD2BET1\_LUT0 0x2dea**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_nr4_sad2bet1_lut0 : // unsigned , default = 0 sad to beta1 for degghost blending, dft = 0
23:16	R/W	2	reg_nr4_sad2bet1_lut1 : // unsigned , default = 2 sad to beta1 for degghost blending, dft = 2
15: 8	R/W	4	reg_nr4_sad2bet1_lut2 : // unsigned , default = 4 sad to beta1 for degghost blending, dft = 4
7: 0	R/W	8	reg_nr4_sad2bet1_lut3 : // unsigned , default = 8 sad to beta1 for degghost blending, dft = 8

**Table 9-1322 NR4\_MCNR\_SAD2BET1\_LUT1 0x2deb**

Bit(s)	R/W	Default	Description
31:24	R/W	16	reg_nr4_sad2bet1_lut4 : // unsigned , default = 16 sad to beta1 for degghost blending, dft = 16
23:16	R/W	32	reg_nr4_sad2bet1_lut5 : // unsigned , default = 32 sad to beta1 for degghost blending, dft = 32
15: 8	R/W	48	reg_nr4_sad2bet1_lut6 : // unsigned , default = 48 sad to beta1 for degghost blending, dft = 48
7: 0	R/W	64	reg_nr4_sad2bet1_lut7 : // unsigned , default = 64 sad to beta1 for degghost blending, dft = 64

**Table 9-1323 NR4\_MCNR\_SAD2BET1\_LUT2 0x2dec**

Bit(s)	R/W	Default	Description
31:24	R/W	80	reg_nr4_sad2bet1_lut8 : // unsigned , default = 80 sad to beta1 for degghost blending, dft = 80
23:16	R/W	96	reg_nr4_sad2bet1_lut9 : // unsigned , default = 96 sad to beta1 for degghost blending, dft = 96
15: 8	R/W	112	reg_nr4_sad2bet1_lut10 : // unsigned , default = 112 sad to beta1 for degghost blending, dft = 112
7: 0	R/W	128	reg_nr4_sad2bet1_lut11 : // unsigned , default = 128 sad to beta1 for degghost blending, dft = 128

**Table 9-1324 NR4\_MCNR\_SAD2BET1\_LUT3 0x2ded**

Bit(s)	R/W	Default	Description
31:24	R/W	160	reg_nr4_sad2bet1_lut12 : // unsigned , default = 160 sad to beta1 for degghost blending, dft = 160
23:16	R/W	192	reg_nr4_sad2bet1_lut13 : // unsigned , default = 192 sad to beta1 for degghost blending, dft = 192
15: 8	R/W	224	reg_nr4_sad2bet1_lut14 : // unsigned , default = 224 sad to beta1 for degghost blending, dft = 224
7: 0	R/W	255	reg_nr4_sad2bet1_lut15 : // unsigned , default = 255 sad to beta1 for degghost blending, dft = 255

**Table 9-1325 NR4\_MCNR\_SAD2BET2\_LUT0 0x2dee**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_nr4_sad2bet2_lut0 : // unsigned , default = 0 sad to beta2 for snr and mcnr blending, dft = 0
23:16	R/W	1	reg_nr4_sad2bet2_lut1 : // unsigned , default = 1 sad to beta2 for snr and mcnr blending, dft = 2
15: 8	R/W	2	reg_nr4_sad2bet2_lut2 : // unsigned , default = 2 sad to beta2 for snr and mcnr blending, dft = 4
7: 0	R/W	4	reg_nr4_sad2bet2_lut3 : // unsigned , default = 4 sad to beta2 for snr and mcnr blending, dft = 8

**Table 9-1326 NR4\_MCNR\_SAD2BET2\_LUT1 0x2def**

Bit(s)	R/W	Default	Description
31:24	R/W	8	reg_nr4_sad2bet2_lut4 : // unsigned , default = 8 sad to beta2 for snr and mcnr blending, dft = 16
23:16	R/W	16	reg_nr4_sad2bet2_lut5 : // unsigned , default = 16 sad to beta2 for snr and mcnr blending, dft = 32
15: 8	R/W	32	reg_nr4_sad2bet2_lut6 : // unsigned , default = 32 sad to beta2 for snr and mcnr blending, dft = 48
7: 0	R/W	48	reg_nr4_sad2bet2_lut7 : // unsigned , default = 48 sad to beta2 for snr and mcnr blending, dft = 64

**Table 9-1327 NR4\_MCNR\_SAD2BET2\_LUT2 0x2df0**

Bit(s)	R/W	Default	Description
31:24	R/W	64	reg_nr4_sad2bet2_lut8 : // unsigned , default = 64 sad to beta2 for snr and mcnr blending, dft = 80
23:16	R/W	80	reg_nr4_sad2bet2_lut9 : // unsigned , default = 80 sad to beta2 for snr and mcnr blending, dft = 96
15: 8	R/W	96	reg_nr4_sad2bet2_lut10 : // unsigned , default = 96 sad to beta2 for snr and mcnr blending, dft = 112
7: 0	R/W	112	reg_nr4_sad2bet2_lut11 : // unsigned , default = 112 sad to beta2 for snr and mcnr blending, dft = 128

**Table 9-1328 NR4\_MCNR\_SAD2BET2\_LUT3 0x2df1**

Bit(s)	R/W	Default	Description
31:24	R/W	128	reg_nr4_sad2bet2_lut12 : // unsigned , default = 128 sad to beta2 for snr and mcnr blending, dft = 160
23:16	R/W	160	reg_nr4_sad2bet2_lut13 : // unsigned , default = 160 sad to beta2 for snr and mcnr blending, dft = 192
15: 8	R/W	224	reg_nr4_sad2bet2_lut14 : // unsigned , default = 224 sad to beta2 for snr and mcnr blending, dft = 224
7: 0	R/W	255	reg_nr4_sad2bet2_lut15 : // unsigned , default = 255 sad to beta2 for snr and mcnr blending, dft = 255

**Table 9-1329 NR4\_MCNR\_RO\_U\_SUM 0x2df2**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_nr4_u_sum : // unsigned , default = 0 sum of U of current field/frame

**Table 9-1330 NR4\_MCNR\_RO\_V\_SUM 0x2df3**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_nr4_v_sum : // unsigned , default = 0 sum of V of current field/frame

**Table 9-1331 NR4\_MCNR\_RO\_GRDU\_SUM 0x2df4**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_nr4_grdu_sum : // unsigned , default = 0 sum of gradient U of current field/ frame

**Table 9-1332 NR4\_MCNR\_RO\_GRDV\_SUM 0x2df5**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_nr4_grdv_sum : // unsigned , default = 0 sum of gradient V of current field/ frame

**Table 9-1333 NR4\_TOP\_CTRL 0x2dff**

Bit(s)	R/W	Default	Description
31-20	W/R	0	reg_gclk_ctrl
19	W/R	0	reg_text_en ( tm2_revb : DI_MTN_CTRL 0x170b, bit29 )
18	W/R	0	reg_nr4_mcnr_en
17	W/R	0	reg_nr2_en
16	W/R	0	reg_nr4_en
15	W/R	0	reg_nr2_proc_en
14	W/R	0	reg_det3d_en
13	W/R	0	di_polar_en
12	W/R	0	reg_cfr_enable
11-9	W/R	0	reg_3dnr_en_l
8-6	W/R	0	reg_3dnr_en_r
5	W/R	0	reg_nr4_inbuf_ctrl
4	W/R	0	reg_nr4_snr2_en
3	W/R	0	reg_nr4_scene_change_en
2	W/R	0	nr2_sw_en
1	W/R	0	reg_cure_en tm2_revb: DI_NR_CTRL0□0x1707, bit26
0	W/R	0	reg_nr4_scene_change_flg

**Table 9-1334 NR4\_MCNR\_SAD\_GAIN 0x3700**

Bit(s)	R/W	Default	Description
24	R/W	0	reg_nr4_bld12vs3_usemaxsad : // unsigned , default = 0 use minsad/maxsad instead of minsad/avgsad to decision if it was texture or flat region, 1: use minsad/ maxsad
23:16	R/W	64	reg_nr4_bld12vs3_rate_gain : // unsigned , default = 64 gain to minsad/maxsad or minsad/avgsad before LUT, 64 normalized as "1"

Bit(s)	R/W	Default	Description
15: 8	R/W	32	reg_nr4_bld1vs2_rate_gain : // unsigned , default = 32 gain to minsad/maxsad or minsad/avgsad before the LUT, 64 normalized as "1"
7: 0	R/W	64	reg_nr4_coefblt_gain : // unsigned , default = 64 gain to final coefblt, normalized 64 as "1"

Table 9-1335 NR4\_MCNR\_LPF\_CTRL 0x3701

Bit(s)	R/W	Default	Description
30:22	R/W	0	reg_nr4_preflt_alpofst : // signed , default = 0 pre filter alpha ofst
21:16	R/W	16	reg_nr4_preflt_alpgain : // unsigned , default = 16 pre filter alpha gain
15:14	R/W	3	reg_nr4_preflt_alpsel : // unsigned , default = 3 pre filter alpha selection for adaptive blending, 0: mv pointed sad, 1: weighted mv pointed sad, 2or3: coefblt
13: 8	R/W	8	reg_nr4_avgsad_gain : // unsigned , default = 8 gain for avg sad before luts
6	R/W	1	reg_nr4_maxsad_mod : // unsigned , default = 1 max sad select mode, 0: mx2_sad, 1: max sad
5	R/W	1	reg_nr4_minsad_mod : // unsigned , default = 1 min sad select mode, 0: sad with min err, 1: min sad
4	R/W	1	reg_nr4_minmaxsad_lpf : // unsigned , default = 1 mode of lpf for minmaxsad, 0: no LPF, 1: [1 2 1]/4
3	R/W	1	reg_nr4_avgsad_lpf : // unsigned , default = 1 mode of lpf for avgsad, 0: no LPF, 1: [1 2 1]/4
2	R/W	1	reg_nr4_minavgsad_ratio_lpf : // unsigned , default = 1 mode of lpf for minsad/avgsad and zmvvad/avgsad, 0: no LPF, 1: [1 2 1]/4
1	R/W	1	reg_nr4_bldvs_lut_lpf : // unsigned , default = 1 mode of lpf for bld12vs3 and bld1vs2 LUT results, 0: no LPF, 1: [1 2 1]/4
0	R/W	1	reg_nr4_final_coef_lpf : // unsigned , default = 1 mode of lpf for final coef_blt_blend123, 0: no LPF, 1: [1 2 1]/4

Table 9-1336 NR4\_MCNR\_BLD\_VS3LUT0 0x3702

Bit(s)	R/W	Default	Description
29:24	R/W	0	reg_nr4_bld12vs3_lut0 : // unsigned , default = 0
21:16	R/W	8	reg_nr4_bld12vs3_lut1 : // unsigned , default = 8
13: 8	R/W	10	reg_nr4_bld12vs3_lut2 : // unsigned , default = 10
5: 0	R/W	11	reg_nr4_bld12vs3_lut3 : // unsigned , default = 11

Table 9-1337 NR4\_MCNR\_BLD\_VS3LUT1 0x3703

Bit(s)	R/W	Default	Description
29:24	R/W	12	reg_nr4_bld12vs3_lut4 : // unsigned , default = 12
21:16	R/W	14	reg_nr4_bld12vs3_lut5 : // unsigned , default = 14
13: 8	R/W	16	reg_nr4_bld12vs3_lut6 : // unsigned , default = 16
5: 0	R/W	24	reg_nr4_bld12vs3_lut7 : // unsigned , default = 24

**Table 9-1338 NR4\_MCNR\_BLD\_VS3LUT2 0x3704**

Bit(s)	R/W	Default	Description
29:24	R/W	50	reg_nr4_bld12vs3_lut8 : // unsigned , default = 50
21:16	R/W	58	reg_nr4_bld12vs3_lut9 : // unsigned , default = 58
13: 8	R/W	63	reg_nr4_bld12vs3_lut10 : // unsigned , default = 63
5: 0	R/W	63	reg_nr4_bld12vs3_lut11 : // unsigned , default = 63

**Table 9-1339 NR4\_MCNR\_BLD\_VS2LUT0 0x3705**

Bit(s)	R/W	Default	Description
29:24	R/W	63	reg_nr4_bld1vs2_lut0 : // unsigned , default = 63
21:16	R/W	32	reg_nr4_bld1vs2_lut1 : // unsigned , default = 32
13: 8	R/W	16	reg_nr4_bld1vs2_lut2 : // unsigned , default = 16
5: 0	R/W	8	reg_nr4_bld1vs2_lut3 : // unsigned , default = 8

**Table 9-1340 NR4\_MCNR\_BLD\_VS2LUT1 0x3706**

Bit(s)	R/W	Default	Description
29:24	R/W	4	reg_nr4_bld1vs2_lut4 : // unsigned , default = 4
21:16	R/W	2	reg_nr4_bld1vs2_lut5 : // unsigned , default = 2
13: 8	R/W	1	reg_nr4_bld1vs2_lut6 : // unsigned , default = 1
5: 0	R/W	0	reg_nr4_bld1vs2_lut7 : // unsigned , default = 0

**Table 9-1341 NR4\_COEFBLT\_LUT10 0x3707**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_nr4_coefblt_lut10 : // signed , default = -128
23:16	R/W	0x0	reg_nr4_coefblt_lut11 : // signed , default = -128
15: 8	R/W	0x0	reg_nr4_coefblt_lut12 : // signed , default = -126
7: 0	R/W	0x0	reg_nr4_coefblt_lut13 : // signed , default = -124

**Table 9-1342 NR4\_COEFBLT\_LUT11 0x3708**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_nr4_coefblt_lut14 : // signed , default = -120
23:16	R/W	0x0	reg_nr4_coefblt_lut15 : // signed , default = -110
15: 8	R/W	0x0	reg_nr4_coefblt_lut16 : // signed , default = -100
7: 0	R/W	0x0	reg_nr4_coefblt_lut17 : // signed , default = -90

**Table 9-1343 NR4\_COEFBLT\_LUT12 0x3709**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_nr4_coefblt_lut18 : // signed , default = -56
23:16	R/W	0x0	reg_nr4_coefblt_lut19 : // signed , default = -32
15: 8	R/W	0x0	reg_nr4_coefblt_lut110 : // signed , default = -64
7: 0	R/W	0x0	reg_nr4_coefblt_lut111 : // signed , default = -128

**Table 9-1344 NR4\_COEFBLT\_LUT20 0x370a**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_nr4_coefblt_lut20 : // signed , default = -128
23:16	R/W	0x0	reg_nr4_coefblt_lut21 : // signed , default = -120
15: 8	R/W	0x0	reg_nr4_coefblt_lut22 : // signed , default = -112
7: 0	R/W	0x0	reg_nr4_coefblt_lut23 : // signed , default = -104

**Table 9-1345 NR4\_COEFBLT\_LUT21 0x370b**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_nr4_coefblt_lut24 : // signed , default = -96
23:16	R/W	0x0	reg_nr4_coefblt_lut25 : // signed , default = -88
15: 8	R/W	0x0	reg_nr4_coefblt_lut26 : // signed , default = -76
7: 0	R/W	0x0	reg_nr4_coefblt_lut27 : // signed , default = -64

**Table 9-1346 NR4\_COEFBLT\_LUT22 0x370c**

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_nr4_coefblt_lut28 : // signed , default = -48
23:16	R/W	0x0	reg_nr4_coefblt_lut29 : // signed , default = -32
15: 8	R/W	0x0	reg_nr4_coefblt_lut210 : // signed , default = -64
7: 0	R/W	0x0	reg_nr4_coefblt_lut211 : // signed , default = -108

**Table 9-1347 NR4\_COEFBLT\_LUT30 0x370d**

Bit(s)	R/W	Default	Description
31:24	R/W	8	reg_nr4_coefblt_lut30 : // signed , default = 8
23:16	R/W	16	reg_nr4_coefblt_lut31 : // signed , default = 16
15: 8	R/W	24	reg_nr4_coefblt_lut32 : // signed , default = 24
7: 0	R/W	30	reg_nr4_coefblt_lut33 : // signed , default = 30



**Table 9-1348 NR4\_COEFBLT\_LUT31 0x370e**

Bit(s)	R/W	Default	Description
31:24	R/W	36	reg_nr4_coefblt_lut34 : // signed , default = 36
23:16	R/W	48	reg_nr4_coefblt_lut35 : // signed , default = 48
15: 8	R/W	70	reg_nr4_coefblt_lut36 : // signed , default = 70
7: 0	R/W	96	reg_nr4_coefblt_lut37 : // signed , default = 96

**Table 9-1349 NR4\_COEFBLT\_LUT32 0x370f**

Bit(s)	R/W	Default	Description
31:24	R/W	120	reg_nr4_coefblt_lut38 : // signed , default = 120
23:16	R/W	64	reg_nr4_coefblt_lut39 : // signed , default = 64
15: 8	R/W	16	reg_nr4_coefblt_lut310 : // signed , default = 16
7: 0	R/W	0x0	reg_nr4_coefblt_lut311 : // signed , default = -8

**Table 9-1350 NR4\_COEFBLT\_CONV 0x3710**

Bit(s)	R/W	Default	Description
23:16	R/W	0	reg_nr4_coefblt_convmin : // unsigned , default = 0 minimum of coef. bilateral conversion
15: 8	R/W	255	reg_nr4_coefblt_convmax : // unsigned , default = 255 maximum of coef. bilateral conversion
7: 0	R/W	128	reg_nr4_coefblt_convmid : // unsigned , default = 128 value at midpoint of coef. bilateral conversion

**Table 9-1351 NR4\_DBGWIN\_YX0 0x3711**

Bit(s)	R/W	Default	Description
29:16	R/W	100	reg_nr4_dgbwin_yx0 : // unsigned , default = 100 ystart for debug window
13: 0	R/W	160	reg_nr4_dgbwin_yx1 : // unsigned , default = 160 yend for debug window

**Table 9-1352 NR4\_DBGWIN\_YX1 0x3712**

Bit(s)	R/W	Default	Description
29:16	R/W	200	reg_nr4_dgbwin_yx2 : // unsigned , default = 200 xstart for debug window
13: 0	R/W	300	reg_nr4_dgbwin_yx3 : // unsigned , default = 300 xend for debug window

**Table 9-1353 NR4\_NM\_X\_CFG 0x3713**

Bit(s)	R/W	Default	Description
29:16	R/W	8	reg_nr4_nm_xst : // unsigned , default = 8 start for noise meter statistic, dft = 8
13: 0	R/W	711	reg_nr4_nm_xed : // unsigned , default = 711 end for noise meter statistic, dft = HSIZE-8-1;

**Table 9-1354 NR4\_NM\_Y\_CFG 0x3714**

Bit(s)	R/W	Default	Description
29:16	R/W	8	reg_nr4_nm_yst : // unsigned , default = 8 start for noise meter statistic, dft = 8;
13: 0	R/W	231	reg_nr4_nm_yed : // unsigned , default = 231 end for noise meter statistic, dft = VSIZE-8-1;

**Table 9-1355 NR4\_NM\_SAD\_THD 0x3715**

Bit(s)	R/W	Default	Description
7: 0	R/W	255	reg_nr4_nm_sad_thd : // unsigned , default = 255 threshold for (flat region) sad count, dft = 4

**Table 9-1356 NR4\_MCNR\_BANDSPLIT\_PRAM 0x3716**

Bit(s)	R/W	Default	Description
4	R/W	1	reg_nr4_mc_use_bandsplit : // unsigned , default = 1 separate lp and us for mc IIR filter, 0: no BS used; 1: use BS
3	R/W	1	reg_nr4_mc_apply_on_lp : // unsigned , default = 1 use mcnr only on lowpass portion;
2	R/W	1	reg_nr4_mc_apply_on_us : // unsigned , default = 1 use mcnr only on lp complementary portion;
1: 0	R/W	1	reg_nr4_mc_zmvbs_use_adplpf : // unsigned , default = 1 use adaptive LPF for the zmv pointing data for MCNR, for abs(mvx)<th

**Table 9-1357 NR4\_MCNR\_ALP1\_SGN\_COR 0x3717**

Bit(s)	R/W	Default	Description
31:24	R/W	10	reg_nr4_mc_aph1_sgn_coring0 : // unsigned , default = 10 coring to cur-pre before do sgn decision
23:16	R/W	7	reg_nr4_mc_aph1_sgn_coring1 : // unsigned , default = 7 coring to cur-pre before do sgn decision
15: 8	R/W	90	reg_nr4_mc_aph1_sgn_core_max0 : // unsigned , default = 90 maximum of coring, default = 30/15
7: 0	R/W	15	reg_nr4_mc_aph1_sgn_core_max1 : // unsigned , default = 15 maximum of coring, default = 30/15

**Table 9-1358 NR4\_MCNR\_ALP1\_SGN\_PRAM 0x3718**

Bit(s)	R/W	Default	Description
10	R/W	1	reg_nr4_mc_alp1_sgn_half : // unsigned , default = 1 half block sgn sum mode enable, 0: only use 3x5 whole block sum of sgns; 1: use max(sgn_3x5, sqrt(sgn_left+sgn_right))
9	R/W	1	reg_nr4_mc_alp1_sgn_frczmv : // unsigned , default = 1 force zmv to calculate the sign_sum;
8	R/W	1	reg_nr4_mc_alp1_sgnmvx_mode : // unsigned , default = 1 blend mode of sgnlut and mvxlut blend mode: 0: sgnlut+ mvxlut; 1: max(sgnlut, mvxlut), default = 1

Bit(s)	R/W	Default	Description
7: 4	R/W	4	reg_nr4_mc_aph1_sgn_crate0 : // unsigned , default = 4 rate to var, norm to 16 as 1, default = 2
3: 0	R/W	2	reg_nr4_mc_aph1_sgn_crate1 : // unsigned , default = 2 rate to var, norm to 16 as 1, default = 2

Table 9-1359 NR4\_MCNR\_ALP1\_MVX\_LUT1 0x3719

Bit(s)	R/W	Default	Description
31:28	R/W	14	reg_nr4_mc_alp1_mv_x_luty3 : // unsigned , default = 14 alp1 of luma vas mvx (0~7), and alp1 vs mvy(0,1)
27:24	R/W	14	reg_nr4_mc_alp1_mv_x_lutc3 : // unsigned , default = 14 alp1 of chrm vas mvx (0~7), and alp1 vs mvy(0,1)
23:20	R/W	12	reg_nr4_mc_alp1_mv_x_luty2 : // unsigned , default = 12 alp1 of luma vas mvx (0~7), and alp1 vs mvy(0,1)
19:16	R/W	12	reg_nr4_mc_alp1_mv_x_lutc2 : // unsigned , default = 12 alp1 of chrm vas mvx (0~7), and alp1 vs mvy(0,1)
15:12	R/W	5	reg_nr4_mc_alp1_mv_x_luty1 : // unsigned , default = 5 alp1 of luma vas mvx (0~7), and alp1 vs mvy(0,1)
11: 8	R/W	5	reg_nr4_mc_alp1_mv_x_lutc1 : // unsigned , default = 5 alp1 of chrm vas mvx (0~7), and alp1 vs mvy(0,1)
7: 4	R/W	3	reg_nr4_mc_alp1_mv_x_luty0 : // unsigned , default = 3 alp1 of luma vas mvx (0~7), and alp1 vs mvy(0,1)
3: 0	R/W	3	reg_nr4_mc_alp1_mv_x_lutc0 : // unsigned , default = 3 alp1 of chrm vas mvx (0~7), and alp1 vs mvy(0,1)

Table 9-1360 NR4\_MCNR\_ALP1\_MVX\_LUT2 0x371a

Bit(s)	R/W	Default	Description
31:28	R/W	15	reg_nr4_mc_alp1_mv_x_luty7 : // unsigned , default = 15 alp1 of luma vas mvx (0~7), and alp1 vs mvy(0,1)
27:24	R/W	15	reg_nr4_mc_alp1_mv_x_lutc7 : // unsigned , default = 15 alp1 of chrm vas mvx (0~7), and alp1 vs mvy(0,1)
23:20	R/W	15	reg_nr4_mc_alp1_mv_x_luty6 : // unsigned , default = 15 alp1 of luma vas mvx (0~7), and alp1 vs mvy(0,1)
19:16	R/W	15	reg_nr4_mc_alp1_mv_x_lutc6 : // unsigned , default = 15 alp1 of chrm vas mvx (0~7), and alp1 vs mvy(0,1)
15:12	R/W	15	reg_nr4_mc_alp1_mv_x_luty5 : // unsigned , default = 15 alp1 of luma vas mvx (0~7), and alp1 vs mvy(0,1)
11: 8	R/W	15	reg_nr4_mc_alp1_mv_x_lutc5 : // unsigned , default = 15 alp1 of chrm vas mvx (0~7), and alp1 vs mvy(0,1)
7: 4	R/W	15	reg_nr4_mc_alp1_mv_x_luty4 : // unsigned , default = 15 alp1 of luma vas mvx (0~7), and alp1 vs mvy(0,1)
3: 0	R/W	15	reg_nr4_mc_alp1_mv_x_lutc4 : // unsigned , default = 15 alp1 of chrm vas mvx (0~7), and alp1 vs mvy(0,1)

**Table 9-1361 NR4\_MCNR\_ALP1\_MVX\_LUT3 0x371b**

Bit(s)	R/W	Default	Description
7: 4	R/W	6	reg_nr4_mc_alp1_mvx_luty8 : // unsigned , default = 6 alp1 of luma vas mvx (0~7), and alp1 vs mvy(0,1)
3: 0	R/W	6	reg_nr4_mc_alp1_mvx_lutc8 : // unsigned , default = 6 alp1 of chrm vas mvx (0~7), and alp1 vs mvy(0,1)

**Table 9-1362 NR4\_MCNR\_ALP1\_LP\_PRAM 0x371c**

Bit(s)	R/W	Default	Description
17:16	R/W	1	reg_nr4_mc_alp1_lp_sel : // unsigned , default = 1 mode for alp1_lp for lp portion IIR, 0: apha1, 1:dc_dif vs ac analysis; 2: gain/ofst of alp1; 3: max of 1/ 2 results
15: 8	R/W	64	reg_nr4_mc_alp1_lp_gain : // unsigned , default = 64 gain to alp1 to get the alp1_lp = alp1*gain/32 + ofset, default =64;
7: 0	R/W	0	reg_nr4_mc_alp1_lp_ofst : // signed , default = 0 offset to alp1 to get the alp1_lp = alp1*gain/32 + ofset, default =10;

**Table 9-1363 NR4\_MCNR\_ALP1\_SGN\_LUT1 0x371d**

Bit(s)	R/W	Default	Description
31:28	R/W	3	reg_nr4_mc_alp1_sgn_lut0 : // unsigned , default = 3 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move
27:24	R/W	3	reg_nr4_mc_alp1_sgn_lut1 : // unsigned , default = 3 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move
23:20	R/W	3	reg_nr4_mc_alp1_sgn_lut2 : // unsigned , default = 3 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move
19:16	R/W	4	reg_nr4_mc_alp1_sgn_lut3 : // unsigned , default = 4 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move
15:12	R/W	5	reg_nr4_mc_alp1_sgn_lut4 : // unsigned , default = 5 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move
11: 8	R/W	6	reg_nr4_mc_alp1_sgn_lut5 : // unsigned , default = 6 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move
7: 4	R/W	7	reg_nr4_mc_alp1_sgn_lut6 : // unsigned , default = 7 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move
3: 0	R/W	8	reg_nr4_mc_alp1_sgn_lut7 : // unsigned , default = 8 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move

**Table 9-1364 NR4\_MCNR\_ALP1\_SGN\_LUT2 0x371e**

Bit(s)	R/W	Default	Description
31:28	R/W	9	reg_nr4_mc_alp1_sgn_lut8 : // unsigned , default = 9 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move
27:24	R/W	10	reg_nr4_mc_alp1_sgn_lut9 : // unsigned , default = 10 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move
23:20	R/W	11	reg_nr4_mc_alp1_sgn_lut10 : // unsigned , default = 11 alp1 vs x=abs sgn(cur-pre) , if x is small, less possibility of flat region move

Bit(s)	R/W	Default	Description
19:16	R/W	12	reg_nr4_mc_alp1_sgn_lut11 : // unsigned , default = 12 alp1 vs x=abs sgn(cur-pre), if x is small, less possibility of flat region move
15:12	R/W	13	reg_nr4_mc_alp1_sgn_lut12 : // unsigned , default = 13 alp1 vs x=abs sgn(cur-pre), if x is small, less possibility of flat region move
11: 8	R/W	14	reg_nr4_mc_alp1_sgn_lut13 : // unsigned , default = 14 alp1 vs x=abs sgn(cur-pre), if x is small, less possibility of flat region move
7: 4	R/W	15	reg_nr4_mc_alp1_sgn_lut14 : // unsigned , default = 15 alp1 vs x=abs sgn(cur-pre), if x is small, less possibility of flat region move
3: 0	R/W	15	reg_nr4_mc_alp1_sgn_lut15 : // unsigned , default = 15 alp1 vs x=abs sgn(cur-pre), if x is small, less possibility of flat region move

**Table 9-1365 NR4\_RO\_NM\_SAD\_SUM 0x371f**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_nr4_nm_sad_sum : // unsigned , default = 0 sum of sad, for scene change detection, in noise meter

**Table 9-1366 NR4\_RO\_NM\_SAD\_CNT 0x3720**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_nr4_nm_sad_cnt : // unsigned , default = 0 cnt of sad, for scene change detection, in noise meter

**Table 9-1367 NR4\_RO\_NM\_VAR\_SUM 0x3721**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_nr4_nm_var_sum : // unsigned , default = 0 sum of var, for noise level detection, in noise meter

**Table 9-1368 NR4\_RO\_NM\_VAR\_SCNT 0x3722**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_nr4_nm_var_cnt : // unsigned , default = 0 cnt of var, for noise level detection, in noise meter

**Table 9-1369 NR4\_RO\_NM\_VAR\_MIN\_MAX 0x3723**

Bit(s)	R/W	Default	Description
21:12	R.O	1023	ro_nr4_nm_min_var : // unsigned , default = 1023 min of var, for noise level detection, in noise meter
9: 0	R.O	0	ro_nr4_nm_max_var : // unsigned , default = 0 max of var, for noise level detection, in noise meter

**Table 9-1370 NR4\_RO\_NR4\_DBGPIX\_NUM 0x3724**

Bit(s)	R/W	Default	Description
27: 0	R.O	0	ro_nr4_dbgpix_num : // unsigned , default = 0 number of pixels statistic invoved (removed?)

**Table 9-1371 NR4\_RO\_NR4\_BLDVS2\_SUM 0x3725**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_nr4_bld1vs2_sum : // unsigned , default = 0 sum of blend_1vs2 with the debug window

**Table 9-1372 NR4\_BLDVS3\_SUM 0x3726**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_nr4_bld12vs3_sum : // unsigned , default = 0 sum of blend_12vs3 with the debug window

**Table 9-1373 NR4\_COEF12\_SUM 0x3727**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_nr4_coef12_sum : // signed , default = 0 sum of coef_blt_blend12 with the debug window, under 8 bits precision

**Table 9-1374 NR4\_COEF123\_SUM 0x3728**

Bit(s)	R/W	Default	Description
31: 0	R.O	0	ro_nr4_coef123_sum : // signed , default = 0 sum of coef_final with the debug window, under 8 bits precision

**Table 9-1375 XLR\_CTRL 0x3730**

Bit(s)	R/W	Default	Description
22	R/W	1	reg_xlr_en : // unsigned , default = 1 enable bits for xlr function, 1: enable, 0: disable default = 1
21	R/W	1	reg_xlr_side_en : // unsigned , default = 1 enable to filter the above and below lines with xlr filter, default = 1
20	R/W	1	reg_xlr_3lines : // unsigned , default = 1 3 lines version enable default= 0
19:16	R/W	12	reg_xlr_simp_gain : // unsigned , default = 12 gain to simp to decide if need the XLR, default= 12
13: 8	R/W	6	reg_xlr_oopl_gain : // unsigned , default = 6 gain to out-of-phase lp error to decide if need the XLR, birn to 16 as 1, default= 3
7: 0	R/W	64	reg_xlr_dislp_thrd : // unsigned , default = 64 threshold to lp error to discard XLR. default= 80

**Table 9-1376 XLR\_THRD 0x3731**

Bit(s)	R/W	Default	Description
27:24	R/W	3	reg_xlr_txt_core : // unsigned , default = 3 coring to texture default = 3, (3/32)
23:16	R/W	5	reg_xlr_err_thrd2 : // unsigned , default = 5 threshold to error to decide blending coef, 0, 1/4, 1/2, 1.0, default= {20, 10, 5}
15: 8	R/W	10	reg_xlr_err_thrd1 : // unsigned , default = 10 threshold to error to decide blending coef, 0, 1/4, 1/2, 1.0, default= {20, 10, 5}
7: 0	R/W	20	reg_xlr_err_thrd0 : // unsigned , default = 20 threshold to error to decide blending coef, 0, 1/4, 1/2, 1.0, default= {20, 10, 5}

**Table 9-1377 XLR\_HCT\_THRD 0x3732**

Bit(s)	R/W	Default	Description
27:26	R/W	1	reg_xlr_hct_step : // unsigned , default = 1 horizontal chroma diff steps, 0: abs[-1 (2) -1]; 1: abs[-1 0 (2) 0 -1], 2: abs[-1 0 0 (2) 0 0 -1]; 3:abs[-1 0 0 0 (2) 0 0 0 -1]
25:24	R/W	1	reg_xlr_hct_lpf : // unsigned , default = 1 horizontal chroma diff low-pass filter enable, 0: no HLPF; 1: [1 2 1]; 2: max[-1:1]; 3: max[-2:2]
23:16	R/W	20	reg_xlr_hct_thr : // unsigned , default = 20 horizontal chroma diff threshold for xlr enable, to save the horizontal no color tran mode
15: 8	R/W	8	reg_xlr_sat_thr : // unsigned , default = 8 saturation threshold for xlr enable, the smaller of the threshold , the more will do xlr;
7: 2	R/W	3	reg_xlr_hmargin : // unsigned , default = 3 left/right number of pixels without xlr;
0	R/W	0	reg_xlr_hpf_only : // unsigned , default = 0 enable for only do xlr filter on high pass portion of the data, instead full data

**Table 9-1378 NR\_DB\_FLT\_CTRL 0x3738**

Bit(s)	R/W	Default	Description
26	R/W	0	reg_nrdeband_reset1 : // unsigned , default = 0 0 : no reset seed 1: reload chroma seed
25	R/W	0	reg_nrdeband_reset0 : // unsigned , default = 0 0 : no reset seed 1: reload luma seed
24	R/W	0	reg_nrdeband_rgb : // unsigned , default = 0 0 : yuv 1: RGB
23	R/W	1	reg_nrdeband_en11 : // unsigned , default = 1 debanding registers of side lines, [0] for luma, same for below
22	R/W	1	reg_nrdeband_en10 : // unsigned , default = 1 debanding registers of side lines, [1] for chroma, same for below
21	R/W	1	reg_nrdeband_siderand : // unsigned , default = 1 options to use side two lines use the rand, instead of use for the YUV three component of middle line, 0: seed [3]/bandrand[3] for middle line yuv; 1: seed[3]/bandrand[3] for nearby three lines Y;
20	R/W	0	reg_nrdeband_randmode : // unsigned , default = 0 mode of rand noise adding, 0: same noise strength for all difs; else: strenght of noise will not exceed the difs, MIN((pPKReg->reg_nrdeband_bandrand[m]), noise[m])
19:17	R/W	6	reg_nrdeband_bandrand2 : // unsigned , default = 6
15:13	R/W	6	reg_nrdeband_bandrand1 : // unsigned , default = 6
11: 9	R/W	6	reg_nrdeband_bandrand0 : // unsigned , default = 6

Bit(s)	R/W	Default	Description
7	R/W	1	reg_nrdeband_hpxor1 : // unsigned , default = 1 debanding random hp portion xor, [0] for luma
6	R/W	1	reg_nrdeband_hpxor0 : // unsigned , default = 1 debanding random hp portion xor, [1] for chroma
5	R/W	1	reg_nrdeband_en1 : // unsigned , default = 1 debanding registers, for luma
4	R/W	1	reg_nrdeband_en0 : // unsigned , default = 1 debanding registers, for chroma
3: 2	R/W	2	reg_nrdeband_lpf_mode1 : // unsigned , default = 2 lpf mode, 0: 3x3, 1:3x5; 2: 5x5; 3:5x7
1: 0	R/W	2	reg_nrdeband_lpf_mode0 : // unsigned , default = 2 lpf mode, 0: 3x3, 1:3x5; 2: 5x5; 3:5x7

Table 9-1379 NR\_DB\_FLT\_YC\_THRD 0x3739

Bit(s)	R/W	Default	Description
31:28	R/W	9	reg_nrdeband_luma_th3 : // unsigned , default = 9 threshold to $ Y-Y _{lpf}$ , if < th [0] use lpf
27:24	R/W	7	reg_nrdeband_luma_th2 : // unsigned , default = 7 elseif <th[1] use $(lpf^3 + y)/4$
23:20	R/W	6	reg_nrdeband_luma_th1 : // unsigned , default = 6 elseif <th[1] use $(lpf^3 + y)/4$ elseif <th[2] $(lpf^1 + y)/2$
19:16	R/W	5	reg_nrdeband_luma_th0 : // unsigned , default = 5 elseif <th[1] use $(lpf^3 + y)/4$ elseif elseif <th[3] $(lpf^1 + 3*y)/4$ ; else
15:12	R/W	9	reg_nrdeband_chrm_th3 : // unsigned , default = 9 threshold to $ Y-Y _{lpf}$ , if < th [0] use lpf
11: 8	R/W	7	reg_nrdeband_chrm_th2 : // unsigned , default = 7 elseif <th[1] use $(lpf^3 + y)/4$
7: 4	R/W	6	reg_nrdeband_chrm_th1 : // unsigned , default = 6 elseif <th[1] use $(lpf^3 + y)/4$ elseif <th[2] $(lpf^1 + y)/2$
3: 0	R/W	5	reg_nrdeband_chrm_th0 : // unsigned , default = 5 elseif <th[1] use $(lpf^3 + y)/4$ elseif elseif

Table 9-1380 NR\_DB\_FLT\_RANDLUT 0x373a

Bit(s)	R/W	Default	Description
23:21	R/W	1	reg_nrdeband_randslut7 : // unsigned , default = 1 lut0
20:18	R/W	1	reg_nrdeband_randslut6 : // unsigned , default = 1 lut0
17:15	R/W	1	reg_nrdeband_randslut5 : // unsigned , default = 1 lut0
14:12	R/W	1	reg_nrdeband_randslut4 : // unsigned , default = 1 lut0
11: 9	R/W	1	reg_nrdeband_randslut3 : // unsigned , default = 1 lut0
8: 6	R/W	1	reg_nrdeband_randslut2 : // unsigned , default = 1 lut0
5: 3	R/W	1	reg_nrdeband_randslut1 : // unsigned , default = 1 lut0
2: 0	R/W	1	reg_nrdeband_randslut0 : // unsigned , default = 1 lut0



**Table 9-1381 NR\_DB\_FLT\_PXI\_THRD 0x373b**

Bit(s)	R/W	Default	Description
25:16	R/W	0	reg_nrdeband_yc_th1 : // unsigned , default = 0 to luma/ u/v  for using the denoise
9: 0	R/W	0	reg_nrdeband_yc_th0 : // unsigned , default = 0 to luma/ u/v  for using the denoise

**Table 9-1382 NR\_DB\_FLT\_SEED\_Y 0x373c**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8240	reg_nrdeband_seed0 : // unsigned , default = 1621438240 noise adding seed for Y. seed[0]= 0x60a52f20; as default

**Table 9-1383 NR\_DB\_FLT\_SEED\_U 0x373d**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8247	reg_nrdeband_seed1 : // unsigned , default = 1621438247 noise adding seed for U. seed[0]= 0x60a52f27; as default

**Table 9-1384 NR\_DB\_FLT\_SEED\_V 0x373e**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8242	reg_nrdeband_seed2 : // unsigned , default = 1621438242 noise adding seed for V. seed[0]= 0x60a52f22; as default

**Table 9-1385 NR\_DB\_FLT\_SEED3 0x373f**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8242	reg_nrdeband_seed3 : // unsigned , default = 1621438242 noise adding seed for V. seed[0]= 0x60a52f22; as default

**Table 9-1386 NRWR\_DBG\_AXI\_CMD\_CNT 0x2090**

Bit(s)	R/W	Default	Description
31:0	RO	0	wr_mif_dbg_axi_cmd_cnt unsigned, default = 0,

**Table 9-1387 NRWR\_DBG\_AXI\_DAT\_CNT 0x2091**

Bit(s)	R/W	Default	Description
31:0	RO	0	wr_mif_dbg_axi_dat_cnt unsigned, default = 0,

**Table 9-1388 DI\_NRWR\_CANVAS 0x2092**

Bit(s)	R/W	Default	Description
31:16	/	0	reserved
15:8	R/W	0	wmif_canvas_index_chroma unsigned,default = 0,
7 :0	R/W	0	wmif_canvas_index_luma unsigned,default = 0,

**Table 9-1389 DI\_NRWR\_URGENT 0x2093**

Bit(s)	R/W	Default	Description
31:16	/	0	reserved
15 :0	R/W	0	wmif_urgent_ctrl unsigned,default = 0,

**Table 9-1390 DI\_NRWR\_X 0x2094**

Bit(s)	R/W	Default	Description
31	/		reserved
30	R/W		wmif_rev_x unsigned,default = 0, bit
29:16	R/W		wmif_end_x unsigned,default = 0,
13:0	R/W		wmif_start_x unsigned,default = 0,

**Table 9-1391 DI\_NRWR\_Y 0x2095**

Bit(s)	R/W	Default	Description
31	/		reserved
30	R/W		wmif_rev_x unsigned,default = 0, bit
29:16	R/W		wmif_end_x unsigned,default = 0,
13:0	R/W		wmif_start_x unsigned,default = 0,

**Table 9-1392 DI\_NRWR\_CTRL 0x2097**

Bit(s)	R/W	Default	Description
31	R/W	0	Pending_ddr_wrrsp_nrwr
30	R/W	0	Nrwr_reg_swap
29-26	R/W	0	Nrwr_burst_lim
25	R/W	0	Nrwr_canvas_syncen
24	R/W	0	Nrwr_no_clk_gate
23-22	R/W	0	Nrwr_rgb_mode , 0: 4:2:2 to one canvas; 1: 4:4:4 to one canvas; 2: Y to luma canvas, CbCr to chroma canvas, for NV12/21; 3: Reserved.
21-20	R/W	0	Nrwr_hconv_mode
19-18	R/W	0	Nrwr_vconv_mode
17	R/W	0	Nrwr_swap_cbcr

Bit(s)	R/W	Default	Description
16	R/W	0	Nrwr_urgent
15-8	R/W	0	Nrwr_canvas_index_chroma
7-0	R/W	0	Nrwr_canvas_index_luma

Table 9-1393 DI\_NRWR\_SHRK\_CTRL 0x2097

Bit(s)	R/W	Default	Description
31:10	R/W	0	reserved
9:8	R/W	0	reg_vshrk_mode unsigned, default = 0, 0:1/2 horizontal shrink 1:1/4 horizontal shrink 2:1/8 horizontal shrink bit
7:6	R/W	0	reg_hshrk_mode unsigned, default = 0, 0:1/2 vertical shrink 1:1/4 vertical shrink 2:1/8 vertical shrink
5:2	R/W	0	reg_gclk_ctrl unsigned, default = 0
1	R/W	0	reg_frm_rst unsigned, default = 0
0	R/W	0	reg_shrk_en unsigned, default = 0

Table 9-1394 DI\_NRWR\_SHRK\_SIZE 0x2098

Bit(s)	R/W	Default	Description
31:26	R/W	0	reserved
25:13	R/W	0	reg_frm_hsize unsigned, default = 1920, hsize in
12:0	R/W	0	reg_frm_vsize unsigned, default = 1080, vsize in

Table 9-1395 DI\_NRWR\_CROP\_CTRL 0x209a

Bit(s)	R/W	Default	Description
31	R/W	0	reg_crop_en : unsigned, RW,default = 0,dimm_layer enable signal
3:0	R/W	4	reg_hold_line : unsigned, RW,default = 4,dimm_layer data

Table 9-1396 DI\_NRWR\_CROP\_DIMM\_CTRL 0x209b

Bit(s)	R/W	Default	Description
31	R/W	0	reg_dimm_layer_en : unsigned, RW,default = 0,dimm_layer enable signal
29:0	R/W	0x0008-0200	reg_dimm_data : unsigned, RW,default = 29'h00080200,dimm_layer data

Table 9-1397 DI\_NRWR\_CROP\_SIZE\_IN 0x209c

Bit(s)	R/W	Default	Description
28:16	R/W	1920	reg_crop_hsize : unsigned, default = 1920 , pic horz size in unit: pixel
12:0	R/W	1080	reg_crop_vsize : unsigned, default = 1080 , pic vert size in unit: pixel

**Table 9-1398 DI\_NRWR\_CROP\_HSCOPE 0x209d**

Bit(s)	R/W	Default	Description
28:16	R/W	1919	reg_cropwin_end_h : unsigned, default = 1919 ;
12:0	R/W	0	reg_cropwin_bgn_h : unsigned, default = 0 ;

**Table 9-1399 DI\_NRWR\_CROP\_VSCOPE 0x209e**

Bit(s)	R/W	Default	Description
28:16	R/W	1079	reg_cropwin_end_v : unsigned, default = 1079 ;
12:0	R/W	0	reg_cropwin_bgn_v : unsigned, default = 0 ;

**Table 9-1400 DIWR\_DBG\_AXI\_CMD\_CNT 0x20f0**

Bit(s)	R/W	Default	Description
31:0	RO	0	wr_mif_dbg_axi_cmd_cnt unsigned, default = 0,

**Table 9-1401 DIWR\_DBG\_AXI\_DAT\_CNT 0x20f1**

Bit(s)	R/W	Default	Description
31:0	RO	0	wr_mif_dbg_axi_dat_cnt unsigned, default = 0,

**Table 9-1402 DI\_DIWR\_CANVAS 0x20f2**

Bit(s)	R/W	Default	Description
31:16	/	0	reserved
15:8	R/W	0	wmif_canvas_index_chroma unsigned,default = 0,
7 :0	R/W	0	wmif_canvas_index_luma unsigned,default = 0,

**Table 9-1403 DI\_DIWR\_URGENT 0x20f3**

Bit(s)	R/W	Default	Description
31:16	/	0	reserved
15 :0	R/W	0	wmif_urgent_ctrl unsigned,default = 0,

**Table 9-1404 DI\_DIWR\_X 0x20f4**

Bit(s)	R/W	Default	Description
31	/		reserved
30	R/W		wmif_rev_x unsigned,default = 0, bit
29:16	R/W		wmif_end_x unsigned,default = 0,
13:0	R/W		wmif_start_x unsigned,default = 0,



**Table 9-1408 DI\_DIWR\_SHRK\_SIZE 0x20f8**

Bit(s)	R/W	Default	Description
31:26	R/W	0	reserved
25:13	R/W	0	reg_frm_hsize unsigned, default = 1920, hsize in
12:0	R/W	0	reg_frm_vsize unsigned, default = 1080, vsize in

**Table 9-1409 DI\_DIWR\_CROP\_CTRL 0x20fa**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_crop_en : unsigned, RW,default = 0,dimm_layer enable signal
3:0	R/W	4	reg_hold_line : unsigned, RW,default = 4,dimm_layer data

**Table 9-1410 DI\_DIWR\_CROP\_DIMM\_CTRL 0x20fb**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_dimm_layer_en : unsigned, RW,default = 0,dimm_layer enable signal
29:0	R/W	0x0008-0200	reg_dimm_data : unsigned, RW,default = 29'h00080200,dimm_layer data

**Table 9-1411 DI\_DIWR\_CROP\_SIZE\_IN 0x20fc**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	reg_crop_hsize : unsigned, default = 1920 , pic horz size in unit: pixel
12:0	R/W	1080	reg_crop_vsize : unsigned, default = 1080 , pic vert size in unit: pixel

**Table 9-1412 DI\_DIWR\_CROP\_HSCOPE 0x20fd**

Bit(s)	R/W	Default	Description
28:16	R/W	1919	reg_cropwin_end_h : unsigned, default = 1919 ;
12:0	R/W	0	reg_cropwin_bgn_h : unsigned, default = 0 ;

**Table 9-1413 DI\_DIWR\_CROP\_VSCOPE 0x20fe**

Bit(s)	R/W	Default	Description
28:16	R/W	1079	reg_cropwin_end_v : unsigned, default = 1079 ;
12:0	R/W	0	reg_cropwin_bgn_v : unsigned, default = 0 ;

### 9.2.3.16 VIUB Top-Level Registers

Table 9-1414 VIU\_SW\_RESET 1a01

Bit(s)	R/W	Default	Description
31	R/W	0	dolby1b_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
30	R/W	0	dolby1a_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
30	R/W	0	mali_afbcd2_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
28	R/W	0	mali_afbcd2_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0),Only active when MALI_AFBCD2_TOP_CTRL[23] high
27	R/W	0	mali_afbcd1_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
26	R/W	0	mali_afbcd1_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0) ,Only active when MALI_AFBCD1_TOP_CTRL[23] high
25	R/W	0	vd_mif_afbcd2_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
24	R/W	0	vd_mif_afbcd1_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
23	R/W	0	vd_mif_afbcd0_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
22	R/W	0	mali_afbcd0_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
21	R/W	0	mali_afbcd0_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0) ,Only active when MALI_AFBCD_TOP_CTRL[23] high
20	R/W	0	afbc_arb_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
19	R/W	0	afbc_arb_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
17	R/W	0	ods4_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
16	R/W	0	ods3_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
15	R/W	0	ods2_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
14	R/W	0	ods1_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
13	R/W	0	vpp_axi_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
12	R/W	0	lc_stts_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
9	R/W	0	vks_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
8	R/W	0	vpp2_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)

Bit(s)	R/W	Default	Description
7	R/W	0	vpp1_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
6	R/W	0	prime_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
5	R/W	0	hist_spl_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
4	R/W	0	ldim_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
3	R/W	0	dolby2b_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
2	R/W	0	dolby2a_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
1	R/W	0	dolby_tv_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
0	R/W	0	vpp_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)

Table 9-1415 VIU\_SW\_RESET0 1a02

Bit(s)	R/W	Default	Description
2	R/W	0	dolby1c_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
1	R/W	0	dolby2d_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)
0	R/W	0	dolby2c_sw_reset : // unsigned , RW , default = 0 Reset module by write pulse (0->1->0)

Table 9-1416 VIU\_MISC\_CTRL0 1a06

Bit(s)	R/W	Default	Description
10	R/W	0	vsync2_int_ctrl : // unsigned , RW , default = 0 vsync_int_pre = go_field_ff & ((~viu_misc_ctrl0[10])   viu_misc_ctrl0[10] & (~field_ff));
9	R/W	0	vsync1_int_ctrl : // unsigned , RW , default = 0 vsync_int_pre = go_field_ff & ((~viu_misc_ctrl0[9] )   viu_misc_ctrl0[9] & (~field_ff));
8	R/W	0	vsync0_int_ctrl : // unsigned , RW , default = 0 vsync_int_pre = go_field_ff & ((~viu_misc_ctrl0[8] )   viu_misc_ctrl0[8] & (~field_ff));
0	R/W	0	scan_reg : // unsigned , RW , default = 0 just some register bit, used for scan coverage

Table 9-1417 VD1\_AFBCD0\_MISC\_CTRL 1a0a

Bit(s)	R/W	Default	Description
11	R/W	0	afbc0_osd3_mux_vd1 : // unsigned , RW , default = 0 0:afbcd to vd1 1:osd3 to vd1
8	R/W	0	di_mif0_en : // unsigned , RW , default = 0 0:mif_out to vpp 1:mif_out to di



**Table 9-1418 VD2\_AFBCD1\_MISC\_CTRL 1a0b**

Bit(s)	R/W	Default	Description
15:14	R/W	0	vd2_in_mux_sel : // unsigned , RW ,default = 0 0:vd2 to vpp_wrap_vd2 1:vd2 to vpp_wrap_osd3 2: vd2 to vpp_wrap_osd4
11	R/W	0	afbc1_osd4_mux_vd2 : // unsigned , RW ,default = 0 0:afbc to vd2 1:osd4 to vd2
8:0	R/W	0	vd2_osd_mux_alpha : // unsigned , RW ,default = 0 alpha register for vd2_in_mux_sel == 1/2

**Table 9-1419 WR\_BACK\_MISC\_CTRL 1a0d**

Bit(s)	R/W	Default	Description
3	R/W	0	wrbak_chan3_hsyn_en : // unsigned , RW ,default = 0 vpp0 wrbak0 hsync enable
2	R/W	0	wrbak_chan2_hsyn_en : // unsigned , RW ,default = 0 vpp0 wrbak1 hsync enable
1	R/W	0	wrbak_chan1_hsyn_en : // unsigned , RW ,default = 0 vpp1 wrbak0 hsync enable
0	R/W	0	wrbak_chan0_hsyn_en : // unsigned , RW ,default = 0 vpp2 wrbak0 hsync enable

**Table 9-1420 OSD\_PATH\_MISC\_CTRL 1a0e**

Bit(s)	R/W	Default	Description
31:16	R/W	0	vpp_osd_4mux4_sel : // unsigned , RW ,default = 0 vpp_{osd4,osd3,osd2,osd1}_din_mux 1:osd1 2:osd2 3:osd3 4:osd4 else:close

**Table 9-1421 MALI\_AFBCD\_TOP\_CTRL 1a0f**

Bit(s)	R/W	Default	Description
23	R/W	0	mali_afbcd1_sw_rst_sel : // unsigned ,RW , default = 0 1:mali_afbcd0_sw_reset [0] 2:go_field
22	R/W	0	reg_osd2_secure : // unsigned ,RW , default = 0 osd_axi secure bit
21	R/W	0	reg_osd2_axi_sel : // unsigned ,RW , default = 0 1:osd read mali_afbcd_data 0: osd read mif data
10	R/W	0	osd2_din_ext_mode : // unsigned ,RW , default = 0 1:add two bits 0 at high bits 0: add two bits 0 at low bits
19	R/W	0	osd2_dolby_bypass_en : // unsigned ,RW , default = 0 1:data bypass_osd2_dolby 0:data into_osd2_dolby
18	R/W	0	reg_mali_afbcd_sec : // unsigned ,RW , default = 0 mali_afbcd_axi secure bit
17	R/W	0	reg_osd1_secure : // unsigned ,RW , default = 0 osd_axi secure bit
16	R/W	0	reg_osd1_axi_sel : // unsigned ,RW , default = 0 1:osd read mali_afbcd_data 0: osd read mif data
15	R/W	0	osd1_din_ext_mode : // unsigned ,RW , default = 0 1:add two bits 0 at high bits 0: add two bits 0 at low bits
14	R/W	0	osd1_dolby_bypass_en : // unsigned ,RW , default = 0 1:data bypass_osd1_dolby 0:data into_osd1_dolby
13:12	R/W	0	mali_afbcd_gclk_ctrl : // unsigned ,RW , default = 0 mali_afbcd_gclk_ctrl
8	R/W	1	mali_afbcd_data_128b : // unsigned ,RW , default = 1 mali_afbcd_data_128b

Bit(s)	R/W	Default	Description
6:4	R/W	7	mali_afbcd_max_len : // unsigned ,RW , default = 7 mali_afbcd_max_len
2:0	R/W	5	mali_afbcd_axi_outstanding_trans : // unsigned ,RW , default = 5 mali_afbcd_axi_outstanding_trans

Table 9-1422 MALI\_AFBCD1\_TOP\_CTRL 1a55

Bit(s)	R/W	Default	Description
23	R/W	0	mali_afbcd1_sw_rst_sel : // unsigned ,RW , default = 0 1:mali_afbcd1_sw_reset [0] 2:go_field
22	R/W	0	reg_osd3_secure : // unsigned ,RW , default = 0 osd_axi secure bit
21	R/W	0	reg_osd3_axi_sel : // unsigned ,RW , default = 0 1:osd read mali_afbcd_data 0: osd read mif data
10	R/W	0	osd3_din_ext_mode : // unsigned ,RW , default = 0 1:add two bits 0 at high bits 0: add two bits 0 at low bits
19	R/W	0	osd3_dolby_bypass_en : // unsigned ,RW , default = 0 1:data bypass_osd3_dolby 0:data into_osd3_dolby
13:12	R/W	0	mali_afbcd1_gclk_ctrl : // unsigned ,RW , default = 0 mali_afbcd1_gclk_ctrl
8	R/W	1	mali_afbcd1_data_128b : // unsigned ,RW , default = 1 mali_afbcd1_data_128b
6:4	R/W	7	mali_afbcd1_max_len : // unsigned ,RW , default = 7 mali_afbcd1_max_len
2:0	R/W	5	mali_afbcd1_axi_outstanding_trans : // unsigned ,RW , default = 5 mali_afbcd1_axi_outstanding_trans

Table 9-1423 MALI\_AFBCD2\_TOP\_CTRL 1a56

Bit(s)	R/W	Default	Description
23	R/W	0	mali_afbcd2_sw_rst_sel : // unsigned ,RW , default = 0 1:mali_afbcd2_sw_reset [0] 2:go_field
22	R/W	0	reg_osd4_secure : // unsigned ,RW , default = 0 osd_axi secure bit
21	R/W	0	reg_osd4_axi_sel : // unsigned ,RW , default = 0 1:osd read mali_afbcd_data 0: osd read mif data
10	R/W	0	osd4_din_ext_mode : // unsigned ,RW , default = 0 1:add two bits 0 at high bits 0: add two bits 0 at low bits
19	R/W	0	osd4_dolby_bypass_en : // unsigned ,RW , default = 0 1:data bypass_osd4_dolby 0:data into_osd4_dolby
13:12	R/W	0	mali_afbcd2_gclk_ctrl : // unsigned ,RW , default = 0 mali_afbcd2_gclk_ctrl
8	R/W	1	mali_afbcd2_data_128b : // unsigned ,RW , default = 1 mali_afbcd2_data_128b
6:4	R/W	7	mali_afbcd2_max_len : // unsigned ,RW , default = 7 mali_afbcd2_max_len
2:0	R/W	5	mali_afbcd2_axi_outstanding_trans : // unsigned ,RW , default = 5 mali_afbcd2_axi_outstanding_trans

**Table 9-1424 PATH\_START\_SEL 1a8a**

Bit(s)	R/W	Default	Description
29:28	R/W	2	osd3_path_start_sel : //unsigned,RW ,default = 2 ,0:use vpp0_go_field 1:use vpp1_go_field 2:use vpp2_go_field
25:24	R/W	1	osd2_path_start_sel : //unsigned,RW ,default = 1 ,0:use vpp0_go_field 1:use vpp1_go_field 2:use vpp2_go_field
9 :8	R/W	2	vd3_path_start_sel : //unsigned,RW ,default = 2 ,0:use vpp0_go_field 1:use vpp1_go_field 2:use vpp2_go_field
5 :4	R/W	1	vd2_path_start_sel : //unsigned,RW ,default = 1 ,0:use vpp0_go_field 1:use vpp1_go_field 2:use vpp2_go_field

**Table 9-1425 VIU\_SECURE\_DUMMY 1a08**

Bit(s)	R/W	Default	Description
29:0	R/W	1	sec_dummy_data : //unsigned,RW ,default = 1 sec_dummy_data

**Table 9-1426 VIU\_FRM\_CTRL 1a51**

Bit(s)	R/W	Default	Description
29	R/W	0	viu0_frm_start : //unsigned,RW ,default = 0 pulse start vpp0 by write 1
26	R/W	0	viu0_chk_dout_sec : //unsigned,RW ,default = 0
19:17	R/W	0	viu0_frm_phs_mode : //unsigned,RW ,default = 0 1:auto start vpp0 by max{viu0_hold_line_num,module_hold_line_num}
16	R/W	0	viu0_frm_start_sel : //unsigned,RW ,default = 0 1:auto start vpp0 by hold_line_numhold_line_num 0:pulse start vpp0
12:0	R/W	4	viu0_hold_line_num : //unsigned,RW ,default = 4

**Table 9-1427 VIU1\_FRM\_CTRL 1a8d**

Bit(s)	R/W	Default	Description
29	R/W	0	viu1_frm_start : //unsigned,RW ,default = 0 //pulse start vpp1 by write 1
26	R/W	0	viu1_chk_dout_sec : //unsigned,RW ,default = 0
19:17	R/W	0	viu1_frm_phs_mode : //unsigned,RW ,default = 0 1:auto start vpp1 by max{viu1_hold_line_num,module_hold_line_num}
16	R/W	0	viu1_frm_start_sel : //unsigned,RW ,default = 0 1:auto start vpp1 by hold_line_numhold_line_num 0:pulse start vpp1
12:0	R/W	4	viu1_hold_line_num : //unsigned,RW ,default = 4

**Table 9-1428 VIU2\_FRM\_CTRL 1a8e**

Bit(s)	R/W	Default	Description
29	R/W	0	viu2_frm_start : //unsigned,RW ,default = 0 //pulse start vpp1 by write 1
26	R/W	0	viu2_chk_dout_sec : //unsigned,RW ,default = 0

Bit(s)	R/W	Default	Description
19:17	R/W	0	viu2_frm_phs_mode : //unsigned,RW ,default = 0 1:auto start vpp1 by max{viu2_hold_line_num,module_hold_line_num}
16	R/W	0	viu2_frm_start_sel : //unsigned,RW ,default = 0 1:auto start vpp1 by hold_line_numhold_line_num 0:pulse start vpp2
12:0	R/W	4	viu2_hold_line_num : //unsigned,RW ,default = 4

Table 9-1429 VIU\_MISC\_CTRL1 1a07

Bit(s)	R/W	Default	Description
6	R/W	0	reg_viu2_cfg_err_clr : //unsigned,RW ,default = 0 //
5	R/W	0	reg_viu1_cfg_err_clr : //unsigned,RW ,default = 0 //
4	R/W	0	reg_viu0_cfg_err_clr : //unsigned,RW ,default = 0 //
2	R/W	0	reg_viu2_cfg_done : //unsigned,RW ,default = 0 //
1	R/W	0	reg_viu1_cfg_done : //unsigned,RW ,default = 0 //
0	R/W	0	reg_viu0_cfg_done : //unsigned,RW ,default = 0 //

Table 9-1430 VIU\_SECURE\_REG 1a04

Bit(s)	R/W	Default	Description
31	R/W	0x0	osd1_dolby_secure :
30	R/W	0	core1c_secure : //unsigned,RW ,default = 0
29	R/W	0	core1b_secure : //unsigned,RW ,default = 0
28	R/W	0	dolbytv_secure : //unsigned,RW ,default = 0
27	R/W	0	core1a_secure : //unsigned,RW ,default = 0
26	R/W	0	primesl_secure : //unsigned,RW ,default = 0
25	R.O	0	probe_secure : //unsigned,RW ,default = 0
24	R/W	0	vks_secure : //unsigned,RW ,default = 0
22	R/W	0	mali_afbcd2_cbus_secure : //unsigned,RW ,default = 0
21	R/W	0	mali_afbcd1_cbus_secure : //unsigned,RW ,default = 0
20	R/W	0	mali_afbcd_cbus_secure : //unsigned,RW ,default = 0
19	R/W	0	reg_osd4_cbus_secure : //unsigned,RW ,default = 0
18	R/W	0	reg_osd3_cbus_secure : //unsigned,RW ,default = 0
17	R/W	0	reg_osd2_cbus_secure : //unsigned,RW ,default = 0
16	R/W	0	reg_osd1_cbus_secure : //unsigned,RW ,default = 0
15	R/W	0	osd4_dolby_secure : //unsigned,RW ,default = 0
14	R/W	0	osd3_dolby_secure : //unsigned,RW ,default = 0
13	R/W	0	osd2_dolby_secure : //unsigned,RW ,default = 0
11	R/W	0	wrbak3_secure : //unsigned,RW ,default = 0



Table 9-1435 VIU\_GCLK\_CTRL 0x1a4f

Bit(s)	R/W	Default	Description
7:0	R/W	0	viu_gclk_ctrl : // unsigned , default =0

### 9.2.3.17 VPP\_ARB\_CTRL Registers

Table 9-1436 VPP\_AFBC\_RDARB\_MODE 0x3970

Bit(s)	R/W	Default	Description
31-24	R/W	0	unused
23-20	R/W	0	Rdarb_sel
16	R/W	0	Rdarb_arb_mode
1-0	R/W	0	Rdarb_gate_clk_ctrl

Table 9-1437 VPP\_AFBC\_RDARB\_REQEN\_SLV 0x3971

Bit(s)	R/W	Default	Description
3-0	R/W	0	Rdarb_dc_req_en 1: request is valid, 0: invalid Bit0 : osd1 Bit1 : osd2 Bit2 : osd3 Bit3 : osd4

Table 9-1438 VPP\_AFBC\_RDARB\_WEIGHT0\_SLV 0x3972

Bit(s)	R/W	Default	Description
23-0	R/W	0	Rddc_weigh_sxn

Table 9-1439 VPP\_AFBC\_ARB\_DBG\_CTRL 0x3974

Bit(s)	R/W	Default	Description
31-0	RO	0	Det_cmd_ctrl

Table 9-1440 VPP\_RDARB\_MODE 0x3978

Bit(s)	R/W	Default	Description
27-20	R/W	0	Rdarb_sel 1: request is valid, 0: invalid Bit20: osd1 Bit21: osd2 Bit22: vd1 Bit23: vd2 Bit24: osd3 Bit25: osd4 Bit26: dolby0 Bit27:mali_afbc
17:16	R/W	0	Rdarb_arb_mode
3-0	R/W	0	Rdarb_gate_clk_ctrl

Table 9-1441 VPP\_RDARB\_REQEN\_SLV 0x3979

Bit(s)	R/W	Default	Description
15-0	R/W	0	Rdarb_dc_req_en 1: request is valid, 0: invalid Bit0: osd1 come from axi_port0 Bit1: osd2 come from axi_port0 Bit2: vd1 come from axi_port0 Bit3: vd2 come from axi_port0 Bit4: osd3 come from axi_port0 Bit5: osd4 come from axi_port0 Bit6: dolby0 come from axi_port0 Bit7: mali_afbc come from axi_port0 Bit8: osd1 come from axi_port1 Bit9: osd2 come from axi_port1 Bit10: vd1 come from axi_port1 Bit11: vd2 come from axi_port1 Bit12: osd3 come from axi_port1 Bit13: osd4 come from axi_port1 Bit14: dolby0 come from axi_port1 Bit15: mali_afbc come from axi_port1

Table 9-1442 VPP\_RDARB\_WEIGHT0\_SLV 0x397a

Bit(s)	R/W	Default	Description
29-0	R/W	0	Rddc_weigh_sxn

Table 9-1443 VPP\_RDARB\_WEIGHT1\_SLV 0x397b

Bit(s)	R/W	Default	Description
17-0	RO	0	Rddc_weigh_sxn

Table 9-1444 VPP\_ARB\_DBG\_CTRL 0x397c

Bit(s)	R/W	Default	Description
31-0	RO	0	Det_cmd_ctrl

### 9.2.3.18 Osd\_blend Registers

Table 9-1445 VIU\_OSD\_BLEND\_CTRL 0x39b0

Bit(s)	R/W	Default	Description
31:29	R/W	0x0	hold_lines : //unsigned , default = 3'h0, hold_lines(line) after go_field ,module active
28:27	R/W	0x3	blend2_premult_en : //unsigned , default = 2'h3, blend2 input premult label 1:premult input 0:unpremult input
26	R/W	0x1	din0_byp_blend : //unsigned , default = 1'h1, blend_din0 bypass to dout0 1:bypass 0:blend_din0 input to blend0
25	R/W	0x1	din2_osd_sel : //unsigned , default = 1'h1, blend1_dout bypass to blend2 0:blend1_dout to blend2 1:blend1_dout to dout1

Bit(s)	R/W	Default	Description
24	R/W	0x1	din3_osd_sel : //unsigned , default = 1'h1, blend1_din3 bypass to dout1 1:bypass 0:blend_din3 input to blend1
23:20	R/W	0x5	blend_din_en : //unsigned , default = 4'h5, blend enable bits ,four bits for four input
19:16	R/W	0x0	din_premult_en : //unsigned , default = 4'h0, input premult label bits,four bits for four input
15:0	R/W	0x2341	din_reoder_sel : //unsigned , default = 16'h2341,osd_blend input reorder exp : din_reoder_sel[3:0] = 1 ,blend_din0 select osd1 din_reoder_sel [3:0] = 2 ,blend_din0 select osd2 din_reoder_sel [3:0] = 3 ,blend_din0 select osd3 din_reoder_sel [3:0] = else,blend_din0 no src din_reoder_sel [7:4] fot blend_din1

Table 9-1446 VIU\_OSD\_BLEND\_CTRL1 0x39c0

Bit(s)	R/W	Default	Description
17:16	R/W	0	reg_alp1_mapping_mode : //unsigned , default = 0 , osd_blend dout1 alpha divisor mode 8bit alpha:set 0 9bit alpha:set 3
14:13	R/W	0	reg_div1_gclk_en : //unsigned , default = 0 , osd_blend dout1 alpha divisor gclk_en
12	R/W	0	reg_div1_alpha_en : //unsigned , default = 0 , osd_blend dout1 alpha divisor gclk_en enable
9:8	R/W	0	osdbld_gclk_ctrl : //unsigned , default = 0 , osdbld_gclk_ctrl
5:4	R/W	0	reg_alp_mapping_mode : //unsigned , default = 0 , osd_blend dout0 alpha divisor mode 8bit alpha:set 0 9bit alpha:set 3
2:1	R/W	0	reg_div_gclk_en : //unsigned , default = 0 , osd_blend dout0 alpha divisor gclk_en
0	R/W	0	reg_div_alpha_en : //unsigned , default = 0 , osd_blend dout0 alpha divisor gclk_en enable

Table 9-1447 VIU\_OSD\_BLEND\_DIN0\_SCOPE\_H 0x39b1

Bit(s)	R/W	Default	Description
28:16	R/W	0x2d0	bld_din0_h_end : ///unsigned , default = 13'h2d0,blend_din0 h_end
12:0	R/W	0x0	bld_din0_h_start : ///unsigned , default = 13'h0 ,blend_din0 h_start

Table 9-1448 VIU\_OSD\_BLEND\_DIN0\_SCOPE\_V 0x39b2

Bit(s)	R/W	Default	Description
28:16	R/W	0x1e0	bld_din0_v_end : ///unsigned , default = 13'h1e0,blend_din0 v_end
12:0	R/W	0x0	bld_din0_v_start : ///unsigned , default = 13'h0,blend_din0 v_start

Table 9-1449 VIU\_OSD\_BLEND\_DIN1\_SCOPE\_H 0x39b3

Bit(s)	R/W	Default	Description
28:16	R/W	0x2d0	bld_din1_h_end : ///unsigned , default = 13'h2d0
12:0	R/W	0x0	bld_din1_h_start : ///unsigned , default = 13'h0



**Table 9-1450 VIU\_OSD\_BLEND\_DIN1\_SCOPE\_V 0x39b4**

Bit(s)	R/W	Default	Description
28:16	R/W	0x1e0	bld_din1_v_end : ///unsigned , default = 13'h1e0
12:0	R/W	0x0	bld_din1_v_start : ///unsigned , default = 13'h0

**Table 9-1451 VIU\_OSD\_BLEND\_DIN2\_SCOPE\_H 0x39b5**

Bit(s)	R/W	Default	Description
28:16	R/W	0x2d0	bld_din2_h_end : ///unsigned , default = 13'h2d0
12:0	R/W	0x0	bld_din2_h_start : ///unsigned , default = 13'h0

**Table 9-1452 VIU\_OSD\_BLEND\_DIN2\_SCOPE\_V 0x39b6**

Bit(s)	R/W	Default	Description
28:16	R/W	0x1e0	bld_din2_v_end : ///unsigned , default = 13'h1e0
12:0	R/W	0x0	bld_din2_v_start : ///unsigned , default = 13'h0

**Table 9-1453 VIU\_OSD\_BLEND\_DIN3\_SCOPE\_H 0x39b7**

Bit(s)	R/W	Default	Description
28:16	R/W	0x2d0	bld_din3_h_end : ///unsigned , default = 13'h2d0
12:0	R/W	0x0	bld_din3_h_start : ///unsigned , default = 13'h0

**Table 9-1454 VIU\_OSD\_BLEND\_DIN3\_SCOPE\_V 0x39b8**

Bit(s)	R/W	Default	Description
28:16	R/W	0x1e0	bld_din3_v_end : ///unsigned , default = 13'h1e0
12:0	R/W	0x0	bld_din3_v_start : ///unsigned , default = 13'h0

**Table 9-1455 VIU\_OSD\_BLEND\_DUMMY\_DATA0 0x39b9**

Bit(s)	R/W	Default	Description
23:16	R/W	0x00	blend0_dummy_data_y : //unsigned , default = 8'h00
15:8	R/W	0x80	blend0_dummy_data_cb : //unsigned , default = 8'h80
7:0	R/W	0x80	blend0_dummy_data_cr : //unsigned , default = 8'h80

**Table 9-1456 VIU\_OSD\_BLEND\_DUMMY\_ALPHA 0x39ba**

Bit(s)	R/W	Default	Description
28:20	R/W	0x0	blend0_dummy_alpha : //unsigned , default = 9'h0
19:11	R/W	0x0	blend1_dummy_alpha : //unsigned , default = 9'h0
8:0	R/W	0x0	blend2_dummy_alpha : //unsigned , default = 9'h0

**Table 9-1457 VIU\_OSD\_BLEND\_BLEND0\_SIZE 0x39bb**

Bit(s)	R/W	Default	Description
28:16	R/W	0x1e0	blend0_vsize : //unsigned , default = 13'h1e0,blend0_vsize
12:0	R/W	0x2d0	blend0_hsize : //unsigned , default = 13'h2d0,blend0_hsize

**Table 9-1458 VIU\_OSD\_BLEND\_BLEND1\_SIZE 0x39bc**

Bit(s)	R/W	Default	Description
28:16	R/W	0x1e0	blend1_vsize : //unsigned , default = 13'h1e0
12:0	R/W	0x2d0	blend1_hsize : //unsigned , default = 13'h2d0

### 9.2.3.19 Pre/Post Blend Registers

**Table 9-1459 VPP\_PRE\_BLEND\_CTRL 0x3960**

Bit(s)	R/W	Default	Description
27:20	R/W	0x4	hold_lines : //unsigned , default = 8'h4,blend work after hold_lines line after go_ field
1:0	R/W	0x1	gclk_ctrl : //unsigned , default = 16'h1,gating ctrl

**Table 9-1460 VPP\_PRE\_BLEND\_BLEND\_DUMMY\_DATA 0x3961**

Bit(s)	R/W	Default	Description
23:16	R/W	0x0	blend0_dummy_data_y : //unsigned , default = 8'h0,blend dummy data
15:8	R/W	0x80	blend0_dummy_data_cb : //unsigned , default = 8'h80,blend dummy data
7:0	R/W	0x80	blend0_dummy_data_cr : //unsigned , default = 8'h80 ,blend dummy data

**Table 9-1461 VPP\_PRE\_BLEND\_DUMMY\_ALPHA 0x3962**

Bit(s)	R/W	Default	Description
28:20	R/W	0x0	blend0_dummy_alpha : //unsigned , default = 9'h0,blend dummy alpha
19:11	R/W	0x0	blend1_dummy_alpha : //unsigned , default = 9'h0,blend dummy alpha
8:0	R/W	0x0	blend2_dummy_alpha : //unsigned , default = 9'h0,blend dummy alpha

**Table 9-1462 VPP\_PRE\_BLEND2\_RO\_CURRENT\_XY 0x3963**

Bit(s)	R/W	Default	Description
28:16	R.O	0x0	ro_current_x : //unsigned , default = 32'h0,blend out x point
12:0	R.O	0x0	ro_current_y : //unsigned , default = 32'h0,blend out x point

**Table 9-1463 VPP\_POST\_PRE\_BLEND\_CTRL 0x3967**

Bit(s)	R/W	Default	Description
27:20	R/W	0x4	hold_lines : //unsigned , default = 8'h4,blend work after hold_lines line after go_field
1:0	R/W	0x1	gclk_ctrl : //unsigned , default = 16'h1,gating ctrl

**Table 9-1464 VPP\_POST\_BLEND\_BLEND\_DUMMY\_DATA 0x3968**

Bit(s)	R/W	Default	Description
23:16	R/W	0x0	blend0_dummy_data_y : //unsigned , default = 8'h0,blend dummy data
15:8	R/W	0x80	blend0_dummy_data_cb : //unsigned , default = 8'h80,blend dummy data
7:0	R/W	0x80	blend0_dummy_data_cr : //unsigned , default = 8'h80 ,blend dummy data

**Table 9-1465 VPP\_POST-BLEND\_DUMMY\_ALPHA 0x3969**

Bit(s)	R/W	Default	Description
28:20	R/W	0x0	blend0_dummy_alpha : //unsigned , default = 9'h0,blend dummy alpha
19:11	R/W	0x0	blend1_dummy_alpha : //unsigned , default = 9'h0,blend dummy alpha
8:0	R/W	0x0	blend2_dummy_alpha : //unsigned , default = 9'h0,blend dummy alpha

**Table 9-1466 VPP\_POST\_BLEND2\_RO\_CURRENT\_XY 0x396a**

Bit(s)	R/W	Default	Description
28:16	R.O	0x0	ro_current_x : //unsigned , default = 32'h0,blend out x point
12:0	R.O	0x0	ro_current_y : //unsigned , default = 32'h0,blend out x point

### 9.2.3.20 VPP Registers

**Table 9-1467 VPP\_DUMMY\_DATA 0x1d00**

Bit(s)	R/W	Default	Description
29:20	R/W	0	VD1_SC_Y : // unsigned , default = 0x10,dummy data used in the VD1 scaler,according VPP_DOLBY_CTRL[17] 1:set 8bit value 2:set 10bit value
19:10	R/W	0	VD1_SC_CB : // unsigned , default = 0x80,dummy data used in the VD1 scaler,according VPP_DOLBY_CTRL[17] 1:set 8bit value 2:set 10bit value
9 :0	R/W	0	VD1_SC_CR : // unsigned , default = 0x80,dummy data used in the VD1 scaler,according VPP_DOLBY_CTRL[17] 1:set 8bit value 2:set 10bit value

**Table 9-1468 VPP\_LINE\_IN\_LENGTH 0x1d01**

Bit(s)	R/W	Default	Description
13:0	R/W	14	line_in_length : // unsigned , default = 14'd1920,VD1 scaler input hsize

**Table 9-1469 VPP\_PIC\_IN\_HEIGHT 0x1d02**

Bit(s)	R/W	Default	Description
12:0	R/W	0x1fff	line_in_height : // unsigned , default = 13'h1fff,VD1 scaler input vsize

**Table 9-1470 VPP\_PREBLEND\_VD1\_H\_START\_END 0x1d1a**

Bit(s)	R/W	Default	Description
28:16	R/W	0	start : //unsigned , default = 0x0000 ,preblend video1 horizontal start
12:0	R/W	0	end : //unsigned , default = 0x077f ,preblend video1 horizontal end

**Table 9-1471 VPP\_PREBLEND\_VD1\_V\_START\_END 0x1d1b**

Bit(s)	R/W	Default	Description
28:16	R/W	0	start : //unsigned , default = 0x0000 ,preblend video1 vertical start
12:0	R/W	0	end : //unsigned , default = 0x0437 ,preblend video1 vertical end

**Table 9-1472 VPP\_POSTBLEND\_VD1\_H\_START\_END 0x1d1c**

Bit(s)	R/W	Default	Description
28:16	R/W	0	start : //unsigned , default = 0x0000 ,postblend video1 horizontal start
12:0	R/W	0	end : //unsigned , default = 0x077f ,postblend video1 horizontal end

**Table 9-1473 VPP\_POSTBLEND\_VD1\_V\_START\_END 0x1d1d**

Bit(s)	R/W	Default	Description
31:29	/	/	/
28:16	R/W	0	//unsigned , default = 0x0000 ,postblend video1 vertical start Bit
15:13	/	/	/
12:0		0x077f	//unsigned , default = 0x077f ,postblend video1 vertical end

**Table 9-1474 VPP\_BLEND\_VD2\_H\_START\_END 0x1d1e**

Bit(s)	R/W	Default	Description
28:16	R/W	0	start : //unsigned , default = 0x0000 ,preblend/postblend video2 horizontal start
12:0	R/W	0	end : //unsigned , default = 0x077f ,preblend/postblend video2 horizontal end

**Table 9-1475 VPP\_BLEND\_VD2\_V\_START\_END 0x1d1f**

Bit(s)	R/W	Default	Description
28:16	R/W	0	start : //unsigned , default = 0x0000 ,preblend/postblend video2 vertical start
12:0	R/W	0	end : //unsigned , default = 0x077f ,preblend/postblend video2 vertical end

**Table 9-1476 VPP\_PREBLEND\_H\_SIZE 0x1d20**

Bit(s)	R/W	Default	Description
29:16	R/W	14	prebld_v_size : //unsigned , default = 14'd1080 ,preblend output vsize
13:0	R/W	14	prebld_h_size : //unsigned , default = 14'd1920 ,preblend output hsize

**Table 9-1477 VPP\_POSTBLEND\_H\_SIZE 0x1d21**

Bit(s)	R/W	Default	Description
29:16	R/W	14	postbld_v_size : //unsigned , default = 14'd1080 ,postblend output vsize
13:0	R/W	14	postbld_h_size : //unsigned , default = 14'd1920 ,postblend output hsize

**Table 9-1478 VPP hold lines VPP\_HOLD\_LINES 0x1d22**

Bit(s)	R/W	Default	Description
15:8	R/W	4	prebld_hold_lines : //unsigned , default = 4,preblend hold lines
7:0	R/W	4	postbld_hold_lines : //unsigned , default = 4,postblend hold lines

**Table 9-1479 VPP\_MISC 0x1d26**

Bit(s)	R/W	Default	Description
30	R/W	0	Color_manage_en //CM2 top enable
29	R/W	0	vpp_vks_en : //unsigned , default = 0 , vkstone enable
28	R/W	0	color_manage_en : //unsigned , default = 0 , color management enable
27	R/W	0	vd2_use_viu2_out_en : //unsigned , default = 0 , if true, vd2 use viu2 output as the input, otherwise use normal vd2 from memory
26:18	R/W	0	vd2 : alpha //unsigned , default = 0 , video 2 prebld/postbld alpha
8	R/W	1	0: vpp_gamma before 3D_LUT 1: gammar after 3D_lut
7	R/W	1	postbld_en : //unsigned , default = 1 , postblend module enable ,din_reoder_sel of post blend can work when postbld_en high.
6	R/W	0	prebld_en : //unsigned , default = 0 , preblend module enable ,din_reoder_sel of post blend can work when postbld_en high.
3	R/W	0	sr4c1_path_sel : //unsigned , default = 0 , choose sr1 position 1:sr1 bettween dnlp & CM 0:sr1 bettween position after postblend
2	R/W	0	disable_rst_afifo : //unsigned , default = 0 , if true, disable resetting async fifo every vsync, otherwise every vsync the aync fifo will be reseted.
1	R/W	0	sr4c0_path_sel : //unsigned , default = 0 , choose sr0 position 1:sr0 bettween prebld & vd1_scale 0:sr0 bettween position after dnlp

**Table 9-1480 VPP\_OFIFO\_SIZE 0x1d27**

Bit(s)	R/W	Default	Description
31:20	R/W	0	ofifo_line_lenm1 : //unsigned , default = 0xffff , ofifo line length minus 1
19	R/W	0	vs_ctrl : //unsigned , default = 0 , if true invert input vs
18	R/W	0	hs_ctrl : //unsigned , default = 0 , if true invert input hs
17	R/W	0	force_top_bot_field_en : //unsigned , default = 0 , force top/bottom field, enable
16	R/W	0	fforce_top_bot_field : //unsigned , default = 0 , force top/bottom field, 0: top, 1: bottom
15	R/W	0	force_go_field : //unsigned , default = 0 , force one go_field, one pluse, write only
14	R/W	0	force_go_line : //unsigned , default = 0 , force one go_line, one pluse, write only
13:0	R/W	0	ofifo_size : //unsigned , default = 0x1000 , ofifo size (actually only bit 13:1 is valid), always even number ,max 4096

**Table 9-1481 VPP\_FIFO\_STATUS 0x1d28**

Bit(s)	R/W	Default	Description
29:25	R.O	0	ro_sco_ff_buf_count : //unsigned , default = 0,current scale out fifo counter
24:14	R.O	0	ro_afifo_count : //unsigned , default = 0,current enc afifo counter
13:1	R.O	0	ro_ofifo_buf_count : //unsigned , default = 0,current vpp line fifo ofifo counter

**Table 9-1482 VPP\_MATRIX\_PROBE\_COLOR 0x1d5c**

Bit(s)	R/W	Default	Description
31:24	R.O	0	ro_probe_y_l : //unsigned , default = 0,low 8 bits of component 0
23:12	R.O	0	ro_probe_cr : //unsigned , default = 0,component 1
11:0	R.O	0	ro_probe_cb : //unsigned , default = 0,component 2

**Table 9-1483 VPP\_MATRIX\_PROBE\_COLOR1 0x1dd7**

Bit(s)	R/W	Default	Description
31	R.O	0x0	ro_probe_pix_v : //it means this probe is valid in the last field/frame
3:0	R.O	0	ro_probe_y_h : //unsigned , default = 0,high 4 bits of component 0

**Table 9-1484 VPP\_MATRIX\_HL\_COLOR 0x1d5d**

Bit(s)	R/W	Default	Description
23:16	R/W	0	hl_y : //unsigned , default = 0,component 0
15:8	R/W	0	hl_cb : //unsigned , default = 0,component 1
7:0	R/W	0	hl_cr : //unsigned , default = 0,component 2

**Table 9-1485 VPP\_MATRIX\_PROBE\_POS 0x1d5e**

Bit(s)	R/W	Default	Description
28:16	R/W	0	probe_x : //unsigned , default = 0,probe x, position
12:0	R/W	0	probe_y : //unsigned , default = 0,probe y, position

**Table 9-1486 VPP\_MATRIX\_CTRL 0x1d5f**

Bit(s)	R/W	Default	Description
16	R/W	0	highlight_en : //unsigned , default = 0, highlight enable
15	R/W	0	probe_post : //unsigned , default = 0, if true, probe pixel data after matrix, otherwise probe pixel data before matrix
14:10	R/W	0	probel_sel : //unsigned , default = 0, active when VIU_SECURE_REG[9] high, probel sel : 5'b00001 :vadj1 5'b00010 :vadj2 5'b00100 :osd2 5'b01000 :postbld 5'b10000 :osd1

**Table 9-1487 VPP\_GAINOFF\_CTRL0 0x1d6a**

Bit(s)	R/W	Default	Description
31	R/W	0	enable : //unsigned , default = 0, gainoff module enable
30	R/W	0	enable_sel : //unsigned , default = 0, gainoff module enable sync sel
26:16	R/W	0	gain0 : //unsigned , default = 0, gainoff module gain0
10:0	R/W	0	gain1 : //unsigned , default = 0, gainoff module gain1

**Table 9-1488 VPP\_GAINOFF\_CTRL1 0x1d6b**

Bit(s)	R/W	Default	Description
26:16	R/W	0	gain2 : //unsigned , default = 0, gainoff module gain2
(DW +0):0	R/W	0	offset0 : //signed ,bitwidth is DW+1,DW is the chip path data width, default = 0, gainoff module offset0

**Table 9-1489 VPP\_GAINOFF\_CTRL2 0x1d6c**

Bit(s)	R/W	Default	Description
(DW +16):16	R/W	0	offset1 : //signed ,bitwidth is DW+1,DW is the chip path data width, default = 0, gainoff module offset1
(DW +0):0	R/W	0	offset2 : //signed , bitwidth is DW+1,DW is the chip path data width,default = 0, gainoff module offset2

**Table 9-1490 VPP\_GAINOFF\_CTRL3 0x1d6d**

Bit(s)	R/W	Default	Description
(DW+16):16	R/W	0	pre_offset0 : //signed , bitwidth is DW+1,DW is the chip path data width,default = 0, gainoff module pre_offset0
(DW+0):0	R/W	0	pre_offset1 : //signed , bitwidth is DW+1,DW is the chip path data width,default = 0, gainoff module pre_offset1

**Table 9-1491 VPP\_GAINOFF\_CTRL4 0x1d6e**

Bit(s)	R/W	Default	Description
(DW+0):0	R/W	0	pre_offset2 : //signed , bitwidth is DW+1,DW is the chip path data width,default = 0, gainoff module pre_offset2

**Table 9-1492 VPP\_GAINOFF\_GCLK\_CTRL 0x1d6f**

Bit(s)	R/W	Default	Description
1:0	R/W	0	gainoff_gclk_ctrl : //unsigned , default = 0,gainoff_gclk_ctrl

**Table 9-1493 VPP\_GCLK\_CTRL0 0x1d72**

Bit(s)	R/W	Default	Description
5:4	R/W	0	offo_clk1 : //unsigned , default = 0,gating clock of out linefifo in vpp
3:2	R/W	0	clk0 : //unsigned , default = 0,clk swtich of all vpp module
1	R/W	0	reg_gclk : //unsigned , default = 0,registers gate clk of vpp module

**Table 9-1494 VPP\_GCLK\_CTRL1 0x1d73**

Bit(s)	R/W	Default	Description
27:20	R/W	0	gclk_ctrl_wm : //unsigned , default = 0,gating clock of water_mark
18:15	R/W	0	dolby3_gclk_ctrl : //unsigned , default = 0,gating clock of dolby3
3:0	R/W	0	cm_gclk_ctrl : //unsigned , default = 0,gating clock of color manage

**Table 9-1495 VPP\_MISC1 0x1d76**

Bit(s)	R/W	Default	Description
20:12	R/W	0	vd1_prebld_alpha : //unsigned , default = 0,VD1 alpha for preblend
8:0	R/W	0	vd1_postbld_alpha : //unsigned , default = 0,VD1 alpha for postblend

**Table 9-1496 VPP\_SRSC\_L\_GCLK\_CTRL 0x1d77**

Bit(s)	R/W	Default	Description
15:8	R/W	0	gclk_ctrl_sr1 : //unsigned , default = 0 , gating clock of sr1
7:0	R/W	0	gclk_ctrl_sr0 : //unsigned , default = 0 , gating clock of sr0



**Table 9-1497 VPP\_BLACKEXT\_CTRL 0x1d80**

Bit(s)	R/W	Default	Description
31:24	R/W	0	blackext_start : //unsigned , default = 0 ,blackext_start
23:16	R/W	0	blackext_slope1 : //unsigned , default = 0 ,blackext_slope1
15:8	R/W	0	blackext_midpt : //unsigned , default = 0 ,blackext_midpt
7:0	R/W	0	blackext_slope2 : //unsigned , default = 0 ,blackext_slope2

**Table 9-1498 VPP\_DNLP\_CTRL\_00 0x1d81**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region03 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region02 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region01 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region00 output value

**Table 9-1499 VPP\_DNLP\_CTRL\_01 0x1d82**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region07 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region06 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region05 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region04 output value

**Table 9-1500 VPP\_DNLP\_CTRL\_02 0x1d83**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region11 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region10 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region09 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region08 output value

**Table 9-1501 VPP\_DNLP\_CTRL\_03 0x1d84**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region15 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region14 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region13 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region12 output value

**Table 9-1502 VPP\_DNLP\_CTRL\_04 0x1d85**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region19 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region18 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region17 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region16 output value

**Table 9-1503 VPP\_DNLP\_CTRL\_05 0x1d86**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region23 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region22 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region21 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region20 output value

**Table 9-1504 VPP\_DNLP\_CTRL\_06 0x1d87**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region27 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region26 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region25 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region24 output value

**Table 9-1505 VPP\_DNLP\_CTRL\_07 0x1d88**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region31 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region30 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region29 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region28 output value

**Table 9-1506 VPP\_DNLP\_CTRL\_08 0x1d89**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region35 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region34 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region33 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region32 output value

**Table 9-1507 VPP\_DNLP\_CTRL\_09 0x1d8a**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region39 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region38 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region37 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region36 output value

**Table 9-1508 VPP\_DNLP\_CTRL\_10 0x1d8b**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region43 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region42 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region41 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region40 output value

**Table 9-1509 VPP\_DNLP\_CTRL\_11 0x1d8c**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region47 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region46 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region45 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region44 output value

**Table 9-1510 VPP\_DNLP\_CTRL\_12 0x1d8d**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region51 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region50 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region49 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region48 output value

**Table 9-1511 VPP\_DNLP\_CTRL\_13 0x1d8e**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region55 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region54 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region53 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region52 output value

**Table 9-1512 VPP\_DNLP\_CTRL\_14 0x1d8f**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region59 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region58 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region57 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region56 output value

**Table 9-1513 VPP\_DNLP\_CTRL\_15 0x1d90**

Bit(s)	R/W	Default	Description
31:24	R/W	0	data0 : //unsigned , default = 0 ,bottom of region63 output value
23:16	R/W	0	data1 : //unsigned , default = 0 ,bottom of region62 output value
15:8	R/W	0	data2 : //unsigned , default = 0 ,bottom of region61 output value
7:0	R/W	0	data3 : //unsigned , default = 0 ,bottom of region60 output value

**Table 9-1514 VPP\_SRSHARP0\_CTRL 0x1d91**

Bit(s)	R/W	Default	Description
28:16	R/W	0	srsharp_demo_split_sz : //unsigned , default = 0 ,srsharp demo top/bot left/right width
5:4	R/W	0	srsharp_demo_disp_post : //unsigned , default = 0 ,srsharp demo display position
3	R/W	0	srsharp_demo_en : //unsigned , default = 0 ,srsharp demo enable
2	R/W	0	srsharp_c444to422_en : //unsigned , default = 0 ,srsharp format444 convert 422 enable
1	R/W	0	srsharp_buf_en : //unsigned , default = 0 ,srsharp buffer enable
0	R/W	0	srsharp_en : //unsigned , default = 0 ,srsharp enable

**Table 9-1515 VPP\_SRSHARP1\_CTRL 0x1d92**

Bit(s)	R/W	Default	Description
28:16	R/W	0	srsharp_demo_split_sz : //unsigned , default = 0 ,srsharp demo top/bot left/right width
5:4	R/W	0	srsharp_demo_disp_post : //unsigned , default = 0 ,srsharp demo display position
3	R/W	0	srsharp_demo_en : //unsigned , default = 0 ,srsharp demo enable
2	R/W	0	srsharp_c444to422_en : //unsigned , default = 0 ,srsharp format444 convert 422 enable
1	R/W	0	srsharp_buf_en : //unsigned , default = 0 ,srsharp buffer enable
0	R/W	0	srsharp_en : //unsigned , default = 0 ,srsharp enable

Table 9-1516 VPP\_DOLBY\_CTRL 0x1d93

Bit(s)	R/W	Default	Description
25:24	R/W	0	dolby3_path_sel : //unsigned , default = 0 , 1:dolby2->osd_mat->post_blend->dolby3->wm 0:dolby2->dolby3->osd_mat->post_blend->wm
20	R/W	0	Wap vpp vd1/vd1 0:unswap 1:swap
17	R/W	0	pps_dummy_data_mode : //unsigned , default = 0 ,pps_dummy_data_mode 1: vd1_scale need setting 8 bits 0:vd1_scale need setting 10 bits
16	/	/	reserved
10	R/W	0	vpp_clip_ext_mode2 : //unsigned , default = 0 ,Vpp out clip mode 1:10bit 0:12bit
9	R/W	0	vpp_clip_ext_mode1 : //unsigned , default = 0 ,Vpp Vd2 input clip mode 1:10bit 0:12bit
8	R/W	0	vpp_clip_ext_mode0 : //unsigned , default = 0 ,Vpp Vd1 input clip mode 1:10bit 0:12bit
3	R/W	0	vpp_dolby3_en : //unsigned , default = 0 ,dolby_core3 enable,active high
2	R/W	0	vpp_dpath_sel2 : //unsigned , default = 0 ,by_pass from post2_marix to outbuffer
1	R/W	0	vpp_dpath_sel1 : //unsigned , default = 0 ,by_pass from gainoff to vks
0	R/W	0	vpp_dpath_sel0 : //unsigned , default = 0 , 1: by_pass from preblend to VADJ1 0: unbypass

Table 9-1517 VPP\_SYNC\_SEL0 0x1d96

Bit(s)	R/W	Default	Description
31:0	R/W	0	sync_sel_bits : //unsigned ,default = 0,sync_sel bits for VPP_DOLBY_CTRL

Table 9-1518 VPP\_CCORING\_CTRL 0x1da0

Bit(s)	R/W	Default	Description
27:16	R/W	0	reg_bypass_ccoring_ythd : // unsigned , default = 0 , bypass_ccoring_ythd
15:8	R/W	0	ccoring_th : // unsigned , default = 0 , Chroma coring threshold
3:0	R/W	0	ccoring_slope : // unsigned , default = 0 , Chroma coring slope

Table 9-1519 VPP\_VE\_ENABLE\_CTRL 0x1da1

Bit(s)	R/W	Default	Description
29:28	R/W	0	dnlp_gclk_ctrl : // unsigned , default = 0 ,dnlp gclk ctrl
27:26	R/W	0	blackext_gclk_ctrl : // unsigned , default = 0 ,blackext gclk ctrl
25:24	R/W	0	ccoring_gclk_ctrl : // unsigned , default = 0 ,chroma coring gclk ctrl
20	R/W	0	demo_ccoring_enable : // unsigned , default = 0 ,demo chroma coring enable
19	R/W	0	demo_blackext_enable : // unsigned , default = 0 ,demo black extension enable
18	R/W	0	demo_dnlp_enable : // unsigned , default = 0 ,demo dynamic nonlinear luma processing enable
15:14	R/W	0	demo_disp_position : // unsigned , default = 0 ,2'b00: demo adjust on top, 2'b01: demo adjust on bottom, 2'b10: demo adjust on left, 2'b11: demo adjust on right

Bit(s)	R/W	Default	Description
4	R/W	0	ccoring_en : // unsigned , default = 0 , chroma coring enable
3	R/W	0	blackext_en : // unsigned , default = 0 , black enxtension enable
2	R/W	0	dnlp_en : // unsigned , default = 0 , dynamic nonlinear luma processing enable

Table 9-1520 VPP\_VE\_DEMO\_LEFT\_TOP\_SCREEN\_WIDTH 0x1da2

Bit(s)	R/W	Default	Description
12:0	R/W	0	ve_demo_left_top_screen_width : // unsigned , default = 0 demo left or top screen width

Table 9-1521 VPP\_VE\_DEMO\_CENTER\_BAR 0x1da3

Bit(s)	R/W	Default	Description
31	R/W	0	ve_demo_center_bar : // unsigned , default = 0 center bar enable
27:24	R/W	0	ve_demo_center_bar : // unsigned , default = 0 center bar width (*2)
23:16	R/W	0	ve_demo_center_bar : // unsigned , default = 0 center bar Cr (*4)
15:8	R/W	0	ve_demo_center_bar : // unsigned , default = 0 center bar Cb (*4)
7:0	R/W	0	ve_demo_center_bar : // unsigned , default = 0 center bar y (*4)

Table 9-1522 VPP\_VE\_H\_V\_SIZE 0x1da4

Bit(s)	R/W	Default	Description
28:16	R/W	780	ve_line_length : // unsigned , default = 780 ve_line_length
12:0	R/W	438	ve_pic_height : // unsigned , default = 438 ve_pic_height

Table 9-1523 VPP\_OUT\_H\_V\_SIZE 0x1da5

Bit(s)	R/W	Default	Description
28:16	R/W	780	vppout_line_length : / unsigned , default = 780 vd1_scale_out hsize
12:0	R/W	438	vppout_pic_height : // unsigned , default = 438 vd1_scale_out vsize

Table 9-1524 VPP\_VDO\_MEAS\_CTRL 0x1da8

Bit(s)	R/W	Default	Description
10:0	R/W	0	vdo_meas_ctrl : // unsigned , default = 0 vdo_meas_ctrl

**Table 9-1525 VPP\_VDO\_MEAS\_VS\_COUNT\_HI 0x1da9**

Bit(s)	R/W	Default	Description
19 :16	RO	0	ro_ind_meas_count_n // unsigned , default = 0 ind_meas_count_n, every number of sync_span vsyncs, this counter add 1
15 :0,	RO	0	ro_counter_h // unsigned , default = 0 high bit portion of counter

**Table 9-1526 VPP\_VDO\_MEAS\_VS\_COUNT\_LO 0x1daa**

Bit(s)	R/W	Default	Description
31 :0	RO	0	ro_counter_l // unsigned , default = 0, low bit portion of counter

**Table 9-1527 VPP\_INPUT\_CTRL 0x1dab**

Bit(s)	R/W	Default	Description
11:9	R/W	0	vd2_sel : // unsigned , default = 0, 001: select vd1_din, 010: select vd2_din, 011: select d2d3_l_din, 100: d2d3_r_din, otherwise no selection
8:6	R/W	0	vd1_l_sel : // unsigned , default = 0, 001: select vd1_din, 010: select vd2_din, 011: select d2d3_l_din, 100: d2d3_r_din, otherwise no selection, vd1_l_sel selected cannot be used as the source of vd1_r_sel or vd2_sel
5:3	R/W	0	vd1_r_sel : // unsigned , default = 0, 001: select vd1_din, 010: select vd2_din, 011: select d2d3_l_din, 100: d2d3_r_din, otherwise no selection, useful only vd1_interleave_mode is not 00. And the source vd1_r_sel used can not used for the vd2_sel any more bit 2:0 vd1_interleave_mode // unsigned , default = 0, 000: no interleave, 001: pixel interleaving, 010: line interleaving, 011: 2 pixel interleaving, 100: 2 line interleaving

**Table 9-1528 VPP\_CTI\_CTRL2 0x1dac**

Bit(s)	R/W	Default	Description
25:24	R/W	0	cti_bpf_sel : // unsigned , default = 0
20:16	R/W	0	cti_blend_factor_gama : // unsigned , default = 0
12:8	R/W	0	cti_blend_factor_beta : // unsigned , default = 0
4:0	R/W	0	cti_blend_factor_alpha : // unsigned , default = 0

**Table 9-1529 VPP\_WRBAK\_CTRL\_SEC 0x1dad**

Bit(s)	R/W	Default	Description
31	R/W	0	vpp_wrbak_sel // unsigned , default = 0, 1: VPP_WRBAK_CTRL regs set to vpp_wrbak_data_ini, Cbus can't access 0: VPP_WRBAK_CTRL reg can be written bit
30:0	R/W	0	vpp_wrbak_data_ini // unsigned , default = 0,

**Table 9-1530 VD1\_BLEND\_SRC\_CTRL\_SEC 0x1dae**

Bit(s)	R/W	Default	Description
31	R/W	0	vd1_blend_src_sel : // unsigned , default = 0, 1: VD1_BLEND_SRC_CTRL regs set to vd1_blend_src_data_ini,Cbus can't access 0: VD1_BLEND_SRC_CTRL reg can be written
30:0	R/W	0	vd1_blend_src_data_ini : // unsigned , default = 0,

**Table 9-1531 VD2\_BLEND\_SRC\_CTRL\_SEC 0x1daf**

Bit(s)	R/W	Default	Description
31	R/W	0	vd2_blend_src_sel : // unsigned , default = 0, 1: VD2_BLEND_SRC_CTRL regs set to vd1_blend_src_data_ini,Cbus can't access 0: VD2_BLEND_SRC_CTRL reg can be written
30:0	R/W	0	vd2_blend_src_data_ini : // unsigned , default = 0,

**Table 9-1532 OSD1\_BLEND\_SRC\_CTRL\_SEC 0x1db0**

Bit(s)	R/W	Default	Description
31	R/W	0	osd1_blend_src_sel : // unsigned , default = 0, 1: OSD1_BLEND_SRC_CTRL regs set to osd1_blend_src_data_ini,Cbus can't access 0: OSD1_BLEND_SRC_CTRL reg can be written
30:0	R/W	0	osd1_blend_src_data_ini : // unsigned , default = 0,

**Table 9-1533 OSD2\_BLEND\_SRC\_CTRL\_SEC 0x1db1**

Bit(s)	R/W	Default	Description
31	R/W	0	osd2_blend_src_sel : // unsigned , default = 0, 1: OSD2_BLEND_SRC_CTRL regs set to osd2_blend_src_data_ini,Cbus can't access 0: OSD2_BLEND_SRC_CTRL reg can be written
30:0	R/W	0	osd2_blend_src_data_ini : // unsigned , default = 0,

**Table 9-1534 VPP\_INT\_LINE\_NUM 0x1dce**

Bit(s)	R/W	Default	Description
12:0	R/W	0x0	interrupt_line_num : //unsigned, default== 0x1fff,line number use to generate interrupt when line == this number

**Table 9-1535 VPP\_OFIFO\_URG\_CTRL 0x1dd8**

Bit(s)	R/W	Default	Description
31	R/W	0x0	urgent_hold : //unsigned, default== 0, urgent fifo hold enable
28:12	R/W	0x0	urgent_fifo_th : //unsigned, default== 0, urgent fifo hold line threshold
15	R/W	0x0	urgent_ctrl_en : //unsigned, default== 0, urgent_ctrl_en
14	R/W	0x0	urgent_wr : //unsigned, default== 0, urgent_wr, if true for write buffer
13	R/W	0x0	out_inv_en : //unsigned, default== 0, out_inv_en



Bit(s)	R/W	Default	Description
12	R/W	0x0	urgent_ini_value : //unsigned, default == 0, urgent_ini_value
11:6	R/W	0x0	up_th : //unsigned, default == 0, up_th up threshold
5:0	R/W	0x0	dn_th : //unsigned, default == 0, dn_th dn threshold

**Table 9-1536 VPP\_CLIP\_MISC0 0x1dd9**

Bit(s)	R/W	Default	Description
29:20	R/W	1023	r : // unsigned, default == 1023, final clip r channel top
19:10	R/W	1023	g : // unsigned, default == 1023, final clip g channel top
9: 0	R/W	1023	b : // unsigned, default == 1023, final clip b channel top

**Table 9-1537 VPP\_CLIP\_MISC1 0x1dda**

Bit(s)	R/W	Default	Description
29:20	R/W	0x0	r : // unsigned, default == 0, final clip r channel bottom
19:10	R/W	0x0	g : // unsigned, default == 0, final clip g channel bottom
9: 0	R/W	0x0	b : // unsigned, default == 0, final clip b channel bottom

**Table 9-1538 VPP\_VD1\_CLIP\_MISC0 0x1de1**

Bit(s)	R/W	Default	Description
29:20	R/W	1023	r : //unsigned, default == 1023, vd1 clip r channel top
19:10	R/W	1023	g : //unsigned, default == 1023, vd1 clip g channel top
9: 0	R/W	1023	b : //unsigned, default == 1023, vd1 clip b channel top

**Table 9-1539 VPP\_VD1\_CLIP\_MISC1 0x1de2**

Bit(s)	R/W	Default	Description
29:20	R/W	0x0	r : //unsigned, default = 0, vd1 clip r channel bottom
19:10	R/W	0x0	g : //unsigned, default = 0, vd1 clip g channel bottom
9: 0	R/W	0x0	b : //unsigned, default = 0, vd1 clip b channel bottom

**Table 9-1540 VPP\_VD2\_CLIP\_MISC0 0x1de3**

Bit(s)	R/W	Default	Description
29:20	R/W	1023	r : //unsigned, default = 1023, vd2 clip r channel top
19:10	R/W	1023	g : //unsigned, default = 1023, vd2 clip g channel top
9: 0	R/W	1023	b : //unsigned, default = 1023, vd2 clip b channel top

**Table 9-1541 VPP\_VD2\_CLIP\_MISC1 0x1de4**

Bit(s)	R/W	Default	Description
29:20	R/W	0x0	r : // unsigned, default = 0, vd2 clip r channel bottom
19:10	R/W	0x0	g : // unsigned, default = 0, vd2 clip g channel bottom
9: 0	R/W	0x0	b : // unsigned, default = 0, vd2 clip b channel bottom

**Table 9-1542 VPP\_VD2\_HDR\_IN\_SIZE 0x1df0**

Bit(s)	R/W	Default	Description
18:16	R/W	0	vd2_in_v_size : // unsigned, default = 0x2d0, VPP VD2 input vsize
12:0	R/W	0	vd2_in_h_size : // unsigned, default = 0x1e0, VPP VD2 input hsize

**Table 9-1543 VPP\_OSD1\_IN\_SIZE 0x1df1**

Bit(s)	R/W	Default	Description
18:16	R/W	0	osd1_in_v_size : // unsigned, default = 0x2d0, VPP osd1 input vsize
12:0	R/W	0	osd1_in_h_size : // unsigned, default = 0x1e0, VPP osd1 input hsize

**Table 9-1544 VPP\_GCLK\_CTRL2 0x1df2**

Bit(s)	R/W	Default	Description
13:13	R/W	0	vks_gclk_ctrl : // unsigned, default = 0 ,vks gating clock

**Table 9-1545 VD2\_PPS\_DUMMY\_DATA 0x1df4**

Bit(s)	R/W	Default	Description
23:16	R/W	0x0	Y : //unsigned, default = 0, vd2 scale dummy data
15:8	R/W	0x0	CB : //unsigned, default = 0, vd2 scale dummy data
7: 0	R/W	0x0	CR : //unsigned, default = 0, vd2 scale dummy data

**Table 9-1546 VPP\_OSD1\_BLD\_H\_SCOPE 0x1df5**

Bit(s)	R/W	Default	Description
28:16	R/W	0	blend_osd1_h_start : //unsigned, default = 0x0
12:0	R/W	0	blend_osd1_h_end : //unsigned, default = 0x2d0

**Table 9-1547 VPP\_OSD1\_BLD\_V\_SCOPE 0x1df6**

Bit(s)	R/W	Default	Description
28:16	R/W	0	blend_osd1_v_start : //unsigned, default = 0x0
12:0	R/W	0	blend_osd1_v_end : //unsigned, default = 0x1e0

**Table 9-1548 VPP\_OSD2\_BLD\_H\_SCOPE 0x1df7**

Bit(s)	R/W	Default	Description
28:16	R/W	0x0	blend_osd2_h_start : //unsigned, default = 0
12:0	R/W	0x0	blend_osd2_h_end : //unsigned, default = 0x2d0

**Table 9-1549 VPP\_OSD2\_BLD\_V\_SCOPE 0x1df8**

Bit(s)	R/W	Default	Description
28:16	R/W	0x0	blend_osd2_v_start : //unsigned, default = 0
12:0	R/W	0x0	blend_osd2_v_end : //unsigned, default = 0x1e0

**Table 9-1550 VPP\_WRBAK\_CTRL 0x1df9**

Bit(s)	R/W	Default	Description
27:16	R/W	0	wrbak_din_inblank : //unsigned, default = 0,
12:8	R/W	0	wrbak_din_only_en : //unsigned, default = 0, Bit0: vd2 only Bit1: osd1 only Bit2: osd2 only Bit3: vpp out only Bit4: preblend vd1 only
6:4	R/W	0	wrbak_chan1_sel : //unsigned, default = 0,1:vd1 2:vd2 3:osd1 4:osd2 5:posd_blend 6:vpp output 7:preblend vd1
2:0	R/W	0	wrbak_chan0_sel : //unsigned, default = 0,1:vd1 2:vd2 3:osd1 4:osd2 5:posd_blend 6:vpp output 7:preblend vd1

**Table 9-1551 VPP\_SLEEP\_CTRL 0x1dfa**

Bit(s)	R/W	Default	Description
31	R/W	1	sleep_always_en : //unsigned, default = 1
30	R/W	0	sleep_always_dis : //unsigned, default = 0
29:16	R/W	0	sleep_line_len : //unsigned, default = 0
15:14	R/W	0	sleep_mode : //unsigned, default = 0
13:0	R/W	0	sleep_beg_line : //unsigned, default = 0

**Table 9-1552 VD1\_BLEND\_SRC\_CTRL 0x1dfb**

Bit(s)	R/W	Default	Description
16	R/W	0	vd1_postbld_premult : //unsigned, default = 0
11:8	R/W	1	vd1_postbld_src : //unsigned, default = 1 , 0:close 1:vd1 2:vd2 3:osd1 4:osd2
4	R/W	0	vd1_prebld_premult : //unsigned, default = 0
3:0	R/W	1	vd1_prebld_src : //unsigned, default = 1 , 0:close 1:vd1 2:vd2 3:osd1 4:osd2

**Table 9-1553 VD2\_BLEND\_SRC\_CTRL 0x1dfc**

Bit(s)	R/W	Default	Description
20	R/W	0	vd2_blend_path_sel : //unsigned, default = 0
16	R/W	0	vd2_postbld_premult : //unsigned, default = 0
11:8	R/W	2	vd2_postbld_src : //unsigned, default = 2 , 0:close 1:vd1 2:vd2 3:osd1 4:osd2
4	R/W	0	vd2_prebld_premult : //unsigned, default = 0
3:0	R/W	0	vd2_prebld_src : //unsigned, default = 0 ,0:close 1:vd1 2:vd2 3:osd1 4:osd2

**Table 9-1554 OSD1\_BLEND\_SRC\_CTRL 0x1dfd**

Bit(s)	R/W	Default	Description
20	R/W	0	osd1_blend_path_sel : //unsigned, default = 0
16	R/W	0	osd1_postbld_premult : //unsigned, default = 0
11:8	R/W	3	osd1_postbld_src : //unsigned, default = 3 , 0:close 1:vd1 2:vd2 3:osd1 4:osd2
4	R/W	0	osd1_prebld_premult : //unsigned, default = 0
3:0	R/W	0	osd1_prebld_src : //unsigned, default = 0 ,0:close 1:vd1 2:vd2 3:osd1 4:osd2

**Table 9-1555 OSD2\_BLEND\_SRC\_CTRL 0x1dfe**

Bit(s)	R/W	Default	Description
20	R/W	0	osd2_blend_path_sel : //unsigned, default = 0
16	R/W	0	osd2_postbld_premult : //unsigned, default = 0
11:8	R/W	4	osd2_postbld_src : //unsigned, default = 4 , 0:close 1:vd1 2:vd2 3:osd1 4:osd2
4	R/W	0	osd2_prebld_premult : //unsigned, default = 0
3:0	R/W	0	osd2_prebld_src : //unsigned, default = 0 , 0:close 1:vd1 2:vd2 3:osd1 4:osd2

**Table 9-1556 VPP\_CRC\_CHK 0x1db3**

Bit(s)	R/W	Default	Description
31	W	0	1:write a pulse to start crc
1	W	0	1:pulse start crc (bit31) 0:go_field start crc
0	W	0	reg_crc_en 1:open crc

**Table 9-1557 VPP\_RO\_CRCSUM 0x1db2**

Bit(s)	R/W	Default	Description
31-0	R	0	vpp crc result (lock by go_field)

**Table 9-1558 VPP\_OSD\_SCALE\_CTRL 0x1dff**

Bit(s)	R/W	Default	Description
2	R/W	1	10bit_12bit_ext_mode// 1:add low 2bits 2'b00 0:add high two bits 2'b00 Osd1_scale before osd_blend :this should be 0 Osd1_scale after osd_blend :this should be 1
1	R/W	1	10bit_12bit_ext_mode_bypass Osd1_scale before osd_blend :this should be 0 Osd1_scale after osd_blend :this should be 1
0	R/W	0	Osd1 scale position change 0:after osd_blend 1:before osd_blend

### 9.2.3.21 CM Registers

**Table 9-1559 VPP\_CHROMA\_ADDR\_PORT 0x1d70**

Bit(s)	R/W	Default	Description
31-0	R/W	0	Color management address port

**Table 9-1560 VPP\_CHROMA\_DATA\_PORT 0x1d71**

Bit(s)	R/W	Default	Description
31-0	R/W	0	Color management data port

Color management internal registers is indirectly accessed by the registers VPP\_CHROMA\_ADDR\_PORT and VPP\_CHROMA\_DATA\_PORT.

Color management registers

The example to access the Color management registers is like this:

- Wr(VPP\_CHROMA\_ADDR\_PORT);
- Wr(VPP\_CHROMA\_DATA\_PORT);

**Table 9-1561 REG\_CHROMA\_CONTROL 0x30**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_chroma_en. enable color manage function 1'b1: enable 1'b0: bypass
6	R/W	0	sat_sel. uv_max or $u^2+v^2$ selected as sat for reference 1'b1: uv_max(default) 1'b0: $u^2+v^2$
5	R/W	0	uv_adj_en. final uv_adjust enable 1'b1: enable 1'b0: bypass
2	R/W	0	hue_en. rgb to hue enable 1'b1: enable(default) 1'b0: bypass
1-0	R/W	0	csc_sel. define input YUV with different color type 2'b00: 601(16-235) 2'b01: 709(16-235) 2'b10: 601(0-255) 2'b11: 709(0-255)

**Table 9-1562 SAT\_BYB\_NODE0 0x200**

Bit(s)	R/W	Default	Description
31-24	R/W	0	The 4th node, the same as below
23-16	R/W	0	The 3rd node, the same as below
15-8	R/W	0	The 2nd node, the same as below
7-0	R/W	0	Signed, The 1st node about saturation gain offset along Y coordinate, the gain normalized to 128 as "1".

**Table 9-1563 SAT\_BYB\_NODE1 0x201**

Bit(s)	R/W	Default	Description
31-24	R/W	0	The 8th node, the same as below
23-16	R/W	0	The 7th node, the same as below
15-8	R/W	0	The 6th node, the same as below
7-0	R/W	0	Signed, The 5th node about saturation gain offset along Y coordinate, the gain normalized to 128 as "1".

**Table 9-1564 SAT\_BYB\_NODE2 0x202**

Bit(s)	R/W	Default	Description
31-8	R/W	0	reserved
7-0	R/W	0	Signed, The 9th node about saturation gain offset along Y coordinate, the gain normalized to 128 as "1".

**Table 9-1565 SAT\_SRC\_NODE 0x203**

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	R/W	0x800	unsign, Threshold of input saturation for second & third piece. i.e. it is boundary for reg_CM2_Adj_Sat_via_HS[1][:]and reg_CM2_Adj_Sat_via_HS[2][:]
15-12	R/W	0	reserved
11-0	R/W	0x400	unsign, Threshold of input saturation for first and second piece.i.e. it is boundary for reg_CM2_Adj_Sat_via_HS[0][:] and reg_CM2_Adj_Sat_via_HS[1][:]

**Table 9-1566 CM\_ENH\_SFT\_MODE 0x204**

Bit(s)	R/W	Default	Description
31-11	R/W	0	reserved
10-8	R/W	0	Hue offset adjustments scale for Reg_CM2_Adj_Hue_via_H[:]& Reg_CM2_Adj_Hue_via_S[:]& Reg_CM2_Adj_Hue_via_Y[:] 0: no scale up; 1: upscale by 2 - (-128,127)x2; 2: upscale by 4 - (-128,127)x4; 3: upscale by 8 - (-128,127)x8;

Bit(s)	R/W	Default	Description
7-6	R/W	0	reserved
5-4	R/W	0	Luma offset adjustments scale for reg_CM2_Adj_Luma_via_Hue[i]: 0: no scale up; 1: upscale by2 - (-128,127)x2; 2: upscale by 4 - (-128,127)x4; 3: upscale by 8 - (-128,127)x8;
3-2	R/W	0	Saturation again adjustments scale for reg_CM2_Adj_Sat_via_Y[:][:] &Reg_CM2_Adj_SatGLBgain_via_Y[:]: 0: no scale up/down; 1: dnscale by 2 (-128,127)/2; 2: dnscale by 4 (-128,127)/4; 3: dnscale by 8 (-128,127)/8;
1-0	R/W	0	Saturation again adjustments scale for reg_CM2_Adj_Sat_via_HS[:][:]: 0: no scale up/down; 1: dnscale by 2 (-128,127)/2; 2: dnscale by 4 (-128,127)/4; 3: dnscale by 8 (-128,127)/8;

Table 9-1567 FRM\_SIZE 0x205

Bit(s)	R/W	Default	Description
31-29	R/W	0	reserved
28-16	R/W	0x438	The frame height size
15-13	R/W	0	reserved
12-0	R/W	0x780	The frame width size

Table 9-1568 FITLER\_CFG 0x206

Bit(s)	R/W	Default	Description
31-5	R/W	0	reserved
4	R/W	0	Horizontal Interleave filter (zero-padding) for 3D considerations: 0: using non-zero padding LPF 1: using zero-padding LPF
3-0	R/W	0	Apply CM on LP portion or original video pixels options: bits[1:0]: is for Luma path control; bits[3:2]: is for U/V path control; 0: no filter but still match the delay; 1: 5 taps LP filter 2: 9 taps LP filter 3: 13 taps LP filter

Table 9-1569 CM\_GLOBAL\_GAIN 0x207

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	R/W	0x200	Global Saturation Gain for general color adjustments (0~4095 ==> 0~8), 512 normalized to "1".
15-12	R/W	0	reserved
11-0	R/W	0	Global Hue offsets for general color adjustments (0~4095 ==> 0~360 degree)

**Table 9-1570 CM\_ENH\_CTL 0x208**

Bit(s)	R/W	Default	Description
31-6	R/W	0	reserved
5	R/W	0	CM Bypass 1: Bypass 0: not bypass
4	R/W	0	Enable signal for CM2 Hue adjustments;
3	R/W	0	Enable signal for CM2 Saturation adjustments;
2	R/W	0	Enable signal for CM2 Luma adjustments;
1	R/W	0	cm2_filt_en :apply cm on lp portion enable
0	R/W	0	CM1 enable signal

**Table 9-1571 ROI\_X\_SCOPE 0x209**

Bit(s)	R/W	Default	Description
31-29	R/W	0	reserved
28-16	R/W	0	Ending col index of the Region of Interest (ROI)
15-12	R/W	0	reserved
12-0	R/W	0	Start col index of the Region of Interest (ROI)

**Table 9-1572 ROI\_Y\_SCOPE 0x20a**

Bit(s)	R/W	Default	Description
31-29	R/W	0	reserved
28-16	R/W	0	Ending col index of the Region of Interest (ROI)
15-13	R/W	0	reserved
12-0	R/W	0	Start col index of the Region of Interest (ROI)

**Table 9-1573 POI\_XY\_DIR 0x20b**

Bit(s)	R/W	Default	Description
31-29	R/W	0	reserved
28-16	R/W	0	Row index of the pixel(position) of Interest (POI)
15-13	R/W	0	reserved
12-0	R/W	0	Col index of the pixel(position) of Interest (POI)



**Table 9-1574 COI\_Y\_SCOPE 0x20c**

Bit(s)	R/W	Default	Description
31-16	R/W	0	reserved
15-8	R/W	0	Higher bound of luma value for color of interest (COI), 8bits precision
7-0	R/W	0	Lower bound of luma value for color of interest (COI), 8bits precision

**Table 9-1575 COI\_H\_SCOPE 0x20d**

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	R/W	0	Higher bound of Hue value for color of interest (COI), 12 8bits precision
15-12	R/W	0	reserved
11-0	R/W	0	Lower bound of Hue value for color of interest (COI), 12 bits precision

**Table 9-1576 COI\_S\_SCOPE 0x20e**

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	R/W	0	Higher bound of Sat value for color of interest (COI), 12 8bits precision
15-12	R/W	0	reserved
11-0	R/W	0	Lower bound of Sat value for color of interest (COI), 12 bits precision

**Table 9-1577 IFO\_MODE 0x20f**

Bit(s)	R/W	Default	Description
31-8	R/W	0	reserved
7-4	R/W	0	Mode control for COI replacement, bit[3:2] control COI pixels: 0: no replacement for COI pixels 1: disable CM2 enhance for COI pixels; 2: keep COI pixels Y but replace HS by [*HS]; 3: replace COI pixels to [*YHS] bit[1:0] controls non-COI pixels: 0: no replacement for non-COI pixels 1: disable CM2 enhance for non-COI pixels; 2: keep COI pixels Y but replace HS by [*HS]; 3: replace non- COI pixels to [*YHS]
3-0	R/W	0	Enhance mode control of pixels inside and outside Region of Interest (ROI) , bit [3:2] control ROI: 0: enable CM2 processing in ROI; 1: disable CM2 processing in ROI; 2: keep ROI pixels Y but replace HS by [*HS]; 3: ow ROI pixels to [*YHS] bit [1:0] control pixels other than ROI similarly. 0: enable CM2 processing in non-ROI; 1: disable CM2 processing in non-ROI; 2: keep ROI pixels Y but replace HS by [*HS]; 3: ow non-ROI pixels to [*YHS]

**Table 9-1578 POI\_RPL\_MODE 0x210**

Bit(s)	R/W	Default	Description
31-4	R/W	0	reserved
27-16	R/W	0	Reg_cm2_after_v_offset
15-4	R/W	0	Reg_cm2_after_u_offset
3-0	R/W	0	Pixel of interest (POI) replacement mode: 0: no replacements; 1: one pixel of POI position replaced to [*YHS] 2: 3X3 pixels centering POI position replaced to [*YHS] 3: 5X5 pixels centering POI position replaced to [*YHS] ... 15: 29X29 pixels centering POI position replaced to [*YHS]

**Table 9-1579 DEMO\_OWR\_YHS 0x211**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Saturation value overwriting to ROI/POI/COI; 12bits precision, equal to saturation precision.
23-12	R/W	0	Hue value overwriting to ROI/POI/COI; 12 bits precision, equal to 1/4 hue precision. E.g. { Reg_CM2Demo_OWR_H, 2'h0}
11-0	R/W	0	Luma value overwriting to ROI/POI/COI; 8bits precision, equal to 1/4 luma precision, e.g. { Reg_CM2Demo_OWR_Y, 2'h0}

**Table 9-1580 DEMO\_POI\_Y 0x212**

Bit(s)	R/W	Default	Description
31-8	RO	0	Reserved
7-0	RO	0	Luma value for pixel of interest (POI), only get locked higher 8bits

**Table 9-1581 DEMO\_POI\_H 0x213**

Bit(s)	R/W	Default	Description
31-12	RO	0	Reserved
11-0	RO	0	Hue value for pixel of interest (POI), only get locked higher 12bits

**Table 9-1582 DEMO\_POI\_S 0x214**

Bit(s)	R/W	Default	Description
31-12	RO	0	Reserved
11-0	RO	0	Saturation value for pixel of interest (POI), only get locked higher 12bits

**Table 9-1583 LUMA\_ADJ\_LIMT 0x215**

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
23-16	R/W	0	Slope to do the Luma adjust degrade speed based on Saturation. It was normalized to 16 as '1'.
15-12	R/W	0	reserved
11-0	R/W	0	Threshold to saturation to do Luma adjustment degrade. Only pixels' saturation lower than this threshold will degrade the Luma adjustment.

**Table 9-1584 SAT\_ADJ\_LIMT 0x216**

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
23-16	R/W	0	Slope to do the Sat adjust degrade speed based on Saturation. It was normalized to 16 as '1'.
15-12	R/W	0	reserved
11-0	R/W	0	Threshold to saturation to do Sat adjustment degrade. Only pixels' saturation lower than this threshold will degrade the Luma adjustment.

**Table 9-1585 HUE\_ADJ\_LIMT 0x217**

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
23-16	R/W	0	Slope to do the Hue adjust degrade speed based on Saturation. It was normalized to 16 as '1'.
15-12	R/W	0	reserved
11-0	R/W	0	Threshold to saturation to do Hue adjustment degrade. Only pixels' saturation lower than this threshold will degrade the Luma adjustment.

**Table 9-1586 UVHS\_OFST 0x218**

Bit(s)	R/W	Default	Description
31-24	R/W	0	reserved
23-12	R/W	0	U offset after CM2, under s10 scale
11-8	R/W	0	reserved
7-0	R/W	0	U offset before CM2, under s10 scale.

**Table 9-1587 HUE\_CFG\_PARA 0x219**

Bit(s)	R/W	Default	Description
31-17	R/W	0	reserved
16	R/W	0	Options to protect HUE after CM2 adjustments. This will be added to avoid HUE distortion if Saturation is enhanced too much.
15-13	R/W	0	Hue adjustment via HS the Saturation division mode: 0: 1024/2048/3072, 4095;

Bit(s)	R/W	Default	Description
			1: 512, 1024, 1536, 2048; 2: 256, 512, 768, 1024; 3: 128, 256, 384, 512; 4: 512/1024/2048/4096; 5: 256/512/1024/2048; 6: 128/256/512/1024; 7: 64, 128, 256, 512
12	R/W	0	Hue slice division mode: 0: 32 pieces, 360/32 degrees each slice; 1/up: first 20 slices with 360/64 degrees each slice, others 360/16 degrees each slices. Notes, this option provide options to get more precise Hue adjustments for FTC/ Red and so on
11-0	R/W	0	Hue offset before CM2 adjustment, this will provide options to divide the Hue slices with a precise offset. But need to compensate back with the global Hue after CM2 adjustments

**Table 9-1588 DEMO\_SPLT\_CFG 0x21a**

Bit(s)	R/W	Default	Description
31-22	R/W	0	reserved
21-20	R/W	0	Demo split post
19-16	R/W	0	Demo split width
12-0	R/W	0	Demo split mode

**Table 9-1589 DEMO\_SPLT\_YHS 0x21b**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Luma value
23-12	R/W	0	Hue value
11-0	R/W	0	Sat value

**Table 9-1590 XVYCC\_YSCP\_REG 0x21c**

Bit(s)	R/W	Default	Description
27:16	R/W	0x3ff	xvycc_y_max
11:0	R/W	0x0	xvycc_y_min

**Table 9-1591 XVYCC\_USCP\_REG 0x21d**

Bit(s)	R/W	Default	Description
27:16	R/W	0x3ff	xvycc_u_max
11:0	R/W	0x0	xvycc_u_min

**Table 9-1592 XUYCC\_VSCP\_REG 0x21e**

Bit(s)	R/W	Default	Description
27:16	R/W	0x3ff	xvycc_v_max
11:0	R/W	0x0	xvycc_v_min

**Table 9-1593 LUMA\_ADJ0\_REG 0x21f**

Bit(s)	R/W	Default	Description
21:12	R/W	0x0	Reg_cm_luma_blacklevel
11:0	R/W	0x0	Reg_cm_luma_contrast

**Table 9-1594 LUMA\_ADJ1\_REG 0x220**

Bit(s)	R/W	Default	Description
28:16	R/W	0x0	Reg_sta_sat_hist_mode
12:0	R/W	0x0	Reg_cm_luma_brightness

**Table 9-1595 STA\_WIN\_XYXY0\_REG 0x221**

Bit(s)	R/W	Default	Description
29:16	R/W	0x0	Reg_sta_win_xyxy1
13:0	R/W	0x0	Reg_sta_win_xyxy0

**Table 9-1596 STA\_WIN\_XYXY1\_REG 0x222**

Bit(s)	R/W	Default	Description
29:16	R/W	0x0	Reg_sta_win_xyxy3
13:0	R/W	0x0	Reg_sta_win_xyxy2

**Table 9-1597 STA\_CFG\_REG 0x223**

Bit(s)	R/W	Default	Description
31:30	R/W	0x1	Reg_sta_enable
29:24	R/W	0x1d	Reg_sta_hist_scale
23:16	R/W	0x20	Reg_sta_hue_hist_sat_thrd
15:8	R/W	0x1	Reg_sta_blk_thrd
7:0	R/W	0x254	Reg_sta_brt_thrd

**Table 9-1598 STA\_SAT\_HIST0\_REG 0x224**

Bit(s)	R/W	Default	Description
24	R/W	0x0	Reg_cm_ro_frame
23:16	R/W	0x80	Reg_sta_sat_hist_rang_thr2
15:8	R/W	0x40	Reg_sta_sat_hist_rang_thr1
7:0	R/W	0x0	Reg_sta_sat_hist_rang_thr0

**Table 9-1599 STA\_SAT\_HIST1\_REG 0x225**

Bit(s)	R/W	Default	Description
23:16	R/W	0xff	Reg_sta_sat_hist_rang_thr5
15:8	R/W	0xff	Reg_sta_sat_hist_rang_thr4
7:0	R/W	0xc0	Reg_sta_sat_hist_rang_thr3

**Table 9-1600 RO\_CM\_HUE\_HIST\_BIN0~RO\_CM\_HUE\_HIST\_BIN31 0x226~0x245**

Bit(s)	R/W	Default	Description
31:0	RO	0x0	RO_CM_HUE_HIST_BIN0~RO_CM_HUE_HIST_BIN31

**Table 9-1601 RO\_CM\_SAT\_HIST\_BIN0~RO\_CM\_SAT\_HIST\_BIN31 0x246~0x265**

Bit(s)	R/W	Default	Description
31:0	RO	0x0	RO_CM_SAT_HIST_BIN0~RO_CM_SAT_HIST_BIN31

**Table 9-1602 RO\_CM\_BLK\_BIN 0x266**

Bit(s)	R/W	Default	Description
31:0	RO	0x0	RO_CM_BLK_BIN

**Table 9-1603 RO\_CM\_BRT\_BIN 0x267**

Bit(s)	R/W	Default	Description
31:0	RO	0x0	RO_CM_BRT_BIN

The main adjust parameter is saved according to 32 hue node order, one by one. The parameter of each hue node is same. All the parameter for each hue occupy 5 register-addr-space. For the addr-off-set aligned, we allocate 8 addr-space to each node parameter, for example,

The parameter of 1st node uses the address space : 0x100, 0x101, 0x102, 0x103, 0x104,

The 2nd node uses the address space : 0x108, 0x109, 0x10a, 0x10b, 0x10c,

The 3rd node uses the address space : 0x110, 0x111, 0x112, 0x113, 0x114,

**Table 9-1604 CM2\_ENH\_COEF0\_H00 0x100**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Same as last
23-16	R/W	0	Same as last
15-8	R/W	0	Signed, Saturation gain offset for three pieces saturation on Hue section (totally 32 sections) node 0; the gain normalized to 128 as "1".
7-0	R/W	0	Signed, Luma offsets for Hue section (totally 32 sections) nodes 0 , range (-128,127)

**Table 9-1605 CM2\_ENH\_COEF1\_H00 0x101**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Signed, Hue offset for each four pieces Luma region on each Hue section (totally 32 sections) nodes, this is for $y=2/4$ , hue node 0
23-16	R/W	0	Signed, Hue offset for each four pieces Luma region on each Hue section (totally 32 sections) nodes, this is for $y=1/4$ , hue node 0
15-8	R/W	0	Signed, Hue offset for each four pieces Luma region on each Hue section (totally 32 sections) nodes, this is for $y=0$ , hue node 0
7-0	R/W	0	Signed, Hue offset on Hue section (totally 32 sections) node 0

**Table 9-1606 CM2\_ENH\_COEF2\_H00 0x102**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Signed, Hue offset for each four pieces Saturation region on each Hue section (totally 32 sections) nodes; This is for $sat = 1/4$ , hue node 0
23-16	R/W	0	Signed, Hue offset for each four pieces Saturation region on each Hue section (totally 32 sections) nodes; This is for $sat = 0$ , hue node 0
15-8	R/W	0	Signed, Hue offset for each four pieces Luma region on each Hue section (totally 32 sections) nodes, this is for $y=4/4$ , hue node 0
7-0	R/W	0	Signed, Hue offset for each four pieces Luma region on each Hue section (totally 32 sections) nodes, this is for $y=3/4$ , hue node 0

**Table 9-1607 CM2\_ENH\_COEF3\_H00 0x103**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Saturation gain offset for four pieces Luma on each Hue section (totally 32 sections) nodes; the gain normalized to 128 as "1". This is for $sat = 0$ , hue node 0
23-16	R/W	0	Signed, Hue offset for each four pieces Saturation region on each Hue section (totally 32 sections) nodes; This is for $sat = 4/4$ , hue node 0
15-8	R/W	0	Signed, Hue offset for each four pieces Saturation region on each Hue section (totally 32 sections) nodes; This is for $sat = 3/4$ , hue node 0
7-0	R/W	0	Signed, Hue offset for each four pieces Saturation region on each Hue section (totally 32 sections) nodes; This is for $sat = 2/4$ , hue node 0

**Table 9-1608 CM2\_ENH\_COEF4\_H00 0x104**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Saturation gain offset for four pieces Luma on each Hue section (totally 32 sections) nodes; the gain normalized to 128 as "1". This is for sat = 4/4, hue node 0
23-16	R/W	0	Saturation gain offset for four pieces Luma on each Hue section (totally 32 sections) nodes; the gain normalized to 128 as "1". This is for sat = 3/4, hue node 0
15-8	R/W	0	Saturation gain offset for four pieces Luma on each Hue section (totally 32 sections) nodes; the gain normalized to 128 as "1". This is for sat = 2/4, hue node 0
7-0	R/W	0	Saturation gain offset for four pieces Luma on each Hue section (totally 32 sections) nodes; the gain normalized to 128 as "1". This is for sat = 1/4, hue node 0

CM2\_ENH\_COEF0\_H01 0x108

CM2\_ENH\_COEF1\_H01 0x109

CM2\_ENH\_COEF2\_H01 0x10a

CM2\_ENH\_COEF3\_H01 0x10b

CM2\_ENH\_COEF4\_H01 0x10c

CM2\_ENH\_COEF0\_H02 0x110

CM2\_ENH\_COEF1\_H02 0x111

CM2\_ENH\_COEF2\_H02 0x112

CM2\_ENH\_COEF3\_H02 0x113

CM2\_ENH\_COEF4\_H02 0x114

.....

CM2\_ENH\_COEF0\_H31 0x1f8

CM2\_ENH\_COEF1\_H31 0x1f9

CM2\_ENH\_COEF2\_H31 0x1fa

CM2\_ENH\_COEF3\_H31 0x1fb

CM2\_ENH\_COEF4\_H31

### 9.2.3.22 VPP OSD1 SCALER Registers

**Table 9-1609 VPP\_OSD\_VSC\_PHASE\_STEP 0x5a00**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0100-0000	4.24 format



**Table 9-1610 VPP\_OSD\_VSC\_INI\_PHASE 0x5a01**

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	bottom vertical scaler initial phase
15-0	R/W	0x0	top vertical scaler initial phase

**Table 9-1611 VPP\_OSD\_VSC\_CTRL0 0x5a02**

Bit(s)	R/W	Default	Description
25	R/W	0	Repeat last line en : 1:enable repeat last line 0:disable repeat last line
24	R/W	0x0	osd vertical Scaler enable
23	R/W	0x0	osd_prog_interlace 0: current field is progressive, 1: current field is interlace
22-21	R/W	0x0	osd_vsc_double_line_mode, bit1, double input width and half input height, bit0, change line buffer becomes 2 lines
20	R/W	0x0	osd_vsc_phase0_always_en
19	R/W	0x0	osd_vsc_nearest_en
17-16	R/W	0x0	osd_vsc_bot_rpt_l0_num
14-11	R/W	0x0	osd_vsc_bot_ini_rcv_num
9-8	R/W	0x0	osd_vsc_top_rpt_l0_num
6-3	R/W	0x0	osd_vsc_top_ini_rcv_num
2-0	R/W	0x0	osd_vsc_bank_length

**Table 9-1612 VPP\_OSD\_HSC\_PHASE\_STEP 0x5a03**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0100-0000	4.24 format

**Table 9-1613 VPP\_OSD\_HSC\_INI\_PHASE 0x5a04**

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	horizontal scaler initial phase1
15-0	R/W	0x0	horizontal scaler initial phase0

**Table 9-1614 VPP\_OSD\_HSC\_CTRL0 0x5a05**

Bit(s)	R/W	Default	Description
22	R/W	0x0	osd horizontal Scaler enable
21	R/W	0x0	osd_hsc_double_pix_mode
20	R/W	0x0	osd_hsc_phase0_always_en
19	R/W	0x0	osd_vsc_nearest_en
17-16	R/W	0x0	osd_hsc_rpt_p0_num1

Bit(s)	R/W	Default	Description
14-11	R/W	0x0	osd_hsc_ini_rcv_num1
9-8	R/W	0x0	osd_hsc_rpt_p0_num0
6-3	R/W	0x0	osd_hsc_ini_rcv_num0
2-0	R/W	0x0	osd_hsc_bank_length

Table 9-1615 VPP\_OSD\_HSC\_INI\_PAT\_CTRL 0x5a06

Bit(s)	R/W	Default	Description
15-8	R/W	0x0	for 3D quincunx sub-sampling. pattern, each patten 1 bit, from lsb -> msb
6-4	R/W	0x0	pattern start
2-0	R/W	0x0	pattern end

Table 9-1616 VPP\_OSD\_SC\_DUMMY\_DATA 0x5a07

Bit(s)	R/W	Default	Description
31-24	R/W	0x0	componet 0 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
23-16	R/W	0x0	component 1 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
15-8	R/W	0x0	component 2 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
7-0	R/W	0x0	component 3 , alpha

Table 9-1617 VPP\_OSD\_SC\_CTRL0 0x5a08

Bit(s)	R/W	Default	Description
27:16	R/W	0	osd_sc_gclk_ctrl : //unsigned,default = 0,osd_sc_gclk_ctrl
13	R/W	0	osd_sc_din_osd_alpha_mode : //unsigned,default = 0,osc_sc_din_osd2_alpha_mode, 1: (alpha >= 128) ? alpha - 1: alpha, 0: (alpha >=1) ? alpha - 1: alpha.
12	R/W	0	osd_sc_dout_alpha_mode : //unsigned,default = 0,osc_sc_alpha_mode, 1: (alpha >= 128) ? alpha + 1: alpha, 0: (alpha >=1) ? alpha + 1: alpha.
11:4	R/W	0	osd_sc_alpha : //unsigned,default = 0,default alpha for vd1 or vd2 if they are selected as the source
3	R/W	0	osd_sc_path_en : //unsigned,default = 0,osd scaler path enable
2	R/W	0	osd_sc_en : //unsigned,default = 0,osd scaler enable

Table 9-1618 VPP\_OSD\_SCI\_WH\_M1 0x5a09

Bit(s)	R/W	Default	Description
28-16	R/W	0x0	OSD scaler input width minus 1
12-0	R/W	0x0	OSD scaler input height minus 1

**Table 9-1619 VPP\_OSD\_SCO\_H\_START\_END 0x5a0a**

Bit(s)	R/W	Default	Description
27-16	R/W	0x0	OSD scaler output horizontal start
11-0	R/W	0x0	OSD scaler output horizontal end

**Table 9-1620 VPP\_OSD\_SCO\_V\_START\_END 0x5a0b**

Bit(s)	R/W	Default	Description
27-16	R/W	0x0	OSD scaler output vertical start
11-0	R/W	0x0	OSD scaler output vertical end

**Table 9-1621 VPP\_OSD\_SCALE\_COEF\_IDX 0x5a0c**

Bit(s)	R/W	Default	Description
15	R/W	0x0	Because there are many coefficients used in the vertical filter and horizontal filters, //indirect access the coefficients of vertical filter and horizontal filter is used. //For vertical filter, there are 33x4 coefficients //For horizontal filter, there are 33x4 coefficients //Bit 15 index increment, if bit9 == 1 then (0: index increase 1, 1: index increase 2) else (index increase 2)
14	R/W	0x0	1: read coef through cbus enable, just for debug purpose in case when we wanna check the coef in ram in correct or not
9	R/W	0x0	if true, use 9bit resolution coef, other use 8bit resolution coef
8	R/W	0x0	type of index, 0: vertical coef, 1: horizontal coef
6-0	R/W	0x0	coef index

**Table 9-1622 VPP\_OSD\_SCALE\_COEF 0x5a0d**

Bit(s)	R/W	Default	Description
31-0	R/W	0x0	

**Table 9-1623 OSD\_DB\_FLT\_CTRL 0x5a20**

Bit(s)	R/W	Default	Description
26	R/W	1	reg_nrdeband_reset1 : // unsigned , default = 0 0 : no reset seed 1: reload chroma seed
25	R/W	1	reg_nrdeband_reset0 : // unsigned , default = 0 0 : no reset seed 1: reload luma seed
24	R/W	0	reg_nrdeband_rgb : // unsigned , default = 0 0 : yuv 1: RGB
23	R/W	0	reg_nrdeband_en11 : // unsigned , default = 1 debanding registers of side lines, [0] for luma, same for below
22	R/W	0	reg_nrdeband_en10 : // unsigned , default = 1 debanding registers of side lines, [1] for chroma, same for below
21	R/W	1	reg_nrdeband_siderand : // unsigned , default = 1 options to use side two lines use the rand, instead of use for the YUV three component of middle line, 0: seed

Bit(s)	R/W	Default	Description
			[3]/bandrand[3] for middle line yuv; 1: seed[3]/bandrand[3] for nearby three lines Y;
20	R/W	0	reg_nrdeband_randmode : // unsigned , default = 0 mode of rand noise adding, 0: same noise strength for all difs; else: strenght of noise will not exceed the difs, MIN((pPKReg->reg_nrdeband_bandrand[m]), noise[m])
19:17	R/W	6	reg_nrdeband_bandrand2 : // unsigned , default = 6
15:13	R/W	6	reg_nrdeband_bandrand1 : // unsigned , default = 6
11: 9	R/W	6	reg_nrdeband_bandrand0 : // unsigned , default = 6
7	R/W	1	reg_nrdeband_hpxor1 : // unsigned , default = 1 debanding random hp portion xor, [0] for luma
6	R/W	1	reg_nrdeband_hpxor0 : // unsigned , default = 1 debanding random hp portion xor, [1] for chroma
5	R/W	0	reg_nrdeband_en1 : // unsigned , default = 1 debanding registers, for luma
4	R/W	0	reg_nrdeband_en0 : // unsigned , default = 1 debanding registers, for chroma
3: 2	R/W	2	reg_nrdeband_lpf_mode1 : // unsigned , default = 2 lpf mode, 0: 3x3, 1:3x5; 2: 5x5; 3:5x7
1: 0	R/W	2	reg_nrdeband_lpf_mode0 : // unsigned , default = 2 lpf mode, 0: 3x3, 1:3x5; 2: 5x5; 3:5x7

Table 9-1624 OSD\_DB\_FLT\_CTRL1 0x5a21

Bit(s)	R/W	Default	Description
31:18			reserved
17:16	R/W	2	Reg_osddeband_noise_rs
15:12	R/W	8	Reg_osddeband_randgain
11			reserved
10:8	R/W	6	Reg_osddedband_bandrand5
7			reserved
6: 4	R/W	6	Reg_osddedband_bandrand4
3			reserved
2: 0	R/W	6	Reg_osddeband_bandrand3

Table 9-1625 OSD\_DB\_FLT\_LUMA\_THRD 0x5a22

Bit(s)	R/W	Default	Description
31:30			
29:24	R/W	36	reg_nrdeband_luma_th3 : // unsigned , default = 7 elseif <th[1] use (lpf*3 + y)/4
23:22			
21:16	R/W	28	reg_nrdeband_luma_th2 : // unsigned , default = 5 elseif <th[1] use (lpf*3 + y)/4elseif elseif <th[3] (lpf*1 + 3*y)/4; else
15:14			

Bit(s)	R/W	Default	Description
13: 8	R/W	24	reg_nrdeband_luma_th1 : // unsigned , default = 7 elseif <th[1] use (lpf*3 + y)/4
7: 6			
5: 0	R/W	20	reg_nrdeband_luma_th0 : // unsigned , default = 5 elseif <th[1] use (lpf*3 + y)/4elseif elseif

Table 9-1626 OSD\_DB\_FLT\_LUMA\_THRD 0x5a23

Bit(s)	R/W	Default	Description
31:30			
29:24	R/W	36	reg_nrdeband_chrm_th3 : // unsigned , default = 7 elseif <th[1] use (lpf*3 + y)/4
23:22			
21:16	R/W	28	reg_nrdeband_chrm_th2 : // unsigned , default = 5 elseif <th[1] use (lpf*3 + y)/4elseif elseif <th[3] (lpf*1 + 3*y)/4; else
15:14			
13: 8	R/W	24	reg_nrdeband_chrm_th1 : // unsigned , default = 7 elseif <th[1] use (lpf*3 + y)/4
7: 6			
5: 0	R/W	20	reg_nrdeband_chrm_th0 : // unsigned , default = 5 elseif <th[1] use (lpf*3 + y)/4elseif elseif

Table 9-1627 OSD\_DB\_FLT\_RANLUT 0x5a24

Bit(s)	R/W	Default	Description
23:21	R/W	1	reg_nrdeband_randslut7 : // unsigned , default = 1 lut0
20:18	R/W	1	reg_nrdeband_randslut6 : // unsigned , default = 1 lut0
17:15	R/W	1	reg_nrdeband_randslut5 : // unsigned , default = 1 lut0
14:12	R/W	1	reg_nrdeband_randslut4 : // unsigned , default = 1 lut0
11: 9	R/W	1	reg_nrdeband_randslut3 : // unsigned , default = 1 lut0
8: 6	R/W	1	reg_nrdeband_randslut2 : // unsigned , default = 1 lut0
5: 3	R/W	1	reg_nrdeband_randslut1 : // unsigned , default = 1 lut0
2: 0	R/W	1	reg_nrdeband_randslut0 : // unsigned , default = 1 lut0

Table 9-1628 OSD\_DB\_FLT\_PXI\_THRD 0x5a25

Bit(s)	R/W	Default	Description
25:16	R/W	0	reg_nrdeband_yc_th1 : // unsigned , default = 0 to luma/ u/v  for using the denoise
9: 0	R/W	0	reg_nrdeband_yc_th0 : // unsigned , default = 0 to luma/ u/v  for using the denoise

**Table 9-1629 OSD\_DB\_FLT\_SEED\_Y 0x35a26**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8240	reg_nrdeband_seed0 : // unsigned , default = 1621438240 noise adding seed for Y. seed[0]= 0x60a52f20; as default

**Table 9-1630 OSD\_DB\_FLT\_SEED\_U 0x5a27**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8247	reg_nrdeband_seed1 : // unsigned , default = 1621438247 noise adding seed for U. seed[0]= 0x60a52f27; as default

**Table 9-1631 OSD\_DB\_FLT\_SEED\_V 0x5a28**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8242	reg_nrdeband_seed2 : // unsigned , default = 1621438242 noise adding seed for V. seed[0]= 0x60a52f22; as default

**Table 9-1632 OSD\_DB\_FLT\_SEED3 0x5a29**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8242	reg_nrdeband_seed3 : // unsigned , default = 1621438242 noise adding seed for V. seed[0]= 0x60a52f22; as default

**Table 9-1633 OSD\_DB\_FLT\_SEED4 0x5a2a**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8242	reg_nrdeband_seed4 : // unsigned , default = 1621438242 noise adding seed for V. seed[0]= 0x60a52f22; as default

**Table 9-1634 OSD\_DB\_FLT\_SEED5 0x35a2b**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8242	reg_nrdeband_seed5 : // unsigned , default = 1621438242 noise adding seed for V. seed[0]= 0x60a52f22; as default

### 9.2.3.23 VPP OSD2 SCALER Registers

**Table 9-1635 OSD2\_VSC\_PHASE\_STEP 0x5a40**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0100-0000	4.24 format

**Table 9-1636 OSD2\_VSC\_INI\_PHASE 0x5a41**

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	bottom vertical scaler initial phase
15-0	R/W	0x0	top vertical scaler initial phase

**Table 9-1637 OSD2\_VSC\_CTRL0 0x5a42**

Bit(s)	R/W	Default	Description
25	R/W	0	Repeat last line en : 1:enable repeat last line 0:disable repeat last line
24	R/W	0x0	osd vertical Scaler enable
23	R/W	0x0	osd_prog_interlace 0: current field is progressive, 1: current field is interlace
22-21	R/W	0x0	osd_vsc_double_line_mode, bit1, double input width and half input height, bit0, change line buffer becomes 2 lines
20	R/W	0x0	osd_vsc_phase0_always_en
19	R/W	0x0	osd_vsc_nearest_en
17-16	R/W	0x0	osd_vsc_bot_rpt_l0_num
14-11	R/W	0x0	osd_vsc_bot_ini_rcv_num
9-8	R/W	0x0	osd_vsc_top_rpt_l0_num
6-3	R/W	0x0	osd_vsc_top_ini_rcv_num
2-0	R/W	0x0	osd_vsc_bank_length

**Table 9-1638 OSD2\_HSC\_PHASE\_STEP 0x5a43**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0100-0000	4.24 format

**Table 9-1639 OSD2\_HSC\_INI\_PHASE 0x5a44**

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	horizontal scaler initial phase1
15-0	R/W	0x0	horizontal scaler initial phase0

**Table 9-1640 OSD2\_HSC\_CTRL0 0x5a45**

Bit(s)	R/W	Default	Description
22	R/W	0x0	osd horizontal Scaler enable
21	R/W	0x0	osd_hsc_double_pix_mode
20	R/W	0x0	osd_hsc_phase0_always_en
19	R/W	0x0	osd_vsc_nearest_en
17-16	R/W	0x0	osd_hsc_rpt_p0_num1

Bit(s)	R/W	Default	Description
14-11	R/W	0x0	osd_hsc_ini_rcv_num1
9-8	R/W	0x0	osd_hsc_rpt_p0_num0
6-3	R/W	0x0	osd_hsc_ini_rcv_num0
2-0	R/W	0x0	osd_hsc_bank_length

Table 9-1641 OSD2\_HSC\_INI\_PAT\_CTRL 0x5a46

Bit(s)	R/W	Default	Description
15-8	R/W	0x0	for 3D quincunx sub-sampling. pattern, each patten 1 bit, from lsb -> msb
6-4	R/W	0x0	pattern start
2-0	R/W	0x0	pattern end

Table 9-1642 OSD2\_SC\_DUMMY\_DATA 0x5a47

Bit(s)	R/W	Default	Description
31-24	R/W	0x0	componet 0 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
23-16	R/W	0x0	component 1 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
15-8	R/W	0x0	component 2 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
7-0	R/W	0x0	component 3 , alpha

Table 9-1643 OSD2\_SC\_CTRL0 0x5a48

Bit(s)	R/W	Default	Description
27:16	R/W	0	osd_sc_gclk_ctrl : //unsigned,default = 0,osd_sc_gclk_ctrl
13	R/W	0	osd_sc_din_osd_alpha_mode : //unsigned,default = 0,osc_sc_din_osd2_alpha_mode, 1: (alpha >= 128) ? alpha -1: alpha, 0: (alpha >=1) ? alpha - 1: alpha.
12	R/W	0	osd_sc_dout_alpha_mode : //unsigned,default = 0,osc_sc_alpha_mode, 1: (alpha >= 128) ? alpha + 1: alpha, 0: (alpha >=1) ? alpha + 1: alpha.
11:4	R/W	0	osd_sc_alpha : //unsigned,default = 0,default alpha for vd1 or vd2 if they are selected as the source
3	R/W	0	osd_sc_path_en : //unsigned,default = 0,osd scaler path enable
2	R/W	0	osd_sc_en : //unsigned,default = 0,osd scaler enable

Table 9-1644 OSD2\_SCI\_WH\_M1 0x5a49

Bit(s)	R/W	Default	Description
28-16	R/W	0x0	OSD scaler input width minus 1
12-0	R/W	0x0	OSD scaler input height minus 1



**Table 9-1645 OSD2\_SCO\_H\_START\_END 0x5a4a**

Bit(s)	R/W	Default	Description
27-16	R/W	0x0	OSD scaler output horizontal start
11-0	R/W	0x0	OSD scaler output horizontal end

**Table 9-1646 OSD2\_SCO\_V\_START\_END 0x5a4b**

Bit(s)	R/W	Default	Description
27-16	R/W	0x0	OSD scaler output vertical start
11-0	R/W	0x0	OSD scaler output vertical end

**Table 9-1647 OSD2\_SCALE\_COEF\_IDX 0x5a4c**

Bit(s)	R/W	Default	Description
15	R/W	0x0	Because there are many coefficients used in the vertical filter and horizontal filters, //indirect access the coefficients of vertical filter and horizontal filter is used. //For vertical filter, there are 33x4 coefficients //For horizontal filter, there are 33x4 coefficients //Bit 15 index increment, if bit9 == 1 then (0: index increase 1, 1: index increase 2) else (index increase 2)
14	R/W	0x0	1: read coef through cbus enable, just for debug purpose in case when we wanna check the coef in ram in correct or not
9	R/W	0x0	if true, use 9bit resolution coef, other use 8bit resolution coef
8	R/W	0x0	type of index, 0: vertical coef, 1: horizontal coef
6-0	R/W	0x0	coef index

**Table 9-1648 OSD2\_SCALE\_COEF 0x5a4d**

Bit(s)	R/W	Default	Description
31-0	R/W	0x0	

### 9.2.3.24 VPP OSD3 Scaler Registers

**Table 9-1649 OSD34\_VSC\_PHASE\_STEP 0x5a80**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0100-0000	4.24 format

**Table 9-1650 OSD34\_VSC\_INI\_PHASE 0x5a81**

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	bottom vertical scaler initial phase
15-0	R/W	0x0	top vertical scaler initial phase

**Table 9-1651 OSD34\_VSC\_CTRL0 0x5a82**

Bit(s)	R/W	Default	Description
25	R/W	0	Repeat last line en : 1:enable repeat last line 0:disable repeat last line
24	R/W	0x0	osd vertical Scaler enable
23	R/W	0x0	osd_prog_interlace 0: current field is progressive, 1: current field is interlace
22-21	R/W	0x0	osd_vsc_double_line_mode, bit1, double input width and half input height, bit0, change line buffer becomes 2 lines
20	R/W	0x0	osd_vsc_phase0_always_en
19	R/W	0x0	osd_vsc_nearest_en
17-16	R/W	0x0	osd_vsc_bot_rpt_l0_num
14-11	R/W	0x0	osd_vsc_bot_ini_rcv_num
9-8	R/W	0x0	osd_vsc_top_rpt_l0_num
6-3	R/W	0x0	osd_vsc_top_ini_rcv_num
2-0	R/W	0x0	osd_vsc_bank_length

**Table 9-1652 OSD34\_HSC\_PHASE\_STEP 0x5a83**

Bit(s)	R/W	Default	Description
27-0	R/W	0x0100-0000	4.24 format

**Table 9-1653 OSD34\_HSC\_INI\_PHASE 0x5a84**

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	horizontal scaler initial phase1
15-0	R/W	0x0	horizontal scaler initial phase0

**Table 9-1654 OSD34\_HSC\_CTRL0 0x5a85**

Bit(s)	R/W	Default	Description
22	R/W	0x0	osd horizontal Scaler enable
21	R/W	0x0	osd_hsc_double_pix_mode
20	R/W	0x0	osd_hsc_phase0_always_en
19	R/W	0x0	osd_vsc_nearest_en
17-16	R/W	0x0	osd_hsc_rpt_p0_num1
14-11	R/W	0x0	osd_hsc_ini_rcv_num1
9-8	R/W	0x0	osd_hsc_rpt_p0_num0
6-3	R/W	0x0	osd_hsc_ini_rcv_num0
2-0	R/W	0x0	osd_hsc_bank_length

**Table 9-1655 OSD34\_HSC\_INI\_PAT\_CTRL 0x5a86**

Bit(s)	R/W	Default	Description
15-8	R/W	0x0	for 3D quincunx sub-sampling. pattern, each patten 1 bit, from lsb -> msb
6-4	R/W	0x0	pattern start
2-0	R/W	0x0	pattern end

**Table 9-1656 OSD34\_SC\_DUMMY\_DATA 0x5a87**

Bit(s)	R/W	Default	Description
31-24	R/W	0x0	componet 0 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
23-16	R/W	0x0	component 1 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
15-8	R/W	0x0	component 2 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
7-0	R/W	0x0	component 3 , alpha

**Table 9-1657 OSD34\_SC\_CTRL0 0x5a88**

Bit(s)	R/W	Default	Description
27:16	R/W	0	osd_sc_gclk_ctrl : //unsigned,default = 0,osd_sc_gclk_ctrl
13	R/W	0	osd_sc_din_osd_alpha_mode : //unsigned,default = 0,osc_sc_din_osd2_alpha_mode, 1: (alpha >= 128) ? alpha -1: alpha, 0: (alpha >=1) ? alpha - 1: alpha.
12	R/W	0	osd_sc_dout_alpha_mode : //unsigned,default = 0,osc_sc_alpha_mode, 1: (alpha >= 128) ? alpha + 1: alpha, 0: (alpha >=1) ? alpha + 1: alpha.
11:4	R/W	0	osd_sc_alpha : //unsigned,default = 0,default alpha for vd1 or vd2 if they are selected as the source
3	R/W	0	osd_sc_path_en : //unsigned,default = 0,osd scaler path enable
2	R/W	0	osd_sc_en : //unsigned,default = 0,osd scaler enable

**Table 9-1658 OSD34\_SCI\_WH\_M1 0x5a89**

Bit(s)	R/W	Default	Description
28-16	R/W	0x0	OSD scaler input width minus 1
12-0	R/W	0x0	OSD scaler input height minus 1

**Table 9-1659 OSD34\_SCO\_H\_START\_END 0x5a8a**

Bit(s)	R/W	Default	Description
27-16	R/W	0x0	OSD scaler output horizontal start
11-0	R/W	0x0	OSD scaler output horizontal end

Table 9-1660 OSD34\_SCO\_V\_START\_END 0x5a8b

Bit(s)	R/W	Default	Description
27-16	R/W	0x0	OSD scaler output vertical start
11-0	R/W	0x0	OSD scaler output vertical end

Table 9-1661 OSD34\_SCALE\_COEF\_IDX 0x5a8c

Bit(s)	R/W	Default	Description
15	R/W	0x0	Because there are many coefficients used in the vertical filter and horizontal filters, //indirect access the coefficients of vertical filter and horizontal filter is used. //For vertical filter, there are 33x4 coefficients //For horizontal filter, there are 33x4 coefficients //Bit 15 index increment, if bit9 == 1 then (0: index increase 1, 1: index increase 2) else (index increase 2)
14	R/W	0x0	1: read coef through cbus enable, just for debug purpose in case when we wanna check the coef in ram in correct or not
9	R/W	0x0	if true, use 9bit resolution coef, other use 8bit resolution coef
8	R/W	0x0	type of index, 0: vertical coef, 1: horizontal coef
6-0	R/W	0x0	coef index

Table 9-1662 OSD34\_SCALE\_COEF 0x5a8d

Bit(s)	R/W	Default	Description
31-0	R/W	0x0	

### 9.2.3.25 VPP OSD4 Scaler Registers

Table 9-1663 OSD4\_VSC\_PHASE\_STEP 0x5ac0

Bit(s)	R/W	Default	Description
27-0	R/W	0x0100-0000	4.24 format

Table 9-1664 OSD4\_VSC\_INI\_PHASE 0x5ac1

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	bottom vertical scaler initial phase
15-0	R/W	0x0	top vertical scaler initial phase

Table 9-1665 OSD4\_VSC\_CTRL0 0x5ac2

Bit(s)	R/W	Default	Description
25	R/W	0	Repeat last line en : 1:enable repeat last line 0:disable repeat last line
24	R/W	0x0	osd vertical Scaler enable

Bit(s)	R/W	Default	Description
23	R/W	0x0	osd_prog_interlace 0: current field is progressive, 1: current field is interlace
22-21	R/W	0x0	osd_vsc_double_line_mode, bit1, double input width and half input height, bit0, change line buffer becomes 2 lines
20	R/W	0x0	osd_vsc_phase0_always_en
19	R/W	0x0	osd_vsc_nearest_en
17-16	R/W	0x0	osd_vsc_bot_rpt_l0_num
14-11	R/W	0x0	osd_vsc_bot_ini_rcv_num
9-8	R/W	0x0	osd_vsc_top_rpt_l0_num
6-3	R/W	0x0	osd_vsc_top_ini_rcv_num
2-0	R/W	0x0	osd_vsc_bank_length

Table 9-1666 OSD4\_HSC\_PHASE\_STEP 0x5ac3

Bit(s)	R/W	Default	Description
27-0	R/W	0x0100-0000	4.24 format

Table 9-1667 OSD4\_HSC\_INI\_PHASE 0x5ac4

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	horizontal scaler initial phase1
15-0	R/W	0x0	horizontal scaler initial phase0

Table 9-1668 OSD4\_HSC\_CTRL0 0x5ac5

Bit(s)	R/W	Default	Description
22	R/W	0x0	osd horizontal Scaler enable
21	R/W	0x0	osd_hsc_double_pix_mode
20	R/W	0x0	osd_hsc_phase0_always_en
19	R/W	0x0	osd_vsc_nearest_en
17-16	R/W	0x0	osd_hsc_rpt_p0_num1
14-11	R/W	0x0	osd_hsc_ini_rcv_num1
9-8	R/W	0x0	osd_hsc_rpt_p0_num0
6-3	R/W	0x0	osd_hsc_ini_rcv_num0
2-0	R/W	0x0	osd_hsc_bank_length

**Table 9-1669 OSD4\_HSC\_INI\_PAT\_CTRL 0x5ac6**

Bit(s)	R/W	Default	Description
15-8	R/W	0x0	for 3D quincunx sub-sampling. pattern, each pattern 1 bit, from lsb -> msb
6-4	R/W	0x0	pattern start
2-0	R/W	0x0	pattern end

**Table 9-1670 OSD4\_SC\_DUMMY\_DATA 0x5ac7**

Bit(s)	R/W	Default	Description
31-24	R/W	0x0	component 0 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
23-16	R/W	0x0	component 1 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
15-8	R/W	0x0	component 2 ,data = (dst_data >> 4) ,so 8 bit in 12bit mode, 6bit in 10bit mode
7-0	R/W	0x0	component 3 , alpha

**Table 9-1671 OSD4\_SC\_CTRL0 0x5ac8**

Bit(s)	R/W	Default	Description
27:16	R/W	0	osd_sc_gclk_ctrl : //unsigned,default = 0,osd_sc_gclk_ctrl
13	R/W	0	osd_sc_din_osd_alpha_mode : //unsigned,default = 0,osc_sc_din_osd2_alpha_mode, 1: (alpha >= 128) ? alpha - 1: alpha, 0: (alpha >=1) ? alpha - 1: alpha.
12	R/W	0	osd_sc_dout_alpha_mode : //unsigned,default = 0,osc_sc_alpha_mode, 1: (alpha >= 128) ? alpha + 1: alpha, 0: (alpha >=1) ? alpha + 1: alpha.
11:4	R/W	0	osd_sc_alpha : //unsigned,default = 0,default alpha for vd1 or vd2 if they are selected as the source
3	R/W	0	osd_sc_path_en : //unsigned,default = 0,osd scaler path enable
2	R/W	0	osd_sc_en : //unsigned,default = 0,osd scaler enable

**Table 9-1672 OSD4\_SCI\_WH\_M1 0x5ac9**

Bit(s)	R/W	Default	Description
28-16	R/W	0x0	OSD scaler input width minus 1
12-0	R/W	0x0	OSD scaler input height minus 1

**Table 9-1673 OSD4\_SCO\_H\_START\_END 0x5aca**

Bit(s)	R/W	Default	Description
27-16	R/W	0x0	OSD scaler output horizontal start
11-0	R/W	0x0	OSD scaler output horizontal end

**Table 9-1674 OSD4\_SCO\_V\_START\_END 0x5acb**

Bit(s)	R/W	Default	Description
27-16	R/W	0x0	OSD scaler output vertical start
11-0	R/W	0x0	OSD scaler output vertical end

**Table 9-1675 OSD4\_SCALE\_COEF\_IDX 0x5acc**

Bit(s)	R/W	Default	Description
15	R/W	0x0	Because there are many coefficients used in the vertical filter and horizontal filters, //indirect access the coefficients of vertical filter and horizontal filter is used. //For vertical filter, there are 33x4 coefficients //For horizontal filter, there are 33x4 coefficients //Bit 15 index increment, if bit9 == 1 then (0: index increase 1, 1: index increase 2) else (index increase 2)
14	R/W	0x0	1: read coef through cbus enable, just for debug purpose in case when we wanna check the coef in ram in correct or not
9	R/W	0x0	if true, use 9bit resolution coef, other use 8bit resolution coef
8	R/W	0x0	type of index, 0: vertical coef, 1: horizontal coef
6-0	R/W	0x0	coef index

**Table 9-1676 OSD4\_SCALE\_COEF 0x5acd**

Bit(s)	R/W	Default	Description
31-0	R/W	0x0	

### 9.2.3.26 VPP VD1 SCALER Registers

Because there are many coefficients used in the vertical filter and horizontal filters, indirect access the coefficients of vertical filter and horizontal filter is used. For vertical filter, there are 33x4 coefficients For horizontal filter, there are 33x4 coefficients.

**Table 9-1677 VPP\_SCALE\_COEF\_IDX 0x1d03**

Bit(s)	R/W	Default	Description
15	R/W	0	index_inc : // unsigned , default = 0x0 ,index increment, if bit9 == 1 then (0: index increase 1, 1: index increase 2) else (index increase 2)
14	R/W	0	rd_cbus_coef_en : // unsigned , default = 0x0 ,1: read coef through cbus enable, just for debug purpose in case when we wanna check the coef in ram in correct or not
13	R/W	0	vf_sep_coef_en : // unsigned , default = 0x0 ,if true, vertical separated coef enable
9	R/W	0	high_reso_en : // unsigned , default = 0x0 ,if true, use 9bit resolution coef, other use 8bit resolution coef
8:7	R/W	0	type_index : // unsigned , default = 0x0 ,type of index, 00: vertical coef, 01: vertical chroma coef, 10: horizontal coef, 11: reserved
6:0	R/W	0	coef_index : // unsigned , default = 0x0 ,coef index

**Table 9-1678 VPP\_SCALE\_COEF 0x1d04**

Bit(s)	R/W	Default	Description
31:24	R/W	0	coef0 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter
23:16	R/W	0	coef1 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter
15:8	R/W	0	coef2 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter
7 :0	R/W	0	coef3 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter

**Table 9-1679 VPP\_VSC\_REGION12\_STARTP 0x1d05**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region1_startp : //unsigned , default = 0 ,region1 startp
12:0	R/W	0	region2_startp : //unsigned , default = 0 ,region2 startp

**Table 9-1680 VPP\_VSC\_REGION34\_STARTP 0x1d06**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region3_startp
12:0	R/W	0	region4_startp

**Table 9-1681 VPP\_VSC\_REGION4\_ENDP 0x1d07**

Bit(s)	R/W	Default	Description
12:0	R/W	13	region4_endp : //unsigned , default = 13'd1079 ,region4 endp

**Table 9-1682 VPP\_VSC\_START\_PHASE\_STEP 0x1d08**

Bit(s)	R/W	Default	Description
27:24	R/W	1	integer_part : //unsigned , default = 1,vertical start phase step, (source/dest)* (2^24),integer part of step
23:0	R/W	0	fraction_part : //unsigned , default = 0,vertical start phase step, (source/dest)* (2^24),fraction part of step

**Table 9-1683 VPP\_VSC\_REGION0\_PHASE\_SLOPE 0x1d09**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,vertical scaler region0 phase slope, region0 phase slope

**Table 9-1684 VPP\_VSC\_REGION1\_PHASE\_SLOPE 0x1d0a**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region1_phase_slope : //signed , default = 0,region1 phase slope



**Table 9-1685 VPP\_VSC\_REGION3\_PHASE\_SLOPE 0x1d0b**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region3_phase_slope : //signed , default = 0,region3 phase slope

**Table 9-1686 VPP\_VSC\_REGION4\_PHASE\_SLOPE 0x1d0c**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region4_phase_slope : //signed , default = 0,region4 phase slope

**Table 9-1687 VPP\_VSC\_PHASE\_CTRL 0x1d0d**

Bit(s)	R/W	Default	Description
18:17	R/W	0	vsc_double_line_mode : //unsigned , default = 0, double line mode, input/output line width of vscler becomes 2X, so only 2 line buffer in this case, use for 3D line by line interleave scaling bit1 true, double the input width and half input height, bit0 true, change line buffer 2 lines instead of 4 lines
16	R.O	0	prog_interlace : //unsigned , default = 0,0: progressive output, 1: interlace output
15	R/W	0	vsc_bot_l0_out_en : //unsigned , default = 0,vertical scaler output line0 in advance or not for bottom field
14:13	R/W	1	vsc_bot_rpt_l0_num : //unsigned , default = 1,vertical scaler initial repeat line0 number for bottom field
11:8	R/W	4	vsc_bot_ini_rcv_num : //unsigned , default = 4,vertical scaler initial receiving number for bottom field
7	R/W	0	vsc_top_l0_out_en : //unsigned , default = 0,vertical scaler output line0 in advance or not for top field
6:5	R/W	1	vsc_top_rpt_l0_num : //unsigned , default = 1,vertical scaler initial repeat line0 number for top field
3:0	R/W	4	vsc_top_ini_rcv_num : //unsigned , default = 4,vertical scaler initial receiving number for top field

**Table 9-1688 VPP\_VSC\_INI\_PHASE 0x1d0e**

Bit(s)	R/W	Default	Description
31:16	R/W	0,	//unsigned , default = 0,vertical scaler field initial phase for bottom field Bit 15:0 //unsigned , default = 0,vertical scaler field initial phase for top field

**Table 9-1689 VPP\_HSC\_REGION12\_STARTP 0x1d10**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region1_startp : //unsigned , default = 0,region1 startp
12:0	R/W	0	region2_startp : //unsigned , default = 0,region2 startp

**Table 9-1690 VPP\_HSC\_REGION34\_STARTP 0x1d11**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region3 : startp //unsigned , default = 0x780,region3 startp
12:0	R/W	0	region4 : startp //unsigned , default = 0x780,region4 startp

**Table 9-1691 VPP\_HSC\_REGION4\_ENDP 0x1d12**

Bit(s)	R/W	Default	Description
12:0	R/W	13	region4 : startp //unsigned , default = 13'd1919,region4 startp

**Table 9-1692 horizontal start phase step, (source/dest)\*(2^24) VPP\_HSC\_START\_PHASE\_STEP 0x1d13**

Bit(s)	R/W	Default	Description
27:24	R/W	1	integer_part : //unsigned , default = 1,integer part of step
23:0	R/W	0	fraction_part : //unsigned , default = 0,fraction part of step

**Table 9-1693 VPP\_HSC\_REGION0\_PHASE\_SLOPE 0x1d14**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region0 phase slope

**Table 9-1694 VPP\_HSC\_REGION1\_PHASE\_SLOPE 0x1d15**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region1 phase slope

**Table 9-1695 VPP\_HSC\_REGION3\_PHASE\_SLOPE 0x1d16**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region3 phase slope

**Table 9-1696 VPP\_HSC\_REGION4\_PHASE\_SLOPE 0x1d17**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region4 phase slope

**Table 9-1697 VPP\_HSC\_PHASE\_CTRL 0x1d18**

Bit(s)	R/W	Default	Description
22:21	R/W	1	hsc_rpt_p0_num0 : //unsigned , default = 1 ,horizontal scaler initial repeat pixel0 number0
19:16	R/W	4	hsc_ini_rcv_num0 : //unsigned , default = 4 ,horizontal scaler initial receiving number0
15:0	R/W	0	hsc_ini_phase0 : //unsigned , default = 0 ,horizontal scaler top field initial phase0

**Table 9-1698 VPP\_SC\_MISC 0x1d19**

Bit(s)	R/W	Default	Description
24	R/W	0	Repeat last line en : 1:enable repeat last line 0:disable repeat last line
22	R/W	0	hsc_len_div2_en : //unsigned , default = 0 ,if true, divide VSC line length 2 as the HSC input length, othwise VSC length length is the same as the VSC line length just for special usage, more flexibility
21	R/W	0	lbuf_mode : //unsigned , default = 0 ,if true, prevsc uses lin buffer, otherwise prevsc does not use line buffer, it should be same as prevsc_en
20	R/W	0	prehsc_en : //unsigned , default = 0 ,prehsc_en
19	R/W	0	prevsc_en : //unsigned , default = 0 ,prevsc_en
18	R/W	0	vsc_en : //unsigned , default = 0 ,vsc_en
17	R/W	0	hsc_en : //unsigned , default = 0 ,hsc_en
16	R/W	0	sc_top_en : //unsigned , default = 0 ,scale_top_en
15	R/W	0	sc_vd_en : //unsigned , default = 0 ,video1 scale out enable
12	R/W	1	hsc_nonlinear_4region_en : //unsigned , default = 1 ,if true, region0,region4 are nonlinear regions, otherwise they are not scaling regions, for horizontal scaler
10:8	R/W	0	hsc_bank_length : //unsigned , default = 0 ,horizontal scaler bank length
5	R/W	4	vsc_phase_field_mode : //unsigned , default = 4 ,vertical scaler phase field mode, if true, disable the opposite parity line output, more bandwidth needed if output 1080i
4	R/W	0	vsc_nonlinear_4region_en : //unsigned , default = 0 ,if true, region0,region4 are nonlinear regions, otherwise they are not scaling regions, for vertical scaler
2:0	R/W	4	vsc_bank_length : //unsigned , default = 4 ,vertical scaler bank length

**Table 9-1699 VPP\_SCO\_FIFO\_CTRL 0x1d33**

Bit(s)	R/W	Default	Description
27:16	R/W	0	sco_fifo_line_lenm1 : //unsigned , default = 0xfff, scale out fifo line length minus 1
12:0	R/W	0	sco_fifo_size : //unsigned , default = 0x200, scale out fifo size (actually only bit 11:1 is valid, 11:1, max 1024), always even number

For 3D quincunx sub-sampling and horizontal pixel by pixel 3D interleaving.

**Table 9-1700 VPP\_HSC\_PHASE\_CTRL1 0x1d34**

Bit(s)	R/W	Default	Description
27:24	R/W	0	prehsc_mode : //unsigned , default = 0,prehsc_mode, bit 3:2, prehsc odd line interp mode, bit 1:0, prehsc even line interp mode, each 2bit, 00: pix0+pix1/2, average, 01: pix1, 10: pix0
23	R/W	0	hsc_double_pix_mode : //unsigned , default = 0,horizontal scaler double pixel mode
22:21	R/W	1	hsc_rpt_p0_num1 : //unsigned , default = 1,horizontal scaler initial repeat pixel0 number1
19:16	R/W	4	hsc_ini_rcv_num1 : //unsigned , default = 4,horizontal scaler initial receiving number1
15:0	R/W	0	hsc_ini_phase1 : //unsigned , default = 0,horizontal scaler top field initial phase1

For 3D quincunx sub-sampling.

**Table 9-1701 VPP\_HSC\_INI\_PAT\_CTRL 0x1d35**

Bit(s)	R/W	Default	Description
31:24	R/W	0	prehsc_pattern : //unsigned , default = 0, prehsc pattern, each pattern 1 bit, from lsb -> msb
22:20	R/W	0	prehsc_pat_star : //unsigned , default = 0, prehsc pattern start
18:16	R/W	0	prehsc_pat_end : //unsigned , default = 0, prehsc pattern end
15:8	R/W	0	hsc_pattern : //unsigned , default = 0, hsc pattern, each pattern 1 bit, from lsb -> msb
6:4	R/W	0	hsc_pat_start : //unsigned , default = 0, hsc pattern start
2:0	R/W	0	hsc_pat_end : //unsigned , default = 0, hsc pattern end

**Table 9-1702 VPP\_SC\_GCLK\_CTRL 0x1d35**

Bit(s)	R/W	Default	Description
15:0	R/W	0	vpp_sc_gclk_ctrl : //unsigned , default = 0 ,scale clock gate

### 9.2.3.27 VPP VD2 SCALER Registers

Because there are many coefficients used in the vertical filter and horizontal filters, indirect access the coefficients of vertical filter and horizontal filter is used. For vertical filter, there are 33x4 coefficients. For horizontal filter, there are 33x4 coefficients.

**Table 9-1703 VD2\_SCALE\_COEF\_IDX 0x3933**

Bit(s)	R/W	Default	Description
15	R/W	0	index_inc : // unsigned , default = 0x0 ,index increment, if bit9 == 1 then (0: index increase 1, 1: index increase 2) else (index increase 2)
14	R/W	0	rd_cbus_coef_en : // unsigned , default = 0x0 ,1: read coef through cbus enable, just for debug purpose in case when we wanna check the coef in ram in correct or not
13	R/W	0	vf_sep_coef_en : // unsigned , default = 0x0 ,if true, vertical separated coef enable

Bit(s)	R/W	Default	Description
9	R/W	0	high_reso_en : // unsigned , default = 0x0 ,if true, use 9bit resolution coef, other use 8bit resolution coef
8:7	R/W	0	type_index : // unsigned , default = 0x0 ,type of index, 00: vertical coef, 01: vertical chroma coef: 10: horizontal coef, 11: reserved
6:0	R/W	0	coef_index : // unsigned , default = 0x0 ,coef index

**Table 9-1704 VD2\_SCALE\_COEF 0x3934**

Bit(s)	R/W	Default	Description
31:24	R/W	0	coef0 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter
23:16	R/W	0	coef1 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter
15:8	R/W	0	coef2 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter
7 :0	R/W	0	coef3 : //signed , default = 0x0 , coefficients for vertical filter and horizontal filter

**Table 9-1705 VD2\_VSC\_REGION12\_STARTP 0x3935**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region1_startp : //unsigned , default = 0 ,region1 startp
12:0	R/W	0	region2_startp : //unsigned , default = 0 ,region2 startp

**Table 9-1706 VD2\_VSC\_REGION34\_STARTP 0x3936**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region3_startp : //unsigned , default = 0x0438,region3 startp
12:0	R/W	0	region4_startp : //unsigned , default = 0x0438,region4 startp

**Table 9-1707 VD2\_VSC\_REGION4\_ENDP 0x3937**

Bit(s)	R/W	Default	Description
12:0	R/W	13	region4_endp : //unsigned , default = 13'd1079 ,region4 endp

**Table 9-1708 VD2\_VSC\_START\_PHASE\_STEP 0x3938**

Bit(s)	R/W	Default	Description
27:24	R/W	1	integer_part : //unsigned , default = 1,vertical start phase step, (source/dest)* (2^24),integer part of step
23:0	R/W	0	fraction_part : //unsigned , default = 0,vertical start phase step, (source/dest)* (2^24),fraction part of step

**Table 9-1709 VD2\_VSC\_REGION0\_PHASE\_SLOPE 0x3939**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,vertical scaler region0 phase slope, region0 phase slope

**Table 9-1710 VD2\_VSC\_REGION1\_PHASE\_SLOPE 0x393a**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region1_phase_slope : //signed , default = 0,region1 phase slope

**Table 9-1711 VD2\_VSC\_REGION3\_PHASE\_SLOPE 0x393b**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region3_phase_slope : //signed , default = 0,region3 phase slope

**Table 9-1712 VD2\_VSC\_REGION4\_PHASE\_SLOPE 0x393c**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region4_phase_slope : //signed , default = 0,region4 phase slope

**Table 9-1713 VD2\_VSC\_PHASE\_CTRL 0x393d**

Bit(s)	R/W	Default	Description
18:17	R/W	0	vsc_double_line_mode : //unsigned , default = 0, double line mode, input/output line width of vscler becomes 2X, so only 2 line buffer in this case, use for 3D line by line interleave scaling bit1 true, double the input width and half input height, bit0 true, change line buffer 2 lines instead of 4 lines
16	R.O	0	prog_interlace : //unsigned , default = 0,0: progressive output, 1: interlace output
15	R/W	0	vsc_bot_l0_out_en : //unsigned , default = 0,vertical scaler output line0 in advance or not for bottom field
14:13	R/W	1	vsc_bot_rpt_l0_num : //unsigned , default = 1,vertical scaler initial repeat line0 number for bottom field
11:8	R/W	4	vsc_bot_ini_rcv_num : //unsigned , default = 4,vertical scaler initial receiving number for bottom field
7	R/W	0	vsc_top_l0_out_en : //unsigned , default = 0,vertical scaler output line0 in advance or not for top field
6:5	R/W	1	vsc_top_rpt_l0_num : //unsigned , default = 1,vertical scaler initial repeat line0 number for top field
3:0	R/W	4	vsc_top_ini_rcv_num : //unsigned , default = 4,vertical scaler initial receiving number for top field

**Table 9-1714 VD2\_VSC\_INI\_PHASE 0x393e**

Bit(s)	R/W	Default	Description
31:16	R/W	0,	//unsigned , default = 0,vertical scaler field initial phase for bottom field Bit 15:0 //unsigned , default = 0,vertical scaler field initial phase for top field

**Table 9-1715 VD2\_HSC\_REGION12\_STARTP 0x393f**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region1_startp : //unsigned , default = 0,region1 startp
12:0	R/W	0	region2_startp : //unsigned , default = 0,region2 startp

**Table 9-1716 VD2\_HSC\_REGION34\_STARTP 0x3940**

Bit(s)	R/W	Default	Description
28:16	R/W	0	region3 : startp //unsigned , default = 0x780,region3 startp
12:0	R/W	0	region4 : startp //unsigned , default = 0x780,region4 startp

**Table 9-1717 VD2\_HSC\_REGION4\_ENDP 0x3941**

Bit(s)	R/W	Default	Description
12:0	R/W	13	region4 : startp //unsigned , default = 13'd1919,region4 startp

Horizontal start phase step, (source/dest)\*(2<sup>24</sup>)

**Table 9-1718 VD2\_HSC\_START\_PHASE\_STEP 0x3942**

Bit(s)	R/W	Default	Description
27:24	R/W	1	integer_part : //unsigned , default = 1,integer part of step
23:0	R/W	0	fraction_part : //unsigned , default = 0,fraction part of step

**Table 9-1719 VD2\_HSC\_REGION0\_PHASE\_SLOPE 0x3943**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region0 phase slope

**Table 9-1720 VD2\_HSC\_REGION1\_PHASE\_SLOPE 0x3944**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region1 phase slope

**Table 9-1721 VD2\_HSC\_REGION3\_PHASE\_SLOPE 0x3945**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region3 phase slope

**Table 9-1722 VD2\_HSC\_REGION4\_PHASE\_SLOPE 0x3946**

Bit(s)	R/W	Default	Description
24:0	R/W	0	region0_phase_slope : //signed , default = 0,region4 phase slope

**Table 9-1723 VD2\_HSC\_PHASE\_CTRL 0x3947**

Bit(s)	R/W	Default	Description
22:21	R/W	1	hsc_rpt_p0_num0 : //unsigned , default = 1 ,horizontal scaler initial repeat pixel0 number0
19:16	R/W	4	hsc_ini_rcv_num0 : //unsigned , default = 4 ,horizontal scaler initial receiving number0
15:0	R/W	0	hsc_ini_phase0 : //unsigned , default = 0 ,horizontal scaler top field initial phase0

**Table 9-1724 VD2\_SC\_MISC 0x3948**

Bit(s)	R/W	Default	Description
24	R/W	0	Repeat last line en : 1:enable repeat last line 0:disable repeat last line
22	R/W	0	hsc_len_div2_en : //unsigned , default = 0 ,if true, divide VSC line length 2 as the HSC input length, othwise VSC length length is the same as the VSC line length just for special usage, more flexibility
21	R/W	0	lbuf_mode : //unsigned , default = 0 ,if true, prevsc uses lin buffer, otherwise prevsc does not use line buffer, it should be same as prevsc_en
20	R/W	0	prehsc_en : //unsigned , default = 0 ,prehsc_en
19	R/W	0	prevsc_en : //unsigned , default = 0 ,prevsc_en
18	R/W	0	vsc_en : //unsigned , default = 0 ,vsc_en
17	R/W	0	hsc_en : //unsigned , default = 0 ,hsc_en
16	R/W	0	sc_top_en : //unsigned , default = 0 ,scale_top_en
15	R/W	0	sc_vd_en : //unsigned , default = 0 ,video1 scale out enable
12	R/W	1	hsc_nonlinear_4region_en : //unsigned , default = 1 ,if true, region0,region4 are nonlinear regions, otherwise they are not scaling regions, for horizontal scaler
10:8	R/W	0	hsc_bank_length : //unsigned , default = 0 ,horizontal scaler bank length
5	R/W	4	vsc_phase_field_mode : //unsigned , default = 4 ,vertical scaler phase field mode, if true, disable the opposite parity line output, more bandwidth needed if output 1080i
4	R/W	0	vsc_nonlinear_4region_en : //unsigned , default = 0 ,if true, region0,region4 are nonlinear regions, otherwise they are not scaling regions, for vertical scaler
2:0	R/W	4	vsc_bank_length : //unsigned , default = 4 ,vertical scaler bank length



**Table 9-1725 VD2\_SCO\_FIFO\_CTRL 0x3949**

Bit(s)	R/W	Default	Description
27:16	R/W	0	sco_fifo_line_lenm1 : //unsigned , default = 0xffff, scale out fifo line length minus 1
12:0	R/W	0	sco_fifo_size : //unsigned , default = 0x200, scale out fifo size (actually only bit 11:1 is valid, 11:1, max 1024), always even number

For 3D quincunx sub-sampling and horizontal pixel by pixel 3D interleaving.

**Table 9-1726 VD2\_HSC\_PHASE\_CTRL1 0x394a**

Bit(s)	R/W	Default	Description
27:24	R/W	0	prehsc_mode : //unsigned , default = 0,prehsc_mode, bit 3:2, prehsc odd line interp mode, bit 1:0, prehsc even line interp mode, each 2bit, 00: pix0+pix1/2, average, 01: pix1, 10: pix0
23	R/W	0	hsc_double_pix_mode : //unsigned , default = 0,horizontal scaler double pixel mode
22:21	R/W	1	hsc_rpt_p0_num1 : //unsigned , default = 1,horizontal scaler initial repeat pixel0 number1
19:16	R/W	4	hsc_ini_rcv_num1 : //unsigned , default = 4,horizontal scaler initial receiving number1
15:0	R/W	0	hsc_ini_phase1 : //unsigned , default = 0,horizontal scaler top field initial phase1

For 3D quincunx sub-sampling:

**Table 9-1727 VD2\_HSC\_INI\_PAT\_CTRL 0x394b**

Bit(s)	R/W	Default	Description
31:24	R/W	0	prehsc_pattern : //unsigned , default = 0, prehsc pattern, each patten 1 bit, from lsb -> msb
22:20	R/W	0	prehsc_pat_star : //unsigned , default = 0, prehsc pattern start
18:16	R/W	0	prehsc_pat_end : //unsigned , default = 0, prehsc pattern end
15:8	R/W	0	hsc_pattern : //unsigned , default = 0, hsc pattern, each patten 1 bit, from lsb -> msb
6:4	R/W	0	hsc_pat_start : //unsigned , default = 0, hsc pattern start
2:0	R/W	0	hsc_pat_end : //unsigned , default = 0, hsc pattern end

**Table 9-1728 VD2\_SC\_GCLK\_CTRL 0x394c**

Bit(s)	R/W	Default	Description
15:0	R/W	0	vpp_sc_gclk_ctrl : //unsigned , default = 0 ,scale clock gate

### 9.2.3.28 Supper Scale/ Sharpness Registers

**Table 9-1729 SRSHARP0\_SHARP\_HVSIZE 0x5000**

Bit(s)	R/W	Default	Description
28:16	R/W	0d1920	reg_pknr_hsize : . unsigned , default = 1920
12: 0	R/W	0d1080	reg_pknr_vsize : . unsigned , default = 1080

**Table 9-1730 SRSHARP0\_SHARP\_HVBLANK\_NUM 0x5001**

Bit(s)	R/W	Default	Description
23: 16	R/W	0d8	reg_deband_hblank : . unsigned , default = 8
15: 8	R/W	0d20	reg_pknr_hblank_num : . unsigned , default = 20
7: 0	R/W	0d60	reg_pknr_vblank_num : . unsigned , default = 60

**Table 9-1731 SRSHARP0\_NR\_GAUSSIAN\_MODE 0x5002**

Bit(s)	R/W	Default	Description
13□12	R/W	0d0	reg_nr_gau_cnorm : : Y adaptive coef norm, 0: 128, 1: 256, 2: 512, 3: 1024 .unsigned , default = 0
9□8	R/W	0d0	reg_nr_gau_cnorm : : C adaptive coef norm, 0: 128, 1: 256, 2: 512, 3: 1024 .unsigned , default = 0
5□4	R/W	0d1	reg_nr_gau_ymode : : 0 3x3 filter; 1: 5x5 filter, 2/3: adaptive coef . unsigned , default = 1
1□0	R/W	0d1	reg_nr_gau_cmode : : 0 3x3 filter; 1: 5x5 filter, 2/3: adaptive coef . unsigned , default = 1

**Table 9-1732 SRSHARP0\_PK\_CON\_2CIRHPGAIN\_TH\_RATE 0x5005**

Bit(s)	R/W	Default	Description
31:24	R/W	0d25	reg_pk_cirhpcon2gain0 : : threshold0 of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 25
23:16	R/W	0d60	reg_pk_cirhpcon2gain1 : : threshold1 of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 60
15: 8	R/W	0d80	reg_pk_cirhpcon2gain5 : : rate0 (for hpcon<th0) of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 80
7: 0	R/W	0d20	reg_pk_cirhpcon2gain6 : : rate1 (for hpcon>th1) of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 20

**Table 9-1733 SRSHARP0\_PK\_CON\_2CIRHPGAIN\_LIMIT 0x5006**

Bit(s)	R/W	Default	Description
31:24	R/W	0d96	reg_pk_cirhpcon2gain2 : : level limit(for hpcon<th0) of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 96
23:16	R/W	0d96	reg_pk_cirhpcon2gain3 : : level limit(for th0<hpcon<th1) of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 96
15: 8	R/W	0d5	reg_pk_cirhpcon2gain4 : : level limit(for hpcon>th1) of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 5

**Table 9-1734 SRSHARP0\_PK\_CON\_2CIRBPGAIN\_TH\_RATE 0x5007**

Bit(s)	R/W	Default	Description
31:24	R/W	0d20	reg_pk_cirbpcon2gain0 : : threshold0 of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 20
23:16	R/W	0d50	reg_pk_cirbpcon2gain1 : : threshold1 of curve to map bpcon to bpgain for circle bp filter (all 8 direction same).. unsigned , default = 50
15: 8	R/W	0d50	reg_pk_cirbpcon2gain5 : : rate0 (for bpcon<th0) of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 50
7: 0	R/W	0d25	reg_pk_cirbpcon2gain6 : : rate1 (for bpcon>th1) of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 25

**Table 9-1735 SRSHARP0\_PK\_CON\_2CIRBPGAIN\_LIMIT 0x5008**

Bit(s)	R/W	Default	Description
31:24	R/W	0d40	reg_pk_cirbpcon2gain2 : : level limit(for bpcon<th0) of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 40
23:16	R/W	0d40	reg_pk_cirbpcon2gain3 : : level limit(for th0<bpcon<th1) of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 40
15: 8	R/W	0d5	reg_pk_cirbpcon2gain4 : : level limit(for bpcon>th1) of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 5

**Table 9-1736 SRSHARP0\_PK\_CON\_2DRTHPGAIN\_TH\_RATE 0x5009**

Bit(s)	R/W	Default	Description
31:24	R/W	0d25	reg_pk_drthpcon2gain0 : : threshold0 of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 25
23:16	R/W	0d60	reg_pk_drthpcon2gain1 : : threshold1 of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 60
15: 8	R/W	0d80	reg_pk_drthpcon2gain5 : : rate0 (for hpcon<th0) of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 80
7: 0	R/W	0d20	reg_pk_drthpcon2gain6 : : rate1 (for hpcon>th1) of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 20

**Table 9-1737 SRSHARP0\_PK\_CON\_2DRTHPGAIN\_LIMIT 0x500a**

Bit(s)	R/W	Default	Description
31:24	R/W	0d90	reg_pk_drthpcon2gain2 : : level limit(for hpcon<th0) of curve to map hpcon to hpgain for directional hp filter (best direction).. unsigned , default = 90
23:16	R/W	0d96	reg_pk_drthpcon2gain3 : : level limit(for th0<hpcon<th1) of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 96
15: 8	R/W	0d5	reg_pk_drthpcon2gain4 : : level limit(for hpcon>th1) of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 5

**Table 9-1738 SRSHARP0\_PK\_CON\_2DRTPGAIN\_TH\_RATE 0x500b**

Bit(s)	R/W	Default	Description
31:24	R/W	0d20	reg_pk_drtbpcon2gain0 : : threshold0 of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 20
23:16	R/W	0d50	reg_pk_drtbpcon2gain1 : : threshold1 of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 50
15: 8	R/W	0d50	reg_pk_drtbpcon2gain5 : : rate0 (for bpcon<th0) of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 50
7: 0	R/W	0d25	reg_pk_drtbpcon2gain6 : : rate1 (for bpcon>th1) of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 25

**Table 9-1739 SRSHARP0\_PK\_CON\_2DRTPGAIN\_LIMIT 0x500c**

Bit(s)	R/W	Default	Description
31:24	R/W	0d40	reg_pk_drtbpcon2gain2 : : level limit(for bpcon<th0) of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 40
23:16	R/W	0d40	reg_pk_drtbpcon2gain3 : : level limit(for th0<bpcon<th1) of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 40
15: 8	R/W	0d5	reg_pk_drtbpcon2gain4 : : level limit(for bpcon>th1) of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 5

**Table 9-1740 SRSHARP0\_PK\_CIRFB\_LPF\_MODE 0x500d**

Bit(s)	R/W	Default	Description
29:28	R/W	0d1	reg_cirhp_horz_mode : : no horz filter on HP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 . unsigned , default = 1
25:24	R/W	0d1	reg_cirhp_vert_mode : : no vert filter on HP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 . unsigned , default = 1
21:20	R/W	0d1	reg_cirhp_diag_mode : : filter on HP; 1: [1 2 1]/4; . unsigned , default = 1
13:12	R/W	0d1	reg_cirbp_horz_mode : : no horz filter on BP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 . unsigned , default = 1
9: 8	R/W	0d1	reg_cirbp_vert_mode : : no vert filter on BP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 . unsigned , default = 1
5: 4	R/W	0d1	reg_cirbp_diag_mode : : filter on BP; 1: [1 2 1]/4; . unsigned , default = 1

**Table 9-1741 SRSHARP0\_PK\_DRTFB\_LPF\_MODE 0x500e**

Bit(s)	R/W	Default	Description
29:28	R/W	0d1	reg_drthp_horz_mode :: no horz filter on HP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 2 . unsigned , default = 1
25:24	R/W	0d1	reg_drthp_vert_mode :: no vert filter on HP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 2 . unsigned , default = 1
21:20	R/W	0d1	reg_drthp_diag_mode :: filter on HP; 1: [1 2 1]/4; 1 . unsigned , default = 1
13:12	R/W	0d1	reg_drtbp_horz_mode :: no horz filter on BP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 2 . unsigned , default = 1
9: 8	R/W	0d1	reg_drtbp_vert_mode :: no vert filter on BP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 2 . unsigned , default = 1
5: 4	R/W	0d1	reg_drtbp_diag_mode :: filter on BP; 1: [1 2 1]/4; 1 . unsigned , default = 1

**Table 9-1742 SRSHARP0\_PK\_CIRFB\_HP\_CORING 0x500f**

Bit(s)	R/W	Default	Description
21:16	R/W	0d4	reg_cirhp_horz_core :: coring of HP for Horz . unsigned , default = 4
13: 8	R/W	0d4	reg_cirhp_vert_core :: coring of HP for Vert . unsigned , default = 4
5: 0	R/W	0d4	reg_cirhp_diag_core :: coring of HP for Diag . unsigned , default = 4

**Table 9-1743 SRSHARP0\_PK\_CIRFB\_BP\_CORING 0x5010**

Bit(s)	R/W	Default	Description
21:16	R/W	0d4	reg_cirbp_horz_core :: coring of HP for Horz . unsigned , default = 4
13: 8	R/W	0d4	reg_cirbp_vert_core :: coring of HP for Vert . unsigned , default = 4
5: 0	R/W	0d4	reg_cirbp_diag_core :: coring of HP for Diag . unsigned , default = 4

**Table 9-1744 SRSHARP0\_PK\_DRTFB\_HP\_CORING 0x3e11**

Bit(s)	R/W	Default	Description
21:16	R/W	0d4	reg_drthp_horz_core :: coring of HP for Horz . unsigned , default = 4
13: 8	R/W	0d4	reg_drthp_vert_core :: coring of HP for Vert . unsigned , default = 4
5: 0	R/W	0d4	reg_drthp_diag_core :: coring of HP for Diag . unsigned , default = 4

**Table 9-1745 SRSHARP0\_PK\_DRTFB\_BP\_CORING 0x5012**

Bit(s)	R/W	Default	Description
21:16	R/W	0d4	reg_drtbp_horz_core :: coring of HP for Horz . unsigned , default = 4
13: 8	R/W	0d4	reg_drtbp_vert_core :: coring of HP for Vert . unsigned , default = 4
5: 0	R/W	0d4	reg_drtbp_diag_core :: coring of HP for Diag . unsigned , default = 4

**Table 9-1746 SRSHARP0\_PK\_CIRFB\_BLEND\_GAIN 0x5013**

Bit(s)	R/W	Default	Description
31:28	R/W	0d8	reg_hp_cir_hgain :: normalized 8 as '1' . unsigned , default = 8
27:24	R/W	0d8	reg_hp_cir_vgain :: normalized 8 as '1' . unsigned , default = 8
23:20	R/W	0d8	reg_hp_cir_dgain :: normalized 8 as '1' . unsigned , default = 8
15:12	R/W	0d8	reg_bp_cir_hgain :: normalized 8 as '1' . unsigned , default = 8
11: 8	R/W	0d8	reg_bp_cir_vgain :: normalized 8 as '1' . unsigned , default = 8
7: 4	R/W	0d8	reg_bp_cir_dgain :: normalized 8 as '1' . unsigned , default = 8

**Table 9-1747 SRSHARP0\_NR\_ALPY\_SSD\_GAIN\_OFST 0x5014**

Bit(s)	R/W	Default	Description
15: 8	R/W	0d16	reg_nr_alp0_ssd_gain :: gain to max ssd normalized 16 as '1' . unsigned , default = 16
5: 0	R/W	0x0	reg_nr_alp0_ssd_ofst :: offset to ssd before dividing to min_err . signed , default = -2

**Table 9-1748 SRSHARP0\_NR\_ALP0Y\_ERR2CURV\_TH\_RATE 0x5015**

Bit(s)	R/W	Default	Description
31:24	R/W	0d10	reg_nr_alp0_minerr_ypar0 :: threshold0 of curve to map mierr to alp0 for luma channel, this will be set value of flat region mierr that no need blur. 0~255.. unsigned , default = 10
23:16	R/W	0d25	reg_nr_alp0_minerr_ypar1 :: threshold1 of curve to map mierr to alp0 for luma channel, this will be set value of texture region mierr that can not blur.. unsigned , default = 25
15: 8	R/W	0d80	reg_nr_alp0_minerr_ypar5 :: rate0 (for mierr<th0) of curve to map mierr to alp0 for luma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 80
7: 0	R/W	0d64	reg_nr_alp0_minerr_ypar6 :: rate1 (for mierr>th1) of curve to map mierr to alp0 for luma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 64

**Table 9-1749 SRSHARP0\_NR\_ALP0Y\_ERR2CURV\_LIMIT 0x5016**

Bit(s)	R/W	Default	Description
31:24	R/W	0d63	reg_nr_alp0_minerr_ypar2 :: level limit(for mierr<th0) of curve to map mierr to alp0 for luma channel, this will be set to alp0 that we can do for flat region. 0~255.. unsigned , default = 63
23:16	R/W	0d0	reg_nr_alp0_minerr_ypar3 :: level limit(for th0<mierr<th1) of curve to map mierr to alp0 for luma channel, this will be set to alp0 that we can do for misc region. 0~255.. unsigned , default = 0
15: 8	R/W	0d63	reg_nr_alp0_minerr_ypar4 :: level limit(for mierr>th1) of curve to map mierr to alp0 for luma channel, this will be set to alp0 that we can do for texture region. 0~255.. unsigned , default = 63

**Table 9-1750 SRSHARP0\_NR\_ALP0C\_ERR2CURV\_TH\_RATE 0x5017**

Bit(s)	R/W	Default	Description
31:24	R/W	0d10	reg_nr_alp0_minerr_cpar0 : : threshold0 of curve to map mierr to alp0 for chroma channel, this will be set value of flat region mierr that no need blur.. unsigned , default = 10
23:16	R/W	0d25	reg_nr_alp0_minerr_cpar1 : : threshold1 of curve to map mierr to alp0 for chroma channel, this will be set value of texture region mierr that can not blur.. unsigned , default = 25
15: 8	R/W	0d80	reg_nr_alp0_minerr_cpar5 : : rate0 (for mierr<th0) of curve to map mierr to alp0 for chroma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 80
7: 0	R/W	0d64	reg_nr_alp0_minerr_cpar6 : : rate1 (for mierr>th1) of curve to map mierr to alp0 for chroma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 64

**Table 9-1751 SRSHARP0\_NR\_ALP0C\_ERR2CURV\_LIMIT 0x5018**

Bit(s)	R/W	Default	Description
31:24	R/W	0d63	reg_nr_alp0_minerr_cpar2 : : level limit(for mierr<th0) of curve to map mierr to alp0 for chroma channel, this will be set to alp0 that we can do for flat region. 0~255.. unsigned , default = 63
23:16	R/W	0d0	reg_nr_alp0_minerr_cpar3 : : level limit(for th0<mierr<th1) of curve to map mierr to alp0 for chroma channel, this will be set to alp0 that we can do for misc region. 0~255.. unsigned , default = 0
15: 8	R/W	0d63	reg_nr_alp0_minerr_cpar4 : : level limit(for mierr>th1) of curve to map mierr to alp0 for chroma channel, this will be set to alp0 that we can do for texture region. 0~255.. unsigned , default = 63

**Table 9-1752 SRSHARP0\_NR\_ALP0\_MIN\_MAX 0x5019**

Bit(s)	R/W	Default	Description
29:24	R/W	0d2	reg_nr_alp0_ymin : : normalized to 64 as '1' . unsigned , default = 2
21:16	R/W	0d63	reg_nr_alp0_ymax : : normalized to 64 as '1' . unsigned , default = 63
13: 8	R/W	0d2	reg_nr_alp0_cmin : : normalized to 64 as '1' . unsigned , default = 2
5: 0	R/W	0d63	reg_nr_alp0_cmax : : normalized to 64 as '1' . unsigned , default = 63

**Table 9-1753 SRSHARP0\_NR\_ALP1\_MIERR\_CORING 0x501a**

Bit(s)	R/W	Default	Description
16	R/W	0d0	reg_nr_alp1_maxerr_mode : : 0 max err; 1: xerr . unsigned , default = 0
13: 8	R/W	0d0	reg_nr_alp1_core_rate : : normalized 64 as "1" . unsigned , default = 0
5: 0	R/W	0d3	reg_nr_alp1_core_ofst : : normalized 64 as "1" . signed , default = 3

**Table 9-1754 SRSHARP0\_NR\_ALP1\_ERR2CURV\_TH\_RATE 0x501b**

Bit(s)	R/W	Default	Description
31:24	R/W	0d0	reg_nr_alp1_minerr_par0 :: threshold0 of curve to map mierr to alp1 for luma/chroma channel, this will be set value of flat region mierr that no need directional NR. 0~255.. unsigned , default = 0
23:16	R/W	0d24	reg_nr_alp1_minerr_par1 :: threshold1 of curve to map mierr to alp1 for luma/chroma channel, this will be set value of texture region mierr that can not do directional NR. 0~255.. unsigned , default = 24
15: 8	R/W	0d0	reg_nr_alp1_minerr_par5 :: rate0 (for mierr<th0) of curve to map mierr to alp1 for luma/chroma channel. the larger of the value, the deep of the slope.. unsigned , default = 0
7: 0	R/W	0d20	reg_nr_alp1_minerr_par6 :: rate1 (for mierr>th1) of curve to map mierr to alp1 for luma/chroma channel. the larger of the value, the deep of the slope. 0~255. unsigned , default = 20

**Table 9-1755 SRSHARP0\_NR\_ALP1\_ERR2CURV\_LIMIT 0x501c**

Bit(s)	R/W	Default	Description
31:24	R/W	0d0	reg_nr_alp1_minerr_par2 :: level limit(for mierr<th0) of curve to map mierr to alp1 for luma/chroma channel, this will be set to alp1 that we can do for flat region. 0~255.. unsigned , default = 0
23:16	R/W	0d16	reg_nr_alp1_minerr_par3 :: level limit(for th0<mierr<th1) of curve to map mierr to alp1 for luma/chroma channel, this will be set to alp1 that we can do for misc region. 0~255.. unsigned , default = 16
15: 8	R/W	0d63	reg_nr_alp1_minerr_par4 :: level limit(for mierr>th1) of curve to map mierr to alp1 for luma/chroma channel, this will be set to alp1 that we can do for texture region. 0~255.255 before. unsigned , default = 63

**Table 9-1756 SRSHARP0\_NR\_ALP1\_MIN\_MAX 0x501d**

Bit(s)	R/W	Default	Description
29:24	R/W	0d0	reg_nr_alp1_ymin :: normalized to 64 as '1' . unsigned , default = 0
21:16	R/W	0d63	reg_nr_alp1_ymax :: normalized to 64 as '1' . unsigned , default = 63
13: 8	R/W	0d0	reg_nr_alp1_cmin :: normalized to 64 as '1' . unsigned , default = 0
5: 0	R/W	0d63	reg_nr_alp1_cmax :: normalized to 64 as '1' . unsigned , default = 63

**Table 9-1757 SRSHARP0\_PK\_ALP2\_MIERR\_CORING 0x501e**

Bit(s)	R/W	Default	Description
16	R/W	0d1	reg_pk_alp2_maxerr_mode :: 0 max err; 1: xerr . unsigned , default = 1
13: 8	R/W	0d13	reg_pk_alp2_core_rate :: normalized 64 as "1" . unsigned , default = 13
5: 0	R/W	0d1	reg_pk_alp2_core_ofst :: normalized 64 as "1" . signed , default = 1



**Table 9-1758 SRSHARP0\_PK\_ALP2\_ERR2CURV\_TH\_RATE 0x501f**

Bit(s)	R/W	Default	Description
31:24	R/W	0d0	reg_pk_alp2_minerr_par0 : : threshold0 of curve to map mierr to alp2 for luma channel, this will be set value of flat region mierr that no need peaking.. unsigned , default = 0
23:16	R/W	0d24	reg_pk_alp2_minerr_par1 : : threshold1 of curve to map mierr to alp2 for luma channel, this will be set value of texture region mierr that can not do peaking. 0~255.. unsigned , default = 24
15: 8	R/W	0d0	reg_pk_alp2_minerr_par5 : : rate0 (for mierr<th0) of curve to map mierr to alp2 for luma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 0
7: 0	R/W	0d20	reg_pk_alp2_minerr_par6 : : rate1 (for mierr>th1) of curve to map mierr to alp2 for luma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 20

**Table 9-1759 SRSHARP0\_PK\_ALP2\_ERR2CURV\_LIMIT 0x5020**

Bit(s)	R/W	Default	Description
31:24	R/W	0d0	reg_pk_alp2_minerr_par2 : : level limit(for mierr<th0) of curve to map mierr to alp2 for luma channel, this will be set to alp2 that we can do for flat region. 0~255.. unsigned , default = 0
23:16	R/W	0d16	reg_pk_alp2_minerr_par3 : : level limit(for th0<mierr<th1) of curve to map mierr to alp2 for luma channel, this will be set to alp2 that we can do for misc region. 0~255.. unsigned , default = 16
15: 8	R/W	0d63	reg_pk_alp2_minerr_par4 : : level limit(for mierr>th1) of curve to map mierr to alp2 for luma channel, this will be set to alp2 that we can do for texture region. 0~255. default = 63;. unsigned , default = 255

**Table 9-1760 SRSHARP0\_PK\_ALP2\_MIN\_MAX 0x5021**

Bit(s)	R/W	Default	Description
13: 8	R/W	0d0	reg_pk_alp2_min : : normalized to 64 as '1' . unsigned , default = 0
5: 0	R/W	0d63	reg_pk_alp2_max : : normalized to 64 as '1' . unsigned , default = 63

**Table 9-1761 SRSHARP0\_PK\_FINALGAIN\_HP\_BP 0x5022**

Bit(s)	R/W	Default	Description
17: 16	R/W	0d0	reg_final_gain_rs : : right shift bits for the gain normalization, 0 normal to 32 as 1; 1 normal to 64 as 1; -2 normal to 8 as 1; -1 normal to 16 as 1 . signed , default = 0
15: 8	R/W	0d40	reg_hp_final_gain : : gain to highpass boost result (including directional/circle blending), normalized 32 as '1', 0~255. 1.25 * 32. unsigned , default = 40
7: 0	R/W	0d30	reg_bp_final_gain : : gain to bandpass boost result (including directional/circle blending), normalized 32 as '1', 0~255. 1.25 * 32. unsigned , default = 30

**Table 9-1762 SRSHARP0\_PK\_OS\_HORZ\_CORE\_GAIN 0x5023**

Bit(s)	R/W	Default	Description
31:24	R/W	0d8	reg_pk_os_hsidecore : : side coring (not to current pixel) to adaptive overshoot margin in horizontal direction. the larger of this value, the less overshoot admitted 0~255;. unsigned , default = 8
23:16	R/W	0d20	reg_pk_os_hsidegain : : side gain (not to current pixel) to adaptive overshoot margin in horizontal direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20
15: 8	R/W	0d2	reg_pk_os_hmidcore : : mid coring (to current pixel) to adaptive overshoot margin in horizontal direction. the larger of this value, the less overshoot admitted 0~255;. unsigned , default = 2
7: 0	R/W	0d20	reg_pk_os_hmidgain : : mid gain (to current pixel) to adaptive overshoot margin in horizontal direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20

**Table 9-1763 SRSHARP0\_PK\_OS\_VERT\_CORE\_GAIN 0x5024**

Bit(s)	R/W	Default	Description
31:24	R/W	0d8	reg_pk_os_vsidecore : : side coring (not to current pixel) to adaptive overshoot margin in vertical direction. the larger of this value, the less overshoot admitted 0~255;. unsigned , default = 8
23:16	R/W	0d20	reg_pk_os_vsidegain : : side gain (not to current pixel) to adaptive overshoot margin in vertical direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20
15: 8	R/W	0d2	reg_pk_os_vmidcore : : mid coring (to current pixel) to adaptive overshoot margin in vertical direction. the larger of this value, the less overshoot admitted 0~255;. unsigned , default = 2
7: 0	R/W	0d20	reg_pk_os_vmidgain : : mid gain (to current pixel) to adaptive overshoot margin in vertical direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20

**Table 9-1764 SRSHARP0\_PK\_OS\_ADPT\_MISC 0x5025**

Bit(s)	R/W	Default	Description
31:24	R/W	0d40	reg_pk_os_minerr_core : : coring to minerr for adaptive overshoot margin. the larger of this value, the less overshoot admitted 0~255;. unsigned , default = 40
23:16	R/W	0d6	reg_pk_os_minerr_gain : : gain to minerr based adaptive overshoot margin. normalized to 64 as '1'. 0~255;. unsigned , default = 6
15: 8	R/W	0d200	reg_pk_os_adpt_max : : maximum limit adaptive overshoot margin (4x). 0~255; . unsigned , default = 200
7: 0	R/W	0d20	reg_pk_os_adpt_min : : minimum limit adaptive overshoot margin (1x). 0~255; . unsigned , default = 20

**Table 9-1765 SRSHARP0\_PK\_OS\_STATIC 0x5026**

Bit(s)	R/W	Default	Description
29:28	R/W	0d2	reg_pk_osh_mode : : 0~3: (2x+1) window in H direction . unsigned , default = 2
25:24	R/W	0d2	reg_pk_osv_mode : : 0~3: (2x+1) window in V direction . unsigned , default = 2

Bit(s)	R/W	Default	Description
21:12	R/W	0d200	reg_pk_os_down : : static negative overshoot margin. 0~1023; . unsigned , default = 200
9: 0	R/W	0d200	reg_pk_os_up : : static positive overshoot margin. 0~1023; . unsigned , default = 200

Table 9-1766 SRSHARP0\_PK\_NR\_ENABLE 0x5027

Bit(s)	R/W	Default	Description
3: 2	R/W	0d0	reg_3d_mode : , 0: no 3D; 1: L/R; 2: T/B; 3: horizontal interleaved, dft = 0 // . unsigned , default = 0
1	R/W	0d1	reg_pk_en : . unsigned , default = 1
0	R/W	0d1	reg_nr_en : . unsigned , default = 1

Table 9-1767 SRSHARP0\_PK\_DRT\_SAD\_MISC 0x5028

Bit(s)	R/W	Default	Description
31:24	R/W	0d24	reg_pk_sad_ver_gain : : gain to sad[4], 16 normalized to "1"; . unsigned , default = 24
23:16	R/W	0d24	reg_pk_sad_hor_gain : : gain to sad[0], 16 normalized to "1"; . unsigned , default = 24
10: 9	R/W	0d0	reg_pk_bias_diag : : bias towards diag . unsigned , default = 0
4: 0	R/W	0d24	reg_pk_drt_force : : force direction of drt peaking filter, h2b: 0:hp drt force, 1: bp drt force; 2: bp+hp drt force, 3: no force;. unsigned , default = 24

Table 9-1768 SRSHARP0\_NR\_TI\_DNLP\_BLEND 0x5029

Bit(s)	R/W	Default	Description
10: 8	R/W	0d4	reg_dnlp_input_mode : : dnlp input options. 0: org_y; 1: gau_y; 2: gauadp_y; 3: edgadplpf_y; 4: nr_y;5: lti_y; 6: pk_y (before os);7: pk_y (after os). unsigned , default = 4
3: 2	R/W	0d1	reg_nr_cti_blend_mode : : blend mode of nr and lti result: 0: nr; 1:cti; 2: (nr+cti)/2; 3:cti + dlt_nr . unsigned , default = 1
1: 0	R/W	0d1	reg_nr_lti_blend_mode : : blend mode of nr and lti result: 0: nr; 1:lti; 2: (nr+lti)/2; 3:lti + dlt_nr . unsigned , default = 1

Table 9-1769 SRSHARP0\_TI\_DIR\_CORE\_ALPHA 0x502a

Bit(s)	R/W	Default	Description
29:24	R/W	0d10	reg_adp_lti_dir_alp_core_ofst : : ofst to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err*64; dft=10. unsigned , default = 10
19:16	R/W	0d0	reg_adp_lti_dir_alp_core_rate : : ofset to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err*64; dft=0/32. unsigned , default = 0
13: 8	R/W	0d0	reg_adp_lti_dir_alpmin : : min value of alpha, alpha = (min_err+x +ofst)/max_err*64; dft=10 . unsigned , default = 0
5: 0	R/W	0d63	reg_adp_lti_dir_alpmax : : max value of alpha, alpha = (min_err+x +ofst)/max_err*64; dft=63 . unsigned , default = 63

Table 9-1770 SRSHARP0\_CTI\_DIR\_ALPHA 0x502b

Bit(s)	R/W	Default	Description
29:24	R/W	0d5	reg_adp_cti_dir_alp_core_ofst : : ofst to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err*64; dft=10. unsigned , default = 5
19:16	R/W	0d0	reg_adp_cti_dir_alp_core_rate : : ofset to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err*64; dft=0/32. unsigned , default = 0
13: 8	R/W	0d0	reg_adp_cti_dir_alpmin : : min value of alpha, alpha = (min_err +x+ofst)/max_err*64; dft=10 . unsigned , default = 0
5: 0	R/W	0d63	reg_adp_cti_dir_alpmax : : max value of alpha, alpha = (min_err +x+ofst)/max_err*64; dft=63 . unsigned , default = 63

Table 9-1771 SRSHARP0\_LTI\_CTI\_DF\_GAIN 0x502c

Bit(s)	R/W	Default	Description
29:24	R/W	0d16	reg_adp_lti_hdf_gain : : 8 normalized to "1"; default = 16 . unsigned , default = 16
21:16	R/W	0d12	reg_adp_lti_vdf_gain : : 8 normalized to "1"; default = 12 . unsigned , default = 12
13: 8	R/W	0d16	reg_adp_cti_hdf_gain : : 8 normalized to "1"; default = 16 . unsigned , default = 16
5: 0	R/W	0d12	reg_adp_cti_vdf_gain : : 8 normalized to "1"; default = 12 . unsigned , default = 12

Table 9-1772 SRSHARP0\_LTI\_CTI\_DIR\_AC\_DBG 0x502d

Bit(s)	R/W	Default	Description
30	R/W	0d1	reg_adp_lti_dir_lpf : : 0: no lpf; 1: [1 2 2 2 1]/8 lpf . unsigned , default = 1
28	R/W	0d0	reg_adp_lti_dir_difmode : : 0: y_dif; 1: y_dif + (u_dif+v_dif)/2; . unsigned , default = 0
26	R/W	0d1	reg_adp_cti_dir_lpf : : 0: no lpf; 1: [1 2 2 2 1]/8 lpf dft=1 . unsigned , default = 1
25:24	R/W	0d0	reg_adp_cti_dir_difmode : : 0: (u_dif+v_dif); 1: y_dif/2 + (u_dif+v_dif)*3/4; 2: y_dif + (u_dif+v_dif)/2; 3: y_dif*2 (not recommended). unsigned , default = 0
23:22	R/W	0d3	reg_adp_hvlti_dcblend_mode : : 0: hlti_dc; 1:vlti_dc; 2: avg 3; blend on alpha . unsigned , default = 3
21:20	R/W	0d3	reg_adp_hvcti_dcblend_mode : : 0: hcti_dc; 1:vcti_dc; 2: avg 3; blend on alpha . unsigned , default = 3
19:18	R/W	0d3	reg_adp_hvlti_acblend_mode : : hlti_ac; 1:vlti_ac; 2: add 3;;adaptive to alpha . unsigned , default = 3
17:16	R/W	0d3	reg_adp_hvcti_acblend_mode : : hcti_ac; 1:vcti_ac; 2: add 3;; adaptive to alpha . unsigned , default = 3
14:12	R/W	0d0	reg_adp_hlti_debug : , for hlti debug, default = 0 . unsigned , default = 0
10: 8	R/W	0d0	reg_adp_vlti_debug : , for vlti debug, default = 0 . unsigned , default = 0
6: 4	R/W	0d0	reg_adp_hcti_debug : , for hcti debug, default = 0 . unsigned , default = 0
2: 0	R/W	0d0	reg_adp_vcti_debug : , for vcti debug, default = 0 . unsigned , default = 0

Table 9-1773 SRSHARP0\_HCTI\_FLT\_CLP\_DC 0x502e

Bit(s)	R/W	Default	Description
28	R/W	0d1	reg_adp_hcti_en : , 0: no cti, 1: new cti, default = 1 . unsigned , default = 1
27:26	R/W	0d3	reg_adp_hcti_vdnflt : , 0: no lpf; 1:[0,2,4,2,0], 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
25:24	R/W	0d2	reg_adp_hcti_hdnflt : , 0: no lpf; 1:[0, 0, 0, 4, 8, 4, 0, 0, 0], 2:[0, 0, 2, 4, 4, 4, 2, 0, 0], 3: [1, 2, 2, 2, 2, 2, 2, 1], default = 2. unsigned , default = 2
23:22	R/W	0d3	reg_adp_hcti_ddnflt : , 0: no lpf; 1:[0,2,4,2,0], 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
21:20	R/W	0d2	reg_adp_hcti_lpf0flt : , 0:no filter; 1:sigma=0.75, 2: sigma = 1.0, 3: sigma = 1.5, default = 2 . unsigned , default = 2
19:18	R/W	0d2	reg_adp_hcti_lpf1flt : , 0:no filter; 1:sigma= 2.0, 2: sigma = 3.0, 3: sigma = 4.0, default = 2 . unsigned , default = 2
17:16	R/W	0d2	reg_adp_hcti_lpf2flt : , 0:no filter; 1:sigma=5.0, 2: sigma = 9.0, 3: sigma = 13.0, default = 2 . unsigned , default = 2
15:12	R/W	0d7	reg_adp_hcti_hard_clp_win : , window size, 0~8, default = 7 . unsigned , default = 7
11: 8	R/W	0d3	reg_adp_hcti_hard_win_min : , window size, 0~8, default = 3 . unsigned , default = 3
4	R/W	0d1	reg_adp_hcti_clp_mode : , 0: hard clip, 1: adaptive clip, default = 1 . unsigned , default = 1
2: 0	R/W	0d0	reg_adp_hcti_dc_mode : , 0:dn, 1:lpf0, 2:lpf1, 3:lpf2, 4: lpf3: 5: vdn result; 6/7: org, default = 0 . unsigned , default = 0

Table 9-1774 SRSHARP0\_HCTI\_BST\_GAIN 0x502f

Bit(s)	R/W	Default	Description
31:24	R/W	0d80	reg_adp_hcti_bst_gain0 : : gain of the bandpass 0 (lpf1-lpf2)- LBP, default = 80 . unsigned , default = 80
23:16	R/W	0d96	reg_adp_hcti_bst_gain1 : : gain of the bandpass 1 (lpf0-lpf1)- BP, default = 96 . unsigned , default = 96
15: 8	R/W	0d64	reg_adp_hcti_bst_gain2 : : gain of the bandpass 2 (hdn-lpf0)- HP, default = 64 . unsigned , default = 64
7: 0	R/W	0d16	reg_adp_hcti_bst_gain3 : : gain of the unsharp band (yuvin-hdn) - US, default = 16 . unsigned , default = 16

Table 9-1775 SRSHARP0\_HCTI\_BST\_CORE 0x5030

Bit(s)	R/W	Default	Description
31:24	R/W	0d0	reg_adp_hcti_bst_core0 : : core of the bandpass 0 (lpf1-lpf2)- LBP, default = 0 . unsigned , default = 0
23:16	R/W	0d0	reg_adp_hcti_bst_core1 : : core of the bandpass 1 (lpf0-lpf1)- BP, default = 0 . unsigned , default = 0
15: 8	R/W	0d0	reg_adp_hcti_bst_core2 : : core of the bandpass 2 (hdn-lpf0)- HP, default = 0 . unsigned , default = 0
7: 0	R/W	0d0	reg_adp_hcti_bst_core3 : : core of the unsharp band (yuvin-hdn) - US, default = 0 . unsigned , default = 0

**Table 9-1776 SRSHARP0\_HCTI\_CON\_2\_GAIN\_0 0x5031**

Bit(s)	R/W	Default	Description
31:29	R/W	0d2	reg_adp_hcti_con_mode : : con mode 0:[0, 0,-1, 1, 0, 0, 0]+[0, 0, 0, 1,-1, 0, 0], 1:[0, 0,-1, 0, 1, 0, 0], 2:[0,-1, 0, 0, 0, 1, 0], 3:[-1, 0, 0, 0, 0, 0, 1], 4: ..... default = 2. unsigned , default = 2
28:26	R/W	0d3	reg_adp_hcti_dx_mode : : dx mode 0: [-1 1 0]; 1~7: [-1 (2x+1)"0" 1], default = 3 . unsigned , default = 3
25:24	R/W	0d1	reg_adp_hcti_con_lpf : : lpf mode of the con: 0: [1 2 1]/4; 1:[1 2 2 1]/8, default = 1 . unsigned , default = 1
23:16	R/W	0d25	reg_adp_hcti_con_2_gain0 : , default = 25 . unsigned , default = 25
15: 8	R/W	0d60	reg_adp_hcti_con_2_gain1 : , default = 60 . unsigned , default = 60
7: 0	R/W	0d0	reg_adp_hcti_con_2_gain2 : 0;, default = 0 . unsigned , default = 0

**Table 9-1777 SRSHARP0\_HCTI\_CON\_2\_GAIN\_1 0x5032**

Bit(s)	R/W	Default	Description
31:24	R/W	0d96	reg_adp_hcti_con_2_gain3 : 96;, default = 96 . unsigned , default = 96
23:16	R/W	0d5	reg_adp_hcti_con_2_gain4 : 5;, default = 5 . unsigned , default = 5
15: 8	R/W	0d80	reg_adp_hcti_con_2_gain5 : 80;, default = 80 . unsigned , default = 80
7: 0	R/W	0d20	reg_adp_hcti_con_2_gain6 : 20;, default = 20 . unsigned , default = 20

**Table 9-1778 SRSHARP0\_HCTI\_OS\_MARGIN 0x5033**

Bit(s)	R/W	Default	Description
7: 0	R/W	0d0	reg_adp_hcti_os_margin : : margin for hcti overshoot, default = 0 . unsigned , default = 0

**Table 9-1779 SRSHARP0\_HLTI\_FLT\_CLP\_DC 0x5034**

Bit(s)	R/W	Default	Description
28	R/W	0d1	reg_adp_hlti_en : , 0: no cti, 1: new cti, default = 1 . unsigned , default = 1
27:26	R/W	0d2	reg_adp_hlti_vdn_fit : , 0: no lpf; 1:[0,2,4,2,0], 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 2 . unsigned , default = 2
25:24	R/W	0d2	reg_adp_hlti_hdn_fit : , 0: no lpf; 1:[0, 0, 0, 4, 8, 4, 0, 0, 0], 2:[0, 0, 2, 4, 4, 4, 2, 0, 0], 3: [1, 2, 2, 2, 2, 2, 2, 2, 1], default = 2. unsigned , default = 2
23:22	R/W	0d2	reg_adp_hlti_ddn_fit : , 0: no lpf; 1:[0,2,4,2,0], 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 2 . unsigned , default = 2
21:20	R/W	0d2	reg_adp_hlti_lpf0_fit : , 0:no filter; 1:sigma=0.75, 2: sigma = 1.0, 3: sigma = 1.5, default = 2 . unsigned , default = 2
19:18	R/W	0d2	reg_adp_hlti_lpf1_fit : , 0:no filter; 1:sigma= 2.0, 2: sigma = 3.0, 3: sigma = 4.0, default = 2 . unsigned , default = 2
17:16	R/W	0d2	reg_adp_hlti_lpf2_fit : , 0:no filter; 1:sigma=5.0, 2: sigma = 9.0, 3: sigma = 13.0, default = 2 . unsigned , default = 2

Bit(s)	R/W	Default	Description
15:12	R/W	0d2	reg_adp_hlti_hard_clp_win : , window size, 0~8, default = 2 . unsigned , default = 2
11: 8	R/W	0d1	reg_adp_hlti_hard_win_min : , window size, 0~8, default = 1 . unsigned , default = 1
4	R/W	0d0	reg_adp_hlti_clp_mode : , 0: hard clip, 1: adaptive clip, default = 0 . unsigned , default = 0
2: 0	R/W	0d4	reg_adp_hlti_dc_mode : , 0:dn, 1:lpf0, 2:lpf1, 3:lpf2, 4: lpf3: 5: vdn result; 6/7:org, default = 4 . unsigned , default = 4

**Table 9-1780 SRSHARP0\_HLTI\_BST\_GAIN 0x5035**

Bit(s)	R/W	Default	Description
31:24	R/W	0d40	reg_adp_hlti_bst_gain0 : : gain of the bandpass 0 (lpf1-lpf2)- LBP, default = 40 . unsigned , default = 40
23:16	R/W	0d48	reg_adp_hlti_bst_gain1 : : gain of the bandpass 1 (lpf0-lpf1)- BP, default = 48 . unsigned , default = 48
15: 8	R/W	0d32	reg_adp_hlti_bst_gain2 : : gain of the bandpass 2 (hdn-lpf0)- HP, default = 32 . unsigned , default = 32
7: 0	R/W	0d16	reg_adp_hlti_bst_gain3 : : gain of the unsharp band (yuvin-hdn) - US, default = 16 . unsigned , default = 16

**Table 9-1781 SRSHARP0\_HLTI\_BST\_CORE 0x5036**

Bit(s)	R/W	Default	Description
31:24	R/W	0d5	reg_adp_hlti_bst_core0 : : core of the bandpass 0 (lpf1-lpf2)- LBP, default = 5 . unsigned , default = 5
23:16	R/W	0d5	reg_adp_hlti_bst_core1 : : core of the bandpass 1 (lpf0-lpf1)- BP, default = 5 . unsigned , default = 5
15: 8	R/W	0d5	reg_adp_hlti_bst_core2 : : core of the bandpass 2 (hdn-lpf0)- HP, default = 5 . unsigned , default = 5
7: 0	R/W	0d3	reg_adp_hlti_bst_core3 : : core of the unsharp band (yuvin-hdn) - US, default = 3 . unsigned , default = 3

**Table 9-1782 SRSHARP0\_HLTI\_CON\_2\_GAIN\_0 0x5037**

Bit(s)	R/W	Default	Description
31:29	R/W	0d2	reg_adp_hlti_con_mode : : con mode 0:[0, 0,-1, 1, 0, 0, 0]+[0, 0, 0, 1,-1, 0, 0], 1: [0, 0,-1, 0, 1, 0, 0], 2: [0,-1, 0, 0, 0, 1, 0], 3:[-1, 0, 0, 0, 0, 0, 1], 4: ....., default = 2 . unsigned , default = 2
28:26	R/W	0d3	reg_adp_hlti_dx_mode : : dx mode 0: [-1 1 0]; 1~7: [-1 (2x+1)"0" 1], default = 3 . unsigned , default = 3
25:24	R/W	0d1	reg_adp_hlti_con_lpf : : lpf mode of the con: 0: [1 2 1]/4; 1:[1 2 2 1]/8, default = 1 . unsigned , default = 1
23:16	R/W	0d25	reg_adp_hlti_con_2_gain0 : 25;, default = 25 . unsigned , default = 25
15: 8	R/W	0d60	reg_adp_hlti_con_2_gain1 : 60;, default = 60 . unsigned , default = 60
7: 0	R/W	0d90	reg_adp_hlti_con_2_gain2 : 0;, default = 90 . unsigned , default = 90

**Table 9-1783 SRSHARP0\_HLTI\_CON\_2\_GAIN\_1 0x5038**

Bit(s)	R/W	Default	Description
31:24	R/W	0d96	reg_adp_hlti_con_2_gain3 : 96;, default = 96 . unsigned , default = 96
23:16	R/W	0d95	reg_adp_hlti_con_2_gain4 : 5;, default = 95 . unsigned , default = 95
15: 8	R/W	0d80	reg_adp_hlti_con_2_gain5 : 80;, default = 80 . unsigned , default = 80
7: 0	R/W	0d20	reg_adp_hlti_con_2_gain6 : 20;, default = 20 . unsigned , default = 20

**Table 9-1784 SRSHARP0\_HLTI\_OS\_MARGIN 0x5039**

Bit(s)	R/W	Default	Description
7: 0	R/W	0d0	reg_adp_hlti_os_margin : : margin for hlti overshoot, default = 0 . unsigned , default = 0

**Table 9-1785 SRSHARP0\_VLTI\_FLT\_CON\_CLP 0x503a**

Bit(s)	R/W	Default	Description
14	R/W	0d1	reg_adp_vlti_en : : enable bit of vlti, default = 1 . unsigned , default = 1
13:12	R/W	0d3	reg_adp_vlti_hxn_fit : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
11:10	R/W	0d3	reg_adp_vlti_dxn_fit : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
9: 8	R/W	0d3	reg_adp_vlti_han_fit : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
7: 6	R/W	0d3	reg_adp_vlti_dan_fit : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
5: 4	R/W	0d2	reg_adp_vlti_dx_mode : : 0:[-1 1] 1:[-1 0 -1]; 2/3: [-1 0 0 0 -1], default = 2 . unsigned , default = 2
2	R/W	0d1	reg_adp_vlti_con_lpf : : lpf mode of the con: 0: [1 2 1]/4; 1:[1 2 2 2 1]/8, default = 1 . unsigned , default = 1
0	R/W	0d1	reg_adp_vlti_hard_clp_win : : window size; 0: 1x3 window; 1: 1x5 window, default = 1 . unsigned , default = 1

**Table 9-1786 SRSHARP0\_VLTI\_BST\_GAIN 0x503b**

Bit(s)	R/W	Default	Description
23:16	R/W	0d32	reg_adp_vlti_bst_gain0 : : gain to boost filter [-1 2 -1];, default = 32 . unsigned , default = 32
15: 8	R/W	0d32	reg_adp_vlti_bst_gain1 : : gain to boost filter [-1 0 2 0 -1];, default = 32 . unsigned , default = 32
7: 0	R/W	0d32	reg_adp_vlti_bst_gain2 : : gain to boost filter usf, default = 32 . unsigned , default = 32



**Table 9-1787 SRSHARP0\_VLTI\_BST\_CORE 0x503c**

Bit(s)	R/W	Default	Description
23:16	R/W	0d5	reg_adp_vlti_bst_core0 : : coring to boost filter [-1 2 -1];, default = 5 . unsigned , default = 5
15: 8	R/W	0d5	reg_adp_vlti_bst_core1 : : coring to boost filter [-1 0 2 0 -1];, default = 5 . unsigned , default = 5
7: 0	R/W	0d3	reg_adp_vlti_bst_core2 : : coring to boost filter usf, default = 3 . unsigned , default = 3

**Table 9-1788 SRSHARP0\_VLTI\_CON\_2\_GAIN\_0 0x503d**

Bit(s)	R/W	Default	Description
31:24	R/W	0d25	reg_adp_vlti_con_2_gain0 : 25;,, default = 25 . unsigned , default = 25
23:16	R/W	0d69	reg_adp_vlti_con_2_gain1 : 60;,, default = 69 . unsigned , default = 60
15: 8	R/W	0d90	reg_adp_vlti_con_2_gain2 : 0;,, default = 90 . unsigned , default = 90
7: 0	R/W	0d96	reg_adp_vlti_con_2_gain3 : 96;,, default = 96 . unsigned , default = 96

**Table 9-1789 SRSHARP0\_VLTI\_CON\_2\_GAIN\_1 0x503e**

Bit(s)	R/W	Default	Description
31:24	R/W	0d95	reg_adp_vlti_con_2_gain4 : 5;,, default = 95 . unsigned , default = 95
23:16	R/W	0d80	reg_adp_vlti_con_2_gain5 : 80;,, default = 80 . unsigned , default = 80
15: 8	R/W	0d20	reg_adp_vlti_con_2_gain6 : 20;,, default = 20 . unsigned , default = 20
7: 0	R/W	0d0	reg_adp_vlti_os_margin : : margin for vlti overshoot, default = 0 . unsigned , default = 0

**Table 9-1790 SRSHARP0\_VCTI\_FLT\_CON\_CLP 0x503f**

Bit(s)	R/W	Default	Description
14	R/W	0d1	reg_adp_vcti_en : : enable bit of vlti, default = 1 . unsigned , default = 1
13:12	R/W	0d3	reg_adp_vcti_hxn_ft : : 0: no dn; 1: [1 2 1]/4; 2: [1 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
11:10	R/W	0d3	reg_adp_vcti_dxn_ft : : 0: no dn; 1: [1 2 1]/4; 2: [1 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
9: 8	R/W	0d3	reg_adp_vcti_han_ft : : 0: no dn; 1: [1 2 1]/4; 2: [1 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
7: 6	R/W	0d3	reg_adp_vcti_dan_ft : : 0: no dn; 1: [1 2 1]/4; 2: [1 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
5: 4	R/W	0d2	reg_adp_vcti_dx_mode : : 0:[-1 1] 1:[-1 0 -1]; 2/3: [-1 0 0 0 -1], default = 2 . unsigned , default = 2
2	R/W	0d1	reg_adp_vcti_con_lpf : : lpf mode of the con: 0: [1 2 1]/4; 1:[1 2 2 1]/8, default = 1 . unsigned , default = 1
0	R/W	0d1	reg_adp_vcti_hard_clp_win : : window size; 0: 1x3 window; 1: 1x5 window, default = 1 . unsigned , default = 1

**Table 9-1791 SRSHARP0\_VCTI\_BST\_GAIN 0x5040**

Bit(s)	R/W	Default	Description
23:16	R/W	0d0	reg_adp_vcti_bst_gain0 :: gain to boost filter [-1 2 -1];, default = 0 . unsigned , default = 0
15: 8	R/W	0d0	reg_adp_vcti_bst_gain1 :: gain to boost filter [-1 0 2 0 -1];, default = 0 . unsigned , default = 0
7: 0	R/W	0d0	reg_adp_vcti_bst_gain2 :: gain to boost filter usf, default = 0 . unsigned , default = 0

**Table 9-1792 SRSHARP0\_VCTI\_BST\_CORE 0x5041**

Bit(s)	R/W	Default	Description
23:16	R/W	0d0	reg_adp_vcti_bst_core0 :: coring to boost filter [-1 2 -1];, default = 0 . unsigned , default = 0
15: 8	R/W	0d0	reg_adp_vcti_bst_core1 :: coring to boost filter [-1 0 2 0 -1];, default = 0 . unsigned , default = 0
7: 0	R/W	0d0	reg_adp_vcti_bst_core2 :: coring to boost filter usf, default = 0 . unsigned , default = 0

**Table 9-1793 SRSHARP0\_VCTI\_CON\_2\_GAIN\_0 0x5042**

Bit(s)	R/W	Default	Description
31:24	R/W	0d25	reg_adp_vcti_con_2_gain0 : 25;,, default = 25 . unsigned , default = 25
23:16	R/W	0d60	reg_adp_vcti_con_2_gain1 : 60;,, default = 60 . unsigned , default = 60
15: 8	R/W	0d90	reg_adp_vcti_con_2_gain2 : 0;,, default = 90 . unsigned , default = 90
7: 0	R/W	0d96	reg_adp_vcti_con_2_gain3 : 96;,, default = 96 . unsigned , default = 96

**Table 9-1794 SRSHARP0\_VCTI\_CON\_2\_GAIN\_1 0x5043**

Bit(s)	R/W	Default	Description
31:24	R/W	0d95	reg_adp_vcti_con_2_gain4 : 5;,, default = 95 . unsigned , default = 95
23:16	R/W	0d80	reg_adp_vcti_con_2_gain5 : 80;,, default = 80 . unsigned , default = 80
15: 8	R/W	0d20	reg_adp_vcti_con_2_gain6 : 20;,, default = 20 . unsigned , default = 20
7: 0	R/W	0d0	reg_adp_vcti_os_margin :: margin for vcti overshoot, default = 0 . unsigned , default = 0

**Table 9-1795 SRSHARP0\_SHARP\_3DLIMIT 0x5044**

Bit(s)	R/W	Default	Description
28:16	R/W	0d0	reg_3d_mid_width : ,width of left part of 3d input, dft = half size of input width default = 0 . unsigned , default = 960
12: 0	R/W	0d0	reg_3d_mid_height : ,height of left part of 3d input, dft = half size of input height default = 0 . unsigned , default = 540

**Table 9-1796 SRSHARP0\_DNLP\_EN 0x5045**

Bit(s)	R/W	Default	Description
15:8	R/W	0d0	reg_dnlp_hblank : . unsigned , default = 8
0	R/W	0d1	reg_dnlp_en : . unsigned , default = 1

**Table 9-1797 SRSHARP0\_DEMO\_CTRL 0x5056**

Bit(s)	R/W	Default	Description
18:17	R/W	0d2	demo_disp_position : . unsigned , default = 2
16	R/W	0d0	demo_hsvsharp_enable : . unsigned , default = 0
12: 0	R/W	0d360	demo_left_top_screen_width : . unsigned , default = 360

**Table 9-1798 SRSHARP0\_SHARP\_SR2\_CTRL 0x5057**

Bit(s)	R/W	Default	Description
24	R/W	0	reg_sr2_bic_pknr_bypass : bypass peaking/TI/Cubic
23:22	R/W		reserved
21:16	R/W	24	sr2_pk_la_err_dis_rate, low angle and high angle error should not be no less than nearby_error* rate/64
15: 8	R/W	16	sr2_pk_sad_diag_gain, gain to sad[2] and sad[6], 16 normalized to 1
7	R/W	0	sr2_vert_outphs, vertical output pixel phase, 0: 0 phase; 1: 1/2 phase
6	R/W	0	sr2_horz_outphs, horizontal output pixel phase, 0: 0 phase; 1: 1/2 phase
5	R/W	0	sr2_vert_ratio , vertical scale ratio, 0-> 1:1; 1-> 1:2
4	R/W	0	sr2_horz_ratio , horizontal scale ratio, 0-> 1:1; 1-> 1:2
3	R/W	1	sr2_bic_norm , normalization of bicubical: 0: 128; 1: 64
2	R/W	0	sr2_enable , 1 to enable super scaler
1	R/W	0	sr2_sharp_prc_lr_hbic,
0	R/W	0	sr2_sharp_prc_lr, 1: LTI/CTI/NR/Peaking processing using LR grid. 0: on HR grid; 1:on LR grid, horizontally no upscale, but using simple bic.

**Table 9-1799 SRSHARP0\_SHARP\_SR2\_YBIC\_HCOEF0 0x5058**

Bit(s)	R/W	Default	Description
31:24	R/W	0	sr2_y_bic_hcoeff03, signed
23:16	R/W	0	sr2_y_bic_hcoeff02, signed
15: 8	R/W	64	sr2_y_bic_hcoeff01, signed
7: 0	R/W	0	sr2_y_bic_hcoeff00, signed

**Table 9-1800 SRSHARP0\_SHARP\_SR2\_YBIC\_HCOEF1 0x5059**

Bit(s)	R/W	Default	Description
31:24	R/W	-4	sr2_y_bic_hcoeff13 , signed
23:16	R/W	36	sr2_y_bic_hcoeff12 , signed
15: 8	R/W	36	sr2_y_bic_hcoeff11 , signed
7: 0	R/W	-4	sr2_y_bic_hcoeff10 , signed

**Table 9-1801 SRSHARP0\_SHARP\_SR2\_CBIC\_HCOEF0 0x505a**

Bit(s)	R/W	Default	Description
31:24	R/W	0	sr2_c_bic_hcoeff03 , signed
23:16	R/W	21	sr2_c_bic_hcoeff02 , signed
15: 8	R/W	22	sr2_c_bic_hcoeff01 , signed
7: 0	R/W	21	sr2_c_bic_hcoeff00 , signed

**Table 9-1802 SRSHARP0\_SHARP\_SR2\_CBIC\_HCOEF1 0x505b**

Bit(s)	R/W	Default	Description
31:24	R/W	-4	sr2_c_bic_hcoeff13 , signed
23:16	R/W	36	sr2_c_bic_hcoeff12 , signed
15: 8	R/W	36	sr2_c_bic_hcoeff11 , signed
7: 0	R/W	-4	sr2_c_bic_hcoeff10 , signed

**Table 9-1803 SRSHARP0\_SHARP\_SR2\_YBIC\_VCOEF0 0x505c**

Bit(s)	R/W	Default	Description
31:24	R/W	0	sr2_y_bic_vcoeff03 , signed
23:16	R/W	0	sr2_y_bic_vcoeff02 , signed
15: 8	R/W	64	sr2_y_bic_vcoeff01 , signed
7: 0	R/W	0	sr2_y_bic_vcoeff00 , signed

**Table 9-1804 SRSHARP0\_SHARP\_SR2\_YBIC\_VCOEF1 0x505d**

Bit(s)	R/W	Default	Description
31:24	R/W	-4	sr2_y_bic_vcoeff13 , signed
23:16	R/W	36	sr2_y_bic_vcoeff12 , signed
15: 8	R/W	36	sr2_y_bic_vcoeff11 , signed
7: 0	R/W	-4	sr2_y_bic_vcoeff10 , signed

**Table 9-1805 SRSHARP0\_SHARP\_SR2\_CBIC\_VCOEF0 0x505e**

Bit(s)	R/W	Default	Description
31:24	R/W	0	sr2_c_bic_vcoeff03 , signed
23:16	R/W	21	sr2_c_bic_vcoeff02 , signed
15: 8	R/W	22	sr2_c_bic_vcoeff01 , signed
7: 0	R/W	21	sr2_c_bic_vcoeff00 , signed

**Table 9-1806 SRSHARP0\_SHARP\_SR2\_CBIC\_VCOEF1 0x505f**

Bit(s)	R/W	Default	Description
31:24	R/W	-4	sr2_c_bic_vcoeff13 , signed
23:16	R/W	36	sr2_c_bic_vcoeff12 , signed
15: 8	R/W	36	sr2_c_bic_vcoeff11 , signed
7: 0	R/W	-4	sr2_c_bic_vcoeff10 , signed

**Table 9-1807 SRSHARP0\_SHARP\_SR2\_MISC 0x5060**

Bit(s)	R/W	Default	Description
31:2	R/W		reserved
1	R/W	0	sr2_cmpmux_bef , 0 : no swap for YUV/RGB; 1: swap for YUV/RGB, YUV/RGB->UVY/GBR
0	R/W	0	sr2_cmpmux_aft , 0 : no swap for YUV/RGB; 1: swap for YUV/RGB, UVY/GBR->YUV/RGB

**Table 9-1808 SRSHARP0\_SR3\_SAD\_CTRL 0x5061**

Bit(s)	R/W	Default	Description
31:30	R/W		reserved
29:24	R/W	0d6	reg_sr3_pk_sad_core_rate : rate of coring.
23:22	R/W		reserved
21:16	R/W	0d6	reg_sr3_lti_sad_core_rate : rate of coring.
15:14	R/W		reserved
13:8	R/W	0d6	reg_sr3_cti_sad_core_rate : rate of coring.
7	R/W	0d1	reg_sr3_lti_hsad_mode: mode for hsad of lti calculation, 0:block based; 1: other sharp
6	R/W	0d1	reg_sr3_cti_hsad_mode: mode for hsad of cti calculation, 0:block based; 1: other sharp
5	R/W	0d1	reg_sr3_lti_dsad_mode: mode for dsad of lti calculation, 0:block based; 1: other sharp
4	R/W	0d1	reg_sr3_cti_dsad_mode: mode for dsad of cti calculation, 0:block based; 1: other sharp
3	R/W	0d1	reg_sr3_lti_vsad_mode: mode for vsad of lti calculation, 0:block based; 1: other sharp

Bit(s)	R/W	Default	Description
2	R/W	0d1	reg_sr3_cti_vsad_mode: mode for vsad of cti calculation, 0:block based; 1: other sharp
1	R/W	0d1	reg_sr3_lti_hsad_hlpf: hlpf for hsad of lti calculation, 0:no hlpf; 1: with [1 2 1] hlpf.
0	R/W	0d1	reg_sr3_cti_hsad_hlpf: hlpf for hsad of cti calculation, 0:no hlpf; 1: with [1 2 1] hlpf.

Table 9-1809 SRSHARP0\_SR3\_PK\_CTRL0 0x5062

Bit(s)	R/W	Default	Description
31:12	R/W		reserved
11	R/W	0d1	reg_sr3_pk_sad_mode: mode for sad of peaking calculation, 0: block based; 1: other sharp.
10	R/W	0d1	reg_sr3_pk_hsad_hlpf: hlpf for hsad for peaking calculation,0:no hlpf; 1: with [1 2 2 2 1] hlpf.
9	R/W	0d1	reg_sr3_pk_vsad_hlpf: hlpf for vsad for peaking calculation,0:no hlpf; 1: with [1 2 2 2 1] hlpf.
8	R/W	0d1	reg_sr3_pk_dsad_hlpf: hlpf for dsad for peaking calculation,0:no hlpf; 1: with [1 2 2 2 1] hlpf.
7:6	R/W	0d3	reg_sr3_pk_hpdrtr_mode: mode for HPdrtr filter
5:4	R/W	0d3	reg_sr3_pk_bpdrtr_mode: mode for BPdrtr filter
3:2	R/W	0d3	reg_pk_drtrbld_range: range of the min2 and min direction distance
1	R/W		reserved
0	R/W	0d0	reg_sr3_pk_ti_blend_mode: blend mode of the TI and PK result

Table 9-1810 SRSHARP0\_SR3\_PK\_CTRL1 0x5063

Bit(s)	R/W	Default	Description
31	R/W		reserved
30:28	R/W	0d1	reg_sr3_pk_Hp_hvcon_replace8_maxsad: replace HP hvcon by maxsad
26:24	R/W	0d1	reg_sr3_pk_bp_hvcon_replace8_maxsad: replace BP hvcon by maxsad
23:16	R/W	0d32	reg_sr3_pk_hp_hvcon_replace8lv_gain: gain to local variant before calculating the hv gain for peaking.
15:8	R/W	0d32	reg_sr3_pk_bp_hvcon_replace8lv_gain: gain to local variant before calculating the hv gain for peaking.
7	R/W	0d1	reg_sr3_sad_intlev_mode: interleave detect xerr mode: 0 max; 1: sum
6	R/W	0d1	reg_sr3_sad_intlev_mode1: mode 1 of using diagonal protection: 1: with diagonal protection
5:0	R/W	0d12	reg_sr3_sad_intlev_gain: interleave detection for sad gain applied, normalized to 8 as 1

**Table 9-1811 SRSHARP0\_DEJ\_CTRL 0x5064**

Bit(s)	R/W	Default	Description
15:8	R/W	8	reg_sr3_dejaggy_hblank
7:4	R/W		reserved
3:2	R/W	0d3	reg_sr3_dejaggy_sameside_prct: enable of sr3 dejaggy same side curve protect from filter, [0] for proc path; [1] for ctrl path.
1	R/W	0d1	reg_sr3_dejaggy_sameside_mode: mode of sameside flag decision
0	R/W	0d1	reg_sr3_dejaggy_enable: enable of sr3 dejaggy

**Table 9-1812 SRSHARP0\_DEJ\_ALPHA 0x5065**

Bit(s)	R/W	Default	Description
31:28	R/W	0d0	reg_sr3_dejaggy_ctrchrom_alpha_1 : alpha for LR video LPF
27:24	R/W	0d15	reg_sr3_dejaggy_ctrchrom_alpha_0 : alpha for LR video LPF
23:20	R/W	0d0	reg_sr3_dejaggy_ctrluma_alpha_1 : alpha for LR video LPF
19:16	R/W	0d15	reg_sr3_dejaggy_ctrluma_alpha_0 : alpha for LR video LPF
15:12	R/W	0d4	reg_sr3_dejaggy_procchrom_alpha_1: alpha for LR video LPF
11:8	R/W	0d6	reg_sr3_dejaggy_procchrom_alpha_0: alpha for LR video LPF
7:4	R/W	0d4	reg_sr3_dejaggy_procluma_alpha_1: alpha for LR video LPF
3:0	R/W	0d6	reg_sr3_dejaggy_procluma_alpha_0: alpha for LR video LPF

**Table 9-1813 SRSHARP0\_SR3\_DRTLPF\_EN 0x5066**

Bit(s)	R/W	Default	Description
31:15	R/W		reserved
14:8	R/W	0d0	reg_pk_debug_edge
6:4	R/W	0d0	reg_sr3_drtlpf_theta_en
2:0	R/W	0d7	reg_sr3_drtlpf_enable: directional lpf on Y/U/V channels

**Table 9-1814 SRSHARP0\_SR3\_DRTLPF\_ALPHA\_0 0x5067**

Bit(s)	R/W	Default	Description
31:28	R/W		reserved
27:24	R/W	0d9	reg_sr3_drtlpf_alpha_3
23:20	R/W		reserved
19:16	R/W	0d10	reg_sr3_drtlpf_alpha_2
15:12	R/W		reserved
11:8	R/W	0d11	reg_sr3_drtlpf_alpha_1
7:4	R/W		reserved
3:0	R/W	0d12	reg_sr3_drtlpf_alpha_0: directional lpf alpha coef for min_sad/max_sad compare

**Table 9-1815 SRSHARP0\_SR3\_DRTLPF\_ALPHA\_1 0x5068**

Bit(s)	R/W	Default	Description
31:28	R/W		reserved
27:24	R/W	0d1	reg_sr3_drtlpf_alpha_7
23:20	R/W		reserved
19:16	R/W	0d4	reg_sr3_drtlpf_alpha_6
15:12	R/W		reserved
11:8	R/W	0d7	reg_sr3_drtlpf_alpha_5
7:4	R/W		reserved
3:0	R/W	0d8	reg_sr3_drtlpf_alpha_4: directional lpf alpha coef for min_sad/max_sad compare

**Table 9-1816 SRSHARP0\_SR3\_DRTLPF\_ALPHA\_2 0x5069**

Bit(s)	R/W	Default	Description
31:28	R/W		reserved
27:24	R/W	0d0	reg_sr3_drtlpf_alpha_11
23:20	R/W		reserved
19:16	R/W	0d0	reg_sr3_drtlpf_alpha_10
15:12	R/W		reserved
11:8	R/W	0d0	reg_sr3_drtlpf_alpha_9
7:4	R/W		reserved
3:0	R/W	0d0	reg_sr3_drtlpf_alpha_8: directional lpf alpha coef for min_sad/max_sad compare

**Table 9-1817 SRSHARP0\_SR3\_DRTLPF\_ALPHA\_OFST 0x506a**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reg_sr3_drtlpf_alpha_ofst7
27:24	R/W	0	reg_sr3_drtlpf_alpha_ofst6
23:20	R/W	0	reg_sr3_drtlpf_alpha_ofst5
19:16	R/W	-2	reg_sr3_drtlpf_alpha_ofst4
15:12	R/W	0	reg_sr3_drtlpf_alpha_ofst3
11:8	R/W	0	reg_sr3_drtlpf_alpha_ofst2
7:4	R/W	0	reg_sr3_drtlpf_alpha_ofst1
3:0	R/W	-2	reg_sr3_drtlpf_alpha_ofst0: directional lpf alpha coef offset of each direction.



**Table 9-1818 SRSHARP0\_SR3\_DERING\_CTRL 0x506b**

Bit(s)	R/W	Default	Description
31	R/W		reserved
30:28	R/W	1	reg_sr3_dering_enable: dering enable
27	R/W		reserved
26:24	R/W	3	reg_sr3_dering_varlpf_mode: local variant LPF mode. 0: no filter; 1: erosion 3x3; 2: 3x3 lpf; 3: 3x3 erosion + lpf
23:20	R/W	9	reg_sr3_dering_maxrange: range of dering in LR resolution.
19:18	R/W		reserved
17:16	R/W	2	reg_sr3_dering_lcvar_blend_mode: mode for lcvar calculation. 0:HV blend; 1:diag blend; 2:HV blend + V; 3: HV blend+Diag blend
15:8	R/W	40	reg_sr3_dering_lcvar_gain: gain to local variant and normalized to 32 as 1
7:0	R/W	28	reg_sr3_dering_lcvar_nearby_maxsad_th: threshold to use near side maxsad if that side is larger than this threshold, otherwise use the max one.

**Table 9-1819 SRSHARP0\_SR3\_DERING\_LUMA2PKGAIN\_0TO3 0x506c**

Bit(s)	R/W	Default	Description
31:24	R/W	255	reg_sr3_dering_luma2pkgain3: level limit(for th0<bpcon<th1) of curve for dering pkgain base on LPF luma level
23:16	R/W	255	reg_sr3_dering_luma2pkgain2: level limit(for bpcon<th0) of curve for dering pkgain base on LPF luma level
15:8	R/W	200	reg_sr3_dering_luma2pkgain1: threshold 1 of curve for dering pkgain based on LPF luma level.
7:0	R/W	30	reg_sr3_dering_luma2pkgain0: threshold 0 of curve for dering pkgain based on LPF luma level.

**Table 9-1820 SRSHARP0\_SR3\_DERING\_LUMA2PKGAIN\_4TO6 0x506d**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:16	R/W	24	reg_sr3_dering_luma2pkgain6: rate1 (for bpcon>th1) of curve for dering pkOS based on LPF luma level.
15:8	R/W	50	reg_sr3_dering_luma2pkgain5: rate0 (for bpcon<th0) of curve for dering pkOS based on LPF luma level.
7:0	R/W	255	reg_sr3_dering_luma2pkgain4: level limit(for bpcon>th1) of curve for dering pkgain base on LPF luma level

**Table 9-1821 SRSHARP0\_SR3\_DERING\_LUMA2PKOS\_0TO3 0x506e**

Bit(s)	R/W	Default	Description
31:24	R/W	255	reg_sr3_dering_luma2pkos3: level limit(for th0<bpcon<th1) of curve for dering pkOS base on LPF luma level
23:16	R/W	255	reg_sr3_dering_luma2pkos2: level limit(for bpcon<th0) of curve for dering pkOS base on LPF luma level

Bit(s)	R/W	Default	Description
15:8	R/W	200	reg_sr3_deriving_luma2pkos1: threshold 1 of curve for deriving pkOS based on LPF luma level.
7:0	R/W	30	reg_sr3_deriving_luma2pkos0: threshold 0 of curve for deriving pkOS based on LPF luma level.

**Table 9-1822 SRSHARP0\_SR3\_DERIVING\_LUMA2PKOS\_4TO6 0x506f**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:16	R/W	24	reg_sr3_deriving_luma2pkos6: rate1 (for bpcon>th1) of curve for deriving pkOS based on LPF luma level.
15:8	R/W	50	reg_sr3_deriving_luma2pkos5: rate0 (for bpcon<th0) of curve for deriving pkOS based on LPF luma level.
7:0	R/W	255	reg_sr3_deriving_luma2pkos4: level limit(for bpcon>th1) of curve for deriving pkOS base on LPF luma level

**Table 9-1823 SRSHARP0\_SR3\_DERIVING\_GAINVS\_MADSAD 0x5070**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reg_sr3_deriving_gainvs_maxsad7
27:24	R/W	0	reg_sr3_deriving_gainvs_maxsad6
23:20	R/W	0	reg_sr3_deriving_gainvs_maxsad5
19:16	R/W	0	reg_sr3_deriving_gainvs_maxsad4
15:12	R/W	0	reg_sr3_deriving_gainvs_maxsad3
11:8	R/W	0	reg_sr3_deriving_gainvs_maxsad2
7:4	R/W	4	reg_sr3_deriving_gainvs_maxsad1
3:0	R/W	8	reg_sr3_deriving_gainvs_maxsad0: pkgain vs maxsad value, 8 node interpolations.

**Table 9-1824 SRSHARP0\_SR3\_DERIVING\_GAINVS\_VR2MAX 0x5071**

Bit(s)	R/W	Default	Description
31:28	R/W	15	reg_sr3_deriving_gainvs_vr2max7
27:24	R/W	15	reg_sr3_deriving_gainvs_vr2max6
23:20	R/W	15	reg_sr3_deriving_gainvs_vr2max5
19:16	R/W	15	reg_sr3_deriving_gainvs_vr2max4
15:12	R/W	14	reg_sr3_deriving_gainvs_vr2max3
11:8	R/W	12	reg_sr3_deriving_gainvs_vr2max2
7:4	R/W	2	reg_sr3_deriving_gainvs_vr2max1
3:0	R/W	0	reg_sr3_deriving_gainvs_vr2max0: pkgain vs ratio

**Table 9-1825 SRSHARP0\_SR3\_DERING\_PARAM0 0x5072**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:16	R/W	10	reg_sr3_dering_lcvar_floor
15:8	R/W	32	reg_sr3_dering_vr2max_gain: gain to max before feeding to LUT
7:6	R/W		reserved
5:0	R/W	16	reg_sr3_dering_vr2max_limt: limit of maxsad

**Table 9-1826 SRSHARP0\_SR3\_DRTLPF\_THETA 0x5073**

Bit(s)	R/W	Default	Description
31:0	R/W	0xfe-c96420	reg_sr3_drtlpf_theta: u4x8 directional lpf beta coef for min_sad/min2_sad compared to x=0:7 correspond to [1:8]/16; 0 means no drtLPF, 15: 100% alpha dependant drtLPF.

**Table 9-1827 SRSHARP0\_SATPRT\_CTRL 0x5074**

Bit(s)	R/W	Default	Description
31:28	R/W		reserved
27:16	R/W	5	reg_satprt_sat_core: 4x will be coring to cor(irgb_max-irgb_min) to calculate the oy_delt, the smaller the more protection to color, the larger only the rich color will be protected.
15:8	R/W	64	reg_satprt_sat_rate: rate to cor(irgb_max-irgb_min) to calculate the oy_delt, the larger the more protection to color; norm 16 as 1
7:4	R/W		reserved
3:2	R/W	1	reg_satprt_csc_mode: CSC mode of current yuv input: 0:601; 1:709; 2:BT2020 NCL; 3 reserved
1	R/W	1	reg_satprt_is_lmt: flag telling the YUV is limited range data or full rang data; 1 is limited data
0	R/W	0	reg_satprt_enable: 1 to enable of saturation protection for dnlp adjustments

**Table 9-1828 SRSHARP0\_SATPRT\_DIVM 0x5075**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:0	R/W	{128,12-8,128}	reg_satprt_div_m: u8x3, 1/m, normalized to 128 as 1.

**Table 9-1829 SRSHARP0\_DB\_FLT\_CTRL 0x5077**

Bit(s)	R/W	Default	Description
26	R/W	0	reg_nrdeband_reset1 : // unsigned , default = 0 0 : no reset seed 1: reload chroma seed
25	R/W	0	reg_nrdeband_reset0 : // unsigned , default = 0 0 : no reset seed 1: reload luma seed

Bit(s)	R/W	Default	Description
24	R/W	0	reg_nrdeband_rgb : // unsigned , default = 0 0 : yuv 1: RGB
23	R/W	1	reg_nrdeband_en11 : // unsigned , default = 1 debanding registers of side lines, [0] for luma, same for below
22	R/W	1	reg_nrdeband_en10 : // unsigned , default = 1 debanding registers of side lines, [1] for chroma, same for below
21	R/W	1	reg_nrdeband_siderand : // unsigned , default = 1 options to use side two lines use the rand, instead of use for the YUV three component of middle line, 0: seed [3]/bandrand[3] for middle line yuv; 1: seed[3]/bandrand[3] for nearby three lines Y;
20	R/W	0	reg_nrdeband_randmode : // unsigned , default = 0 mode of rand noise adding, 0: same noise strength for all difs; else: strenght of noise will not exceed the difs, MIN((pPKReg->reg_nrdeband_bandrand[m]), noise[m])
19:17	R/W	6	reg_nrdeband_bandrand2 : // unsigned , default = 6
15:13	R/W	6	reg_nrdeband_bandrand1 : // unsigned , default = 6
11: 9	R/W	6	reg_nrdeband_bandrand0 : // unsigned , default = 6
7	R/W	1	reg_nrdeband_hpxor1 : // unsigned , default = 1 debanding random hp portion xor, [0] for luma
6	R/W	1	reg_nrdeband_hpxor0 : // unsigned , default = 1 debanding random hp portion xor, [1] for chroma
5	R/W	1	reg_nrdeband_en1 : // unsigned , default = 1 debanding registers, for luma
4	R/W	1	reg_nrdeband_en0 : // unsigned , default = 1 debanding registers, for chroma
3: 2	R/W	2	reg_nrdeband_lpf_mode1 : // unsigned , default = 2 lpf mode, 0: 3x3, 1:3x5; 2: 5x5; 3:5x7
1: 0	R/W	2	reg_nrdeband_lpf_mode0 : // unsigned , default = 2 lpf mode, 0: 3x3, 1:3x5; 2: 5x5; 3:5x7

Table 9-1830 SRSHARP0\_DB\_FLT\_RANLUT 0x5079

Bit(s)	R/W	Default	Description
23:21	R/W	1	reg_nrdeband_randslut7 : // unsigned , default = 1 lut0
20:18	R/W	1	reg_nrdeband_randslut6 : // unsigned , default = 1 lut0
17:15	R/W	1	reg_nrdeband_randslut5 : // unsigned , default = 1 lut0
14:12	R/W	1	reg_nrdeband_randslut4 : // unsigned , default = 1 lut0
11: 9	R/W	1	reg_nrdeband_randslut3 : // unsigned , default = 1 lut0
8: 6	R/W	1	reg_nrdeband_randslut2 : // unsigned , default = 1 lut0
5: 3	R/W	1	reg_nrdeband_randslut1 : // unsigned , default = 1 lut0
2: 0	R/W	1	reg_nrdeband_randslut0 : // unsigned , default = 1 lut0

**Table 9-1831 SRSHARP0\_DB\_FLT\_PXI\_THRD 0x507a**

Bit(s)	R/W	Default	Description
25:16	R/W	0	reg_nrdeband_yc_th1 : // unsigned , default = 0 to luma/ u/v  for using the denoise
9: 0	R/W	0	reg_nrdeband_yc_th0 : // unsigned , default = 0 to luma/ u/v  for using the denoise

**Table 9-1832 SRSHARP0\_DB\_FLT\_SEED\_Y 0x507b**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8240	reg_nrdeband_seed0 : // unsigned , default = 1621438240 noise adding seed for Y. seed[0]= 0x60a52f20; as default

**Table 9-1833 SRSHARP0\_DB\_FLT\_SEED\_U 0x507c**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8247	reg_nrdeband_seed1 : // unsigned , default = 1621438247 noise adding seed for U. seed[0]= 0x60a52f27; as default

**Table 9-1834 SRSHARP0\_DB\_FLT\_SEED\_V 0x507d**

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8242	reg_nrdeband_seed2 : // unsigned , default = 1621438242 noise adding seed for V. seed[0]= 0x60a52f22; as default

**Table 9-1835 SRSHARP0\_PKGAIN\_VSLUMA\_LUT\_L 0x507e**

Bit(s)	R/W	Default	Description
31:28	R/W	5	reg_pkgain_vsluma_lut7
27:24	R/W	6	reg_pkgain_vsluma_lut6
23:20	R/W	6	reg_pkgain_vsluma_lut5
19:16	R/W	6	reg_pkgain_vsluma_lut4
15:12	R/W	7	reg_pkgain_vsluma_lut3
11:8	R/W	10	reg_pkgain_vsluma_lut2
7:4	R/W	12	reg_pkgain_vsluma_lut1
3:0	R/W	8	reg_pkgain_vsluma_lut0

**Table 9-1836 SRSHARP0\_PKGAIN\_VSLUMA\_LUT\_H 0x507f**

Bit(s)	R/W	Default	Description
31:4	R/W		reserved
3:0	R/W	4	reg_pkgain_vsluma_lut8

**Table 9-1837 SRSHARP0\_PKOSHT\_VSLUMA\_LUT\_L 0x5080**

Bit(s)	R/W	Default	Description
31:28	R/W	5	reg_pkosht_vsluma_lut7
27:24	R/W	6	reg_pkosht_vsluma_lut6
23:20	R/W	6	reg_pkosht_vsluma_lut5
19:16	R/W	6	reg_pkosht_vsluma_lut4
15:12	R/W	7	reg_pkosht_vsluma_lut3
11:8	R/W	10	reg_pkosht_vsluma_lut2
7:4	R/W	12	reg_pkosht_vsluma_lut1
3:0	R/W	8	reg_pkosht_vsluma_lut0

**Table 9-1838 SRSHARP0\_PKOSHT\_VSLUMA\_LUT\_H 0x5081**

Bit(s)	R/W	Default	Description
31:4	R/W		reserved
3:0	R/W	4	reg_pkosht_vsluma_lut8

**Table 9-1839 SRSHARP0\_SATPRT\_LMT\_RGB1 0x5082**

Bit(s)	R/W	Default	Description
27:16	R/W	0d0	reg_satprt_lmt_g:
11: 0	R/W	0d0	reg_satprt_lmt_r: limit of RGB channel, for limited range RGB, 12bits

**Table 9-1840 SRSHARP0\_SATPRT\_LMT\_RGB2 0x5083**

Bit(s)	R/W	Default	Description
31:16	R/W	0d0	reserved
11: 0	R/W	0d0	reg_satprt_lmt_b: limit of RGB channel, for limited range RGB

**Table 9-1841 SRSHARP0\_SHARP\_GATE\_CLK\_CTRL\_0 0x5084**

Bit(s)	R/W	Default	Description
31:0	R/W	0d0	Gate clock control [01:00]: sharp input control unit [03:02]: deband unit [05:04]: dejaggy unit [07:06]: dnlp unit [09:08]: demo control unit [11:10]: horiz interp unit

**Table 9-1842 SRSHARP0\_SHARP\_GATE\_CLK\_CTRL\_1 0x5085**

Bit(s)	R/W	Default	Description
31:0	R/W	0d0	Gate clock control [01:00]: sr_top "pipe_ctrl" [03:02]: drt [05:04]: ssd [07:06]: cubic [09:08]: edi [11:10]: pkgainsad [13:12]: bicomux [15:14]: bicin_fifo [17:16]: lpf4pkgain_fifo [19:18]: min2hvgain_fifo [21:20]: sad4pkgain_fifo [23:22]: dirminmax4xtl_fifo [25:24]: drtsad8_fifo [27:26]: ssdmax_fifo [29:28]: pkminmax_fifo [31:30]: cirdrtgain_fifo

**Table 9-1843 SRSHARP0\_SHARP\_GATE\_CLK\_CTRL\_2 0x5086**

Bit(s)	R/W	Default	Description
31:0	R/W	0d0	Gate clock control [01:00]: bufdiff_fifo [03:02]: osvar_fifo [05:04]: pkhvgain unit [07:06]: pkgain unit [09:08]: TI unit [11:10]: pk unit [13:12]: locvar unit [15:14]: hvconc unit

**Table 9-1844 SRSHARP0\_SHARP\_GATE\_CLK\_CTRL\_3 0x5087**

Bit(s)	R/W	Default	Description
31:0	R/W	0d0	Gate clock control [01:00]: TI, htl Y [03:02]: TI, vtl Y [05:04]: TI, htl U [07:06]: TI, vtl U [09:08]: TI, htl V [11:10]: TI, vtl V [13:12]: TI, lumaminmax_fifo [15:14]: TI, chrmmminmax_fifo

**Table 9-1845 SRSHARP0\_SHARP\_DPS\_CTRL 0x5088**

Bit(s)	R/W	Default	Description
31:0	R/W	0d0	Power saving control : hvcon : nrssd : os filter : cubic 5 lines to 9 lines : dering : locvar : drtlpf : hlti : hcti : vlti : vcti : lti blend : cti blend : htishort (no used) : nr Y filter : nr C filter : nr belnd : pkti blend : os ctrl : pk HP : pk BP [26:24] dejaggy power saving control [30:28] reserved

**Table 9-1846 SRSHARP0\_DNLP\_00 0x5090**

Bit(s)	R/W	Default	Description
31:0	R/W	0x0010-0008	reg_dnlp_ygrid0 :: dnlp00 . unsigned , default = 32'h00100008

**Table 9-1847 SRSHARP0\_DNLP\_01 0x5091**

Bit(s)	R/W	Default	Description
31:0	R/W	0x0020-0018	reg_dnlp_ygrid1 :: dnlp01 . unsigned , default = 32'h00200018

**Table 9-1848 SRSHARP0\_DNLP\_02 0x5092**

Bit(s)	R/W	Default	Description
31:0	R/W	0x0030-0028	reg_dnlp_ygrid2 :: dnlp02 . unsigned , default = 32'h00300028

**Table 9-1849 SRSHARP0\_DNLP\_03 0x5093**

Bit(s)	R/W	Default	Description
31:0	R/W	0x0040-0038	reg_dnlp_ygrid3 :: dnlp03 . unsigned , default = 32'h00400038



**Table 9-1850 SRSHARP0\_DNLP\_04 0x5094**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0050-0048	reg_dnlp_ygrid4 :: dnlp04 . unsigned , default = 32'h00500048

**Table 9-1851 SRSHARP0\_DNLP\_05 0x5095**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0068-005c	reg_dnlp_ygrid5 :: dnlp05 . unsigned , default = 32'h0068005c

**Table 9-1852 SRSHARP0\_DNLP\_06 0x5096**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0080-0074	reg_dnlp_ygrid6 :: dnlp06 . unsigned , default = 32'h00800074

**Table 9-1853 SRSHARP0\_DNLP\_07 0x5097**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x00-a00090	reg_dnlp_ygrid7 :: dnlp07 . unsigned , default = 32'h00a00090

**Table 9-1854 SRSHARP0\_DNLP\_08 0x5098**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x00-c000b0	reg_dnlp_ygrid8 :: dnlp08 . unsigned , default = 32'h00c000b0

**Table 9-1855 SRSHARP0\_DNLP\_09 0x5099**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x00-e000d0	reg_dnlp_ygrid9 :: dnlp09 . unsigned , default = 32'h00e000d0

**Table 9-1856 SRSHARP0\_DNLP\_10 0x509a**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0100-00f0	reg_dnlp_ygrid10 :: dnlp10 . unsigned , default = 32'h010000f0

**Table 9-1857 SRSHARP0\_DNLP\_11 0x509b**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x012-c0114	reg_dnlp_ygrid11 :: dnlp11 . unsigned , default = 32'h012c0114

**Table 9-1858 SRSHARP0\_DNLP\_12 0x509c**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0154-0140	reg_dnlp_ygrid12 :: dnlp12 . unsigned , default = 32'h01540140

**Table 9-1859 SRSHARP0\_DNLP\_13 0x509d**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0180-016c	reg_dnlp_ygrid13 :: dnlp13 . unsigned , default = 32'h0180016c

**Table 9-1860 SRSHARP0\_DNLP\_14 0x509e**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x01-c001a0	reg_dnlp_ygrid14 :: dnlp14 . unsigned , default = 32'h01c001a0

**Table 9-1861 SRSHARP0\_DNLP\_15 0x509f**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0200-01e0	reg_dnlp_ygrid15 :: dnlp15 . unsigned , default = 32'h020001e0

**Table 9-1862 SRSHARP0\_DNLP\_16 0x50a0**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0240-0220	reg_dnlp_ygrid16 :: dnlp16 . unsigned , default = 32'h02400220

**Table 9-1863 SRSHARP0\_DNLP\_17 0x50a1**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0280-0260	reg_dnlp_ygrid17 :: dnlp17 . unsigned , default = 32'h02800260

**Table 9-1864 SRSHARP0\_DNLP\_18 0x50a2**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x02-b00298	reg_dnlp_ygrid18 :: dnlp18 . unsigned , default = 32'h02b00298

**Table 9-1865 SRSHARP0\_DNLP\_19 0x50a3**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x02-e002c8	reg_dnlp_ygrid19 :: dnlp19 . unsigned , default = 32'h02e002c8

**Table 9-1866 SRSHARP0\_DNLP\_20 0x50a4**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0300-02f0	reg_dnlp_ygrid20 :: dnlp20 . unsigned , default = 32'h030002f0

**Table 9-1867 SRSHARP0\_DNLP\_21 0x50a5**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0320-0310	reg_dnlp_ygrid21 :: dnlp21 . unsigned , default = 32'h03200310

**Table 9-1868 SRSHARP0\_DNLP\_22 0x50a6**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0338-032c	reg_dnlp_ygrid22 :: dnlp22 . unsigned , default = 32'h0338032c

**Table 9-1869 SRSHARP0\_DNLP\_23 0x50a7**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0350-0348	reg_dnlp_ygrid23 :: dnlp23 . unsigned , default = 32'h03500348

**Table 9-1870 SRSHARP0\_DNLP\_24 0x50a8**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x035c-0358	reg_dnlp_ygrid24 :: dnlp24 . unsigned , default = 32'h035c0358

**Table 9-1871 SRSHARP0\_DNLP\_25 0x50a9**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0368-0360	reg_dnlp_ygrid25 :: dnlp25 . unsigned , default = 32'h03680360

**Table 9-1872 SRSHARP0\_DNLP\_26 0x50aa**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0378-0370	reg_dnlp_ygrid26 :: dnlp26 . unsigned , default = 32'h03780370

**Table 9-1873 SRSHARP0\_DNLP\_27 0x50ab**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0388-0380	reg_dnlp_ygrid27 :: dnlp27 . unsigned , default = 32'h03880380

**Table 9-1874 SRSHARP0\_DNLP\_28 0x50ac**

Bit(s)	R/W	Default	Description
31:0	R/W	0x03-a00390	reg_dnlp_ygrid28 :: dnlp28 . unsigned , default = 32'h03a00390

**Table 9-1875 SRSHARP0\_DNLP\_29 0x50ad**

Bit(s)	R/W	Default	Description
31:0	R/W	0x03-c003b0	reg_dnlp_ygrid29 :: dnlp29 . unsigned , default = 32'h03c003b0

**Table 9-1876 SRSHARP0\_DNLP\_30 0x50ae**

Bit(s)	R/W	Default	Description
31:0	R/W	0x03-e003d0	reg_dnlp_ygrid30 :: dnlp30 . unsigned , default = 32'h03e003d0

**Table 9-1877 SRSHARP0\_DNLP\_31 0x50af**

Bit(s)	R/W	Default	Description
31:0	R/W	0x03-fc03f0	reg_dnlp_ygrid31 :: dnlp31 . unsigned , default = 32'h03fc03f0

**Table 9-1878 SRSHARP0\_SHARP\_SYNC\_CTRL 0x50b0**

Bit(s)	R/W	Default	Description
15:0	R/W	0d0	Shadow control for some setting: 0 to use VSYNC to shadow : SR all bypass : reg_sr2_bic_pknr_bypass [2]: reg_3d_mode [3]: reg_nrdeband_en11 [4]: reg_nrdeband_en10 [5]: reg_nrdeband_en1 [6]: reg_nrdeband_en0 [7]: reg_dejaggy_enable [8]: reg_sr2_sharp_prc_lr [9]: reg_sharp_pk_en [10]: reg_sharp_nr_en [11]: reg_adp_hlti_en [12]: reg_adp_hcti_en [13]: reg_adp_vlti_en [14]: reg_adp_vcti_en [15]: reg_dnlp_en

**Table 9-1879 SRSHARP0\_NR\_GAU\_YH\_COEF02 0x50b2**

Bit(s)	R/W	Default	Description
29:20	R/W	0d32	luma's horizontal adaptive coef0 . signed , default = 32
19:10	R/W	0d32	luma's horizontal adaptive coef1 . signed , default = 32
9:0	R/W	0d16	luma's horizontal adaptive coef2 . signed , default = 16

**Table 9-1880 SRSHARP0\_NR\_GAU\_YH\_COEF34 0x50b3**

Bit(s)	R/W	Default	Description
19:10	R/W	0d0	luma's horizontal adaptive coef3 . signed , default = 0
9:0	R/W	0d0	luma's horizontal adaptive coef4 . signed , default = 0

**Table 9-1881 SRSHARP0\_NR\_GAU\_YV\_COEF1 0x50b4**

Bit(s)	R/W	Default	Description
29:20	R/W	0d32	luma's vertical adaptive coef0 . signed , default = 32
19:10	R/W	0d32	luma's vertical adaptive coef1 . signed , default = 32
9:0	R/W	0d16	luma's vertical adaptive coef2 . signed , default = 16

**Table 9-1882 SRSHARP0\_NR\_GAU\_CH\_COEF02 0x50b5**

Bit(s)	R/W	Default	Description
29:20	R/W	0d32	chroma's horizontal adaptive coef0 . signed , default = 32
19:10	R/W	0d32	chroma's horizontal adaptive coef1 . signed , default = 32
9:0	R/W	0d16	chroma's horizontal adaptive coef2 . signed , default = 16

**Table 9-1883 SRSHARP0\_NR\_GAU\_CH\_COEF34 0x50b6**

Bit(s)	R/W	Default	Description
19:10	R/W	0d0	chroma's horizontal adaptive coef3 . signed , default = 0
9:0	R/W	0d0	chroma's horizontal adaptive coef4 . signed , default = 0

**Table 9-1884 SRSHARP0\_NR\_GAU\_CV\_COEF1 0x50b7**

Bit(s)	R/W	Default	Description
29:20	R/W	0d32	chroma's vertical adaptive coef0 . signed , default = 32
19:10	R/W	0d32	chroma's vertical adaptive coef1 . signed , default = 32
9:0	R/W	0d16	chroma's vertical adaptive coef2 . signed , default = 16

**Table 9-1885 SRSHARP0\_DB\_FLT\_CTRL1 0x50b8**

Bit(s)	R/W	Default	Description
17:16	R/W	0d2	reg_nrdeband_noise_rs .unsigned , default = 2
15:12	R/W	0d8	reg_nrdeband_randgain .unsigned , default = 8

**Table 9-1886 SRSHARP0\_DB\_FLT\_LUMA\_THRD 0x50b9**

Bit(s)	R/W	Default	Description
29:24	R/W	36	reg_nrdeband_luma_th3 : // unsigned , default = 36 threshold to  Y-Y lpf, if < th[0] use lpf
21:16	R/W	28	reg_nrdeband_luma_th2 : // unsigned , default = 28 elseif <th[1] use (lpf*3 + y)/4
13:8	R/W	24	reg_nrdeband_luma_th1 : // unsigned , default = 24 elseif <th[1] use (lpf*3 + y)/4elseif <th[2] (lpf*1 + y)/2
5:0	R/W	20	reg_nrdeband_luma_th0 : // unsigned , default = 20 elseif <th[1] use (lpf*3 + y)/4elseif elseif <th[3] (lpf*1 + 3*y)/4; else

**Table 9-1887 SRSHARP0\_DB\_FLT\_CHRM\_THRD 0x50ba**

Bit(s)	R/W	Default	Description
29:24	R/W	36	reg_nrdeband_chrm_th3 : // unsigned , default = 36 threshold to  Y-Y lpf, if < th[0] use lpf
21:16	R/W	28	reg_nrdeband_chrm_th2 : // unsigned , default = 28 elseif <th[1] use (lpf*3 + y)/4
13:8	R/W	24	reg_nrdeband_chrm_th1 : // unsigned , default = 24 elseif <th[1] use (lpf*3 + y)/4elseif <th[2] (lpf*1 + y)/2
5:0	R/W	20	reg_nrdeband_chrm_th0 : // unsigned , default = 20 elseif <th[1] use (lpf*3 + y)/4elseif elseif <th[3] (lpf*1 + 3*y)/4; else

**Table 9-1888 SRSHARP0\_FMETER\_CTRL 0x5089**

Bit(s)	R/W	Default	Description
11:10	R/W	0	reg_fmeter_vwin_mm : vertical window size, 0:1 cloumn,1:3cloumn, 2or3:5cloumn.unsigned,default = 0
9:8	R/W	0	reg_fmeter_hwin_mm : horizontal window size, 0:1x7, 1:1x9, 2or3: 1x11 .unsigned , default = 0
7	R/W	0	reg_fmeter_d2_mode : selectino filter D2, 0: [0 -2 0 0 2], 1: [-2 0 0 0 2] .unsigned , default = 0
6	R/W	0	reg_fmeter_v2_mode : selection filter V2, 0: [0 -2 0 0 2], 1: [-2 0 0 0 2] .unsigned , default = 0
5:4	R/W	0	reg_fmeter_h2_mode: selection filter H2, 0: [0 0 0 -2 0 0 2 0 0], 1: [-2 0 0 0 2], 2or3: [0-2 0 0 0 0 2 0] .unsigned , default = 0
0	R/W	0	reg_freq_meter_en: freq meter enable .unsigned , default = 0

**Table 9-1889 SRSHARP0\_FMETER\_WIN\_HOR 0x508a**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	reg_fmeter_xwin_ed: window location: hend .unsigned , default = 1920
12:0	R/W	0	reg_fmeter_xwin_st: window location: hstart .unsigned , default = 0

**Table 9-1890 SRSHARP0\_FMETER\_WIN\_VER 0x508b**

Bit(s)	R/W	Default	Description
28:16	R/W	1080	reg_fmeter_ywin_ed: window location: vend .unsigned , default = 1080
12:0	R/W	0	reg_fmeter_ywin_st: window location: vstart .unsigned , default = 0

**Table 9-1891 SRSHARP0\_FMETER\_CORING 0x508c**

Bit(s)	R/W	Default	Description
31:24	R/W	4	reg_fmeter_low_bound: low bound for threshold .unsigned , default = 4
23:16	R/W	4	reg_fmeter_coring_d: coring of diff before compare with threshold for diagonal frequency .unsigned , default = 4
15:8	R/W	4	reg_fmeter_coring_v: coring of diff before compare with threshold for vertical frequency .unsigned , default = 4
7:0	R/W	4	reg_fmeter_coring_h: coring of diff before compare with threshold for horizontal frequency .unsigned , default = 4

**Table 9-1892 SRSHARP0\_FMETER\_RATIO\_H 0x508d**

Bit(s)	R/W	Default	Description
21:16	R/W	16	reg_fmeter_ratio_h2: ratio for threshold calc. for horizontal frequency, 16 is normalized as "1" .unsigned , default = 16
13:8	R/W	16	reg_fmeter_ratio_h1: ratio for threshold calc. for horizontal frequency, 16 is normalized as "1" .unsigned , default = 16
5:0	R/W	16	reg_fmeter_ratio_h0: ratio for threshold calc. for horizontal frequency, 16 is normalized as "1" .unsigned , default = 16

**Table 9-1893 SRSHARP0\_FMETER\_RATIO\_V 0x508e**

Bit(s)	R/W	Default	Description
21:16	R/W	16	reg_fmeter_ratio_v2: ratio for threshold calc. for vertical frequency, 16 is normalized as "1" .unsigned , default = 16
13:8	R/W	16	reg_fmeter_ratio_v1: ratio for threshold calc. for vertical frequency, 16 is normalized as "1" .unsigned , default = 16
5:0	R/W	16	reg_fmeter_ratio_v0: ratio for threshold calc. for vertical frequency, 16 is normalized as "1" .unsigned , default = 16

**Table 9-1894 SRSHARP0\_FMETER\_RATIO\_D 0x508f**

Bit(s)	R/W	Default	Description
21:16	R/W	16	reg_fmeter_ratio_d2: ratio for threshold calc. for diagonal frequency, 16 is normalized as "1" .unsigned , default = 16
13:8	R/W	16	reg_fmeter_ratio_d1: ratio for threshold calc. for diagonal frequency, 16 is normalized as "1" .unsigned , default = 16
5:0	R/W	16	reg_fmeter_ratio_d0: ratio for threshold calc. for diagonal frequency, 16 is normalized as "1" .unsigned , default = 16

**Table 9-1895 SRSHARP0\_RO\_FMETER\_HCNT\_TYPE0 0x5046**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_hcnt_type0: count for horizontal frequency

**Table 9-1896 SRSHARP0\_RO\_FMETER\_HCNT\_TYPE1 0x5047**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_hcnt_type1: count for horizontal frequency

**Table 9-1897 SRSHARP0\_RO\_FMETER\_HCNT\_TYPE2 0x5048**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_hcnt_type2: count for horizontal frequency

**Table 9-1898 SRSHARP0\_RO\_FMETER\_HCNT\_TYPE3 0x5049**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_hcnt_type3: count for horizontal frequency

**Table 9-1899 SRSHARP0\_RO\_FMETER\_VCNT\_TYPE0 0x504a**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_vcnt_type0: count for vertical frequency

**Table 9-1900 SRSHARP0\_RO\_FMETER\_VCNT\_TYPE1 0x504b**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_vcnt_type1: count for vertical frequency

**Table 9-1901 SRSHARP0\_RO\_FMETER\_VCNT\_TYPE2 0x504c**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_vcnt_type2: count for vertical frequency

**Table 9-1902 SRSHARP0\_RO\_FMETER\_VCNT\_TYPE3 0x504d**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_vcnt_type3: count for vertical frequency

**Table 9-1903 SRSHARP0\_RO\_FMETER\_PDCNT\_TYPE0 0x504e**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_pdcnt_type0: count for positive diagonal frequency



**Table 9-1904 SRSHARP0\_RO\_FMETER\_PDCNT\_TYPE1 0x504f**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_pdcnt_type1: count for positive diagonal frequency

**Table 9-1905 SRSHARP0\_RO\_FMETER\_PDCNT\_TYPE2 0x5050**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_pdcnt_type2: count for positive diagonal frequency

**Table 9-1906 SRSHARP0\_RO\_FMETER\_PDCNT\_TYPE3 0x5051**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_pdcnt_type3: count for positive diagonal frequency

**Table 9-1907 SRSHARP0\_RO\_FMETER\_NDCNT\_TYPE0 0x5052**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_ndcnt_type0: count for negative diagonal frequency

**Table 9-1908 SRSHARP0\_RO\_FMETER\_NDCNT\_TYPE1 0x5053**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_ndcnt_type1: count for negative diagonal frequency

**Table 9-1909 SRSHARP0\_RO\_FMETER\_NDCNT\_TYPE2 0x5054**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_ndcnt_type2: count for negative diagonal frequency

**Table 9-1910 SRSHARP0\_RO\_FMETER\_NDCNT\_TYPE3 0x5055**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_ndcnt_type3: count for negative diagonal frequency

**Table 9-1911 SRSHARP0\_SR7\_DRTLPF\_EN 0x5100**

Bit(s)	R/W	Default	Description
31:11			reserved
10	RW	0	reg_sr7_drtlpf_beta_en2: enable of direction ambiguity protection for drt_lpf, beta for drt filter coef base on the x=cal_drt_dif8(min_idx,min2_idx)
9	RW	0	reg_sr7_drtlpf_beta_en1: enable of direction ambiguity protection for drt_lpf, beta for drt filter coef base on the x=cal_drt_dif8(min_idx,min2_idx)

Bit(s)	R/W	Default	Description
8	RW	0	reg_sr7_drtlpf_beta_en0: enable of direction ambiguity protection for drt_lpf, beta for drt filter coef base on the $x=cal\_drt\_dif8(min\_idx,min2\_idx)$
7:6			reserved
5	RW	0	reg_sr7_drtlpf_edge_en2: enable of direction lpf based on edge strength
4	RW	0	reg_sr7_drtlpf_edge_en1: enable of direction lpf based on edge strength
3	RW	0	reg_sr7_drtlpf_edge_en0: enable of direction lpf based on edge strength
2	RW	1	reg_sr7_drtlpf_sdfd_en2: sdfd gamma (HF burst compare to real edge protection) enable
1	RW	1	reg_sr7_drtlpf_sdfd_en1: sdfd gamma (HF burst compare to real edge protection) enable.
0	RW	1	reg_sr7_drtlpf_sdfd_en0: sdfd gamma (HF burst compare to real edge protection) enable.

Table 9-1912 SRSHARP0\_SR7\_DRTLPF\_BETA 0x5101

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	0	reg_sr7_drtlpf_beta3: beta for drt filter coef base on the $x=cal\_drt\_dif8(min\_idx,min2\_idx)$
11:8	RW	4	reg_sr7_drtlpf_beta2: beta for drt filter coef base on the $x=cal\_drt\_dif8(min\_idx,min2\_idx)$
7:4	RW	8	reg_sr7_drtlpf_beta1: beta for drt filter coef base on the $x=cal\_drt\_dif8(min\_idx,min2\_idx)$
3:0	RW	15	reg_sr7_drtlpf_beta0: beta for drt filter coef base on the $x=cal\_drt\_dif8(min\_idx,min2\_idx)$ , $beta = lut[x-1]$ , the larger of $x$ , means the higher possibility for ambiguity, $beta=0$ use org wo lpf.

Table 9-1913 SRSHARP0\_SR7\_PKBLD\_BETA 0x5102

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	0	reg_sr7_pkdrtbld_beta3: beta for drt and cir blend base on the $x=cal\_drt\_dif8(min\_idx,min2\_idx)$
11:8	RW	4	reg_sr7_pkdrtbld_beta2: beta for drt and cir blend base on the $x=cal\_drt\_dif8(min\_idx,min2\_idx)$
7:4	RW	8	reg_sr7_pkdrtbld_beta1: beta for drt and cir blend base on the $x=cal\_drt\_dif8(min\_idx,min2\_idx)$
3:0	RW	15	reg_sr7_pkdrtbld_beta0: beta for drt and cir blend base on the $x=cal\_drt\_dif8(min\_idx,min2\_idx)$ , $beta = lut[x-1]$ , the larger of $x$ , means the higher possibility for ambiguity, $beta=0$ use org wo lpf.

**Table 9-1914 SRSHARP0\_SR7\_XLTIBLD\_BETA 0x5103**

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	0	reg_sr7_xlti_dcblld_beta3: beta for denoise and org dc blend base on the x=cal_drt_dif8(min_idx,min2_idx)
11:8	RW	4	reg_sr7_xlti_dcblld_beta2: beta for denoise and org dc blend base on the x=cal_drt_dif8(min_idx,min2_idx)
7:4	RW	8	reg_sr7_xlti_dcblld_beta1: beta for denoise and org dc blend base on the x=cal_drt_dif8(min_idx,min2_idx)
3:0	RW	15	reg_sr7_xlti_dcblld_beta0: beta for denoise and org dc blend base on the x=cal_drt_dif8(min_idx,min2_idx) beta = lut[x-1], the larger of x, means the higher possibility for ambiguity, beta=0 use org wo lpf.

**Table 9-1915 SRSHARP0\_SR7\_DRTLPF\_EDGE0 0x5104**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drtlpf_edge7: edge lamda for drt lpf base on max_sad [0:16:128~255]
27:24	RW	15	reg_sr7_drtlpf_edge6: edge lamda for drt lpf base on max_sad [0:16:128~255]
23:20	RW	15	reg_sr7_drtlpf_edge5: edge lamda for drt lpf base on max_sad [0:16:128~255]
19:16	RW	15	reg_sr7_drtlpf_edge4: edge lamda for drt lpf base on max_sad [0:16:128~255]
15:12	RW	15	reg_sr7_drtlpf_edge3: edge lamda for drt lpf base on max_sad [0:16:128~255]
11:8	RW	15	reg_sr7_drtlpf_edge2: edge lamda for drt lpf base on max_sad [0:16:128~255]
7:4	RW	15	reg_sr7_drtlpf_edge1: edge lamda for drt lpf base on max_sad [0:16:128~255]
3:0	RW	15	reg_sr7_drtlpf_edge0: edge lamda for drt lpf base on max_sad [0:16:128~255]

**Table 9-1916 SRSHARP0\_SR7\_DRTLPF\_EDGE1 0x5105**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drtlpf_edge15: edge lamda for drt lpf base on max_sad [0:16:128~255]
27:24	RW	15	reg_sr7_drtlpf_edge14: edge lamda for drt lpf base on max_sad [0:16:128~255]
23:20	RW	15	reg_sr7_drtlpf_edge13: edge lamda for drt lpf base on max_sad [0:16:128~255]
19:16	RW	15	reg_sr7_drtlpf_edge12: edge lamda for drt lpf base on max_sad [0:16:128~255]
15:12	RW	15	reg_sr7_drtlpf_edge11: edge lamda for drt lpf base on max_sad [0:16:128~255]
11:8	RW	15	reg_sr7_drtlpf_edge10: edge lamda for drt lpf base on max_sad [0:16:128~255]
7:4	RW	15	reg_sr7_drtlpf_edge9: edge lamda for drt lpf base on max_sad [0:16:128~255]
3:0	RW	15	reg_sr7_drtlpf_edge8: edge lamda for drt lpf base on max_sad [0:16:128~255]

**Table 9-1917 SRSHARP0\_SR7\_DRTLPF\_SDCOR0 0x5106**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drtlpf_sdcor7: gamma for drt lpf base on SD'[0:16:128~255]
27:24	RW	15	reg_sr7_drtlpf_sdcor6: gamma for drt lpf base on SD'[0:16:128~255]
23:20	RW	15	reg_sr7_drtlpf_sdcor5: gamma for drt lpf base on SD'[0:16:128~255]
19:16	RW	15	reg_sr7_drtlpf_sdcor4: gamma for drt lpf base on SD'[0:16:128~255]
15:12	RW	15	reg_sr7_drtlpf_sdcor3: gamma for drt lpf base on SD'[0:16:128~255]
11:8	RW	15	reg_sr7_drtlpf_sdcor2: gamma for drt lpf base on SD'[0:16:128~255]
7:4	RW	15	reg_sr7_drtlpf_sdcor1: gamma for drt lpf base on SD'[0:16:128~255]
3:0	RW	15	reg_sr7_drtlpf_sdcor0: gamma for drt lpf base on SD'[0:16:128~255]

**Table 9-1918 SRSHARP0\_SR7\_DRTLPF\_SDCOR1 0x5107**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drtlpf_sdcor15: gamma for drt lpf base on SD'[0:16:128~255]
27:24	RW	15	reg_sr7_drtlpf_sdcor14: gamma for drt lpf base on SD'[0:16:128~255]
23:20	RW	15	reg_sr7_drtlpf_sdcor13: gamma for drt lpf base on SD'[0:16:128~255]
19:16	RW	15	reg_sr7_drtlpf_sdcor12: gamma for drt lpf base on SD'[0:16:128~255]
15:12	RW	15	reg_sr7_drtlpf_sdcor11: gamma for drt lpf base on SD'[0:16:128~255]
11:8	RW	15	reg_sr7_drtlpf_sdcor10: gamma for drt lpf base on SD'[0:16:128~255]
7:4	RW	15	reg_sr7_drtlpf_sdcor9: gamma for drt lpf base on SD'[0:16:128~255]
3:0	RW	15	reg_sr7_drtlpf_sdcor8: gamma for drt lpf base on SD'[0:16:128~255]

**Table 9-1919 SRSHARP0\_SR7\_CTIGAIN\_SDCOR0 0x5108**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_ctigain_sdcor7: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
27:24	RW	15	reg_sr7_ctigain_sdcor6: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
23:20	RW	15	reg_sr7_ctigain_sdcor5: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
19:16	RW	15	reg_sr7_ctigain_sdcor4: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
15:12	RW	15	reg_sr7_ctigain_sdcor3: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
11:8	RW	15	reg_sr7_ctigain_sdcor2: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
7:4	RW	15	reg_sr7_ctigain_sdcor1: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
3:0	RW	15	reg_sr7_ctigain_sdcor0: adaptive gamma for cti boost gain base on SD'[0:16:128~255]

**Table 9-1920 SRSHARP0\_SR7\_CTIGAIN\_SDCOR1 0x5109**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_ctigain_sdcor15: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
27:24	RW	15	reg_sr7_ctigain_sdcor14: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
23:20	RW	15	reg_sr7_ctigain_sdcor13: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
19:16	RW	15	reg_sr7_ctigain_sdcor12: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
15:12	RW	15	reg_sr7_ctigain_sdcor11: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
11:8	RW	15	reg_sr7_ctigain_sdcor10: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
7:4	RW	15	reg_sr7_ctigain_sdcor9: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
3:0	RW	15	reg_sr7_ctigain_sdcor8: adaptive gamma for cti boost gain base on SD'[0:16:128~255]

**Table 9-1921 SRSHARP0\_SR7\_LTIGAIN\_SDCOR0 0x510a**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_ltigain_sdcor7: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
27:24	RW	15	reg_sr7_ltigain_sdcor6: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
23:20	RW	15	reg_sr7_ltigain_sdcor5: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
19:16	RW	15	reg_sr7_ltigain_sdcor4: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
15:12	RW	15	reg_sr7_ltigain_sdcor3: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
11:8	RW	15	reg_sr7_ltigain_sdcor2: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
7:4	RW	15	reg_sr7_ltigain_sdcor1: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
3:0	RW	15	reg_sr7_ltigain_sdcor0: adaptive gamma for lti boost gain base on SD'[0:16:128~255]

**Table 9-1922 SRSHARP0\_SR7\_LTIGAIN\_SDCOR1 0x510b**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_ltigain_sdcor15: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
27:24	RW	15	reg_sr7_ltigain_sdcor14: adaptive gamma for lti boost gain base on SD'[0:16:128~255]

Bit(s)	R/W	Default	Description
23:20	RW	15	reg_sr7_ltigain_sdcor13: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
19:16	RW	15	reg_sr7_ltigain_sdcor12: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
15:12	RW	15	reg_sr7_ltigain_sdcor11: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
11:8	RW	15	reg_sr7_ltigain_sdcor10: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
7:4	RW	15	reg_sr7_ltigain_sdcor9: adaptive gamma for lti boost gain base on SD'[0:16:128~255]
3:0	RW	15	reg_sr7_ltigain_sdcor8: adaptive gamma for lti boost gain base on SD'[0:16:128~255]

Table 9-1923 SRSHARP0\_SR7\_HLTIBPF\_TAP0 0x510c

Bit(s)	R/W	Default	Description
31:24	RW	0	reg_sr7_hlti_bpf_tap153: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
23:16	RW	-34	reg_sr7_hlti_bpf_tap152: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
15:8	RW	-1	reg_sr7_hlti_bpf_tap151: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0, signed
7:0	RW	120	reg_sr7_hlti_bpf_tap150: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed

Table 9-1924 SRSHARP0\_SR7\_HLTIBPF\_TAP1 0x510d

Bit(s)	R/W	Default	Description
31:24	RW	2	reg_sr7_hlti_bpf_tap157: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
23:16	RW	-7	reg_sr7_hlti_bpf_tap156: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
15:8	RW	1	reg_sr7_hlti_bpf_tap155: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
7:0	RW	-21	reg_sr7_hlti_bpf_tap154: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed

Table 9-1925 SRSHARP0\_SR7\_HCTIBPF\_TAP0 0x510e

Bit(s)	R/W	Default	Description
31:24	RW	0	reg_sr7_clti_bpf_tap153: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed
23:16	RW	-34	reg_sr7_clti_bpf_tap152: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed

Bit(s)	R/W	Default	Description
15:8	RW	-1	reg_sr7_clti_bpf_tap151: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed
7:0	RW	120	reg_sr7_clti_bpf_tap150: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed

**Table 9-1926 SRSHARP0\_SR7\_HCTIBPF\_TAP1 0x510f**

Bit(s)	R/W	Default	Description
31:24	RW	2	reg_sr7_clti_bpf_tap157: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed
23:16	RW	-7	reg_sr7_clti_bpf_tap156: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed
15:8	RW	1	reg_sr7_clti_bpf_tap155: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed
7:0	RW	-21	reg_sr7_clti_bpf_tap154: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed

**Table 9-1927 SRSHARP0\_SR7\_PKLONGBPF\_HTAP0 0x5110**

Bit(s)	R/W	Default	Description
31:24	RW	-24	reg_sr7_pk_long_bpf_hztap153: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
23:16	RW	-19	reg_sr7_pk_long_bpf_hztap152: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
15:8	RW	31	reg_sr7_pk_long_bpf_hztap151: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
7:0	RW	66	reg_sr7_pk_long_bpf_hztap150: tap15 BPF for horizontal peaking filter, only store half of the filter ,signed

**Table 9-1928 SRSHARP0\_SR7\_PKLONGBPF\_HTAP1 0x5111**

Bit(s)	R/W	Default	Description
31:24	RW	-6	reg_sr7_pk_long_bpf_hztap157: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
23:16	RW	-10	reg_sr7_pk_long_bpf_hztap156: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
15:8	RW	-1	reg_sr7_pk_long_bpf_hztap155: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
7:0	RW	-4	reg_sr7_pk_long_bpf_hztap154: tap15 BPF for horizontal peaking filter, only store half of the filter,signed

**Table 9-1929 SRSHARP0\_SR7\_PKLONGHPF\_HTAP0 0x5112**

Bit(s)	R/W	Default	Description
31:24	RW	27	reg_sr7_pk_long_hpf_hztap153: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
23:16	RW	-15	reg_sr7_pk_long_hpf_hztap152: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
15:8	RW	-32	reg_sr7_pk_long_hpf_hztap151: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
7:0	RW	58	reg_sr7_pk_long_hpf_hztap150: tap15 HPF for horizontal peaking filter, only store half of the filter,signed

**Table 9-1930 SRSHARP0\_SR7\_PKLONGHPF\_HTAP1 0x5113**

Bit(s)	R/W	Default	Description
31:24	RW	8	reg_sr7_pk_long_hpf_hztap157: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
23:16	RW	-6	reg_sr7_pk_long_hpf_hztap156: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
15:8	RW	-1	reg_sr7_pk_long_hpf_hztap155: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
7:0	RW	-10	reg_sr7_pk_long_hpf_hztap154: tap15 HPF for horizontal peaking filter, only store half of the filter,signed

**Table 9-1931 SRSHARP0\_SR7\_VLTIBPF\_TAP0 0x5114**

Bit(s)	R/W	Default	Description
31:24	RW	0	reg_sr7_vlti_bpf_tap093: tap09 BPF for vlti (gain3), only store half of the filter, signed
23:16	RW	-41	reg_sr7_vlti_bpf_tap092: tap09 BPF for vlti (gain3), only store half of the filter, signed
15:8	RW	0	reg_sr7_vlti_bpf_tap091: tap09 BPF for vlti (gain3), only store half of the filter, signed
7:0	RW	126	reg_sr7_vlti_bpf_tap090: tap09 BPF for vlti (gain3), only store half of the filter, signed

**Table 9-1932 SRSHARP0\_SR7\_VLTIBPF\_TAP1 0x5115**

Bit(s)	R/W	Default	Description
31:8			reserved
7:0	RW	-22	reg_sr7_vlti_bpf_tap094: tap09 BPF for vlti (gain3), only store half of the filter, signed



**Table 9-1933 SRSHARP0\_SR7\_VCTIBPF\_TAP0 0x5116**

Bit(s)	R/W	Default	Description
31:24	RW	0	reg_sr7_vcti_bpf_tap093: tap09 BPF for vcti (gain3), only store half of the filter, signed
23:16	RW	-41	reg_sr7_vcti_bpf_tap092: tap09 BPF for vcti (gain3), only store half of the filter, signed
15:8	RW	0	reg_sr7_vcti_bpf_tap091: tap09 BPF for vcti (gain3), only store half of the filter, signed
7:0	RW	126	reg_sr7_vcti_bpf_tap090: tap09 BPF for vcti (gain3), only store half of the filter, signed

**Table 9-1934 SRSHARP0\_SR7\_VCTIBPF\_TAP1 0x5117**

Bit(s)	R/W	Default	Description
31:8			reserved
7:0	RW	-22	reg_sr7_vcti_bpf_tap094: tap09 BPF for vcti (gain3), only store half of the filter,, signed

**Table 9-1935 SRSHARP0\_SR7\_PKLONGBPF\_VTAP0 0x5118**

Bit(s)	R/W	Default	Description
31:24	RW	-32	reg_sr7_pk_long_bpf_vdtap093: tap09 BPF for vertical and diagonal peaking filter,signed
23:16	RW	-28	reg_sr7_pk_long_bpf_vdtap092: tap09 BPF for vertical and diagonal peaking filter,signed
15:8	RW	30	reg_sr7_pk_long_bpf_vdtap091: tap09 BPF for vertical and diagonal peaking filter,signed
7:0	RW	68	reg_sr7_pk_long_bpf_vdtap090: tap09 BPF for vertical and diagonal peaking filter,signed

**Table 9-1936 SRSHARP0\_SR7\_PKLONGBPF\_VTAP1 0x5119**

Bit(s)	R/W	Default	Description
31:8			reserved
7:0	RW	-4	reg_sr7_pk_long_bpf_vdtap094: tap09 BPF for vertical and diagonal peaking filter,signed

**Table 9-1937 SRSHARP0\_SR7\_PKLONGHPF\_VTAP0 0x511a**

Bit(s)	R/W	Default	Description
31:24	RW	30	reg_sr7_pk_long_hpf_vdtap093: tap09 HPF for vertical and diagonal peaking filter,signed
23:16	RW	-28	reg_sr7_pk_long_hpf_vdtap092: tap09 HPF for vertical and diagonal peaking filter,signed

Bit(s)	R/W	Default	Description
15:8	RW	-31	reg_sr7_pk_long_hpf_vdtap091: tap09 HPF for vertical and diagonal peaking filter,signed
7:0	RW	68	reg_sr7_pk_long_hpf_vdtap090: tap09 HPF for vertical and diagonal peaking filter,signed

Table 9-1938 SRSHARP0\_SR7\_PKLONGHPF\_VTAP1 0x511b

Bit(s)	R/W	Default	Description
31:8			reserved
7:0	RW	-5	reg_sr7_pk_long_hpf_vdtap094: tap09 HPF for vertical and diagonal peaking filter,signed

Table 9-1939 SRSHARP0\_SR7\_CIRBPLONG\_ALP 0x511c

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	15	reg_sr7_cirbp_long_alpha3: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
11:8	RW	15	reg_sr7_cirbp_long_alpha2: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
7:4	RW	15	reg_sr7_cirbp_long_alpha1: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
3:0	RW	15	reg_sr7_cirbp_long_alpha0: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)

Table 9-1940 SRSHARP0\_SR7\_CIRHPLONG\_ALP 0x511d

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	15	reg_sr7_cirhp_long_alpha3: alpha to blend hp_long to the cirpk filter corresponding angle (0/45/90/135)
11:8	RW	15	reg_sr7_cirhp_long_alpha2: alpha to blend hp_long to the cirpk filter corresponding angle (0/45/90/135)
7:4	RW	15	reg_sr7_cirhp_long_alpha1: alpha to blend hp_long to the cirpk filter corresponding angle (0/45/90/135)
3:0	RW	15	reg_sr7_cirhp_long_alpha0: alpha to blend hp_long to the cirpk filter corresponding angle (0/45/90/135)

Table 9-1941 SRSHARP0\_SR7\_DRTBPLONG\_ALP 0x511e

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drtbp_long_alpha7: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
27:24	RW	15	reg_sr7_drtbp_long_alpha6: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)

Bit(s)	R/W	Default	Description
23:20	RW	15	reg_sr7_drtbp_long_alpha5: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
19:16	RW	15	reg_sr7_drtbp_long_alpha4: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
15:12	RW	15	reg_sr7_drtbp_long_alpha3: alpha to blend bp_long to the drtpk filter corresponding angle (0/45/90/135)
11:8	RW	15	reg_sr7_drtbp_long_alpha2: alpha to blend bp_long to the drtpk filter corresponding angle (0/45/90/135)
7:4	RW	15	reg_sr7_drtbp_long_alpha1: alpha to blend bp_long to the drtpk filter corresponding angle (0/45/90/135)
3:0	RW	15	reg_sr7_drtbp_long_alpha0: alpha to blend bp_long to the drtpk filter corresponding angle (0/45/90/135)

Table 9-1942 SRSHARP0\_SR7\_DRTHPLONG\_ALP 0x511f

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drthp_long_alpha7: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
27:24	RW	15	reg_sr7_drthp_long_alpha6: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
23:20	RW	15	reg_sr7_drthp_long_alpha5: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
19:16	RW	15	reg_sr7_drthp_long_alpha4: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
15:12	RW	15	reg_sr7_drthp_long_alpha3: alpha to blend hp_long to the drtpk filter corresponding angle (0/45/90/135)
11:8	RW	15	reg_sr7_drthp_long_alpha2: alpha to blend hp_long to the drtpk filter corresponding angle (0/45/90/135)
7:4	RW	15	reg_sr7_drthp_long_alpha1: alpha to blend hp_long to the drtpk filter corresponding angle (0/45/90/135)
3:0	RW	15	reg_sr7_drthp_long_alpha0: alpha to blend hp_long to the drtpk filter corresponding angle (0/45/90/135)

Table 9-1943 SRSHARP0\_SR7\_PKMINMAXCIR\_BLD\_LUT2D0 0x5120

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_pk_mimaxerr2_cirbld_lut2d7
27:24	RW	0	reg_sr7_pk_mimaxerr2_cirbld_lut2d6
23:20	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d5
19:16	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d4
15:12	RW	4	reg_sr7_pk_mimaxerr2_cirbld_lut2d3
11:8	RW	6	reg_sr7_pk_mimaxerr2_cirbld_lut2d2

Bit(s)	R/W	Default	Description
7:4	RW	12	reg_sr7_pk_mimaxerr2_cirbld_lut2d1
3:0	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d0: 2D-LUT for alpha2 for cir-PK blender, larger coef means less drt-pk, x-min_err, y-max_err

**Table 9-1944 SRSHARP0\_SR7\_PKMINMAXCIR\_BLD\_LUT2D1 0x5121**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d15
27:24	RW	0	reg_sr7_pk_mimaxerr2_cirbld_lut2d14
23:20	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d13
19:16	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d12
15:12	RW	4	reg_sr7_pk_mimaxerr2_cirbld_lut2d11
11:8	RW	6	reg_sr7_pk_mimaxerr2_cirbld_lut2d10
7:4	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d9
3:0	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d8

**Table 9-1945 SRSHARP0\_SR7\_PKMINMAXCIR\_BLD\_LUT2D2 0x5122**

Bit(s)	R/W	Default	Description
31:28	RW	6	reg_sr7_pk_mimaxerr2_cirbld_lut2d23
27:24	RW	10	reg_sr7_pk_mimaxerr2_cirbld_lut2d22
23:20	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d21
19:16	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d20
15:12	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d19
11:8	RW	4	reg_sr7_pk_mimaxerr2_cirbld_lut2d18
7:4	RW	6	reg_sr7_pk_mimaxerr2_cirbld_lut2d17
3:0	RW	12	reg_sr7_pk_mimaxerr2_cirbld_lut2d16

**Table 9-1946 SRSHARP0\_SR7\_PKMINMAXCIR\_BLD\_LUT2D3 0x5123**

Bit(s)	R/W	Default	Description
31:28	RW	14	reg_sr7_pk_mimaxerr2_cirbld_lut2d31
27:24	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d30
23:20	RW	4	reg_sr7_pk_mimaxerr2_cirbld_lut2d29
19:16	RW	8	reg_sr7_pk_mimaxerr2_cirbld_lut2d28
15:12	RW	10	reg_sr7_pk_mimaxerr2_cirbld_lut2d27
11:8	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d26

Bit(s)	R/W	Default	Description
7:4	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d25
3:0	RW	4	reg_sr7_pk_mimaxerr2_cirbld_lut2d24

Table 9-1947 SRSARP0\_SR7\_PKMINMAXCIR\_BLD\_LUT2D4 0x5124

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d35
11:8	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d34
7:4	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d33
3:0	RW	12	reg_sr7_pk_mimaxerr2_cirbld_lut2d32

Table 9-1948 SRSARP0\_SR7\_PKMINMAXLPF\_BLD\_LUT2D0 0x5125

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_pk_mimaxerr2_lpfbld_lut2d7
27:24	RW	0	reg_sr7_pk_mimaxerr2_lpfbld_lut2d6
23:20	RW	2	reg_sr7_pk_mimaxerr2_lpfbld_lut2d5
19:16	RW	2	reg_sr7_pk_mimaxerr2_lpfbld_lut2d4
15:12	RW	4	reg_sr7_pk_mimaxerr2_lpfbld_lut2d3
11:8	RW	6	reg_sr7_pk_mimaxerr2_lpfbld_lut2d2
7:4	RW	12	reg_sr7_pk_mimaxerr2_lpfbld_lut2d1
3:0	RW	15	reg_sr7_pk_mimaxerr2_lpfbld_lut2d0: 2D-LUT for alpha2 for PK lpf along edge blender, lareger coef means less lpf along edge,x-min_err, y-max_err

Table 9-1949 SRSARP0\_SR7\_PKMINMAXLPF\_BLD\_LUT2D1 0x5126

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_pk_mimaxerr2_lpfbld_lut2d15
27:24	RW	0	reg_sr7_pk_mimaxerr2_lpfbld_lut2d14
23:20	RW	2	reg_sr7_pk_mimaxerr2_lpfbld_lut2d13
19:16	RW	2	reg_sr7_pk_mimaxerr2_lpfbld_lut2d12
15:12	RW	4	reg_sr7_pk_mimaxerr2_lpfbld_lut2d11
11:8	RW	6	reg_sr7_pk_mimaxerr2_lpfbld_lut2d10
7:4	RW	15	reg_sr7_pk_mimaxerr2_lpfbld_lut2d9
3:0	RW	15	reg_sr7_pk_mimaxerr2_lpfbld_lut2d8

**Table 9-1950 SRSHARP0\_SR7\_PKMINMAXLPF\_BLD\_LUT2D2 0x5127**

Bit(s)	R/W	Default	Description
31:28	RW	6	reg_sr7_pk_mimaxerr2_lpfblld_lut2d23
27:24	RW	10	reg_sr7_pk_mimaxerr2_lpfblld_lut2d22
23:20	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d21
19:16	RW	2	reg_sr7_pk_mimaxerr2_lpfblld_lut2d20
15:12	RW	2	reg_sr7_pk_mimaxerr2_lpfblld_lut2d19
11:8	RW	4	reg_sr7_pk_mimaxerr2_lpfblld_lut2d18
7:4	RW	6	reg_sr7_pk_mimaxerr2_lpfblld_lut2d17
3:0	RW	12	reg_sr7_pk_mimaxerr2_lpfblld_lut2d16

**Table 9-1951 SRSHARP0\_SR7\_PKMINMAXLPF\_BLD\_LUT2D3 0x5128**

Bit(s)	R/W	Default	Description
31:28	RW	14	reg_sr7_pk_mimaxerr2_lpfblld_lut2d31
27:24	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d30
23:20	RW	4	reg_sr7_pk_mimaxerr2_lpfblld_lut2d29
19:16	RW	8	reg_sr7_pk_mimaxerr2_lpfblld_lut2d28
15:12	RW	10	reg_sr7_pk_mimaxerr2_lpfblld_lut2d27
11:8	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d26
7:4	RW	2	reg_sr7_pk_mimaxerr2_lpfblld_lut2d25
3:0	RW	4	reg_sr7_pk_mimaxerr2_lpfblld_lut2d24

**Table 9-1952 SRSHARP0\_SR7\_PKMINMAXLPF\_BLD\_LUT2D4 0x5129**

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d35
11:8	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d34
7:4	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d33
3:0	RW	12	reg_sr7_pk_mimaxerr2_lpfblld_lut2d32

**Table 9-1953 SRSHARP0\_SR7\_PKDRT\_BLD\_EN 0x512a**

Bit(s)	R/W	Default	Description
31:1			reserved
0	RW	0	reg_sr7_pkdrtdld_beta_en: enable of direction ambiguity protection for drt and cir blend

**Table 9-1954 SRSHARP0\_SR7\_DRTDIF\_TH 0x512b**

Bit(s)	R/W	Default	Description
31:16			reserved
15:8	RW	128	reg_sr7_drtdif_min2sad_th1
7:0	RW	128	reg_sr7_drtdif_min2sad_th0:for min2_sad threshold for ambiguity ignoring

**Table 9-1955 SRSHARP0\_SR7\_TIBLD\_PRT 0x512c**

Bit(s)	R/W	Default	Description
31:16			reserved
13	RW	1	reg_sr7_hlti_dcbld_beta_en:enable of denoise protection for dc org and denoise blend
12	RW	1	reg_sr7_vlti_dcbld_beta_en:enable of denoise protection for dc org and denoise blend
11:8	RW	0	reg_sr7_xcti_dcbld_beta1:beta for denoise and org dc blend base on the x=cal_drt_dif4(min_idx,min2_idx)
7:4	RW	15	reg_sr7_xcti_dcbld_beta0:beta for denoise and org dc blend base on the x=cal_drt_dif4(min_idx,min2_idx)
3	RW	1	reg_sr7_hcti_dcbld_beta_en:enable of denoise protection for dc org and denoise blend
2	RW	1	reg_sr7_vcti_dcbld_beta_en:enable of denoise protection for dc org and denoise blend
1	RW	0	reg_sr7_hcti_dcbld_use_ybeta:enable to use beta from hlti
0	RW	0	reg_sr7_vcti_dcbld_use_ybeta:enable to use beta from vlti

**Table 9-1956 SRSHARP0\_SR7\_HTI\_OPT\_FORCE 0x512d**

Bit(s)	R/W	Default	Description
31:6			reserved
7:6	RW	1	reg_sr7_diag_force_hti1:force hti for diagonal edges luma/chroma, 0: bypass, 1: diagonal use vti, 2:diagonal use hti
5:4	RW	1	reg_sr7_diag_force_hti0:force hti for diagonal edges luma/chroma, 0: bypass, 1: diagonal use vti, 2:diagonal use hti
3	RW	1	reg_sr7_horz_force_vti1:force vti for horizontal edges luma/chroma, 0: bias hti, 1: horizontal edge force 100% vti
2	RW	1	reg_sr7_horz_force_vti0:force vti for horizontal edges luma/chroma, 0: bias hti, 1: horizontal edge force 100% vti
1	RW	1	reg_sr7_alph_force_hvsad1:alpha = minsad*64/maxsad, force minsad=sad_h, maxsad=sad_v for alpha calculation
0	RW	1	reg_sr7_alph_force_hvsad0:alpha = minsad*64/maxsad, force minsad=sad_h, maxsad=sad_v for alpha calculation,[0]for luma, [1] for chroma

**Table 9-1957 SRSHARP0\_SR7\_HVTI\_FINALGAIN 0x512e**

Bit(s)	R/W	Default	Description
31:28			reserved
27:16	RW	256	reg_sr7_hvti_finalgain1: final gain for HVTI boost, for easier level tuning for application
15:12			reserved
11:0	RW	256	reg_sr7_hvti_finalgain0: final gain for HVTI boost, for easier level tuning for application

**Table 9-1958 SRSHARP0\_SR7\_TIOS\_SDRATIO 0x512f**

Bit(s)	R/W	Default	Description
31:30			reserved
29:24	RW	3	reg_sr7_hti_osmargin_sdratio1:ratio for HTI OS margin adaptive to sd
23:22			reserved
21:16	RW	3	reg_sr7_hti_osmargin_sdratio0:ratio for HTI OS margin adaptive to sd
15:14			reserved
13:8	RW	3	reg_sr7_vti_osmargin_sdratio1:ratio for VTI OS margin adaptive to sd
7:6			reserved
5:0	RW	3	reg_sr7_vti_osmargin_sdratio0:ratio for VTI OS margin adaptive to sd

**Table 9-1959 SRSHARP0\_SR7\_XTI\_SDFDEN 0x5130**

Bit(s)	R/W	Default	Description
31:1			reserved
1	RW	1	reg_sr7_xtigain_sdfd_en1: adaptive gamma for cti boost gain enable.
0	RW	1	reg_sr7_xtigain_sdfd_en0: adaptive gamma for cti boost gain enable.[0]for luma, [1] for chroma

**Table 9-1960 SRSHARP0\_SR7\_FDSD\_PARAM 0x5131**

Bit(s)	R/W	Default	Description
31:26			reserved
25	RW	1	reg_fdsd_vlpf_en:use vertical [1 2 1] lpf filter for the fd and sd derivativecalculations
24	RW	1	reg_fdsd_hlpf_en:use horizontal [1 2 1] lpf filter for the fd and sd derivativecalculations
23:20	RW	0	reg_sd_coring_th1:coring to SD'= MAX(SD - FD*ratio - coring,0)
19:16	RW	0	reg_sd_coring_th0:coring to SD'= MAX(SD - FD*ratio - coring,0)
15:14			reserved
13:8	RW	4	reg_sd_coring_ratio2fd1:ratio to FD for adaptive coring to SD'= MAX(SD - FD*ratio - coring,0)



Bit(s)	R/W	Default	Description
7:6			reserved
5:0	RW	4	reg_sd_coring_ratio2fd0:ratio to FD for adaptive coring to SD'= MAX(SD - FD*ratio - coring,0)

Table 9-1961 SRSHARP0\_SR7\_TI\_BPF\_EN 0x5132

Bit(s)	R/W	Default	Description
31:4			reserved
3	RW	1	reg_sr7_hlti_bpf_en:enable of tap09 BPF for hlti (gain3), only store half of the filter
2	RW	1	reg_sr7_hcti_bpf_en:enable of tap09 BPF for hcti (gain3), only store half of the filter
1	RW	1	reg_sr7_vlti_bpf_en:enable of tap09 BPF for vlti (gain3), only store half of the filter
0	RW	1	reg_sr7_vcti_bpf_en:enable of tap09 BPF for vlti (gain3), only store half of the filter

Table 9-1962 SRSHARP0\_SR7\_PKLONG\_PF\_EN 0x5133

Bit(s)	R/W	Default	Description
31:2			reserved
1	RW	1	reg_sr7_pk_long_bpf_en:enable of long BPF for peaking
0	RW	1	reg_sr7_pk_long_hpf_en:enable of long HPF for peaking

Table 9-1963 SRSHARP0\_SR7\_PKLONG\_PF\_GAIN 0x5134

Bit(s)	R/W	Default	Description
31:24	RW	64	reg_sr7_pk_long_bpf_hzgain:gain to long horizontal BPF for peaking
23:16	RW	64	reg_sr7_pk_long_bpf_vdgain:gain to long vertical and diagonal BPF for peaking
15:8	RW	64	reg_sr7_pk_long_hpf_hzgain:gain to long horizontal HPF for peaking
7:0	RW	64	reg_sr7_pk_long_hpf_vdgain:gain to long vertical and diagonal HPF for peaking

Table 9-1964 SRSHARP0\_SR7\_PKMINMAX\_BLD 0x5135

Bit(s)	R/W	Default	Description
31:4			reserved
3	RW	1	reg_sr7_pk_mimaxerr2_cirbld_on_bp:enable to use adaptive blender of drtBP vs cirBP alpha2, the larger of cell, the more cirPK results, 0: alp2=lut2d(minerr, maxerr),1:alp2=min_err/maxerr
2	RW	1	reg_sr7_pk_mimaxerr2_cirbld_on_hp:enable to use adaptive blender of drtHP vs cirHP alpha2, the larger of cell, the more cirPK results, 0: alp2=lut2d(minerr, maxerr),1:alp2=min_err/maxerr

Bit(s)	R/W	Default	Description
1	RW	1	reg_sr7_pk_mimaxerr2_lpfbd_on_bp:enable to use adaptive blender of BP result lpf along edge based on the mimaxsad relationship
0	RW	1	reg_sr7_pk_mimaxerr2_lpfbd_on_hp:enable to use adaptive blender of HP result lpf along edge based on the mimaxsad relationship

Table 9-1965 SRSHARP0\_SR7\_TI\_CONMAXERR\_GAIN 0x5136

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	8	reg_sr7_hti_conmaxerr_gain1:con=max(con, maxerr) for the horizontal transition calculation, gain=0, same as s6 and prev, [0] for y and [1] for uv
11:8	RW	8	reg_sr7_hti_conmaxerr_gain0:con=max(con, maxerr) for the horizontal transition calculation, gain=0, same as s6 and prev, [0] for y and [1] for uv
7:4	RW	8	reg_sr7_vti_conmaxerr_gain1:con=max(con, maxerr) for the vertical transition calculation, gain=0, same as s6 and prev, [0] for y and [1] for uv
3:0	RW	8	reg_sr7_vti_conmaxerr_gain0:con=max(con, maxerr) for the vertical transition calculation, gain=0, same as s6 and prev, [0] for y and [1] for uv

Table 9-1966 SRSHARP0\_SR7\_CC\_PK\_ADJ 0x5137

Bit(s)	R/W	Default	Description
31:25			reserved
24	RW	1	reg_sr7_cc_enable:color compensation enable, 0: no CC, 1: with CC
23:22			reserved
21:20	RW	0	reg_sr7_cc_yinp_sel:color compensation input luma selection, 0: org_y; 1: gau_y; 2:gauadp_y; 3:edgeadp_y (same as dnlp input sel)
19:18	RW	2	reg_sr7_cc_ydlt_sel:color compensation output luma selection, 0: peaking+lti output; 1: dnlp output; 2/3: peaking+lti+dnlp
17:16	RW	1	reg_sr7_cc_sat_norm:normalization of lut cell to saturation. 0: norm to 8 as 1.0, sat[0:23/8]; 1: norm to 16 as 1.0, sat[1/16:31/16]; 2: norm to 32 as 1.0, sat[17/32:47/32]; 3: norm to 64 as 1.0, sat[49/64:79/64]
15:8	RW	64	reg_sr7_cc_ydlt_psc1:prescale to the y-delta (if >0) before feeding to y-lumadlt, normalized 64 as 1.0
7:0	RW	64	reg_sr7_cc_ydlt_nsc1:prescale to the y-delta (if <0) before feeding to y-lumadlt, normalized 64 as 1.0

Table 9-1967 SRSHARP0\_SR7\_CC\_LUT0 0x5138

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_cc_lut003:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=0
27:24	RW	0	reg_sr7_cc_lut002:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=0
23:20	RW	0	reg_sr7_cc_lut001:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=0
19:16	RW	0	reg_sr7_cc_lut000:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=0
15:12	RW	8	reg_sr7_cc_lut103:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=1/8

Bit(s)	R/W	Default	Description
11:8	RW	4	reg_sr7_cc_lut102:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=1/8
7:4	RW	2	reg_sr7_cc_lut101:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=1/8
3:0	RW	1	reg_sr7_cc_lut100:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=1/8

Table 9-1968 SRSHARP0\_SR7\_CC\_LUT1 0x5139

Bit(s)	R/W	Default	Description
31:28			reserved
27:24	RW	12	reg_sr7_cc_lut202:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=2/8
23:20	RW	8	reg_sr7_cc_lut201:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=2/8
19:16	RW	4	reg_sr7_cc_lut200:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=2/8
15:12			reserved
11:8	RW	15	reg_sr7_cc_lut302:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=3/8
7:4	RW	10	reg_sr7_cc_lut301:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=3/8
3:0	RW	5	reg_sr7_cc_lut300:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=3/8

Table 9-1969 SRSHARP0\_SR7\_CC\_LUT2 0x513a

Bit(s)	R/W	Default	Description
31:24			reserved
23:20	RW	15	reg_sr7_cc_lut401:valid for y-lumadlt= 1/4, 1/2 ; of x-lumainput=4/8
19:16	RW	8	reg_sr7_cc_lut400:valid for y-lumadlt= 1/4, 1/2; of x-lumainput=4/8
15:12	RW	13	reg_sr7_cc_lut501:valid for y-lumadlt= 1/4, 1/2; of x-lumainput=5/8
11:8	RW	6	reg_sr7_cc_lut500:valid for y-lumadlt= 1/4, 1/2; of x-lumainput=5/8
7:4	RW	5	reg_sr7_cc_lut600:valid for y-lumadlt= 1/4 of x-lumainput=6/8
3:0	RW	4	reg_sr7_cc_lut700:valid for y-lumadlt= 1/4 of x-lumainput=7/8

Table 9-1970 SRSHARP0\_SR7\_CC\_LUT3 0x513b

Bit(s)	R/W	Default	Description
31:24			reserved
23:20	RW	4	reg_sr7_cc_lut710:valid for y-lumadlt= -1/4,; of x-lumainput=1/8
19:16	RW	5	reg_sr7_cc_lut610:valid for y-lumadlt= -1/4,; of x-lumainput=2/8
15:12	RW	13	reg_sr7_cc_lut511:valid for y-lumadlt= -1/4, -1/2; of x-lumainput=3/8
11:8	RW	6	reg_sr7_cc_lut510:valid for y-lumadlt= -1/4, -1/2; of x-lumainput=3/8
7:4	RW	15	reg_sr7_cc_lut411:valid for y-lumadlt= -1/4, -1/2 of x-lumainput=4/8
3:0	RW	8	reg_sr7_cc_lut410:valid for y-lumadlt= -1/4 , -1/2 of x-lumainput=4/8

**Table 9-1971 SRSHARP0\_SR7\_CC\_LUT4 0x513c**

Bit(s)	R/W	Default	Description
31:24			reserved
23:20	RW	15	reg_sr7_cc_lut312:valid for y-lumadlt= -1/4,-1/2, -3/4; of x-lumainput=5/8
19:16	RW	10	reg_sr7_cc_lut311:valid for y-lumadlt= -1/4,-1/2, -3/4; of x-lumainput=5/8
15:12	RW	5	reg_sr7_cc_lut310:valid for y-lumadlt= -1/4,-1/2, -3/4; of x-lumainput=5/8
11:8	RW	12	reg_sr7_cc_lut212:valid for y-lumadlt= -1/4, -1/2, -3/4;; of x-lumainput=6/8
7:4	RW	8	reg_sr7_cc_lut211:valid for y-lumadlt= -1/4, -1/2, -3/4; of x-lumainput=6/8
3:0	RW	4	reg_sr7_cc_lut210:valid for y-lumadlt= -1/4, -1/2, -3/4; of x-lumainput=6/8

**Table 9-1972 SRSHARP0\_SR7\_CC\_LUT5 0x513d**

Bit(s)	R/W	Default	Description
31:28	RW	8	reg_sr7_cc_lut113:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=7/8
27:24	RW	4	reg_sr7_cc_lut112:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=7/8
23:20	RW	2	reg_sr7_cc_lut111:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=7/8
19:16	RW	1	reg_sr7_cc_lut110:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=7/8
15:12	RW	0	reg_sr7_cc_lut013:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=8/8
11:8	RW	0	reg_sr7_cc_lut012:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=8/8
7:4	RW	0	reg_sr7_cc_lut011:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=8/8
3:0	RW	0	reg_sr7_cc_lut010:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=8/8

**Table 9-1973 SRSHARP0\_SR7\_GRAPHIC\_CTRL 0x513e**

Bit(s)	R/W	Default	Description
31:11			reserved
10	RW	1	reg_sr7_grph_en:enable graphic statistic
9	RW	1	reg_sr7_grph_hflt:horizontal filter, 0: [0 1 -1], 1: [-1 2 -1]
8	RW	1	reg_sr7_grph_vflt: vertical filter, 0: [0 1 -1], 1: [-1 2 -1]
7:0	RW	0	reg_sr7_grph_dif_cor:coring for dif while count for graphic

**Table 9-1974 SRSHARP0\_SR7\_GRAPHIC\_THD\_GAIN 0x513f**

Bit(s)	R/W	Default	Description
31:24	RW	2	reg_sr7_grph_fit_thd: flat threshold for dif while count for graphic.
23:16	RW	40	reg_sr7_grph_dtl_thd:detail threshold for dif while count for graphic
15:8	RW	32	reg_sr7_grph_hgain:horizontal gain for fast squart of hp
7:0	RW	32	reg_sr7_grph_vgain:vertical gain for fast squart of hp

**Table 9-1975 SRSHARP0\_SR7\_RO\_GRAPHIC\_FLT\_CNT 0x5140**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_sr7_grphflt_cnt:: flat count numbers for graphic

**Table 9-1976 SRSHARP0\_SR7\_RO\_GRAPHIC\_DTL\_CNT 0x5141+**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_sr7_grphdtl_cnt::detail count numbers for graphic

**Table 9-1977 SRSHARP0\_SR7\_CLR\_PRT\_PARAM 0x5142**

Bit(s)	R/W	Default	Description
31:18			reserved
17:16	RW	3	reg_sr7_clr_prct_inpsel:input UV selection for color protection, 0: org; 1: NRout; 2: CTlout; 3: (NR+Tlout)/2
15:8	RW	64	reg_sr7_clr_prct_dnlp_gain:gain to de-boost of dnlp_dlt base on color region, norm to 64 as 1.0, set to 0 as disable.
7:0	RW	64	reg_sr7_clr_prct_peak_gain:gain to de-boost of peak_dlt base on color region, norm to 64 as 1.0, set to 0 as disable

**Table 9-1978 SRSHARP0\_SR7\_CLR\_PRT\_LC\_GAIN 0x5143**

Bit(s)	R/W	Default	Description
31:24			reserved
23:16	RW	64	reg_sr7_clr_prct_lc_gain2:gain to de-boost of lc_dlt (y/u/v) base on color region, norm to 64 as 1.0, set to 0 as disable
15:8	RW	64	reg_sr7_clr_prct_lc_gain1:gain to de-boost of lc_dlt (y/u/v) base on color region, norm to 64 as 1.0, set to 0 as disable
7:0	RW	64	reg_sr7_clr_prct_lc_gain0:gain to de-boost of lc_dlt (y/u/v) base on color region, norm to 64 as 1.0, set to 0 as disable

**Table 9-1979 SRSHARP0\_SR7\_CLR\_PRT\_LUT0 0x5144**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut7:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut6:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut5:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut4:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut3:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut2:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut1:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut0:color protection lut, 16 is normalized to 1

**Table 9-1980 SRSHARP0\_SR7\_CLR\_PRT\_LUT1 0x5145**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut15:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut14:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut13:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut12:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut11:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut10:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut9:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut8:color protection lut, 16 is normalized to 1

**Table 9-1981 SRSHARP0\_SR7\_CLR\_PRT\_LUT2 0x5146**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut23:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut22:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut21:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut20:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut19:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut18:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut17:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut16:color protection lut, 16 is normalized to 1

**Table 9-1982 SRSHARP0\_SR7\_CLR\_PRT\_LUT3 0x5147**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut31:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut30:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut29:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut28:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut27:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut26:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut25:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut24:color protection lut, 16 is normalized to 1

**Table 9-1983 SRSHARP0\_SR7\_CLR\_PRT\_LUT4 0x5148**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut39:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut38:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut37:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut36:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut35:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut34:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut33:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut32:color protection lut, 16 is normalized to 1

**Table 9-1984 SRSHARP0\_SR7\_CLR\_PRT\_LUT5 0x5149**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut47:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut46:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut45:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut44:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut43:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut42:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut41:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut40:color protection lut, 16 is normalized to 1

**Table 9-1985 SRSHARP0\_SR7\_CLR\_PRT\_LUT6 0x514a**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut55:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut54:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut53:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut52:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut51:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut50:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut49:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut48:color protection lut, 16 is normalized to 1

**Table 9-1986 SRSHARP0\_SR7\_CLR\_PRT\_LUT7 0x514b**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut63:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut62:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut61:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut60:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut59:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut58:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut57:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut56:color protection lut, 16 is normalized to 1

**Table 9-1987 SRSHARP0\_SR7\_CLR\_PRT\_LUT8 0x514c**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut71:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut70:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut69:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut68:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut67:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut66:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut65:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut64:color protection lut, 16 is normalized to 1

**Table 9-1988 SRSHARP0\_SR7\_CLR\_PRT\_LUT9 0x514d**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut79:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut78:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut77:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut76:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut75:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut74:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut73:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut72:color protection lut, 16 is normalized to 1



**Table 9-1989 SRSHARP0\_SR7\_CLR\_PRT\_LUT10 0x514e**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut87:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut86:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut85:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut84:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut83:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut82:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut81:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut80:color protection lut, 16 is normalized to 1

**Table 9-1990 SRSHARP0\_SR7\_CLR\_PRT\_LUT11 0x514f**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut95:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut94:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut93:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut92:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut91:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut90:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut89:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut88:color protection lut, 16 is normalized to 1

**Table 9-1991 SRSHARP0\_SR7\_CLR\_PRT\_LUT12 0x5150**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut103:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut102:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut101:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut100:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut99:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut98:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut97:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut96:color protection lut, 16 is normalized to 1

**Table 9-1992 SRSHARP0\_SR7\_CLR\_PRT\_LUT13 0x5151**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut111:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut110:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut109:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut108:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut107:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut106:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut105:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut104:color protection lut, 16 is normalized to 1

**Table 9-1993 SRSHARP0\_SR7\_CLR\_PRT\_LUT14 0x5152**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut119:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut118:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut117:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut116:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut115:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut114:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut113:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut112:color protection lut, 16 is normalized to 1

**Table 9-1994 SRSHARP0\_SR7\_CLR\_PRT\_LUT15 0x5153**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut127:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut126:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut125:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut124:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut123:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut122:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut121:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut120:color protection lut, 16 is normalized to 1

**Table 9-1995 SRSHARP0\_SR7\_CLR\_PRT\_LUT16 0x5154**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut135:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut134:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut133:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut132:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut131:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut130:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut129:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut128:color protection lut, 16 is normalized to 1

**Table 9-1996 SRSHARP0\_SR7\_CLR\_PRT\_LUT17 0x5155**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut143:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut142:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut141:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut140:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut139:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut138:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut137:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut136:color protection lut, 16 is normalized to 1

**Table 9-1997 SRSHARP0\_SR7\_CLR\_PRT\_LUT18 0x5156**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut151:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut150:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut149:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut148:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut147:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut146:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut145:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut144:color protection lut, 16 is normalized to 1

**Table 9-1998 SRSHARP0\_SR7\_CLR\_PRT\_LUT19 0x5157**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut159:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut158:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut157:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut156:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut155:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut154:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut153:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut152:color protection lut, 16 is normalized to 1

**Table 9-1999 SRSHARP0\_SR7\_CLR\_PRT\_LUT20 0x5158**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut167:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut166:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut165:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut164:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut163:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut162:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut161:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut160:color protection lut, 16 is normalized to 1

**Table 9-2000 SRSHARP0\_SR7\_CLR\_PRT\_LUT21 0x5159**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut175:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut174:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut173:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut172:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut171:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut170:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut169:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut168:color protection lut, 16 is normalized to 1

**Table 9-2001 SRSHARP0\_SR7\_CLR\_PRT\_LUT22 0x515a**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut183:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut182:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut181:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut180:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut179:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut178:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut177:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut176:color protection lut, 16 is normalized to 1

**Table 9-2002 SRSHARP0\_SR7\_CLR\_PRT\_LUT23 0x515b**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut191:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut190:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut189:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut188:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut187:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut186:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut185:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut184:color protection lut, 16 is normalized to 1

**Table 9-2003 SRSHARP0\_SR7\_CLR\_PRT\_LUT24 0x515c**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut199:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut198:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut197:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut196:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut195:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut194:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut193:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut192:color protection lut, 16 is normalized to 1

**Table 9-2004 SRSHARP0\_SR7\_CLR\_PRT\_LUT25 0x515d**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut207:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut206:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut205:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut204:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut203:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut202:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut201:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut200:color protection lut, 16 is normalized to 1

**Table 9-2005 SRSHARP0\_SR7\_CLR\_PRT\_LUT26 0x515e**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut215:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut214:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut213:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut212:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut211:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut210:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut209:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut208:color protection lut, 16 is normalized to 1

**Table 9-2006 SRSHARP0\_SR7\_CLR\_PRT\_LUT27 0x515f**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut223:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut222:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut221:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut220:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut219:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut218:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut217:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut216:color protection lut, 16 is normalized to 1

**Table 9-2007 SRSHARP0\_SR7\_CLR\_PRT\_LUT28 0x5160**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut231:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut230:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut229:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut228:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut227:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut226:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut225:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut224:color protection lut, 16 is normalized to 1

**Table 9-2008 SRSHARP0\_SR7\_CLR\_PRT\_LUT29 0x5161**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut239:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut238:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut237:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut236:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut235:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut234:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut233:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut232:color protection lut, 16 is normalized to 1

**Table 9-2009 SRSHARP0\_SR7\_CLR\_PRT\_LUT30 0x5162**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut247:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut246:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut245:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut244:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut243:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut242:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut241:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut240:color protection lut, 16 is normalized to 1

**Table 9-2010 SRSHARP0\_SR7\_CLR\_PRT\_LUT31 0x5163**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut255:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut254:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut253:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut252:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut251:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut250:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut249:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut248:color protection lut, 16 is normalized to 1

**Table 9-2011 SRSHARP1\_SHARP\_HVSIZE 0x5200**

Bit(s)	R/W	Default	Description
28:16	R/W	0d1920	reg_pknr_hsize : . unsigned , default = 1920
12: 0	R/W	0d1080	reg_pknr_vsize : . unsigned , default = 1080

**Table 9-2012 SRSHARP1\_SHARP\_HVBLANK\_NUM 0x5201**

Bit(s)	R/W	Default	Description
23: 16	R/W	0d8	reg_deband_hblank : . unsigned , default = 8
15: 8	R/W	0d20	reg_pknr_hblank_num : . unsigned , default = 20
7: 0	R/W	0d60	reg_pknr_vblank_num : . unsigned , default = 60

**Table 9-2013 SRSHARP1\_NR\_GAUSSIAN\_MODE 0x5202**

Bit(s)	R/W	Default	Description
13□12	R/W	0d0	reg_nr_gau_cnorm : : Y adaptive coef norm, 0: 128, 1: 256, 2: 512, 3: 1024 .unsigned , default = 0
9□8	R/W	0d0	reg_nr_gau_cnorm : : C adaptive coef norm, 0: 128, 1: 256, 2: 512, 3: 1024 .unsigned , default = 0
5□4	R/W	0d1	reg_nr_gau_ymode : : 0 3x3 filter; 1: 5x5 filter, 2/3: adaptive coef .unsigned , default = 1
1□0	R/W	0d1	reg_nr_gau_cmode : : 0 3x3 filter; 1: 5x5 filter, 2/3: adaptive coef .unsigned , default = 1

**Table 9-2014 SRSHARP1\_PK\_CON\_2CIRHPGAIN\_TH\_RATE 0x5205**

Bit(s)	R/W	Default	Description
31:24	R/W	0d25	reg_pk_cirhpcon2gain0 : : threshold0 of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 25
23:16	R/W	0d60	reg_pk_cirhpcon2gain1 : : threshold1 of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 60



Bit(s)	R/W	Default	Description
15: 8	R/W	0d80	reg_pk_cirhpcon2gain5 :: rate0 (for hpcon<th0) of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 80
7: 0	R/W	0d20	reg_pk_cirhpcon2gain6 :: rate1 (for hpcon>th1) of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 20

Table 9-2015 SRSHARP1\_PK\_CON\_2CIRHPGAIN\_LIMIT 0x5206

Bit(s)	R/W	Default	Description
31:24	R/W	0d96	reg_pk_cirhpcon2gain2 :: level limit(for hpcon<th0) of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 96
23:16	R/W	0d96	reg_pk_cirhpcon2gain3 :: level limit(for th0<hpcon<th1) of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 96
15: 8	R/W	0d5	reg_pk_cirhpcon2gain4 :: level limit(for hpcon>th1) of curve to map hpcon to hpgain for circle hp filter (all 8 direction same). 0~255.. unsigned , default = 5

Table 9-2016 SRSHARP1\_PK\_CON\_2CIRBPGAIN\_TH\_RATE 0x5207

Bit(s)	R/W	Default	Description
31:24	R/W	0d20	reg_pk_cirbpcon2gain0 :: threshold0 of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 20
23:16	R/W	0d50	reg_pk_cirbpcon2gain1 :: threshold1 of curve to map bpcon to bpgain for circle bp filter (all 8 direction same).. unsigned , default = 50
15: 8	R/W	0d50	reg_pk_cirbpcon2gain5 :: rate0 (for bpcon<th0) of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 50
7: 0	R/W	0d25	reg_pk_cirbpcon2gain6 :: rate1 (for bpcon>th1) of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 25

Table 9-2017 SRSHARP1\_PK\_CON\_2CIRBPGAIN\_LIMIT 0x5208

Bit(s)	R/W	Default	Description
31:24	R/W	0d40	reg_pk_cirbpcon2gain2 :: level limit(for bpcon<th0) of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 40
23:16	R/W	0d40	reg_pk_cirbpcon2gain3 :: level limit(for th0<bpcon<th1) of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 40
15: 8	R/W	0d5	reg_pk_cirbpcon2gain4 :: level limit(for bpcon>th1) of curve to map bpcon to bpgain for circle bp filter (all 8 direction same). 0~255.. unsigned , default = 5

Table 9-2018 SRSHARP1\_PK\_CON\_2DRTHPGAIN\_TH\_RATE 0x5209

Bit(s)	R/W	Default	Description
31:24	R/W	0d25	reg_pk_drthpcon2gain0 :: threshold0 of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 25
23:16	R/W	0d60	reg_pk_drthpcon2gain1 :: threshold1 of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 60

Bit(s)	R/W	Default	Description
15: 8	R/W	0d80	reg_pk_drthpcon2gain5 : : rate0 (for hpcon<th0) of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 80
7: 0	R/W	0d20	reg_pk_drthpcon2gain6 : : rate1 (for hpcon>th1) of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 20

**Table 9-2019 SRSHARP1\_PK\_CON\_2DRTHPGAIN\_LIMIT 0x520a**

Bit(s)	R/W	Default	Description
31:24	R/W	0d90	reg_pk_drthpcon2gain2 : : level limit(for hpcon<th0) of curve to map hpcon to hpgain for directional hp filter (best direction).. unsigned , default = 90
23:16	R/W	0d96	reg_pk_drthpcon2gain3 : : level limit(for th0<hpcon<th1) of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 96
15: 8	R/W	0d5	reg_pk_drthpcon2gain4 : : level limit(for hpcon>th1) of curve to map hpcon to hpgain for directional hp filter (best direction). 0~255.. unsigned , default = 5

**Table 9-2020 SRSHARP1\_PK\_CON\_2DRTPGAIN\_TH\_RATE 0x520b**

Bit(s)	R/W	Default	Description
31:24	R/W	0d20	reg_pk_drtbpcon2gain0 : : threshold0 of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 20
23:16	R/W	0d50	reg_pk_drtbpcon2gain1 : : threshold1 of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 50
15: 8	R/W	0d50	reg_pk_drtbpcon2gain5 : : rate0 (for bpcon<th0) of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 50
7: 0	R/W	0d25	reg_pk_drtbpcon2gain6 : : rate1 (for bpcon>th1) of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 25

**Table 9-2021 SRSHARP1\_PK\_CON\_2DRTPGAIN\_LIMIT 0x520c**

Bit(s)	R/W	Default	Description
31:24	R/W	0d40	reg_pk_drtbpcon2gain2 : : level limit(for bpcon<th0) of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 40
23:16	R/W	0d40	reg_pk_drtbpcon2gain3 : : level limit(for th0<bpcon<th1) of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 40
15: 8	R/W	0d5	reg_pk_drtbpcon2gain4 : : level limit(for bpcon>th1) of curve to map bpcon to bpgain for directional bp filter (best direction). 0~255.. unsigned , default = 5

**Table 9-2022 SRSHARP1\_PK\_CIRFB\_LPF\_MODE 0x520d**

Bit(s)	R/W	Default	Description
29:28	R/W	0d1	reg_cirhp_horz_mode : : no horz filter on HP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 . unsigned , default = 1
25:24	R/W	0d1	reg_cirhp_vert_mode : : no vert filter on HP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 . unsigned , default = 1
21:20	R/W	0d1	reg_cirhp_diag_mode : : filter on HP; 1: [1 2 1]/4; . unsigned , default = 1

Bit(s)	R/W	Default	Description
13:12	R/W	0d1	reg_cirbp_horz_mode : : no horz filter on BP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 . unsigned , default = 1
9: 8	R/W	0d1	reg_cirbp_vert_mode : : no vert filter on BP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 . unsigned , default = 1
5: 4	R/W	0d1	reg_cirbp_diag_mode : : filter on BP; 1: [1 2 1]/4; . unsigned , default = 1

Table 9-2023 SRSHARP1\_PK\_DRTFB\_LPF\_MODE 0x520e

Bit(s)	R/W	Default	Description
29:28	R/W	0d1	reg_drthp_horz_mode : : no horz filter on HP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 2 . unsigned , default = 1
25:24	R/W	0d1	reg_drthp_vert_mode : : no vert filter on HP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 2 . unsigned , default = 1
21:20	R/W	0d1	reg_drthp_diag_mode : : filter on HP; 1: [1 2 1]/4; 1 . unsigned , default = 1
13:12	R/W	0d1	reg_drtbp_horz_mode : : no horz filter on BP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 2 . unsigned , default = 1
9: 8	R/W	0d1	reg_drtbp_vert_mode : : no vert filter on BP; 1: [1 2 1]/4; 2/3: [1 2 2 2 1]/8 2 . unsigned , default = 1
5: 4	R/W	0d1	reg_drtbp_diag_mode : : filter on BP; 1: [1 2 1]/4; 1 . unsigned , default = 1

Table 9-2024 SRSHARP1\_PK\_CIRFB\_HP\_CORING 0x520f

Bit(s)	R/W	Default	Description
21:16	R/W	0d4	reg_cirhp_horz_core : : coring of HP for Horz . unsigned , default = 4
13: 8	R/W	0d4	reg_cirhp_vert_core : : coring of HP for Vert . unsigned , default = 4
5: 0	R/W	0d4	reg_cirhp_diag_core : : coring of HP for Diag . unsigned , default = 4

Table 9-2025 SRSHARP1\_PK\_CIRFB\_BP\_CORING 0x5210

Bit(s)	R/W	Default	Description
21:16	R/W	0d4	reg_cirbp_horz_core : : coring of HP for Horz . unsigned , default = 4
13: 8	R/W	0d4	reg_cirbp_vert_core : : coring of HP for Vert . unsigned , default = 4
5: 0	R/W	0d4	reg_cirbp_diag_core : : coring of HP for Diag . unsigned , default = 4

Table 9-2026 SRSHARP1\_PK\_DRTFB\_HP\_CORING 0x5211

Bit(s)	R/W	Default	Description
21:16	R/W	0d4	reg_drthp_horz_core : : coring of HP for Horz . unsigned , default = 4
13: 8	R/W	0d4	reg_drthp_vert_core : : coring of HP for Vert . unsigned , default = 4
5: 0	R/W	0d4	reg_drthp_diag_core : : coring of HP for Diag . unsigned , default = 4

**Table 9-2027 SRSHARP1\_PK\_DRTFB\_BP\_CORING 0x5212**

Bit(s)	R/W	Default	Description
21:16	R/W	0d4	reg_drtbp_horz_core :: coring of HP for Horz . unsigned , default = 4
13: 8	R/W	0d4	reg_drtbp_vert_core :: coring of HP for Vert . unsigned , default = 4
5: 0	R/W	0d4	reg_drtbp_diag_core :: coring of HP for Diag . unsigned , default = 4

**Table 9-2028 SRSHARP1\_PK\_CIRFB\_BLEND\_GAIN 0x5213**

Bit(s)	R/W	Default	Description
31:28	R/W	0d8	reg_hp_cir_hgain :: normalized 8 as '1' . unsigned , default = 8
27:24	R/W	0d8	reg_hp_cir_vgain :: normalized 8 as '1' . unsigned , default = 8
23:20	R/W	0d8	reg_hp_cir_dgain :: normalized 8 as '1' . unsigned , default = 8
15:12	R/W	0d8	reg_bp_cir_hgain :: normalized 8 as '1' . unsigned , default = 8
11: 8	R/W	0d8	reg_bp_cir_vgain :: normalized 8 as '1' . unsigned , default = 8
7: 4	R/W	0d8	reg_bp_cir_dgain :: normalized 8 as '1' . unsigned , default = 8

**Table 9-2029 SRSHARP1\_NR\_ALPY\_SSD\_GAIN\_OFST 0x5214**

Bit(s)	R/W	Default	Description
15: 8	R/W	0d16	reg_nr_alp0_ssd_gain :: gain to max ssd normalized 16 as '1' . unsigned , default = 16
5: 0	R/W	0x0	reg_nr_alp0_ssd_ofst :: offset to ssd before dividing to min_err . signed , default = -2

**Table 9-2030 SRSHARP1\_NR\_ALP0Y\_ERR2CURV\_TH\_RATE 0x5215**

Bit(s)	R/W	Default	Description
31:24	R/W	0d10	reg_nr_alp0_minerr_ypar0 :: threshold0 of curve to map mierr to alp0 for luma channel, this will be set value of flat region mierr that no need blur. 0~255.. unsigned , default = 10
23:16	R/W	0d25	reg_nr_alp0_minerr_ypar1 :: threshold1 of curve to map mierr to alp0 for luma channel, this will be set value of texture region mierr that can not blur.. unsigned , default = 25
15: 8	R/W	0d80	reg_nr_alp0_minerr_ypar5 :: rate0 (for mierr<th0) of curve to map mierr to alp0 for luma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 80
7: 0	R/W	0d64	reg_nr_alp0_minerr_ypar6 :: rate1 (for mierr>th1) of curve to map mierr to alp0 for luma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 64

**Table 9-2031 SRSHARP1\_NR\_ALP0Y\_ERR2CURV\_LIMIT 0x5216**

Bit(s)	R/W	Default	Description
31:24	R/W	0d63	reg_nr_alp0_minerr_ypar2 : : level limit(for mierr<th0) of curve to map mierr to alp0 for luma channel, this will be set to alp0 that we can do for flat region. 0~255.. unsigned , default = 63
23:16	R/W	0d0	reg_nr_alp0_minerr_ypar3 : : level limit(for th0<mierr<th1) of curve to map mierr to alp0 for luma channel, this will be set to alp0 that we can do for misc region. 0~255.. unsigned , default = 0
15: 8	R/W	0d63	reg_nr_alp0_minerr_ypar4 : : level limit(for mierr>th1) of curve to map mierr to alp0 for luma channel, this will be set to alp0 that we can do for texture region. 0~255.. unsigned , default = 63

**Table 9-2032 SRSHARP1\_NR\_ALP0C\_ERR2CURV\_TH\_RATE 0x5217**

Bit(s)	R/W	Default	Description
31:24	R/W	0d10	reg_nr_alp0_minerr_cpar0 : : threshold0 of curve to map mierr to alp0 for chroma channel, this will be set value of flat region mierr that no need blur.. unsigned , default = 10
23:16	R/W	0d25	reg_nr_alp0_minerr_cpar1 : : threshold1 of curve to map mierr to alp0 for chroma channel, this will be set value of texture region mierr that can not blur.. unsigned , default = 25
15: 8	R/W	0d80	reg_nr_alp0_minerr_cpar5 : : rate0 (for mierr<th0) of curve to map mierr to alp0 for chroma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 80
7: 0	R/W	0d64	reg_nr_alp0_minerr_cpar6 : : rate1 (for mierr>th1) of curve to map mierr to alp0 for chroma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 64

**Table 9-2033 SRSHARP1\_NR\_ALP0C\_ERR2CURV\_LIMIT 0x5218**

Bit(s)	R/W	Default	Description
31:24	R/W	0d63	reg_nr_alp0_minerr_cpar2 : : level limit(for mierr<th0) of curve to map mierr to alp0 for chroma channel, this will be set to alp0 that we can do for flat region. 0~255.. unsigned , default = 63
23:16	R/W	0d0	reg_nr_alp0_minerr_cpar3 : : level limit(for th0<mierr<th1) of curve to map mierr to alp0 for chroma channel, this will be set to alp0 that we can do for misc region. 0~255.. unsigned , default = 0
15: 8	R/W	0d63	reg_nr_alp0_minerr_cpar4 : : level limit(for mierr>th1) of curve to map mierr to alp0 for chroma channel, this will be set to alp0 that we can do for texture region. 0~255.. unsigned , default = 63

**Table 9-2034 SRSHARP1\_NR\_ALP0\_MIN\_MAX 0x5219**

Bit(s)	R/W	Default	Description
29:24	R/W	0d2	reg_nr_alp0_ymin : : normalized to 64 as '1' . unsigned , default = 2
21:16	R/W	0d63	reg_nr_alp0_ymax : : normalized to 64 as '1' . unsigned , default = 63
13: 8	R/W	0d2	reg_nr_alp0_cmin : : normalized to 64 as '1' . unsigned , default = 2
5: 0	R/W	0d63	reg_nr_alp0_cmax : : normalized to 64 as '1' . unsigned , default = 63

**Table 9-2035 SRSHARP1\_NR\_ALP1\_MIERR\_CORING 0x521a**

Bit(s)	R/W	Default	Description
16	R/W	0d0	reg_nr_alp1_maxerr_mode : : 0 max err; 1: xerr . unsigned , default = 0
13: 8	R/W	0d0	reg_nr_alp1_core_rate : : normalized 64 as "1" . unsigned , default = 0
5: 0	R/W	0d3	reg_nr_alp1_core_ofst : : normalized 64 as "1" . signed , default = 3

**Table 9-2036 SRSHARP1\_NR\_ALP1\_ERR2CURV\_TH\_RATE 0x521b**

Bit(s)	R/W	Default	Description
31:24	R/W	0d0	reg_nr_alp1_minerr_par0 : : threshold0 of curve to map mierr to alp1 for luma/ chroma channel, this will be set value of flat region mierr that no need directional NR. 0~255.. unsigned , default = 0
23:16	R/W	0d24	reg_nr_alp1_minerr_par1 : : threshold1 of curve to map mierr to alp1 for luma/ chroma channel, this will be set value of texture region mierr that can not do directional NR. 0~255.. unsigned , default = 24
15: 8	R/W	0d0	reg_nr_alp1_minerr_par5 : : rate0 (for mierr<th0) of curve to map mierr to alp1 for luma/chroma channel. the larger of the value, the deep of the slope.. unsigned , default = 0
7: 0	R/W	0d20	reg_nr_alp1_minerr_par6 : : rate1 (for mierr>th1) of curve to map mierr to alp1 for luma/chroma channel. the larger of the value, the deep of the slope. 0~255. unsigned , default = 20

**Table 9-2037 SRSHARP1\_NR\_ALP1\_ERR2CURV\_LIMIT 0x521c**

Bit(s)	R/W	Default	Description
31:24	R/W	0d0	reg_nr_alp1_minerr_par2 : : level limit(for mierr<th0) of curve to map mierr to alp1 for luma/chroma channel, this will be set to alp1 that we can do for flat region. 0~255.. unsigned , default = 0
23:16	R/W	0d16	reg_nr_alp1_minerr_par3 : : level limit(for th0<mierr<th1) of curve to map mierr to alp1 for luma/chroma channel, this will be set to alp1 that we can do for misc region. 0~255.. unsigned , default = 16
15: 8	R/W	0d63	reg_nr_alp1_minerr_par4 : : level limit(for mierr>th1) of curve to map mierr to alp1 for luma/chroma channel, this will be set to alp1 that we can do for texture region. 0~255.255 before. unsigned , default = 63

**Table 9-2038 SRSHARP1\_NR\_ALP1\_MIN\_MAX 0x521d**

Bit(s)	R/W	Default	Description
29:24	R/W	0d0	reg_nr_alp1_ymin : : normalized to 64 as '1' . unsigned , default = 0
21:16	R/W	0d63	reg_nr_alp1_ymax : : normalized to 64 as '1' . unsigned , default = 63
13: 8	R/W	0d0	reg_nr_alp1_cmin : : normalized to 64 as '1' . unsigned , default = 0
5: 0	R/W	0d63	reg_nr_alp1_cmax : : normalized to 64 as '1' . unsigned , default = 63

**Table 9-2039 SRSHARP1\_PK\_ALP2\_MIERR\_CORING 0x521e**

Bit(s)	R/W	Default	Description
16	R/W	0d1	reg_pk_alp2_maxerr_mode : : 0 max err; 1: xerr . unsigned , default = 1
13: 8	R/W	0d13	reg_pk_alp2_core_rate : : normalized 64 as "1" . unsigned , default = 13
5: 0	R/W	0d1	reg_pk_alp2_core_ofst : : normalized 64 as "1" . signed , default = 1

**Table 9-2040 SRSHARP1\_PK\_ALP2\_ERR2CURV\_TH\_RATE 0x521f**

Bit(s)	R/W	Default	Description
31:24	R/W	0d0	reg_pk_alp2_minerr_par0 : : threshold0 of curve to map mierr to alp2 for luma channel, this will be set value of flat region mierr that no need peaking.. unsigned , default = 0
23:16	R/W	0d24	reg_pk_alp2_minerr_par1 : : threshold1 of curve to map mierr to alp2 for luma channel, this will be set value of texture region mierr that can not do peaking. 0~255.. unsigned , default = 24
15: 8	R/W	0d0	reg_pk_alp2_minerr_par5 : : rate0 (for mierr<th0) of curve to map mierr to alp2 for luma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 0
7: 0	R/W	0d20	reg_pk_alp2_minerr_par6 : : rate1 (for mierr>th1) of curve to map mierr to alp2 for luma channel. the larger of the value, the deep of the slope. 0~255.. unsigned , default = 20

**Table 9-2041 SRSHARP1\_PK\_ALP2\_ERR2CURV\_LIMIT 0x5220**

Bit(s)	R/W	Default	Description
31:24	R/W	0d0	reg_pk_alp2_minerr_par2 : : level limit(for mierr<th0) of curve to map mierr to alp2 for luma channel, this will be set to alp2 that we can do for flat region. 0~255.. unsigned , default = 0
23:16	R/W	0d16	reg_pk_alp2_minerr_par3 : : level limit(for th0<mierr<th1) of curve to map mierr to alp2 for luma channel, this will be set to alp2 that we can do for misc region. 0~255.. unsigned , default = 16
15: 8	R/W	0d63	reg_pk_alp2_minerr_par4 : : level limit(for mierr>th1) of curve to map mierr to alp2 for luma channel, this will be set to alp2 that we can do for texture region. 0~255. default = 63;. unsigned , default = 255

**Table 9-2042 SRSHARP1\_PK\_ALP2\_MIN\_MAX 0x5221**

Bit(s)	R/W	Default	Description
13: 8	R/W	0d0	reg_pk_alp2_min : : normalized to 64 as '1' . unsigned , default = 0
5: 0	R/W	0d63	reg_pk_alp2_max : : normalized to 64 as '1' . unsigned , default = 63

**Table 9-2043 SRSHARP1\_PK\_FINALGAIN\_HP\_BP 0x5222**

Bit(s)	R/W	Default	Description
17: 16	R/W	0d0	reg_final_gain_rs : : right shift bits for the gain normalization, 0 normal to 32 as 1; 1 normal to 64 as 1; -2 normal to 8 as 1; -1 normal to 16 as 1. signed , default = 0
15: 8	R/W	0d40	reg_hp_final_gain : : gain to highpass boost result (including directional/circle blending), normalized 32 as '1', 0~255. 1.25 * 32. unsigned , default = 40
7: 0	R/W	0d30	reg_bp_final_gain : : gain to bandpass boost result (including directional/circle blending), normalized 32 as '1', 0~255. 1.25 * 32. unsigned , default = 30

**Table 9-2044 SRSHARP1\_PK\_OS\_HORZ\_CORE\_GAIN 0x5223**

Bit(s)	R/W	Default	Description
31:24	R/W	0d8	reg_pk_os_hsidecore : : side coring (not to current pixel) to adaptive overshoot margin in horizontal direction. the larger of this value, the less overshoot admitted 0~255;. unsigned , default = 8
23:16	R/W	0d20	reg_pk_os_hsidegain : : side gain (not to current pixel) to adaptive overshoot margin in horizontal direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20
15: 8	R/W	0d2	reg_pk_os_hmidcore : : mid coring (to current pixel) to adaptive overshoot margin in horizontal direction. the larger of this value, the less overshoot admitted 0~255;. unsigned , default = 2
7: 0	R/W	0d20	reg_pk_os_hmidgain : : mid gain (to current pixel) to adaptive overshoot margin in horizontal direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20

**Table 9-2045 SRSHARP1\_PK\_OS\_VERT\_CORE\_GAIN 0x5224**

Bit(s)	R/W	Default	Description
31:24	R/W	0d8	reg_pk_os_vsidecore : : side coring (not to current pixel) to adaptive overshoot margin in vertical direction. the larger of this value, the less overshoot admitted 0~255;. unsigned , default = 8
23:16	R/W	0d20	reg_pk_os_vsidegain : : side gain (not to current pixel) to adaptive overshoot margin in vertical direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20
15: 8	R/W	0d2	reg_pk_os_vmidcore : : mid coring (to current pixel) to adaptive overshoot margin in vertical direction. the larger of this value, the less overshoot admitted 0~255;. unsigned , default = 2
7: 0	R/W	0d20	reg_pk_os_vmidgain : : mid gain (to current pixel) to adaptive overshoot margin in vertical direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20

**Table 9-2046 SRSHARP1\_PK\_OS\_ADPT\_MISC 0x5225**

Bit(s)	R/W	Default	Description
31:24	R/W	0d40	reg_pk_os_minerr_core : : coring to minerr for adaptive overshoot margin. the larger of this value, the less overshoot admitted 0~255;. unsigned , default = 40
23:16	R/W	0d6	reg_pk_os_minerr_gain : : gain to minerr based adaptive overshoot margin. normalized to 64 as '1'. 0~255;. unsigned , default = 6



Bit(s)	R/W	Default	Description
15: 8	R/W	0d200	reg_pk_os_adpt_max : : maximum limit adaptive overshoot margin (4x). 0~255; . unsigned , default = 200
7: 0	R/W	0d20	reg_pk_os_adpt_min : : minimum limit adaptive overshoot margin (1x). 0~255; . unsigned , default = 20

**Table 9-2047 SRSHARP1\_PK\_OS\_STATIC 0x5226**

Bit(s)	R/W	Default	Description
29:28	R/W	0d2	reg_pk_osh_mode : : 0~3: (2x+1) window in H direction . unsigned , default = 2
25:24	R/W	0d2	reg_pk_osv_mode : : 0~3: (2x+1) window in V direction . unsigned , default = 2
21:12	R/W	0d200	reg_pk_os_down : : static negative overshoot margin. 0~1023; . unsigned , default = 200
9: 0	R/W	0d200	reg_pk_os_up : : static positive overshoot margin. 0~1023; . unsigned , default = 200

**Table 9-2048 SRSHARP1\_PK\_NR\_ENABLE 0x5227**

Bit(s)	R/W	Default	Description
3: 2	R/W	0d0	reg_3d_mode : , 0: no 3D; 1: L/R; 2: T/B; 3: horizontal interleaved, dft = 0 // . unsigned , default = 0
1	R/W	0d1	reg_pk_en : . unsigned , default = 1
0	R/W	0d1	reg_nr_en : . unsigned , default = 1

**Table 9-2049 SRSHARP1\_PK\_DRT\_SAD\_MISC 0x5228**

Bit(s)	R/W	Default	Description
31:24	R/W	0d24	reg_pk_sad_ver_gain : : gain to sad[4], 16 normalized to "1"; . unsigned , default = 24
23:16	R/W	0d24	reg_pk_sad_hor_gain : : gain to sad[0], 16 normalized to "1"; . unsigned , default = 24
10: 9	R/W	0d0	reg_pk_bias_diag : : bias towards diag . unsigned , default = 0
4: 0	R/W	0d24	reg_pk_drt_force : : force direction of drt peaking filter, h2b: 0:hp drt force, 1: bp drt force; 2: bp+hp drt force, 3: no force;. unsigned , default = 24

**Table 9-2050 SRSHARP1\_NR\_TI\_DNLP\_BLEND 0x5229**

Bit(s)	R/W	Default	Description
10: 8	R/W	0d4	reg_dnlp_input_mode : : dnlp input options. 0: org_y; 1: gau_y; 2: gauadp_y; 3: edgadp_y; 4: nr_y; 5: lti_y; 6: pk_y (before os); 7: pk_y (after os). unsigned , default = 4
3: 2	R/W	0d1	reg_nr_cti_blend_mode : : blend mode of nr and lti result: 0: nr; 1:cti; 2: (nr+cti)/2; 3:cti + dlt_nr . unsigned , default = 1
1: 0	R/W	0d1	reg_nr_lti_blend_mode : : blend mode of nr and lti result: 0: nr; 1:lti; 2: (nr+lti)/2; 3:lti + dlt_nr . unsigned , default = 1

**Table 9-2051 SRSHARP1\_TI\_DIR\_CORE\_ALPHA 0x522a**

Bit(s)	R/W	Default	Description
29:24	R/W	0d10	reg_adp_lti_dir_alp_core_ofst : : ofst to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err*64; dft=10. unsigned , default = 10
19:16	R/W	0d0	reg_adp_lti_dir_alp_core_rate : : ofset to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err*64; dft=0/32. unsigned , default = 0
13: 8	R/W	0d0	reg_adp_lti_dir_alpmin : : min value of alpha, alpha = (min_err+x +ofst)/max_err*64; dft=10 . unsigned , default = 0
5: 0	R/W	0d63	reg_adp_lti_dir_alpmax : : max value of alpha, alpha = (min_err+x +ofst)/max_err*64; dft=63 . unsigned , default = 63

**Table 9-2052 SRSHARP1\_CTI\_DIR\_ALPHA 0x522b**

Bit(s)	R/W	Default	Description
29:24	R/W	0d5	reg_adp_cti_dir_alp_core_ofst : : ofst to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err*64; dft=10. unsigned , default = 5
19:16	R/W	0d0	reg_adp_cti_dir_alp_core_rate : : ofset to min_err, alpha = (min_err - (max_err - min_err)*rate + ofst)/max_err*64; dft=0/32. unsigned , default = 0
13: 8	R/W	0d0	reg_adp_cti_dir_alpmin : : min value of alpha, alpha = (min_err +x+ofst)/max_err*64; dft=10 . unsigned , default = 0
5: 0	R/W	0d63	reg_adp_cti_dir_alpmax : : max value of alpha, alpha = (min_err +x+ofst)/max_err*64; dft=63 . unsigned , default = 63

**Table 9-2053 SRSHARP1\_LTI\_CTI\_DF\_GAIN 0x522c**

Bit(s)	R/W	Default	Description
29:24	R/W	0d16	reg_adp_lti_hdf_gain : : 8 normalized to "1"; default = 16 . unsigned , default = 16
21:16	R/W	0d12	reg_adp_lti_vdf_gain : : 8 normalized to "1"; default = 12 . unsigned , default = 12
13: 8	R/W	0d16	reg_adp_cti_hdf_gain : : 8 normalized to "1"; default = 16 . unsigned , default = 16
5: 0	R/W	0d12	reg_adp_cti_vdf_gain : : 8 normalized to "1"; default = 12 . unsigned , default = 12

**Table 9-2054 SRSHARP1\_LTI\_CTI\_DIR\_AC\_DBG 0x522d**

Bit(s)	R/W	Default	Description
30	R/W	0d1	reg_adp_lti_dir_lpf : : 0: no lpf; 1: [1 2 2 2 1]/8 lpf . unsigned , default = 1
28	R/W	0d0	reg_adp_lti_dir_difmode : : 0: y_dif; 1: y_dif + (u_dif+v_dif)/2; . unsigned , default = 0
26	R/W	0d1	reg_adp_cti_dir_lpf : : 0: no lpf; 1: [1 2 2 2 1]/8 lpf dft=1 . unsigned , default = 1
25:24	R/W	0d0	reg_adp_cti_dir_difmode : : 0: (u_dif+v_dif); 1: y_dif/2 + (u_dif+v_dif)*3/4; 2: y_dif + (u_dif+v_dif)/2; 3: y_dif*2 (not recommended). unsigned , default = 0
23:22	R/W	0d3	reg_adp_hvlti_dcblend_mode : : 0: hlti_dc; 1:vlti_dc; 2: avg 3; blend on alpha . unsigned , default = 3
21:20	R/W	0d3	reg_adp_hvcti_dcblend_mode : : 0: hcti_dc; 1:vcti_dc; 2: avg 3; blend on alpha . unsigned , default = 3

Bit(s)	R/W	Default	Description
19:18	R/W	0d3	reg_adp_hvlti_aclblend_mode : : hlti_ac; 1:vlti_ac; 2: add 3;:adaptive to alpha . unsigned , default = 3
17:16	R/W	0d3	reg_adp_hvcti_aclblend_mode : : hcti_ac; 1:vcti_ac; 2: add 3;: adaptive to alpha . unsigned , default = 3
14:12	R/W	0d0	reg_adp_hlti_debug : , for hlti debug, default = 0 . unsigned , default = 0
10: 8	R/W	0d0	reg_adp_vlti_debug : , for vlti debug, default = 0 . unsigned , default = 0
6: 4	R/W	0d0	reg_adp_hcti_debug : , for hcti debug, default = 0 . unsigned , default = 0
2: 0	R/W	0d0	reg_adp_vcti_debug : , for vcti debug, default = 0 . unsigned , default = 0

Table 9-2055 SRSHARP1\_HCTI\_FLT\_CLP\_DC 0x522e

Bit(s)	R/W	Default	Description
28	R/W	0d1	reg_adp_hcti_en : , 0: no cti, 1: new cti, default = 1 . unsigned , default = 1
27:26	R/W	0d3	reg_adp_hcti_vdn_fit : , 0: no lpf; 1:[0,2,4,2,0], 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
25:24	R/W	0d2	reg_adp_hcti_hdn_fit : , 0: no lpf; 1:[0, 0, 0, 4, 8, 4, 0, 0, 0], 2:[0, 0, 2, 4, 4, 4, 2, 0, 0], 3: [1, 2, 2, 2, 2, 2, 2, 2, 1], default = 2. unsigned , default = 2
23:22	R/W	0d3	reg_adp_hcti_ddn_fit : , 0: no lpf; 1:[0,2,4,2,0], 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
21:20	R/W	0d2	reg_adp_hcti_lpf0_fit : , 0:no filter; 1:sigma=0.75, 2: sigma = 1.0, 3: sigma = 1.5, default = 2 . unsigned , default = 2
19:18	R/W	0d2	reg_adp_hcti_lpf1_fit : , 0:no filter; 1:sigma= 2.0, 2: sigma = 3.0, 3: sigma = 4.0, default = 2 . unsigned , default = 2
17:16	R/W	0d2	reg_adp_hcti_lpf2_fit : , 0:no filter; 1:sigma=5.0, 2: sigma = 9.0, 3: sigma = 13.0, default = 2 . unsigned , default = 2
15:12	R/W	0d7	reg_adp_hcti_hard_clp_win : , window size, 0~8, default = 7 . unsigned , default = 7
11: 8	R/W	0d3	reg_adp_hcti_hard_win_min : , window size, 0~8, default = 3 . unsigned , default = 3
4	R/W	0d1	reg_adp_hcti_clp_mode : , 0: hard clip, 1: adaptive clip, default = 1 . unsigned , default = 1
2: 0	R/W	0d0	reg_adp_hcti_dc_mode : , 0:dn, 1:lpf0, 2:lpf1, 3:lpf2, 4: lpf3: 5: vdn result; 6/7: org, default = 0 . unsigned , default = 0

Table 9-2056 SRSHARP1\_HCTI\_BST\_GAIN 0x522f

Bit(s)	R/W	Default	Description
31:24	R/W	0d80	reg_adp_hcti_bst_gain0 : : gain of the bandpass 0 (lpf1-lpf2)- LBP, default = 80 . unsigned , default = 80
23:16	R/W	0d96	reg_adp_hcti_bst_gain1 : : gain of the bandpass 1 (lpf0-lpf1)- BP, default = 96 . unsigned , default = 96
15: 8	R/W	0d64	reg_adp_hcti_bst_gain2 : : gain of the bandpass 2 (hdn-lpf0)- HP, default = 64 . unsigned , default = 64
7: 0	R/W	0d16	reg_adp_hcti_bst_gain3 : : gain of the unsharp band (yuvin-hdn) - US, default = 16 . unsigned , default = 16

**Table 9-2057 SRSHARP1\_HCTI\_BST\_CORE 0x5230**

Bit(s)	R/W	Default	Description
31:24	R/W	0d0	reg_adp_hcti_bst_core0 : : core of the bandpass 0 (lpf1-lpf2)- LBP, default = 0 . unsigned , default = 0
23:16	R/W	0d0	reg_adp_hcti_bst_core1 : : core of the bandpass 1 (lpf0-lpf1)- BP, default = 0 . unsigned , default = 0
15: 8	R/W	0d0	reg_adp_hcti_bst_core2 : : core of the bandpass 2 (hdn-lpf0)- HP, default = 0 . unsigned , default = 0
7: 0	R/W	0d0	reg_adp_hcti_bst_core3 : : core of the unsharp band (yuvin-hdn) - US, default = 0 . unsigned , default = 0

**Table 9-2058 SRSHARP1\_HCTI\_CON\_2\_GAIN\_0 0x5231**

Bit(s)	R/W	Default	Description
31:29	R/W	0d2	reg_adp_hcti_con_mode : : con mode 0:[0, 0,-1, 1, 0, 0, 0]+[0, 0, 0, 1,-1, 0, 0], 1:[0, 0,-1, 0, 1, 0, 0], 2:[0,-1, 0, 0, 0, 1, 0], 3:[-1, 0, 0, 0, 0, 0, 1], 4: .... default = 2 . unsigned , default = 2
28:26	R/W	0d3	reg_adp_hcti_dx_mode : : dx mode 0: [-1 1 0]; 1~7: [-1 (2x+1)"0" 1], default = 3 . unsigned , default = 3
25:24	R/W	0d1	reg_adp_hcti_con_lpf : : lpf mode of the con: 0: [1 2 1]/4; 1:[1 2 2 2 1]/8, default = 1 . unsigned , default = 1
23:16	R/W	0d25	reg_adp_hcti_con_2_gain0 : , default = 25 . unsigned , default = 25
15: 8	R/W	0d60	reg_adp_hcti_con_2_gain1 : , default = 60 . unsigned , default = 60
7: 0	R/W	0d0	reg_adp_hcti_con_2_gain2 : 0;, default = 0 . unsigned , default = 0

**Table 9-2059 SRSHARP1\_HCTI\_CON\_2\_GAIN\_1 0x5232**

Bit(s)	R/W	Default	Description
31:24	R/W	0d96	reg_adp_hcti_con_2_gain3 : 96;, default = 96 . unsigned , default = 96
23:16	R/W	0d5	reg_adp_hcti_con_2_gain4 : 5;, default = 5 . unsigned , default = 5
15: 8	R/W	0d80	reg_adp_hcti_con_2_gain5 : 80;, default = 80 . unsigned , default = 80
7: 0	R/W	0d20	reg_adp_hcti_con_2_gain6 : 20;, default = 20 . unsigned , default = 20

**Table 9-2060 SRSHARP1\_HCTI\_OS\_MARGIN 0x5233**

Bit(s)	R/W	Default	Description
7: 0	R/W	0d0	reg_adp_hcti_os_margin : : margin for hcti overshoot, default = 0 . unsigned , default = 0

**Table 9-2061 SRSHARP1\_HLTI\_FLT\_CLP\_DC 0x5234**

Bit(s)	R/W	Default	Description
28	R/W	0d1	reg_adp_hlти_en : , 0: no cti, 1: new cti, default = 1 . unsigned , default = 1
27:26	R/W	0d2	reg_adp_hlти_vdn_fit : , 0: no lpf; 1:[0,2,4,2,0], 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 2 . unsigned , default = 2
25:24	R/W	0d2	reg_adp_hlти_hdn_fit : , 0: no lpf; 1:[0, 0, 0, 4, 8, 4, 0, 0, 0], 2:[0, 0, 2, 4, 4, 4, 2, 0, 0], 3: [1, 2, 2, 2, 2, 2, 2, 1], default = 2. unsigned , default = 2
23:22	R/W	0d2	reg_adp_hlти_ddn_fit : , 0: no lpf; 1:[0,2,4,2,0], 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 2 . unsigned , default = 2
21:20	R/W	0d2	reg_adp_hlти_lpf0_fit : , 0:no filter; 1:sigma=0.75, 2: sigma = 1.0, 3: sigma = 1.5, default = 2 . unsigned , default = 2
19:18	R/W	0d2	reg_adp_hlти_lpf1_fit : , 0:no filter; 1:sigma= 2.0, 2: sigma = 3.0, 3: sigma = 4.0, default = 2 . unsigned , default = 2
17:16	R/W	0d2	reg_adp_hlти_lpf2_fit : , 0:no filter; 1:sigma=5.0, 2: sigma = 9.0, 3: sigma = 13.0, default = 2 . unsigned , default = 2
15:12	R/W	0d2	reg_adp_hlти_hard_clip_win : , window size, 0~8, default = 2 . unsigned , default = 2
11: 8	R/W	0d1	reg_adp_hlти_hard_win_min : , window size, 0~8, default = 1 . unsigned , default = 1
4	R/W	0d0	reg_adp_hlти_clip_mode : , 0: hard clip, 1: adaptive clip, default = 0 . unsigned , default = 0
2: 0	R/W	0d4	reg_adp_hlти_dc_mode : , 0:dn, 1:lpf0, 2:lpf1, 3:lpf2, 4: lpf3: 5: vdn result; 6/7:org, default = 4 . unsigned , default = 4

**Table 9-2062 SRSHARP1\_HLTI\_BST\_GAIN 0x5235**

Bit(s)	R/W	Default	Description
31:24	R/W	0d40	reg_adp_hlти_bst_gain0 : : gain of the bandpass 0 (lpf1-lpf2)- LBP, default = 40 . unsigned , default = 40
23:16	R/W	0d48	reg_adp_hlти_bst_gain1 : : gain of the bandpass 1 (lpf0-lpf1)- BP, default = 48 . unsigned , default = 48
15: 8	R/W	0d32	reg_adp_hlти_bst_gain2 : : gain of the bandpass 2 (hdn-lpf0)- HP, default = 32 . unsigned , default = 32
7: 0	R/W	0d16	reg_adp_hlти_bst_gain3 : : gain of the unsharp band (yuvin-hdn) - US, default = 16 . unsigned , default = 16

**Table 9-2063 SRSHARP1\_HLTI\_BST\_CORE 0x5236**

Bit(s)	R/W	Default	Description
31:24	R/W	0d5	reg_adp_hlти_bst_core0 : : core of the bandpass 0 (lpf1-lpf2)- LBP, default = 5 . unsigned , default = 5
23:16	R/W	0d5	reg_adp_hlти_bst_core1 : : core of the bandpass 1 (lpf0-lpf1)- BP, default = 5 . unsigned , default = 5
15: 8	R/W	0d5	reg_adp_hlти_bst_core2 : : core of the bandpass 2 (hdn-lpf0)- HP, default = 5 . unsigned , default = 5
7: 0	R/W	0d3	reg_adp_hlти_bst_core3 : : core of the unsharp band (yuvin-hdn) - US, default = 3 . unsigned , default = 3

**Table 9-2064 SRSHARP1\_HLTI\_CON\_2\_GAIN\_0 0x5237**

Bit(s)	R/W	Default	Description
31:29	R/W	0d2	reg_adp_hlti_con_mode : : con mode 0:[0, 0,-1, 1, 0, 0, 0]+[0, 0, 0, 1,-1, 0, 0], 1:[0, 0,-1, 0, 1, 0, 0], 2:[0,-1, 0, 0, 0, 1, 0], 3:[-1, 0, 0, 0, 0, 0, 1], 4: ....., default = 2. unsigned , default = 2
28:26	R/W	0d3	reg_adp_hlti_dx_mode : : dx mode 0: [-1 1 0]; 1~7: [-1 (2x+1)"0" 1], default = 3 . unsigned , default = 3
25:24	R/W	0d1	reg_adp_hlti_con_lpf : : lpf mode of the con: 0: [1 2 1]/4; 1:[1 2 2 1]/8, default = 1 . unsigned , default = 1
23:16	R/W	0d25	reg_adp_hlti_con_2_gain0 : 25;, default = 25 . unsigned , default = 25
15: 8	R/W	0d60	reg_adp_hlti_con_2_gain1 : 60;, default = 60 . unsigned , default = 60
7: 0	R/W	0d90	reg_adp_hlti_con_2_gain2 : 0;, default = 90 . unsigned , default = 90

**Table 9-2065 SRSHARP1\_HLTI\_CON\_2\_GAIN\_1 0x5238**

Bit(s)	R/W	Default	Description
31:24	R/W	0d96	reg_adp_hlti_con_2_gain3 : 96;, default = 96 . unsigned , default = 96
23:16	R/W	0d95	reg_adp_hlti_con_2_gain4 : 5;, default = 95 . unsigned , default = 95
15: 8	R/W	0d80	reg_adp_hlti_con_2_gain5 : 80;, default = 80 . unsigned , default = 80
7: 0	R/W	0d20	reg_adp_hlti_con_2_gain6 : 20;, default = 20 . unsigned , default = 20

**Table 9-2066 SRSHARP1\_HLTI\_OS\_MARGIN 0x5239**

Bit(s)	R/W	Default	Description
7: 0	R/W	0d0	reg_adp_hlti_os_margin : : margin for hlti overshoot, default = 0 . unsigned , default = 0

**Table 9-2067 SRSHARP1\_VLTI\_FLT\_CON\_CLP 0x523a**

Bit(s)	R/W	Default	Description
14	R/W	0d1	reg_adp_vlti_en : : enable bit of vlti, default = 1 . unsigned , default = 1
13:12	R/W	0d3	reg_adp_vlti_hxn_fit : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
11:10	R/W	0d3	reg_adp_vlti_dxn_fit : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
9: 8	R/W	0d3	reg_adp_vlti_han_fit : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
7: 6	R/W	0d3	reg_adp_vlti_dan_fit : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
5: 4	R/W	0d2	reg_adp_vlti_dx_mode : : 0:[-1 1] 1:[-1 0 -1]; 2/3: [-1 0 0 0 -1], default = 2 . unsigned , default = 2

Bit(s)	R/W	Default	Description
2	R/W	0d1	reg_adp_vlti_con_lpf : : lpf mode of the con: 0: [1 2 1]/4; 1:[1 2 2 2 1]/8, default = 1 . unsigned , default = 1
0	R/W	0d1	reg_adp_vlti_hard_clp_win : : window size; 0: 1x3 window; 1: 1x5 window, default = 1 . unsigned , default = 1

**Table 9-2068 SRSHARP1\_VLTI\_BST\_GAIN 0x523b**

Bit(s)	R/W	Default	Description
23:16	R/W	0d32	reg_adp_vlti_bst_gain0 : : gain to boost filter [-1 2 -1];, default = 32 . unsigned , default = 32
15: 8	R/W	0d32	reg_adp_vlti_bst_gain1 : : gain to boost filter [-1 0 2 0 -1];, default = 32 . unsigned , default = 32
7: 0	R/W	0d32	reg_adp_vlti_bst_gain2 : : gain to boost filter usf, default = 32 . unsigned , default = 32

**Table 9-2069 SRSHARP1\_VLTI\_BST\_CORE 0x523c**

Bit(s)	R/W	Default	Description
23:16	R/W	0d5	reg_adp_vlti_bst_core0 : : coring to boost filter [-1 2 -1];, default = 5 . unsigned , default = 5
15: 8	R/W	0d5	reg_adp_vlti_bst_core1 : : coring to boost filter [-1 0 2 0 -1];, default = 5 . unsigned , default = 5
7: 0	R/W	0d3	reg_adp_vlti_bst_core2 : : coring to boost filter usf, default = 3 . unsigned , default = 3

**Table 9-2070 SRSHARP1\_VLTI\_CON\_2\_GAIN\_0 0x523d**

Bit(s)	R/W	Default	Description
31:24	R/W	0d25	reg_adp_vlti_con_2_gain0 : 25;, default = 25 . unsigned , default = 25
23:16	R/W	0d69	reg_adp_vlti_con_2_gain1 : 60;, default = 69 . unsigned , default = 60
15: 8	R/W	0d90	reg_adp_vlti_con_2_gain2 : 0;, default = 90 . unsigned , default = 90
7: 0	R/W	0d96	reg_adp_vlti_con_2_gain3 : 96;, default = 96 . unsigned , default = 96

**Table 9-2071 SRSHARP1\_VLTI\_CON\_2\_GAIN\_1 0x523e**

Bit(s)	R/W	Default	Description
31:24	R/W	0d95	reg_adp_vlti_con_2_gain4 : 5;, default = 95 . unsigned , default = 95
23:16	R/W	0d80	reg_adp_vlti_con_2_gain5 : 80;, default = 80 . unsigned , default = 80
15: 8	R/W	0d20	reg_adp_vlti_con_2_gain6 : 20;, default = 20 . unsigned , default = 20
7: 0	R/W	0d0	reg_adp_vlti_os_margin : : margin for vlti overshoot, default = 0 . unsigned , default = 0

**Table 9-2072 SRSHARP1\_VCTI\_FLT\_CON\_CLP 0x523f**

Bit(s)	R/W	Default	Description
14	R/W	0d1	reg_adp_vcti_en : : enable bit of vlti, default = 1 . unsigned , default = 1
13:12	R/W	0d3	reg_adp_vcti_hxnflt : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
11:10	R/W	0d3	reg_adp_vcti_dxnflt : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
9: 8	R/W	0d3	reg_adp_vcti_hanflt : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
7: 6	R/W	0d3	reg_adp_vcti_danflt : : 0: no dn; 1: [1 2 1]/4; 2 : [1 2 2 2 1]/8 3:[1 0 2 0 1]/4, default = 3 . unsigned , default = 3
5: 4	R/W	0d2	reg_adp_vcti_dx_mode : : 0:[-1 1] 1:[-1 0 -1]; 2/3: [-1 0 0 0 -1], default = 2 . unsigned , default = 2
2	R/W	0d1	reg_adp_vcti_con_lpf : : lpf mode of the con: 0: [1 2 1]/4; 1:[1 2 2 2 1]/8, default = 1 . unsigned , default = 1
0	R/W	0d1	reg_adp_vcti_hard_clp_win : : window size; 0: 1x3 window; 1: 1x5 window, default = 1 . unsigned , default = 1

**Table 9-2073 SRSHARP1\_VCTI\_BST\_GAIN 0x5240**

Bit(s)	R/W	Default	Description
23:16	R/W	0d0	reg_adp_vcti_bst_gain0 : : gain to boost filter [-1 2 -1];, default = 0 . unsigned , default = 0
15: 8	R/W	0d0	reg_adp_vcti_bst_gain1 : : gain to boost filter [-1 0 2 0 -1];, default = 0 . unsigned , default = 0
7: 0	R/W	0d0	reg_adp_vcti_bst_gain2 : : gain to boost filter usf, default = 0 . unsigned , default = 0

**Table 9-2074 SRSHARP1\_VCTI\_BST\_CORE 0x5241**

Bit(s)	R/W	Default	Description
23:16	R/W	0d0	reg_adp_vcti_bst_core0 : : coring to boost filter [-1 2 -1];, default = 0 . unsigned , default = 0
15: 8	R/W	0d0	reg_adp_vcti_bst_core1 : : coring to boost filter [-1 0 2 0 -1];, default = 0 . unsigned , default = 0
7: 0	R/W	0d0	reg_adp_vcti_bst_core2 : : coring to boost filter usf, default = 0 . unsigned , default = 0

**Table 9-2075 SRSHARP1\_VCTI\_CON\_2\_GAIN\_0 0x5242**

Bit(s)	R/W	Default	Description
31:24	R/W	0d25	reg_adp_vcti_con_2_gain0 : 25; , default = 25 . unsigned , default = 25
23:16	R/W	0d60	reg_adp_vcti_con_2_gain1 : 60; , default = 60 . unsigned , default = 60
15: 8	R/W	0d90	reg_adp_vcti_con_2_gain2 : 0; , default = 90 . unsigned , default = 90
7: 0	R/W	0d96	reg_adp_vcti_con_2_gain3 : 96; , default = 96 . unsigned , default = 96



**Table 9-2076 SRSHARP1\_VCTI\_CON\_2\_GAIN\_1 0x5243**

Bit(s)	R/W	Default	Description
31:24	R/W	0d95	reg_adp_vcti_con_2_gain4 : 5;, default = 95 . unsigned , default = 95
23:16	R/W	0d80	reg_adp_vcti_con_2_gain5 : 80;, default = 80 . unsigned , default = 80
15: 8	R/W	0d20	reg_adp_vcti_con_2_gain6 : 20;, default = 20 . unsigned , default = 20
7: 0	R/W	0d0	reg_adp_vcti_os_margin : : margin for vcti overshoot, default = 0 . unsigned , default = 0

**Table 9-2077 SRSHARP1\_SHARP\_3DLIMIT 0x5244**

Bit(s)	R/W	Default	Description
28:16	R/W	0d0	reg_3d_mid_width : ,width of left part of 3d input, dft = half size of input width default = 0 . unsigned , default = 960
12: 0	R/W	0d0	reg_3d_mid_height : ,height of left part of 3d input, dft = half size of input height default = 0 . unsigned , default = 540

**Table 9-2078 SRSHARP1\_DNLP\_EN 0x5245**

Bit(s)	R/W	Default	Description
15□8	R/W	0d0	reg_dnlp_hblank: . unsigned , default = 8
0	R/W	0d1	reg_dnlp_en : . unsigned , default = 1

**Table 9-2079 SRSHARP1\_DEMO\_CTRL 0x5256**

Bit(s)	R/W	Default	Description
18:17	R/W	0d2	demo_disp_position : . unsigned , default = 2
16	R/W	0d0	demo_hsvsharp_enable : . unsigned , default = 0
12: 0	R/W	0d360	demo_left_top_screen_width : : . unsigned , default = 360

**Table 9-2080 SRSHARP1\_SHARP\_SR2\_CTRL 0x5257**

Bit(s)	R/W	Default	Description
24	R/W	0	reg_sr2_bic_pknr_bypass : bypass peaking/TI/Cubic
23:22	R/W		reserved
21:16	R/W	24	sr2_pk_la_err_dis_rate, low angle and high angle error should not be no less than nearby_error* rate/64
15: 8	R/W	16	sr2_pk_sad_diag_gain, gain to sad[2] and sad[6], 16 normalized to 1
7	R/W	0	sr2_vert_outphs, vertical output pixel phase, 0: 0 phase; 1: 1/2 phase
6	R/W	0	sr2_horz_outphs, horizontal output pixel phase, 0: 0 phase; 1: 1/2 phase
5	R/W	0	sr2_vert_ratio , vertical scale ratio, 0-> 1:1; 1-> 1:2
4	R/W	0	sr2_horz_ratio , horizontal scale ratio, 0-> 1:1; 1-> 1:2
3	R/W	1	sr2_bic_norm , normalization of bicubical: 0: 128; 1: 64

Bit(s)	R/W	Default	Description
2	R/W	0	sr2_enable , 1 to enable super scaler
1	R/W	0	sr2_sharp_prc_lr_hbic,
0	R/W	0	sr2_sharp_prc_lr, 1: LTI/CTI/NR/Peaking processing using LR grid. 0: on HR grid; 1:on LR grid, horizontally no upscale, but using simple bic.

**Table 9-2081 SRSHARP1\_SHARP\_SR2\_YBIC\_HCOEF0 0x5258**

Bit(s)	R/W	Default	Description
31:24	R/W	0	sr2_y_bic_hcoeff03, signed
23:16	R/W	0	sr2_y_bic_hcoeff02, signed
15: 8	R/W	64	sr2_y_bic_hcoeff01, signed
7: 0	R/W	0	sr2_y_bic_hcoeff00, signed

**Table 9-2082 SRSHARP1\_SHARP\_SR2\_YBIC\_HCOEF1 0x5259**

Bit(s)	R/W	Default	Description
31:24	R/W	-4	sr2_y_bic_hcoeff13 , signed
23:16	R/W	36	sr2_y_bic_hcoeff12 , signed
15: 8	R/W	36	sr2_y_bic_hcoeff11 , signed
7: 0	R/W	-4	sr2_y_bic_hcoeff10 , signed

**Table 9-2083 SRSHARP1\_SHARP\_SR2\_CBIC\_HCOEF0 0x525a**

Bit(s)	R/W	Default	Description
31:24	R/W	0	sr2_c_bic_hcoeff03 , signed
23:16	R/W	21	sr2_c_bic_hcoeff02 , signed
15: 8	R/W	22	sr2_c_bic_hcoeff01 , signed
7: 0	R/W	21	sr2_c_bic_hcoeff00 , signed

**Table 9-2084 SRSHARP1\_SHARP\_SR2\_CBIC\_HCOEF1 0x525b**

Bit(s)	R/W	Default	Description
31:24	R/W	-4	sr2_c_bic_hcoeff13 , signed
23:16	R/W	36	sr2_c_bic_hcoeff12 , signed
15: 8	R/W	36	sr2_c_bic_hcoeff11 , signed
7: 0	R/W	-4	sr2_c_bic_hcoeff10 , signed

**Table 9-2085 SRSHARP1\_SHARP\_SR2\_YBIC\_VCOEF0 0x525c**

Bit(s)	R/W	Default	Description
31:24	R/W	0	sr2_y_bic_vcoeff03 , signed
23:16	R/W	0	sr2_y_bic_vcoeff02 , signed
15: 8	R/W	64	sr2_y_bic_vcoeff01 , signed
7: 0	R/W	0	sr2_y_bic_vcoeff00 , signed

**Table 9-2086 SRSHARP1\_SHARP\_SR2\_YBIC\_VCOEF1 0x525d**

Bit(s)	R/W	Default	Description
31:24	R/W	-4	sr2_y_bic_vcoeff13 , signed
23:16	R/W	36	sr2_y_bic_vcoeff12 , signed
15: 8	R/W	36	sr2_y_bic_vcoeff11 , signed
7: 0	R/W	-4	sr2_y_bic_vcoeff10 , signed

**Table 9-2087 SRSHARP1\_SHARP\_SR2\_CBIC\_VCOEF0 0x525e**

Bit(s)	R/W	Default	Description
31:24	R/W	0	sr2_c_bic_vcoeff03 , signed
23:16	R/W	21	sr2_c_bic_vcoeff02 , signed
15: 8	R/W	22	sr2_c_bic_vcoeff01 , signed
7: 0	R/W	21	sr2_c_bic_vcoeff00 , signed

**Table 9-2088 SRSHARP1\_SHARP\_SR2\_CBIC\_VCOEF1 0x525f**

Bit(s)	R/W	Default	Description
31:24	R/W	-4	sr2_c_bic_vcoeff13 , signed
23:16	R/W	36	sr2_c_bic_vcoeff12 , signed
15: 8	R/W	36	sr2_c_bic_vcoeff11 , signed
7: 0	R/W	-4	sr2_c_bic_vcoeff10 , signed

**Table 9-2089 SRSHARP1\_SHARP\_SR2\_MISC 0x5260**

Bit(s)	R/W	Default	Description
31:2	R/W		reserved
1	R/W	0	sr2_cmpmux_bef , 0 : no swap for YUV/RGB; 1: swap for YUV/RGB, YUV/RGB->UVY/GBR
0	R/W	0	sr2_cmpmux_aft , 0 : no swap for YUV/RGB; 1: swap for YUV/RGB, UVY/GBR->YUV/RGB

**Table 9-2090 SRSHARP1\_SR3\_SAD\_CTRL 0x5261**

Bit(s)	R/W	Default	Description
31:30	R/W		reserved
29:24	R/W	0d6	reg_sr3_pk_sad_core_rate : rate of coring.
23:22	R/W		reserved
21:16	R/W	0d6	reg_sr3_lti_sad_core_rate : rate of coring.
15:14	R/W		reserved
13:8	R/W	0d6	reg_sr3_cti_sad_core_rate : rate of coring.
7	R/W	0d1	reg_sr3_lti_hsad_mode: mode for hsad of lti calculation, 0:block based; 1: other sharp
6	R/W	0d1	reg_sr3_cti_hsad_mode: mode for hsad of cti calculation, 0:block based; 1: other sharp
5	R/W	0d1	reg_sr3_lti_dsad_mode: mode for dsad of lti calculation, 0:block based; 1: other sharp
4	R/W	0d1	reg_sr3_cti_dsad_mode: mode for dsad of cti calculation, 0:block based; 1: other sharp
3	R/W	0d1	reg_sr3_lti_vsad_mode: mode for vsad of lti calculation, 0:block based; 1: other sharp
2	R/W	0d1	reg_sr3_cti_vsad_mode: mode for vsad of cti calculation, 0:block based; 1: other sharp
1	R/W	0d1	reg_sr3_lti_hsad_hlpf: hlpf for hsad of lti calculation, 0:no hlpf; 1: with [1 2 1] hlpf.
0	R/W	0d1	reg_sr3_cti_hsad_hlpf: hlpf for hsad of cti calculation, 0:no hlpf; 1: with [1 2 1] hlpf.

**Table 9-2091 SRSHARP1\_SR3\_PK\_CTRL0 0x5262**

Bit(s)	R/W	Default	Description
31:12	R/W		reserved
11	R/W	0d1	reg_sr3_pk_sad_mode: mode for sad of peaking calculation, 0: block based; 1: other sharp.
10	R/W	0d1	reg_sr3_pk_hsad_hlpf: hlpf for hsad for peaking calculation,0:no hlpf; 1: with [1 2 2 2 1] hlpf.
9	R/W	0d1	reg_sr3_pk_vsad_hlpf: hlpf for vsad for peaking calculation,0:no hlpf; 1: with [1 2 2 2 1] hlpf.
8	R/W	0d1	reg_sr3_pk_dsad_hlpf: hlpf for dsad for peaking calculation,0:no hlpf; 1: with [1 2 2 2 1] hlpf.
7:6	R/W	0d3	reg_sr3_pk_hpdrtr_mode: mode for HPdrtr filter
5:4	R/W	0d3	reg_sr3_pk_bpdrtr_mode: mode for BPdrtr filter
3:2	R/W	0d3	reg_pk_drtrbld_range: range of the min2 and min direction distance
1	R/W		reserved
0	R/W	0d0	reg_sr3_pk_ti_blend_mode: blend mode of the TI and PK result

**Table 9-2092 SRSHARP1\_SR3\_PK\_CTRL1 0x5263**

Bit(s)	R/W	Default	Description
31	R/W		reserved
30:28	R/W	0d1	reg_sr3_pk_hp_hvcon_replace8_maxsad: replace HP hvcon by maxsad
26:24	R/W	0d1	reg_sr3_pk_bp_hvcon_replace8_maxsad: replace BP hvcon by maxsad
23:16	R/W	0d32	reg_sr3_pk_hp_hvcon_replace8lv_gain: gain to local variant before calculating the hv gain for peaking.
15:8	R/W	0d32	reg_sr3_pk_bp_hvcon_replace8lv_gain: gain to local variant before calculating the hv gain for peaking.
7	R/W	0d1	reg_sr3_sad_intlev_mode: interleave detect xerr mode: 0 max; 1: sum
6	R/W	0d1	reg_sr3_sad_intlev_mode1: mode 1 of using diagonal protection: 1: with diagonal protection
5:0	R/W	0d12	reg_sr3_sad_intlev_gain: interleave detection for sad gain applied, normalized to 8 as 1

**Table 9-2093 SRSHARP1\_DEJ\_CTRL 0x5264**

Bit(s)	R/W	Default	Description
15:8	R/W	8	reg_sr3_dejaggy_hblank
7:4	R/W		reserved
3:2	R/W	0d3	reg_sr3_dejaggy_sameside_prct: enable of sr3 dejaggy same side curve protect from filter, [0] for proc path; [1] for ctrl path.
1	R/W	0d1	reg_sr3_dejaggy_sameside_mode: mode of sameside flag decision
0	R/W	0d1	reg_sr3_dejaggy_enable: enable of sr3 dejaggy

**Table 9-2094 SRSHARP1\_DEJ\_ALPHA 0x5265**

Bit(s)	R/W	Default	Description
31:28	R/W	0d0	reg_sr3_dejaggy_ctrchrom_alpha_1 : alpha for LR video LPF
27:24	R/W	0d15	reg_sr3_dejaggy_ctrchrom_alpha_0 : alpha for LR video LPF
23:20	R/W	0d0	reg_sr3_dejaggy_ctrluma_alpha_1 : alpha for LR video LPF
19:16	R/W	0d15	reg_sr3_dejaggy_ctrluma_alpha_0 : alpha for LR video LPF
15:12	R/W	0d4	reg_sr3_dejaggy_procchrom_alpha_1: alpha for LR video LPF
11:8	R/W	0d6	reg_sr3_dejaggy_procchrom_alpha_0: alpha for LR video LPF
7:4	R/W	0d4	reg_sr3_dejaggy_procluma_alpha_1: alpha for LR video LPF
3:0	R/W	0d6	reg_sr3_dejaggy_procluma_alpha_0: alpha for LR video LPF

**Table 9-2095 SRSHARP1\_SR3\_DRTLPF\_EN 0x5266**

Bit(s)	R/W	Default	Description
31:15	R/W		reserved
14:8	R/W	0d0	reg_pk_debug_edge
6:4	R/W	0d0	reg_sr3_drtlpf_theta_en
2:0	R/W	0d7	reg_sr3_drtlpf_enable: directional lpf on Y/U/V channels

**Table 9-2096 SRSHARP1\_SR3\_DRTLPF\_ALPHA\_0 0x5267**

Bit(s)	R/W	Default	Description
31:28	R/W		reserved
27:24	R/W	0d9	reg_sr3_drtlpf_alpha_3
23:20	R/W		reserved
19:16	R/W	0d10	reg_sr3_drtlpf_alpha_2
15:12	R/W		reserved
11:8	R/W	0d11	reg_sr3_drtlpf_alpha_1
7:4	R/W		reserved
3:0	R/W	0d12	reg_sr3_drtlpf_alpha_0: directional lpf alpha coef for min_sad/max_sad compare

**Table 9-2097 SRSHARP1\_SR3\_DRTLPF\_ALPHA\_1 0x5268**

Bit(s)	R/W	Default	Description
31:28	R/W		reserved
27:24	R/W	0d1	reg_sr3_drtlpf_alpha_7
23:20	R/W		reserved
19:16	R/W	0d4	reg_sr3_drtlpf_alpha_6
15:12	R/W		reserved
11:8	R/W	0d7	reg_sr3_drtlpf_alpha_5
7:4	R/W		reserved
3:0	R/W	0d8	reg_sr3_drtlpf_alpha_4: directional lpf alpha coef for min_sad/max_sad compare

**Table 9-2098 SRSHARP1\_SR3\_DRTLPF\_ALPHA\_2 0x5269**

Bit(s)	R/W	Default	Description
31:28	R/W		reserved
27:24	R/W	0d0	reg_sr3_drtlpf_alpha_11
23:20	R/W		reserved
19:16	R/W	0d0	reg_sr3_drtlpf_alpha_10
15:12	R/W		reserved

Bit(s)	R/W	Default	Description
11:8	R/W	0d0	reg_sr3_drtlpf_alpha_9
7:4	R/W		reserved
3:0	R/W	0d0	reg_sr3_drtlpf_alpha_8: directional lpf alpha coef for min_sad/max_sad compare

Table 9-2099 SRSHARP1\_SR3\_DRTLPF\_ALPHA\_OFST 0x526a

Bit(s)	R/W	Default	Description
31:28	R/W	0	reg_sr3_drtlpf_alpha_ofst7
27:24	R/W	0	reg_sr3_drtlpf_alpha_ofst6
23:20	R/W	0	reg_sr3_drtlpf_alpha_ofst5
19:16	R/W	-2	reg_sr3_drtlpf_alpha_ofst4
15:12	R/W	0	reg_sr3_drtlpf_alpha_ofst3
11:8	R/W	0	reg_sr3_drtlpf_alpha_ofst2
7:4	R/W	0	reg_sr3_drtlpf_alpha_ofst1
3:0	R/W	-2	reg_sr3_drtlpf_alpha_ofst0: directional lpf alpha coef offset of each direction.

Table 9-2100 SRSHARP1\_SR3\_DERING\_CTRL 0x526b

Bit(s)	R/W	Default	Description
31	R/W		reserved
30:28	R/W	1	reg_sr3_dering_enable: dering enable
27	R/W		reserved
26:24	R/W	3	reg_sr3_dering_varlpf_mode: local variant LPF mode. 0: no filter; 1: erosion 3x3; 2: 3x3 lpf; 3: 3x3 erosion + lpf
23:20	R/W	9	reg_sr3_dering_maxrange: range of dering in LR resolution.
19:18	R/W		reserved
17:16	R/W	2	reg_sr3_dering_lcvar_blend_mode: mode for lcvar calculation. 0:HV blend; 1:diag blend; 2:HV blend + V; 3: HV blend+Diag blend
15:8	R/W	40	reg_sr3_dering_lcvar_gain: gain to local variant and normalized to 32 as 1
7:0	R/W	28	reg_sr3_dering_lcvar_nearby_maxsad_th: threshold to use near side maxsad if that side is larger than this threshold, otherwise use the max one.

Table 9-2101 SRSHARP1\_SR3\_DERING\_LUMA2PKGAIN\_0TO3 0x526c

Bit(s)	R/W	Default	Description
31:24	R/W	255	reg_sr3_dering_luma2pkgain3: level limit(for th0<bpcon<th1) of curve for dering pkgain base on LPF luma level
23:16	R/W	255	reg_sr3_dering_luma2pkgain2: level limit(for bpcon<th0) of curve for dering pkgain base on LPF luma level

Bit(s)	R/W	Default	Description
15:8	R/W	200	reg_sr3_deriving_luma2pkgain1: threshold 1 of curve for deriving pkgain based on LPF luma level.
7:0	R/W	30	reg_sr3_deriving_luma2pkgain0: threshold 0 of curve for deriving pkgain based on LPF luma level.

**Table 9-2102 SRSHARP1\_SR3\_DERIVING\_LUMA2PKGAIN\_4TO6 0x526d**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:16	R/W	24	reg_sr3_deriving_luma2pkgain6: rate1 (for bpcon>th1) of curve for deriving pkOS based on LPF luma level.
15:8	R/W	50	reg_sr3_deriving_luma2pkgain5: rate0 (for bpcon<th0) of curve for deriving pkOS based on LPF luma level.
7:0	R/W	255	reg_sr3_deriving_luma2pkgain4: level limit(for bpcon>th1) of curve for deriving pkgain base on LPF luma level

**Table 9-2103 SRSHARP1\_SR3\_DERIVING\_LUMA2PKOS\_0TO3 0x526e**

Bit(s)	R/W	Default	Description
31:24	R/W	255	reg_sr3_deriving_luma2pkos3: level limit(for th0<bpcon<th1) of curve for deriving pkOS base on LPF luma level
23:16	R/W	255	reg_sr3_deriving_luma2pkos2: level limit(for bpcon<th0) of curve for deriving pkOS base on LPF luma level
15:8	R/W	200	reg_sr3_deriving_luma2pkos1: threshold 1 of curve for deriving pkOS based on LPF luma level.
7:0	R/W	30	reg_sr3_deriving_luma2pkos0: threshold 0 of curve for deriving pkOS based on LPF luma level.

**Table 9-2104 SRSHARP1\_SR3\_DERIVING\_LUMA2PKOS\_4TO6 0x526f**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:16	R/W	24	reg_sr3_deriving_luma2pkos6: rate1 (for bpcon>th1) of curve for deriving pkOS based on LPF luma level.
15:8	R/W	50	reg_sr3_deriving_luma2pkos5: rate0 (for bpcon<th0) of curve for deriving pkOS based on LPF luma level.
7:0	R/W	255	reg_sr3_deriving_luma2pkos4: level limit(for bpcon>th1) of curve for deriving pkOS base on LPF luma level

**Table 9-2105 SRSHARP1\_SR3\_DERIVING\_GAINVS\_MADSAD 0x5270**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reg_sr3_deriving_gainvs_maxsad7
27:24	R/W	0	reg_sr3_deriving_gainvs_maxsad6
23:20	R/W	0	reg_sr3_deriving_gainvs_maxsad5



Bit(s)	R/W	Default	Description
19:16	R/W	0	reg_sr3_deriving_gainvs_maxsad4
15:12	R/W	0	reg_sr3_deriving_gainvs_maxsad3
11:8	R/W	0	reg_sr3_deriving_gainvs_maxsad2
7:4	R/W	4	reg_sr3_deriving_gainvs_maxsad1
3:0	R/W	8	reg_sr3_deriving_gainvs_maxsad0: pkgain vs maxsad value, 8 node interpolations.

**Table 9-2106 SRSHARP1\_SR3\_DERIVING\_GAINVS\_VR2MAX 0x5271**

Bit(s)	R/W	Default	Description
31:28	R/W	15	reg_sr3_deriving_gainvs_vr2max7
27:24	R/W	15	reg_sr3_deriving_gainvs_vr2max6
23:20	R/W	15	reg_sr3_deriving_gainvs_vr2max5
19:16	R/W	15	reg_sr3_deriving_gainvs_vr2max4
15:12	R/W	14	reg_sr3_deriving_gainvs_vr2max3
11:8	R/W	12	reg_sr3_deriving_gainvs_vr2max2
7:4	R/W	2	reg_sr3_deriving_gainvs_vr2max1
3:0	R/W	0	reg_sr3_deriving_gainvs_vr2max0: pkgain vs ratio

**Table 9-2107 SRSHARP1\_SR3\_DERIVING\_PARAM0 0x5272**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:16	R/W	10	reg_sr3_deriving_lcvar_floor
15:8	R/W	32	reg_sr3_deriving_vr2max_gain: gain to max before feeding to LUT
7:6	R/W		reserved
5:0	R/W	16	reg_sr3_deriving_vr2max_limt: limit of maxsad

**Table 9-2108 SRSHARP1\_SR3\_DRTLPF\_THETA 0x5273**

Bit(s)	R/W	Default	Description
31:0	R/W	0xfe-c96420	reg_sr3_drtlpf_theta: u4x8 directional lpf beta coef for min_sad/min2_sad compared to x=0:7 correspond to [1:8]/16; 0 means no drtLPF, 15: 100% alpha dependant drtLPF.

**Table 9-2109 SRSHARP1\_SATPRT\_CTRL 0x5274**

Bit(s)	R/W	Default	Description
31:28	R/W		reserved
27:16	R/W	5	reg_satprt_sat_core: 4x will be coring to cor(irgb_max-irgb_min) to calculate the oy_delt, the smaller the more protection to color, the larger only the rich color will be protected.
15:8	R/W	64	reg_satprt_sat_rate: rate to cor(irgb_max-irgb_min) to calculate the oy_delt, the larger the more protection to color; norm 16 as 1
7:4	R/W		reserved
3:2	R/W	1	reg_satprt_csc_mode: CSC mode of current yuv input: 0:601; 1:709; 2:BT2020 NCL; 3 reserved
1	R/W	1	reg_satprt_is_lmt: flag telling the YUV is limited range data or full rang data; 1 is limited data
0	R/W	0	reg_satprt_enable: 1 to enable of saturation protection for dnlp adjustments

**Table 9-2110 SRSHARP1\_SATPRT\_DIVM 0x5275**

Bit(s)	R/W	Default	Description
31:24	R/W		reserved
23:0	R/W	{128,12-8,128}	reg_satprt_div_m: u8x3, 1/m, normalized to 128 as 1.

**Table 9-2111 SRSHARP1\_DB\_FLT\_CTRL 0x5277**

Bit(s)	R/W	Default	Description
26	R/W	0	reg_nrdeband_reset1 : // unsigned , default = 0 0 : no reset seed 1: reload chroma seed
25	R/W	0	reg_nrdeband_reset0 : // unsigned , default = 0 0 : no reset seed 1: reload luma seed
24	R/W	0	reg_nrdeband_rgb : // unsigned , default = 0 0 : yuv 1: RGB
23	R/W	1	reg_nrdeband_en11 : // unsigned , default = 1 debanding registers of side lines, [0] for luma, same for below
22	R/W	1	reg_nrdeband_en10 : // unsigned , default = 1 debanding registers of side lines, [1] for chroma, same for below
21	R/W	1	reg_nrdeband_siderand : // unsigned , default = 1 options to use side two lines use the rand, instead of use for the YUV three component of middle line, 0: seed [3]/bandrand[3] for middle line yuv; 1: seed[3]/bandrand[3] for nearby three lines Y;
20	R/W	0	reg_nrdeband_randmode : // unsigned , default = 0 mode of rand noise adding, 0: same noise strength for all difs; else: strenght of noise will not exceed the difs, MIN((pPKReg->reg_nrdeband_bandrand[m]), noise[m])
19:17	R/W	6	reg_nrdeband_bandrand2 : // unsigned , default = 6
15:13	R/W	6	reg_nrdeband_bandrand1 : // unsigned , default = 6
11: 9	R/W	6	reg_nrdeband_bandrand0 : // unsigned , default = 6
7	R/W	1	reg_nrdeband_hpxor1 : // unsigned , default = 1 debanding random hp portion xor, [0] for luma

Bit(s)	R/W	Default	Description
6	R/W	1	reg_nrdeband_hpxor0 : // unsigned , default = 1 debanding random hp portion xor, [1] for chroma
5	R/W	1	reg_nrdeband_en1 : // unsigned , default = 1 debanding registers, for luma
4	R/W	1	reg_nrdeband_en0 : // unsigned , default = 1 debanding registers, for chroma
3: 2	R/W	2	reg_nrdeband_lpf_mode1 : // unsigned , default = 2 lpf mode, 0: 3x3, 1:3x5; 2: 5x5; 3:5x7
1: 0	R/W	2	reg_nrdeband_lpf_mode0 : // unsigned , default = 2 lpf mode, 0: 3x3, 1:3x5; 2: 5x5; 3:5x7

Table 9-2112 SRSHARP1\_DB\_FLT\_RANLUT 0x5279

Bit(s)	R/W	Default	Description
23:21	R/W	1	reg_nrdeband_randslut7 : // unsigned , default = 1 lut0
20:18	R/W	1	reg_nrdeband_randslut6 : // unsigned , default = 1 lut0
17:15	R/W	1	reg_nrdeband_randslut5 : // unsigned , default = 1 lut0
14:12	R/W	1	reg_nrdeband_randslut4 : // unsigned , default = 1 lut0
11: 9	R/W	1	reg_nrdeband_randslut3 : // unsigned , default = 1 lut0
8: 6	R/W	1	reg_nrdeband_randslut2 : // unsigned , default = 1 lut0
5: 3	R/W	1	reg_nrdeband_randslut1 : // unsigned , default = 1 lut0
2: 0	R/W	1	reg_nrdeband_randslut0 : // unsigned , default = 1 lut0

Table 9-2113 SRSHARP1\_DB\_FLT\_PXI\_THRD 0x527a

Bit(s)	R/W	Default	Description
25:16	R/W	0	reg_nrdeband_yc_th1 : // unsigned , default = 0 to luma/ u/v  for using the denoise
9: 0	R/W	0	reg_nrdeband_yc_th0 : // unsigned , default = 0 to luma/ u/v  for using the denoise

Table 9-2114 SRSHARP1\_DB\_FLT\_SEED\_Y 0x527b

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8240	reg_nrdeband_seed0 : // unsigned , default = 1621438240 noise adding seed for Y. seed[0]= 0x60a52f20; as default

Table 9-2115 SRSHARP1\_DB\_FLT\_SEED\_U 0x527c

Bit(s)	R/W	Default	Description
31: 0	R/W	162143-8247	reg_nrdeband_seed1 : // unsigned , default = 1621438247 noise adding seed for U. seed[0]= 0x60a52f27; as default

**Table 9-2116 SRSHARP1\_DB\_FLT\_SEED\_V 0x527d**

Bit(s)	R/W	Default	Description
31:0	R/W	162143-8242	reg_nrdeband_seed2 : // unsigned , default = 1621438242 noise adding seed for V. seed[0]= 0x60a52f22; as default

**Table 9-2117 SRSHARP1\_PKGAIN\_VSLUMA\_LUT\_L 0x527e**

Bit(s)	R/W	Default	Description
31:28	R/W	5	reg_pkgain_vsluma_lut7
27:24	R/W	6	reg_pkgain_vsluma_lut6
23:20	R/W	6	reg_pkgain_vsluma_lut5
19:16	R/W	6	reg_pkgain_vsluma_lut4
15:12	R/W	7	reg_pkgain_vsluma_lut3
11:8	R/W	10	reg_pkgain_vsluma_lut2
7:4	R/W	12	reg_pkgain_vsluma_lut1
3:0	R/W	8	reg_pkgain_vsluma_lut0

**Table 9-2118 SRSHARP1\_PKGAIN\_VSLUMA\_LUT\_H 0x527f**

Bit(s)	R/W	Default	Description
31:4	R/W		reserved
3:0	R/W	4	reg_pkgain_vsluma_lut8

**Table 9-2119 SRSHARP1\_PKOSHT\_VSLUMA\_LUT\_L 0x5280**

Bit(s)	R/W	Default	Description
31:28	R/W	5	reg_pkosht_vsluma_lut7
27:24	R/W	6	reg_pkosht_vsluma_lut6
23:20	R/W	6	reg_pkosht_vsluma_lut5
19:16	R/W	6	reg_pkosht_vsluma_lut4
15:12	R/W	7	reg_pkosht_vsluma_lut3
11:8	R/W	10	reg_pkosht_vsluma_lut2
7:4	R/W	12	reg_pkosht_vsluma_lut1
3:0	R/W	8	reg_pkosht_vsluma_lut0

**Table 9-2120 SRSHARP1\_PKOSHT\_VSLUMA\_LUT\_H 0x5281**

Bit(s)	R/W	Default	Description
31:4	R/W		reserved
3:0	R/W	4	reg_pkosht_vsluma_lut8

**Table 9-2121 SRSHARP1\_SATPRT\_LMT\_RGB1 0x5282**

Bit(s)	R/W	Default	Description
27:16	R/W	0d0	reg_satprt_lmt_g:
11: 0	R/W	0d0	reg_satprt_lmt_r: limit of RGB channel, for limited range RGB, 12bits

**Table 9-2122 SRSHARP1\_SATPRT\_LMT\_RGB2 0x5283**

Bit(s)	R/W	Default	Description
31:16	R/W	0d0	reserved
11: 0	R/W	0d0	reg_satprt_lmt_b: limit of RGB channel, for limited range RGB

**Table 9-2123 SRSHARP1\_SHARP\_GATE\_CLK\_CTRL\_0 0x5284**

Bit(s)	R/W	Default	Description
31:0	R/W	0d0	Gate clock control [01:00]: sharp input control unit [03:02]: deband unit [05:04]: dejaggy unit [07:06]: dnlp unit [09:08]: demo control unit [11:10]: horiz interp unit

**Table 9-2124 SRSHARP1\_SHARP\_GATE\_CLK\_CTRL\_1 0x5285**

Bit(s)	R/W	Default	Description
31:0	R/W	0d0	Gate clock control [01:00]: sr_top "pipe_ctrl" [03:02]: drt [05:04]: ssd [07:06]: cubic [09:08]: edi [11:10]: pkgainsad [13:12]: bicomux [15:14]: bicin_fifo [17:16]: lpf4pkgain_fifo [19:18]: min2hvgain_fifo [21:20]: sad4pkgain_fifo [23:22]: dirminmax4xtl_fifo [25:24]: drtsad8_fifo [27:26]: ssdmax_fifo [29:28]: pkminmax_fifo [31:30]: cirdrtgain_fifo

Table 9-2125 SRSHARP1\_SHARP\_GATE\_CLK\_CTRL\_2 0x5286

Bit(s)	R/W	Default	Description
31:0	R/W	0d0	Gate clock control [01:00]: bufdiff_fifo [03:02]: osvar_fifo [05:04]: pkhvgain unit [07:06]: pkgain unit [09:08]: TI unit [11:10]: pk unit [13:12]: locvar unit [15:14]: hvconc unit

Table 9-2126 SRSHARP1\_SHARP\_GATE\_CLK\_CTRL\_3 0x5287

Bit(s)	R/W	Default	Description
31:0	R/W	0d0	Gate clock control [01:00]: TI, htl Y [03:02]: TI, vti Y [05:04]: TI, htl U [07:06]: TI, vtl U [09:08]: TI, htl V [11:10]: TI, vtl V [13:12]: TI, lumaminmax_fifo [15:14]: TI, chrminmax_fifo

Table 9-2127 SRSHARP1\_SHARP\_DPS\_CTRL 0x5288

Bit(s)	R/W	Default	Description
31:0	R/W	0d0	Power saving control : hvcon : nrssd : os filter : cubic 5 lines to 9 lines : dering : locvar : drtlpf : hlti : hcti : vlti : vcti : lti blend : cti blend : htishort (no used) : nr Y filter : nr C filter : nr belnd : pkti blend : os ctrl : pk HP : pk BP [26:24] dejaggy power saving control [30:28] reserved

Table 9-2128 SRSHARP1\_DNLP\_00 0x5290

Bit(s)	R/W	Default	Description
31:0	R/W	0x0010-0008	reg_dnlp_ygrid0 : : dnlp00 . unsigned , default = 32'h00100008

**Table 9-2129 SRSHARP1\_DNLP\_01 0x5291**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0020-0018	reg_dnlp_ygrid1 :: dnlp01 . unsigned , default = 32'h00200018

**Table 9-2130 SRSHARP1\_DNLP\_02 0x5292**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0030-0028	reg_dnlp_ygrid2 :: dnlp02 . unsigned , default = 32'h00300028

**Table 9-2131 SRSHARP1\_DNLP\_03 0x5293**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0040-0038	reg_dnlp_ygrid3 :: dnlp03 . unsigned , default = 32'h00400038

**Table 9-2132 SRSHARP1\_DNLP\_04 0x5294**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0050-0048	reg_dnlp_ygrid4 :: dnlp04 . unsigned , default = 32'h00500048

**Table 9-2133 SRSHARP1\_DNLP\_05 0x5295**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0068-005c	reg_dnlp_ygrid5 :: dnlp05 . unsigned , default = 32'h0068005c

**Table 9-2134 SRSHARP1\_DNLP\_06 0x5296**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0080-0074	reg_dnlp_ygrid6 :: dnlp06 . unsigned , default = 32'h00800074

**Table 9-2135 SRSHARP1\_DNLP\_07 0x5297**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x00-a00090	reg_dnlp_ygrid7 :: dnlp07 . unsigned , default = 32'h00a00090

**Table 9-2136 SRSHARP1\_DNLP\_08 0x5298**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x00-c000b0	reg_dnlp_ygrid8 :: dnlp08 . unsigned , default = 32'h00c000b0

**Table 9-2137 SRSHARP1\_DNLP\_09 0x5299**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x00-e000d0	reg_dnlp_ygrid9 :: dnlp09 . unsigned , default = 32'h00e000d0

**Table 9-2138 SRSHARP1\_DNLP\_10 0x529a**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0100-00f0	reg_dnlp_ygrid10 :: dnlp10 . unsigned , default = 32'h010000f0

**Table 9-2139 SRSHARP1\_DNLP\_11 0x529b**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x012-c0114	reg_dnlp_ygrid11 :: dnlp11 . unsigned , default = 32'h012c0114

**Table 9-2140 SRSHARP1\_DNLP\_12 0x529c**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0154-0140	reg_dnlp_ygrid12 :: dnlp12 . unsigned , default = 32'h01540140

**Table 9-2141 SRSHARP1\_DNLP\_13 0x529d**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0180-016c	reg_dnlp_ygrid13 :: dnlp13 . unsigned , default = 32'h0180016c

**Table 9-2142 SRSHARP1\_DNLP\_14 0x529e**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x01-c001a0	reg_dnlp_ygrid14 :: dnlp14 . unsigned , default = 32'h01c001a0

**Table 9-2143 SRSHARP1\_DNLP\_15 0x529f**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0200-01e0	reg_dnlp_ygrid15 :: dnlp15 . unsigned , default = 32'h020001e0

**Table 9-2144 SRSHARP1\_DNLP\_16 0x52a0**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0240-0220	reg_dnlp_ygrid16 :: dnlp16 . unsigned , default = 32'h02400220



**Table 9-2145 SRSHARP1\_DNLP\_17 0x52a1**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0280-0260	reg_dnlp_ygrid17 :: dnlp17 . unsigned , default = 32'h02800260

**Table 9-2146 SRSHARP1\_DNLP\_18 0x52a2**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x02-b00298	reg_dnlp_ygrid18 :: dnlp18 . unsigned , default = 32'h02b00298

**Table 9-2147 SRSHARP1\_DNLP\_19 0x52a3**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x02-e002c8	reg_dnlp_ygrid19 :: dnlp19 . unsigned , default = 32'h02e002c8

**Table 9-2148 SRSHARP1\_DNLP\_20 0x52a4**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0300-02f0	reg_dnlp_ygrid20 :: dnlp20 . unsigned , default = 32'h030002f0

**Table 9-2149 SRSHARP1\_DNLP\_21 0x52a5**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0320-0310	reg_dnlp_ygrid21 :: dnlp21 . unsigned , default = 32'h03200310

**Table 9-2150 SRSHARP1\_DNLP\_22 0x52a6**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0338-032c	reg_dnlp_ygrid22 :: dnlp22 . unsigned , default = 32'h0338032c

**Table 9-2151 SRSHARP1\_DNLP\_23 0x52a7**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0350-0348	reg_dnlp_ygrid23 :: dnlp23 . unsigned , default = 32'h03500348

**Table 9-2152 SRSHARP1\_DNLP\_24 0x52a8**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x035-c0358	reg_dnlp_ygrid24 :: dnlp24 . unsigned , default = 32'h035c0358

**Table 9-2153 SRSHARP1\_DNLP\_25 0x52a9**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0368-0360	reg_dnlp_ygrid25 :: dnlp25 . unsigned , default = 32'h03680360

**Table 9-2154 SRSHARP1\_DNLP\_26 0x52aa**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0378-0370	reg_dnlp_ygrid26 :: dnlp26 . unsigned , default = 32'h03780370

**Table 9-2155 SRSHARP1\_DNLP\_27 0x52ab**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x0388-0380	reg_dnlp_ygrid27 :: dnlp27 . unsigned , default = 32'h03880380

**Table 9-2156 SRSHARP1\_DNLP\_28 0x52ac**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x03-a00390	reg_dnlp_ygrid28 :: dnlp28 . unsigned , default = 32'h03a00390

**Table 9-2157 SRSHARP1\_DNLP\_29 0x52ad**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x03-c003b0	reg_dnlp_ygrid29 :: dnlp29 . unsigned , default = 32'h03c003b0

**Table 9-2158 SRSHARP1\_DNLP\_30 0x52ae**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x03-e003d0	reg_dnlp_ygrid30 :: dnlp30 . unsigned , default = 32'h03e003d0

**Table 9-2159 SRSHARP1\_DNLP\_31 0x52af**

Bit(s)	R/W	Default	Description
31: 0	R/W	0x03f-c03f0	reg_dnlp_ygrid31 :: dnlp31 . unsigned , default = 32'h03fc03f0

**Table 9-2160 SRSHARP1\_SHARP\_SYNC\_CTRL 0x52b0**

Bit(s)	R/W	Default	Description
15:0	R/W	0d0	Shadow control for some setting: 0 to use VSYNC to shadow : SR all bypass : reg_sr2_bic_pknr_bypass [2]: reg_3d_mode [3]: reg_nrdeband_en11 [4]: reg_nrdeband_en10 [5]: reg_nrdeband_en1 [6]: reg_nrdeband_en0 [7]: reg_dejaggy_enable [8]: reg_sr2_sharp_prc_lr [9]: reg_sharp_pk_en [10]: reg_sharp_nr_en [11]: reg_adp_hlti_en [12]: reg_adp_hcti_en [13]: reg_adp_vlti_en [14]: reg_adp_vcti_en [15]: reg_dnlp_en

**Table 9-2161 SRSHARP1\_LC\_INPUT\_MUX 0x52b1**

Bit(s)	R/W	Default	Description
31:7	R/W	0	reserved
6:4	R/W	5	reg_lcinput_yssel: local contrast luma input options 0: org_y; 1: gau_y; 2: gauadp_y; 3: edgadplpf_y; 4: nr_y; 5: lti_y; 6: pk_y (before os); 7: pk_y (after os)
3	R/W	0	reserved
2:0	R/W	5	reg_lcinput_csels: local contrast chroma input options 0: org_c; 1: gau_c; 2: gauadp_c; 3: edg-adplpf_c; 4: nr_c; 5: cti_c; 6: pk_c

**Table 9-2162 SRSHARP1\_NR\_GAU\_YH\_COEF02 0x52b2**

Bit(s)	R/W	Default	Description
29:20	R/W	0d32	luma's horizontal adaptive coef0 . signed , default = 32
19:10	R/W	0d32	luma's horizontal adaptive coef1 . signed , default = 32
9:0	R/W	0d16	luma's horizontal adaptive coef2 . signed , default = 16

**Table 9-2163 SRSHARP1\_NR\_GAU\_YH\_COEF34 0x52b3**

Bit(s)	R/W	Default	Description
19:10	R/W	0d0	luma's horizontal adaptive coef3 . signed , default = 0
9:0	R/W	0d0	luma's horizontal adaptive coef4 . signed , default = 0

**Table 9-2164 SRSHARP1\_NR\_GAU\_YV\_COEF1 0x52b4**

Bit(s)	R/W	Default	Description
29:20	R/W	0d32	luma's vertical adaptive coef0 . signed , default = 32
19:10	R/W	0d32	luma's vertical adaptive coef1 . signed , default = 32
9:0	R/W	0d16	luma's vertical adaptive coef2 . signed , default = 16

**Table 9-2165 SRSHARP1\_NR\_GAU\_CH\_COEF02 0x52b5**

Bit(s)	R/W	Default	Description
29:20	R/W	0d32	chroma's horizontal adaptive coef0 . signed , default = 32
19:10	R/W	0d32	chroma's horizontal adaptive coef1 . signed , default = 32
9:0	R/W	0d16	chroma's horizontal adaptive coef2 . signed , default = 16

**Table 9-2166 SRSHARP1\_NR\_GAU\_CH\_COEF34 0x52b6**

Bit(s)	R/W	Default	Description
19:10	R/W	0d0	chroma's horizontal adaptive coef3 . signed , default = 0
9:0	R/W	0d0	chroma's horizontal adaptive coef4 . signed , default = 0

**Table 9-2167 SRSHARP1\_NR\_GAU\_CV\_COEF1 0x52b7**

Bit(s)	R/W	Default	Description
29:20	R/W	0d32	chroma's vertical adaptive coef0 . signed , default = 32
19:10	R/W	0d32	chroma's vertical adaptive coef1 . signed , default = 32
9:0	R/W	0d16	chroma's vertical adaptive coef2 . signed , default = 16

**Table 9-2168 SRSHARP1\_DB\_FLT\_CTRL1 0x52b8**

Bit(s)	R/W	Default	Description
17:16	R/W	0d2	reg_nrdeband_noise_rs .unsigned , default = 2
15:12	R/W	0d8	reg_nrdeband_randgain .unsigned , default = 8

**Table 9-2169 SRSHARP1\_DB\_FLT\_LUMA\_THRD 0x52b9**

Bit(s)	R/W	Default	Description
29:24	R/W	36	reg_nrdeband_luma_th3 : // unsigned , default = 36 threshold to  Y-Y  <sub>lpf</sub> , if < th[0] use lpf
21:16	R/W	28	reg_nrdeband_luma_th2 : // unsigned , default = 28 elseif <th[1] use (lpf*3 + y)/4
13:8	R/W	24	reg_nrdeband_luma_th1 : // unsigned , default = 24 elseif <th[1] use (lpf*3 + y)/4elseif <th[2] (lpf*1 + y)/2
5:0	R/W	20	reg_nrdeband_luma_th0 : // unsigned , default = 20 elseif <th[1] use (lpf*3 + y)/4elseif elseif <th[3] (lpf*1 + 3*y)/4; else

Table 9-2170 SRSHARP1\_DB\_FLT\_CHRM\_THRD 0x52ba

Bit(s)	R/W	Default	Description
29:24	R/W	36	reg_nrdeband_chrm_th3 : // unsigned , default = 36 threshold to $ Y-Y _{lpf}$ , if $< th[0]$ use lpf
21:16	R/W	28	reg_nrdeband_chrm_th2 : // unsigned , default = 28 elseif $< th[1]$ use $(lpf^3 + y)/4$
13:8	R/W	24	reg_nrdeband_chrm_th1 : // unsigned , default = 24 elseif $< th[1]$ use $(lpf^3 + y)/4$ elseif $< th[2]$ $(lpf^3 + y)/2$
5:0	R/W	20	reg_nrdeband_chrm_th0 : // unsigned , default = 20 elseif $< th[1]$ use $(lpf^3 + y)/4$ elseif elseif $< th[3]$ $(lpf^3 + 3*y)/4$ ; else

Table 9-2171 SRSHARP1\_LC\_TOP\_CTRL 0x52c0

Bit(s)	R/W	Default	Description
31:17	R/W	0	reserved
16	R/W	0	reg_lc_sync_ctrl: 0 to shadow reg_lc_enable.
15:8	R/W	8	reg_lc_hblank: blank between two line.
7:5	R/W	0	reserved
4	R/W	0	reg_lc_enable: enable signal for local contrast enhancement, 1-enable; 0 disable
3:1	R/W	0	reserved
0	R/W	0	reg_lc_blkblend_mode:use bilinear interpolation between blocks, 0: no interpolation 1: blender enabled

Table 9-2172 SRSHARP1\_LC\_HV\_NUM 0x52c1

Bit(s)	R/W	Default	Description
31:13	R/W	0	reserved
12:8	R/W	12	reg_lc_blk_hnum: lc processing region number of V, maximum to (STA_LEN_H-1) (0~12)
7:5	R/W	0	reserved
4:0	R/W	8	reg_lc_blk_vnum: lc processing region number of H, maximum to (STA_LEN_V-1) (0~8)

Table 9-2173 SRSHARP1\_LC\_SAT\_LUT\_0\_1 0x52c2

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	12	reg_lc_satur_lut_0: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	32	reg_lc_satur_lut_1: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2174 SRSHARP1\_LC\_SAT\_LUT\_2\_3 0x52c3**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	56	reg_lc_satur_lut_2: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	84	reg_lc_satur_lut_3: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2175 SRSHARP1\_LC\_SAT\_LUT\_4\_5 0x52c4**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	115	reg_lc_satur_lut_4: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	149	reg_lc_satur_lut_5: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2176 SRSHARP1\_LC\_SAT\_LUT\_6\_7 0x52c5**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	185	reg_lc_satur_lut_6: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	223	reg_lc_satur_lut_7: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2177 SRSHARP1\_LC\_SAT\_LUT\_8\_9 0x52c6**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	263	reg_lc_satur_lut_8: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	305	reg_lc_satur_lut_9: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2178 SRSHARP1\_LC\_SAT\_LUT\_10\_11 0x52c7**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	348	reg_lc_satur_lut_10: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

Bit(s)	R/W	Default	Description
15:12	R/W	0	reserved
11:0	R/W	393	reg_lc_satur_lut_11: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

Table 9-2179 SRSHARP1\_LC\_SAT\_LUT\_12\_13 0x52c8

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	440	reg_lc_satur_lut_12: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	488	reg_lc_satur_lut_13: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

Table 9-2180 SRSHARP1\_LC\_SAT\_LUT\_14\_15 0x52c9

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	537	reg_lc_satur_lut_14: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	588	reg_lc_satur_lut_15: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

Table 9-2181 SRSHARP1\_LC\_SAT\_LUT\_16\_17 0x52ca

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	640	reg_lc_satur_lut_16: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	694	reg_lc_satur_lut_17: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

Table 9-2182 SRSHARP1\_LC\_SAT\_LUT\_18\_19 0x52cb

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	748	reg_lc_satur_lut_18: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	804	reg_lc_satur_lut_19: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2183 SRSHARP1\_LC\_SAT\_LUT\_20\_21 0x52cc**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	861	reg_lc_satur_lut_20: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	919	reg_lc_satur_lut_21: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2184 SRSHARP1\_LC\_SAT\_LUT\_22\_23 0x52cd**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	978	reg_lc_satur_lut_22: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	1038	reg_lc_satur_lut_23: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2185 SRSHARP1\_LC\_SAT\_LUT\_24\_25 0x52ce**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	1099	reg_lc_satur_lut_24: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	1161	reg_lc_satur_lut_25: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2186 SRSHARP1\_LC\_SAT\_LUT\_26\_27 0x52cf**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	1224	reg_lc_satur_lut_26: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	1287	reg_lc_satur_lut_27: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2187 SRSHARP1\_LC\_SAT\_LUT\_28\_29 0x52d0**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	1352	reg_lc_satur_lut_28: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$



Bit(s)	R/W	Default	Description
15:12	R/W	0	reserved
11:0	R/W	1418	reg_lc_satur_lut_29: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2188 SRSHARP1\_LC\_SAT\_LUT\_30\_31 0x52d1**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	1485	reg_lc_satur_lut_30: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	1552	reg_lc_satur_lut_31: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2189 SRSHARP1\_LC\_SAT\_LUT\_32\_33 0x52d2**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	1620	reg_lc_satur_lut_32: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	1690	reg_lc_satur_lut_33: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2190 SRSHARP1\_LC\_SAT\_LUT\_34\_35 0x52d3**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	1760	reg_lc_satur_lut_34: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	1830	reg_lc_satur_lut_35: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2191 SRSHARP1\_LC\_SAT\_LUT\_36\_37 0x52d4**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	1902	reg_lc_satur_lut_36: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	1974	reg_lc_satur_lut_37: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2192 SRSHARP1\_LC\_SAT\_LUT\_38\_39 0x52d5**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	2047	reg_lc_satur_lut_38: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	2121	reg_lc_satur_lut_39: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2193 SRSHARP1\_LC\_SAT\_LUT\_40\_41 0x52d6**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	2196	reg_lc_satur_lut_40: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	2271	reg_lc_satur_lut_41: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2194 SRSHARP1\_LC\_SAT\_LUT\_42\_43 0x52d7**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	2347	reg_lc_satur_lut_42: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	2424	reg_lc_satur_lut_43: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2195 SRSHARP1\_LC\_SAT\_LUT\_44\_45 0x52d8**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	2502	reg_lc_satur_lut_44: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	2580	reg_lc_satur_lut_45: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2196 SRSHARP1\_LC\_SAT\_LUT\_46\_47 0x52d9**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	2659	reg_lc_satur_lut_46: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

Bit(s)	R/W	Default	Description
15:12	R/W	0	reserved
11:0	R/W	2738	reg_lc_satur_lut_47: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

Table 9-2197 SRSHARP1\_LC\_SAT\_LUT\_48\_49 0x52da

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	2818	reg_lc_satur_lut_48: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	2899	reg_lc_satur_lut_49: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

Table 9-2198 SRSHARP1\_LC\_SAT\_LUT\_50\_51 0x52db

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	2981	reg_lc_satur_lut_50: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	3063	reg_lc_satur_lut_51: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

Table 9-2199 SRSHARP1\_LC\_SAT\_LUT\_52\_53 0x52dc

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	3146	reg_lc_satur_lut_52: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	3229	reg_lc_satur_lut_53: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

Table 9-2200 SRSHARP1\_LC\_SAT\_LUT\_54\_55 0x52dd

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	3313	reg_lc_satur_lut_54: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	3398	reg_lc_satur_lut_55: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2201 SRSHARP1\_LC\_SAT\_LUT\_56\_57 0x52de**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	3483	reg_lc_satur_lut_56: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	3569	reg_lc_satur_lut_57: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2202 SRSHARP1\_LC\_SAT\_LUT\_58\_59 0x52df**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	3655	reg_lc_satur_lut_58: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	3742	reg_lc_satur_lut_59: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2203 SRSHARP1\_LC\_SAT\_LUT\_60\_61 0x52e0**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	3830	reg_lc_satur_lut_60: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$
15:12	R/W	0	reserved
11:0	R/W	3918	reg_lc_satur_lut_61: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2204 SRSHARP1\_LC\_SAT\_LUT\_62 0x52e1**

Bit(s)	R/W	Default	Description
31:12	R/W	0	reserved
11:0	R/W	4007	reg_lc_satur_lut_62: saturation protection curve, normalized to 4096 as 1, default = $y=x^{1.2}$

**Table 9-2205 SRSHARP1\_LC\_CURVE\_BLK\_HIDX\_0\_1 0x52e2**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:16	R/W	0	reg_lc_blk_hidx_0: block boundary x-index
15:14	R/W	0	reserved
13:0	R/W	320	reg_lc_blk_hidx_1: block boundary x-index

**Table 9-2206 SRSHARP1\_LC\_CURVE\_BLK\_HIDX\_2\_3 0x52e3**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:16	R/W	640	reg_lc_blk_hidx_2: block boundary x-index
15:14	R/W	0	reserved
13:0	R/W	960	reg_lc_blk_hidx_3: block boundary x-index

**Table 9-2207 SRSHARP1\_LC\_CURVE\_BLK\_HIDX\_4\_5 0x52e4**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:16	R/W	1280	reg_lc_blk_hidx_4: block boundary x-index
15:14	R/W	0	reserved
13:0	R/W	1600	reg_lc_blk_hidx_5: block boundary x-index

**Table 9-2208 SRSHARP1\_LC\_CURVE\_BLK\_HIDX\_6\_7 0x52e5**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:16	R/W	1920	reg_lc_blk_hidx_6: block boundary x-index
15:14	R/W	0	reserved
13:0	R/W	2240	reg_lc_blk_hidx_7: block boundary x-index

**Table 9-2209 SRSHARP1\_LC\_CURVE\_BLK\_HIDX\_8\_9 0x52e6**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:16	R/W	2560	reg_lc_blk_hidx_8: block boundary x-index
15:14	R/W	0	reserved
13:0	R/W	2880	reg_lc_blk_hidx_9: block boundary x-index

**Table 9-2210 SRSHARP1\_LC\_CURVE\_BLK\_HIDX\_10\_11 0x52e7**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:16	R/W	3200	reg_lc_blk_hidx_10: block boundary x-index
15:14	R/W	0	reserved
13:0	R/W	3520	reg_lc_blk_hidx_11: block boundary x-index

**Table 9-2211 SRSHARP1\_LC\_CURVE\_BLK\_HIDX\_12 0x52e8**

Bit(s)	R/W	Default	Description
31:14	R/W	0	reserved
13:0	R/W	3840	reg_lc_blk_hidx_1: block boundary x-index

**Table 9-2212 SRSHARP1\_LC\_CURVE\_BLK\_VIDX\_0\_1 0x52e9**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:16	R/W	0	reg_lc_blk_vidx_0: block boundary y-index
15:14	R/W	0	reserved
13:0	R/W	270	reg_lc_blk_vidx_1: block boundary y-index

**Table 9-2213 SRSHARP1\_LC\_CURVE\_BLK\_VIDX\_2\_3 0x52ea**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:16	R/W	540	reg_lc_blk_vidx_2: block boundary y-index
15:14	R/W	0	reserved
13:0	R/W	810	reg_lc_blk_vidx_3: block boundary y-index

**Table 9-2214 SRSHARP1\_LC\_CURVE\_BLK\_VIDX\_4\_5 0x52eb**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:16	R/W	1080	reg_lc_blk_vidx_4: block boundary y-index
15:14	R/W	0	reserved
13:0	R/W	1350	reg_lc_blk_vidx_5: block boundary y-index

**Table 9-2215 SRSHARP1\_LC\_CURVE\_BLK\_VIDX\_6\_7 0x52ec**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:16	R/W	1620	reg_lc_blk_vidx_6: block boundary y-index
15:14	R/W	0	reserved
13:0	R/W	1890	reg_lc_blk_vidx_7: block boundary y-index

**Table 9-2216 SRSHARP1\_LC\_CURVE\_BLK\_HIDX\_8 0x52ed**

Bit(s)	R/W	Default	Description
31:14	R/W	0	reserved
13:0	R/W	2160	reg_lc_blk_vidx_8: block boundary y-index

**Table 9-2217 SRSHARP1\_LC\_YUV2RGB\_MAT\_0\_1 0x52ee**

Bit(s)	R/W	Default	Description
31:29	R/W	0	reserved
28:16	R/W	1192	reg_lc_yuv2rgb_mat3x3_0: yuv2rgb 3x3 matrix
15:13	R/W	0	reserved
12:0	R/W	0	reg_lc_yuv2rgb_mat3x3_1: yuv2rgb 3x3 matrix

**Table 9-2218 SRSHARP1\_LC\_YUV2RGB\_MAT\_2\_3 0x52ef**

Bit(s)	R/W	Default	Description
31:29	R/W	0	reserved
28:16	R/W	1836	reg_lc_yuv2rgb_mat3x3_2: yuv2rgb 3x3 matrix
15:13	R/W	0	reserved
12:0	R/W	1192	reg_lc_yuv2rgb_mat3x3_3: yuv2rgb 3x3 matrix

**Table 9-2219 SRSHARP1\_LC\_YUV2RGB\_MAT\_4\_5 0x52f0**

Bit(s)	R/W	Default	Description
31:29	R/W	0	reserved
28:16	R/W	-218	reg_lc_yuv2rgb_mat3x3_4: yuv2rgb 3x3 matrix
15:13	R/W	0	reserved
12:0	R/W	-514	reg_lc_yuv2rgb_mat3x3_5: yuv2rgb 3x3 matrix

**Table 9-2220 SRSHARP1\_LC\_YUV2RGB\_MAT\_6\_7 0x52f1**

Bit(s)	R/W	Default	Description
31:29	R/W	0	reserved
28:16	R/W	1192	reg_lc_yuv2rgb_mat3x3_6: yuv2rgb 3x3 matrix
15:13	R/W	0	reserved
12:0	R/W	2166	reg_lc_yuv2rgb_mat3x3_7: yuv2rgb 3x3 matrix

**Table 9-2221 SRSHARP1\_LC\_YUV2RGB\_MAT\_8 0x52f2**

Bit(s)	R/W	Default	Description
31:13	R/W	0	reserved
12:0	R/W	0	reg_lc_yuv2rgb_mat3x3_8: yuv2rgb 3x3 matrix

**Table 9-2222 SRSHARP1\_LC\_RGB2YUV\_MAT\_0\_1 0x52f3**

Bit(s)	R/W	Default	Description
31:29	R/W	0	reserved
28:16	R/W	187	reg_lc_rgb2yuv_mat3x3_0: rgb2yuv 3x3 matrix
15:13	R/W	0	reserved
12:0	R/W	629	reg_lc_rgb2yuv_mat3x3_1: rgb2yuv 3x3 matrix

**Table 9-2223 SRSHARP1\_LC\_RGB2YUV\_MAT\_2\_3 0x52f4**

Bit(s)	R/W	Default	Description
31:29	R/W	0	reserved
28:16	R/W	63	reg_lc_rgb2yuv_mat3x3_2: rgb2yuv 3x3 matrix
15:13	R/W	0	reserved
12:0	R/W	-103	reg_lc_rgb2yuv_mat3x3_3: rgb2yuv 3x3 matrix

**Table 9-2224 SRSHARP1\_LC\_RGB2YUV\_MAT\_4\_5 0x52f5**

Bit(s)	R/W	Default	Description
31:29	R/W	0	reserved
28:16	R/W	-346	reg_lc_rgb2yuv_mat3x3_4: rgb2yuv 3x3 matrix
15:13	R/W	0	reserved
12:0	R/W	450	reg_lc_rgb2yuv_mat3x3_5: rgb2yuv 3x3 matrix

**Table 9-2225 SRSHARP1\_LC\_RGB2YUV\_MAT\_6\_7 0x52f6**

Bit(s)	R/W	Default	Description
31:29	R/W	0	reserved
28:16	R/W	450	reg_lc_rgb2yuv_mat3x3_6: rgb2yuv 3x3 matrix
15:13	R/W	0	reserved
12:0	R/W	-409	reg_lc_rgb2yuv_mat3x3_7: rgb2yuv 3x3 matrix

**Table 9-2226 SRSHARP1\_LC\_RGB2YUV\_MAT\_8 0x52f7**

Bit(s)	R/W	Default	Description
31:13	R/W	0	reserved
12:0	R/W	41	reg_lc_rgb2yuv_mat3x3_8: rgb2yuv 3x3 matrix

**Table 9-2227 SRSHARP1\_LC\_YUV2RGB\_OFST 0x52f8**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	64	reg_lc_yuv2rgb_offset_0: yuv2rgb pre-offset to yuv



Bit(s)	R/W	Default	Description
15:12	R/W	0	reserved
11:0	R/W	512	reg_lc_yuv2rgb_offset_1: yuv2rgb pre-offset to yuv

**Table 9-2228 SRSHARP1\_LC\_YUV2RGB\_CLIP 0x52f9**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:28	R/W	0	reg_lc_yuv2rgb_rs: matrix normalization right shift extra bits, norm= (1<<(8+rs))
27:16	R/W	0	reg_lc_yuv2rgb_clip_0: yuv2rgb converted RGB clipping range
15:12	R/W	0	reserved
11:0	R/W	1023	reg_lc_yuv2rgb_clip_1: yuv2rgb converted RGB clipping range

**Table 9-2229 SRSHARP1\_LC\_RGB2YUV\_OFST 0x52fa**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:16	R/W	64	reg_lc_rgb2yuv_offset_0: rgb2yuv post-offset to yuv
15:12	R/W	0	reserved
11:0	R/W	512	reg_lc_rgb2yuv_offset_1: rgb2yuv post-offset to yuv

**Table 9-2230 SRSHARP1\_LC\_RGB2YUV\_CLIP 0x52fb**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:28	R/W	0	reg_lc_rgb2yuv_rs: matrix normalization right shift extra bits, norm= (1<<(10+rs))
27:16	R/W	0	reg_lc_rgb2yuv_clip_0: rgb2yuv converted YUV clipping range
15:12	R/W	0	reserved
11:0	R/W	1023	reg_lc_rgb2yuv_clip_1: rgb2yuv converted YUV clipping range

**Table 9-2231 SRSHARP1\_LC\_MAP\_RAM\_CTRL 0x52fc**

Bit(s)	R/W	Default	Description
31:1	R/W	0	reserved
0	R/W	0	reg_lc_cbus2ram_en: 1 to enable CBUS write/read data to/from LC MAP RAM

**Table 9-2232 SRSHARP1\_LC\_MAP\_RAM\_ADDR 0x52fd**

Bit(s)	R/W	Default	Description
31	W	0	1 to pre-read from LC MAP RAM to CBUS
30:7	R/W	0	reserved

Bit(s)	R/W	Default	Description
6:4	R/W	0	Vidx of LC MAP RAM
3:0	R/W	0	Hidx of LC MAP RA:

Table 9-2233 SRSHARP1\_LC\_MAP\_RAM\_DATA 0x52fe

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:0	R/W	0	LC MAP RAM data for write/read

Table 9-2234 SRSHARP1\_FMETER\_CTRL 0x5289

Bit(s)	R/W	Default	Description
11:10	R/W	0	reg_fmeter_vwin_mm : vertical window size, 0:1 cloumn,1:3cloumn, 2or3:5cloumn.unsigned,default = 0
9:8	R/W	0	reg_fmeter_hwin_mm : horizontal window size, 0:1x7, 1:1x9, 2or3: 1x11 .unsigned , default = 0
7	R/W	0	reg_fmeter_d2_mode : selectino filter D2, 0: [0 -2 0 0 2], 1: [-2 0 0 0 2] .unsigned , default = 0
6	R/W	0	reg_fmeter_v2_mode : selection filter V2, 0: [0 -2 0 0 2], 1: [-2 0 0 0 2] .unsigned , default = 0
5:4	R/W	0	reg_fmeter_h2_mode: selection filter H2, 0: [0 0 0 -2 0 0 2 0 0], 1: [-2 0 0 0 2], 2or3: [0-2 0 0 0 0 2 0] .unsigned , default = 0
0	R/W	0	reg_freq_meter_en: freq meter enable .unsigned , default = 0

Table 9-2235 SRSHARP1\_FMETER\_WIN\_HOR 0x528a

Bit(s)	R/W	Default	Description
28:16	R/W	1920	reg_fmeter_xwin_ed: window location: hend .unsigned , default = 1920
12:0	R/W	0	reg_fmeter_xwin_st: window location: hstart .unsigned , default = 0

Table 9-2236 SRSHARP1\_FMETER\_WIN\_VER 0x528b

Bit(s)	R/W	Default	Description
28:16	R/W	1080	reg_fmeter_ywin_ed: window location: vend .unsigned , default = 1080
12:0	R/W	0	reg_fmeter_ywin_st: window location: vstart .unsigned , default = 0

Table 9-2237 SRSHARP1\_FMETER\_CORING 0x528c

Bit(s)	R/W	Default	Description
31:24	R/W	4	reg_fmeter_low_bound: low bound for threshold .unsigned , default = 4
23:16	R/W	4	reg_fmeter_coring_d: coring of diff before compare with threshold for diagonal frequency .unsigned , default = 4

Bit(s)	R/W	Default	Description
15:8	R/W	4	reg_fmeter_coring_v: coring of diff before compare with threshold for vertical frequency .unsigned , default = 4
7:0	R/W	4	reg_fmeter_coring_h: coring of diff before compare with threshold for horizontal frequency .unsigned , default = 4

Table 9-2238 SRSHARP1\_FMETER\_RATIO\_H 0x528d

Bit(s)	R/W	Default	Description
21:16	R/W	16	reg_fmeter_ratio_h2: ratio for threshold calc. for horizontal frequency, 16 is normalized as "1" .unsigned , default = 16
13:8	R/W	16	reg_fmeter_ratio_h1: ratio for threshold calc. for horizontal frequency, 16 is normalized as "1" .unsigned , default = 16
5:0	R/W	16	reg_fmeter_ratio_h0: ratio for threshold calc. for horizontal frequency, 16 is normalized as "1" .unsigned , default = 16

Table 9-2239 SRSHARP1\_FMETER\_RATIO\_V 0x528e

Bit(s)	R/W	Default	Description
21:16	R/W	16	reg_fmeter_ratio_v2: ratio for threshold calc. for vertical frequency, 16 is normalized as "1" .unsigned , default = 16
13:8	R/W	16	reg_fmeter_ratio_v1: ratio for threshold calc. for vertical frequency, 16 is normalized as "1" .unsigned , default = 16
5:0	R/W	16	reg_fmeter_ratio_v0: ratio for threshold calc. for vertical frequency, 16 is normalized as "1" .unsigned , default = 16

Table 9-2240 SRSHARP1\_FMETER\_RATIO\_D 0x528f

Bit(s)	R/W	Default	Description
21:16	R/W	16	reg_fmeter_ratio_d2: ratio for threshold calc. for diagonal frequency, 16 is normalized as "1" .unsigned , default = 16
13:8	R/W	16	reg_fmeter_ratio_d1: ratio for threshold calc. for diagonal frequency, 16 is normalized as "1" .unsigned , default = 16
5:0	R/W	16	reg_fmeter_ratio_d0: ratio for threshold calc. for diagonal frequency, 16 is normalized as "1" .unsigned , default = 16

Table 9-2241 SRSHARP1\_RO\_FMETER\_HCNT\_TYPE0 0x5246

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_hcnt_type0: count for horizontal frequency

Table 9-2242 SRSHARP1\_RO\_FMETER\_HCNT\_TYPE1 0x5247

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_hcnt_type1: count for horizontal frequency

**Table 9-2243 SRSHARP1\_RO\_FMETER\_HCNT\_TYPE2 0x5248**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_hcnt_type2: count for horizontal frequency

**Table 9-2244 SRSHARP1\_RO\_FMETER\_HCNT\_TYPE3 0x5249**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_hcnt_type3: count for horizontal frequency

**Table 9-2245 SRSHARP1\_RO\_FMETER\_VCNT\_TYPE0 0x524a**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_vcnt_type0: count for vertical frequency

**Table 9-2246 SRSHARP1\_RO\_FMETER\_VCNT\_TYPE1 0x524b**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_vcnt_type1: count for vertical frequency

**Table 9-2247 SRSHARP1\_RO\_FMETER\_VCNT\_TYPE2 0x524c**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_vcnt_type2: count for vertical frequency

**Table 9-2248 SRSHARP1\_RO\_FMETER\_VCNT\_TYPE3 0x524d**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_vcnt_type3: count for vertical frequency

**Table 9-2249 SRSHARP1\_RO\_FMETER\_PDCNT\_TYPE0 0x524e**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_pdcnt_type0: count for positive diagonal frequency

**Table 9-2250 SRSHARP1\_RO\_FMETER\_PDCNT\_TYPE1 0x524f**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_pdcnt_type1: count for positive diagonal frequency

**Table 9-2251 SRSHARP1\_RO\_FMETER\_PDCNT\_TYPE2 0x5250**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_pdcnt_type2: count for positive diagonal frequency

**Table 9-2252 SRSHARP1\_RO\_FMETER\_PDCNT\_TYPE3 0x5251**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_pdcnt_type3: count for positive diagonal frequency

**Table 9-2253 SRSHARP1\_RO\_FMETER\_NDCNT\_TYPE0 0x5252**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_ndcnt_type0: count for negative diagonal frequency

**Table 9-2254 SRSHARP1\_RO\_FMETER\_NDCNT\_TYPE1 0x5253**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_ndcnt_type1: count for negative diagonal frequency

**Table 9-2255 SRSHARP1\_RO\_FMETER\_NDCNT\_TYPE2 0x5254**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_ndcnt_type2: count for negative diagonal frequency

**Table 9-2256 SRSHARP1\_RO\_FMETER\_NDCNT\_TYPE3 0x5255**

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_fmeter_ndcnt_type3: count for negative diagonal frequency

**Table 9-2257 SRSHARP1\_SR7\_DRTLPF\_EN 0x5300**

Bit(s)	R/W	Default	Description
31:11			reserved
10	RW	0	reg_sr7_drtlpf_beta_en2: enable of direction ambiguity protection for drt_lpf, beta for drt filter coef base on the x=cal_drt_dif8(min_idx,min2_idx)
9	RW	0	reg_sr7_drtlpf_beta_en1: enable of direction ambiguity protection for drt_lpf, beta for drt filter coef base on the x=cal_drt_dif8(min_idx,min2_idx)
8	RW	0	reg_sr7_drtlpf_beta_en0: enable of direction ambiguity protection for drt_lpf, beta for drt filter coef base on the x=cal_drt_dif8(min_idx,min2_idx)
7:6			reserved
5	RW	0	reg_sr7_drtlpf_edge_en2: enable of direction lpf based on edge strength
4	RW	0	reg_sr7_drtlpf_edge_en1: enable of direction lpf based on edge strength
3	RW	0	reg_sr7_drtlpf_edge_en0: enable of direction lpf based on edge strength
2	RW	1	reg_sr7_drtlpf_sdfd_en2: sdfd gamma (HF burst compare to real edge protection) enable

Bit(s)	R/W	Default	Description
1	RW	1	reg_sr7_drtlpf_sdfd_en1: sdfd gamma (HF burst compare to real edge protection) enable.
0	RW	1	reg_sr7_drtlpf_sdfd_en0: sdfd gamma (HF burst compare to real edge protection) enable.

Table 9-2258 SRSHARP1\_SR7\_DRTLPF\_BETA 0x5301

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	0	reg_sr7_drtlpf_beta3: beta for drt filter coef base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$
11:8	RW	4	reg_sr7_drtlpf_beta2: beta for drt filter coef base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$
7:4	RW	8	reg_sr7_drtlpf_beta1: beta for drt filter coef base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$
3:0	RW	15	reg_sr7_drtlpf_beta0: beta for drt filter coef base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$ , $\beta = lut[x-1]$ , the larger of x, means the higher possibility for ambiguity, $\beta=0$ use org wo lpf.

Table 9-2259 SRSHARP1\_SR7\_PKBLD\_BETA 0x5302

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	0	reg_sr7_pkdrtbl_beta3: beta for drt and cir blend base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$
11:8	RW	4	reg_sr7_pkdrtbl_beta2: beta for drt and cir blend base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$
7:4	RW	8	reg_sr7_pkdrtbl_beta1: beta for drt and cir blend base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$
3:0	RW	15	reg_sr7_pkdrtbl_beta0: beta for drt and cir blend base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$ , $\beta = lut[x-1]$ , the larger of x, means the higher possibility for ambiguity, $\beta=0$ use org wo lpf.

Table 9-2260 SRSHARP1\_SR7\_XLTIBLD\_BETA 0x5303

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	0	reg_sr7_xlti_dcbl_beta3: beta for denoise and org dc blend base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$
11:8	RW	4	reg_sr7_xlti_dcbl_beta2: beta for denoise and org dc blend base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$
7:4	RW	8	reg_sr7_xlti_dcbl_beta1: beta for denoise and org dc blend base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$
3:0	RW	15	reg_sr7_xlti_dcbl_beta0: beta for denoise and org dc blend base on the $x=cal\_drt\_dif8(min\_idx, min2\_idx)$ , $\beta = lut[x-1]$ , the larger of x, means the higher possibility for ambiguity, $\beta=0$ use org wo lpf.

**Table 9-2261 SRSHARP1\_SR7\_DRTLPF\_EDGE0 0x5304**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drtlpf_edge7: edge lamda for drt lpf base on max_sad [0:16:128~255]
27:24	RW	15	reg_sr7_drtlpf_edge6: edge lamda for drt lpf base on max_sad [0:16:128~255]
23:20	RW	15	reg_sr7_drtlpf_edge5: edge lamda for drt lpf base on max_sad [0:16:128~255]
19:16	RW	15	reg_sr7_drtlpf_edge4: edge lamda for drt lpf base on max_sad [0:16:128~255]
15:12	RW	15	reg_sr7_drtlpf_edge3: edge lamda for drt lpf base on max_sad [0:16:128~255]
11:8	RW	15	reg_sr7_drtlpf_edge2: edge lamda for drt lpf base on max_sad [0:16:128~255]
7:4	RW	15	reg_sr7_drtlpf_edge1: edge lamda for drt lpf base on max_sad [0:16:128~255]
3:0	RW	15	reg_sr7_drtlpf_edge0: edge lamda for drt lpf base on max_sad [0:16:128~255]

**Table 9-2262 SRSHARP1\_SR7\_DRTLPF\_EDGE1 0x5305**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drtlpf_edge15: edge lamda for drt lpf base on max_sad [0:16:128~255]
27:24	RW	15	reg_sr7_drtlpf_edge14: edge lamda for drt lpf base on max_sad [0:16:128~255]
23:20	RW	15	reg_sr7_drtlpf_edge13: edge lamda for drt lpf base on max_sad [0:16:128~255]
19:16	RW	15	reg_sr7_drtlpf_edge12: edge lamda for drt lpf base on max_sad [0:16:128~255]
15:12	RW	15	reg_sr7_drtlpf_edge11: edge lamda for drt lpf base on max_sad [0:16:128~255]
11:8	RW	15	reg_sr7_drtlpf_edge10: edge lamda for drt lpf base on max_sad [0:16:128~255]
7:4	RW	15	reg_sr7_drtlpf_edge9: edge lamda for drt lpf base on max_sad [0:16:128~255]
3:0	RW	15	reg_sr7_drtlpf_edge8: edge lamda for drt lpf base on max_sad [0:16:128~255]

**Table 9-2263 SRSHARP1\_SR7\_DRTLPF\_SDCOR0 0x5306**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drtlpf_sdcor7: gamma for drt lpf base on SD'[0:16:128~255]
27:24	RW	15	reg_sr7_drtlpf_sdcor6: gamma for drt lpf base on SD'[0:16:128~255]
23:20	RW	15	reg_sr7_drtlpf_sdcor5: gamma for drt lpf base on SD'[0:16:128~255]
19:16	RW	15	reg_sr7_drtlpf_sdcor4: gamma for drt lpf base on SD'[0:16:128~255]
15:12	RW	15	reg_sr7_drtlpf_sdcor3: gamma for drt lpf base on SD'[0:16:128~255]
11:8	RW	15	reg_sr7_drtlpf_sdcor2: gamma for drt lpf base on SD'[0:16:128~255]
7:4	RW	15	reg_sr7_drtlpf_sdcor1: gamma for drt lpf base on SD'[0:16:128~255]
3:0	RW	15	reg_sr7_drtlpf_sdcor0: gamma for drt lpf base on SD'[0:16:128~255]

**Table 9-2264 SRSHARP1\_SR7\_DRTLPF\_SDCOR1 0x5307**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drtlpf_sdcor15: gamma for drt lpf base on SD'[0:16:128~255]
27:24	RW	15	reg_sr7_drtlpf_sdcor14: gamma for drt lpf base on SD'[0:16:128~255]
23:20	RW	15	reg_sr7_drtlpf_sdcor13: gamma for drt lpf base on SD'[0:16:128~255]
19:16	RW	15	reg_sr7_drtlpf_sdcor12: gamma for drt lpf base on SD'[0:16:128~255]
15:12	RW	15	reg_sr7_drtlpf_sdcor11: gamma for drt lpf base on SD'[0:16:128~255]
11:8	RW	15	reg_sr7_drtlpf_sdcor10: gamma for drt lpf base on SD'[0:16:128~255]
7:4	RW	15	reg_sr7_drtlpf_sdcor9: gamma for drt lpf base on SD'[0:16:128~255]
3:0	RW	15	reg_sr7_drtlpf_sdcor8: gamma for drt lpf base on SD'[0:16:128~255]

**Table 9-2265 SRSHARP1\_SR7\_CTIGAIN\_SDCOR0 0x5308**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_ctigain_sdcor7: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
27:24	RW	15	reg_sr7_ctigain_sdcor6: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
23:20	RW	15	reg_sr7_ctigain_sdcor5: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
19:16	RW	15	reg_sr7_ctigain_sdcor4: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
15:12	RW	15	reg_sr7_ctigain_sdcor3: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
11:8	RW	15	reg_sr7_ctigain_sdcor2: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
7:4	RW	15	reg_sr7_ctigain_sdcor1: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
3:0	RW	15	reg_sr7_ctigain_sdcor0: adaptive gamma for cti boost gain base on SD'[0:16:128~255]

**Table 9-2266 SRSHARP1\_SR7\_CTIGAIN\_SDCOR1 0x5309**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_ctigain_sdcor15: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
27:24	RW	15	reg_sr7_ctigain_sdcor14: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
23:20	RW	15	reg_sr7_ctigain_sdcor13: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
19:16	RW	15	reg_sr7_ctigain_sdcor12: adaptive gamma for cti boost gain base on SD'[0:16:128~255]
15:12	RW	15	reg_sr7_ctigain_sdcor11: adaptive gamma for cti boost gain base on SD'[0:16:128~255]



Bit(s)	R/W	Default	Description
11:8	RW	15	reg_sr7_ctigain_sdcor10: adaptive gamma for cti boost gain base on SD[0:16:128~255]
7:4	RW	15	reg_sr7_ctigain_sdcor9: adaptive gamma for cti boost gain base on SD[0:16:128~255]
3:0	RW	15	reg_sr7_ctigain_sdcor8: adaptive gamma for cti boost gain base on SD[0:16:128~255]

Table 9-2267 SRSHARP1\_SR7\_LTIGAIN\_SDCOR0 0x530a

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_ltigain_sdcor7: adaptive gamma for lti boost gain base on SD[0:16:128~255]
27:24	RW	15	reg_sr7_ltigain_sdcor6: adaptive gamma for lti boost gain base on SD[0:16:128~255]
23:20	RW	15	reg_sr7_ltigain_sdcor5: adaptive gamma for lti boost gain base on SD[0:16:128~255]
19:16	RW	15	reg_sr7_ltigain_sdcor4: adaptive gamma for lti boost gain base on SD[0:16:128~255]
15:12	RW	15	reg_sr7_ltigain_sdcor3: adaptive gamma for lti boost gain base on SD[0:16:128~255]
11:8	RW	15	reg_sr7_ltigain_sdcor2: adaptive gamma for lti boost gain base on SD[0:16:128~255]
7:4	RW	15	reg_sr7_ltigain_sdcor1: adaptive gamma for lti boost gain base on SD[0:16:128~255]
3:0	RW	15	reg_sr7_ltigain_sdcor0: adaptive gamma for lti boost gain base on SD[0:16:128~255]

Table 9-2268 SRSHARP1\_SR7\_LTIGAIN\_SDCOR1 0x530b

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_ltigain_sdcor15: adaptive gamma for lti boost gain base on SD[0:16:128~255]
27:24	RW	15	reg_sr7_ltigain_sdcor14: adaptive gamma for lti boost gain base on SD[0:16:128~255]
23:20	RW	15	reg_sr7_ltigain_sdcor13: adaptive gamma for lti boost gain base on SD[0:16:128~255]
19:16	RW	15	reg_sr7_ltigain_sdcor12: adaptive gamma for lti boost gain base on SD[0:16:128~255]
15:12	RW	15	reg_sr7_ltigain_sdcor11: adaptive gamma for lti boost gain base on SD[0:16:128~255]
11:8	RW	15	reg_sr7_ltigain_sdcor10: adaptive gamma for lti boost gain base on SD[0:16:128~255]
7:4	RW	15	reg_sr7_ltigain_sdcor9: adaptive gamma for lti boost gain base on SD[0:16:128~255]
3:0	RW	15	reg_sr7_ltigain_sdcor8: adaptive gamma for lti boost gain base on SD[0:16:128~255]

**Table 9-2269 SRSHARP1\_SR7\_HLTIBPF\_TAP0 0x530c**

Bit(s)	R/W	Default	Description
31:24	RW	0	reg_sr7_hlti_bpf_tap153: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
23:16	RW	-34	reg_sr7_hlti_bpf_tap152: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
15:8	RW	-1	reg_sr7_hlti_bpf_tap151: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
7:0	RW	120	reg_sr7_hlti_bpf_tap150: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed

**Table 9-2270 SRSHARP1\_SR7\_HLTIBPF\_TAP1 0x530d**

Bit(s)	R/W	Default	Description
31:24	RW	2	reg_sr7_hlti_bpf_tap157: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
23:16	RW	-7	reg_sr7_hlti_bpf_tap156: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
15:8	RW	1	reg_sr7_hlti_bpf_tap155: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed
7:0	RW	-21	reg_sr7_hlti_bpf_tap154: tap15 BPF for hlti (gain3), only store half of the filter normalized to 256 as 1.0,signed

**Table 9-2271 SRSHARP1\_SR7\_HCTIBPF\_TAP0 0x530e**

Bit(s)	R/W	Default	Description
31:24	RW	0	reg_sr7_clti_bpf_tap153: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed
23:16	RW	-34	reg_sr7_clti_bpf_tap152: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed
15:8	RW	-1	reg_sr7_clti_bpf_tap151: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed
7:0	RW	120	reg_sr7_clti_bpf_tap150: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed

**Table 9-2272 SRSHARP1\_SR7\_HCTIBPF\_TAP1 0x530f**

Bit(s)	R/W	Default	Description
31:24	RW	2	reg_sr7_clti_bpf_tap157: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed
23:16	RW	-7	reg_sr7_clti_bpf_tap156: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed

Bit(s)	R/W	Default	Description
15:8	RW	1	reg_sr7_clti_bpf_tap155: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed
7:0	RW	-21	reg_sr7_clti_bpf_tap154: tap15 BPF for clti (gain3), only store half of the filter normalized to 256 as 1.0,signed

Table 9-2273 SRSHARP1\_SR7\_PKLONGBPF\_HTAP0 0x5310

Bit(s)	R/W	Default	Description
31:24	RW	-24	reg_sr7_pk_long_bpf_hztap153: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
23:16	RW	-19	reg_sr7_pk_long_bpf_hztap152: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
15:8	RW	31	reg_sr7_pk_long_bpf_hztap151: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
7:0	RW	66	reg_sr7_pk_long_bpf_hztap150: tap15 BPF for horizontal peaking filter, only store half of the filter,signed

Table 9-2274 SRSHARP1\_SR7\_PKLONGBPF\_HTAP1 0x5311

Bit(s)	R/W	Default	Description
31:24	RW	-6	reg_sr7_pk_long_bpf_hztap157: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
23:16	RW	-10	reg_sr7_pk_long_bpf_hztap156: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
15:8	RW	-1	reg_sr7_pk_long_bpf_hztap155: tap15 BPF for horizontal peaking filter, only store half of the filter,signed
7:0	RW	-4	reg_sr7_pk_long_bpf_hztap154: tap15 BPF for horizontal peaking filter, only store half of the filter,signed

Table 9-2275 SRSHARP1\_SR7\_PKLONGHPF\_HTAP0 0x5312

Bit(s)	R/W	Default	Description
31:24	RW	27	reg_sr7_pk_long_hpf_hztap153: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
23:16	RW	-15	reg_sr7_pk_long_hpf_hztap152: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
15:8	RW	-32	reg_sr7_pk_long_hpf_hztap151: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
7:0	RW	58	reg_sr7_pk_long_hpf_hztap150: tap15 HPF for horizontal peaking filter, only store half of the filter,signed

**Table 9-2276 SRSHARP1\_SR7\_PKLONGHPF\_HTAP1 0x5313**

Bit(s)	R/W	Default	Description
31:24	RW	8	reg_sr7_pk_long_hpf_hztap157: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
23:16	RW	-6	reg_sr7_pk_long_hpf_hztap156: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
15:8	RW	-1	reg_sr7_pk_long_hpf_hztap155: tap15 HPF for horizontal peaking filter, only store half of the filter,signed
7:0	RW	-10	reg_sr7_pk_long_hpf_hztap154: tap15 HPF for horizontal peaking filter, only store half of the filter,signed

**Table 9-2277 SRSHARP1\_SR7\_VLTIBPF\_TAP0 0x5314**

Bit(s)	R/W	Default	Description
31:24	RW	0	reg_sr7_vlti_bpf_tap093: tap09 BPF for vlti (gain3), only store half of the filter, signed
23:16	RW	-41	reg_sr7_vlti_bpf_tap092: tap09 BPF for vlti (gain3), only store half of the filter, signed
15:8	RW	0	reg_sr7_vlti_bpf_tap091: tap09 BPF for vlti (gain3), only store half of the filter, signed
7:0	RW	126	reg_sr7_vlti_bpf_tap090: tap09 BPF for vlti (gain3), only store half of the filter, signed

**Table 9-2278 SRSHARP1\_SR7\_VLTIBPF\_TAP1 0x5315**

Bit(s)	R/W	Default	Description
31:8			reserved
7:0	RW	-22	reg_sr7_vlti_bpf_tap094: tap09 BPF for vlti (gain3), only store half of the filter, signed

**Table 9-2279 SRSHARP1\_SR7\_VCTIBPF\_TAP0 0x5316**

Bit(s)	R/W	Default	Description
31:24	RW	0	reg_sr7_vcti_bpf_tap093: tap09 BPF for vcti (gain3), only store half of the filter, signed
23:16	RW	-41	reg_sr7_vcti_bpf_tap092: tap09 BPF for vcti (gain3), only store half of the filter, signed
15:8	RW	0	reg_sr7_vcti_bpf_tap091: tap09 BPF for vcti (gain3), only store half of the filter, signed
7:0	RW	126	reg_sr7_vcti_bpf_tap090: tap09 BPF for vcti (gain3), only store half of the filter, signed

**Table 9-2280 SRSHARP1\_SR7\_VCTIBPF\_TAP1 0x5317**

Bit(s)	R/W	Default	Description
31:8			reserved
7:0	RW	-22	reg_sr7_vcti_bpf_tap094: tap09 BPF for vcti (gain3), only store half of the filter, signed

**Table 9-2281 SRSHARP1\_SR7\_PKLONGBPF\_VTAP0 0x5318**

Bit(s)	R/W	Default	Description
31:24	RW	-32	reg_sr7_pk_long_bpf_vdtap093: tap09 BPF for vertical and diagonal peaking filter, signed
23:16	RW	-28	reg_sr7_pk_long_bpf_vdtap092: tap09 BPF for vertical and diagonal peaking filter, signed
15:8	RW	30	reg_sr7_pk_long_bpf_vdtap091: tap09 BPF for vertical and diagonal peaking filter, signed
7:0	RW	68	reg_sr7_pk_long_bpf_vdtap090: tap09 BPF for vertical and diagonal peaking filter, signed

**Table 9-2282 SRSHARP1\_SR7\_PKLONGBPF\_VTAP1 0x5319**

Bit(s)	R/W	Default	Description
31:8			reserved
7:0	RW	-4	reg_sr7_pk_long_bpf_vdtap094: tap09 BPF for vertical and diagonal peaking filter, signed

**Table 9-2283 SRSHARP1\_SR7\_PKLONGHPF\_VTAP0 0x531a**

Bit(s)	R/W	Default	Description
31:24	RW	30	reg_sr7_pk_long_hpf_vdtap093: tap09 HPF for vertical and diagonal peaking filter, signed
23:16	RW	-28	reg_sr7_pk_long_hpf_vdtap092: tap09 HPF for vertical and diagonal peaking filter, signed
15:8	RW	-31	reg_sr7_pk_long_hpf_vdtap091: tap09 HPF for vertical and diagonal peaking filter, signed
7:0	RW	68	reg_sr7_pk_long_hpf_vdtap090: tap09 HPF for vertical and diagonal peaking filter, signed

**Table 9-2284 SRSHARP1\_SR7\_PKLONGHPF\_VTAP1 0x531b**

Bit(s)	R/W	Default	Description
31:8			reserved
7:0	RW	-5	reg_sr7_pk_long_hpf_vdtap094: tap09 HPF for vertical and diagonal peaking filter, signed

**Table 9-2285 SRSHARP1\_SR7\_CIRBPLONG\_ALP 0x531c**

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	15	reg_sr7_cirbp_long_alpha3: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
11:8	RW	15	reg_sr7_cirbp_long_alpha2: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
7:4	RW	15	reg_sr7_cirbp_long_alpha1: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
3:0	RW	15	reg_sr7_cirbp_long_alpha0: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)

**Table 9-2286 SRSHARP1\_SR7\_CIRHPLONG\_ALP 0x531d**

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	15	reg_sr7_cirhp_long_alpha3: alpha to blend hp_long to the cirpk filter corresponding angle (0/45/90/135)
11:8	RW	15	reg_sr7_cirhp_long_alpha2: alpha to blend hp_long to the cirpk filter corresponding angle (0/45/90/135)
7:4	RW	15	reg_sr7_cirhp_long_alpha1: alpha to blend hp_long to the cirpk filter corresponding angle (0/45/90/135)
3:0	RW	15	reg_sr7_cirhp_long_alpha0: alpha to blend hp_long to the cirpk filter corresponding angle (0/45/90/135)

**Table 9-2287 SRSHARP1\_SR7\_DRTBPLONG\_ALP 0x531e**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drtbp_long_alpha7: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
27:24	RW	15	reg_sr7_drtbp_long_alpha6: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
23:20	RW	15	reg_sr7_drtbp_long_alpha5: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
19:16	RW	15	reg_sr7_drtbp_long_alpha4: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
15:12	RW	15	reg_sr7_drtbp_long_alpha3: alpha to blend bp_long to the drtpk filter corresponding angle (0/45/90/135)
11:8	RW	15	reg_sr7_drtbp_long_alpha2: alpha to blend bp_long to the drtpk filter corresponding angle (0/45/90/135)
7:4	RW	15	reg_sr7_drtbp_long_alpha1: alpha to blend bp_long to the drtpk filter corresponding angle (0/45/90/135)
3:0	RW	15	reg_sr7_drtbp_long_alpha0: alpha to blend bp_long to the drtpk filter corresponding angle (0/45/90/135)

**Table 9-2288 SRSHARP1\_SR7\_DRTHPLONG\_ALP 0x531f**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_drthp_long_alpha7: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
27:24	RW	15	reg_sr7_drthp_long_alpha6: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
23:20	RW	15	reg_sr7_drthp_long_alpha5: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
19:16	RW	15	reg_sr7_drthp_long_alpha4: alpha to blend bp_long to the cirpk filter corresponding angle (0/45/90/135)
15:12	RW	15	reg_sr7_drthp_long_alpha3: alpha to blend hp_long to the drtpk filter corresponding angle (0/45/90/135)
11:8	RW	15	reg_sr7_drthp_long_alpha2: alpha to blend hp_long to the drtpk filter corresponding angle (0/45/90/135)
7:4	RW	15	reg_sr7_drthp_long_alpha1: alpha to blend hp_long to the drtpk filter corresponding angle (0/45/90/135)
3:0	RW	15	reg_sr7_drthp_long_alpha0: alpha to blend hp_long to the drtpk filter corresponding angle (0/45/90/135)

**Table 9-2289 SRSHARP1\_SR7\_PKMINMAXCIR\_BLD\_LUT2D0 0x5320**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_pk_mimaxerr2_cirbld_lut2d7
27:24	RW	0	reg_sr7_pk_mimaxerr2_cirbld_lut2d6
23:20	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d5
19:16	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d4
15:12	RW	4	reg_sr7_pk_mimaxerr2_cirbld_lut2d3
11:8	RW	6	reg_sr7_pk_mimaxerr2_cirbld_lut2d2
7:4	RW	12	reg_sr7_pk_mimaxerr2_cirbld_lut2d1
3:0	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d0: 2D-LUT for alpha2 for cir-PK blender, larger coef means less drt-pk, x-min_err, y-max_err

**Table 9-2290 SRSHARP1\_SR7\_PKMINMAXCIR\_BLD\_LUT2D1 0x5321**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d15
27:24	RW	0	reg_sr7_pk_mimaxerr2_cirbld_lut2d14
23:20	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d13
19:16	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d12
15:12	RW	4	reg_sr7_pk_mimaxerr2_cirbld_lut2d11
11:8	RW	6	reg_sr7_pk_mimaxerr2_cirbld_lut2d10

Bit(s)	R/W	Default	Description
7:4	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d9
3:0	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d8

**Table 9-2291 SRSHARP1\_SR7\_PKMINMAXCIR\_BLD\_LUT2D2 0x5322**

Bit(s)	R/W	Default	Description
31:28	RW	6	reg_sr7_pk_mimaxerr2_cirbld_lut2d23
27:24	RW	10	reg_sr7_pk_mimaxerr2_cirbld_lut2d22
23:20	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d21
19:16	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d20
15:12	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d19
11:8	RW	4	reg_sr7_pk_mimaxerr2_cirbld_lut2d18
7:4	RW	6	reg_sr7_pk_mimaxerr2_cirbld_lut2d17
3:0	RW	12	reg_sr7_pk_mimaxerr2_cirbld_lut2d16

**Table 9-2292 SRSHARP1\_SR7\_PKMINMAXCIR\_BLD\_LUT2D3 0x5323**

Bit(s)	R/W	Default	Description
31:28	RW	14	reg_sr7_pk_mimaxerr2_cirbld_lut2d31
27:24	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d30
23:20	RW	4	reg_sr7_pk_mimaxerr2_cirbld_lut2d29
19:16	RW	8	reg_sr7_pk_mimaxerr2_cirbld_lut2d28
15:12	RW	10	reg_sr7_pk_mimaxerr2_cirbld_lut2d27
11:8	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d26
7:4	RW	2	reg_sr7_pk_mimaxerr2_cirbld_lut2d25
3:0	RW	4	reg_sr7_pk_mimaxerr2_cirbld_lut2d24

**Table 9-2293 SRSHARP1\_SR7\_PKMINMAXCIR\_BLD\_LUT2D4 0x5324**

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d35
11:8	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d34
7:4	RW	15	reg_sr7_pk_mimaxerr2_cirbld_lut2d33
3:0	RW	12	reg_sr7_pk_mimaxerr2_cirbld_lut2d32



**Table 9-2294 SRSHARP1\_SR7\_PKMINMAXLPF\_BLD\_LUT2D0 0x5325**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_pk_mimaxerr2_lpfblld_lut2d7
27:24	RW	0	reg_sr7_pk_mimaxerr2_lpfblld_lut2d6
23:20	RW	2	reg_sr7_pk_mimaxerr2_lpfblld_lut2d5
19:16	RW	2	reg_sr7_pk_mimaxerr2_lpfblld_lut2d4
15:12	RW	4	reg_sr7_pk_mimaxerr2_lpfblld_lut2d3
11:8	RW	6	reg_sr7_pk_mimaxerr2_lpfblld_lut2d2
7:4	RW	12	reg_sr7_pk_mimaxerr2_lpfblld_lut2d1
3:0	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d0: 2D-LUT for alpha2 for PK lpf along edge blender, lareger coef means less lpf along edge,x-min_err, y-max_err

**Table 9-2295 SRSHARP1\_SR7\_PKMINMAXLPF\_BLD\_LUT2D1 0x5326**

Bit(s)	R/W	Default	Description
31:28	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d15
27:24	RW	0	reg_sr7_pk_mimaxerr2_lpfblld_lut2d14
23:20	RW	2	reg_sr7_pk_mimaxerr2_lpfblld_lut2d13
19:16	RW	2	reg_sr7_pk_mimaxerr2_lpfblld_lut2d12
15:12	RW	4	reg_sr7_pk_mimaxerr2_lpfblld_lut2d11
11:8	RW	6	reg_sr7_pk_mimaxerr2_lpfblld_lut2d10
7:4	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d9
3:0	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d8

**Table 9-2296 SRSHARP1\_SR7\_PKMINMAXLPF\_BLD\_LUT2D2 0x5327**

Bit(s)	R/W	Default	Description
31:28	RW	6	reg_sr7_pk_mimaxerr2_lpfblld_lut2d23
27:24	RW	10	reg_sr7_pk_mimaxerr2_lpfblld_lut2d22
23:20	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d21
19:16	RW	2	reg_sr7_pk_mimaxerr2_lpfblld_lut2d20
15:12	RW	2	reg_sr7_pk_mimaxerr2_lpfblld_lut2d19
11:8	RW	4	reg_sr7_pk_mimaxerr2_lpfblld_lut2d18
7:4	RW	6	reg_sr7_pk_mimaxerr2_lpfblld_lut2d17
3:0	RW	12	reg_sr7_pk_mimaxerr2_lpfblld_lut2d16

**Table 9-2297 SRSHARP1\_SR7\_PKMINMAXLPF\_BLD\_LUT2D3 0x5328**

Bit(s)	R/W	Default	Description
31:28	RW	14	reg_sr7_pk_mimaxerr2_lpfblld_lut2d31
27:24	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d30
23:20	RW	4	reg_sr7_pk_mimaxerr2_lpfblld_lut2d29
19:16	RW	8	reg_sr7_pk_mimaxerr2_lpfblld_lut2d28
15:12	RW	10	reg_sr7_pk_mimaxerr2_lpfblld_lut2d27
11:8	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d26
7:4	RW	2	reg_sr7_pk_mimaxerr2_lpfblld_lut2d25
3:0	RW	4	reg_sr7_pk_mimaxerr2_lpfblld_lut2d24

**Table 9-2298 SRSHARP1\_SR7\_PKMINMAXLPF\_BLD\_LUT2D4 0x5329**

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d35
11:8	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d34
7:4	RW	15	reg_sr7_pk_mimaxerr2_lpfblld_lut2d33
3:0	RW	12	reg_sr7_pk_mimaxerr2_lpfblld_lut2d32

**Table 9-2299 SRSHARP1\_SR7\_PKDRT\_BLD\_EN 0x532a**

Bit(s)	R/W	Default	Description
31:1			reserved
0	RW	0	reg_sr7_pkdrtblld_beta_en: enable of direction ambiguity protection for drt and cir blend

**Table 9-2300 SRSHARP1\_SR7\_DRTDIF\_TH 0x532b**

Bit(s)	R/W	Default	Description
31:16			reserved
15::8	RW	128	reg_sr7_drtdif_min2sad_th1
7:0	RW	128	reg_sr7_drtdif_min2sad_th0:for min2_sad threshold for ambiguity ignoring

**Table 9-2301 SRSHARP1\_SR7\_TIBLD\_PRT 0x532c**

Bit(s)	R/W	Default	Description
31:16			reserved
13	RW	1	reg_sr7_hlti_dcblld_beta_en:enable of denoise protection for dc org and denoise blend
12	RW	1	reg_sr7_vlti_dcblld_beta_en:enable of denoise protection for dc org and denoise blend

Bit(s)	R/W	Default	Description
11:8	RW	0	reg_sr7_xcti_dcblnd_beta1:beta for denoise and org dc blend base on the x=cal_drt_dif4(min_idx,min2_idx)
7:4	RW	15	reg_sr7_xcti_dcblnd_beta0:beta for denoise and org dc blend base on the x=cal_drt_dif4(min_idx,min2_idx)
3	RW	1	reg_sr7_hcti_dcblnd_beta_en:enable of denoise protection for dc org and denoise blend
2	RW	1	reg_sr7_vcti_dcblnd_beta_en:enable of denoise protection for dc org and denoise blend
1	RW	0	reg_sr7_hcti_dcblnd_use_ybeta:enable to use beta from hlti
0	RW	0	reg_sr7_vcti_dcblnd_use_ybeta:enable to use beta from vlti

Table 9-2302 SRSARP1\_SR7\_HTI\_OPT\_FORCE 0x532d

Bit(s)	R/W	Default	Description
31:6			reserved
7:6	RW	1	reg_sr7_diag_force_hti1:force hti: for diagonal edges luma/chroma, 0: bypass, 1: diagonal use vti, 2:diagonal use hti
5:4	RW	1	reg_sr7_diag_force_hti0:force hti for diagonal edges luma/chroma, 0: bypass, 1: diagonal use vti, 2:diagonal use hti
3	RW	1	reg_sr7_horz_force_vti1:force vti for horizontal edges luma/chroma, 0: bias hti, 1: horizontal edge force 100% vti
2	RW	1	reg_sr7_horz_force_vti0:force vti for horizontal edges luma/chroma, 0: bias hti, 1: horizontal edge force 100% vti
1	RW	1	reg_sr7_alph_force_hvsad1:alpha = minsad*64/maxsad, force minsad=sad_h, maxsad=sad_v for alpha calculation
0	RW	1	reg_sr7_alph_force_hvsad0:alpha = minsad*64/maxsad, force minsad=sad_h, maxsad=sad_v for alpha calculation.[0]for luma, [1] for chroma

Table 9-2303 SRSARP1\_SR7\_HVTI\_FINALGAIN 0x532e

Bit(s)	R/W	Default	Description
31:28			reserved
27:16	RW	256	reg_sr7_hvti_finalgain1: final gain for HVTI boost, for easier level tuning for application
15:12			reserved
11:0	RW	256	reg_sr7_hvti_finalgain0: final gain for HVTI boost, for easier level tuning for application

Table 9-2304 SRSARP1\_SR7\_TIOS\_SDRATIO 0x532f

Bit(s)	R/W	Default	Description
31:30			reserved
29:24	RW	3	reg_sr7_hti_osmargin_sdratio1:ratio for HTI OS margin adaptive to sd
23:22			reserved

Bit(s)	R/W	Default	Description
21:16	RW	3	reg_sr7_hti_osmargin_sdratio0:ratio for HTI OS margin adaptive to sd
15:14			reserved
13:8	RW	3	reg_sr7_vti_osmargin_sdratio1:ratio for VTI OS margin adaptive to sd
7:6			reserved
5:0	RW	3	reg_sr7_vti_osmargin_sdratio0:ratio for VTI OS margin adaptive to sd

Table 9-2305 SRSHARP1\_SR7\_XTI\_SDFDEN 0x5330

Bit(s)	R/W	Default	Description
31:1			reserved
1	RW	1	reg_sr7_xtigain_sdfd_en1: adaptive gamma for cti boost gain enable.
0	RW	1	reg_sr7_xtigain_sdfd_en0: adaptive gamma for cti boost gain enable.[0]for luma, [1] for chroma

Table 9-2306 SRSHARP1\_SR7\_FDSD\_PARAM 0x5331

Bit(s)	R/W	Default	Description
31:26			reserved
25	RW	1	reg_fdsd_vlpf_en:use vertical [1 2 1] lpf filter for the fd and sd derivativecalculations
24	RW	1	reg_fdsd_hlpf_en:use horizontal [1 2 1] lpf filter for the fd and sd derivativecalculations
23:20	RW	0	reg_sd_coring_th1:coring to $SD' = \text{MAX}(SD - FD * \text{ratio} - \text{coring}, 0)$
19:16	RW	0	reg_sd_coring_th0:coring to $SD' = \text{MAX}(SD - FD * \text{ratio} - \text{coring}, 0)$
15:14			reserved
13:8	RW	4	reg_sd_coring_ratio2fd1:ratio to FD for adaptive coring to $SD' = \text{MAX}(SD - FD * \text{ratio} - \text{coring}, 0)$
7:6			reserved
5:0	RW	4	reg_sd_coring_ratio2fd0:ratio to FD for adaptive coring to $SD' = \text{MAX}(SD - FD * \text{ratio} - \text{coring}, 0)$

Table 9-2307 SRSHARP1\_SR7\_TI\_BPF\_EN 0x5332

Bit(s)	R/W	Default	Description
31:4			reserved
3	RW	1	reg_sr7_hlti_bpf_en:enable of tap09 BPF for hlti (gain3), only store half of the filter
2	RW	1	reg_sr7_hcti_bpf_en:enable of tap09 BPF for hcti (gain3), only store half of the filter
1	RW	1	reg_sr7_vlti_bpf_en:enable of tap09 BPF for vcti (gain3), only store half of the filter
0	RW	1	reg_sr7_vcti_bpf_en:enable of tap09 BPF for vlti (gain3), only store half of the filter

**Table 9-2308 SRSHARP1\_SR7\_PKLONG\_PF\_EN 0x5333**

Bit(s)	R/W	Default	Description
31:2			reserved
1	RW	1	reg_sr7_pk_long_bpf_en:enable of long BPF for peaking
0	RW	1	reg_sr7_pk_long_hpf_en:enable of long HPF for peaking

**Table 9-2309 SRSHARP1\_SR7\_PKLONG\_PF\_GAIN 0x5334**

Bit(s)	R/W	Default	Description
31:24	RW	64	reg_sr7_pk_long_bpf_hzgain:gain to long horizontal BPF for peaking
23:16	RW	64	reg_sr7_pk_long_bpf_vdgain:gain to long vertical and diagonal BPF for peaking
15:8	RW	64	reg_sr7_pk_long_hpf_hzgain:gain to long horizontal HPF for peaking
7:0	RW	64	reg_sr7_pk_long_hpf_vdgain:gain to long vertical and diagonal HPF for peaking

**Table 9-2310 SRSHARP1\_SR7\_PKMINMAX\_BLD 0x5335**

Bit(s)	R/W	Default	Description
31:4			reserved
3	RW	1	reg_sr7_pk_mimaxerr2_cirbld_on_bp:enable to use adaptive blender of drtBP vs cirBP alpha2, the larger of cell, the more cirPK results, 0: alp2=lut2d(minerr, maxerr), 1:alp2=min_err/maxerr
2	RW	1	reg_sr7_pk_mimaxerr2_cirbld_on_hp:enable to use adaptive blender of drtHP vs cirHP alpha2, the larger of cell, the more cirPK results, 0: alp2=lut2d(minerr, maxerr), 1:alp2=min_err/maxerr
1	RW	1	reg_sr7_pk_mimaxerr2_lpfbld_on_bp:enable to use adaptive blender of BP result lpf along edge based on the mimaxsad relationship
0	RW	1	reg_sr7_pk_mimaxerr2_lpfbld_on_hp:enable to use adaptive blender of HP result lpf along edge based on the mimaxsad relationship

**Table 9-2311 SRSHARP1\_SR7\_TI\_CONMAXERR\_GAIN 0x5336**

Bit(s)	R/W	Default	Description
31:16			reserved
15:12	RW	8	reg_sr7_hti_conmaxerr_gain1:con=max(con, maxerr) for the horizontal transition calculation, gain=0, same as s6 and prev, [0] for y and [1] for uv
11:8	RW	8	reg_sr7_hti_conmaxerr_gain0:con=max(con, maxerr) for the horizontal transition calculation, gain=0, same as s6 and prev, [0] for y and [1] for uv
7:4	RW	8	reg_sr7_vti_conmaxerr_gain1:con=max(con, maxerr) for the vertical transition calculation, gain=0, same as s6 and prev, [0] for y and [1] for uv
3:0	RW	8	reg_sr7_vti_conmaxerr_gain0:con=max(con, maxerr) for the vertical transition calculation, gain=0, same as s6 and prev, [0] for y and [1] for uv

**Table 9-2312 SRSHARP1\_SR7\_CC\_PK\_ADJ 0x5337**

Bit(s)	R/W	Default	Description
31:25			reserved
24	RW	1	reg_sr7_cc_enable:color compensation enable, 0: no CC, 1: with CC
23:22			reserved
21:20	RW	0	reg_sr7_cc_yinp_sel:color compensation input luma selection, 0: org_y; 1: gau_y; 2:gauadp_y; 3:edgeadp_y (same as dnlp input sel)
19:18	RW	2	reg_sr7_cc_ydlt_sel:color compensation output luma selection, 0: peaking+lti output; 1: dnlp output; 2/3: peaking+lti+dnlp
17:16	RW	1	reg_sr7_cc_sat_norm:normalization of lut cell to saturation. 0: norm to 8 as 1.0, sat[0:23/8]; 1: norm to 16 as 1.0, sat[1/16:31/16]; 2: norm to 32 as 1.0, sat[17/32:47/32]; 3: norm to 64 as 1.0, sat[49/64:79/64]
15:8	RW	64	reg_sr7_cc_ydlt_psc1:prescale to the y-delta (if >0) before feeding to y-lumadlt, normalized 64 as 1.0
7:0	RW	64	reg_sr7_cc_ydlt_nsc1:prescale to the y-delta (if <0) before feeding to y-lumadlt, normalized 64 as 1.0

**Table 9-2313 SRSHARP1\_SR7\_CC\_LUT0 0x5338**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_cc_lut003:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=0
27:24	RW	0	reg_sr7_cc_lut002:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=0
23:20	RW	0	reg_sr7_cc_lut001:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=0
19:16	RW	0	reg_sr7_cc_lut000:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=0
15:12	RW	8	reg_sr7_cc_lut103:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=1/8
11:8	RW	4	reg_sr7_cc_lut102:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=1/8
7:4	RW	2	reg_sr7_cc_lut101:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=1/8
3:0	RW	1	reg_sr7_cc_lut100:valid for y-lumadlt= 1/4, 1/2, 3/4, 1.0; of x-lumainput=1/8

**Table 9-2314 SRSHARP1\_SR7\_CC\_LUT1 0x5339**

Bit(s)	R/W	Default	Description
31:28			reserved
27:24	RW	12	reg_sr7_cc_lut202:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=2/8
23:20	RW	8	reg_sr7_cc_lut201:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=2/8
19:16	RW	4	reg_sr7_cc_lut200:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=2/8
15:12			reserved
11:8	RW	15	reg_sr7_cc_lut302:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=3/8
7:4	RW	10	reg_sr7_cc_lut301:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=3/8
3:0	RW	5	reg_sr7_cc_lut300:valid for y-lumadlt= 1/4, 1/2, 3/4; of x-lumainput=3/8

**Table 9-2315 SRSHARP1\_SR7\_CC\_LUT2 0x533a**

Bit(s)	R/W	Default	Description
31:24			reserved
23:20	RW	15	reg_sr7_cc_lut401:valid for y-lumadlt= 1/4, 1/2 ; of x-lumainput=4/8
19:16	RW	8	reg_sr7_cc_lut400:valid for y-lumadlt= 1/4, 1/2; of x-lumainput=4/8
15:12	RW	13	reg_sr7_cc_lut501:valid for y-lumadlt= 1/4, 1/2; of x-lumainput=5/8
11:8	RW	6	reg_sr7_cc_lut500:valid for y-lumadlt= 1/4, 1/2; of x-lumainput=5/8
7:4	RW	5	reg_sr7_cc_lut600:valid for y-lumadlt= 1/4 of x-lumainput=6/8
3:0	RW	4	reg_sr7_cc_lut700:valid for y-lumadlt= 1/4 of x-lumainput=7/8

**Table 9-2316 SRSHARP1\_SR7\_CC\_LUT3 0x533b**

Bit(s)	R/W	Default	Description
31:24			reserved
23:20	RW	4	reg_sr7_cc_lut710:valid for y-lumadlt= -1/4,; of x-lumainput=1/8
19:16	RW	5	reg_sr7_cc_lut610:valid for y-lumadlt= -1/4,; of x-lumainput=2/8
15:12	RW	13	reg_sr7_cc_lut511:valid for y-lumadlt= -1/4, -1/2; of x-lumainput=3/8
11:8	RW	6	reg_sr7_cc_lut510:valid for y-lumadlt= -1/4, -1/2; of x-lumainput=3/8
7:4	RW	15	reg_sr7_cc_lut411:valid for y-lumadlt= -1/4, -1/2 of x-lumainput=4/8
3:0	RW	8	reg_sr7_cc_lut410:valid for y-lumadlt= -1/4 , -1/2 of x-lumainput=4/8

**Table 9-2317 SRSHARP1\_SR7\_CC\_LUT4 0x533c**

Bit(s)	R/W	Default	Description
31:24			reserved
23:20	RW	15	reg_sr7_cc_lut312:valid for y-lumadlt= -1/4,-1/2, -3/4; of x-lumainput=5/8
19:16	RW	10	reg_sr7_cc_lut311:valid for y-lumadlt= -1/4,-1/2, -3/4; of x-lumainput=5/8
15:12	RW	5	reg_sr7_cc_lut310:valid for y-lumadlt= -1/4,-1/2, -3/4; of x-lumainput=5/8
11:8	RW	12	reg_sr7_cc_lut212:valid for y-lumadlt= -1/4, -1/2, -3/4;; of x-lumainput=6/8
7:4	RW	8	reg_sr7_cc_lut211:valid for y-lumadlt= -1/4, -1/2, -3/4; of x-lumainput=6/8
3:0	RW	4	reg_sr7_cc_lut210:valid for y-lumadlt= -1/4, -1/2, -3/4; of x-lumainput=6/8

**Table 9-2318 SRSHARP1\_SR7\_CC\_LUT5 0x533d**

Bit(s)	R/W	Default	Description
31:28	RW	8	reg_sr7_cc_lut113:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=7/8
27:24	RW	4	reg_sr7_cc_lut112:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=7/8
23:20	RW	2	reg_sr7_cc_lut111:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=7/8
19:16	RW	1	reg_sr7_cc_lut110:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=7/8

Bit(s)	R/W	Default	Description
15:12	RW	0	reg_sr7_cc_lut013:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=8/8
11:8	RW	0	reg_sr7_cc_lut012:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=8/8
7:4	RW	0	reg_sr7_cc_lut011:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=8/8
3:0	RW	0	reg_sr7_cc_lut010:valid for y-lumadlt= -1/4, -1/2, -3/4, -1.0; of x-lumainput=8/8

Table 9-2319 SRSHARP1\_SR7\_GRAPHIC\_CTRL 0x533e

Bit(s)	R/W	Default	Description
31:11			reserved
10	RW	1	reg_sr7_grph_en:enable graphic statistic
9	RW	1	reg_sr7_grph_hflt:horizontal filter, 0: [0 1 -1], 1: [-1 2 -1]
8	RW	1	reg_sr7_grph_vflt: vertical filter, 0: [0 1 -1], 1: [-1 2 -1]
7:0	RW	0	reg_sr7_grph_dif_cor:coring for dif while count for graphic

Table 9-2320 SRSHARP1\_SR7\_GRAPHIC\_THD\_GAIN 0x533f

Bit(s)	R/W	Default	Description
31:24	RW	2	reg_sr7_grph_fit_thd: flat threshold for dif while count for graphic.
23:16	RW	40	reg_sr7_grph_dtl_thd:detail threshold for dif while count for graphic
15:8	RW	32	reg_sr7_grph_hgain:horizontal gain for fast squart of hp
7:0	RW	32	reg_sr7_grph_vgain:vertical gain for fast squart of hp

Table 9-2321 SRSHARP1\_SR7\_RO\_GRAPHIC\_FLT\_CNT 0x5340

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_sr7_grph_fit_cnt:: flat count numbers for graphic

Table 9-2322 SRSHARP1\_SR7\_RO\_GRAPHIC\_DTL\_CNT 0x5341+

Bit(s)	R/W	Default	Description
31:0	RO	0	ro_sr7_grph_dtl_cnt::detail count numbers for graphic

Table 9-2323 SRSHARP1\_SR7\_CLR\_PRT\_PARAM 0x5342

Bit(s)	R/W	Default	Description
31:18			reserved
17:16	RW	3	reg_sr7_clr_prct_inpsel:input UV selection for color protection, 0: org; 1: NRout; 2: CTlout; 3: (NR+Tlout)/2



Bit(s)	R/W	Default	Description
15:8	RW	64	reg_sr7_clr_prct_dnlp_gain:gain to de-boost of dnlp_dlt base on color region, norm to 64 as 1.0, set to 0 as disable.
7:0	RW	64	reg_sr7_clr_prct_peak_gain:gain to de-boost of peak_dlt base on color region, norm to 64 as 1.0, set to 0 as disable

Table 9-2324 SRSHARP1\_SR7\_CLR\_PRT\_LC\_GAIN 0x5343

Bit(s)	R/W	Default	Description
31:24			reserved
23:16	RW	64	reg_sr7_clr_prct_lc_gain2:gain to de-boost of lc_dlt (y/u/v) base on color region, norm to 64 as 1.0, set to 0 as disable
15:8	RW	64	reg_sr7_clr_prct_lc_gain1:gain to de-boost of lc_dlt (y/u/v) base on color region, norm to 64 as 1.0, set to 0 as disable
7:0	RW	64	reg_sr7_clr_prct_lc_gain0:gain to de-boost of lc_dlt (y/u/v) base on color region, norm to 64 as 1.0, set to 0 as disable

Table 9-2325 SRSHARP1\_SR7\_CLR\_PRT\_LUT0 0x5344

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut7:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut6:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut5:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut4:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut3:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut2:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut1:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut0:color protection lut, 16 is normalized to 1

Table 9-2326 SRSHARP1\_SR7\_CLR\_PRT\_LUT1 0x5345

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut15:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut14:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut13:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut12:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut11:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut10:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut9:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut8:color protection lut, 16 is normalized to 1

**Table 9-2327 SRSHARP1\_SR7\_CLR\_PRT\_LUT2 0x5346**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut23:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut22:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut21:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut20:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut19:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut18:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut17:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut16:color protection lut, 16 is normalized to 1

**Table 9-2328 SRSHARP1\_SR7\_CLR\_PRT\_LUT3 0x5347**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut31:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut30:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut29:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut28:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut27:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut26:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut25:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut24:color protection lut, 16 is normalized to 1

**Table 9-2329 SRSHARP1\_SR7\_CLR\_PRT\_LUT4 0x5348**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut39:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut38:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut37:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut36:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut35:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut34:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut33:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut32:color protection lut, 16 is normalized to 1

**Table 9-2330 SRSHARP1\_SR7\_CLR\_PRT\_LUT5 0x5349**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut47:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut46:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut45:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut44:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut43:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut42:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut41:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut40:color protection lut, 16 is normalized to 1

**Table 9-2331 SRSHARP1\_SR7\_CLR\_PRT\_LUT6 0x534a**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut55:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut54:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut53:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut52:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut51:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut50:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut49:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut48:color protection lut, 16 is normalized to 1

**Table 9-2332 SRSHARP1\_SR7\_CLR\_PRT\_LUT7 0x534b**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut63:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut62:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut61:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut60:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut59:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut58:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut57:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut56:color protection lut, 16 is normalized to 1

**Table 9-2333 SRSHARP1\_SR7\_CLR\_PRT\_LUT8 0x534c**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut71:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut70:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut69:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut68:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut67:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut66:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut65:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut64:color protection lut, 16 is normalized to 1

**Table 9-2334 SRSHARP1\_SR7\_CLR\_PRT\_LUT9 0x534d**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut79:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut78:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut77:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut76:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut75:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut74:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut73:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut72:color protection lut, 16 is normalized to 1

**Table 9-2335 SRSHARP1\_SR7\_CLR\_PRT\_LUT10 0x534e**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut87:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut86:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut85:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut84:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut83:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut82:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut81:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut80:color protection lut, 16 is normalized to 1

**Table 9-2336 SRSHARP1\_SR7\_CLR\_PRT\_LUT11 0x534f**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut95:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut94:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut93:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut92:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut91:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut90:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut89:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut88:color protection lut, 16 is normalized to 1

**Table 9-2337 SRSHARP1\_SR7\_CLR\_PRT\_LUT12 0x5350**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut103:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut102:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut101:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut100:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut99:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut98:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut97:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut96:color protection lut, 16 is normalized to 1

**Table 9-2338 SRSHARP1\_SR7\_CLR\_PRT\_LUT13 0x5351**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut111:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut110:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut109:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut108:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut107:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut106:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut105:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut104:color protection lut, 16 is normalized to 1

**Table 9-2339 SRSHARP1\_SR7\_CLR\_PRT\_LUT14 0x5352**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut119:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut118:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut117:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut116:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut115:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut114:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut113:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut112:color protection lut, 16 is normalized to 1

**Table 9-2340 SRSHARP1\_SR7\_CLR\_PRT\_LUT15 0x5353**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut127:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut126:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut125:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut124:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut123:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut122:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut121:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut120:color protection lut, 16 is normalized to 1

**Table 9-2341 SRSHARP1\_SR7\_CLR\_PRT\_LUT16 0x5354**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut135:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut134:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut133:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut132:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut131:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut130:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut129:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut128:color protection lut, 16 is normalized to 1

**Table 9-2342 SRSARP1\_SR7\_CLR\_PRT\_LUT17 0x5355**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut143:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut142:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut141:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut140:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut139:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut138:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut137:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut136:color protection lut, 16 is normalized to 1

**Table 9-2343 SRSARP1\_SR7\_CLR\_PRT\_LUT18 0x5356**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut151:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut150:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut149:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut148:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut147:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut146:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut145:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut144:color protection lut, 16 is normalized to 1

**Table 9-2344 SRSARP1\_SR7\_CLR\_PRT\_LUT19 0x5357**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut159:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut158:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut157:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut156:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut155:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut154:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut153:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut152:color protection lut, 16 is normalized to 1

**Table 9-2345 SRSARP1\_SR7\_CLR\_PRT\_LUT20 0x5358**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut167:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut166:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut165:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut164:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut163:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut162:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut161:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut160:color protection lut, 16 is normalized to 1

**Table 9-2346 SRSARP1\_SR7\_CLR\_PRT\_LUT21 0x5359**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut175:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut174:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut173:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut172:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut171:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut170:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut169:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut168:color protection lut, 16 is normalized to 1

**Table 9-2347 SRSARP1\_SR7\_CLR\_PRT\_LUT22 0x535a**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut183:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut182:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut181:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut180:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut179:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut178:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut177:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut176:color protection lut, 16 is normalized to 1



**Table 9-2348 SRSHARP1\_SR7\_CLR\_PRT\_LUT23 0x535b**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut191:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut190:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut189:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut188:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut187:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut186:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut185:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut184:color protection lut, 16 is normalized to 1

**Table 9-2349 SRSHARP1\_SR7\_CLR\_PRT\_LUT24 0x535c**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut199:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut198:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut197:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut196:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut195:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut194:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut193:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut192:color protection lut, 16 is normalized to 1

**Table 9-2350 SRSHARP1\_SR7\_CLR\_PRT\_LUT25 0x535d**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut207:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut206:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut205:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut204:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut203:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut202:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut201:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut200:color protection lut, 16 is normalized to 1

**Table 9-2351 SRSARP1\_SR7\_CLR\_PRT\_LUT26 0x535e**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut215:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut214:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut213:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut212:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut211:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut210:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut209:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut208:color protection lut, 16 is normalized to 1

**Table 9-2352 SRSARP1\_SR7\_CLR\_PRT\_LUT27 0x535f**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut223:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut222:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut221:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut220:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut219:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut218:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut217:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut216:color protection lut, 16 is normalized to 1

**Table 9-2353 SRSARP1\_SR7\_CLR\_PRT\_LUT28 0x5360**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut231:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut230:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut229:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut228:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut227:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut226:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut225:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut224:color protection lut, 16 is normalized to 1

**Table 9-2354 SRSARP1\_SR7\_CLR\_PRT\_LUT29 0x5361**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut239:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut238:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut237:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut236:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut235:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut234:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut233:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut232:color protection lut, 16 is normalized to 1

**Table 9-2355 SRSARP1\_SR7\_CLR\_PRT\_LUT30 0x5362**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut247:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut246:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut245:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut244:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut243:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut242:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut241:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut240:color protection lut, 16 is normalized to 1

**Table 9-2356 SRSARP1\_SR7\_CLR\_PRT\_LUT31 0x5363**

Bit(s)	R/W	Default	Description
31:28	RW	0	reg_sr7_clr_prct_lut255:color protection lut, 16 is normalized to 1
27:24	RW	0	reg_sr7_clr_prct_lut254:color protection lut, 16 is normalized to 1
23:20	RW	0	reg_sr7_clr_prct_lut253:color protection lut, 16 is normalized to 1
19:16	RW	0	reg_sr7_clr_prct_lut252:color protection lut, 16 is normalized to 1
15:12	RW	0	reg_sr7_clr_prct_lut251:color protection lut, 16 is normalized to 1
11:8	RW	0	reg_sr7_clr_prct_lut250:color protection lut, 16 is normalized to 1
7:4	RW	0	reg_sr7_clr_prct_lut249:color protection lut, 16 is normalized to 1
3:0	RW	0	reg_sr7_clr_prct_lut248:color protection lut, 16 is normalized to 1

### 9.2.3.29 LC STTS / LC Curve Generator

**Table 9-2357 LC\_CURVE\_CTRL 0x4000**

Bit(s)	R/W	Default	Description
31	R/W	0	reg_lc_int_enable : 1 to enable LC_CURVE done interrupt
30: 14	R/W	0	reserved
13: 12	R/W	0	reg_lc_gclk_ctrl : gated clock control
11: 10	R/W	0	reserved
9: 8	R/W	1	reg_lc_hist_curve_nodes_hlpf: horizontal lpf of the ram_curve_nodes, 0: no LPF, 1= [1 2 1]; 2: [1 2 2 2 1]/8
7: 6	R/W	0	reserved
5: 4	R/W	1	reg_lc_hist_curve_nodes_vlpf: Vertical lpf of the ram_curve_nodes, 0: no LPF, 1= [1 2 1]; 2: [1 2 2 2 1]/8
3: 2	R/W	0	reserved
1	R/W	1	reg_lc_blackbar_mute_en: mute the black bar corresponding bin, 0: no mute, 1: mute enable
0	R/W	0	reg_lc_curve_en

**Table 9-2358 LC\_CURVE\_HV\_NUM 0x4001**

Bit(s)	R/W	Default	Description
31: 13	R/W	0	reserved
12: 8	R/W	12	reg_lc_cur_blk_hnum: lc processing region number of H, maximum to 12
7: 5	R/W	0	reserved
4: 0	R/W	8	reg_lc_cur_blk_vnum: lc processing region number of V, maximum to 8

**Table 9-2359 LC\_CURVE\_LMT\_RAT 0x4002**

Bit(s)	R/W	Default	Description
31: 16	R/W	0	reserved
15: 8	R/W	6	reg_lmtrat_minmax: x/1024 of amount
7: 0	R/W	0x14	reg_lmtrat_valid: x/1024 of amount

**Table 9-2360 LC\_CURVE\_CONTRAST\_LH 0x4003**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x10	reg_lc_contrast_low: contrast gain to the lc for dark side, normalized 256 as "1"
15: 10	R/W	0	reserved
9: 0	R/W	0x18	reg_lc_contrast_hig: contrast gain to the lc for bright side, normalized 256 as "1"

**Table 9-2361 LC\_CURVE\_CONTRAST\_\_LMT\_LH 0x4004**

Bit(s)	R/W	Default	Description
31: 24	R/W	0xa	reg_lc_cntstlmt_low_0: limit for the contrast low
23: 16	R/W	0x14	reg_lc_cntstlmt_hig_0: limit for the contrast high
15: 8	R/W	0xff	reg_lc_cntstlmt_low_1: limit for the contrast low
7: 0	R/W	0xff	reg_lc_cntstlmt_hig_1: limit for the contrast high

**Table 9-2362 LC\_CURVE\_CONTRAST\_SCL\_LH 0x4005**

Bit(s)	R/W	Default	Description
31: 16	R/W	0	reserved
15: 8	R/W	0x40	reg_lc_cntstlmt_low: scale for the contrast low, norm 8 as 1
7: 0	R/W	0x20	reg_lc_cntstlmt_hig: scale for the contrast high, norm 8 as 1

**Table 9-2363 LC\_CURVE\_CONTRAST\_BVN\_LH 0x4006**

Bit(s)	R/W	Default	Description
31: 16	R/W	0	reserved
15: 8	R/W	32	reg_lc_cntstlmt_low: scale to num_m as limit of min_val to minBV distance, to protect mono-color.
7: 0	R/W	32	reg_lc_cntstlmt_hig: scale for the contrast high, norm 8 as 1

**Table 9-2364 LC\_CURVE\_MISC0 0x4007**

Bit(s)	R/W	Default	Description
31: 20	R/W	0	reserved
19: 16	R/W	1	reg_lc_num_m_coring: coring to num_m, soft coring
15: 8	R/W	0x30	reg_lc_vbin_min: 4x is min width of valid histogram bin num
7: 0	R/W	0x38	reg_lc_slope_max_face: maximum slope for the pkBin-maxBV range curve to do face protection

**Table 9-2365 LC\_CURVE\_YPKBV\_RAT 0x4008**

Bit(s)	R/W	Default	Description
31: 24	R/W	0x8c	reg_lc_ypkbv_ratio_0: $x = \text{ratio} * (\text{maxBV} - \text{minBV}) + \text{min\_val}$ as low bound of the ypkBV; normalized to 256 as 1
23: 16	R/W	0xa0	reg_lc_ypkbv_ratio_1
15: 8	R/W	0x78	reg_lc_ypkbv_ratio_2:
7: 0	R/W	0x60	reg_lc_ypkbv_ratio_3:

**Table 9-2366 LC\_CURVE\_YPKBV\_SLP\_LMT 0x4009**

Bit(s)	R/W	Default	Description
31: 16	R/W	0	reserved
15: 8	R/W	0xc	reg_lc_ypkbv_slope_lmt_0: min max slop for the curves to avoid artifacts, for min_slope
7: 0	R/W	0x60	reg_lc_ypkbv_slope_lmt_1: min max slop for the curves to avoid artifacts, for max_slope

**Table 9-2367 LC\_CURVE\_YMINVAL\_LMT\_0\_1 0x400a**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x30	reg_lc_yminval_lmt_0: lmt_val = lmt[minBV(64:64:768)], and yminV = MAX(yminV,lmt_val), for very dark region boost.
15: 10	R/W	0	reserved
9: 0	R/W	0x50	reg_lc_yminval_lmt_1

**Table 9-2368 LC\_CURVE\_YMINVAL\_LMT\_2\_3 0x400b**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x78	reg_lc_yminval_lmt_2
15: 10	R/W	0	reserved
9: 0	R/W	0x80	reg_lc_yminval_lmt_3

**Table 9-2369 LC\_CURVE\_YMINVAL\_LMT\_4\_5 0x400c**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0xa0	reg_lc_yminval_lmt_4
15: 10	R/W	0	reserved
9: 0	R/W	0xc4	reg_lc_yminval_lmt_5

**Table 9-2370 LC\_CURVE\_YMINVAL\_LMT\_6\_7 0x400d**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0xe0	reg_lc_yminval_lmt_6
15: 10	R/W	0	reserved
9: 0	R/W	0x100	reg_lc_yminval_lmt_7

**Table 9-2371 LC\_CURVE\_YMINVAL\_LMT\_8\_9 0x400e**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x120	reg_lc_yminval_lmt_8
15: 10	R/W	0	reserved
9: 0	R/W	0x140	reg_lc_yminval_lmt_9

**Table 9-2372 LC\_CURVE\_YMINVAL\_LMT\_10\_11 0x400f**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x160	reg_lc_yminval_lmt_10
15: 10	R/W	0	reserved
9: 0	R/W	0x190	reg_lc_yminval_lmt_11

**Table 9-2373 LC\_CURVE\_YMAXVAL\_LMT\_0\_1 0x4010**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x78	reg_lc_ymaxval_lmt_0: lmt_val = 4*lmt[maxBV(64:64:1023) and ymaxV = MAX(ymaxV,lmt[maxBV])
15: 10	R/W	0	reserved
9: 0	R/W	0x96	reg_lc_ymaxval_lmt_1

**Table 9-2374 LC\_CURVE\_YMAXVAL\_LMT\_2\_3 0x4011**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0xe6	reg_lc_ymaxval_lmt_2
15: 10	R/W	0	reserved
9: 0	R/W	0x112	reg_lc_ymaxval_lmt_3

**Table 9-2375 LC\_CURVE\_YMAXVAL\_LMT\_4\_5 0x4012**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x14a	reg_lc_ymaxval_lmt_4
15: 10	R/W	0	reserved
9: 0	R/W	0x180	reg_lc_ymaxval_lmt_5

**Table 9-2376 LC\_CURVE\_YMAXVAL\_LMT\_6\_7 0x4013**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x1d4	reg_lc_ymaxval_lmt_6
15: 10	R/W	0	reserved
9: 0	R/W	0x200	reg_lc_ymaxval_lmt_7

**Table 9-2377 LC\_CURVE\_YMAXVAL\_LMT\_8\_9 0x4014**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x240	reg_lc_ymaxval_lmt_8
15: 10	R/W	0	reserved
9: 0	R/W	0x280	reg_lc_ymaxval_lmt_9

**Table 9-2378 LC\_CURVE\_YMAXVAL\_LMT\_10\_11 0x4015**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x280	reg_lc_ymaxval_lmt_10
15: 10	R/W	0	reserved
9: 0	R/W	0x280	reg_lc_ymaxval_lmt_11

**Table 9-2379 LC\_CURVE\_HISTVLD\_THRD 0x4016**

Bit(s)	R/W	Default	Description
31: 12	R/W	0	reserved
11: 0	R/W	0x1fa	reg_lc_histvld_thrd: threshold to compare to bin to get number of valid bins

**Table 9-2380 LC\_CURVE\_BB\_MUTE\_THRD 0x4017**

Bit(s)	R/W	Default	Description
31: 24	R/W	0	reserved
23: 0	R/W	0x2a30	reg_lc_blackbar_mute_thrd: 1/8 of the region

**Table 9-2381 LC\_CURVE\_INT\_STATUS 0x4018**

Bit(s)	R/W	Default	Description
31: 1	R/W	0	reserved
0	R/W	0	Status to indicate LC curve done, write 1 to clear.



**Table 9-2382 LC\_CURVE\_RAM\_CTRL 0x4020**

Bit(s)	R/W	Default	Description
31:1	R/W	0	reserved
0	R/W	0	reg_lc_cbus2ram_en: 1 to enable CBUS read data from LC Curve RAM

**Table 9-2383 LC\_CURVE\_RAM\_ADDR 0x4021**

Bit(s)	R/W	Default	Description
31:7	R/W	0	reserved
6:4	R/W	0	Vidx of LC CURVE RAM to read
3:0	R/W	0	Hidx of LC CURVE RAM to read

**Table 9-2384 LC\_CURVE\_RAM\_DATA 0x4022**

Bit(s)	R/W	Default	Description
31:30	R/W	0	reserved
29:0	R/W	0	LC Curve RAM data for read

**Table 9-2385 LC\_CURVE\_YMINVAL\_LMT\_12\_13 0x4040**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x160	reg_lc_yminval_lmt_12
15: 10	R/W	0	reserved
9: 0	R/W	0x190	reg_lc_yminval_lmt_13

**Table 9-2386 LC\_CURVE\_YMINVAL\_LMT\_14\_15 0x4041**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x160	reg_lc_yminval_lmt_14
15: 10	R/W	0	reserved
9: 0	R/W	0x190	reg_lc_yminval_lmt_15

**Table 9-2387 LC\_CURVE\_YMAXVAL\_LMT\_12\_13 0x4042**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x280	reg_lc_ymaxval_lmt_12
15: 10	R/W	0	reserved
9: 0	R/W	0x280	reg_lc_ymaxval_lmt_13

**Table 9-2388 LC\_CURVE\_YMAXVAL\_LMT\_14\_15 0x4043**

Bit(s)	R/W	De- fault	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x280	reg_lc_ymaxval_lmt_14
15: 10	R/W	0	reserved
9: 0	R/W	0x280	reg_lc_ymaxval_lmt_15

**Table 9-2389 LC\_CURVE\_YPKBV\_LMT\_0\_1 0x4044**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x78	reg_lc_ypkbv_lmt_0: lmt_val = 4*lmt[pkBV(64:64:1023) and ypkBV = MAX(ypkBV,lmt[pkBV])
15: 10	R/W	0	reserved
9: 0	R/W	0x96	reg_lc_ypkbv_lmt_1

**Table 9-2390 LC\_CURVE\_YPKBV\_LMT\_2\_3 0x4045**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0xe6	reg_lc_ypkbv_lmt_2
15: 10	R/W	0	reserved
9: 0	R/W	0x112	reg_lc_ypkbv_lmt_3

**Table 9-2391 LC\_CURVE\_YPKBV\_LMT\_4\_5 0x4046**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x14a	reg_lc_ypkbv_lmt_4
15: 10	R/W	0	reserved
9: 0	R/W	0x180	reg_lc_ypkbv_lmt_5

**Table 9-2392 LC\_CURVE\_YPKBV\_LMT\_6\_7 0x4047**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x1d4	reg_lc_ypkbv_lmt_6
15: 10	R/W	0	reserved
9: 0	R/W	0x200	reg_lc_ypkbv_lmt_7

**Table 9-2393 LC\_CURVE\_YPKBV\_LMT\_8\_9 0x4048**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x240	reg_lc_ypkbv_lmt_8
15: 10	R/W	0	reserved
9: 0	R/W	0x280	reg_lc_ypkbv_lmt_9

**Table 9-2394 LC\_CURVE\_YPKBV\_LMT\_10\_11 0x4049**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x280	reg_lc_ypkbv_lmt_10
15: 10	R/W	0	reserved
9: 0	R/W	0x280	reg_lc_ypkbv_lmt_11

**Table 9-2395 LC\_CURVE\_YPKBV\_LMT\_12\_13 0x404a**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x280	reg_lc_ypkbv_lmt_12
15: 10	R/W	0	reserved
9: 0	R/W	0x280	reg_lc_ypkbv_lmt_13

**Table 9-2396 LC\_CURVE\_YPKBV\_LMT\_14\_15 0x404b**

Bit(s)	R/W	Default	Description
31: 26	R/W	0	reserved
25: 16	R/W	0x280	reg_lc_ypkbv_lmt_14
15: 10	R/W	0	reserved
9: 0	R/W	0x280	reg_lc_ypkbv_lmt_15

**Table 9-2397 LC\_STTS\_GCLK\_CTRL0 0x4028**

Bit(s)	R/W	Default	Description
7:0	R/W	0	gate clock

**Table 9-2398 LC\_STTS\_CTRL0 0x4029**

Bit(s)	R/W	Default	Description
31:24	R/W	0	reg_bit_sel : 12bit to 10bit for 8 input
23:22	R/W	0	no use

Bit(s)	R/W	Default	Description
21	R/W	0	1: input format is RGB 0: input format is YUV
20	R/W	0	1: input format is RGB 0: input format is YUV
18:16	R/W	0	matrix_conv_rs
15:14	R/W	0	no use
12	R/W	0	no use
11	R/W	0	highlight enable
10	R/W	0	probe enable
9	R/W	0	probe post enable
8-6	R/W	0	input data component mux
5-3	R/W	0	input data sel
2	R/W	0	matrix enable
1	R/W	0	no use
0	R/W	0	soft reset

**Table 9-2399 LC\_STTS\_WIDTHM1\_HEIGHTM1 0x402a**

Bit(s)	R/W	Default	Description
28:16	R/W	0	width m1
12:0	R/W	0	height m1

LC\_STTS\_MATRIX\_COEF00\_01 0x402b

See: VPP\_VD1\_MATRIX\_COEF00\_01

LC\_STTS\_MATRIX\_COEF02\_10 0x402c

See: VPP\_VD1\_MATRIX\_COEF02\_10

LC\_STTS\_MATRIX\_COEF11\_12 0x402d

See: VPP\_VD1\_MATRIX\_COEF11\_12

LC\_STTS\_MATRIX\_COEF20\_21 0x402e

See: VPP\_VD1\_MATRIX\_COEF20\_21

LC\_STTS\_MATRIX\_COEF22 0x402f

See: VPP\_VD1\_MATRIX\_COEF22

LC\_STTS\_MATRIX\_OFFSET0\_1 0x4030

See: VPP\_VD1\_MATRIX\_OFFSET0\_1

LC\_STTS\_MATRIX\_OFFSET2 0x4031

See: VPP\_VD1\_MATRIX\_OFFSET2

LC\_STTS\_MATRIX\_PRE\_OFFSET0\_1 0x4032

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET0\_1

LC\_STTS\_MATRIX\_PRE\_OFFSET2 0x4033

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET2

**Table 9-2400 LC\_STTS\_MATRIX\_PROBE\_COLOR 0x4036**

Bit(s)	R/W	Default	Description
29-20	R	0	component 0
19-10	R	0	component 1
9-0	R	0	component 2

**Table 9-2401 LC\_STTS\_MATRIX\_HL\_COLOR 0x4034**

Bit(s)	R/W	Default	Description
23-16	R/W	0	component 0
15-8	R/W	0	component 1
7-0	R/W	0	component 2

**Table 9-2402 LC\_STTS\_MATRIX\_PROBE\_POS 0x4035**

Bit(s)	R/W	Default	Description
28-16	R/W	0	probe x, position
12-0	R/W	0	probe y, position

**Table 9-2403 LC\_STTS\_HIST\_REGION\_IDX 0x4037**

Bit(s)	R/W	Default	Description
31	R	0	local contrast max statistic enable
28	R	0	eol enable
27-25	R	0	vertical line overlap number for max finding
24-22	R	0	horizontal pixel overlap number, 0: 17 pix, 1: 9 pix, 2: 5 pix, 3: 3 pix, 4: 0 pix
20	R	0	1,2,1 low pass filter enable before max/hist statistic
19-16	R	0	region H/V position index, refer to VDIN_LDIM_STTS_HIST_SET_REGION
15	R	0	1: region read index auto increase per read to VDIN_LDIM_STTS_HIST_READ_REGION
6-0	R	0	region read index

**Table 9-2404 LC\_STTS\_HIST\_SET\_REGION 0x4038**

Bit(s)	R/W	Default	Description
28:16	R	0	if LC_STTS_HIST_REGION_IDX[19:16] == 5'h0: read/write hvstart0 if LC_STTS_HIST_REGION_IDX[19:16] == 5'h1: read/write hend01 if LC_STTS_HIST_REGION_IDX[19:16] == 5'h2: read/write vend01 if LC_STTS_HIST_REGION_IDX[19:16] == 5'h3: read/write hend23 if LC_STTS_HIST_REGION_IDX[19:16] == 5'h4: read/write vend23 if LC_STTS_HIST_REGION_IDX[19:16] == 5'h5: read/write hend45 if LC_STTS_HIST_REGION_IDX[19:16] == 5'h6: read/write vend45 if LC_STTS_HIST_REGION_IDX[19:16] == 5'h7: read/write hend67 if LC_STTS_HIST_REGION_IDX[19:16] == 5'h8: read/write vend67 if LC_STTS_HIST_REGION_IDX[19:16] == 5'h9: read/write hend89 if LC_STTS_HIST_REGION_IDX[19:16] == 5'ha: read/write vend89 //hvstart0, Bit 28:16 row0 vstart, Bit 12:0 col0 hstart //hend01, Bit 28:16 col1 hend, Bit 12:0 col0 hend //vend01, Bit 28:16 row1 vend, Bit 12:0 row0 vend //hend23, Bit 28:16 col3 hend, Bit 12:0 col2 hend //vend23, Bit 28:16 row3 vend, Bit 12:0 row2 vend //hend45, Bit 28:16 col5 hend, Bit 12:0 col4 hend //vend45, Bit 28:16 row5 vend, Bit 12:0 row4 vend //hend67, Bit 28:16 col7 hend, Bit 12:0 col6 hend //vend67, Bit 28:16 row7 vend, Bit 12:0 row6 vend //hend89, Bit 28:16 col9 hend, Bit 12:0 col8 hend //vend89, Bit 28:16 row9 vend, Bit 12:0 row8 vend
12:0	R	0	

**Table 9-2405 LC\_STTS\_HIST\_READ\_REGION 0x4039**

Bit(s)	R/W	Default	Description
29:20	R	0	Max_comp2
19:10	R	0	Max_comp1
9:0	R	0	Max_comp0

**Table 9-2406 LC\_STTS\_HIST\_START\_RD\_REGION 0x403a**

Bit(s)	R/W	Default	Description
29:0	R	0	region data

**Table 9-2407 LC\_STTS\_WHITE\_INFO 0x403b**

Bit(s)	R/W	Default	Description
31:8	R	0	white data count
7:0	R/W	0xf0	white data threshold

**Table 9-2408 LC\_STTS\_BLACK\_INFO 0x403c**

Bit(s)	R/W	Default	Description
31:8	R	0	black data count
7:0	R/W	0x10	black data threshold

### 9.2.3.30 OSD1 Registers

Table 9-2409 VIU\_OSD1\_CTRL\_STAT 0x1A10

Bit(s)	R/W	Default	Description
31	R/W	0	osd_cfg_syn_en : // unsigned , default =0 1: module enable sync by go_field 0: normal
30	R/W	0	ENABLE_FREE_CLK. 1 = Use free-running clock; 0 = Use gated clock to save power.
29	R	0	TEST_RD_DSR: Applicable only when OSD debug mode is enabled. 1 = A new pixel is ready at register VIU_OSD1_TEST_RDDATA; 0 = No data ready.
/	/	/	/
27-24	R	0	OSD_BLK_MODE: the input pixel format of which the current OSD block is being processed. Mali src & normal src have different pixel format value for same format
23-22	R	0	OSD_BLK_PTR: The number of the current OSD block that is being processed.
21	R	0	OSD_ENABLE. 1 = OSD display is enabled; 0 = disabled.
20-12	R/W	0	GLOBAL_ALPHA: legal range 0 – 256. It is a 9-bit value that is multiplied to all output pixel's Alpha value, and then normalized, i.e.: $\text{Alpha\_tmp} = \text{Alpha\_internal} + (\text{Alpha\_internal} == 0 ? 0 : 1);$ $\text{Alpha\_out} = (\text{Alpha\_tmp} * \text{GLOBAL\_ALPHA}) / 256;$
11	R/W	0	TEST_RD_EN: OSD debug mode enable. 1 = Output pixels are not routed to VPP, instead they are presented on registers VIU_OSD1_TEST_RDDATA, for CPU to read. 0 = Normal mode, pixels are output to VPP.
10-9	R/W	0	unused
8-5	R/W	0	CTRL_MTCH_Y: For OSD 444, 422 or 16-bit (COLOR_MATRIX = 0 or 1) mode, the input pixels contain no Alpha information, in order to associated the output pixel with an Alpha value, the following steps are taken: If TC_ALPHA_EN = 0, then all output pixels use a default Alpha value 0xFF; If TC_ALPHA_EN = 1, then the Alpha value is looked up by matching the pixel's Y/Cb/Cr against four Alpha registers' Y/Cb/Cr. If the pixel matches any one of the Alpha registers, then this register's Alpha value is used; If the pixel matches with more than one of the Alpha registers, then the lower Alpha register takes priority, e.g. use Alpha Reg0's value if the pixel match both Alpha Reg0 and Reg1; If no match, then use default Alpha value 0xFF. There are two ways of matching: one way is that the pixel has to compare all Y, Cb and Cr value with the Alpha registers; the other way is that the pixel only has to compare Y value with the Alpha registers. CTRL_MTCH_Y defines which way is used to determine a match. Bit[0] is for matching Alpha register 0: 1 = only need to compare Y; 0 = compare all Y, Cb and Cr. Bit[1] is for matching Alpha register 1: 1 = only need to compare Y; 0 = compare all Y, Cb and Cr. Bit[2] is for matching Alpha register 2: 1 = only need to compare Y; 0 = compare all Y, Cb and Cr. Bit[3] is for matching Alpha register 3: 1 = only need to compare Y; 0 = compare all Y, Cb and Cr.
4	R/W	0	CTRL_422TO444. 1 = Enable conversion of 422 format input to 444 format output; 0 = Disable 422 to 444 conversion.
/	/	/	/
2	R/W	0	osd_mem_mode : // unsigned , default =0 0: canvas_araddr 1: linear_araddr (mali src must use this mode)

Bit(s)	R/W	Default	Description
1	R/W	0	premult_en : // unsigned , default =0
0	R/W	0	OSD_BLK_ENABLE: Each bit to enable display an OSD block

Table 9-2410 VIU\_OSD1\_CTRL\_STAT2 0x1A2d

Bit(s)	R/W	Default	Description
31-16	R	0	unused
15	R/W	0	unused
14	R/W	0	Replaced_alpha_en
13-6	R/W	0	Replaced_alpha
5-4	R/W	0	Hold_fifo_lines[6:5]
3	R/W	0	RGBYUV_FULL_RANGE: Select coefficients for applicable output range. 1 = output full range 0-255; 0 = output range 16-235.
2	R/W	0	ALPHA_9B_MODE: Define how to expand 8-bit alpha value to 9-bit. 1 = The formula is (Alpha < 128) ? Alpha : Alpha + 1; 0 = The formula is (Alpha == 0) ? Alpha : Alpha + 1.
1	R/W	0	Pedding status cleanup
0	R/W	0	COLOR_EXPAND_MODE. 1 = Expand the color components to 8-bit by padding LSBs with MSBs. E.g. If the input is 5'b11000, the output is expanded to 8'b11000110; 0 = Expand the color components to 8-bit by padding LSBs with 0.

VIU\_OSD1\_TCOLOR\_AG0 0x1A17

VIU\_OSD1\_TCOLOR\_AG1 0x1A18

VIU\_OSD1\_TCOLOR\_AG2 0x1A19

VIU\_OSD1\_TCOLOR\_AG3 0x1A1a

Table 9-2411 Define Alpha register 0/1/2/3 values

Bit(s)	R/W	Default	Description
31-24	R/W	0xFF	Y or R.
23-16	R/W	0xFF	CB or G.
15-8	R/W	0xFF	CR or B.
7-0	R/W	0xFF	ALPHA.

VIU\_OSD1\_BLK0\_CFG\_W0 0x1A1b

Defines display block 0/1/2/3's property, word 0.

Table 9-2412

Bit(s)	R/W	Default	Description
31	R/W	0	Reserved
30	R/W	0	mali_src_en 1: read data from mali afbcd decoder



Bit(s)	R/W	Default	Description
			0: read data from DDR directly
29	R/W	0	y_rev: 0=normal read, 1=reverse read in Y direction
28	R/W	0	x_rev: 0=normal read, 1=reverse read in X direction
27-24	R/W	0	Reserved
23-16	R/W	0	TBL_ADDR. Linear address, TBL_ADDR = data_addr >> 4 bits
15	R/W	0	LITTLE_ENDIAN: define the of data stored in DDR. 1 = Data stored in DDR memory are of little endian; 0 = Data stored in DDR memory are of big endian.
14	R/W	0	RPT_Y: For reducing data size stored in DDR. 1 = For each line, OSD will display twice; 0 = No repeat, OSD display once per line.
13-12	R/W	0	INTERP_CTRL: If enabled, interpolate a data after each incoming pixels, in order to save DDR bandwidth. 0 = No interpolation; 1 = unused, no interpolation; 2 = Interpolate with preceding pixel value; 3 = Interpolate with the averaged value between the preceding pixel and the next pixel.
11-8	R/W	0	OSD_BLK_MODE: Define the OSD block's input pixel format according bit30. Bit30 == 0: 0 = 2-bit per pixel, totally 4 colors can be looked up from color palette LUT, Only OSD2 have; 1 = 4-bit per pixel, totally 16 colors can be looked up from color palette LUT, Only OSD2 have; 2 = 8-bit per pixel, totally 256 colors can be looked up from color palette LUT, Only OSD2 have; 3 = 4:2:2 mode. Input 32-bit data for 2 pixels. Bit[31:24] is Y0, bit [23:16] is Cb0, bit[15:8] is Y1, bit [7:0] is Cr0, for Alpha value refer to reg VIU_OSD1_CTRL_STAT.CTRL_MTCH_Y; 4 = 16-bit mode. Refer to COLOR_MATRIX; 5 = 32-bit mode. Refer to COLOR_MATRIX; 6 = unused; 7 = 24-bit mode. Refer to COLOR_MATRIX; 8-15 = unused; Bit30 == 1: 0 R8 1 8bit yuv422 2 RGB565 3 RGBA5551 4 RGBA4444 5 RGBA8888 7 RGB888 8 10bit yuv422 9 RGBA1010102 Other: unuse
6	R/W	0	TC_ALPHA_EN: refer to reg VIU_OSD1_CTRL_STAT.CTRL_MTCH_Y. 1 = Enable alpha register matching. 0 = Disable.
5-2	R/W	0	COLOR_MATRIX: Applicable only to 16-bit color mode (OSD_BLK_MODE=4), 32-bit mode (OSD_BLK_MODE=5) and 24-bit mode (OSD_BLK_MODE=7), defines the bit-field allocation of the pixel data. For expanding the bit-fields to full 8-bit, refer to VIU_OSD1_CTRL_STAT2.color_expand_mode. For 16-bit mode (OSD_BLK_MODE=4): 0 = 6:5:5 format. Bit[15:10] is Y[7:2] or R [7:2], bit[9:5] is Cb[7:3] or G[7:3], bit[4:0] is Cr[7:3] or B[7:3], for Alpha value refer to reg VIU_OSD1_CTRL_STAT.CTRL_MTCH_Y; 1 = 8:4:4 format. Bit[15:8] is Y or R, bit[7:4] is Cb[7:4] or G[7:4], bit[3:0] is Cr[7:4] or B[7:4], for Alpha value refer to reg VIU_OSD1_CTRL_STAT.CTRL_MTCH_Y; 4 = 5:6:5 format. Bit[15:11] is Y [7:3] or R[7:3], bit[10:5] is Cb[7:2] or G[7:2], bit[4:0] is Cr[7:3] or B[7:3], for Alpha value refer to reg VIU_OSD1_CTRL_STAT.CTRL_MTCH_Y; For 32-bit mode (OSD_BLK_MODE=5): 0 = RGBA 8:8:8:8 format. Bit[31:24] is Y or R, bit[23:16] is Cb or G; bit[15:8] is Cr or B; bit[7:0] is Alpha; 1 = ARGB 8:8:8:8 format. Bit [31:24] is Alpha, bit[23:16] is Y or R; bit[15:8] is Cb or G; bit[7:0] is Cr or B; 2 = ABGR 8:8:8:8 format. Bit[31:24] is Alpha, bit[23:16] is Cr or B; bit[15:8] is Cb or G; bit[7:0] is Y or R; 3 = BGRA 8:8:8:8 format. Bit[31:24] is Cr or B, bit[23:16] is Cb or G; bit[15:8] is Y or R; bit[7:0] is Alpha. For 24-bit mode (OSD_BLK_MODE=7): 0 = RGB 8:8:8 mode. Bit[23:16] is Y or R, bit[15:8] is Cb or G, bit[7:0] is Cr or B, for Alpha value refer to reg VIU_OSD1_CTRL_STAT.CTRL_MTCH_Y;

Bit(s)	R/W	Default	Description
			5 = BGR 8:8:8 mode. Bit[23:16] is Cr or B, bit[15:8] is Cb or G, bit[7:0] is Y or R, for Alpha value refer to reg VIU_OSD1_CTRL_STAT.CTRL_MTCH_Y.
1	R/W	0	INTERLACE_EN. 1 = Enable interlace mode. 0 = Disable.
0	R/W	0	INTERLACE_SEL_ODD: Applicable only if INTERLACE_EN = 1. 1 = Only output odd lines; 0 = Only output even lines.

**Table 9-2413 VIU\_OSD1\_BLK0\_CFG\_W1 0x1A1c**

Defines display block 0/1/2/3' s property, word 1.

Bit(s)	R/W	Default	Description
31-29	R/W	0	Unused.
28-16	R/W	0	X_END. Virtual canvas co-ordinate.
15-13	R/W	0	Unused.
12-0	R/W	0	X_START. Virtual canvas co-ordinate.

**Table 9-2414 VIU\_OSD1\_BLK0\_CFG\_W2 0x1A1d**

Defines display block 0/1/2/3' s property, word 2.

Bit(s)	R/W	Default	Description
31-29	R/W	0	Unused.
28-16	R/W	0	Y_END. Virtual canvas co-ordinate.
15-13	R/W	0	Unused.
12-0	R/W	0	Y_START. Virtual canvas co-ordinate.

**Table 9-2415 VIU\_OSD1\_BLK0\_CFG\_W3 0x1A1e**

Defines display block 0/1/2/3' s property, word 3.

Bit(s)	R/W	Default	Description
31-28	R/W	0	Unused.
27-16	R/W	0	H_END. Display horizontal co-ordinate.
15-12	R/W	0	Unused.
11-0	R/W	0	H_START. Display horizontal co-ordinate.

**Table 9-2416 VIU\_OSD1\_BLK0\_CFG\_W4 0x1A13**

Defines display block 0/1/2/3' s property, word 4.

Bit(s)	R/W	Default	Description
31-28	R/W	0	Unused.
27-16	R/W	0	V_END. Display vertical co-ordinate.
15-12	R/W	0	Unused.
11-0	R/W	0	V_START. Display vertical co-ordinate.

Table 9-2417 VIU\_OSD1\_BLK1\_CFG\_W4 0x1a14

Bit(s)	R/W	Default	Description
31:0	R/W	0	Frame_addr: // unsigned , default =0 Frame_addr in linear_addr

Table 9-2418 VIU\_OSD1\_BLK2\_CFG\_W4 0x1a15

Bit(s)	R/W	Default	Description
31:0	R/W	0	Line_stride : // unsigned , default =0 Line_stride in linear_addr

Table 9-2419 VIU\_OSD1\_FIFO\_CTRL\_STAT 0x1A2b

Bit(s)	R/W	Default	Description
31	R/W	0	burst_len_sel[2] of [2:0]
30	R/W	0	BYTE_SWAP: In addition to endian control, further define whether to swap upper byte and lower byte within a 16-bit memory word. 1 = Swap, data[15:0] becomes {data[7:0], data[15:8]}; 0 = No swap, data[15:0] is still data[15:0].
29	R/W	0	Div_swap : swap the 2 64bits word in 128bits word
28-24	R/W	0	Fifo_lim : when osd fifo is small than the fifo_lim*16, closed the req port of osd_rd_mif
23-22	R/W	0	Fifo_ctrl: 00 : for 1 word in 1 burst, 01 : for 2 words in 1burst, 10 : for 4 words in 1burst, 11: reserved
21-20	R	0	FIFO_ST: State of the FIFO activity. 0 = Idle; 1 = FIFO requesting; 2 = FIFO request aborting.
19	R	0	FIFO_OVERFLOW.
18-12	R/W	32	FIFO_DEPTH_VAL: Define the depth of FIFO which stores 128-bit data from DDR to be FIFO_DEPTH_VAL * 8.
11-10	R/W	0	BURST_LEN_SEL[1:0] of [2:0]: Define DDR burst request length. 0 = up to 24 per burst; 1 = up to 32 per burst; 2 = up to 48 per burst;/ 3 = up to 64 per burst. 4 = up to 96 per burst, 5 = up to 128 per burst
9-5	R/W	4	HOLD_FIFO_LINES: The number of lines that OSD must wait after VSYNC, before it starts request data from DDR .
4	R/W	0	CLEAR_ERR: One pulse to clear error status.
3	R/W	0	FIFO_SYNC_RST: Set 1 to reset OSD FIFO.

Bit(s)	R/W	Default	Description
2-1	R/W	0	ENDIAN: define the endianness of the 64-bit data stored in memory, and how to convert. 0 = No conversion; 1 = Convert to {din[31:0], din[63:32]}; 2 = Convert to {din[15:0], din[31:16], din[47:32], din[63:48]}; 3 = Convert to {din[47:32], din[63:48], din[15:0], din[31:16]};
0	R/W	0	URGENT. 1 = Set DDR request priority to be urgent; 0 = Set DDR request priority to be normal.

**Table 9-2420 VIU\_OSD1\_TEST\_RDDATA 0x1A2c**

During OSD debug mode (VIU\_OSD1\_CTRL\_STAT.TEST\_RD\_EN = 1), the output pixels will be presented at this register.

Bit(s)	R/W	Default	Description
31-24	R	0	Y or R.
23:16	R	0	Cb or G.
15-8	R	0	Cr or B.
7-0	R	0	Alpha[8:1].

**Table 9-2421 VIU\_OSD1\_PROT\_CTRL 0x1a2e**

Bit(s)	R/W	Default	Description
31:16	R/W	0	urgent_ctrl : // unsigned , default =0
15	R.O	0	prot_en : // unsigned , default =0; 1=Borrow PROT's FIFO storage, either for rotate or non-rotate.
12: 0	R.O	0	prot_fifo_size : // unsigned , default =0; effective FIFO size when prot_en=1.

**Table 9-2422 VIU\_OSD1\_MALI\_UNPACK\_CTRL 0x1a2f**

Bit(s)	R/W	Default	Description
31	R/W	0	mali_unpack_en 1: OSD will unpack mali_src 0: OSD will unpack normal src
28			Alpha_div_en: alpha divisor enable
27:26			Alpha_divisor gating clk
25:24			Alpa_mapping_mode In osd,this bit should be set 0 when Alpha_div_en active, means 8 bits alpha mode
17	/	/	/
16	R/W	0	afbcd_swap_64bit:
15: 12	R/W	1	afbcd_r_reorder,change osd output order when use mali src: 1: r_re = r ; 2: r_re = g ; 3: r_re = b ; 4: r_re = a ;

Bit(s)	R/W	Default	Description
			default: r_re = 0;
11: 8		2	afbcd_r_reorder,change osd output order when use mali src: 1: g_re = r ; 2: g_re = g; 3: g_re = b; 4: g_re = a; default: r_re = 0;
7: 4		3	afbcd_r_reorder,change osd output order when use mali src: 1: b_re = r ; 2: b_re = g; 3: b_re = b; 4: b_re = a; default: r_re = 0;
3: 0		4	afbcd_r_reorder,change osd output order when use mali src: 1: a_re = r ; 2: a_re = g; 3: a_re = b; 4: a_re = a; default: r_re = 0;

Table 9-2423 VIU\_OSD1\_DIMM\_CTRL 0x1adf

Bit(s)	R/W	Default	Description
30	R/W	0	OSD dimm enable,osd out will be one color when this bit active
29:0	R/W	0	Osd_dim_rgb_out ,osd out will be this value when bit30 active

### 9.2.3.31 OSD2 Registers

VIU\_OSD2\_CTRL\_STAT 0x1a30

See □VIU\_OSD1\_CTRL\_STAT

VIU\_OSD2\_CTRL\_STAT2 0x1a4d

See □VIU\_OSD1\_CTRL\_STAT2

VIU\_OSD2\_COLOR\_ADDR 0x1a31

See □VIU\_OSD1\_COLOR\_ADDR

VIU\_OSD2\_COLOR 0x1a32

See □VIU\_OSD1\_COLOR

VIU\_OSD2\_TCOLOR\_AG0 0x1a37

See □VIU\_OSD1\_TCOLOR\_AG0

VIU\_OSD2\_TCOLOR\_AG1 0x1a38

See □ VIU\_OSD1\_TCOLOR\_AG1

VIU\_OSD2\_TCOLOR\_AG2 0x1a39

See □ VIU\_OSD1\_TCOLOR\_AG02

VIU\_OSD2\_TCOLOR\_AG3 0x1a3a

See □ VIU\_OSD1\_TCOLOR\_AG3

VIU\_OSD2\_BLK0\_CFG\_W0 0x1a3b

See □ VIU\_OSD1\_BLK0\_CFG\_W0

VIU\_OSD2\_BLK0\_CFG\_W1 0x1a3c

See □ VIU\_OSD1\_BLK0\_CFG\_W1

VIU\_OSD2\_BLK0\_CFG\_W2 0x1a3d

See □ VIU\_OSD1\_BLK0\_CFG\_W2

VIU\_OSD2\_BLK0\_CFG\_W3 0x1a3e

See □ VIU\_OSD1\_BLK0\_CFG\_W3

VIU\_OSD2\_BLK0\_CFG\_W4 0x1a64

See: VIU\_OSD1\_BLK0\_CFG\_W4

VIU\_OSD2\_BLK1\_CFG\_W4 0x1a65

See: VIU\_OSD2\_BLK1\_CFG\_W4

VIU\_OSD2\_BLK2\_CFG\_W4 0x1a66

See: VIU\_OSD1\_BLK2\_CFG\_W4

VIU\_OSD2\_FIFO\_CTRL\_STAT 0x1a4b

See: VIU\_OSD1\_FIFO\_CTRL\_STAT

VIU\_OSD2\_TEST\_RDDATA 0x1a4c

See: VIU\_OSD1\_TEST\_RDDATA

VIU\_OSD2\_PROT\_CTRL 0x1a4e

See: VIU\_OSD1\_PROT\_CTRL

VIU\_OSD2\_MALI\_UNPACK\_CTRL 0x1abd

See: VIU\_OSD1\_MALI\_UNPACK\_CTRL

VIU\_OSD2\_DIMM\_CTRL 0x1acf

See: VIU\_OSD1\_DIMM\_CTRL

### 9.2.3.32 OSD3 Registers

VIU\_OSD3\_CTRL\_STAT 0x3d80

See □ VIU\_OSD1\_CTRL\_STAT

VIU\_OSD3\_CTRL\_STAT2 0x3d81

See □ VIU\_OSD1\_CTRL\_STAT2

VIU\_OSD3\_COLOR\_ADDR 0x3d82

See □ VIU\_OSD1\_COLOR\_ADDR

VIU\_OSD3\_COLOR 0x3d83

See □ VIU\_OSD1\_COLOR

VIU\_OSD3\_TCOLOR\_AG0 0x3d84

See □ VIU\_OSD1\_TCOLOR\_AG0

VIU\_OSD3\_TCOLOR\_AG1 0x3d85

See □ VIU\_OSD1\_TCOLOR\_AG1

VIU\_OSD3\_TCOLOR\_AG2 0x3d86

See □ VIU\_OSD1\_TCOLOR\_AG02

VIU\_OSD3\_TCOLOR\_AG3 0x3d87

See □ VIU\_OSD1\_TCOLOR\_AG3

VIU\_OSD3\_BLK0\_CFG\_W0 0x3d88

See □ VIU\_OSD1\_BLK0\_CFG\_W0

VIU\_OSD3\_BLK0\_CFG\_W1 0x3d8c

See □ VIU\_OSD1\_BLK0\_CFG\_W1

VIU\_OSD3\_BLK0\_CFG\_W2 0x3d90

See □ VIU\_OSD1\_BLK0\_CFG\_W2

VIU\_OSD3\_BLK0\_CFG\_W3 0x3d94

See □ VIU\_OSD1\_BLK0\_CFG\_W3

VIU\_OSD3\_BLK0\_CFG\_W4 0x3d98

See: VIU\_OSD1\_BLK0\_CFG\_W4  
VIU\_OSD3\_BLK1\_CFG\_W4 0x3d99  
See: VIU\_OSD3\_BLK1\_CFG\_W4  
VIU\_OSD3\_BLK2\_CFG\_W4 0x3d9a  
See: VIU\_OSD1\_BLK2\_CFG\_W4

VIU\_OSD3\_FIFO\_CTRL\_STAT 0x3d9c  
See: VIU\_OSD1\_FIFO\_CTRL\_STAT  
VIU\_OSD3\_TEST\_RDDATA 0x3d9d  
See: VIU\_OSD1\_TEST\_RDDATA  
VIU\_OSD3\_PROT\_CTRL 0x3d9e  
See: VIU\_OSD1\_PROT\_CTRL

VIU\_OSD3\_MALI\_UNPACK\_CTRL 0x3d9f  
See: VIU\_OSD1\_MALI\_UNPACK\_CTRL  
VIU\_OSD3\_DIMM\_CTRL 0x3da0  
See: VIU\_OSD1\_DIMM\_CTRL

### 9.2.3.33 OSD4 Registers

VIU\_OSD4\_CTRL\_STAT 0x3dc0  
See □ VIU\_OSD1\_CTRL\_STAT  
VIU\_OSD4\_CTRL\_STAT2 0x3dc1

See □ VIU\_OSD1\_CTRL\_STAT2  
VIU\_OSD4\_COLOR\_ADDR 0x3dc2

See □ VIU\_OSD1\_COLOR\_ADDR  
VIU\_OSD4\_COLOR 0x3dc3

See □ VIU\_OSD1\_COLOR  
VIU\_OSD4\_TCOLOR\_AG0 0x3dc4

See □ VIU\_OSD1\_TCOLOR\_AG0  
VIU\_OSD4\_TCOLOR\_AG1 0x3dc5

See □ VIU\_OSD1\_TCOLOR\_AG1  
VIU\_OSD4\_TCOLOR\_AG2 0x3dc6



See □ VIU\_OSD1\_TCOLOR\_AG02

VIU\_OSD4\_TCOLOR\_AG3      0x3dc7

See □ VIU\_OSD1\_TCOLOR\_AG3

VIU\_OSD4\_BLK0\_CFG\_W0      0x3dc8

See □ VIU\_OSD1\_BLK0\_CFG\_W0

VIU\_OSD4\_BLK0\_CFG\_W1      0x3dcc

See □ VIU\_OSD1\_BLK0\_CFG\_W1

VIU\_OSD4\_BLK0\_CFG\_W2      0x3dd0

See □ VIU\_OSD1\_BLK0\_CFG\_W2

VIU\_OSD4\_BLK0\_CFG\_W3      0x3dd4

See □ VIU\_OSD1\_BLK0\_CFG\_W3

VIU\_OSD4\_BLK0\_CFG\_W4      0x3dd8

See: VIU\_OSD1\_BLK0\_CFG\_W4

VIU\_OSD4\_BLK1\_CFG\_W4      0x3dd9

See: VIU\_OSD3\_BLK1\_CFG\_W4

VIU\_OSD4\_BLK2\_CFG\_W4      0x3dda

See: VIU\_OSD1\_BLK2\_CFG\_W4

VIU\_OSD4\_FIFO\_CTRL\_STAT    0x3ddc

See: VIU\_OSD1\_FIFO\_CTRL\_STAT

VIU\_OSD4\_TEST\_RDDATA      0x3ddd

See: VIU\_OSD1\_TEST\_RDDATA

VIU\_OSD4\_PROT\_CTRL        0x3dde

See: VIU\_OSD1\_PROT\_CTRL

VIU\_OSD4\_MALI\_UNPACK\_CTRL      0x3ddf

See: VIU\_OSD1\_MALI\_UNPACK\_CTRL

VIU\_OSD4\_DIMM\_CTRL      0x3de0

See: VIU\_OSD1\_DIMM\_CTRL

### 9.2.3.34 VPP\_VD1\_MATRIX Registers

**Table 9-2424 VPP\_VD1\_MATRIX\_COEF00\_01 0x3290**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient00, signed, 3.10
12-0	R/W	0	Coefficient01, signed, 3.10

**Table 9-2425 VPP\_VD1\_MATRIX\_COEF02\_10 0x3291**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient02, signed, 3.10
12-0	R/W	0	Coefficient10, signed, 3.10

**Table 9-2426 VPP\_VD1\_MATRIX\_COEF11\_12 0x3292**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient11, signed, 3.10
12-0	R/W	0	Coefficient12, signed, 3.10

**Table 9-2427 VPP\_VD1\_MATRIX\_COEF20\_21 0x3293**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient20, signed, 3.10
12-0	R/W	0	Coefficient21, signed, 3.10

**Table 9-2428 VPP\_VD1\_MATRIX\_COEF22 0x3294**

Bit(s)	R/W	Default	Description
18-16	R/W	0	convrs
12-0	R/W	0	Coefficient22, signed, 3.10

**Table 9-2429 VPP\_VD1\_MATRIX\_COEF13\_14 0x3295**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient13, signed, 3.10
12-0	R/W	0	Coefficient14, signed, 3.10

**Table 9-2430 VPP\_VD1\_MATRIX\_COEF23\_24 0x3296**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient23, signed, 3.10
12-0	R/W	0	Coefficient24, signed, 3.10

**Table 9-2431 VPP\_VD1\_MATRIX\_COEF15\_25 0x3297**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient15, signed, 3.10
12-0	R/W	0	Coefficient25, signed, 3.10

**Table 9-2432 VPP\_VD1\_MATRIX\_CLIP 0x3298**

Bit(s)	R/W	Default	Description
31-8	R/W	0x0	reserved
7-5	R/W	0x1	Matrix rs
4-3	R/W	0x10	Matrix clmod 0: only 3x3, 1: pre_offseted_ch1,ch2> pre_offseted_ch0, use the added 2x3 coef 2: pre_offseted_ch1,ch2 > 0, use the added 2x3 coef 3: pre_offseted_ch1,ch2 > 512, use the added 2x3 coef
2-0	R/W	0x10	/

**Table 9-2433 VPP\_VD1\_MATRIX\_OFFSET0\_1 0x3299**

Bit(s)	R/W	Default	Description
27-16	R/W	0	Offset0, signed value
11-0	R/W	0	Offset1, signed value

**Table 9-2434 VPP\_VD1\_MATRIX\_OFFSET2 0x329a**

Bit(s)	R/W	Default	Description
11-0	R/W	0	Offset2, signed value

**Table 9-2435 VPP\_VD1\_MATRIX\_PRE\_OFFSET0\_1 0x329b**

Bit(s)	R/W	Default	Description
27-16	R/W	0	pre_Offset0, signed value
11-0	R/W	0	Pre_Offset1, signed value

**Table 9-2436 VPP\_VD1\_MATRIX\_PRE\_OFFSET2 0x329c**

Bit(s)	R/W	Default	Description
11-0	R/W	0	Pre_Offset2, signed value

**Table 9-2437 VPP\_VD1\_MATRIX\_EN\_CTRL 0x329d**

Bit(s)	R/W	Default	Description
5:4	R/W	0	Gate clock ctrl //matrix gating clock ctrl 1x:free clk 00:gating clk 01:close clk
1	R/W	0	Enable_sync_sel //matrix enable sync signal 1:enable sync by vsync 0:enable active directly
0	R/W	0	Conv_en_pre //matrix enable signal ,active high

### 9.2.3.35 VPP\_POST\_MATRIX Registers

VPP\_POST\_MATRIX\_COEF00\_01 0x32b0

See: VPP\_VD1\_MATRIX\_COEF00\_01

VPP\_POST\_MATRIX\_COEF02\_10 0x32b1

See: VPP\_VD1\_MATRIX\_COEF02\_10

VPP\_POST\_MATRIX\_COEF11\_12 0x32b2

See: VPP\_VD1\_MATRIX\_COEF11\_12

VPP\_POST\_MATRIX\_COEF20\_21 0x32b3

See: VPP\_VD1\_MATRIX\_COEF20\_21

VPP\_POST\_MATRIX\_COEF22 0x32b4

See: VPP\_VD1\_MATRIX\_COEF22

VPP\_POST\_MATRIX\_COEF13\_14 0x32b5

See: VPP\_VD1\_MATRIX\_COEF13\_14

VPP\_POST\_MATRIX\_COEF23\_24 0x32b6

See: VPP\_VD1\_MATRIX\_COEF23\_24

VPP\_POST\_MATRIX\_COEF15\_25 0x32b7

See: VPP\_VD1\_MATRIX\_COEF15\_25

VPP\_POST\_MATRIX\_CLIP 0x32b8

See: VPP\_VD1\_MATRIX\_CLIP

VPP\_POST\_MATRIX\_OFFSET0\_1 0x32b9

See: VPP\_VD1\_MATRIX\_OFFSET0\_1

VPP\_POST\_MATRIX\_OFFSET2 0x32ba

See: VPP\_VD1\_MATRIX\_OFFSET2

VPP\_POST\_MATRIX\_PRE\_OFFSET0\_1 0x32bb

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET0\_1

VPP\_POST\_MATRIX\_PRE\_OFFSET2 0x32bc

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET2

VPP\_POST\_MATRIX\_EN\_CTRL 0x32bd

See: VPP\_VD1\_MATRIX\_EN\_CTRL

**Table 9-2438 VPP\_POST\_MATRIX\_SAT 0x32c1**

Bit(s)	R/W	Default	Description
2	R/W	0	sat_o_en: If true, the output result of post matrix output is saturated
1	R/W	0	sat_i_en: If true, the input result of post matrix output is saturated
0	R/W	0	misc_sat_en: same as sat_i_en

### 9.2.3.36 VPP\_POST2\_MATRIX Registers

VPP\_POST2\_MATRIX\_COEF00\_01 0x39a0

See: VPP\_VD1\_MATRIX\_COEF00\_01

VPP\_POST2\_MATRIX\_COEF02\_10 0x39a1

See: VPP\_VD1\_MATRIX\_COEF02\_10

VPP\_POST2\_MATRIX\_COEF11\_12 0x39a2

See: VPP\_VD1\_MATRIX\_COEF11\_12

VPP\_POST2\_MATRIX\_COEF20\_21 0x39a3

See: VPP\_VD1\_MATRIX\_COEF20\_21

VPP\_POST2\_MATRIX\_COEF22 0x39a4

See: VPP\_VD1\_MATRIX\_COEF22

VPP\_POST2\_MATRIX\_COEF13\_14 0x39a5

See: VPP\_VD1\_MATRIX\_COEF13\_14

VPP\_POST2\_MATRIX\_COEF23\_24 0x39a6

See: VPP\_VD1\_MATRIX\_COEF23\_24

VPP\_POST2\_MATRIX\_COEF15\_25 0x39a7

See: VPP\_VD1\_MATRIX\_COEF15\_25

VPP\_POST2\_MATRIX\_CLIP 0x39a8

See: VPP\_VD1\_MATRIX\_CLIP

VPP\_POST2\_MATRIX\_OFFSET0\_1 0x39a9

See: VPP\_VD1\_MATRIX\_OFFSET0\_1

VPP\_POST2\_MATRIX\_OFFSET2 0x39aa

See: VPP\_VD1\_MATRIX\_OFFSET2

VPP\_POST2\_MATRIX\_PRE\_OFFSET0\_1 0x39ab

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET0\_1

VPP\_POST2\_MATRIX\_PRE\_OFFSET2 0x39ac  
See: VPP\_VD1\_MATRIX\_PRE\_OFFSET2

VPP\_POST2\_MATRIX\_EN\_CTRL 0x39ad  
See: VPP\_VD1\_MATRIX\_EN\_CTRL

### 9.2.3.37 VPP\_OSD2\_MATRIX Registers

VPP\_OSD2\_MATRIX\_COEF00\_01 0x3920

See: VPP\_VD1\_MATRIX\_COEF00\_01

VPP\_OSD2\_MATRIX\_COEF02\_10 0x3921

See: VPP\_VD1\_MATRIX\_COEF02\_10

VPP\_OSD2\_MATRIX\_COEF11\_12 0x3922

See: VPP\_VD1\_MATRIX\_COEF11\_12

VPP\_OSD2\_MATRIX\_COEF20\_21 0x3923

See: VPP\_VD1\_MATRIX\_COEF20\_21

VPP\_OSD2\_MATRIX\_COEF22 0x3924

See: VPP\_VD1\_MATRIX\_COEF22

VPP\_OSD2\_MATRIX\_COEF13\_14 0x3925

See: VPP\_VD1\_MATRIX\_COEF13\_14

VPP\_OSD2\_MATRIX\_COEF23\_24 0x3926

See: VPP\_VD1\_MATRIX\_COEF23\_24

VPP\_OSD2\_MATRIX\_COEF15\_25 0x3927

See: VPP\_VD1\_MATRIX\_COEF15\_25

VPP\_OSD2\_MATRIX\_CLIP 0x3928

See: VPP\_VD1\_MATRIX\_CLIP

VPP\_OSD2\_MATRIX\_OFFSET0\_1 0x3929

See: VPP\_VD1\_MATRIX\_OFFSET0\_1

VPP\_OSD2\_MATRIX\_OFFSET2 0x392a

See: VPP\_VD1\_MATRIX\_OFFSET2

VPP\_OSD2\_MATRIX\_PRE\_OFFSET0\_1 0x392b

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET0\_1

VPP\_OSD2\_MATRIX\_PRE\_OFFSET2 0x392c

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET2

VPP\_OSD2\_MATRIX\_EN\_CTRL 0x392d

See: VPP\_VD1\_MATRIX\_EN\_CTRL

### 9.2.3.38 VPP\_WRAP\_OSD1\_MATRIX Registers

VPP\_WRAP\_OSD1\_MATRIX\_COEF00\_01 0x3d60

See: VPP\_VD1\_MATRIX\_COEF00\_01

VPP\_WRAP\_OSD1\_MATRIX\_COEF02\_10 0x3d61

See: VPP\_VD1\_MATRIX\_COEF02\_10

VPP\_WRAP\_OSD1\_MATRIX\_COEF11\_12 0x3d62

See: VPP\_VD1\_MATRIX\_COEF11\_12

VPP\_WRAP\_OSD1\_MATRIX\_COEF20\_21 0x3d63

See: VPP\_VD1\_MATRIX\_COEF20\_21

VPP\_WRAP\_OSD1\_MATRIX\_COEF22 0x3d64

See: VPP\_VD1\_MATRIX\_COEF22

VPP\_WRAP\_OSD1\_MATRIX\_COEF13\_14 0x3d65

See: VPP\_VD1\_MATRIX\_COEF13\_14

VPP\_WRAP\_OSD1\_MATRIX\_COEF23\_24 0x3d66

See: VPP\_VD1\_MATRIX\_COEF23\_24

VPP\_WRAP\_OSD1\_MATRIX\_COEF15\_25 0x3d67

See: VPP\_VD1\_MATRIX\_COEF15\_25

VPP\_WRAP\_OSD1\_MATRIX\_CLIP 0x3d68

See: VPP\_VD1\_MATRIX\_CLIP

VPP\_WRAP\_OSD1\_MATRIX\_OFFSET0\_1 0x3d69

See: VPP\_VD1\_MATRIX\_OFFSET0\_1

VPP\_WRAP\_OSD1\_MATRIX\_OFFSET2 0x3d6a

See: VPP\_VD1\_MATRIX\_OFFSET2

VPP\_WRAP\_OSD1\_MATRIX\_PRE\_OFFSET0\_1 0x3d6b

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET0\_1

VPP\_WRAP\_OSD1\_MATRIX\_PRE\_OFFSET2 0x3d6c

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET2

VPP\_WRAP\_OSD1\_MATRIX\_EN\_CTRL 0x3d6d

See: VPP\_VD1\_MATRIX\_EN\_CTRL

### 9.2.3.39 VPP\_WRAP\_OSD2\_MATRIX Registers

VPP\_WRAP\_OSD2\_MATRIX\_COEF00\_01 0x3d70

See: VPP\_VD1\_MATRIX\_COEF00\_01

VPP\_WRAP\_OSD2\_MATRIX\_COEF02\_10 0x3d71

See: VPP\_VD1\_MATRIX\_COEF02\_10

VPP\_WRAP\_OSD2\_MATRIX\_COEF11\_12 0x3d72

See: VPP\_VD1\_MATRIX\_COEF11\_12

VPP\_WRAP\_OSD2\_MATRIX\_COEF20\_21 0x3d73

See: VPP\_VD1\_MATRIX\_COEF20\_21

VPP\_WRAP\_OSD2\_MATRIX\_COEF22 0x3d74

See: VPP\_VD1\_MATRIX\_COEF22

VPP\_WRAP\_OSD2\_MATRIX\_COEF13\_14 0x3d75

See: VPP\_VD1\_MATRIX\_COEF13\_14

VPP\_WRAP\_OSD2\_MATRIX\_COEF23\_24 0x3d76

See: VPP\_VD1\_MATRIX\_COEF23\_24

VPP\_WRAP\_OSD2\_MATRIX\_COEF15\_25 0x3d77

See: VPP\_VD1\_MATRIX\_COEF15\_25

VPP\_WRAP\_OSD2\_MATRIX\_CLIP 0x3d78

See: VPP\_VD1\_MATRIX\_CLIP

VPP\_WRAP\_OSD2\_MATRIX\_OFFSET0\_1 0x3d79

See: VPP\_VD1\_MATRIX\_OFFSET0\_1

VPP\_WRAP\_OSD2\_MATRIX\_OFFSET2 0x3d7a

See: VPP\_VD1\_MATRIX\_OFFSET2

VPP\_WRAP\_OSD2\_MATRIX\_PRE\_OFFSET0\_1 0x3d7b

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET0\_1

VPP\_WRAP\_OSD2\_MATRIX\_PRE\_OFFSET2 0x3d7c

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET2

VPP\_WRAP\_OSD2\_MATRIX\_EN\_CTRL 0x3d7d

See: VPP\_VD1\_MATRIX\_EN\_CTRL

### 9.2.3.40 VPP\_WRAP\_OSD3\_MATRIX Registers

VPP\_WRAP\_OSD3\_MATRIX\_COEF00\_01 0x3db0

See: VPP\_VD1\_MATRIX\_COEF00\_01



VPP\_WRAP\_OSD3\_MATRIX\_COEF02\_10 0x3db1

See: VPP\_VD1\_MATRIX\_COEF02\_10

VPP\_WRAP\_OSD3\_MATRIX\_COEF11\_12 0x3db2

See: VPP\_VD1\_MATRIX\_COEF11\_12

VPP\_WRAP\_OSD3\_MATRIX\_COEF20\_21 0x3db3

See: VPP\_VD1\_MATRIX\_COEF20\_21

VPP\_WRAP\_OSD3\_MATRIX\_COEF22 0x3db4

See: VPP\_VD1\_MATRIX\_COEF22

VPP\_WRAP\_OSD3\_MATRIX\_COEF13\_14 0x3db5

See: VPP\_VD1\_MATRIX\_COEF13\_14

VPP\_WRAP\_OSD3\_MATRIX\_COEF23\_24 0x3db6

See: VPP\_VD1\_MATRIX\_COEF23\_24

VPP\_WRAP\_OSD3\_MATRIX\_COEF15\_25 0x3db7

See: VPP\_VD1\_MATRIX\_COEF15\_25

VPP\_WRAP\_OSD3\_MATRIX\_CLIP 0x3db8

See: VPP\_VD1\_MATRIX\_CLIP

VPP\_WRAP\_OSD3\_MATRIX\_OFFSET0\_1 0x3db9

See: VPP\_VD1\_MATRIX\_OFFSET0\_1

VPP\_WRAP\_OSD3\_MATRIX\_OFFSET2 0x3dba

See: VPP\_VD1\_MATRIX\_OFFSET2

VPP\_WRAP\_OSD3\_MATRIX\_PRE\_OFFSET0\_1 0x3dbb

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET0\_1

VPP\_WRAP\_OSD3\_MATRIX\_PRE\_OFFSET2 0x3dbc

See: VPP\_VD1\_MATRIX\_PRE\_OFFSET2

VPP\_WRAP\_OSD3\_MATRIX\_EN\_CTRL 0x3dbd

See: VPP\_VD1\_MATRIX\_EN\_CTRL

### 9.2.3.41 HDR Registers

**Table 9-2439 VDIN0\_HDR2\_CTRL 0x1280**

Bit(s)	R/W	Default	Description
20:18	R/W	0	reg_din_swap : // unsigned , default = 0
17	R/W	0	reg_out_fmt : // unsigned , default = 0
16	R/W	0	reg_only_mat : // unsigned , default = 0 ,only use input matrix ,work when hdr disable

Bit(s)	R/W	Default	Description
15	R/W	0	mat_o_en, //output matrix enable ,only work when hdr enable
14	R/W	0	mat_in_en //input matrix enable ,only work when hdr enable
13	R/W	0	reg_VDIN0_HDR2_top_en : // unsigned , default = 0, hdr enable signal
12	R/W	1	reg_cgain_mode : // unsigned , default = 1
7: 6	R/W	1	reg_gmut_mode : // unsigned , default = 1
5	R/W	0	reg_in_shift : // unsigned , default = 0
4	R/W	1	reg_in_fmt : // unsigned , default = 1
3	R/W	1	reg_eo_enable : // unsigned , default = 1
2	R/W	1	reg_oe_enable : // unsigned , default = 1
1	R/W	1	reg_ogain_enable : // unsigned , default = 1
0	R/W	1	reg_cgain_enable : // unsigned , default = 1

Table 9-2440 VDIN0\_HDR2\_CLK\_GATE 0x1281

Bit(s)	R/W	Default	Description
31:30	R/W	0	clk_tm : gate clock ctrl (main clock) // unsigned , default = 0
29:28	R/W	0	output : matrix clock gate ctrl // unsigned , default = 0
25:24	R/W	0	input : matrix clock gate ctrl // unsigned , default = 0
23:22	R/W	0	hdr : top cbus clock gate ctrl // unsigned , default = 0
21:20	R/W	0	eotf : cbus clock gate ctrl // unsigned , default = 0
19:18	R/W	0	oetf : cbus clock gate ctrl // unsigned , default = 0
17:16	R/W	0	gamma : mult cbus clock gate ctrl // unsigned , default = 0
15:14	R/W	0	adaptive : cbus scaler clock gate ctrl // unsigned , default = 0
13:12	R/W	0	cgain : cbus clock gate ctrl // unsigned , default = 0
11:10	R/W	0	eotf : clock gate ctrl // unsigned , default = 0
9:8	R/W	0	oetf : clock gate ctrl // unsigned , default = 0
7:6	R/W	0	gamma : mult clock gate ctrl // unsigned , default = 0
5:4	R/W	0	adaptive : scaler clock gate ctrl // unsigned , default = 0
3:2	R/W	0	uv : gain clock gate ctrl // unsigned , default = 0
1:0	R/W	0	cgain : clock gate ctrl // unsigned , default = 0

Table 9-2441 VDIN0\_HDR2\_MATRIXI\_COEF00\_01 0x1282

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

**Table 9-2442 VDIN0\_HDR2\_MATRIXI\_COEF02\_10 0x1283**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

**Table 9-2443 VDIN0\_HDR2\_MATRIXI\_COEF11\_12 0x1284**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-2444 VDIN0\_HDR2\_MATRIXI\_COEF20\_21 0x1285**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-2445 VDIN0\_HDR2\_MATRIXI\_COEF22 0x1286**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-2446 VDIN0\_HDR2\_MATRIXI\_COEF30\_31 0x1287**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-2447 VDIN0\_HDR2\_MATRIXI\_COEF32\_40 0x1288**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-2448 VDIN0\_HDR2\_MATRIXI\_COEF41\_42 0x1289**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-2449 VDIN0\_HDR2\_MATRIXI\_OFFSET0\_1 0x128A**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-2450 VDIN0\_HDR2\_MATRIXI\_OFFSET2 0x128B**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-2451 VDIN0\_HDR2\_MATRIXI\_PRE\_OFFSET0\_1 0x128C**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-2452 VDIN0\_HDR2\_MATRIXI\_PRE\_OFFSET2 0x128D**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-2453 VDIN0\_HDR2\_MATRIXO\_COEF00\_01 0x128E**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

**Table 9-2454 VDIN0\_HDR2\_MATRIXO\_COEF02\_10 0x128F**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

**Table 9-2455 VDIN0\_HDR2\_MATRIXO\_COEF11\_12 0x1290**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-2456 VDIN0\_HDR2\_MATRIXO\_COEF20\_21 0x1291**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-2457 VDIN0\_HDR2\_MATRIXO\_COEF22 0x1292**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-2458 VDIN0\_HDR2\_MATRIXO\_COEF30\_31 0x1293**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-2459 VDIN0\_HDR2\_MATRIXO\_COEF32\_40 0x1294**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-2460 VDIN0\_HDR2\_MATRIXO\_COEF41\_42 0x1295**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-2461 VDIN0\_HDR2\_MATRIXO\_OFFSET0\_1 0x1296**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-2462 VDIN0\_HDR2\_MATRIXO\_OFFSET2 0x1297**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-2463 VDIN0\_HDR2\_MATRIXO\_PRE\_OFFSET0\_1 0x1298**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-2464 VDIN0\_HDR2\_MATRIXO\_PRE\_OFFSET2 0x1299**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-2465 VDIN0\_HDR2\_MATRIXI\_CLIP 0x129A**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-2466 VDIN0\_HDR2\_MATRIXO\_CLIP 0x129B**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-2467 VDIN0\_HDR2\_CGAIN\_OFFT 0x129C**

Bit(s)	R/W	Default	Description
26:16	R/W	0	reg_cgain_offt2 : // signed , default = 0
10:0	R/W	0	reg_cgain_offt1 : // signed , default = 0

**Table 9-2468 VDIN0\_EOTF\_LUT\_ADDR\_PORT 0x129E**

Bit(s)	R/W	Default	Description
7:0	R/W	0	eotf_lut_addr : // unsigned , default = 0

**Table 9-2469 VDIN0\_EOTF\_LUT\_DATA\_PORT 0x129F**

Bit(s)	R/W	Default	Description
19:0	R/W	0	eotf_lut_data : // unsigned , default = 0

**Table 9-2470 VDIN0\_OETF\_LUT\_ADDR\_PORT 0x12A0**

Bit(s)	R/W	Default	Description
7:0	R/W	0	oetf_lut_addr : // unsigned , default = 0

**Table 9-2471 VDIN0\_OETF\_LUT\_DATA\_PORT 0x12A1**

Bit(s)	R/W	Default	Description
11:0	R/W	0	oetf_lut_data : // unsigned , default = 0

**Table 9-2472 VDIN0\_CGAIN\_LUT\_ADDR\_PORT 0x12A2**

Bit(s)	R/W	Default	Description
7:0	R/W	0	cgain_lut_addr : // unsigned , default = 0

**Table 9-2473 VDIN0\_CGAIN\_LUT\_DATA\_PORT 0x12A3**

Bit(s)	R/W	Default	Description
11:0	R/W	0	cgain_lut_data : // unsigned , default = 0

**Table 9-2474 VDIN0\_HDR2\_CGAIN\_COEF0 0x12A4**

Bit(s)	R/W	Default	Description
27:16	R/W	0	reg_cgain_coef1 : // unsigned , default = 0
11:0	R/W	0	reg_cgain_coef0 : //unsigned , default = 0

**Table 9-2475 VDIN0\_HDR2\_CGAIN\_COEF1 0x12A5**

Bit(s)	R/W	Default	Description
11:0	R/W	0	reg_cgain_coef2 : // unsigned , default = 0

**Table 9-2476 VDIN0\_OGAIN\_LUT\_ADDR\_PORT 0x12A6**

Bit(s)	R/W	Default	Description
7:0	R/W	0	ogain_lut_addr : // unsigned , default = 0

**Table 9-2477 VDIN0\_OGAIN\_LUT\_DATA\_PORT 0x12A7**

Bit(s)	R/W	Default	Description
11:0	R/W	0	ogain_lut_data : // unsigned , default = 0

**Table 9-2478 VDIN0\_HDR2\_ADPS\_CTRL 0x12A8**

Bit(s)	R/W	Default	Description
6	R/W	1	reg_adpscl_bypass2 : // unsigned , default = 1
5	R/W	1	reg_adpscl_bypass1 : // unsigned , default = 1
4	R/W	1	reg_adpscl_bypass0 : // unsigned , default = 1
1:0	R/W	1	reg_adpscl_mode : // unsigned , default = 1

**Table 9-2479 VDIN0\_HDR2\_ADPS\_ALPHA0 0x12A9**

Bit(s)	R/W	Default	Description
29:16	R/W	0x1000	reg_adpscl_alpha1 : // unsigned , default = 0x1000
13:0	R/W	0x1000	reg_adpscl_alpha0 : // unsigned , default = 0x1000

**Table 9-2480 VDIN0\_HDR2\_ADPS\_ALPHA1 0x12AA**

Bit(s)	R/W	Default	Description
27:24	R/W	0xc	reg_adpscl_shift0 : // unsigned , default = 0xc
23:20	R/W	0xc	reg_adpscl_shift1 : // unsigned , default = 0xc
19:16	R/W	0xc	reg_adpscl_shift2 : // unsigned , default = 0xc
13:0	R/W	0x1000	reg_adpscl_alpha2 : // unsigned , default = 0x1000

**Table 9-2481 VDIN0\_HDR2\_ADPS\_BETA0 0x12AB**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta0_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta0 : // unsigned , default = 0xfc000

**Table 9-2482 VDIN0\_HDR2\_ADPS\_BETA1 0x12AC**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta1_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta1 : // unsigned , default = 0xfc000

**Table 9-2483 VDIN0\_HDR2\_ADPS\_BETA2 0x12AD**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta2_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta2 : // unsigned , default = 0xfc000



**Table 9-2484 VDIN0\_HDR2\_ADPS\_COEF0 0x12AE**

Bit(s)	R/W	Default	Description
27:16	R/W	460	reg_adpscl_ys_coef1 : // unsigned , default = 460
11:0	R/W	1188	reg_adpscl_ys_coef0 : // unsigned , default = 1188

**Table 9-2485 VDIN0\_HDR2\_ADPS\_COEF1 0x12AF**

Bit(s)	R/W	Default	Description
11:0	R/W	104	reg_adpscl_ys_coef2 : // unsigned , default = 104

**Table 9-2486 VDIN0\_HDR2\_GMUT\_CTRL 0x12B0**

Bit(s)	R/W	Default	Description
3:0	R/W	14	reg_gmut_shift : // unsigned , default = 14

**Table 9-2487 VDIN0\_HDR2\_GMUT\_COEF0 0x12B1**

Bit(s)	R/W	Default	Description
31:16	R/W	674	reg_gmut_coef01 : // unsigned , default = 674
15:0	R/W	1285	reg_gmut_coef00 : // unsigned , default = 1285

**Table 9-2488 VDIN0\_HDR2\_GMUT\_COEF1 0x12B2**

Bit(s)	R/W	Default	Description
31:16	R/W	142	reg_gmut_coef10 : // unsigned , default = 142
15:0	R/W	89	reg_gmut_coef02 : // unsigned , default = 89

**Table 9-2489 VDIN0\_HDR2\_GMUT\_COEF2 0x12B3**

Bit(s)	R/W	Default	Description
31:16	R/W	23	reg_gmut_coef12 : // unsigned , default = 23
15:0	R/W	1883	reg_gmut_coef11 : // unsigned , default = 1883

**Table 9-2490 VDIN0\_HDR2\_GMUT\_COEF3 0x12B4**

Bit(s)	R/W	Default	Description
31:16	R/W	180	reg_gmut_coef21 : // unsigned , default = 180
15:0	R/W	34	reg_gmut_coef20 : // unsigned , default = 34

**Table 9-2491 VDIN0\_HDR2\_GMUT\_COEF4 0x12B5**

Bit(s)	R/W	Default	Description
15:0	R/W	1834	reg_gmut_coef22 : // unsigned , default = 1834

**Table 9-2492 VDIN0\_HDR2\_PIPE\_CTRL1 0x12B6**

Bit(s)	R/W	Default	Description
31:24	R/W	4	vblank_num_oetf : // unsigned , default = 4
23:16	R/W	4	hblank_num_oetf : // unsigned , default = 4
15:8	R/W	10	vblank_num_eotf : // unsigned , default = 10
7:0	R/W	10	hblank_num_eotf : // unsigned , default = 10

**Table 9-2493 VDIN0\_HDR2\_PIPE\_CTRL2 0x12B7**

Bit(s)	R/W	Default	Description
31:24	R/W	10	vblank_num_cgain : // unsigned , default = 10
23:16	R/W	10	hblank_num_cgain : // unsigned , default = 10
15:8	R/W	11	vblank_num_gmut : // unsigned , default = 11
7:0	R/W	11	hblank_num_gmut : // unsigned , default = 11

**Table 9-2494 VDIN0\_HDR2\_PIPE\_CTRL3 0x12B8**

Bit(s)	R/W	Default	Description
31:24	R/W	22	vblank_num_adps : // unsigned , default = 22
23:16	R/W	2	hblank_num_adps : // unsigned , default = 2
15:8	R/W	4	vblank_num_uv : // unsigned , default = 4
7:0	R/W	4	hblank_num_uv : // unsigned , default = 4

**Table 9-2495 VDIN0\_HDR2\_PROC\_WIN1 0x12B9**

Bit(s)	R/W	Default	Description
28:16	R/W	0	proc_win_h_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_h_st : // unsigned , default = 0

**Table 9-2496 VDIN0\_HDR2\_PROC\_WIN2 0x12BA**

Bit(s)	R/W	Default	Description
31	R/W	0	proc_win_gmut_en : // unsigned , default = 0
30	R/W	0	proc_win_adps_en : // unsigned , default = 0
29	R/W	0	proc_win_cgain_en : // unsigned , default = 0

Bit(s)	R/W	Default	Description
28:16	R/W	0	proc_win_v_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_v_st : // unsigned , default = 0

Table 9-2497 VDIN0\_HDR2\_MATRIXI\_EN\_CTRL 0x12BB

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

Table 9-2498 VDIN0\_HDR2\_MATRIXO\_EN\_CTRL 0x12BC

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

Table 9-2499 VDIN1\_HDR2\_CTRL 0x1380

Bit(s)	R/W	Default	Description
20:18	R/W	0	reg_din_swap : // unsigned , default = 0
17	R/W	0	reg_out_fmt : // unsigned , default = 0
16	R/W	0	reg_only_mat : // unsigned , default = 0 ,only use input matrix ,work when hdr disable
15	R/W	0	mat_o_en, //output matrix enable ,only work when hdr enable
14	R/W	0	mat_in_en //input matrix enable ,only work when hdr enable
13	R/W	0	reg_VDIN0_HDR2_top_en : // unsigned , default = 0, hdr enable signal
12	R/W	1	reg_cgain_mode : // unsigned , default = 1
7: 6	R/W	1	reg_gmut_mode : // unsigned , default = 1
5	R/W	0	reg_in_shift : // unsigned , default = 0
4	R/W	1	reg_in_fmt : // unsigned , default = 1
3	R/W	1	reg_eo_enable : // unsigned , default = 1
2	R/W	1	reg_oe_enable : // unsigned , default = 1

Table 9-2500 VDIN1\_HDR2\_CLK\_GATE 0x1381

Bit(s)	R/W	Default	Description
31:30	R/W	0	clk_tm : gate clock ctrl (main clock) // unsigned , default = 0
29:28	R/W	0	output : matrix clock gate ctrl // unsigned , default = 0

Bit(s)	R/W	Default	Description
25:24	R/W	0	input : matrix clock gate ctrl // unsigned , default = 0
23:22	R/W	0	hdr : top cbus clock gate ctrl // unsigned , default = 0
21:20	R/W	0	eotf : cbus clock gate ctrl // unsigned , default = 0
19:18	R/W	0	oetf : cbus clock gate ctrl // unsigned , default = 0
17:16	R/W	0	gamma : mult cbus clock gate ctrl // unsigned , default = 0
15:14	R/W	0	adaptive : cbus scaler clock gate ctrl // unsigned , default = 0
13:12	R/W	0	cgain : cbus clock gate ctrl // unsigned , default = 0
11:10	R/W	0	eotf : clock gate ctrl // unsigned , default = 0
9:8	R/W	0	oetf : clock gate ctrl // unsigned , default = 0
7:6	R/W	0	gamma : mult clock gate ctrl // unsigned , default = 0
5:4	R/W	0	adaptive : scaler clock gate ctrl // unsigned , default = 0
3:2	R/W	0	uv : gain clock gate ctrl // unsigned , default = 0
1:0	R/W	0	cgain : clock gate ctrl // unsigned , default = 0

Table 9-2501 VDIN1\_HDR2\_MATRIXI\_COEF00\_01 0x1382

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

Table 9-2502 VDIN1\_HDR2\_MATRIXI\_COEF02\_10 0x1383

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

Table 9-2503 VDIN1\_HDR2\_MATRIXI\_COEF11\_12 0x1384

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

Table 9-2504 VDIN1\_HDR2\_MATRIXI\_COEF20\_21 0x1385

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-2505 VDIN1\_HDR2\_MATRIXI\_COEF22 0x1386**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-2506 VDIN1\_HDR2\_MATRIXI\_COEF30\_31 0x1387**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-2507 VDIN1\_HDR2\_MATRIXI\_COEF32\_40 0x1388**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-2508 VDIN1\_HDR2\_MATRIXI\_COEF41\_42 0x1389**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-2509 VDIN1\_HDR2\_MATRIXI\_OFFSET0\_1 0x138A**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-2510 VDIN1\_HDR2\_MATRIXI\_OFFSET2 0x138B**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-2511 VDIN1\_HDR2\_MATRIXI\_PRE\_OFFSET0\_1 0x138C**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-2512 VDIN1\_HDR2\_MATRIXI\_PRE\_OFFSET2 0x138D**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-2513 VDIN1\_HDR2\_MATRIXO\_COEF00\_01 0x138E**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

**Table 9-2514 VDIN1\_HDR2\_MATRIXO\_COEF02\_10 0x138F**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

**Table 9-2515 VDIN1\_HDR2\_MATRIXO\_COEF11\_12 0x1390**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-2516 VDIN1\_HDR2\_MATRIXO\_COEF20\_21 0x1391**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-2517 VDIN1\_HDR2\_MATRIXO\_COEF22 0x1392**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-2518 VDIN1\_HDR2\_MATRIXO\_COEF30\_31 0x1393**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-2519 VDIN1\_HDR2\_MATRIXO\_COEF32\_40 0x1394**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-2520 VDIN1\_HDR2\_MATRIXO\_COEF41\_42 0x1395**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-2521 VDIN1\_HDR2\_MATRIXO\_OFFSET0\_1 0x1396**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-2522 VDIN1\_HDR2\_MATRIXO\_OFFSET2 0x1397**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-2523 VDIN1\_HDR2\_MATRIXO\_PRE\_OFFSET0\_1 0x1398**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-2524 VDIN1\_HDR2\_MATRIXO\_PRE\_OFFSET2 0x1399**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-2525 VDIN1\_HDR2\_MATRIXI\_CLIP 0x139A**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-2526 VDIN1\_HDR2\_MATRIXO\_CLIP 0x139B**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-2527 VDIN1\_HDR2\_CGAIN\_OFFT 0x139C**

Bit(s)	R/W	Default	Description
26:16	R/W	0	reg_cgain_offt2 : // signed , default = 0
10:0	R/W	0	reg_cgain_offt1 : // signed , default = 0

**Table 9-2528 VDIN1\_EOTF\_LUT\_ADDR\_PORT 0x139E**

Bit(s)	R/W	Default	Description
7:0	R/W	0	eotf_lut_addr : // unsigned , default = 0

**Table 9-2529 VDIN1\_EOTF\_LUT\_DATA\_PORT 0x139F**

Bit(s)	R/W	Default	Description
19:0	R/W	0	eotf_lut_data : // unsigned , default = 0

**Table 9-2530 VDIN1\_OETF\_LUT\_ADDR\_PORT 0x13A0**

Bit(s)	R/W	Default	Description
7:0	R/W	0	oetf_lut_addr : // unsigned , default = 0

**Table 9-2531 VDIN1\_OETF\_LUT\_DATA\_PORT 0x13A1**

Bit(s)	R/W	Default	Description
11:0	R/W	0	oetf_lut_data : // unsigned , default = 0

**Table 9-2532 VDIN1\_CGAIN\_LUT\_ADDR\_PORT 0x13A2**

Bit(s)	R/W	Default	Description
7:0	R/W	0	cgain_lut_addr : // unsigned , default = 0

**Table 9-2533 VDIN1\_CGAIN\_LUT\_DATA\_PORT 0x13A3**

Bit(s)	R/W	Default	Description
11:0	R/W	0	cgain_lut_data : // unsigned , default = 0



**Table 9-2534 VDIN1\_HDR2\_CGAIN\_COEF0 0x13A4**

Bit(s)	R/W	Default	Description
27:16	R/W	0	reg_cgain_coef1 : // unsigned , default = 0
11:0	R/W	0	reg_cgain_coef0 : // unsigned , default = 0

**Table 9-2535 VDIN1\_HDR2\_CGAIN\_COEF1 0x13A5**

Bit(s)	R/W	Default	Description
11:0	R/W	0	reg_cgain_coef2 : // unsigned , default = 0

**Table 9-2536 VDIN1\_OGAIN\_LUT\_ADDR\_PORT 0x13A6**

Bit(s)	R/W	Default	Description
7:0	R/W	0	ogain_lut_addr : // unsigned , default = 0

**Table 9-2537 VDIN1\_OGAIN\_LUT\_DATA\_PORT 0x13A7**

Bit(s)	R/W	Default	Description
11:0	R/W	0	ogain_lut_data : // unsigned , default = 0

**Table 9-2538 VDIN1\_HDR2\_ADPS\_CTRL 0x13A8**

Bit(s)	R/W	Default	Description
6	R/W	1	reg_adpscl_bypass2 : // unsigned , default = 1
5	R/W	1	reg_adpscl_bypass1 : // unsigned , default = 1
4	R/W	1	reg_adpscl_bypass0 : // unsigned , default = 1
1:0	R/W	1	reg_adpscl_mode : // unsigned , default = 1

**Table 9-2539 VDIN1\_HDR2\_ADPS\_ALPHA0 0x13A9**

Bit(s)	R/W	Default	Description
29:16	R/W	0x1000	reg_adpscl_alpha1 : // unsigned , default = 0x1000
13:0	R/W	0x1000	reg_adpscl_alpha0 : // unsigned , default = 0x1000

**Table 9-2540 VDIN1\_HDR2\_ADPS\_ALPHA1 0x13AA**

Bit(s)	R/W	Default	Description
27:24	R/W	0xc	reg_adpscl_shift0 : // unsigned , default = 0xc
23:20	R/W	0xc	reg_adpscl_shift1 : // unsigned , default = 0xc
19:16	R/W	0xc	reg_adpscl_shift2 : // unsigned , default = 0xc
13:0	R/W	0x1000	reg_adpscl_alpha2 : // unsigned , default = 0x1000

**Table 9-2541 VDIN1\_HDR2\_ADPS\_BETA0 0x13AB**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta0_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta0 : // unsigned , default = 0xfc000

**Table 9-2542 VDIN1\_HDR2\_ADPS\_BETA1 0x13AC**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta1_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta1 : // unsigned , default = 0xfc000

**Table 9-2543 VDIN1\_HDR2\_ADPS\_BETA2 0x13AD**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta2_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta2 : // unsigned , default = 0xfc000

**Table 9-2544 VDIN1\_HDR2\_ADPS\_COEF0 0x13AE**

Bit(s)	R/W	Default	Description
27:16	R/W	460	reg_adpscl_ys_coef1 : // unsigned , default = 460
11:0	R/W	1188	reg_adpscl_ys_coef0 : // unsigned , default = 1188

**Table 9-2545 VDIN1\_HDR2\_ADPS\_COEF1 0x13AF**

Bit(s)	R/W	Default	Description
11:0	R/W	104	reg_adpscl_ys_coef2 : // unsigned , default = 104

**Table 9-2546 VDIN1\_HDR2\_GMUT\_CTRL 0x13B0**

Bit(s)	R/W	Default	Description
3:0	R/W	14	reg_gmut_shift : // unsigned , default = 14

**Table 9-2547 VDIN1\_HDR2\_GMUT\_COEF0 0x13B1**

Bit(s)	R/W	Default	Description
31:16	R/W	674	reg_gmut_coef01 : // unsigned , default = 674
15:0	R/W	1285	reg_gmut_coef00 : // unsigned , default = 1285

**Table 9-2548 VDIN1\_HDR2\_GMUT\_COEF1 0x13B2**

Bit(s)	R/W	Default	Description
31:16	R/W	142	reg_gmut_coef10 : // unsigned , default = 142
15:0	R/W	89	reg_gmut_coef02 : // unsigned , default = 89

**Table 9-2549 VDIN1\_HDR2\_GMUT\_COEF2 0x13B3**

Bit(s)	R/W	Default	Description
31:16	R/W	23	reg_gmut_coef12 : // unsigned , default = 23
15:0	R/W	1883	reg_gmut_coef11 : // unsigned , default = 1883

**Table 9-2550 VDIN1\_HDR2\_GMUT\_COEF3 0x13B4**

Bit(s)	R/W	Default	Description
31:16	R/W	180	reg_gmut_coef21 : // unsigned , default = 180
15:0	R/W	34	reg_gmut_coef20 : // unsigned , default = 34

**Table 9-2551 VDIN1\_HDR2\_GMUT\_COEF4 0x13B5**

Bit(s)	R/W	Default	Description
15:0	R/W	1834	reg_gmut_coef22 : // unsigned , default = 1834

**Table 9-2552 VDIN1\_HDR2\_PIPE\_CTRL1 0x13B6**

Bit(s)	R/W	Default	Description
31:24	R/W	4	vblank_num_oetf : // unsigned , default = 4
23:16	R/W	4	hblank_num_oetf : // unsigned , default = 4
15:8	R/W	10	vblank_num_eotf : // unsigned , default = 10
7:0	R/W	10	hblank_num_eotf : // unsigned , default = 10

**Table 9-2553 VDIN1\_HDR2\_PIPE\_CTRL2 0x13B7**

Bit(s)	R/W	Default	Description
31:24	R/W	10	vblank_num_cgain : // unsigned , default = 10
23:16	R/W	10	hblank_num_cgain : // unsigned , default = 10
15:8	R/W	11	vblank_num_gmut : // unsigned , default = 11
7:0	R/W	11	hblank_num_gmut : // unsigned , default = 11

**Table 9-2554 VDIN1\_HDR2\_PIPE\_CTRL3 0x13B8**

Bit(s)	R/W	Default	Description
31:24	R/W	22	vblank_num_adps : // unsigned , default = 22
23:16	R/W	2	hblank_num_adps : // unsigned , default = 2
15:8	R/W	4	vblank_num_uv : // unsigned , default = 4
7:0	R/W	4	hblank_num_uv : // unsigned , default = 4

**Table 9-2555 VDIN1\_HDR2\_PROC\_WIN1 0x13B9**

Bit(s)	R/W	Default	Description
28:16	R/W	0	proc_win_h_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_h_st : // unsigned , default = 0

**Table 9-2556 VDIN1\_HDR2\_PROC\_WIN2 0x13BA**

Bit(s)	R/W	Default	Description
31	R/W	0	proc_win_gmut_en : // unsigned , default = 0
30	R/W	0	proc_win_adps_en : // unsigned , default = 0
29	R/W	0	proc_win_cgain_en : // unsigned , default = 0
28:16	R/W	0	proc_win_v_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_v_st : // unsigned , default = 0

**Table 9-2557 VDIN1\_HDR2\_MATRIXI\_EN\_CTRL 0x13BB**

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

**Table 9-2558 VDIN1\_HDR2\_MATRIXO\_EN\_CTRL 0x13BC**

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

**Table 9-2559 VD1\_HDR2\_CTRL 0x3800**

Bit(s)	R/W	Default	Description
20:18	R/W	0	reg_din_swap : // unsigned , default = 0
17	R/W	0	reg_out_fmt : // unsigned , default = 0

Bit(s)	R/W	Default	Description
16	R/W	0	reg_only_mat : // unsigned , default = 0 ,only use input matrix ,work when hdr disable
15	R/W	0	mat_o_en, //output matrix enable ,only work when hdr enable
14	R/W	0	mat_in_en //input matrix enable ,only work when hdr enable
13	R/W	0	reg_VDIN0_HDR2_top_en : // unsigned , default = 0, hdr enable signal
12	R/W	1	reg_cgain_mode : // unsigned , default = 1
7: 6	R/W	1	reg_gmut_mode : // unsigned , default = 1
5	R/W	0	reg_in_shift : // unsigned , default = 0
4	R/W	1	reg_in_fmt : // unsigned , default = 1
3	R/W	1	reg_eo_enable : // unsigned , default = 1
2	R/W	1	reg_oe_enable : // unsigned , default = 1

Table 9-2560 VD1\_HDR2\_CLK\_GATE 0x3801

Bit(s)	R/W	Default	Description
31:30	R/W	0	clk_tm : gate clock ctrl (main clock) // unsigned , default = 0
29:28	R/W	0	output : matrix clock gate ctrl // unsigned , default = 0
25:24	R/W	0	input : matrix clock gate ctrl // unsigned , default = 0
23:22	R/W	0	hdr : top cbus clock gate ctrl // unsigned , default = 0
21:20	R/W	0	eotf : cbus clock gate ctrl // unsigned , default = 0
19:18	R/W	0	oetf : cbus clock gate ctrl // unsigned , default = 0
17:16	R/W	0	gamma : mult cbus clock gate ctrl // unsigned , default = 0
15:14	R/W	0	adaptive : cbus scaler clock gate ctrl // unsigned , default = 0
13:12	R/W	0	cgain : cbus clock gate ctrl // unsigned , default = 0
11:10	R/W	0	eotf : clock gate ctrl // unsigned , default = 0
9:8	R/W	0	oetf : clock gate ctrl // unsigned , default = 0
7:6	R/W	0	gamma : mult clock gate ctrl // unsigned , default = 0
5:4	R/W	0	adaptive : scaler clock gate ctrl // unsigned , default = 0
3:2	R/W	0	uv : gain clock gate ctrl // unsigned , default = 0
1:0	R/W	0	cgain : clock gate ctrl // unsigned , default = 0

Table 9-2561 VD1\_HDR2\_MATRIXI\_COEF00\_01 0x3802

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

**Table 9-2562 VD1\_HDR2\_MATRIXI\_COEF02\_10 0x3803**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

**Table 9-2563 VD1\_HDR2\_MATRIXI\_COEF11\_12 0x3804**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-2564 VD1\_HDR2\_MATRIXI\_COEF20\_21 0x3805**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-2565 VD1\_HDR2\_MATRIXI\_COEF22 0x3806**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-2566 VD1\_HDR2\_MATRIXI\_COEF30\_31 0x3807**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-2567 VD1\_HDR2\_MATRIXI\_COEF32\_40 0x3808**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-2568 VD1\_HDR2\_MATRIXI\_COEF41\_42 0x3809**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-2569 VD1\_HDR2\_MATRIXI\_OFFSET0\_1 0x380A**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-2570 VD1\_HDR2\_MATRIXI\_OFFSET2 0x380B**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-2571 VD1\_HDR2\_MATRIXI\_PRE\_OFFSET0\_1 0x380C**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-2572 VD1\_HDR2\_MATRIXI\_PRE\_OFFSET2 0x380D**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-2573 VD1\_HDR2\_MATRIXO\_COEF00\_01 0x380E**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

**Table 9-2574 VD1\_HDR2\_MATRIXO\_COEF02\_10 0x380F**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

**Table 9-2575 VD1\_HDR2\_MATRIXO\_COEF11\_12 0x3810**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-2576 VD1\_HDR2\_MATRIXO\_COEF20\_21 0x3811**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-2577 VD1\_HDR2\_MATRIXO\_COEF22 0x3812**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-2578 VD1\_HDR2\_MATRIXO\_COEF30\_31 0x3813**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-2579 VD1\_HDR2\_MATRIXO\_COEF32\_40 0x3814**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-2580 VD1\_HDR2\_MATRIXO\_COEF41\_42 0x3815**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-2581 VD1\_HDR2\_MATRIXO\_OFFSET0\_1 0x3816**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-2582 VD1\_HDR2\_MATRIXO\_OFFSET2 0x3817**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0



**Table 9-2583 VD1\_HDR2\_MATRIXO\_PRE\_OFFSET0\_1 0x3818**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-2584 VD1\_HDR2\_MATRIXO\_PRE\_OFFSET2 0x3819**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-2585 VD1\_HDR2\_MATRIXI\_CLIP 0x381A**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-2586 VD1\_HDR2\_MATRIXO\_CLIP 0x381B**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-2587 VD1\_HDR2\_CGAIN\_OFFT 0x381C**

Bit(s)	R/W	Default	Description
26:16	R/W	0	reg_cgain_offt2 : // signed , default = 0
10:0	R/W	0	reg_cgain_offt1 : // signed , default = 0

**Table 9-2588 VD1\_EOTF\_LUT\_ADDR\_PORT 0x381E**

Bit(s)	R/W	Default	Description
7:0	R/W	0	eotf_lut_addr : // unsigned , default = 0

**Table 9-2589 VD1\_EOTF\_LUT\_DATA\_PORT 0x381F**

Bit(s)	R/W	Default	Description
19:0	R/W	0	eotf_lut_data : // unsigned , default = 0

**Table 9-2590 VD1\_OETF\_LUT\_ADDR\_PORT 0x3820**

Bit(s)	R/W	Default	Description
7:0	R/W	0	oetf_lut_addr : // unsigned , default = 0

**Table 9-2591 VD1\_OETF\_LUT\_DATA\_PORT 0x3821**

Bit(s)	R/W	Default	Description
11:0	R/W	0	oetf_lut_data : // unsigned , default = 0

**Table 9-2592 VD1\_CGAIN\_LUT\_ADDR\_PORT 0x3822**

Bit(s)	R/W	Default	Description
7:0	R/W	0	cgain_lut_addr : // unsigned , default = 0

**Table 9-2593 VD1\_CGAIN\_LUT\_DATA\_PORT 0x3823**

Bit(s)	R/W	Default	Description
11:0	R/W	0	cgain_lut_data : // unsigned , default = 0

**Table 9-2594 VD1\_HDR2\_CGAIN\_COEF0 0x3824**

Bit(s)	R/W	Default	Description
27:16	R/W	0	reg_cgain_coef1 : // unsigned , default = 0
11:0	R/W	0	reg_cgain_coef0 : //unsigned , default = 0

**Table 9-2595 VD1\_HDR2\_CGAIN\_COEF1 0x3825**

Bit(s)	R/W	Default	Description
11:0	R/W	0	reg_cgain_coef2 : // unsigned , default = 0

**Table 9-2596 VD1\_OGAIN\_LUT\_ADDR\_PORT 0x3826**

Bit(s)	R/W	Default	Description
7:0	R/W	0	ogain_lut_addr : // unsigned , default = 0

**Table 9-2597 VD1\_OGAIN\_LUT\_DATA\_PORT 0x3827**

Bit(s)	R/W	Default	Description
11:0	R/W	0	ogain_lut_data : // unsigned , default = 0

**Table 9-2598 VD1\_HDR2\_ADPS\_CTRL 0x3828**

Bit(s)	R/W	Default	Description
6	R/W	1	reg_adpscl_bypass2 : // unsigned , default = 1
5	R/W	1	reg_adpscl_bypass1 : // unsigned , default = 1
4	R/W	1	reg_adpscl_bypass0 : // unsigned , default = 1
1:0	R/W	1	reg_adpscl_mode : // unsigned , default = 1

**Table 9-2599 VD1\_HDR2\_ADPS\_ALPHA0 0x3829**

Bit(s)	R/W	Default	Description
29:16	R/W	0x1000	reg_adpscl_alpha1 : // unsigned , default = 0x1000
13:0	R/W	0x1000	reg_adpscl_alpha0 : // unsigned , default = 0x1000

**Table 9-2600 VD1\_HDR2\_ADPS\_ALPHA1 0x382A**

Bit(s)	R/W	Default	Description
27:24	R/W	0xc	reg_adpscl_shift0 : // unsigned , default = 0xc
23:20	R/W	0xc	reg_adpscl_shift1 : // unsigned , default = 0xc
19:16	R/W	0xc	reg_adpscl_shift2 : // unsigned , default = 0xc
13:0	R/W	0x1000	reg_adpscl_alpha2 : // unsigned , default = 0x1000

**Table 9-2601 VD1\_HDR2\_ADPS\_BETA0 0x382B**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta0_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta0 : // unsigned , default = 0xfc000

**Table 9-2602 VD1\_HDR2\_ADPS\_BETA1 0x382C**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta1_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta1 : // unsigned , default = 0xfc000

**Table 9-2603 VD1\_HDR2\_ADPS\_BETA2 0x382D**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta2_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta2 : // unsigned , default = 0xfc000

**Table 9-2604 VD1\_HDR2\_ADPS\_COEF0 0x382E**

Bit(s)	R/W	Default	Description
27:16	R/W	460	reg_adpscl_ys_coef1 : // unsigned , default = 460
11:0	R/W	1188	reg_adpscl_ys_coef0 : // unsigned , default = 1188

**Table 9-2605 VD1\_HDR2\_ADPS\_COEF1 0x382F**

Bit(s)	R/W	Default	Description
11:0	R/W	104	reg_adpscl_ys_coef2 : // unsigned , default = 104

**Table 9-2606 VD1\_HDR2\_GMUT\_CTRL 0x3830**

Bit(s)	R/W	Default	Description
3:0	R/W	14	reg_gmut_shift : // unsigned , default = 14

**Table 9-2607 VD1\_HDR2\_GMUT\_COEF0 0x3831**

Bit(s)	R/W	Default	Description
31:16	R/W	674	reg_gmut_coef01 : // unsigned , default = 674
15:0	R/W	1285	reg_gmut_coef00 : // unsigned , default = 1285

**Table 9-2608 VD1\_HDR2\_GMUT\_COEF1 0x3832**

Bit(s)	R/W	Default	Description
31:16	R/W	142	reg_gmut_coef10 : // unsigned , default = 142
15:0	R/W	89	reg_gmut_coef02 : // unsigned , default = 89

**Table 9-2609 VD1\_HDR2\_GMUT\_COEF2 0x3833**

Bit(s)	R/W	Default	Description
31:16	R/W	23	reg_gmut_coef12 : // unsigned , default = 23
15:0	R/W	1883	reg_gmut_coef11 : // unsigned , default = 1883

**Table 9-2610 VD1\_HDR2\_GMUT\_COEF3 0x3834**

Bit(s)	R/W	Default	Description
31:16	R/W	180	reg_gmut_coef21 : // unsigned , default = 180
15:0	R/W	34	reg_gmut_coef20 : // unsigned , default = 34

**Table 9-2611 VD1\_HDR2\_GMUT\_COEF4 0x3835**

Bit(s)	R/W	Default	Description
15:0	R/W	1834	reg_gmut_coef22 : // unsigned , default = 1834

**Table 9-2612 VD1\_HDR2\_PIPE\_CTRL1 0x3836**

Bit(s)	R/W	Default	Description
31:24	R/W	4	vblank_num_oetf : // unsigned , default = 4
23:16	R/W	4	hblank_num_oetf : // unsigned , default = 4
15:8	R/W	10	vblank_num_eotf : // unsigned , default = 10
7:0	R/W	10	hblank_num_eotf : // unsigned , default = 10

**Table 9-2613 VD1\_HDR2\_PIPE\_CTRL2 0x3837**

Bit(s)	R/W	Default	Description
31:24	R/W	10	vblank_num_cgain : // unsigned , default = 10
23:16	R/W	10	hblank_num_cgain : // unsigned , default = 10
15:8	R/W	11	vblank_num_gmut : // unsigned , default = 11
7:0	R/W	11	hblank_num_gmut : // unsigned , default = 11

**Table 9-2614 VD1\_HDR2\_PIPE\_CTRL3 0x3838**

Bit(s)	R/W	Default	Description
31:24	R/W	22	vblank_num_adps : // unsigned , default = 22
23:16	R/W	2	hblank_num_adps : // unsigned , default = 2
15:8	R/W	4	vblank_num_uv : // unsigned , default = 4
7:0	R/W	4	hblank_num_uv : // unsigned , default = 4

**Table 9-2615 VD1\_HDR2\_PROC\_WIN1 0x3839**

Bit(s)	R/W	Default	Description
28:16	R/W	0	proc_win_h_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_h_st : // unsigned , default = 0

**Table 9-2616 VD1\_HDR2\_PROC\_WIN2 0x383A**

Bit(s)	R/W	Default	Description
31	R/W	0	proc_win_gmut_en : // unsigned , default = 0
30	R/W	0	proc_win_adps_en : // unsigned , default = 0
29	R/W	0	proc_win_cgain_en : // unsigned , default = 0

Bit(s)	R/W	Default	Description
28:16	R/W	0	proc_win_v_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_v_st : // unsigned , default = 0

Table 9-2617 VD1\_HDR2\_MATRIXI\_EN\_CTRL 0x383B

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

Table 9-2618 VD1\_HDR2\_MATRIXO\_EN\_CTRL 0x383C

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

Table 9-2619 VD1\_HDR2\_HIST\_CTRL 0x383D

Bit(s)	R/W	Default	Description
23-16	R/W	0	hist result read index
15-8	R/W	0	gate clock ctrl
5	R/W	0	piecewise mode enable
4	R/W	0	hist window enable
3	R/W	0	data shfit enable
2-0	R/W	0	hist input select 0: e_rgb max 1: e_luma 2/3: e_noliar_sat 4/5: o_before_gamma 6/7 : after_gamma

Table 9-2620 VD1\_HDR2\_HIST\_H\_START\_END 0x383E

Bit(s)	R/W	Default	Description
28-16	R/W	0	hist h start
12-0	R/W	0	hist h end

Table 9-2621 VD1\_HDR2\_HIST\_V\_START\_END 0x383F

Bit(s)	R/W	Default	Description
28-16	R/W	0	hist v start
12-0	R/W	0	hist v end

**Table 9-2622 VD1\_HDR2\_HIST\_RD 0x3840**

Bit(s)	R/W	Default	Description
23-0	R	0	hist 128 bin result

**Table 9-2623 VD2\_HDR2\_CTRL 0x3850**

Bit(s)	R/W	Default	Description
20:18	R/W	0	reg_din_swap : // unsigned , default = 0
17	R/W	0	reg_out_fmt : // unsigned , default = 0
16	R/W	0	reg_only_mat : // unsigned , default = 0 ,only use input matrix ,work when hdr disable
15	R/W	0	mat_o_en, //output matrix enable ,only work when hdr enable
14	R/W	0	mat_in_en //input matrix enable ,only work when hdr enable
13	R/W	0	reg_VDIN0_HDR2_top_en : // unsigned , default = 0, hdr enable signal
12	R/W	1	reg_cgain_mode : // unsigned , default = 1
7: 6	R/W	1	reg_gmut_mode : // unsigned , default = 1
5	R/W	0	reg_in_shift : // unsigned , default = 0
4	R/W	1	reg_in_fmt : // unsigned , default = 1
3	R/W	1	reg_eo_enable : // unsigned , default = 1
2	R/W	1	reg_oe_enable : // unsigned , default = 1

**Table 9-2624 VD2\_HDR2\_CLK\_GATE 0x3851**

Bit(s)	R/W	Default	Description
31:30	R/W	0	clk_tm : gate clock ctrl (main clock) // unsigned , default = 0
29:28	R/W	0	output : matrix clock gate ctrl // unsigned , default = 0
25:24	R/W	0	input : matrix clock gate ctrl // unsigned , default = 0
23:22	R/W	0	hdr : top cbus clock gate ctrl // unsigned , default = 0
21:20	R/W	0	eotf : cbus clock gate ctrl // unsigned , default = 0
19:18	R/W	0	oetf : cbus clock gate ctrl // unsigned , default = 0
17:16	R/W	0	gamma : mult cbus clock gate ctrl // unsigned , default = 0
15:14	R/W	0	adaptive : cbus scaler clock gate ctrl // unsigned , default = 0
13:12	R/W	0	cgain : cbus clock gate ctrl // unsigned , default = 0
11:10	R/W	0	eotf : clock gate ctrl // unsigned , default = 0
9:8	R/W	0	oetf : clock gate ctrl // unsigned , default = 0
7:6	R/W	0	gamma : mult clock gate ctrl // unsigned , default = 0
5:4	R/W	0	adaptive : scaler clock gate ctrl // unsigned , default = 0
3:2	R/W	0	uv : gain clock gate ctrl // unsigned , default = 0
1:0	R/W	0	cgain : clock gate ctrl // unsigned , default = 0

**Table 9-2625 VD2\_HDR2\_MATRIXI\_COEF00\_01 0x3852**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

**Table 9-2626 VD2\_HDR2\_MATRIXI\_COEF02\_10 0x3853**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

**Table 9-2627 VD2\_HDR2\_MATRIXI\_COEF11\_12 0x3854**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-2628 VD2\_HDR2\_MATRIXI\_COEF20\_21 0x3855**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-2629 VD2\_HDR2\_MATRIXI\_COEF22 0x3856**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-2630 VD2\_HDR2\_MATRIXI\_COEF30\_31 0x3857**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-2631 VD2\_HDR2\_MATRIXI\_COEF32\_40 0x3858**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0



**Table 9-2632 VD2\_HDR2\_MATRIXI\_COEF41\_42 0x3859**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-2633 VD2\_HDR2\_MATRIXI\_OFFSET0\_1 0x385A**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-2634 VD2\_HDR2\_MATRIXI\_OFFSET2 0x385B**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-2635 VD2\_HDR2\_MATRIXI\_PRE\_OFFSET0\_1 0x385C**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-2636 VD2\_HDR2\_MATRIXI\_PRE\_OFFSET2 0x385D**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-2637 VD2\_HDR2\_MATRIXO\_COEF00\_01 0x385E**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

**Table 9-2638 VD2\_HDR2\_MATRIXO\_COEF02\_10 0x385F**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

**Table 9-2639 VD2\_HDR2\_MATRIXO\_COEF11\_12 0x3860**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-2640 VD2\_HDR2\_MATRIXO\_COEF20\_21 0x3861**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-2641 VD2\_HDR2\_MATRIXO\_COEF22 0x3862**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-2642 VD2\_HDR2\_MATRIXO\_COEF30\_31 0x3863**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-2643 VD2\_HDR2\_MATRIXO\_COEF32\_40 0x3864**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-2644 VD2\_HDR2\_MATRIXO\_COEF41\_42 0x3865**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-2645 VD2\_HDR2\_MATRIXO\_OFFSET0\_1 0x3866**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-2646 VD2\_HDR2\_MATRIXO\_OFFSET2 0x3867**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-2647 VD2\_HDR2\_MATRIXO\_PRE\_OFFSET0\_1 0x3868**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-2648 VD2\_HDR2\_MATRIXO\_PRE\_OFFSET2 0x3869**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-2649 VD2\_HDR2\_MATRIXI\_CLIP 0x386A**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-2650 VD2\_HDR2\_MATRIXO\_CLIP 0x386B**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-2651 VD2\_HDR2\_CGAIN\_OFFT 0x386C**

Bit(s)	R/W	Default	Description
26:16	R/W	0	reg_cgain_offt2 : // signed , default = 0
10:0	R/W	0	reg_cgain_offt1 : // signed , default = 0

**Table 9-2652 VD2\_EOTF\_LUT\_ADDR\_PORT 0x386E**

Bit(s)	R/W	Default	Description
7:0	R/W	0	eotf_lut_addr : // unsigned , default = 0

**Table 9-2653 VD2\_EOTF\_LUT\_DATA\_PORT 0x386F**

Bit(s)	R/W	Default	Description
19:0	R/W	0	eotf_lut_data : // unsigned , default = 0

**Table 9-2654 VD2\_OETF\_LUT\_ADDR\_PORT 0x3870**

Bit(s)	R/W	Default	Description
7:0	R/W	0	oetf_lut_addr : // unsigned , default = 0

**Table 9-2655 VD2\_OETF\_LUT\_DATA\_PORT 0x3871**

Bit(s)	R/W	Default	Description
11:0	R/W	0	oetf_lut_data : // unsigned , default = 0

**Table 9-2656 VD2\_CGAIN\_LUT\_ADDR\_PORT 0x3872**

Bit(s)	R/W	Default	Description
7:0	R/W	0	cgain_lut_addr : // unsigned , default = 0

**Table 9-2657 VD2\_CGAIN\_LUT\_DATA\_PORT 0x3873**

Bit(s)	R/W	Default	Description
11:0	R/W	0	cgain_lut_data : // unsigned , default = 0

**Table 9-2658 VD2\_HDR2\_CGAIN\_COEF0 0x3874**

Bit(s)	R/W	Default	Description
27:16	R/W	0	reg_cgain_coef1 : // unsigned , default = 0
11:0	R/W	0	reg_cgain_coef0 : // unsigned , default = 0

**Table 9-2659 VD2\_HDR2\_CGAIN\_COEF1 0x3875**

Bit(s)	R/W	Default	Description
11:0	R/W	0	reg_cgain_coef2 : // unsigned , default = 0

**Table 9-2660 VD2\_OGAIN\_LUT\_ADDR\_PORT 0x3876**

Bit(s)	R/W	Default	Description
7:0	R/W	0	ogain_lut_addr : // unsigned , default = 0

**Table 9-2661 VD2\_OGAIN\_LUT\_DATA\_PORT 0x3877**

Bit(s)	R/W	Default	Description
11:0	R/W	0	ogain_lut_data : // unsigned , default = 0

**Table 9-2662 VD2\_HDR2\_ADPS\_CTRL 0x3878**

Bit(s)	R/W	Default	Description
6	R/W	1	reg_adpscl_bypass2 : // unsigned , default = 1
5	R/W	1	reg_adpscl_bypass1 : // unsigned , default = 1
4	R/W	1	reg_adpscl_bypass0 : // unsigned , default = 1
1:0	R/W	1	reg_adpscl_mode : // unsigned , default = 1

**Table 9-2663 VD2\_HDR2\_ADPS\_ALPHA0 0x3879**

Bit(s)	R/W	Default	Description
29:16	R/W	0x1000	reg_adpscl_alpha1 : // unsigned , default = 0x1000
13:0	R/W	0x1000	reg_adpscl_alpha0 : // unsigned , default = 0x1000

**Table 9-2664 VD2\_HDR2\_ADPS\_ALPHA1 0x387A**

Bit(s)	R/W	Default	Description
27:24	R/W	0xc	reg_adpscl_shift0 : // unsigned , default = 0xc
23:20	R/W	0xc	reg_adpscl_shift1 : // unsigned , default = 0xc
19:16	R/W	0xc	reg_adpscl_shift2 : // unsigned , default = 0xc
13:0	R/W	0x1000	reg_adpscl_alpha2 : // unsigned , default = 0x1000

**Table 9-2665 VD2\_HDR2\_ADPS\_BETA0 0x387B**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta0_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta0 : // unsigned , default = 0xfc000

**Table 9-2666 VD2\_HDR2\_ADPS\_BETA1 0x387C**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta1_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta1 : // unsigned , default = 0xfc000

**Table 9-2667 VD2\_HDR2\_ADPS\_BETA2 0x387D**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta2_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta2 : // unsigned , default = 0xfc000

**Table 9-2668 VD2\_HDR2\_ADPS\_COEF0 0x387E**

Bit(s)	R/W	Default	Description
27:16	R/W	460	reg_adpscl_ys_coef1 : // unsigned , default = 460
11:0	R/W	1188	reg_adpscl_ys_coef0 : // unsigned , default = 1188

**Table 9-2669 VD2\_HDR2\_ADPS\_COEF1 0x387F**

Bit(s)	R/W	Default	Description
11:0	R/W	104	reg_adpscl_ys_coef2 : // unsigned , default = 104

**Table 9-2670 VD2\_HDR2\_GMUT\_CTRL 0x3880**

Bit(s)	R/W	Default	Description
3:0	R/W	14	reg_gmut_shift : // unsigned , default = 14

**Table 9-2671 VD2\_HDR2\_GMUT\_COEF0 0x3881**

Bit(s)	R/W	Default	Description
31:16	R/W	674	reg_gmut_coef01 : // unsigned , default = 674
15:0	R/W	1285	reg_gmut_coef00 : // unsigned , default = 1285

**Table 9-2672 VD2\_HDR2\_GMUT\_COEF1 0x3882**

Bit(s)	R/W	Default	Description
31:16	R/W	142	reg_gmut_coef10 : // unsigned , default = 142
15:0	R/W	89	reg_gmut_coef02 : // unsigned , default = 89

**Table 9-2673 VD2\_HDR2\_GMUT\_COEF2 0x3883**

Bit(s)	R/W	Default	Description
31:16	R/W	23	reg_gmut_coef12 : // unsigned , default = 23
15:0	R/W	1883	reg_gmut_coef11 : // unsigned , default = 1883

**Table 9-2674 VD2\_HDR2\_GMUT\_COEF3 0x3884**

Bit(s)	R/W	Default	Description
31:16	R/W	180	reg_gmut_coef21 : // unsigned , default = 180
15:0	R/W	34	reg_gmut_coef20 : // unsigned , default = 34

**Table 9-2675 VD2\_HDR2\_GMUT\_COEF4 0x3885**

Bit(s)	R/W	Default	Description
15:0	R/W	1834	reg_gmut_coef22 : // unsigned , default = 1834

**Table 9-2676 VD2\_HDR2\_PIPE\_CTRL1 0x3886**

Bit(s)	R/W	Default	Description
31:24	R/W	4	vblank_num_oetf : // unsigned , default = 4
23:16	R/W	4	hblank_num_oetf : // unsigned , default = 4
15:8	R/W	10	vblank_num_eotf : // unsigned , default = 10
7:0	R/W	10	hblank_num_eotf : // unsigned , default = 10

**Table 9-2677 VD2\_HDR2\_PIPE\_CTRL2 0x3887**

Bit(s)	R/W	Default	Description
31:24	R/W	10	vblank_num_cgain : // unsigned , default = 10
23:16	R/W	10	hblank_num_cgain : // unsigned , default = 10
15:8	R/W	11	vblank_num_gmut : // unsigned , default = 11
7:0	R/W	11	hblank_num_gmut : // unsigned , default = 11

**Table 9-2678 VD2\_HDR2\_PIPE\_CTRL3 0x3888**

Bit(s)	R/W	Default	Description
31:24	R/W	22	vblank_num_adps : // unsigned , default = 22
23:16	R/W	2	hblank_num_adps : // unsigned , default = 2
15:8	R/W	4	vblank_num_uv : // unsigned , default = 4
7:0	R/W	4	hblank_num_uv : // unsigned , default = 4

**Table 9-2679 VD2\_HDR2\_PROC\_WIN1 0x3889**

Bit(s)	R/W	Default	Description
28:16	R/W	0	proc_win_h_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_h_st : // unsigned , default = 0

**Table 9-2680 VD2\_HDR2\_PROC\_WIN2 0x388A**

Bit(s)	R/W	Default	Description
31	R/W	0	proc_win_gmut_en : // unsigned , default = 0
30	R/W	0	proc_win_adps_en : // unsigned , default = 0
29	R/W	0	proc_win_cgain_en : // unsigned , default = 0
28:16	R/W	0	proc_win_v_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_v_st : // unsigned , default = 0

**Table 9-2681 VD2\_HDR2\_MATRIXI\_EN\_CTRL 0x388B**

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

**Table 9-2682 VD2\_HDR2\_MATRIXO\_EN\_CTRL 0x388C**

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

**Table 9-2683 OSD1\_HDR2\_CTRL 0x38A0**

Bit(s)	R/W	Default	Description
20:18	R/W	0	reg_din_swap : // unsigned , default = 0
17	R/W	0	reg_out_fmt : // unsigned , default = 0
16	R/W	0	reg_only_mat : // unsigned , default = 0 ,only use input matrix ,work when hdr disable
15	R/W	0	mat_o_en, //output matrix enable ,only work when hdr enable
14	R/W	0	mat_in_en //input matrix enable ,only work when hdr enable
13	R/W	0	reg_VDIN0_HDR2_top_en : // unsigned , default = 0, hdr enable signal
12	R/W	1	reg_cgain_mode : // unsigned , default = 1
7: 6	R/W	1	reg_gmut_mode : // unsigned , default = 1
5	R/W	0	reg_in_shift : // unsigned , default = 0
4	R/W	1	reg_in_fmt : // unsigned , default = 1
3	R/W	1	reg_eo_enable : // unsigned , default = 1
2	R/W	1	reg_oe_enable : // unsigned , default = 1



**Table 9-2684 OSD1\_HDR2\_CLK\_GATE 0x38A1**

Bit(s)	R/W	Default	Description
31:30	R/W	0	clk_tm : gate clock ctrl (main clock) // unsigned , default = 0
29:28	R/W	0	output : matrix clock gate ctrl // unsigned , default = 0
25:24	R/W	0	input : matrix clock gate ctrl // unsigned , default = 0
23:22	R/W	0	hdr : top cbus clock gate ctrl // unsigned , default = 0
21:20	R/W	0	eotf : cbus clock gate ctrl // unsigned , default = 0
19:18	R/W	0	oetf : cbus clock gate ctrl // unsigned , default = 0
17:16	R/W	0	gamma : mult cbus clock gate ctrl // unsigned , default = 0
15:14	R/W	0	adaptive : cbus scaler clock gate ctrl // unsigned , default = 0
13:12	R/W	0	cgain : cbus clock gate ctrl // unsigned , default = 0
11:10	R/W	0	eotf : clock gate ctrl // unsigned , default = 0
9:8	R/W	0	oetf : clock gate ctrl // unsigned , default = 0
7:6	R/W	0	gamma : mult clock gate ctrl // unsigned , default = 0
5:4	R/W	0	adaptive : scaler clock gate ctrl // unsigned , default = 0
3:2	R/W	0	uv : gain clock gate ctrl // unsigned , default = 0
1:0	R/W	0	cgain : clock gate ctrl // unsigned , default = 0

**Table 9-2685 OSD1\_HDR2\_MATRIXI\_COEF00\_01 0x38A2**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

**Table 9-2686 OSD1\_HDR2\_MATRIXI\_COEF02\_10 0x38A3**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

**Table 9-2687 OSD1\_HDR2\_MATRIXI\_COEF11\_12 0x38A4**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-2688 OSD1\_HDR2\_MATRIXI\_COEF20\_21 0x38A5**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-2689 OSD1\_HDR2\_MATRIXI\_COEF22 0x38A6**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-2690 OSD1\_HDR2\_MATRIXI\_COEF30\_31 0x38A7**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-2691 OSD1\_HDR2\_MATRIXI\_COEF32\_40 0x38A8**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-2692 OSD1\_HDR2\_MATRIXI\_COEF41\_42 0x38A9**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-2693 OSD1\_HDR2\_MATRIXI\_OFFSET0\_1 0x38AA**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-2694 OSD1\_HDR2\_MATRIXI\_OFFSET2 0x38AB**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-2695 OSD1\_HDR2\_MATRIXI\_PRE\_OFFSET0\_1 0x38AC**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-2696 OSD1\_HDR2\_MATRIXI\_PRE\_OFFSET2 0x38AD**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-2697 OSD1\_HDR2\_MATRIXO\_COEF00\_01 0x38AE**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef00 : // signed , default = 0
12:0	R/W	0	coef01 : // signed , default = 0

**Table 9-2698 OSD1\_HDR2\_MATRIXO\_COEF02\_10 0x38AF**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef02 : // signed , default = 0
12:0	R/W	0	coef10 : // signed , default = 0

**Table 9-2699 OSD1\_HDR2\_MATRIXO\_COEF11\_12 0x38B0**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef11 : // signed , default = 0
12:0	R/W	0	coef12 : // signed , default = 0

**Table 9-2700 OSD1\_HDR2\_MATRIXO\_COEF20\_21 0x38B1**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef20 : // signed , default = 0
12:0	R/W	0	coef21 : // signed , default = 0

**Table 9-2701 OSD1\_HDR2\_MATRIXO\_COEF22 0x38B2**

Bit(s)	R/W	Default	Description
12:0	R/W	0	coef22 : // signed , default = 0

**Table 9-2702 OSD1\_HDR2\_MATRIXO\_COEF30\_31 0x38B3**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef13 : // signed , default = 0
12:0	R/W	0	coef14 : // signed , default = 0

**Table 9-2703 OSD1\_HDR2\_MATRIXO\_COEF32\_40 0x38B4**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef23 : // signed , default = 0
12:0	R/W	0	coef24 : // signed , default = 0

**Table 9-2704 OSD1\_HDR2\_MATRIXO\_COEF41\_42 0x38B5**

Bit(s)	R/W	Default	Description
28:16	R/W	0	coef15 : // signed , default = 0
12:0	R/W	0	coef25 : // signed , default = 0

**Table 9-2705 OSD1\_HDR2\_MATRIXO\_OFFSET0\_1 0x38B6**

Bit(s)	R/W	Default	Description
26:16	R/W	0	offset0 : // signed , default = 0
10:0	R/W	0	offset1 : // signed , default = 0

**Table 9-2706 OSD1\_HDR2\_MATRIXO\_OFFSET2 0x38B7**

Bit(s)	R/W	Default	Description
10:0	R/W	0	offset2 : // signed , default = 0

**Table 9-2707 OSD1\_HDR2\_MATRIXO\_PRE\_OFFSET0\_1 0x38B8**

Bit(s)	R/W	Default	Description
26:16	R/W	0	pre_offset0 : // signed , default = 0
10:0	R/W	0	pre_offset1 : // signed , default = 0

**Table 9-2708 OSD1\_HDR2\_MATRIXO\_PRE\_OFFSET2 0x38B9**

Bit(s)	R/W	Default	Description
10:0	R/W	0	pre_offset2 : // signed , default = 0

**Table 9-2709 OSD1\_HDR2\_MATRIXI\_CLIP 0x38BA**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-2710 OSD1\_HDR2\_MATRIXO\_CLIP 0x38BB**

Bit(s)	R/W	Default	Description
19:8	R/W	0	comp_th : // unsigned , default = 0
7:5	R/W	0	conv_rs : // unsigned , default = 0
4:3	R/W	0	clmod : // unsigned , default = 0

**Table 9-2711 OSD1\_HDR2\_CGAIN\_OFFT 0x38BC**

Bit(s)	R/W	Default	Description
26:16	R/W	0	reg_cgain_offt2 : // signed , default = 0
10:0	R/W	0	reg_cgain_offt1 : // signed , default = 0

**Table 9-2712 OSD1\_EOTF\_LUT\_ADDR\_PORT 0x38be**

Bit(s)	R/W	Default	Description
7:0	R/W	0	eotf_lut_addr : // unsigned , default = 0

**Table 9-2713 OSD1\_EOTF\_LUT\_DATA\_PORT 0x38bf**

Bit(s)	R/W	Default	Description
19:0	R/W	0	eotf_lut_data : // unsigned , default = 0

**Table 9-2714 OSD1\_OETF\_LUT\_ADDR\_PORT 0x38c0**

Bit(s)	R/W	Default	Description
7:0	R/W	0	oetf_lut_addr : // unsigned , default = 0

**Table 9-2715 OSD1\_OETF\_LUT\_DATA\_PORT 0x38c1**

Bit(s)	R/W	Default	Description
11:0	R/W	0	oetf_lut_data : // unsigned , default = 0

**Table 9-2716 OSD1\_CGAIN\_LUT\_ADDR\_PORT 0x38c2**

Bit(s)	R/W	Default	Description
7:0	R/W	0	cgain_lut_addr : // unsigned , default = 0

**Table 9-2717 OSD1\_CGAIN\_LUT\_DATA\_PORT 0x38c3**

Bit(s)	R/W	Default	Description
11:0	R/W	0	cgain_lut_data : // unsigned , default = 0

**Table 9-2718 OSD1\_HDR2\_CGAIN\_COEF0 0x38c4**

Bit(s)	R/W	Default	Description
27:16	R/W	0	reg_cgain_coef1 : // unsigned , default = 0
11:0	R/W	0	reg_cgain_coef0 : // unsigned , default = 0

**Table 9-2719 OSD1\_HDR2\_CGAIN\_COEF1 0x38c5**

Bit(s)	R/W	Default	Description
11:0	R/W	0	reg_cgain_coef2 : // unsigned , default = 0

**Table 9-2720 OSD1\_OGAIN\_LUT\_ADDR\_PORT 0x38c6**

Bit(s)	R/W	Default	Description
7:0	R/W	0	ogain_lut_addr : // unsigned , default = 0

**Table 9-2721 OSD1\_OGAIN\_LUT\_DATA\_PORT 0x38c7**

Bit(s)	R/W	Default	Description
11:0	R/W	0	ogain_lut_data : // unsigned , default = 0

**Table 9-2722 OSD1\_HDR2\_ADPS\_CTRL 0x38c8**

Bit(s)	R/W	Default	Description
6	R/W	1	reg_adpscl_bypass2 : // unsigned , default = 1
5	R/W	1	reg_adpscl_bypass1 : // unsigned , default = 1
4	R/W	1	reg_adpscl_bypass0 : // unsigned , default = 1
1:0	R/W	1	reg_adpscl_mode : // unsigned , default = 1

**Table 9-2723 OSD1\_HDR2\_ADPS\_ALPHA0 0x38c9**

Bit(s)	R/W	Default	Description
29:16	R/W	0x1000	reg_adpscl_alpha1 : // unsigned , default = 0x1000
13:0	R/W	0x1000	reg_adpscl_alpha0 : // unsigned , default = 0x1000

**Table 9-2724 OSD1\_HDR2\_ADPS\_ALPHA1 0x38ca**

Bit(s)	R/W	Default	Description
27:24	R/W	0xc	reg_adpscl_shift0 : // unsigned , default = 0xc
23:20	R/W	0xc	reg_adpscl_shift1 : // unsigned , default = 0xc
19:16	R/W	0xc	reg_adpscl_shift2 : // unsigned , default = 0xc
13:0	R/W	0x1000	reg_adpscl_alpha2 : // unsigned , default = 0x1000

**Table 9-2725 OSD1\_HDR2\_ADPS\_BETA0 0x38cb**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta0_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta0 : // unsigned , default = 0xfc000

**Table 9-2726 OSD1\_HDR2\_ADPS\_BETA1 0x38cc**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta1_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta1 : // unsigned , default = 0xfc000

**Table 9-2727 OSD1\_HDR2\_ADPS\_BETA2 0x38cd**

Bit(s)	R/W	Default	Description
20	R/W	0	reg_adpscl_beta2_s : // unsigned , default = 0
19:0	R/W	0xfc000	reg_adpscl_beta2 : // unsigned , default = 0xfc000

**Table 9-2728 OSD1\_HDR2\_ADPS\_COEF0 0x38ce**

Bit(s)	R/W	Default	Description
27:16	R/W	460	reg_adpscl_ys_coef1 : // unsigned , default = 460
11:0	R/W	1188	reg_adpscl_ys_coef0 : // unsigned , default = 1188

**Table 9-2729 OSD1\_HDR2\_ADPS\_COEF1 0x38cf**

Bit(s)	R/W	Default	Description
11:0	R/W	104	reg_adpscl_ys_coef2 : // unsigned , default = 104

**Table 9-2730 OSD1\_HDR2\_GMUT\_CTRL 0x38d0**

Bit(s)	R/W	Default	Description
3:0	R/W	14	reg_gmut_shift : // unsigned , default = 14

**Table 9-2731 OSD1\_HDR2\_GMUT\_COEF0 0x38d1**

Bit(s)	R/W	Default	Description
31:16	R/W	674	reg_gmut_coef01 : // unsigned , default = 674
15:0	R/W	1285	reg_gmut_coef00 : // unsigned , default = 1285

**Table 9-2732 OSD1\_HDR2\_GMUT\_COEF1 0x38d2**

Bit(s)	R/W	Default	Description
31:16	R/W	142	reg_gmut_coef10 : // unsigned , default = 142
15:0	R/W	89	reg_gmut_coef02 : // unsigned , default = 89

**Table 9-2733 OSD1\_HDR2\_GMUT\_COEF2 0x38d3**

Bit(s)	R/W	Default	Description
31:16	R/W	23	reg_gmut_coef12 : // unsigned , default = 23
15:0	R/W	1883	reg_gmut_coef11 : // unsigned , default = 1883

**Table 9-2734 OSD1\_HDR2\_GMUT\_COEF3 0x38d4**

Bit(s)	R/W	Default	Description
31:16	R/W	180	reg_gmut_coef21 : // unsigned , default = 180
15:0	R/W	34	reg_gmut_coef20 : // unsigned , default = 34

**Table 9-2735 OSD1\_HDR2\_GMUT\_COEF4 0x38d5**

Bit(s)	R/W	Default	Description
15:0	R/W	1834	reg_gmut_coef22 : // unsigned , default = 1834

**Table 9-2736 OSD1\_HDR2\_PIPE\_CTRL1 0x38d6**

Bit(s)	R/W	Default	Description
31:24	R/W	4	vblank_num_oetf : // unsigned , default = 4
23:16	R/W	4	hblank_num_oetf : // unsigned , default = 4
15:8	R/W	10	vblank_num_eotf : // unsigned , default = 10
7:0	R/W	10	hblank_num_eotf : // unsigned , default = 10



**Table 9-2737 OSD1\_HDR2\_PIPE\_CTRL2 0x38d7**

Bit(s)	R/W	Default	Description
31:24	R/W	10	vblank_num_cgain : // unsigned , default = 10
23:16	R/W	10	hblank_num_cgain : // unsigned , default = 10
15:8	R/W	11	vblank_num_gmut : // unsigned , default = 11
7:0	R/W	11	hblank_num_gmut : // unsigned , default = 11

**Table 9-2738 OSD1\_HDR2\_PIPE\_CTRL3 0x38d8**

Bit(s)	R/W	Default	Description
31:24	R/W	22	vblank_num_adps : // unsigned , default = 22
23:16	R/W	2	hblank_num_adps : // unsigned , default = 2
15:8	R/W	4	vblank_num_uv : // unsigned , default = 4
7:0	R/W	4	hblank_num_uv : // unsigned , default = 4

**Table 9-2739 OSD1\_HDR2\_PROC\_WIN1 0x38d9**

Bit(s)	R/W	Default	Description
28:16	R/W	0	proc_win_h_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_h_st : // unsigned , default = 0

**Table 9-2740 OSD1\_HDR2\_PROC\_WIN2 0x38da**

Bit(s)	R/W	Default	Description
31	R/W	0	proc_win_gmut_en : // unsigned , default = 0
30	R/W	0	proc_win_adps_en : // unsigned , default = 0
29	R/W	0	proc_win_cgain_en : // unsigned , default = 0
28:16	R/W	0	proc_win_v_ed : // unsigned , default = 0
12:0	R/W	0	proc_win_v_st : // unsigned , default = 0

**Table 9-2741 OSD1\_HDR2\_MATRIXI\_EN\_CTRL 0x38db**

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

**Table 9-2742 OSD1\_HDR2\_MATRIXO\_EN\_CTRL 0x38dc**

Bit(s)	R/W	Default	Description
5:4	R/W	0	matrix_gclk_ctrl : // unsigned , default = 0
1	R/W	0	enable_sync_sel : // unsigned , default = 0
0	R/W	0	matrix_conv_en : // unsigned , default = 0

**Table 9-2743 OSD1\_HDR2\_HIST\_H\_START\_END 0x38dd**

Bit(s)	R/W	Default	Description
28-16	R/W	0	hist h start
12-0	R/W	0	hist h end

**Table 9-2744 OSD1\_HDR2\_HIST\_V\_START\_END 0x38de**

Bit(s)	R/W	Default	Description
28-16	R/W	0	hist v start
12-0	R/W	0	hist v end

**Table 9-2745 OSD1\_HDR2\_HIST\_V\_START\_END 0x38df**

Bit(s)	R/W	Default	Description
28-16	R/W	0	hist v start
12-0	R/W	0	hist v end

**Table 9-2746 OSD1\_HDR2\_HIST\_RD 0x38e0**

Bit(s)	R/W	Default	Description
23-0	R	0	hist 128 bin result

### 9.2.3.42 VDIN Registers

**Table 9-2747 VDIN\_TOP\_DOUBLE\_CTRL 0x410b**

Bit(s)	R/W	Default	Description
26-24	R/W	5	Vdin1_interrupt mask
22-20	R/W	5	Vdin0_interrupt mask

Bit(s)	R/W	Default	Description
19-16	R/W	0x0	Done flag clear
15-0	R/W	0x310	Vdin reorder sel : Bit3:0 afbce sel 0:disable 1:vdin0 normal 2:vdin0 small 3:vdin1 normal 4:vdin1 small Bit7:4 vdin0 wr mif sel 0:disable 1:vdin0 normal 2:vdin0 small 3:vdin1 normal 4:vdin1 small Bit11:8 vdin1 wr mif sel 0:disable 1:vdin0 normal 2:vdin0 small 3:vdin1 normal 4:vdin1 small Bit15:12 vdin2 wr mif sel 0:disable 1:vdin0 normal 2:vdin0 small 3:vdin1 normal 4:vdin1 small

Table 9-2748 VDIN\_TOP\_MISC 0x410d

Bit(s)	R/W	Default	Description
31-26	R/W	0	Sw_reset
18-16	R/W	3	Vdin0_wr_out_ctrl
15	R/W	0	Vdin line int sel: 1:vdin1 line int 0:vdin0 line int
14	R/W	0	Vid2 irq_sel : 1:afbce frm end 0:vid2 write mif done
13-12	R/W	0	Vdin2 fix disable
11-10	R/W	0	Vdin1_fix_disable
9-8	R/W	0	Vdin0_fix_disable
7-4	R/W	0	Vdin arb gclk ctrl
3-2	R/W	0	Afbce gclk ctrl
1	R/W	0	Afbce path sel: 0:vdin 1:di
0	R/W	0	Ldim stts path sel : 0:vdin0 1:vdin1

VDIN0\_SCALE\_COEF\_IDX 0x1200

VDIN0\_SCALE\_COEF 0x1201

Table 9-2749 VDIN0\_COM\_CTRL0 0x1202

Bit(s)	R/W	Default	Description
31	R/W	0	mpeg_to_vdin_sel, 0: mpeg source to NR directly, 1: mpeg source pass through here
30	R/W	0	mpeg_field info which can be written by software
29	R/W	0	force go_field, pulse signal
28	R/W	0	force go_line, pulse signal
27	R/W	0	enable mpeg_go_field input signal
26-20	R/W	0	hold lines
19	R/W	0	delay go_field function enable
18-12	R/W	0	delay go_field line number
11-10	R/W	0	component2 output switch, 00: select component0 in, 01: select component1 in, 10: select component2 in

Bit(s)	R/W	Default	Description
9-8	R/W	0	component1 output switch, 00: select component0 in, 01: select component1 in, 10: select component2 in
7-6	R/W	0	component0 output switch, 00: select component0 in, 01: select component1 in, 10: select component2 in
5	R/W	0	input window selection function enable
4	R/W	0	enable VDIN common data input, otherwise there will be no video data input
3-0	R/W	0	vdin selection, 1: mpeg_in from dram; 2: bt656 input; 3: Reserved (component input); 4: Reserved(tvdecoder input); 5: Reserved(hdmi rx input); 6: reserved(digital video input); 7: Wr_back 0; 8: reserved(MIPI CSI2) ; 9: Wr_back 1; 10: Reserved(second bt656 input); otherwise no input.

**Table 9-2750 VDIN0\_ACTIVE\_MAX\_PIX\_CNT\_STATUS 0x1203**

Bit(s)	R/W	Default	Description
28-16	R	0	active_max_pix_cnt, readonly
12-0	R	0	active_max_pix_cnt_shadow, readonly

**Table 9-2751 VDIN0\_LCNT\_STATUS 0x1204**

Bit(s)	R/W	Default	Description
28-16	R	0	go_line_cnt, readonly
12-0	R	0	active_line_cnt, readonly

**Table 9-2752 VDIN0\_COM\_STATUS0 0x1205**

Bit(s)	R/W	Default	Description
17	R	0	Vid_wr_pending_dds_wrrsp
16	R	0	Curr_pic_sec
15	R	0	Curr_pic_sec_sav
14-3	R	0	lfifo_buf_cnt
2	R	0	vdin_direct_done status
1	R	0	vdin_nr_done status
0	R	0	field

**Table 9-2753 VDIN0\_COM\_STATUS1 0x1206**

Bit(s)	R/W	Default	Description
31	R	0	vdi4 fifo overflow
29-24	R	0	vdi3_asfifo_cnt
23	R	0	vdi3 fifo overflow
21-16	R	0	vdi3_asfifo_cnt

Bit(s)	R/W	Default	Description
15	R	0	vdi2 fifo overflow
13-8	R	0	vdi2_asfifo_cnt
7	R	0	vdi1 fifo overflow
5-0	R	0	vdi1_asfifo_cnt

**Table 9-2754 VDIN0\_LCNT\_SHADOW\_STATUS 0x1207**

Bit(s)	R/W	Default	Description
28-16	R	0	go_line_cnt_shadow, readonly
12-0	R	0	active_line_cnt_shadow, readonly

**Table 9-2755 VDIN0\_ASFIFO\_CTRL0 0x1208**

Bit(s)	R/W	Default	Description
31	R/W	0	hv size select 0: write mif size 1:vdin hsc input size
30-28	R/W	0	v de start line number
27-20	R/W	0	v blank
19-12	R/W	0	h blank
10-9	R/W	0	bist pattern 0:horizontal gray scale 1: vertical gray scale 2/3:random data
8	R/W	0	Vdin bist enable : replace vdi6 (loop back path)
7	R/W	0	Vdi1 DE enable
6	R/W	0	Vdi1 go field enable
5	R/W	0	Vdi1 go line enable
4	R/W	0	Vdi1 if true, negative active input vsync
3	R/W	0	Vdi1 if true, negative active input hsync
2	R/W	0	Vdi1 vsync soft reset fifo enable
1	R/W	0	Vdi1 overflow status clear
0	R/W	0	Vdi1 asfifo soft reset, level signal

**Table 9-2756 VDIN0\_ASFIFO\_CTRL1 0x1209**

Bit(s)	R/W	Default	Description
23	R/W	0	Vdi4 DE enable
22	R/W	0	Vdi4 go field enable
21	R/W	0	Vdi4 go line enable
20	R/W	0	Vdi4 if true, negative active input vsync
19	R/W	0	Vdi4 if true, negative active input hsync
18	R/W	0	Vdi4 vsync soft reset fifo enable
17	R/W	0	Vdi4 overflow status clear

Bit(s)	R/W	Default	Description
16	R/W	0	Vdi4 asfifo soft reset, level signal
7	R/W	0	Vdi3 DE enable
6	R/W	0	Vdi3 go field enable
5	R/W	0	Vdi3 go line enable
4	R/W	0	Vdi3 if true, negative active input vsync
3	R/W	0	Vdi3 if true, negative active input hsync
2	R/W	0	Vdi3 vsync soft reset fifo enable
1	R/W	0	Vdi3 overflow status clear
0	R/W	0	Vdi3 asfifo soft reset, level signal

Table 9-2757 VDIN0\_WIDTHM1I\_WIDTHM1O 0x120a

Bit(s)	R/W	Default	Description
28-16	R/W	0	input width minus 1, after the window function
12-0	R/W	0	output width minus 1

Table 9-2758 VDIN0\_SC\_MISC\_CTRL 0x120b

Bit(s)	R/W	Default	Description
14-8	R/W	0	hsc_ini_pixi_ptr, signed data, only useful when short_lineo_en is true
7	R/W	0	prehsc_en
6	R/W	0	hsc_en
5	R/W	0	hsc_short_lineo_en, short line output enable
4	R/W	0	hsc_nearest_en
3	R/W	0	Hsc_phase0_always_en
2-0	R/W	0	hsc_bank_length

Table 9-2759 VDIN0\_HSC\_PHASE\_STEP 0x120c

Bit(s)	R/W	Default	Description
28-24	R/W	0	integer portion
23-0	R/W	0	fraction portion

Table 9-2760 VDIN0\_HSC\_INI\_CTRL 0x120d

Bit(s)	R/W	Default	Description
30-29	R/W	0	hscale rpt_p0_num
28-24	R/W	0	hscale ini_rcv_num
23-0	R/W	0	hscale ini_phase

**Table 9-2761 VDIN0\_COM\_STATUS2 0x120e**

Bit(s)	R/W	Default	Description
23	R	0	Vdi7 fifo overflow
21-16	R	0	Vdi7_asfifo_cnt
15	R	0	Vdi6 fifo overflow
13-8	R	0	Vdi6_asfifo_cnt
7	R	0	vdi5 fifo overflow
5-0	R	0	vdi5_asfifo_cnt

**Table 9-2762 VDIN0\_ASFIFO\_CTRL2 0x120f**

Bit(s)	R/W	Default	Description
25	R/W	0	if true, decimation counter sync with first valid DE in the field, //otherwise the decimation counter is not sync with external signal
24	R/W	0	decimation de enable
23-20	R/W	0	decimation phase, which counter value use to decimate,
19-16	R/W	0	decimation number, 0: not decimation, 1: decimation 2, 2: decimation 3 ....
7	R/W	0	Vdi5 DE enable
6	R/W	0	Vdi5 go field enable
5	R/W	0	Vdi5 go line enable
4	R/W	0	Vdi5 if true, negative active input vsync
3	R/W	0	Vdi5 if true, negative active input hsync
2	R/W	0	Vdi5 vsync soft reset fifo enable
1	R/W	0	Vdi5 overflow status clear
0	R/W	0	Vdi5 asfifo soft reset, level signal

**Table 9-2763 VDIN0\_MATRIX\_CTRL 0x1210**

Bit(s)	R/W	Default	Description
7	R/W	0	highlight_enable
6	R/W	0	probe_post, if true, probe pixel data after matrix, otherwise probe pixel data before matrix
5-4	R/W	0	probe_sel, 00: select matrix 0, 01: select matrix 1, otherwise select nothing
3-2	R/W	0	matrix coef idx selection, 00: select mat0, 01: select mat1, otherwise select nothing
1	R/W	0	mat1 conversion matrix enable
0	R/W	0	Mat0 conversion matrix enable

**Table 9-2764 VDIN0\_MATRIX\_COEF00\_01 0x1211**

Bit(s)	R/W	Default	Description
28-16	R/W	0	coef00
12-0	R/W	0	coef01

**Table 9-2765 VDIN0\_MATRIX\_COEF02\_10 0x1212**

Bit(s)	R/W	Default	Description
28-16	R/W	0	coef02
12-0	R/W	0	Coef10

**Table 9-2766 VDIN0\_MATRIX\_COEF11\_12 0x1213**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef11
12-0	R/W	0	Coef12

**Table 9-2767 VDIN0\_MATRIX\_COEF20\_21 0x1214**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef20
12-0	R/W	0	coef21

**Table 9-2768 VDIN0\_MATRIX\_COEF22 0x1215**

Bit(s)	R/W	Default	Description
18-16	R/W	0	convrs
7-0	R/W	0	Coef22

**Table 9-2769 VDIN0\_MATRIX\_OFFSET0\_1 0x1216**

Bit(s)	R/W	Default	Description
26-16	R/W	0	offset0
10-0	R/W	0	Offset1

**Table 9-2770 VDIN0\_MATRIX\_OFFSET2 0x1217**

Bit(s)	R/W	Default	Description
10-0	R/W	0	Offset2



**Table 9-2771 VDIN0\_MATRIX\_PRE\_OFFSET0\_1 0x1218**

Bit(s)	R/W	Default	Description
26-16	R/W	0	Pre_offset0
10-0	R/W	0	Pre_Offset1

**Table 9-2772 VDIN0\_MATRIX\_PRE\_OFFSET2 0x1219**

Bit(s)	R/W	Default	Description
10-0	R/W	0	Pre_Offset2

**Table 9-2773 VDIN0\_LFIFO\_CTRL 0x121a**

Bit(s)	R/W	Default	Description
31	R/W	1	Linebuffer soft reset enable
30	R/W	1	Frame reset enable
28	R/W	0	Discard data enable
18	R/W	0	Vdin channel1 output enable : small size , with hshrink vshrink
17	R/W	1	Vdin channel0 output enable : normal size
16	R/W	0	Pps_path_sel : 0: pps before linebuffer 1:pps after linebuffer
13-0	R/W	0x1000	lfifo_buf_size

**Table 9-2774 VDIN0\_COM\_GCLK\_CTRL 0x121b**

Bit(s)	R/W	Default	Description
15-14	R/W	0	Gate clock control for blackbar detector
13-12	R/W	0	Gate clock control for hist
11-10	R/W	0	Gate clock control for line fifo
9-8	R/W	0	Gate clock control for matrix
7-6	R/W	0	Gate clock control for horizontal scaler
5-4	R/W	0	Gate clock control for pre scaler
3-2	R/W	0	Gate clock control for vdin_com_proc
1-0	R/W	0	Gate clock control for the vdin reg

**Table 9-2775 VDIN0\_INTF\_WIDTHM1 0x121c**

Bit(s)	R/W	Default	Description
12-0	R/W	0	VDIN input interface width minus 1, before the window function, after the de decimation

**Table 9-2776 VDIN0\_WR\_CTRL2 0x121f**

Bit(s)	R/W	Default	Description
19	R/W	0	Vdin0 wr bit10 mode
18	R/W	0	Data_ext_en 1: send out data if req was interrupt by soft reset 0 : normal mode
17:16	R/W	1	Words_lim[1:0] : it would not send out request before Words_lim *16 words were ready
15:12	R/W	1	Burst_lim : 00 , 1 word in 1burst , 01 ,2 words in 1burst, 10, 4 words in 1burst , 11 reserved
10:9	R/W	0	Words_lim[3:2]
8	R/W	0	1: discard data before line fifo, 0: normal mode
7-0	R/W	0	Write chroma canvas address, for NV12/21 mode.

**Table 9-2777 VDIN0\_WR\_CTRL 0x1220**

Bit(s)	R/W	Default	Description
31:30	R/W	0	vdin0_wr_mif_hconv_mode. Applicable only to vdin_write_format=0 or 2. 0=Output every even pixel's CbCr; 1=Output every odd pixel's CbCr; 2=Output an average value per even pair of pixels; 3=Output all CbCr. Only applies to vdin_write_format =2.
29	R/W	0	vdin0_wr_mif_no_clk_gate. If true, enable free-run clock.
28	R/W	0	clear write response counter in the vdin write memory interface
27	R/W	1	eol_sel, 1: use eol as the line end indication, 0: use width as line end indication in the vdin write memory interface
26	R/W	0	vcp_nr_en. Only used in VDIN0. NOT used in VDIN1
25	R/W	1	vcp_wr_en Only used in VDIN0. NOT used in VDIN1
24	R/W	1	vcp_in_en Only used in VDIN0. NOT used in VDIN1
23	R/W	1	vdin frame reset enable, if true, it will provide frame reset during go_field(vsync) to the modules after that
22	R/W	1	vdin line fifo soft reset enable, meaning, if true line fifo will reset during go_field (vsync)
21	R/W	0	vdin direct write done status clear bit
20	R/W	0	vdin NR write done status clear bit
19	R/W	0	Vdin0_wr words swap : swap the 2 64bits word in 128 words
18	R/W	0	vdin0_wr_mif_swap_cbcr. Applicable only to vdin_write_format =2. 0=Output CbCr (NV12); 1=Output CrCb (NV21);
17:16	R/W	0	vdin0_wr_mif_vconv_mode. Applicable only to vdin_write_format=2. 0=Output every even line's CbCr; 1=Output every odd line's CbCr; 2=Reserved; 3=Output all CbCr.
13-12	R/W	0	vdin_write_format, 0: 4:2:2 to one canvas; 1: 4:4:4 to one canvas; 2: Y to luma canvas, CbCr to chroma canvas, for NV12/21; 3: yuv422 full pack mode
11	R/W	0	vdin write canvas double buffer enable, means the canvas address will be latched by vsync before using

Bit(s)	R/W	Default	Description
9	R/W	0	vdin write request urgent
8	R/W	0	vdin write request enable
7-0	R/W	0	Write canvas address (For NV12/21 mode, it's LUMA canvas)

**Table 9-2778 VDIN0\_WR\_H\_START\_END 0x1221**

Bit(s)	R/W	Default	Description
29	R/W	0	if true, horizontal reverse
28-16	R/W	0	start
12-0	R/W	0	end

**Table 9-2779 VDIN0\_WR\_V\_START\_END 0x1222**

Bit(s)	R/W	Default	Description
29	R/W	0	if true, vertical reverse
28-16	R/W	0	start
12-0	R/W	0	end

**Table 9-2780 VDIN0\_VSC\_PHASE\_STEP 0x1223**

Bit(s)	R/W	Default	Description
24-16	R/W	0	integer portion
19-0	R/W	0	fraction portion

**Table 9-2781 VDIN0\_VSC\_INI\_CTRL 0x1224**

Bit(s)	R/W	Default	Description
23	R/W	0	vsc_en, vertical scaler enable
21	R/W	0	vsc_phase0_always_en, when scale up, you have to set it to 1
20-16	R/W	0	ini skip_line_num
15-0	R/W	0	vscaler ini_phase

**Table 9-2782 VDIN0\_SCIN\_HEIGHTM1 0x1225**

Bit(s)	R/W	Default	Description
28-16	R/W	0xc7	Vshrink output height m1
12-0	R/W	0x437	scaler input height minus 1

**Table 9-2783 VDIN0\_DUMMY\_DATA 0x1226**

Bit(s)	R/W	Default	Description
23-16	R/W	0	dummy component 0
15-8	R/W	0x80	dummy component 1
7-0	R/W	0x80	dummy component 2

**Table 9-2784 VDIN0\_MATRIX\_PROBE\_COLOR 0x1228**

Bit(s)	R/W	Default	Description
29-20	R/W	0	component 0
19-10	R/W	0	component 1
9-0	R/W	0	component 2

**Table 9-2785 VDIN0\_MATRIX\_HL\_COLOR 0x1229**

Bit(s)	R/W	Default	Description
23-16	R/W	0	component 0
15-8	R/W	0	component 1
7-0	R/W	0	component 2

**Table 9-2786 VDIN0\_MATRIX\_PROBE\_POS 0x122a**

Bit(s)	R/W	Default	Description
28-16	R/W	0	probe x, position
12-0	R/W	0	probe y, position

**Table 9-2787 VDIN0\_HIST\_CTRL 0x1230**

Bit(s)	R/W	Default	Description
31-24	R/W	0	No use
23-16	R/W	0	No use
11	R/W	0	Hist 34bin only mode
10-9	R/W	0	ldim_stts_din_sel, 00: from matrix0 dout, 01: from vsc_dout, 10: from matrix1 dout, 11: from matrix1 din
8	R/W	0	ldim_stts_en
6-5	R/W	0	hist_dnlp_low the real pixels in each bins got by VDIN_DNLPHISTXX should multiple with $2^{(dnlp\_low+3)}$
4-2	R/W	0	hist_din_sel the source used for hist statistics. 2'b00: from MAT0_dout; 2'b01: from vsc_dout; 2'b10: from mat1_dout, 3: mat1_din 4: hdr2 dout
1	R/W	0	hist_win_en 1'b0: hist used for full picture; 1'b1: hist used for pixels within hist window
0	R/W	0	hist_spl_en 1'b0: disable hist readback; 1'b1: enable hist readback

**Table 9-2788 VDIN0\_HIST\_H\_START\_END 0x1231**

Bit(s)	R/W	Default	Description
28-16	R/W	0	hist_hstart horizontal start value to define hist window
12-0	R/W	0	hist_hend horizontal end value to define hist window

**Table 9-2789 VDIN0\_HIST\_V\_START\_END 0x1232**

Bit(s)	R/W	Default	Description
28-16	R/W	0	hist_vstart vertical start value to define hist window
12-0	R/W	0	hist_vend vertical end value to define hist window

**Table 9-2790 VDIN0\_HIST\_MAX\_MIN 0x1233**

Bit(s)	R/W	Default	Description
15-8	R	0	hist_max maximum value
7-0	R	0	hist_min minimum value

**Table 9-2791 VDIN0\_HIST\_SPL\_VAL 0x1234**

Bit(s)	R/W	Default	Description
31-0	R	0	hist_spl_rd , counts for the total luma value

**Table 9-2792 VDIN0\_HIST\_SPL\_PIX\_CNT 0x1235**

Bit(s)	R/W	Default	Description
21-0	R	0	hist_spl_pixel_count, counts for the total calculated pixels

**Table 9-2793 VDIN0\_HIST\_CHROMA\_SUM 0x1236**

Bit(s)	R/W	Default	Description
31-0	R	0	hist_chroma_sum , counts for the total chroma value

**Table 9-2794 VDIN0\_DNLP\_HIST00 0x1237**

0-255 are split to 64 bins evenly, and VDIN\_DNLP\_HISTXX

They are the statistic numbers of pixels that within each bin.

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 2nd bin
15-0	R	0	counts for the 1st bin

**Table 9-2795 VDINO\_DNLP\_HIST01 0x1238**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 4th bin
15-0	R	0	counts for the 3rd bin

**Table 9-2796 VDINO\_DNLP\_HIST02 0x1239**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 6th bin
15-0	R	0	counts for the 5th bin

**Table 9-2797 VDINO\_DNLP\_HIST03 0x123a**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 8th bin
15-0	R	0	counts for the 7th bin

**Table 9-2798 VDINO\_DNLP\_HIST04 0x123b**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 10th bin
15-0	R	0	counts for the 9th bin

**Table 9-2799 VDINO\_DNLP\_HIST05 0x123c**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 12th bin
15-0	R	0	counts for the 11th bin

**Table 9-2800 VDINO\_DNLP\_HIST06 0x123d**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 14th bin
15-0	R	0	counts for the 13th bin

**Table 9-2801 VDINO\_DNLP\_HIST07 0x123e**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 16th bin
15-0	R	0	counts for the 15th bin

**Table 9-2802 VDINO\_DNLP\_HIST08 0x123f**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 18th bin
15-0	R	0	counts for the 17th bin

**Table 9-2803 VDINO\_DNLP\_HIST09 0x1240**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 20th bin
15-0	R	0	counts for the 19th bin

**Table 9-2804 VDINO\_DNLP\_HIST10 0x1241**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 22nd bin
15-0	R	0	counts for the 21st bin

**Table 9-2805 VDINO\_DNLP\_HIST11 0x1242**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 24th bin
15-0	R	0	counts for the 23rd bin

**Table 9-2806 VDINO\_DNLP\_HIST12 0x1243**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 26th bin
15-0	R	0	counts for the 25th bin

**Table 9-2807 VDINO\_DNLP\_HIST13 0x1244**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 28th bin
15-0	R	0	counts for the 27th bin

**Table 9-2808 VDINO\_DNLP\_HIST14 0x1245**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 30th bin
15-0	R	0	counts for the 29th bin

**Table 9-2809 VDINO\_DNLP\_HIST15 0x1246**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 32nd bin
15-0	R	0	counts for the 31st bin

**Table 9-2810 VDINO\_DNLP\_HIST16 0x1247**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 34th bin
15-0	R	0	counts for the 33rd bin

**Table 9-2811 VDINO\_DNLP\_HIST17 0x1248**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 36th bin
15-0	R	0	counts for the 35th bin

**Table 9-2812 VDINO\_DNLP\_HIST18 0x1249**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 38th bin
15-0	R	0	counts for the 37th bin

**Table 9-2813 VDINO\_DNLP\_HIST19 0x124a**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 40th bin
15-0	R	0	counts for the 39th bin

**Table 9-2814 VDINO\_DNLP\_HIST20 0x124b**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 42nd bin
15-0	R	0	counts for the 41st bin

**Table 9-2815 VDINO\_DNLP\_HIST21 0x124c**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 44th bin
15-0	R	0	counts for the 43rd bin



**Table 9-2816 VDIN0\_DNLP\_HIST22 0x124d**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 46th bin
15-0	R	0	counts for the 45th bin

**Table 9-2817 VDIN0\_DNLP\_HIST23 0x124e**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 48th bin
15-0	R	0	counts for the 47th bin

**Table 9-2818 VDIN0\_DNLP\_HIST24 0x124f**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 50th bin
15-0	R	0	counts for the 49th bin

**Table 9-2819 VDIN0\_DNLP\_HIST25 0x1250**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 52nd bin
15-0	R	0	counts for the 51st bin

**Table 9-2820 VDIN0\_DNLP\_HIST26 0x1251**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 54th bin
15-0	R	0	counts for the 53rd bin

**Table 9-2821 VDIN0\_DNLP\_HIST27 0x1252**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 56th bin
15-0	R	0	counts for the 55th bin

**Table 9-2822 VDIN0\_DNLP\_HIST28 0x1253**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 58th bin
15-0	R	0	counts for the 57th bin

**Table 9-2823 VDINO\_DNLP\_HIST29 0x1254**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 60th bin
15-0	R	0	counts for the 59th bin

**Table 9-2824 VDINO\_DNLP\_HIST30 0x1255**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 62nd bin
15-0	R	0	counts for the 61st bin

**Table 9-2825 VDINO\_DNLP\_HIST31 0x1256**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 64th bin
15-0	R	0	counts for the 63rd bin

**Table 9-2826 VDINO\_WR\_URGENT\_CTRL 0x1257**

Bit(s)	R/W	Default	Description
15	R/W	0	Auto urgent enable
14	R/W	0	Urgent write
9	R/W	0	Write done last sel
8	R/W	0	Reg bvalid enable
7-4	R/W	0	Up_threshold
3-0	R/W	0	Down threshold

**Table 9-2827 VDINO\_RO\_WRMIF\_STATUS 0x1258**

Bit(s)	R/W	Default	Description
20-18	R/W	0	C2 sel
17-15	R/W	0	C1 sel
14-12	R/W	0	C0 sel
11-9	R/W	0	Y2 sel
8-6	R/W	0	Y1 sel
5-3	R/W	0	Y0 sel
2	R/W	0	De tunnel enable
1	R	0	Di pre , nr done
0	R	0	Vdin write mif done

**Table 9-2828 VDIN0\_LDIM\_STTS\_HIST\_READ\_REGION 0x1259**

Bit(s)	R/W	Default	Description
29:20	R	0	Max_comp2
19:10	R	0	Max_comp1
9:0	R	0	Max_comp0

**Table 9-2829 VDIN0\_MEAS\_CTRL0 0x125a**

Bit(s)	R/W	Default	Description
18	R/W	0	reset bit, high active
17	R/W	0	if true, widen hs/vs pulse
16	R/W	0	vsync total counter always accumulating enable
15-12	R/W	0	select hs/vs of video input channel to measure, 0: no selection, 1:vdi1, 2:vid2: 3:vid3, 4:vid4, 5:vdi5, 6:vdi6, 7:vdi7, 8:vdi8, 9:vdi9.
11-4	R/W	0	vsync_span, define how many vsync span need to measure
2-0	R/W	0	meas_hs_index, index to select which HS counter/range

**Table 9-2830 VDIN0\_MEAS\_VS\_COUNT\_HI 0x125b**

Bit(s)	R/W	Default	Description
19-16	R	0	meas_ind_total_count_n, every number of sync_span vsyncs, this count add 1
15-0	R	0	high bit portion of vsync total counter

**Table 9-2831 VDIN0\_MEAS\_VS\_COUNT\_LO 0x125c**

Bit(s)	R/W	Default	Description
31-0	R	0	low bit portion of vsync total counter

**Table 9-2832 VDIN0\_MEAS\_HS\_RANGE 0x125d**

//according to the meas\_hs\_index in register VDIN\_MEAS\_CTRL0 //meas\_hs\_index == 0, first hs range //meas\_hs\_index == 1, second hs range //meas\_hs\_index == 2, third hs range //meas\_hs\_index == 3, fourth hs range

Bit(s)	R/W	Default	Description
28-16	R	0	count_start
12-0	R	0	count_end

**Table 9-2833 VDIN0\_MEAS\_HS\_COUNT 0x125e**

//according to the meas\_hs\_index in register VDIN\_MEAS\_CTRL0, //meas\_hs\_index == 0, first range hs counter, //meas\_hs\_index == 1, second range hs //meas\_hs\_index == 2, third range hs //meas\_hs\_index == 3, fourth range hs

Bit(s)	R/W	Default	Description
23-0	R	0	Hs counter

Table 9-2834 VDIN0\_BLKBAR\_CTRL1 0x125f

Bit(s)	R/W	Default	Description
8	R/W	0	white_enable
7-0	R/W	0	blkbar_white_level

Table 9-2835 VDIN0\_BLKBAR\_CTRL0 0x1260

Bit(s)	R/W	Default	Description
31-24	R/W	0	blkbar_black_level threshold to judge a black point
20-8	R/W	0	blkbar_hwidth left and right region width
7-5	R/W	0	blkbar_comp_sel select yin or uin or vin to be the valid input
4	R/W	0	blkbar_sw_statistic_en enable software statistic of each block black points number
3	R/W	0	blkbar_det_en
2-1	R/W	0	blkbar_din_sel, 0:mat0_dout, 1:vsc_dout, 2:mat1_dout, 3:mat1_din
0	R/W	0	Blkbar_det_top_en

Table 9-2836 VDIN0\_BLKBAR\_H\_START\_END 0x1261

Bit(s)	R/W	Default	Description
28-16	R/W	0	blkbar_hstart. Left region start
12-0	R/W	0	blkbar_hend. Right region end

Table 9-2837 VDIN0\_BLKBAR\_V\_START\_END 0x1262

Bit(s)	R/W	Default	Description
28-16	R/W	0	blkbar_vstart.
12-0	R/W	0	blkbar_vend.

Table 9-2838 VDIN0\_BLKBAR\_CNT\_THRESHOLD 0x1263

Bit(s)	R/W	Default	Description
19-0	R/W	0	blkbar_cnt_threshold. threshold to judge whether a block is totally black

**Table 9-2839 VDIN0\_BLKBAR\_ROW\_TH1\_TH2 0x1264**

Bit(s)	R/W	Default	Description
28-16	R/W	0	blkbar_row_th1. //threshold of the top blackbar
12-0	R/W	0	blkbar_row_th2 //threshold of the bottom blackbar

**Table 9-2840 VDIN0\_BLKBAR\_IND\_LEFT\_START\_END 0x1265**

Bit(s)	R/W	Default	Description
28-16	R	0	blkbar_ind_left_start. horizontal start of the left region in the current searching
12-0	R	0	blkbar_ind_left_end. horizontal end of the left region in the current searching

**Table 9-2841 VDIN0\_BLKBAR\_IND\_RIGHT\_START\_END 0x1266**

Bit(s)	R/W	Default	Description
28-16	R	0	blkbar_ind_right_start.horizontal start of the right region in the current searching
12-0	R	0	blkbar_ind_right_end. horizontal end of the right region in the current searching

**Table 9-2842 VDIN0\_BLKBAR\_IND\_LEFT1\_CNT 0x1267**

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_left1_cnt. Black pixel counter. left part of the left region

**Table 9-2843 VDIN0\_BLKBAR\_IND\_LEFT2\_CNT 0x1268**

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_left2_cnt. Black pixel counter. right part of the left region

**Table 9-2844 VDIN0\_BLKBAR\_IND\_RIGHT1\_CNT 0x1269**

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_right1_cnt. Black pixel counter. left part of the right region

**Table 9-2845 VDIN0\_BLKBAR\_IND\_RIGHT2\_CNT 0x126a**

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_right2_cnt. Black pixel counter. right part of the right region

**Table 9-2846 VDIN0\_BLKBAR\_STATUS0 0x126b**

Bit(s)	R/W	Default	Description
29	R	0	blkbar_ind_black_det_done. LEFT/RIGHT Black detection done
28-16	R	0	blkbar_top_pos. Top black bar position
12-0	R	0	blkbar_bot_pos. Bottom black bar position

**Table 9-2847 VDIN0\_BLKBAR\_STATUS1 0x126c**

Bit(s)	R/W	Default	Description
28-16	R	0	blkbar_left_pos. Left black bar position
12-0	R	0	blkbar_right_pos. Right black bar position

**Table 9-2848 VDIN0\_WIN\_H\_START\_END □□□ 0x126d**

Bit(s)	R/W	Default	Description
28-16	R/W	0	input window H start
12-0	R/W	0	input window H end

**Table 9-2849 VDIN0\_WIN\_V\_START\_END □□□ 0x126e**

Bit(s)	R/W	Default	Description
28-16	R/W	0	input window V start
12-0	R/W	0	input window V end

**Table 9-2850 VDIN0\_ASFIFO\_CTRL3 0x126f**

Bit(s)	R/W	Default	Description
31	R/W	0	Vdi9 DE enable
30	R/W	0	Vdi9 go field enable
29	R/W	0	Vdi9 go line enable
28	R/W	0	Vdi9 if true, negative active input vsync
27	R/W	0	Vdi9 if true, negative active input hsync
26	R/W	0	Vdi9 vsync soft reset fifo enable
25	R/W	0	Vdi9 overflow status clear
24	R/W	0	Vdi9 asfifo soft reset, level signal
15	R/W	0	Vdi7 DE enable
14	R/W	0	Vdi7 go field enable
13	R/W	0	Vdi7 go line enable
12	R/W	0	Vdi7 if true, negative active input vsync
11	R/W	0	Vdi7 if true, negative active input hsync

Bit(s)	R/W	Default	Description
10	R/W	0	Vdi7 vsync soft reset fifo enable
9	R/W	0	Vdi7 overflow status clear
8	R/W	0	Vdi7 asfifo soft reset, level signal
7	R/W	0	Vdi6 DE enable
6	R/W	0	Vdi6 go field enable
5	R/W	0	Vdi6 go line enable
4	R/W	0	Vdi6 if true, negative active input vsync
3	R/W	0	Vdi6 if true, negative active input hsync
2	R/W	0	Vdi6 vsync soft reset fifo enable
1	R/W	0	Vdi6 overflow status clear
0	R/W	0	Vdi6 asfifo soft reset, level signal

Table 9-2851 VDIN0\_COM\_GCLK\_CTRL2 0x1270

Bit(s)	R/W	Default	Description
3-2	R/W	0	Vshrk_clk2 ctrl
1-0	R/W	0	Vshrk_clk1 ctrl

Table 9-2852 VDIN0\_VSHRK\_CTRL 0x1271

Bit(s)	R/W	Default	Description
27	R/W	0	Vshrk enable
26:25	R/W	0	Vshrk mode, 0: 1/2 shrink, 1: 1/4 shrink, 2: 1/8 shrink
24	R/W	0	Vshrink lpf mode, 1: 0.5,1.5,1.5,0.5 lpf for 1/4 shrink, 0.5,1.5,1.5...for 1/8 shrink
23:0	R/W	0	Vshrink padding dummy data

Table 9-2853 VDIN0\_HIST32 0x1272

Bit(s)	R/W	Default	Description
31:0	R	0	Hist 32 mode, [31:16] for white pixel number, 15:0 for black pixel number

Table 9-2854 VDIN0\_COM\_STATUS3 0x1273

Bit(s)	R/W	Default	Description
7	R	0	Vdi9 fifo overflow
5:0	R	0	Vdi9 asfifo cnt

**Table 9-2855 VDIN0\_SYNC\_MASK 0x1274**

Bit(s)	R/W	Default	Description
17	R/W	0	Vsync_mask_en
16	R/W	0	Hsync_mask_en
15-8	R/W	64	Vsync_mask_num: When vsync_mask_en=1, vdin_output_vsync will keep zero for vsync_mask_num*1024 vdin clock cycle after vdin_input_vsync pluse. Example: vdin_clock = 666MHz, vsync_mask_num=255, vdin out vsync will at least keep zero for 0.392ms after one input_vsync
7-0	R/W	64	Hsync_mask_num When hsync_mask_en=1, vdin_output_hsync will keep zero for hsync_mask_num*32 vdin clock cycle after vdin_input_vsync pluse

**Table 9-2856 VDIN0\_DOLBY\_DSC\_CTRL0 0x1275**

Bit(s)	R/W	Default	Description
31	R/W	0	Dolby check enable
30	R/W	0	Tunnel swap mode enable
29-24	R/W	0	Soft reset control, 29 : dsc, 28~27: crc check, 26~24, reserved
16	R/W	0	Little endian mode
15-0	R/W	0	Monitor metadata position

**Table 9-2857 VDIN0\_DOLBY\_DSC\_CTRL1 0x1276**

Bit(s)	R/W	Default	Description
31-16	R/W	0	Metadata pixel start position
7-0	R/W	0	Crc check control

**Table 9-2858 VDIN0\_DOLBY\_DSC\_CTRL2 0x1277**

Bit(s)	R/W	Default	Description
31	R/W	0	Metadata read enable
30	R/W	0	Metadata read address auto-increment
29-20	R/W	0	Metadata sum
17-0	R/W	0	Tunnel mode channel selection

**Table 9-2859 VDIN0\_DOLBY\_DSC\_CTRL3 0x1278**

Bit(s)	R/W	Default	Description
15-0	R/W	0	Select metadata position



**Table 9-2860 VDIN0\_DOLBY\_AXI\_CTRL0 0x1279**

Bit(s)	R/W	Default	Description
31	R/W	0	Memory read enable
30	R/W	0	AXI bus read enable
29	R/W	0	AXI write channel urgent
28	R/W	0	Pack mode little endian
27	R/W	0	AXI bus soft reset
26	R/W	0	Frame reset enable
25	R/W	0	Memory read protection enable
24-16	R/W	0	Memory read sum
15-8	R/W	0	AXI request hold line
7-6	R/W	0	AXI burst length
5	R/W	0	Frame buffer start
4	R/W	0	Buffer start
3-2	R/W	0	AXI status monitor control
1-0	R/W	0	awid

**Table 9-2861 VDIN0\_DOLBY\_AXI\_CTRL1 0x127a**

Bit(s)	R/W	Default	Description
31-0	R/W	0	Buffer start address

**Table 9-2862 VDIN0\_DOLBY\_AXI\_CTRL2 0x127b**

Bit(s)	R/W	Default	Description
31-16	R/W	0	Buffer size
15-0	R/W	0	Frame buffer size

**Table 9-2863 VDIN0\_DOLBY\_AXI\_CTRL3 0x127c**

Bit(s)	R/W	Default	Description
7-0	R/W	0	Hold cycle

Table 9-2864 VDIN0\_WRARB\_MODE 0x12c0

Bit(s)	R/W	Default	Description
23:16	R/W	0	wrarb_sel : uns, default = 0 , wrarb_sel [0]==0 slave dc0 connect master port0 wrarb_sel[0]==1 slave dc0 connect master port1 wrarb_sel [1]==0 slave dc1 connect master port0 wrarb_sel[1]==1 slave dc1 connect master port1 wrarb_sel [2]==0 slave dc2 connect master port0 wrarb_sel[2]==1 slave dc2 connect master port1 wrarb_sel [3]==0 slave dc3 connect master port0 wrarb_sel[3]==1 slave dc3 connect master port1 wrarb_sel [4]==0 slave dc4 connect master port0 wrarb_sel [4]==1 slave dc3 connect master port1 wrarb_sel [5]==0 slave dc5 connect master port0 wrarb_sel[5]==1 slave dc3 connect master port1
8	R/W	0	wrarb_arb_mode : uns, default = 0 , wrarb_arb_mode[0] master port0 arb way,
1:0	R/W	0	wrarb_gate_clk_ctrl : uns, default = 0 , wrarb_gate_clk_ctrl[1:0] master port0 clk gate control

Table 9-2865 VDIN0\_WRARB\_REQEN\_SLV 0x12c1

Bit(s)	R/W	Default	Description
7:0	R/W	0	wrarb_dc_req_en : unsigned , default = 0 wrarb_dc_req_en [0]: the slv0 req to mst port0 enable, wrarb_dc_req_en [1]: the slv1 req to mst port0 enable, wrarb_dc_req_en [2]: the slv2 req to mst port0 enable, wrarb_dc_req_en [0]: the slv0 req to mst port1 enable, wrarb_dc_req_en [1]: the slv1 req to mst port1 enable, wrarb_dc_req_en [2]: the slv2 req to mst port1 enable,

Table 9-2866 VDIN0\_WRARB\_WEIGHT0\_SLV 0x12C2

Bit(s)	R/W	Default	Description
29:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 wrdc_weigh_sxn [0*6+:6]: the slv0 req weigh number wrdc_weigh_sxn [1*6+:6]: the slv1 req weigh number wrdc_weigh_sxn [2*6+:6]: the slv2 req weigh number wrdc_weigh_sxn [3*6+:6]: the slv3 req weigh number wrdc_weigh_sxn [4*6+:6]: the slv4 req weigh number

Table 9-2867 VDIN0\_WRARB\_WEIGHT1\_SLV 0x12C3

Bit(s)	R/W	Default	Description
17:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 the slv5 req weigh number [0*6+:6],the slv6 req weighnumber [1*6+:6],the slv7 req weighnumber [2*6+:6],

Table 9-2868 VDIN0\_RDWR\_ARB\_STATUS 0x12c4

Bit(s)	R/W	Default	Description
2	R.O	0	ro_wrarb_arb_busy : unsigned , default = 0
1	R/W	0x0	reserve :
0	R.O	0	ro_rdarb_arb_busy : unsigned , default = 0

Table 9-2869 VDIN0\_ARB\_DBG\_CTRL 0x12c5

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_det_cmd_ctrl : unsigned , default = 0

**Table 9-2870 VDIN0\_ARB\_DBG\_STAT 0x12C6**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_det_dbg_stat : unsigned , default = 0

**Table 9-2871 VDIN0\_CRC\_CHK 0x12c9**

Bit(s)	R/W	Default	Description
0	W	0	vdin crc check start (pulse)

**Table 9-2872 VDIN0\_RO\_CRC 0x12ca**

Bit(s)	R/W	Default	Description
31-0	R	0	vdin crc result (lock by go_field)

**Table 9-2873 VDIN0\_LINE\_INT 0x12cb**

Bit(s)	R/W	Default	Description
31	R	0	line interrupt flag
30	R	0	write done flag
17	R/W	0	clear line interrupt flag
16	R/W	0	clear write done interrupt flag
15	R/W	0	enable line interrupt
12-0	R/W	0	line interrupt number

**Table 9-2874 VDIN0\_DSC\_CTRL 0x12d0**

Bit(s)	R/W	Default	Description
9:4	R/W	36	reg_dithout_switch : ,uns, default = 36;://{2'h2,2'h1,2'h0}
3	R/W	1	reg_detunnel_en : ,uns, default = 1;
2	R/W	0	reg_detunnel_u_start : ,uns, default = 0;
1	R/W	1	reg_vdin_dith_en : ,uns, default = 1;
0	R/W	1	reg_descramble_en : ,uns, default = 1;

**Table 9-2875 VDIN0\_CFMT\_CTRL 0x12d1**

Bit(s)	R/W	Default	Description
8	R/W	0	reg_chfmt_rpt_pix : uns, default = 0 ; // if true, horizontal formatter use repeating to genereate pixel, otherwise use bilinear interpolation
7:4	R/W	0	reg_chfmt_ini_phase : uns, default = 0 ; // horizontal formatter initial phase
3	R/W	0	reg_chfmt_rpt_p0_en : uns, default = 0 ; // horizontal formatter repeat pixel 0 enable

Bit(s)	R/W	Default	Description
2:1	R/W	1	reg_chfmt_yc_ratio : uns, default = 1 ; // horizontal Y/C ratio, 00: 1:1, 01: 2:1, 10: 4:1
0	R/W	1	reg_chfmt_en : uns, default = 1 ; // horizontal formatter enable

**Table 9-2876 VDIN0\_CFMT\_W 0x12d2**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	reg_chfmt_w : uns, default = 1920 ;horizontal formatter width
12:0	R/W	960	reg_cvfmt_w : uns, default = 960 ;vertical formatter width

**Table 9-2877 VDIN0\_SCB\_CTRL0 0x12d3**

Bit(s)	R/W	Default	Description
11	R/W	1	reg_444c422_gofield_en : uns, default = 1;
10	R/W	1	reg_tunnel_en : uns, default = 1;
9:4	R/W	36	reg_tunnel_outswitch : uns, default = 36;://{2'h2,2'h1,2'h0}
3:2	R/W	0	reg_444c422_mode : uns, default = 0; //0:left 1:right 2,3:avg
1	R/W	0	reg_444c422_bypass : uns, default = 0; 1:bypass

**Table 9-2878 VDIN0\_SCB\_CTRL1 0x12d4**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	reg_444c422_hsize : uns, default = 1920 ;horizontal size
12:0	R/W	960	reg_444c422_vsize : uns, default = 960 ;vertical size

**Table 9-2879 VDIN0\_DSC\_HSIZE 0x12d5**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	reg_detunnel_hsize : uns, default = 1920 ;
12:0	R/W	1920	reg_dither_hsize : uns, default = 1920 ;

**Table 9-2880 VDIN0\_DSC\_DETUNNEL\_SEL 0x12d6**

Bit(s)	R/W	Default	Description
17:0	R/W	34658	reg_detunnel_sel : uns, default = 34658; //{3'h1,3'h0,3'h3 ,3'h5,3'h4,3'h2}

**Table 9-2881 VDIN0\_DSC\_TUNNEL\_SEL 0x12d7**

Bit(s)	R/W	Default	Description
17:0	R/W	69868	reg_tunnel_sel : uns, default = 69868; //= {3'h2,3'h1 ,3'h0,3'h3 ,3'h5,3'h4};

**Table 9-2882 VDIN0\_HDR2\_SIZE 0x12d8**

Bit(s)	R/W	Default	Description
28-16	R/W	0x438	Vdin_hdr2_vsize
12-0	R/W	0x780	Vdin_hdr2_hsize

**Table 9-2883 VDIN0\_VSHRK\_SIZE\_M1 0x12d9**

Bit(s)	R/W	Default	Description
28-16	R/W	0x10d	Vshrink input vsize m1
12-0	R/W	0x437	Vshrink input hsize m1

VDIN0\_DITH\_CTRL 0x12e0

Same as VPU\_VENCL\_DITH\_CTRL

VDIN0\_DITH\_LUT\_1 0x12e1

Same as VPU\_VENCL\_DITH\_LUT\_1

VDIN0\_DITH\_LUT\_2 0x12e2

Same as VPU\_VENCL\_DITH\_LUT\_2

VDIN0\_DITH\_LUT\_3 0x12e3

Same as VPU\_VENCL\_DITH\_LUT\_3

VDIN0\_DITH\_LUT\_4 0x12e4

Same as VPU\_VENCL\_DITH\_LUT\_4

VDIN0\_DITH\_LUT\_5 0x12e5

Same as VPU\_VENCL\_DITH\_LUT\_5

VDIN0\_DITH\_LUT\_6 0x12e6

Same as VPU\_VENCL\_DITH\_LUT\_6

VDIN0\_DITH\_LUT\_7 0x12e7

Same as VPU\_VENCL\_DITH\_LUT\_7

VDIN0\_DITH\_LUT\_8 0x12e8

Same as VPU\_VENCL\_DITH\_LUT\_8

VDIN0\_DITH\_LUT\_9 0x12e9

Same as VPU\_VENCL\_DITH\_LUT\_9

VDIN0\_DITH\_LUT\_10 0x12ea

Same as VPU\_VENCL\_DITH\_LUT\_10

VDIN0\_DITH\_LUT\_10 0x12eb

Same as VPU\_VENCL\_DITH\_LUT\_10

VDIN0\_DITH\_LUT\_11 0x12ec

Same as VPU\_VENCL\_DITH\_LUT\_11

**Table 9-2884 VDIN0\_HSK\_CTRL 0x12ef**

Bit(s)	R/W	Default	Description
22:16	R/W	4	reg_hshrink_size, 2: 1/2 4: 1/4 hshrink, max=64: 1/64 hshrink
12:0	R/W	1920	reg_frm_hsize

**Table 9-2885 VDIN0\_HSK\_COEF\_0 0x12f0**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef00

**Table 9-2886 VDIN0\_HSK\_COEF\_1 0x12f1**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef01

**Table 9-2887 VDIN0\_HSK\_COEF\_2 0x12f2**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef02

**Table 9-2888 VDIN0\_HSK\_COEF\_3 0x12f3**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef03

**Table 9-2889 VDIN0\_HSK\_COEF\_4 0x12f4**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef04

**Table 9-2890 VDIN0\_HSK\_COEF\_5 0x12f5**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef05

**Table 9-2891 VDIN0\_HSK\_COEF\_6 0x12f6**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef06

**Table 9-2892 VDIN0\_HSK\_COEF\_7 0x12f7**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef07

**Table 9-2893 VDIN0\_HSK\_COEF\_8 0x12f8**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef08

**Table 9-2894 VDIN0\_HSK\_COEF\_9 0x12f9**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef09

**Table 9-2895 VDIN0\_HSK\_COEF\_10 0x12fa**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef10

**Table 9-2896 VDIN0\_HSK\_COEF\_11 0x12fb**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef11

**Table 9-2897 VDIN0\_HSK\_COEF\_12 0x12fc**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef12

**Table 9-2898 VDIN0\_HSK\_COEF\_13 0x12fd**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef13

**Table 9-2899 VDIN0\_HSK\_COEF\_14 0x12fe**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef14

**Table 9-2900 VDIN0\_HSK\_COEF\_15 0x12ff**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef15

VDIN1\_SCALE\_COEF\_IDX 0x1300

VDIN1\_SCALE\_COEF 0x1301

**Table 9-2901 VDIN1\_COM\_CTRL0 0x1302**

Bit(s)	R/W	Default	Description
31	R/W	0	mpeg_to_vdin_sel, 0: mpeg source to NR directly, 1: mpeg source pass through here
30	R/W	0	mpeg_field info which can be written by software
29	R/W	0	force go_field, pulse signal
28	R/W	0	force go_line, pulse signal
27	R/W	0	enable mpeg_go_field input signal
26-20	R/W	0	hold lines
19	R/W	0	delay go_field function enable
18-12	R/W	0	delay go_field line number
11-10	R/W	0	component2 output switch, 00: select component0 in, 01: select component1 in, 10: select component2 in
9-8	R/W	0	component1 output switch, 00: select component0 in, 01: select component1 in, 10: select component2 in
7-6	R/W	0	component0 output switch, 00: select component0 in, 01: select component1 in, 10: select component2 in
5	R/W	0	input window selection function enable
4	R/W	0	enable VDIN common data input, otherwise there will be no video data input
3-0	R/W	0	vdin selection, 1: mpeg_in from dram; 2: bt656 input; 3: Reserved (component input); 4: Reserved(tvdecoder input); 5: Reserved(hdmi rx input); 6: reserved(digital video input); 7: Wr_back 0; 8: reserved(MIPI CSI2) ;9: Wr_back 1; 10: Reserved(second bt656 input); otherwise no input.

**Table 9-2902 VDIN1\_ACTIVE\_MAX\_PIX\_CNT\_STATUS 0x1303**

Bit(s)	R/W	Default	Description
28-16	R	0	active_max_pix_cnt, readonly
12-0	R	0	active_max_pix_cnt_shadow, readonly

**Table 9-2903 VDIN1\_LCNT\_STATUS 0x1304**

Bit(s)	R/W	Default	Description
28-16	R	0	go_line_cnt, readonly
12-0	R	0	active_line_cnt, readonly



**Table 9-2904 VDIN1\_COM\_STATUS0 0x1305**

Bit(s)	R/W	Default	Description
12-3	R	0	lfifo_buf_cnt
2	R	0	vdin_direct_done status
1	R	0	vdin_nr_done status
0	R	0	field

**Table 9-2905 VDIN1\_COM\_STATUS1 0x1306**

Bit(s)	R/W	Default	Description
31	R	0	vdi4 fifo overflow
29-24	R	0	vdi3_asfifo_cnt
23	R	0	vdi3 fifo overflow
21-16	R	0	vdi3_asfifo_cnt
15	R	0	vdi2 fifo overflow
13-8	R	0	vdi2_asfifo_cnt
7	R	0	vdi1 fifo overflow
5-0	R	0	vdi1_asfifo_cnt

**Table 9-2906 VDIN1\_LCNT\_SHADOW\_STATUS 0x1307**

Bit(s)	R/W	Default	Description
28-16	R	0	go_line_cnt_shadow, readonly
12-0	R	0	active_line_cnt_shadow, readonly

**Table 9-2907 VDIN1\_ASFIFO\_CTRL0 0x1308**

Bit(s)	R/W	Default	Description
31	R/W	0	hv size select 0: write mif size 1:vdin hsc input size
30-28	R/W	0	v de start line number
27-20	R/W	0	v blank
19-12	R/W	0	h blank
10-9	R/W	0	bist pattern 0:horizontal gray scale 1: vertical gray scale 2/3:random data
8	R/W	0	Vdin bist enable : replace vdi6 (loop back path)
7	R/W	0	Vdi1 DE enable
6	R/W	0	Vdi1 go field enable
5	R/W	0	Vdi1 go line enable
4	R/W	0	Vdi1 if true, negative active input vsync
3	R/W	0	Vdi1 if true, negative active input hsync
2	R/W	0	Vdi1 vsync soft reset fifo enable

Bit(s)	R/W	Default	Description
1	R/W	0	Vdi1 overflow status clear
0	R/W	0	Vdi1 asfifo soft reset, level signal

**Table 9-2908 VDIN1\_ASFIFO\_CTRL1 0x1309**

Bit(s)	R/W	Default	Description
23	R/W	0	Vdi4 DE enable
22	R/W	0	Vdi4 go field enable
21	R/W	0	Vdi4 go line enable
20	R/W	0	Vdi4 if true, negative active input vsync
19	R/W	0	Vdi4 if true, negative active input hsync
18	R/W	0	Vdi4 vsync soft reset fifo enable
17	R/W	0	Vdi4 overflow status clear
16	R/W	0	Vdi4 asfifo soft reset, level signal
7	R/W	0	Vdi3 DE enable
6	R/W	0	Vdi3 go field enable
5	R/W	0	Vdi3 go line enable
4	R/W	0	Vdi3 if true, negative active input vsync
3	R/W	0	Vdi3 if true, negative active input hsync
2	R/W	0	Vdi3 vsync soft reset fifo enable
1	R/W	0	Vdi3 overflow status clear
0	R/W	0	Vdi3 asfifo soft reset, level signal

**Table 9-2909 VDIN1\_WIDTHM1I\_WIDTHM1O 0x130a**

Bit(s)	R/W	Default	Description
28-16	R/W	0	input width minus 1, after the window function
12-0	R/W	0	output width minus 1

**Table 9-2910 VDIN1\_SC\_MISC\_CTRL 0x130b**

Bit(s)	R/W	Default	Description
14-8	R/W	0	hsc_ini_pixi_ptr, signed data, only useful when short_lineo_en is true
7	R/W	0	prehsc_en
6	R/W	0	hsc_en
5	R/W	0	hsc_short_lineo_en, short line output enable
4	R/W	0	hsc_nearest_en
3	R/W	0	Hsc_phase0_always_en

Bit(s)	R/W	Default	Description
3	R/W	0	phase0_always_en
2-0	R/W	0	hsc_bank_length

**Table 9-2911 VDIN1\_HSC\_PHASE\_STEP 0x130c**

Bit(s)	R/W	Default	Description
28-24	R/W	0	integer portion
23-0	R/W	0	fraction portion

**Table 9-2912 VDIN1\_HSC\_INI\_CTRL 0x130d**

Bit(s)	R/W	Default	Description
30-29	R/W	0	hscale rpt_p0_num
28-24	R/W	0	hscale ini_rcv_num
23-0	R/W	0	hscale ini_phase

**Table 9-2913 VDIN1\_COM\_STATUS2 0x130e**

Bit(s)	R/W	Default	Description
23	R	0	Vdi7 fifo overflow
21-16	R	0	Vdi7_asfifo_cnt
15	R	0	Vdi6 fifo overflow
13-8	R	0	Vdi6_asfifo_cnt
7	R	0	vdi5 fifo overflow
5-0	R	0	vdi5_asfifo_cnt

**Table 9-2914 VDIN1\_ASFIFO\_CTRL2 0x130f**

Bit(s)	R/W	Default	Description
25	R/W	0	if true, decimation counter sync with first valid DE in the field, //otherwise the decimation counter is not sync with external signal
24	R/W	0	decimation de enable
23-20	R/W	0	decimation phase, which counter value use to decimate,
19-16	R/W	0	decimation number, 0: not decimation, 1: decimation 2, 2: decimation 3 ....
7	R/W	0	Vdi5 DE enable
6	R/W	0	Vdi5 go field enable
5	R/W	0	Vdi5 go line enable
4	R/W	0	Vdi5 if true, negative active input vsync
3	R/W	0	Vdi5 if true, negative active input hsync
2	R/W	0	Vdi5 vsync soft reset fifo enable

Bit(s)	R/W	Default	Description
1	R/W	0	Vdi5 overflow status clear
0	R/W	0	Vdi5 asfif0 soft reset, level signal

**Table 9-2915 VDIN1\_MATRIX\_CTRL 0x1310**

Bit(s)	R/W	Default	Description
0	R/W	0	post conversion matrix enable

**Table 9-2916 VDIN1\_MATRIX\_COEF00\_01 0x1311**

Bit(s)	R/W	Default	Description
28-16	R/W	0	coef00
12-0	R/W	0	coef01

**Table 9-2917 VDIN1\_MATRIX\_COEF02\_10 0x1312**

Bit(s)	R/W	Default	Description
28-16	R/W	0	coef02
12-0	R/W	0	Coef10

**Table 9-2918 VDIN1\_MATRIX\_COEF11\_12 0x1313**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef11
12-0	R/W	0	Coef12

**Table 9-2919 VDIN1\_MATRIX\_COEF20\_21 0x1314**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef20
12-0	R/W	0	coef21

**Table 9-2920 VDIN1\_ 0x1315**

Bit(s)	R/W	Default	Description
18-16	R/W	0	convrs
7-0	R/W	0	Coef22

**Table 9-2921 VDIN1\_MATRIX\_OFFSET0\_1 0x1316**

Bit(s)	R/W	Default	Description
26-16	R/W	0	offset0
10-0	R/W	0	Offset1

**Table 9-2922 VDIN1\_MATRIX\_OFFSET2 0x1317**

Bit(s)	R/W	Default	Description
10-0	R/W	0	Offset2

**Table 9-2923 VDIN1\_MATRIX\_PRE\_OFFSET0\_1 0x1318**

Bit(s)	R/W	Default	Description
26-16	R/W	0	Pre_offset0
10-0	R/W	0	Pre_Offset1

**Table 9-2924 VDIN1\_MATRIX\_PRE\_OFFSET2 0x1319**

Bit(s)	R/W	Default	Description
10-0	R/W	0	Pre_Offset2

**Table 9-2925 VDIN1\_LFIFO\_CTRL 0x131a**

Bit(s)	R/W	Default	Description
31	R/W	1	Linebuffer soft reset enable
30	R/W	1	Frame reset enable
28	R/W	0	Discard data enable
18	R/W	0	Vdin channel1 output enable
17	R/W	1	Vdin channel0 output enable
16	R/W	0	Pps_path_sel : 0: pps before linebuffer 1:pps after linebuffer
13-0	R/W	0x1000	lfifo_buf_size

**Table 9-2926 VDIN1\_COM\_GCLK\_CTRL 0x131b**

Bit(s)	R/W	Default	Description
15-14	R/W	0	Gate clock control for blackbar detector
13-12	R/W	0	Gate clock control for hist
11-10	R/W	0	Gate clock control for line fifo
9-8	R/W	0	Gate clock control for matrix
7-6	R/W	0	Gate clock control for horizontal scaler
5-4	R/W	0	Gate clock control for pre scaler

Bit(s)	R/W	Default	Description
3-2	R/W	0	Gate clock control for vdin_com_proc
1-0	R/W	0	Gate clock control for the vdin reg

**Table 9-2927 VDIN1\_INTF\_WIDTHM1 0x131c**

Bit(s)	R/W	Default	Description
26	R/W	0	VDIN write mif bvalid_sel: 1. Bvalid_signal from bus, 0: bytes_wr handshakes
25	R/W	0	VDIN write mif burst last sel: 1. All kind of burst last signal include ext_data_last. 0. Used the normal burst last signal
12-0	R/W	0	VDIN input interface width minus 1, before the window function, after the de decimation

**Table 9-2928 VDIN1\_WR\_CTRL2 0x131f**

Bit(s)	R/W	Default	Description
19	R/W	0	Vdin1 wr bit10 mode
18	R/W	0	Data_ext_en 1: send out data if req was interrupt by soft reset 0 : normal mode
17:16	R/W	1	Words_lim[1:0]: it would not send out request before Words_lim *16 words were ready
15:12	R/W	1	Burst_lim : 00 , 1 word in 1burst , 01 , 2 words in 1burst , 10 , 4 words in 1burst , 11 reserved
10:9	R/W	0	Words_lim[3:2]
8	R/W	0	1: discard data before line fifo, 0: normal mode
7-0	R/W	0	Write chroma canvas address, for NV12/21 mode.

**Table 9-2929 VDIN1\_WR\_CTRL 0x1320**

Bit(s)	R/W	Default	Description
31-30	R/W	0	vdin1_wr_mif_hconv_mode. Applicable only to vdin_write_format=0 or 2. 0=Output every even pixel's CbCr; 1=Output every odd pixel's CbCr; 2=Output an average value per even pair of pixels; 3=Output all CbCr. Only applies to vdin_write_format =2.
29	R/W	0	vdin1_wr_mif_no_clk_gate. If true, enable free-run clock.
28	R/W	0	clear write response counter in the vdin write memory interface
27	R/W	1	eol_sel, 1: use eol as the line end indication, 0: use width as line end indication in the vdin write memory interface
23	R/W	1	vdin frame reset enable, if true, it will provide frame reset during go_field(vsync) to the modules after that
22	R/W	1	vdin line fifo soft reset enable, meaning, if true line fifo will reset during go_field (vsync)
21	R/W	0	vdin direct write done status clear bit
20	R/W	0	vdin NR write done status clear bit
19	R/W	0	Vdin0_wr words swap : swap the 2 64bits word in 128 words

Bit(s)	R/W	Default	Description
18	R/W	0	vdin1_wr_mif_swap_cbcr. Applicable only to vdin_write_format =2. 0=Output CbCr (NV12); 1=Output CrCb (NV21);
17:16	R/W	0	vdin1_wr_mif_vconv_mode. Applicable only to vdin_write_format=2. 0=Output every even line's CbCr; 1=Output every odd line's CbCr; 2=Reserved; 3=Output all CbCr.
13-12	R/W	0	vdin_write_format, 0: 4:2:2 to one canvas; 1: 4:4:4 to one canvas; 2: Y to luma canvas, CbCr to chroma canvas, for NV12/21; 3: 422 10 bit full pack mode
11	R/W	0	vdin write canvas double buffer enable, means the canvas address will be latched by vsync before using
9	R/W	0	vdin write request urgent
8	R/W	0	vdin write request enable
7-0	R/W	0	Write canvas address (For NV12/21 mode, it's LUMA canvas)

Table 9-2930 VDIN1\_WR\_H\_START\_END 0x1321

Bit(s)	R/W	Default	Description
27-16	R/W	0	start
11-0	R/W	0	end

Table 9-2931 VDIN1\_WR\_V\_START\_END 0x1322

Bit(s)	R/W	Default	Description
27-16	R/W	0	start
11-0	R/W	0	end

Table 9-2932 VDIN1\_VSC\_PHASE\_STEP 0x1323

Bit(s)	R/W	Default	Description
24-16	R/W	0	integer portion
19-0	R/W	0	fraction portion

Table 9-2933 VDIN1\_VSC\_INI\_CTRL 0x1324

Bit(s)	R/W	Default	Description
23	R/W	0	vsc_en, vertical scaler enable
21	R/W	0	vsc_phase0_always_en, when scale up, you have to set it to 1
20-16	R/W	0	ini skip_line_num
15-0	R/W	0	vscaler ini_phase

**Table 9-2934 VDIN1\_SCIN\_HEIGHTM1 0x1325**

Bit(s)	R/W	Default	Description
28-16	R/W	0xc7	Vshrink output height m1
12-0	R/W	0x437	scaler input height minus 1

**Table 9-2935 VDIN1\_DUMMY\_DATA 0x1326**

Bit(s)	R/W	Default	Description
23-16	R/W	0	dummy component 0
15-8	R/W	0x80	dummy component 1
7-0	R/W	0x80	dummy component 2

**Table 9-2936 VDIN1\_MATRIX\_PROBE\_COLOR 0x1328**

Bit(s)	R/W	Default	Description
29-20	R/W	0	component 0
19-10	R/W	0	component 1
9-0	R/W	0	component 2

**Table 9-2937 VDIN1\_MATRIX\_HL\_COLOR 0x1329**

Bit(s)	R/W	Default	Description
23-16	R/W	0	component 0
15-8	R/W	0	component 1
7-0	R/W	0	component 2

**Table 9-2938 VDIN1\_MATRIX\_PROBE\_POS 0x132a**

Bit(s)	R/W	Default	Description
28-16	R/W	0	probe x, position
12-0	R/W	0	probe y, position

**Table 9-2939 VDIN1\_HIST\_CTRL 0x1330**

Bit(s)	R/W	Default	Description
31-24	R/W	0	Hist pixel white threshold, larger than this will be counted as white pixel number
23-16	R/W	0	Hist pixel black threshold, less than this will be counted as black pixel number
11	R/W	0	Hist 32bin only mode
10-9	R/W	0	ldim_stts_din_sel, 00: from matrix0 dout, 01: from vsc_dout, 10: from matrix1 dout, 11: form matrix1 din
8	R/W	0	ldim_stts_en



Bit(s)	R/W	Default	Description
6-5	R/W	0	hist_dnlp_low the real pixels in each bins got by VDIN_DNLP_HISTXX should multiple with $2^{(dnlp\_low+3)}$
4-2	R/W	0	hist_din_sel the source used for hist statistics. 2'b00: from MAT0_dout; 2'b01: from vsc_dout; 2'b10: from mat1_dout, 3: mat1_din 4:hdr2 dout
1	R/W	0	hist_win_en 1'b0: hist used for full picture; 1'b1: hist used for pixels within hist window
0	R/W	0	hist_spl_en 1'b0: disable hist readback; 1'b1: enable hist readback

**Table 9-2940 VDIN1\_HIST\_H\_START\_END 0x1331**

Bit(s)	R/W	Default	Description
28-16	R/W	0	hist_hstart horizontal start value to define hist window
12-0	R/W	0	hist_hend horizontal end value to define hist window

**Table 9-2941 VDIN1\_HIST\_V\_START\_END 0x1332**

Bit(s)	R/W	Default	Description
28-16	R/W	0	hist_vstart vertical start value to define hist window
12-0	R/W	0	hist_vend vertical end value to define hist window

**Table 9-2942 VDIN1\_HIST\_MAX\_MIN 0x1333**

Bit(s)	R/W	Default	Description
15-8	R	0	hist_max maximum value
7-0	R	0	hist_min minimum value

**Table 9-2943 VDIN1\_HIST\_SPL\_VAL 0x1334**

Bit(s)	R/W	Default	Description
31-0	R	0	hist_spl_rd , counts for the total luma value

**Table 9-2944 VDIN1\_HIST\_SPL\_PIX\_CNT 0x1335**

Bit(s)	R/W	Default	Description
21-0	R	0	hist_spl_pixel_count, counts for the total calculated pixels

**Table 9-2945 VDIN1\_HIST\_CHROMA\_SUM 0x1336**

Bit(s)	R/W	Default	Description
31-0	R	0	hist_chroma_sum , counts for the total chroma value

**Table 9-2946 VDIN1\_DNLP\_HIST00 0x1337**

//0-255 are split to 64 bins evenly, and VDIN\_DNLP\_HISTXX //are the statistic number of pixels that within each bin.

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 2nd bin
15-0	R	0	counts for the 1st bin

**Table 9-2947 VDIN1\_DNLP\_HIST01 0x1338**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 4th bin
15-0	R	0	counts for the 3rd bin

**Table 9-2948 VDIN1\_DNLP\_HIST02 0x1339**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 6th bin
15-0	R	0	counts for the 5th bin

**Table 9-2949 VDIN1\_DNLP\_HIST03 0x133a**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 8th bin
15-0	R	0	counts for the 7th bin

**Table 9-2950 VDIN1\_DNLP\_HIST04 0x133b**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 10th bin
15-0	R	0	counts for the 9th bin

**Table 9-2951 VDIN1\_DNLP\_HIST05 0x133c**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 12th bin
15-0	R	0	counts for the 11th bin

**Table 9-2952 VDIN1\_DNLP\_HIST06 0x133d**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 14th bin
15-0	R	0	counts for the 13th bin

**Table 9-2953 VDIN1\_DNLP\_HIST07 0x133e**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 16th bin
15-0	R	0	counts for the 15th bin

**Table 9-2954 VDIN1\_DNLP\_HIST08 0x133f**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 18th bin
15-0	R	0	counts for the 17th bin

**Table 9-2955 VDIN1\_DNLP\_HIST09 0x1340**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 20th bin
15-0	R	0	counts for the 19th bin

**Table 9-2956 VDIN1\_DNLP\_HIST10 0x1341**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 22nd bin
15-0	R	0	counts for the 21st bin

**Table 9-2957 VDIN1\_DNLP\_HIST11 0x1342**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 24th bin
15-0	R	0	counts for the 23rd bin

**Table 9-2958 VDIN1\_DNLP\_HIST12 0x1343**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 26th bin
15-0	R	0	counts for the 25th bin

**Table 9-2959 VDIN1\_DNLP\_HIST13 0x1344**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 28th bin
15-0	R	0	counts for the 27th bin

**Table 9-2960 VDIN1\_DNLP\_HIST14 0x1345**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 30th bin
15-0	R	0	counts for the 29th bin

**Table 9-2961 VDIN1\_DNLP\_HIST15 0x1346**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 32nd bin
15-0	R	0	counts for the 31st bin

**Table 9-2962 VDIN1\_DNLP\_HIST16 0x1347**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 34th bin
15-0	R	0	counts for the 33rd bin

**Table 9-2963 VDIN1\_DNLP\_HIST17 0x1348**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 36th bin
15-0	R	0	counts for the 35th bin

**Table 9-2964 VDIN1\_DNLP\_HIST18 0x1349**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 38th bin
15-0	R	0	counts for the 37th bin

**Table 9-2965 VDIN1\_DNLP\_HIST19 0x134a**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 40th bin
15-0	R	0	counts for the 39th bin

**Table 9-2966 VDIN1\_DNLP\_HIST20 0x134b**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 42nd bin
15-0	R	0	counts for the 41st bin

**Table 9-2967 VDIN1\_DNLP\_HIST21 0x134c**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 44th bin
15-0	R	0	counts for the 43rd bin

**Table 9-2968 VDIN1\_DNLP\_HIST22 0x134d**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 46th bin
15-0	R	0	counts for the 45th bin

**Table 9-2969 VDIN1\_DNLP\_HIST23 0x134e**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 48th bin
15-0	R	0	counts for the 47th bin

**Table 9-2970 VDIN1\_DNLP\_HIST24 0x134f**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 50th bin
15-0	R	0	counts for the 49th bin

**Table 9-2971 VDIN1\_DNLP\_HIST25 0x1350**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 52nd bin
15-0	R	0	counts for the 51st bin

**Table 9-2972 VDIN1\_DNLP\_HIST26 0x1351**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 54th bin
15-0	R	0	counts for the 53rd bin

**Table 9-2973 VDIN1\_DNLP\_HIST27 0x1352**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 56th bin
15-0	R	0	counts for the 55th bin

**Table 9-2974 VDIN1\_DNLP\_HIST28 0x1353**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 58th bin
15-0	R	0	counts for the 57th bin

**Table 9-2975 VDIN1\_DNLP\_HIST29 0x1354**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 60th bin
15-0	R	0	counts for the 59th bin

**Table 9-2976 VDIN1\_DNLP\_HIST30 0x1355**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 62nd bin
15-0	R	0	counts for the 61st bin

**Table 9-2977 VDIN1\_DNLP\_HIST31 0x1356**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 64th bin
15-0	R	0	counts for the 63rd bin

**Table 9-2978 VDIN1\_WR\_URGENT\_CTRL 0x1357**

Bit(s)	R/W	Default	Description
15	R/W	0	Auto urgent enable
14	R/W	0	Urgent write
9	R/W	0	Write done last sel
8	R/W	0	Reg bvalid enable
7-4	R/W	0	Up_threshold
3-0	R/W	0	Down threshold

**Table 9-2979 VDIN1\_RO\_WRMIF\_STATUS 0x1358**

Bit(s)	R/W	Default	Description
20-18	R/W	0	C2 sel
17-15	R/W	0	C1 sel
14-12	R/W	0	C0 sel
11-9	R/W	0	Y2 sel
8-6	R/W	0	Y1 sel
5-3	R/W	0	Y0 sel

Bit(s)	R/W	Default	Description
2	R/W	0	De tunnel enable
1	R	0	Di pre , nr done
0	R	0	Vdin write mif done

**Table 9-2980 VDIN1\_LDIM\_STTS\_HIST\_READ\_REGION 0x1359**

Bit(s)	R/W	Default	Description
29:20	R	0	Max_comp2
19:10	R	0	Max_comp1
9:0	R	0	Max_comp0

**Table 9-2981 VDIN1\_MEAS\_CTRL0 0x135a**

Bit(s)	R/W	Default	Description
18	R/W	0	reset bit, high active
17	R/W	0	if true, widen hs/vs pulse
16	R/W	0	vsync total counter always accumulating enable
14-12	R/W	0	select hs/vs of video input channel to measure, 0: no selection, 1 vdi1, 2: vid2: 3: vid3, 4:vid4, 5:vdi5, 6:vdi6, 7:vdi7, 8:vdi8, 9:vdi9.
11-4	R/W	0	vsync_span, define how many vsync span need to measure
2-0	R/W	0	meas_hs_index, index to select which HS counter/range

**Table 9-2982 VDIN1\_MEAS\_VS\_COUNT\_HI 0x135b**

Bit(s)	R/W	Default	Description
19-16	R	0	meas_ind_total_count_n, every number of sync_span vsyncs, this count add 1
15-0	R	0	high bit portion of vsync total counter

**Table 9-2983 VDIN1\_MEAS\_VS\_COUNT\_LO 0x135c**

Bit(s)	R/W	Default	Description
31-0	R	0	low bit portion of vsync total counter

**Table 9-2984 VDIN1\_MEAS\_HS\_RANGE 0x135d**

//according to the meas\_hs\_index in register VDIN\_MEAS\_CTRL0 //meas\_hs\_index == 0, first hs range //meas\_hs\_index == 1, second hs range //meas\_hs\_index == 2, third hs range //meas\_hs\_index == 3, fourth hs range

Bit(s)	R/W	Default	Description
28-16	R	0	count_start
12-0	R	0	count_end

**Table 9-2985 VDIN1\_MEAS\_HS\_COUNT 0x135e**

//according to the meas\_hs\_index in register VDIN\_MEAS\_CTRL0, //meas\_hs\_index == 0, first range hs counter, //meas\_hs\_index == 1, second range hs //meas\_hs\_index == 2, third range hs //meas\_hs\_index == 3, fourth range hs

Bit(s)	R/W	Default	Description
23-0	R	0	Hs counter

**Table 9-2986 VDIN1\_BLKBAR\_CTRL1 0x135f**

Bit(s)	R/W	Default	Description
8	R/W	0	white_enable
7-0	R/W	0	blkbar_white_level

**Table 9-2987 VDIN1\_BLKBAR\_CTRL0 0x1360**

Bit(s)	R/W	Default	Description
31-24	R/W	0	blkbar_black_level threshold to judge a black point
20-8	R/W	0	blkbar_hwidth left and right region width
7-5	R/W	0	blkbar_comp_sel select yin or uin or vin to be the valid input
4	R/W	0	blkbar_sw_statistic_en enable software statistic of each block black points number
3	R/W	0	blkbar_det_en
2-1	R/W	0	blkbar_din_sel
0	R/W	0	Blkbar_det_top_en

**Table 9-2988 VDIN1\_BLKBAR\_H\_START\_END 0x1361**

Bit(s)	R/W	Default	Description
28-16	R/W	0	blkbar_hstart. Left region start
12-0	R/W	0	blkbar_hend. Right region end

**Table 9-2989 VDIN1\_BLKBAR\_V\_START\_END 0x1362**

Bit(s)	R/W	Default	Description
28-16	R/W	0	blkbar_vstart.
12-0	R/W	0	blkbar_vend.

**Table 9-2990 VDIN1\_BLKBAR\_CNT\_THRESHOLD 0x1363**

Bit(s)	R/W	Default	Description
19-0	R/W	0	blkbar_cnt_threshold. threshold to judge whether a block is totally black



**Table 9-2991 VDIN1\_BLKBAR\_ROW\_TH1\_TH2 0x1364**

Bit(s)	R/W	Default	Description
28-16	R/W	0	blkbar_row_th1. //threshold of the top blackbar
12-0	R/W	0	blkbar_row_th2 //threshold of the bottom blackbar

**Table 9-2992 VDIN1\_BLKBAR\_IND\_LEFT\_START\_END 0x1365**

Bit(s)	R/W	Default	Description
28-16	R	0	blkbar_ind_left_start. horizontal start of the left region in the current searching
12-0	R	0	blkbar_ind_left_end. horizontal end of the left region in the current searching

**Table 9-2993 VDIN1\_BLKBAR\_IND\_RIGHT\_START\_END 0x1366**

Bit(s)	R/W	Default	Description
28-16	R	0	blkbar_ind_right_start.horizontal start of the right region in the current searching
12-0	R	0	blkbar_ind_right_end. horizontal end of the right region in the current searching

**Table 9-2994 VDIN1\_BLKBAR\_IND\_LEFT1\_CNT 0x1367**

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_left1_cnt. Black pixel counter. left part of the left region

**Table 9-2995 VDIN1\_BLKBAR\_IND\_LEFT2\_CNT 0x1368**

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_left2_cnt. Black pixel counter. right part of the left region

**Table 9-2996 VDIN1\_BLKBAR\_IND\_RIGHT1\_CNT 0x1369**

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_right1_cnt. Black pixel counter. left part of the right region

**Table 9-2997 VDIN1\_BLKBAR\_IND\_RIGHT2\_CNT 0x136a**

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_right2_cnt. Black pixel counter. right part of the right region

**Table 9-2998 VDIN1\_BLKBAR\_STATUS0 0x136b**

Bit(s)	R/W	Default	Description
29	R	0	blkbar_ind_black_det_done. LEFT/RIGHT Black detection done
28-16	R	0	blkbar_top_pos. Top black bar position
12-0	R	0	blkbar_bot_pos. Bottom black bar position

**Table 9-2999 VDIN1\_BLKBAR\_STATUS1 0x136c**

Bit(s)	R/W	Default	Description
28-16	R	0	blkbar_left_pos. Left black bar position
12-0	R	0	blkbar_right_pos. Right black bar position

**Table 9-3000 VDIN1\_WIN\_H\_START\_END 0x136d**

Bit(s)	R/W	Default	Description
28-16	R/W	0	input window H start
12-0	R/W	0	input window H end

**Table 9-3001 VDIN1\_WIN\_V\_START\_END 0x136e**

Bit(s)	R/W	Default	Description
28-16	R/W	0	input window V start
12-0	R/W	0	input window V end

**Table 9-3002 VDIN1\_ASFIFO\_CTRL3 0x136f**

Bit(s)	R/W	Default	Description
15	R/W	0	Vdi7 DE enable
14	R/W	0	Vdi7 go field enable
13	R/W	0	Vdi7 go line enable
12	R/W	0	Vdi7 if true, negative active input vsync
11	R/W	0	Vdi7 if true, negative active input hsync
10	R/W	0	Vdi7 vsync soft reset fifo enable
9	R/W	0	Vdi7 overflow status clear
8	R/W	0	Vdi7 asfifo soft reset, level signal
7	R/W	0	Vdi6 DE enable
6	R/W	0	Vdi6 go field enable
5	R/W	0	Vdi6 go line enable
4	R/W	0	Vdi6 if true, negative active input vsync
3	R/W	0	Vdi6 if true, negative active input hsync

Bit(s)	R/W	Default	Description
2	R/W	0	Vdi6 vsync soft reset fifo enable
1	R/W	0	Vdi6 overflow status clear
0	R/W	0	Vdi6 asfifo soft reset, level signal

Table 9-3003 VDIN1\_COM\_GCLK\_CTRL2 0x1370

Bit(s)	R/W	Default	Description
3-2	R/W	0	Vshrck_clk2 ctrl
1-0	R/W	0	Vshrck_clk1 ctrl

Table 9-3004 VDIN1\_VSHRK\_CTRL 0x1371

Bit(s)	R/W	Default	Description
27	R/W	0	Vshrck enable
26:25	R/W	0	Vshrck mode, 0: 1/2 shrink, 1: 1/4 shrink, 2: 1/8 shrink
24	R/W	0	Vshrink lpf mode, 1: 0.5,1.5,1.5,0.5 lpf for 1/4 shrink, 0.5,1.5,1.5...for 1/8 shrink
23:0	R/W	0	Vshrink padding dummy data

Table 9-3005 VDIN1\_HIST32 0x1372

Bit(s)	R/W	Default	Description
31:0	R	0	Hist 32 mode, [31:16] for white pixel number, 15:0 for black pixel number

Table 9-3006 VDIN1\_COM\_STATUS3 0x1373

Bit(s)	R/W	Default	Description
7	R	0	Vdi9 fifo overflow
5:0	R	0	Vdi9 asfifo cnt

Table 9-3007 VDIN1\_SYNC\_MASK 0x1374

Bit(s)	R/W	Default	Description
17	R/W	0	Vsync_mask_en
16	R/W	0	Hsync_mask_en
15-8	R/W	64	Vsync_mask_num: When vsync_mask_en=1, vdin_output_vsync will keep zero for vsync_mask_num*1024 vdin clock cycle after vdin_input_vsync pluse. Example: vdin_clock = 666MHz, vsync_mask_num=255, vdin out vsync will at least keep zero for 0.392ms after one input_vsync
7-0	R/W	64	Hsync_mask_num When hsync_mask_en=1, vdin_output_hsync will keep zero for hsync_mask_num*32 vdin clock cycle after vdin_input_vsync pluse

**Table 9-3008 VDIN1\_DOLBY\_DSC\_CTRL0 0x1375**

Bit(s)	R/W	Default	Description
31	R/W	0	Dolby check enable
30	R/W	0	Tunnel swap mode enable
29-24	R/W	0	Soft reset control, 29 : dsc, 28~27: crc check, 26~24, reserved
16	R/W	0	Little endian mode
15-0	R/W	0	Monitor metadata position

**Table 9-3009 VDIN1\_DOLBY\_DSC\_CTRL1 0x1376**

Bit(s)	R/W	Default	Description
31-16	R/W	0	Metadata pixel start position
7-0	R/W	0	Crc check control

**Table 9-3010 VDIN1\_DOLBY\_DSC\_CTRL2 0x1377**

Bit(s)	R/W	Default	Description
31	R/W	0	Metadata read enable
30	R/W	0	Metadata read address auto-increment
29-20	R/W	0	Metadata sum
17-0	R/W	0	Tunnel mode channel selection

**Table 9-3011 VDIN1\_DOLBY\_DSC\_CTRL3 0x1378**

Bit(s)	R/W	Default	Description
15-0	R/W	0	Select metadata position

**Table 9-3012 VDIN1\_DOLBY\_AXI\_CTRL0 0x1379**

Bit(s)	R/W	Default	Description
31	R/W	0	Memory read enable
30	R/W	0	AXI bus read enable
29	R/W	0	AXI write channel urgent
28	R/W	0	Pack mode little endian
27	R/W	0	AXI bus soft reset
26	R/W	0	Frame reset enable
25	R/W	0	Memory read protection enable
24-16	R/W	0	Memory read sum
15-8	R/W	0	AXI request hold line
7-6	R/W	0	AXI burst length
5	R/W	0	Frame buffer start

Bit(s)	R/W	Default	Description
4	R/W	0	Buffer start
3-2	R/W	0	AXI status monitor control
1-0	R/W	0	awid

**Table 9-3013 VDIN1\_DOLBY\_AXI\_CTRL1 0x137a**

Bit(s)	R/W	Default	Description
31-0	R/W	0	Buffer start address

**Table 9-3014 VDIN1\_DOLBY\_AXI\_CTRL2 0x137b**

Bit(s)	R/W	Default	Description
31-16	R/W	0	Buffer size
15-0	R/W	0	Frame buffer size

**Table 9-3015 VDIN1\_DOLBY\_AXI\_CTRL3 0x137c**

Bit(s)	R/W	Default	Description
7-0	R/W	0	Hold cycle

**Table 9-3016 VDIN1\_CRC\_CHK 0x13c9**

Bit(s)	R/W	Default	Description
0	W	0	vdin crc check start (pulse)

**Table 9-3017 VDIN1\_RO\_CRC 0x13ca**

Bit(s)	R/W	Default	Description
31-0	R	0	vdin crc result (lock by go_field)

**Table 9-3018 VDIN1\_LINE\_INT 0x13cb**

Bit(s)	R/W	Default	Description
31	R	0	line interrupt flag
30	R	0	write done flag
17	R/W	0	clear line interrupt flag
16	R/W	0	clear write done interrupt flag
15	R/W	0	enable line interrupt
12-0	R/W	0	line interrupt number

**Table 9-3019 VDIN1\_HDR2\_SIZE 0x13d8**

Bit(s)	R/W	Default	Description
28-16	R/W	0x438	Vdin_hdr2_vsize
12-0	R/W	0x780	Vdin_hdr2_hsize

**Table 9-3020 VDIN1\_VSHRK\_SIZE\_M1 0x13d9**

Bit(s)	R/W	Default	Description
28-16	R/W	0x10d	Vshrink input vsize m1
12-0	R/W	0x437	Vshrink input hsize m1

**Table 9-3021 VDIN2\_WR\_CTRL2 0x4102**

Bit(s)	R/W	Default	Description
19	R/W	0	Vdin2_wr_bit10_mode
18	R/W	0	Data_ext_en 1: send out data if req was interrupt by soft reset 0 : normal mode
17:16	R/W	1	Words_lim[1:0]: it would not send out request before Words_lim *16 words were ready
15:12	R/W	1	Burst_lim : 00 , 1 word in 1burst , 01 ,2 words in 1burst, 10, 4 words in 1burst , 11 reserved
10:9	R/W	0	Words_lim[3:2]
8	R/W	0	1: discard data before line fifo, 0: normal mode
7-0	R/W	0	Write chroma canvas address, for NV12/21 mode.

**Table 9-3022 VDIN2\_WR\_CTRL 0x4101**

Bit(s)	R/W	Default	Description
31-30	R/W	0	Vdin2_wr_mif_hconv_mode. Applicable only to vdin_write_format=0 or 2. 0=Output every even pixel's CbCr; 1=Output every odd pixel's CbCr; 2=Output an average value per even pair of pixels; 3=Output all CbCr. Only applies to vdin_write_format =2.
29	R/W	0	vdin1_wr_mif_no_clk_gate. If true, enable free-run clock.
28	R/W	0	clear write response counter in the vdin write memory interface
27	R/W	1	eol_sel, 1: use eol as the line end indication, 0: use width as line end indication in the vdin write memory interface
23	R/W	1	vdin frame reset enable, if true, it will provide frame reset during go_field(vsync) to the modules after that
22	R/W	1	vdin line fifo soft reset enable, meaning, if true line fifo will reset during go_field (vsync)
21	R/W	0	vdin direct write done status clear bit
20	R/W	0	vdin NR write done status clear bit
19	R/W	0	Vdin0_wr_words_swap : swap the 2 64bits word in 128 words
18	R/W	0	vdin1_wr_mif_swap_cbcr. Applicable only to vdin_write_format =2.

Bit(s)	R/W	Default	Description
			0=Output CbCr (NV12); 1=Output CrCb (NV21);
17:16	R/W	0	vdin1_wr_mif_vconv_mode. Applicable only to vdin_write_format=2. 0=Output every even line's CbCr; 1=Output every odd line's CbCr; 2=Reserved; 3=Output all CbCr.
13-12	R/W	0	vdin_write_format, 0: 4:2:2 to one canvas; 1: 4:4:4 to one canvas; 2: Y to luma canvas, CbCr to chroma canvas, for NV12/21; 3: 422 10 bit full pack mode
11	R/W	0	vdin write canvas double buffer enable, means the canvas address will be latched by vsync before using
9	R/W	0	vdin write request urgent
8	R/W	0	vdin write request enable
7-0	R/W	0	Write canvas address (For NV12/21 mode, it's LUMA canvas)

Table 9-3023 VDIN2\_WR\_H\_START\_END 0x4103

Bit(s)	R/W	Default	Description
27-16	R/W	0	H start
11-0	R/W	0	H end

Table 9-3024 VDIN2\_WR\_V\_START\_END 0x4104

Bit(s)	R/W	Default	Description
27-16	R/W	0	V start
11-0	R/W	0	V end

Table 9-3025 VI\_HIST\_CTRL 0x2e00

Bit(s)	R/W	Default	Description
17-16	R/W	0	Spl_sft: the spl are right shift by spl_sft, 0: no shift, 1: right shift by 1, 2,3...
14	R/W	0	Hist_34bin_only, bin 32~63 are not valid, there are 34bins, bin0~bin31, and bin 64 for black pixel, bin 65 for white pixel
13-11	R/W	0	Hist_in_sel: 0: vpp_dout, 1: vpp_vd1_din, 2: vpp_vd2_din, 3: osd1, 4:osd2 5: di pre 6: vdin 7: post blend
10-8	R/W	0	Hist_din_comp_mux: mux of each component, din[9:0],[19:10],[29:20] switches
7-5	R/W	0	Hist_dnlp_low: hist number are shift by (hist_dnlp_low + 3). I.e. Dnlp_low =0, >> 3, dnlp_low=1, >> 4
1	R/W	0	Hist_win_en: hist statistic in a window
0	R/W	0	Luma_histSpl_en, 1: enable the histogram statistic

Table 9-3026 VDIN1\_WRARB\_MODE 0x13c0

Bit(s)	R/W	Default	Description
23:16	R/W	0	wrarb_sel : uns, default = 0 , wrarb_sel [0]==0 slave dc0 connect master port0 wrarb_sel[0]==1 slave dc0 connect master port1 wrarb_sel [1]==0 slave dc1 connect master port0 wrarb_sel[1]==1 slave dc1 connect master port1 wrarb_sel [2]==0 slave dc2 connect master port0 wrarb_sel[2]==1 slave dc2 connect master port1 wrarb_sel [3]==0 slave dc3 connect master port0 wrarb_sel[3]==1 slave dc3 connect master port1 wrarb_sel [4]==0 slave dc4 connect master port0 wrarb_sel [4]==1 slave dc3 connect master port1 wrarb_sel [5]==0 slave dc5 connect master port0 wrarb_sel[5]==1 slave dc3 connect master port1
8	R/W	0	wrarb_arb_mode : uns, default = 0 , wrarb_arb_mode[0] master port0 arb way,
1:0	R/W	0	wrarb_gate_clk_ctrl : uns, default = 0 , wrarb_gate_clk_ctrl[1:0] master port0 clk gate control

Table 9-3027 VDIN1\_WRARB\_REQEN\_SLV 0x13c1

Bit(s)	R/W	Default	Description
7:0	R/W	0	wrarb_dc_req_en : unsigned , default = 0 wrarb_dc_req_en [0]: the slv0 req to mst port0 enable, wrarb_dc_req_en [1]: the slv1 req to mst port0 enable, wrarb_dc_req_en [2]: the slv2 req to mst port0 enable, wrarb_dc_req_en [0]: the slv0 req to mst port1 enable, wrarb_dc_req_en [1]: the slv1 req to mst port1 enable, wrarb_dc_req_en [2]: the slv2 req to mst port1 enable,

Table 9-3028 VDIN1\_WRARB\_WEIGHT0\_SLV 0x13C2

Bit(s)	R/W	Default	Description
29:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 wrdc_weigh_sxn [0*6+:6]: the slv0 req weigh number wrdc_weigh_sxn [1*6+:6]: the slv1 req weigh number wrdc_weigh_sxn [2*6+:6]: the slv2 req weigh number wrdc_weigh_sxn [3*6+:6]: the slv3 req weigh number wrdc_weigh_sxn [4*6+:6]: the slv4 req weigh number

Table 9-3029 VDIN1\_WRARB\_WEIGHT1\_SLV 0x13C3

Bit(s)	R/W	Default	Description
17:0	R/W	0	wrdc_weigh_sxn : unsigned , default = 0 the slv5 req weigh number [0*6+:6],the slv6 req weighnumber [1*6+:6],the slv7 req weighnumber [2*6+:6],

Table 9-3030 VDIN1\_RDWR\_ARB\_STATUS 0x13c4

Bit(s)	R/W	Default	Description
2	R.O	0	ro_wrarb_arb_busy : unsigned , default = 0
1	R/W	0x0	reserve :
0	R.O	0	ro_rdarb_arb_busy : unsigned , default = 0

Table 9-3031 VDIN1\_ARB\_DBG\_CTRL 0x13c5

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_det_cmd_ctrl : unsigned , default = 0



**Table 9-3032 VDIN1\_ARB\_DBG\_STAT 0x13C6**

Bit(s)	R/W	Default	Description
31:0	R.O	0	ro_det_dbg_stat : unsigned , default = 0

**Table 9-3033 VDIN1\_DSC\_CTRL 0x13d0**

Bit(s)	R/W	Default	Description
9:4	R/W	36	reg_dithout_switch : ,uns, default = 36;://{2'h2,2'h1,2'h0}
3	R/W	1	reg_detunnel_en : ,uns, default = 1;
2	R/W	0	reg_detunnel_u_start : ,uns, default = 0;
1	R/W	1	reg_vdin_dith_en : ,uns, default = 1;
0	R/W	1	reg_descramble_en : ,uns, default = 1;

**Table 9-3034 VDIN1\_CFMT\_CTRL 0x13d1**

Bit(s)	R/W	Default	Description
8	R/W	0	reg_chfmt_rpt_pix : uns, default = 0 ; // if true, horizontal formatter use repeating to generete pixel, otherwise use bilinear interpolation
7:4	R/W	0	reg_chfmt_ini_phase : uns, default = 0 ; // horizontal formatter initial phase
3	R/W	0	reg_chfmt_rpt_p0_en : uns, default = 0 ; // horizontal formatter repeat pixel 0 enable
2:1	R/W	1	reg_chfmt_yc_ratio : uns, default = 1 ; // horizontal Y/C ratio, 00: 1:1, 01: 2:1, 10: 4:1
0	R/W	1	reg_chfmt_en : uns, default = 1 ; // horizontal formatter enable

**Table 9-3035 VDIN1\_CFMT\_W 0x13d2**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	reg_chfmt_w : uns, default = 1920 ;horizontal formatter width
12:0	R/W	960	reg_cvfmt_w : uns, default = 960 ;vertical formatter width

**Table 9-3036 VDIN1\_SCB\_CTRL0 0x13d3**

Bit(s)	R/W	Default	Description
11	R/W	1	reg_444c422_gofield_en : uns, default = 1;
10	R/W	1	reg_tunnel_en : uns, default = 1;
9:4	R/W	36	reg_tunnel_outswitch : uns, default = 36;://{2'h2,2'h1,2'h0}
3:2	R/W	0	reg_444c422_mode : uns, default = 0; //0:left 1:right 2,3:avg
1	R/W	0	reg_444c422_bypass : uns, default = 0; 1:bypass

**Table 9-3037 VDIN1\_SCB\_CTRL1 0x13d4**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	reg_444c422_hsize : uns, default = 1920 ;horizontal size
12:0	R/W	960	reg_444c422_vsize : uns, default = 960 ;vertical size

**Table 9-3038 VDIN1\_DSC\_HSIZE 0x13d5**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	reg_detunnel_hsize : uns, default = 1920 ;
12:0	R/W	1920	reg_dither_hsize : uns, default = 1920 ;

**Table 9-3039 VDIN1\_DSC\_DETUNNEL\_SEL 0x13d6**

Bit(s)	R/W	Default	Description
17:0	R/W	34658	reg_detunnel_sel : uns, default = 34658; //{3'h1,3'h0,3'h3 ,3'h5,3'h4,3'h2}

**Table 9-3040 VDIN1\_DSC\_TUNNEL\_SEL 0x13d7**

Bit(s)	R/W	Default	Description
17:0	R/W	69868	reg_tunnel_sel : uns, default = 69868; //= {3'h2,3'h1 ,3'h0,3'h3 ,3'h5,3'h4};

**Table 9-3041 VDIN1\_HDR2\_SIZE 0x13d8**

Bit(s)	R/W	Default	Description
28-16	R/W	0x438	Vdin_hdr2_vsize
12-0	R/W	0x780	Vdin_hdr2_hsize

**Table 9-3042 VDIN1\_VSHRK\_SIZE\_M1 0x13d9**

Bit(s)	R/W	Default	Description
28-16	R/W	0x10d	Vshrink input vsize m1
12-0	R/W	0x437	Vshrink input hsize m1

**Table 9-3043 VDIN1\_HSK\_CTRL 0x13ef**

Bit(s)	R/W	Default	Description
22:16	R/W	4	reg_hshrink_size, 2: 1/2 4: 1/4 hshrink, max=64: 1/64 hshrink
12:0	R/W	1920	reg_frm_hsize

**Table 9-3044 VDIN1\_HSK\_COEF\_0 0x13f0**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef00

**Table 9-3045 VDIN1\_HSK\_COEF\_1 0x13f1**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef01

VDIN1\_DITH\_CTRL 0x13e0

Same as VPU\_VENCL\_DITH\_CTRL

VDIN1\_DITH\_LUT\_1 0x13e1

Same as VPU\_VENCL\_DITH\_LUT\_1

VDIN1\_DITH\_LUT\_2 0x13e2

Same as VPU\_VENCL\_DITH\_LUT\_2

VDIN1\_DITH\_LUT\_3 0x13e3

Same as VPU\_VENCL\_DITH\_LUT\_3

VDIN1\_DITH\_LUT\_4 0x13e4

Same as VPU\_VENCL\_DITH\_LUT\_4

VDIN1\_DITH\_LUT\_5 0x13e5

Same as VPU\_VENCL\_DITH\_LUT\_5

VDIN1\_DITH\_LUT\_6 0x13e6

Same as VPU\_VENCL\_DITH\_LUT\_6

VDIN1\_DITH\_LUT\_7 0x13e7

Same as VPU\_VENCL\_DITH\_LUT\_7

VDIN1\_DITH\_LUT\_8 0x13e8

Same as VPU\_VENCL\_DITH\_LUT\_8

VDIN1\_DITH\_LUT\_9 0x13e9

Same as VPU\_VENCL\_DITH\_LUT\_9

VDIN1\_DITH\_LUT\_10 0x13ea

Same as VPU\_VENCL\_DITH\_LUT\_10

VDIN1\_DITH\_LUT\_10 0x13eb

Same as VPU\_VENCL\_DITH\_LUT\_10

VDIN1\_DITH\_LUT\_11 0x13ec

Same as VPU\_VENCL\_DITH\_LUT\_11

**Table 9-3046 VDIN1\_HSK\_COEF\_2 0x13f2**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef02

**Table 9-3047 VDIN1\_HSK\_COEF\_3 0x13f3**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef03

**Table 9-3048 VDIN1\_HSK\_COEF\_4 0x13f4**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef04

**Table 9-3049 VDIN1\_HSK\_COEF\_5 0x13f5**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef05

**Table 9-3050 VDIN1\_HSK\_COEF\_6 0x13f6**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef06

**Table 9-3051 VDIN1\_HSK\_COEF\_7 0x13f7**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef07

**Table 9-3052 VDIN1\_HSK\_COEF\_8 0x13f8**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef08

**Table 9-3053 VDIN1\_HSK\_COEF\_9 0x13f9**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef09

**Table 9-3054 VDIN1\_HSK\_COEF\_10 0x13fa**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef10

**Table 9-3055 VDIN1\_HSK\_COEF\_11 0x13fb**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef11

**Table 9-3056 VDIN1\_HSK\_COEF\_12 0x13fc**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef12

**Table 9-3057 VDIN1\_HSK\_COEF\_13 0x13fd**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef13

**Table 9-3058 VDIN1\_HSK\_COEF\_14 0x13fe**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef14

**Table 9-3059 VDIN1\_HSK\_COEF\_15 0x13ff**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_hsk_coef15

**Table 9-3060 VI\_HIST\_H\_START\_END 0x2e01**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Hist_hstart, refer to VI_HIST_CTRL[1]
12-0	R/W	0	Hist_hend

**Table 9-3061 VI\_HIST\_V\_START\_END 0x2e02**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Hist_vstart, refer to VI_HIST_CTRL[1]
12-0	R/W	0	Hist_vend

**Table 9-3062 VI\_HIST\_MAX\_MIN 0x2e03**

Bit(s)	R/W	Default	Description
15-8	R	0	hist_max maximum value
7-0	R	0	hist_min minimum value

**Table 9-3063 VI\_HIST\_SPL\_VAL 0x2e04**

Bit(s)	R/W	Default	Description
31-0	R	0	hist_spl_rd , counts for the total luma value

**Table 9-3064 VI\_HIST\_SPL\_PIX\_CNT 0x2e05**

Bit(s)	R/W	Default	Description
21-0	R	0	hist_spl_pixel_count, counts for the total calculated pixels

**Table 9-3065 VI\_HIST\_CHROMA\_SUM 0x2e06**

Bit(s)	R/W	Default	Description
31-0	R	0	hist_chroma_sum , counts for the total chroma value

**Table 9-3066 VI\_DNLP\_HIST00 0x2e07**

//0-255 are split to 64 bins evenly, and VDIN\_DNLP\_HISTXX //are the statistic number of pixels that within each bin.

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 2nd bin
15-0	R	0	counts for the 1st bin

**Table 9-3067 VI\_DNLP\_HIST01 0x2e08**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 4th bin
15-0	R	0	counts for the 3rd bin

**Table 9-3068 VI\_DNLP\_HIST02 0x2e09**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 6th bin
15-0	R	0	counts for the 5th bin

**Table 9-3069 VI\_DNLP\_HIST03 0x2e0a**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 8th bin
15-0	R	0	counts for the 7th bin

**Table 9-3070 VI\_DNLP\_HIST04 0x2e0b**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 10th bin
15-0	R	0	counts for the 9th bin

**Table 9-3071 VI\_DNLP\_HIST05 0x2e0c**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 12th bin
15-0	R	0	counts for the 11th bin

**Table 9-3072 VI\_DNLP\_HIST06 0x2e0d**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 14th bin
15-0	R	0	counts for the 13th bin

**Table 9-3073 VI\_DNLP\_HIST07 0x2e0e**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 16th bin
15-0	R	0	counts for the 15th bin

**Table 9-3074 VI\_DNLP\_HIST08 0x2e0f**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 18th bin
15-0	R	0	counts for the 17th bin

**Table 9-3075 VI\_DNLP\_HIST09 0x2e10**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 20th bin
15-0	R	0	counts for the 19th bin

**Table 9-3076 VI\_DNLP\_HIST10 0x2e11**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 22nd bin
15-0	R	0	counts for the 21st bin

**Table 9-3077 VI\_DNLP\_HIST11 0x2e12**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 24th bin
15-0	R	0	counts for the 23rd bin

**Table 9-3078 VI\_DNLP\_HIST12 0x2e13**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 26th bin
15-0	R	0	counts for the 25th bin

**Table 9-3079 VI\_DNLP\_HIST13 0x2e14**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 28th bin
15-0	R	0	counts for the 27th bin

**Table 9-3080 VI\_DNLP\_HIST14 0x2e15**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 30th bin
15-0	R	0	counts for the 29th bin

**Table 9-3081 VI\_DNLP\_HIST15 0x2e16**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 32nd bin
15-0	R	0	counts for the 31st bin

**Table 9-3082 VI\_DNLP\_HIST16 0x2e17**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 34th bin
15-0	R	0	counts for the 33rd bin

**Table 9-3083 VI\_DNLP\_HIST17 0x2e18**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 36th bin
15-0	R	0	counts for the 35th bin

**Table 9-3084 VI\_DNLP\_HIST18 0x2e19**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 38th bin
15-0	R	0	counts for the 37th bin



**Table 9-3085 VI\_DNLP\_HIST19 0x2e1a**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 40th bin
15-0	R	0	counts for the 39th bin

**Table 9-3086 VI\_DNLP\_HIST20 0x2e1b**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 42nd bin
15-0	R	0	counts for the 41st bin

**Table 9-3087 VI\_DNLP\_HIST21 0x2e1c**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 44th bin
15-0	R	0	counts for the 43rd bin

**Table 9-3088 VI\_DNLP\_HIST22 0x2e1d**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 46th bin
15-0	R	0	counts for the 45th bin

**Table 9-3089 VI\_DNLP\_HIST23 0x2e1e**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 48th bin
15-0	R	0	counts for the 47th bin

**Table 9-3090 VI\_DNLP\_HIST24 0x2e1f**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 50th bin
15-0	R	0	counts for the 49th bin

**Table 9-3091 VI\_DNLP\_HIST25 0x2e20**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 52nd bin
15-0	R	0	counts for the 51st bin

**Table 9-3092 VI\_DNLP\_HIST26 0x2e21**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 54th bin
15-0	R	0	counts for the 53rd bin

**Table 9-3093 VI\_DNLP\_HIST27 0x2e22**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 56th bin
15-0	R	0	counts for the 55th bin

**Table 9-3094 VI\_DNLP\_HIST28 0x2e23**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 58th bin
15-0	R	0	counts for the 57th bin

**Table 9-3095 VI\_DNLP\_HIST29 0x2e24**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 60th bin
15-0	R	0	counts for the 59th bin

**Table 9-3096 VI\_DNLP\_HIST30 0x2e25**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 62nd bin
15-0	R	0	counts for the 61st bin

**Table 9-3097 VI\_DNLP\_HIST31 0x2e26**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 64th bin
15-0	R	0	counts for the 63rd bin

**Table 9-3098 VI\_DNLP\_HIST31 0x2e27**

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 66th bin, for white pix
15-0	R	0	counts for the 65th bin, for black pix

**Table 9-3099 VI\_HIST\_PIC\_SIZE 0x2e28**

Bit(s)	R/W	Default	Description
28-16	R/W	0	Hist_pic_height
12-0	R/W	0	Hist_pic_width

**Table 9-3100 VI\_HIST\_GCLK\_CTRL 0x2e2a**

Bit(s)	R/W	Default	Description
5-4	R/W	0	Gated clock control of hist_clk
3-2	R/W	0	Gated clock control of clk0
1-0	R/W	0	Gated clock control of hist register clock

### 9.2.3.43 Osd\_mali\_afbcd Registers

**Table 9-3101 VPU\_MAFBC\_BLOCK\_ID 0x3a00**

Bit(s)	R/W	Default	Description
31:16	R.O	0x0	PRODUCT_ID : // Contains a product-specific value
15:12	R.O	0x0	VERSION_MAJOR : // Major release number of the AFBC decoder. This is the R part of an RnPn release number.
11:4	R.O	0x0	VERSION_MINOR : // Minor release number of the AFBC decoder. This is the P part of an RnPn release number.
3:0	R.O	0x0	VERSION_STATUS : // The version status of the AFBC decoder release. Starts at 0 and increases by one for each release.

**Table 9-3102 VPU\_MAFBC\_IRQ\_RAW\_STATUS 0x3a01**

Bit(s)	R/W	Default	Description
5	R.O	0x0	SECURE_ID_ERROR : // Activated when an invalid TrustZone Media Protection (TZMP) transaction is detected on the TZMP1 or TZMP2 data fields.
4	R.O	0x0	AXI_ERROR : // Activated when an AXI error is detected.
3	R.O	0x0	DETILING_ERROR : // Activated when a detiling error occurs.
2	R.O	0x0	DECODE_ERROR : // Activated when decoder core indicates a decoder error.
1	R.O	0x0	CONFIGURATION_SWAPPED : // Activated when configuration has been swapped from shadow registers to configuration registers.
0	R.O	0x0	SURFACES_COMPLETED : // Activated when all enabled surfaces have completed and have been fully read out. In continuous mode, this interrupt is triggered each time all surfaces have been read out.

**Table 9-3103 VPU\_MAFBC\_IRQ\_CLEAR 0x3a02**

Bit(s)	R/W	Default	Description
5	R/W	0	SECURE_ID_ERROR : // unsigned , default = 0,Writing a 1 to this bit clears the corresponding bit in the IRQ_RAW_STATUS register.
4	R/W	0	AXI_ERROR : // unsigned , default = 0,Writing a 1 to this bit clears the corresponding bit in the IRQ_RAW_STATUS register.
3	R/W	0	DETILING_ERROR : // unsigned , default = 0,Writing a 1 to this bit clears the corresponding bit in the IRQ_RAW_STATUS register.
2	R/W	0	DECODE_ERROR : // unsigned , default = 0,Writing a 1 to this bit clears the corresponding bit in the IRQ_RAW_STATUS register.
1	R/W	0	CONFIGURATION_SWAPPED : // unsigned , default = 0,Writing a 1 to this bit clears the corresponding bit in the IRQ_RAW_STATUS register.
0	R/W	0	SURFACES_COMPLETED : // unsigned , default = 0,Writing a 1 to this bit clears the corresponding bit in the IRQ_RAW_STATUS register.

**Table 9-3104 VPU\_MAFBC\_IRQ\_MASK 0x3a03**

Bit(s)	R/W	Default	Description
5	R/W	0	SECURE_ID_ERROR : // unsigned , default = 0,When this is set to 1, the corresponding IRQ_RAW_STATUS interrupt is enabled. When this is set to zero, the corresponding IRQ_RAW_STATUS interrupt is disabled.
4	R/W	0	AXI_ERROR : // unsigned , default = 0,When this is set to 1, the corresponding IRQ_RAW_STATUS interrupt is enabled. When this is set to zero, the corresponding IRQ_RAW_STATUS interrupt is disabled.
3	R/W	0	DETILING_ERROR : // unsigned , default = 0,When this is set to 1, the corresponding IRQ_RAW_STATUS interrupt is enabled. When this is set to zero, the corresponding IRQ_RAW_STATUS interrupt is disabled.
2	R/W	0	DECODE_ERROR : // unsigned , default = 0, When this is set to 1, the corresponding IRQ_RAW_STATUS interrupt is enabled. When this is set to zero, the corresponding IRQ_RAW_STATUS interrupt is disabled.
1	R/W	0	CONFIGURATION_SWAPPED : // unsigned , default = 0, When this is set to 1, the corresponding IRQ_RAW_STATUS interrupt is enabled. When this is set to zero, the corresponding IRQ_RAW_STATUS interrupt is disabled.
0	R/W	0	SURFACES_COMPLETED : // unsigned , default = 0, When this is set to 1, the corresponding IRQ_RAW_STATUS interrupt is enabled. When this is set to zero, the corresponding IRQ_RAW_STATUS interrupt is disabled.

**Table 9-3105 VPU\_MAFBC\_IRQ\_STATUS 0x3a04**

Bit(s)	R/W	Default	Description
5	R.O	0x0	SECURE_ID_ERROR : //When this is set to 1, it asserts the corresponding external interrupt. This signals an interrupt request to the application processor.
4	R.O	0x0	AXI_ERROR : //When this is set to 1, it asserts the corresponding external interrupt. This signals an interrupt request to the application processor.
3	R.O	0x0	DETILING_ERROR : //When this is set to 1, it asserts the corresponding external interrupt. This signals an interrupt request to the application processor.
2	R.O	0x0	DECODE_ERROR : //When this is set to 1, it asserts the corresponding external interrupt. This signals an interrupt request to the application processor.

Bit(s)	R/W	Default	Description
1	R.O	0x0	CONFIGURATION_SWAPPED : //When this is set to 1, it asserts the corresponding external interrupt. This signals an interrupt request to the application processor.
0	R.O	0x0	SURFACES_COMPLETED : //When this is set to 1, it asserts the corresponding external interrupt. This signals an interrupt request to the application processor.

Table 9-3106 VPU\_MAFBC\_COMMAND 0x3a05

Bit(s)	R/W	Default	Description
1	R/W	0x0	PENDING_SWAP : //Performs a swap of shadow registers when the current decode operations are completed. After a swap, decoding of the new surfaces starts.
0	R/W	0x0	DIRECT_SWAP : //Performs a swap of shadow registers immediately. Current decode operations are terminated before completion . After termination, decoding of the new surfaces is started

Table 9-3107 VPU\_MAFBC\_STATUS 0x3a06

Bit(s)	R/W	Default	Description
2	R.O	0x0	ERROR : //When this is 1, it indicates that the AFBC decoder is in an unrecoverable state and must be reset using RESETn to continue operation. This is flagged after a DETILING_ERROR.
1	R.O	0x0	SWAPPING : //When this is 1, the AFBC decoder is swapping surface configurations.
0	R.O	0x0	ACTIVE : //When this is 1, the AFBC decoder is decoding surfaces.

Table 9-3108 VPU\_MAFBC\_SURFACE\_CFG 0x3a07

Bit(s)	R/W	Default	Description
16	R/W	0	CONTINUOUS_DECODING_ENABLE : //unsigned , default = 0,Enables continuous decoding of surfaces when it is set to 1. This bit describes what the decoder does when one set of surfaces has been completed. If this is 1, the decoder immediately begins decoding again. In this case, the configuration for the next surface depends on whether the software has performed a swap operation. If no swap is used, the same configuration is used. If this is 0, the decoder waits for a software input.
3	R/W	0	S3_ENABLE : //unsigned , default = 0,When this is 1, surface 3 enabled.
2	R/W	0	S2_ENABLE : //unsigned , default = 0,When this is 1, surface 2 enabled.
1	R/W	0	S1_ENABLE : //unsigned , default = 0,When this is 1, surface 1 enabled.
0	R/W	0	S0_ENABLE : //unsigned , default = 0,When this is 1, surface 0 enabled.

Table 9-3109 VPU\_MAFBC\_AXI\_CFG 0x3a08

Bit(s)	R/W	Default	Description
7:4	R/W	0	CACHE : //unsigned , default = 0,This value is driven on the ARCACHEM signal.
3:0	R/W	0	QOS : //unsigned , default = 0,This value is driven on the ARQOSM signal.

**Table 9-3110 VPU\_MAFBC\_HEADER\_BUF\_ADDR\_LOW\_S0 0x3a10**

Bit(s)	R/W	Default	Description
31:0	R/W	0	HEADER_BUF_ADDR : [31:0] //unsigned , default = 0,Contains bits [31:0] of the header buffer address,Bits [5:0] must be set to 0 for alignment requirements.

**Table 9-3111 VPU\_MAFBC\_HEADER\_BUF\_ADDR\_HIGH\_S0 0x3a11**

Bit(s)	R/W	Default	Description
15:0	R/W	0	HEADER_BUF_ADDR : [47:32] //unsigned , default = 0,Contains the upper 16 bits of the header buffer address

**Table 9-3112 VPU\_MAFBC\_FORMAT\_SPECIFIER\_S0 0x3a12**

Bit(s)	R/W	Default	Description
19	R/W	0	PAYLOAD_LIMIT_EN : // unsigned , default = 0,Enable payload read address check against the min and max payload address boundaries: 0 Disable 1 Enable. This means that out-of-bound reads output the color black.
18	R/W	0	TILED_HEADER_EN : // unsigned , default = 0,Enables tiled header mode:0 Disable tiled header 1 Enable tiled header
17:16	R/W	0	SUPER_BLOCK_ASPECT : // unsigned , default = 0,Selects superblock aspect ratio 00:16x16 pixels 01:32x8 pixels other:reserved
9	R/W	0	BLOCK_SPLIT : //unsigned , default = 0,Enables block split mode: 0 Block split mode off. 1 Block split mode on.
8	R/W	0	YUV_TRANSFORM : //unsigned , default = 0,Enables the internal YUV transform stage: 0 Internal YUV transform off. 1 Internal YUV transform on.
3:0	R/W	0	PIXEL_FORMAT : //unsigned , default = 0,Contains the pixel format configuration.

**Table 9-3113 VPU\_MAFBC\_BUFFER\_WIDTH\_S0 0x3a13**

Bit(s)	R/W	Default	Description
13:0	R/W	0	BUFFER_WIDTH : //unsigned , default = 0,AFBC buffer width in pixels

**Table 9-3114 VPU\_MAFBC\_BUFFER\_HEIGHT\_S0 0x3a14**

Bit(s)	R/W	Default	Description
13:0	R/W	0	BUFFER_HEIGHT : //unsigned , default = 0,AFBC buffer height in pixels

**Table 9-3115 VPU\_MAFBC\_BOUNDING\_BOX\_X\_START\_S0 0x3a15**

Bit(s)	R/W	Default	Description
12:0	R/W	0	BUFFER_X_START : //unsigned , default = 0,The AFBC buffer bounding box minimum x value. Given in pixels.

**Table 9-3116 VPU\_MAFBC\_BOUNDING\_BOX\_X\_END\_S0 0x3a16**

Bit(s)	R/W	Default	Description
12:0	R/W	0	BUFFER_X_START : //unsigned , default = 0,The AFBC buffer bounding box maximum x value. Given in pixels.

**Table 9-3117 VPU\_MAFBC\_BOUNDING\_BOX\_Y\_START\_S0 0x3a17**

Bit(s)	R/W	Default	Description
12:0	R/W	0	BUFFER_Y_START : //unsigned , default = 0,The AFBC buffer bounding box minimum y value. Given in pixels.

**Table 9-3118 VPU\_MAFBC\_BOUNDING\_BOX\_Y\_END\_S0 0x3a18**

Bit(s)	R/W	Default	Description
12:0	R/W	0	BUFFER_Y_START : //unsigned , default = 0,The AFBC buffer bounding box maximum y value. Given in pixels.

**Table 9-3119 VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_LOW\_S0 0x3a19**

Bit(s)	R/W	Default	Description
31:0	R/W	0	OUTPUT_BUF_ADDR : [31:0] //unsigned , default = 0,Contains bits [31:0] of the output buffer address.Bits [6:0] must be set to 0 for alignment requirements. The address, frame size, and pixel format allocate area in the detiler SRAM for the current surface . You must ensure that the allocated area does not exceed the available SRAM, or overlap with other surfaces or planes that are defined in the output buffer.

**Table 9-3120 VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_HIGH\_S0 0x3a1a**

Bit(s)	R/W	Default	Description
15:0	R/W	0x0	OUTPUT_BUF_ADDR : [47:32] //Contains the upper 16 bits of the output buffer address

**Table 9-3121 VPU\_MAFBC\_OUTPUT\_BUF\_STRIDE\_S0 0x3a1b**

Bit(s)	R/W	Default	Description
15:0	R/W	0	OUTPUT_BUF_STRIDE : [15:0] //unsigned , default = 0,Contains bits [15:0] of the output buffer stride.Bits [6:0] must be set to 0 for alignment requirements. The maximum permitted buffer stride is 8192 pixels wide. The byte size depends on the pixel format for the surface.

**Table 9-3122 VPU\_MAFBC\_PREFETCH\_CFG\_S0 0x3a1c**

Bit(s)	R/W	Default	Description
1	R/W	0	PREFETCH_READ_DIRECTION_Y : // unsigned , default = 0,Defines the pre-fetch read direction in Y: 0 Top to bottom 1 Bottom to top
0	R/W	0	PREFETCH_READ_DIRECTION_X : //unsigned , default = 0,Defines the pre-fetch read direction in X 0 Left to right 1 Right to left

**Table 9-3123 VPU\_MAFBC\_PAYLOAD\_MIN\_LOW\_S0 0x3a1d**

Bit(s)	R/W	Default	Description
31:0	R/W	0	PAYLOAD_MIN_LOW : //unsigned , default = 0, Indicates the lower 32 bits of the AFBC payload buffer minimum address.

**Table 9-3124 VPU\_MAFBC\_PAYLOAD\_MIN\_HIGH\_S0 0x3a1e**

Bit(s)	R/W	Default	Description
15:0	R/W	0	PAYLOAD_MIN_HIGH : //unsigned , default = 0, Indicates the higher 16 bits of the AFBC payload buffer minimum address.

**Table 9-3125 VPU\_MAFBC\_PAYLOAD\_MAX\_LOW\_S0 0x3a1f**

Bit(s)	R/W	Default	Description
31:0	R/W	0x0	PAYLOAD_MIN_LOW : //unsigned , indicates the lower 32 bits of the AFBC payload buffer maximum address.

**Table 9-3126 VPU\_MAFBC\_PAYLOAD\_MAX\_HIGH\_S0 0x3a20**

Bit(s)	R/W	Default	Description
15:0	R/W	0	PAYLOAD_MIN_HIGH : //unsigned , default = 0, Indicates the higher 16 bits of the AFBC payload buffer maximum address.

VPU\_MAFBC\_HEADER\_BUF\_ADDR\_LOW\_S1 0x3a30

See VPU\_MAFBC\_HEADER\_BUF\_ADDR\_LOW\_S0

VPU\_MAFBC\_HEADER\_BUF\_ADDR\_HIGH\_S1 0x3a31

See VPU\_MAFBC\_HEADER\_BUF\_ADDR\_HIGH\_S0

VPU\_MAFBC\_FORMAT\_SPECIFIER\_S1 0x3a32

See VPU\_MAFBC\_FORMAT\_SPECIFIER\_S0

VPU\_MAFBC\_BUFFER\_WIDTH\_S1 0x3a33

See VPU\_MAFBC\_BUFFER\_WIDTH\_S0

VPU\_MAFBC\_BUFFER\_HEIGHT\_S1 0x3a34

See VPU\_MAFBC\_BUFFER\_HEIGHT\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_X\_START\_S1 0x3a35

See VPU\_MAFBC\_BOUNDING\_BOX\_X\_START\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_X\_END\_S1 0x3a36



See VPU\_MAFBC\_BOUNDING\_BOX\_X\_END\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_Y\_START\_S1 0x3a37

See VPU\_MAFBC\_BOUNDING\_BOX\_Y\_START\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_Y\_END\_S1 0x3a38

See VPU\_MAFBC\_BOUNDING\_BOX\_Y\_END\_S0

VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_LOW\_S1 0x3a39

See VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_LOW\_S0

VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_HIGH\_S1 0x3a3a

See VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_HIGH\_S0

VPU\_MAFBC\_OUTPUT\_BUF\_STRIDE\_S1 0x3a3b

See VPU\_MAFBC\_OUTPUT\_BUF\_STRIDE\_S0

VPU\_MAFBC\_PREFETCH\_CFG\_S1 0x3a3c

See VPU\_MAFBC\_PREFETCH\_CFG\_S0

VPU\_MAFBC\_PAYLOAD\_MIN\_LOW\_S1 0x3a3d

See VPU\_MAFBC\_PAYLOAD\_MIN\_LOW\_S0

VPU\_MAFBC\_PAYLOAD\_MIN\_HIGH\_S1 0x3a3e

See VPU\_MAFBC\_PAYLOAD\_MIN\_HIGH\_S0

VPU\_MAFBC\_PAYLOAD\_MAX\_LOW\_S1 0x3a3f

See VPU\_MAFBC\_PAYLOAD\_MAX\_LOW\_S0

VPU\_MAFBC\_PAYLOAD\_MAX\_HIGH\_S1 0x3a40

See VPU\_MAFBC\_PAYLOAD\_MAX\_HIGH\_S0

VPU\_MAFBC\_HEADER\_BUF\_ADDR\_LOW\_S2 0x3a50

See VPU\_MAFBC\_HEADER\_BUF\_ADDR\_LOW\_S0

VPU\_MAFBC\_HEADER\_BUF\_ADDR\_HIGH\_S2 0x3a51

See VPU\_MAFBC\_HEADER\_BUF\_ADDR\_HIGH\_S0

VPU\_MAFBC\_FORMAT\_SPECIFIER\_S2 0x3a52

See VPU\_MAFBC\_FORMAT\_SPECIFIER\_S0

VPU\_MAFBC\_BUFFER\_WIDTH\_S2 0x3a53

See VPU\_MAFBC\_BUFFER\_WIDTH\_S0

VPU\_MAFBC\_BUFFER\_HEIGHT\_S2 0x3a54

See VPU\_MAFBC\_BUFFER\_HEIGHT\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_X\_START\_S2 0x3a55

See VPU\_MAFBC\_BOUNDING\_BOX\_X\_START\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_X\_END\_S2 0x3a56

See VPU\_MAFBC\_BOUNDING\_BOX\_X\_END\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_Y\_START\_S2 0x3a57

See VPU\_MAFBC\_BOUNDING\_BOX\_Y\_START\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_Y\_END\_S2 0x3a58

See VPU\_MAFBC\_BOUNDING\_BOX\_Y\_END\_S0

VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_LOW\_S2 0x3a59

See VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_LOW\_S0

VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_HIGH\_S2 0x3a5a

See VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_HIGH\_S0

VPU\_MAFBC\_OUTPUT\_BUF\_STRIDE\_S2 0x3a5b

See VPU\_MAFBC\_OUTPUT\_BUF\_STRIDE\_S0

VPU\_MAFBC\_PREFETCH\_CFG\_S2 0x3a5c

See VPU\_MAFBC\_PREFETCH\_CFG\_S0

VPU\_MAFBC\_PAYLOAD\_MIN\_LOW\_S2 0x3a5d

See VPU\_MAFBC\_PAYLOAD\_MIN\_LOW\_S0

VPU\_MAFBC\_PAYLOAD\_MIN\_HIGH\_S2 0x3a5e

See VPU\_MAFBC\_PAYLOAD\_MIN\_HIGH\_S0

VPU\_MAFBC\_PAYLOAD\_MAX\_LOW\_S2 0x3a5f

See VPU\_MAFBC\_PAYLOAD\_MAX\_LOW\_S0

VPU\_MAFBC\_PAYLOAD\_MAX\_HIGH\_S2 0x3a60

See VPU\_MAFBC\_PAYLOAD\_MAX\_HIGH\_S0

VPU\_MAFBC\_HEADER\_BUF\_ADDR\_LOW\_S3 0x3a70

See VPU\_MAFBC\_HEADER\_BUF\_ADDR\_LOW\_S0

VPU\_MAFBC\_HEADER\_BUF\_ADDR\_HIGH\_S3 0x3a71

See VPU\_MAFBC\_HEADER\_BUF\_ADDR\_HIGH\_S0

VPU\_MAFBC\_FORMAT\_SPECIFIER\_S3 0x3a72

See VPU\_MAFBC\_FORMAT\_SPECIFIER\_S0

VPU\_MAFBC\_BUFFER\_WIDTH\_S3 0x3a73

See VPU\_MAFBC\_BUFFER\_WIDTH\_S0

VPU\_MAFBC\_BUFFER\_HEIGHT\_S3 0x3a74

See VPU\_MAFBC\_BUFFER\_HEIGHT\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_X\_START\_S3 0x3a75

See VPU\_MAFBC\_BOUNDING\_BOX\_X\_START\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_X\_END\_S3 0x3a76

See VPU\_MAFBC\_BOUNDING\_BOX\_X\_END\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_Y\_START\_S3 0x3a77

See VPU\_MAFBC\_BOUNDING\_BOX\_Y\_START\_S0

VPU\_MAFBC\_BOUNDING\_BOX\_Y\_END\_S3 0x3a78

See VPU\_MAFBC\_BOUNDING\_BOX\_Y\_END\_S0

VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_LOW\_S3 0x3a79

See VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_LOW\_S0

VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_HIGH\_S3 0x3a7a

See VPU\_MAFBC\_OUTPUT\_BUF\_ADDR\_HIGH\_S0

VPU\_MAFBC\_OUTPUT\_BUF\_STRIDE\_S3 0x3a7b

See VPU\_MAFBC\_OUTPUT\_BUF\_STRIDE\_S0

VPU\_MAFBC\_PREFETCH\_CFG\_S3 0x3a7c

See VPU\_MAFBC\_PREFETCH\_CFG\_S0

VPU\_MAFBC\_PAYLOAD\_MIN\_LOW\_S3 0x3a7d

See VPU\_MAFBC\_PAYLOAD\_MIN\_LOW\_S0

VPU\_MAFBC\_PAYLOAD\_MIN\_HIGH\_S3 0x3a7e

See VPU\_MAFBC\_PAYLOAD\_MIN\_HIGH\_S0

VPU\_MAFBC\_PAYLOAD\_MAX\_LOW\_S3 0x3a7f

See VPU\_MAFBC\_PAYLOAD\_MAX\_LOW\_S0

VPU\_MAFBC\_PAYLOAD\_MAX\_HIGH\_S3 0x3a80

See VPU\_MAFBC\_PAYLOAD\_MAX\_HIGH\_S0

### 9.2.3.44 Primesl Registers

**Table 9-3127 PRIMESL\_LUTC\_ADDR\_PORT 0x3980**

Bit(s)	R/W	Default	Description
8: 0	R/W	0	lutc_addr : // unsigned , default = 0 = 'h0,

**Table 9-3128 PRIMESL\_LUTC\_DATA\_PORT 0x3981**

Bit(s)	R/W	Default	Description
10: 0	R/W	0	lutc_data : // unsigned , default = 0 = 'h0,

**Table 9-3129 PRIMESL\_LUTP\_ADDR\_PORT 0x3982**

Bit(s)	R/W	Default	Description
8: 0	R/W	0	lutp_addr : // unsigned , default = 0 = 'h0,

**Table 9-3130 PRIMESL\_LUTP\_DATA\_PORT 0x3983**

Bit(s)	R/W	Default	Description
13: 0	R/W	0	lutp_data : // unsigned , default = 0 = 'h0,

**Table 9-3131 PRIMESL\_LUTD\_ADDR\_PORT 0x3984**

Bit(s)	R/W	Default	Description
8: 0	R/W	0	lutd_data : // unsigned , default = 0 = 'h0,

**Table 9-3132 PRIMESL\_LUTD\_DATA\_PORT 0x3985**

Bit(s)	R/W	Default	Description
11: 0	R/W	0	lutd_data : // unsigned , default = 0 = 'h0,

**Table 9-3133 PRIMESL\_CTRL0 0x3990**

Bit(s)	R/W	Default	Description
29	R/W	0	legacy_mode_en : // unsigned , default = 0 = 'h0
28	R/W	1	clip_en : // unsigned , default = 1 = 'h0
26:16	R/W	0	inv_chroma_ratio : // unsigned , default = 0 = 'h0,
14: 4	R/W	0	inv_y_ratio : // unsigned , default = 0 = 'h0,
3	R/W	0	reg_gclk_ctrl : // unsigned , default = 0 = 'h0,
2: 1	R/W	0	gclk_ctrl : // unsigned , default = 0 = 'h0,
0	R/W	1	primesl_en : // unsigned , default = 1 = 'h0,

**Table 9-3134 PRIMESL\_CTRL1 0x3991**

Bit(s)	R/W	Default	Description
25:16	R/W	0	l_headroom : // unsigned , default = 0 = 'h0,
9: 0	R/W	0	footroom : // unsigned , default = 0 = 'h0,

**Table 9-3135 PRIMESL\_CTRL2 0x3992**

Bit(s)	R/W	Default	Description
9: 0	R/W	0	c_headroom : // unsigned , default = 0 = 'h0,

**Table 9-3136 PRIMESL\_CTRL3 0x3993**

Bit(s)	R/W	Default	Description
29:16	R/W	0	mub : // unsigned , default = 0 = 'h0,
13: 0	R/W	0	mua : // unsigned , default = 0 = 'h0,

**Table 9-3137 PRIMESL\_CTRL4 0x3994**

Bit(s)	R/W	Default	Description
25:16	R/W	0	oct_7_1 : // signed , default = 0 = 'h0,
9: 0	R/W	0	oct_7_0 : // signed , default = 0 = 'h0,

**Table 9-3138 PRIMESL\_CTRL5 0x3995**

Bit(s)	R/W	Default	Description
25:16	R/W	0	oct_7_3 : // signed , default = 0 = 'h0,
9: 0	R/W	0	oct_7_2 : // signed , default = 0 = 'h0,

**Table 9-3139 PRIMESL\_CTRL6 0x3996**

Bit(s)	R/W	Default	Description
25:16	R/W		oct_7_5 : // signed , default = 0 = 'h0,
9: 0	R/W		oct_7_4 : // signed , default = 0 = 'h0,

**Table 9-3140 PRIMESL\_CTRL7 0x3997**

Bit(s)	R/W	Default	Description
9: 0	R/W	0	oct_7_6 : // signed , default = 0 = 'h0,

**Table 9-3141 PRIMESL\_CTRL8 0x3998**

Bit(s)	R/W	Default	Description
28:16	R/W	0	d_lut_threshold_3_1 : // unsigned , default = 0 = 'h0,
12: 0	R/W	0	d_lut_threshold_3_0 : // unsigned , default = 0 = 'h0,

**Table 9-3142 PRIMESL\_CTRL9 0x3999**

Bit(s)	R/W	Default	Description
12: 0	R/W	0	d_lut_threshold_3_2 : // unsigned , default = 0 = 'h0,

**Table 9-3143 PRIMESL\_CTRL10 0x399a**

Bit(s)	R/W	Default	Description
15:12	R/W	0	d_lut_step_4_3 : // unsigned , default = 0 = 'h0,
11: 8	R/W	0	d_lut_step_4_2 : // unsigned , default = 0 = 'h0,
7: 4	R/W	0	d_lut_step_4_1 : // unsigned , default = 0 = 'h0,
3: 0	R/W	0	d_lut_step_4_0 : // unsigned , default = 0 = 'h0,

**Table 9-3144 PRIMESL\_CTRL11 0x399b**

Bit(s)	R/W	Default	Description
28:16	R/W	0	rgb2yuv_9_0 : // signed , default = 0 = 'h0,
12: 0	R/W	0	rgb2yuv_9_1 : // signed , default = 0 = 'h0,

**Table 9-3145 PRIMESL\_CTRL12 0x399c**

Bit(s)	R/W	Default	Description
28:16	R/W	0	rgb2yuv_9_2 : // signed , default = 0 = 'h0,
12: 0	R/W	0	rgb2yuv_9_3 : // signed , default = 0 = 'h0,

**Table 9-3146 PRIMESL\_CTRL13 0x399d**

Bit(s)	R/W	Default	Description
28:16	R/W	0	rgb2yuv_9_4 : // signed , default = 0 = 'h0,
12: 0	R/W	0	rgb2yuv_9_5 : // signed , default = 0 = 'h0,

**Table 9-3147 PRIMESL\_CTRL14 0x399e**

Bit(s)	R/W	Default	Description
28:16	R/W	0	rgb2yuv_9_6 : // signed , default = 0 = 'h0,
12: 0	R/W	0	rgb2yuv_9_7 : // signed , default = 0 = 'h0,

**Table 9-3148 PRIMESL\_CTRL15 0x399f**

Bit(s)	R/W	Default	Description
12: 0	R/W	0	rgb2yuv_9_8 : // signed , default = 0 = 'h0,

**Table 9-3149 PRIMESL\_CTRL16 0x39e0**

Bit(s)	R/W	Default	Description
31	R/W	0	reserved
30	R/W	0	byp_s_lut://unsigned
29	R/W	0	byp_d_lut://unsigned

Bit(s)	R/W	Default	Description
28	R/W	0	byp_mat://unsigned
27	R/W	0	rgb_swap://unsigned
26:24	R/W	0	uv_shift://unsigned
23:22	R/W	0	rgb_shift://unsigned
21:20	R/W	3	rgb_clip://unsigned
19:18	R/W	/	reserved
17:16	R/W	0	rgb_s://unsigned
15:14	R/W	/	reserved
13:0	R/W	1024	reg_s://signed

Table 9-3150 PRIMESL\_OMAT\_OFFSET0 0x39e1

Bit(s)	R/W	Default	Description
31:28	R/W	/	reserved
27:16	R/W	512	pre_offset0// unsigned , default = 0 = 512,
15:12	R/W	/	reserved
11: 0	R/W	512	pre_offset1: // unsigned , default = 0 = 512,

Table 9-3151 PRIMESL\_OMAT\_OFFSET1 0x39e2

Bit(s)	R/W	Default	Description
31:28	R/W	/	reserved
27:16	R/W	512	pre_offset2// unsigned , default = 0 = 512,
15:12	R/W	/	reserved
11: 0	R/W	256	Offset0: // unsigned , default = 0 = 512,

Table 9-3152 PRIMESL\_OMAT\_OFFSET2 0x39e3

Bit(s)	R/W	Default	Description
31:28	R/W	/	reserved
27:16	R/W	2048	offset1// unsigned , default = 0 = 512,
15:12	R/W	/	reserved
11: 0	R/W	2048	offset2: // unsigned , default = 0 = 512,



### 9.2.3.45 VADJ1/RGB\_CONBRI Registers

**Table 9-3153 VPP\_VADJ1\_MISC 0x3280**

Bit(s)	R/W	Default	Description
5:4	R/W	0	Gate clk ctrl for vadj1
3	/	/	/
2	R/W	0	Sync enable 1:vadj1 regs work when go_field come 1:vadj1 regs directly
1	R/W	0	minus black level enable for vadj1
0	R/W	0	Module enable

**Table 9-3154 VPP\_VADJ1\_BLACK\_VAL 0X3281**

Bit(s)	R/W	Default	Description
26:16	R/W	13'h200	Black_luma
10:0	R/W	13'h40	Black_chroma

**Table 9-3155 VPP\_VADJ1\_Y 0x3282**

Bit(s)	R/W	Default	Description
18-8	R/W	0	brightness, signed value
7-0	R/W	0x1d0	contrast, unsigned value, contrast from $0 \leq \text{contrast} < 2$

**Table 9-3156 VPP\_VADJ1\_MA\_MB 0x3283**

$$cb' = cb*ma + cr*mb \quad cr' = cb*mc + cr*md$$

Bit(s)	R/W	Default	Description
25-16	R/W	0x100	MA, signed value, $-2 < MA < 2$
9-0	R/W	0	MB, signed value, $-2 < MB < 2$

**Table 9-3157 VPP\_VADJ1\_MC\_MD 0x3284**

Bit(s)	R/W	Default	Description
25-16	R/W	0	MC, signed value, $-2 < MC < 2$
9-0	R/W	0x100	MD, signed value, $-2 < MD < 2$

**Table 9-3158 VPP\_VADJ1\_CURV\_0 0x3285**

Bit(s)	R/W	Default	Description
31:24	R/W	0	vadj1_softcon_curv0_ci, u8
23:12	R/W	0	vadj1_softcon_curv0_b, u12
11:0	R/W	0	vadj1_softcon_curv0_a, s12

**Table 9-3159 VPP\_VADJ1\_CURV\_1 0x3286**

Bit(s)	R/W	Default	Description
31:13			Reserved
12:4	R/W	0	vadj1_softcon_curv0_g, s9
3			Reserved
2:0	R/W	0	vadj1_softcon_curv0_cs, u3

**Table 9-3160 VPP\_VADJ1\_CURV\_2 0x3287**

Bit(s)	R/W	Default	Description
31:24	R/W	0	vadj1_softcon_curv1_ci, u8
23:12	R/W	0	vadj1_softcon_curv1_b, u12
11:0	R/W	0	vadj1_softcon_curv1_a, s12

**Table 9-3161 VPP\_VADJ1\_CURV\_3 0x3288**

Bit(s)	R/W	Default	Description
31:13			Reserved
12:4	R/W	0	vadj1_softcon_curv1_g, s9
3			Reserved
2:0	R/W	0	vadj1_softcon_curv1_cs, u3

**Table 9-3162 VPP\_VD1\_RGB\_CTRST 0x3289**

Bit(s)	R/W	Default	Description
31:28			Reserved
27:16	R/W	1024	Vd1_rgb_ctrst: contrast for rgb, normalized 1024 as '1.0'
15			Reserved
14:2	R/W	64	Vd1_rgb_ctrst_blklvl: contrast black level to be subtract before and add back after the contrast gain operation
1	R/W	0	Vd1_rgbbst_en: 1 to enable the RGB_BST
0	R/W	1	Vd1_rgb_ctrst_prt: enable signal to protect saturation in rgb (no clipping) during contrast adjustment

**Table 9-3163 VPP\_VD1\_RGB\_BRGHT 0x328A**

Bit(s)	R/W	Default	Description
31:14			Reserved
13:2	R/W	0	Vd1_rgb_brght: brightness level in rgb domain

Bit(s)	R/W	Default	Description
1	R/W	1	Vd1_rgb_brgh_t_prt: enable signal to protect saturation in rgb (no clipping) during brightness adjustment
0	R/W	0	Vd1_rgb_ctrst_dlut_x2: Enable signal to do x2 to the dlut cells before subtracting from the normalized gain_max; 0:x1 1:x2

**Table 9-3164 VPP\_VD1\_RGB\_DLUT\_0\_3 0x328B**

Bit(s)	R/W	Default	Description
31:24	R/W	255	Vd1_rgbbst_dlut0: Differential gain to normalized gain_max to customized protection curve. e.g. [255 205 171 147 128 113 102 93 85 78 73 68] for protection of not boost for pixels larger than 240
23:16	R/W	205	Vd1_rgbbst_dlut1: same as Vd1_rgbbst_dlut0
15:8	R/W	171	Vd1_rgbbst_dlut2: same as Vd1_rgbbst_dlut0
7:0	R/W	147	Vd1_rgbbst_dlut3: same as Vd1_rgbbst_dlut0

**Table 9-3165 VPP\_VD1\_RGB\_DLUT\_4\_7 0x328c**

Bit(s)	R/W	Default	Description
31:24	R/W	128	Vd1_rgbbst_dlut4: same as Vd1_rgbbst_dlut0
23:16	R/W	113	Vd1_rgbbst_dlut5: same as Vd1_rgbbst_dlut0
15:8	R/W	102	Vd1_rgbbst_dlut6: same as Vd1_rgbbst_dlut0
7:0	R/W	93	Vd1_rgbbst_dlut7: same as Vd1_rgbbst_dlut0

**Table 9-3166 VPP\_VD1\_RGB\_DLUT\_8\_11 0x328d**

Bit(s)	R/W	Default	Description
31:24	R/W	85	Vd1_rgbbst_dlut8: same as Vd1_rgbbst_dlut0
23:16	R/W	78	Vd1_rgbbst_dlut9: same as Vd1_rgbbst_dlut0
15:8	R/W	73	Vd1_rgbbst_dlut10: same as Vd1_rgbbst_dlut0
7:0	R/W	68	Vd1_rgbbst_dlut11: same as Vd1_rgbbst_dlut0

**Table 9-3167 VPP\_VADJ2\_MISC 0x32a0**

Bit(s)	R/W	Default	Description
5:4	R/W	0	Gate clk ctrl for vadj2
3	/	/	/
2	R/W	0	Sync enable 1:vadj2 regs work when go_field come 1:vadj2 regs directly
1	R/W	0	minus black level enable for vadj2
0	R/W	0	Module enable

**Table 9-3168 VPP\_VADJ2\_BLACK\_VAL 0x32a1**

Bit(s)	R/W	Default	Description
26:16	R/W	13'h200	Black_luma
10:0	R/W	13'h40	Black_chroma

**Table 9-3169 VPP\_VADJ2\_Y 0x32a2**

Bit(s)	R/W	Default	Description
18-8	R/W	0	brightness, signed value
7-0	R/W	0x1d0	contrast, unsigned value, contrast from $0 \leq \text{contrast} < 2$

**Table 9-3170 VPP\_VADJ2\_MA\_MB 0x32a3**

$$cb' = cb*ma + cr*mb \quad cr' = cb*mc + cr*md$$

Bit(s)	R/W	Default	Description
25-16	R/W	0x100	MA, signed value, $-2 < MA < 2$
9-0	R/W	0	MB, signed value, $-2 < MB < 2$

**Table 9-3171 VPP\_VADJ2\_MC\_MD 0x32a4**

Bit(s)	R/W	Default	Description
25-16	R/W	0	MC, signed value, $-2 < MC < 2$
9-0	R/W	0x100	MD, signed value, $-2 < MD < 2$

**Table 9-3172 VPP\_VADJ2\_CURV\_0 0x32a5**

Bit(s)	R/W	Default	Description
31:24	R/W	0	Vadj2_softcon_curv0_ci, u8
23:12	R/W	0	Vadj2_softcon_curv0_b, u12
11:0	R/W	0	Vadj2_softcon_curv0_a, s12

**Table 9-3173 VPP\_VADJ2\_CURV\_1 0x32a6**

Bit(s)	R/W	Default	Description
31:13			Reserved
12:4	R/W	0	Vadj2_softcon_curv0_g, s9
3			Reserved
2:0	R/W	0	Vadj2_softcon_curv0_cs, u3

**Table 9-3174 VPP\_VADJ2\_CURV\_2 0x32a7**

Bit(s)	R/W	Default	Description
31:24	R/W	0	Vadj2_softcon_curv1_ci, u8
23:12	R/W	0	Vadj2_softcon_curv1_b, u12
11:0	R/W	0	Vadj2_softcon_curv1_a, s12

**Table 9-3175 VPP\_VADJ2\_CURV\_3 0x32a8**

Bit(s)	R/W	Default	Description
31:13			Reserved
12:4	R/W	0	Vadj2_softcon_curv1_g, s9
3			Reserved
2:0	R/W	0	Vadj2_softcon_curv1_cs, u3

**Table 9-3176 VPP\_POST\_RGB\_CTRST 0x32a9**

Bit(s)	R/W	Default	Description
31:28			Reserved
27:16	R/W	1024	Post_rgb_ctrst: contrast for rgb, normalized 1024 as '1.0'
15:12			Reserved
11:2	R/W	64	Post_rgb_ctrst_blkvl: contrast black level to be subtract before and add back after the contrast gain operation
1	R/W	0	Post_rgbbst_en: 1 to enable the RGB_BST
0	R/W	1	Post_rgb_ctrst_prt: enable signal to protect saturation in rgb (no clipping) during contrast adjustment

**Table 9-3177 VPP\_POST\_RGB\_BRGHT 0x32aa**

Bit(s)	R/W	Default	Description
31:14			Reserved
13:2	R/W	0	Post_rgb_brght: brightness level in rgb domain
1	R/W	1	Post_rgb_brght_prt: enable signal to protect saturation in rgb (no clipping) during brightness adjustment
0	R/W	0	Post_rgb_ctrst_dlut_x2: Enable signal to do x2 to the dlut cells before subtracting from the normalized gain_max; 0:x1 1:x2

**Table 9-3178 VPP\_POST\_RGB\_DLUT\_0\_3 0x32ab**

Bit(s)	R/W	Default	Description
31:24	R/W	255	Post_rgbbst_dlut0: Differential gain to normalized gain_max to customized protection curve. e.g. [255 205 171 147 128 113 102 93 85 78 73 68] for protection of not boost for pixels larger than 240
23:16	R/W	205	Post_rgbbst_dlut1: same as Post_rgbbst_dlut0

Bit(s)	R/W	Default	Description
15:8	R/W	171	Post_rgbbst_dlut2: same as Post_rgbbst_dlut0
7:0	R/W	147	Post_rgbbst_dlut3: same as Post_rgbbst_dlut0

Table 9-3179 VPP\_POST\_RGB\_DLUT\_4\_7 0x32ac

Bit(s)	R/W	Default	Description
31:24	R/W	128	Post_rgbbst_dlut4: same as Post_rgbbst_dlut0
23:16	R/W	113	Post_rgbbst_dlut5: same as Post_rgbbst_dlut0
15:8	R/W	102	Post_rgbbst_dlut6: same as Post_rgbbst_dlut0
7:0	R/W	93	Post_rgbbst_dlut7: same as Post_rgbbst_dlut0

Table 9-3180 VPP\_POST\_RGB\_DLUT\_8\_11 0x32ad

Bit(s)	R/W	Default	Description
31:24	R/W	85	Post_rgbbst_dlut8: same as Post_rgbbst_dlut0
23:16	R/W	78	Post_rgbbst_dlut9: same as Post_rgbbst_dlut0
15:8	R/W	73	Post_rgbbst_dlut10: same as Post_rgbbst_dlut0
7:0	R/W	68	Post_rgbbst_dlut11: same as Post_rgbbst_dlut0

Table 9-3181 VPP\_POST\_MATRIX\_SAT 0x32c1

Bit(s)	R/W	Default	Description
2	R/W	85	Sato_en
1	R/W	78	Sati_en
0	R/W	73	Misc_sat_en

### 9.2.3.46 3D LUT Registers

Table 9-3182 VPP\_LUT3D\_CTRL 0x39d0

Bit(s)	R/W	Default	Description
9:8	R/W	0	Gated clock control
7	R/W		reserved
6:4	R/W		reg_lut3d_extnd_en: enable to set LUT value of 1023 to value 1024.
3	R/W		reserved
2	R/W	0	1 to shaddow the "reg_lut3d_enable" by VSYNC
1	R/W		reserved
0	R/W	0	reg_lut3d_enable: 1 to enable 3D LUT

**Table 9-3183 VPP\_LUT3D\_CBUS2RAM\_CTRL 0x39d1**

Bit(s)	R/W	Default	Description
0	R/W	0	1 to enable CBUS to configure the LUT3D RAMs, 0 for in normal working status.

**Table 9-3184 VPP\_LUT3D\_RAM\_ADDR 0x39d2**

Bit(s)	R/W	Default	Description
32	R/W	0	LUT3D RAMs Address port

**Table 9-3185 VPP\_LUT3D\_RAM\_DATA 0x39d3**

Bit(s)	R/W	Default	Description
32	R/W	0	LUT3D RAMs data port

### 9.2.3.47 VPP GAMMA Registers

**Table 9-3186 VPP\_GAMMA\_CTRL 0x39d4**

Bit(s)	R/W	Default	Description
31:10	R/W	0	Reserved
9:7	R/W	0	Reg_soft_mode
6	R/W	0	Reg_enc_rst_mode
5	R/W	0	Reg_enc_en_mode
2:1	R/W	0	Reg_gclk_ctrl
0	R/W	0	Reg_gamma_en

VPP\_GAMMA\_BIN\_ADDR 0x39d5

VPP\_GAMMA\_BIN\_DATA 0x39d6

### 9.2.3.48 AFBCE Registers

**Table 9-3187 AFBCE\_ENABLE 0x41a0**

Bit(s)	R/W	Default	Description
31:20	R/W	0	gclk_ctrl : unsigned , default = 0, gclk_ctrl of afbe submodule
19:16	R/W	0	afbce_sync_sel : unsigned , default = 0, shadow aome size/scope/mode registers in afbce
13	R/W	0	enc_rst_mode : unsigned , default = 0, 1: chose pulse rst mode ,need write a pule by
12	R/W	0	enc_en_mode : unsigned , default = 0, 1: chose pulse staert mode, need write a pule by AFBCE_ENABLE[0],0:automatic start
8	R/W	0	enc_enable : unsigned , default = 0, 1: open mode 0:close module
0	R/W	0	Afbce start encod pulse ,active when AFBCE_ENABLE[12] high

**Table 9-3188 AFBCE\_MODE 0x41a1**

Bit(s)	R/W	Default	Description
31:29	R/W	0	soft_rst : unsigned, default = 0 ,bit[0] async rst [1] sync rst [2] sync rst by vsync
27:26	R/W	0	rev_mode : unsigned, default = 0 , reverse mode
25:24	R/W	3	mif_urgent : unsigned, default = 3 , info mif and data mif urgent
22:16	R/W	4	hold_line_num : unsigned, default = 4, hold_line_num
15:14	R/W	1	burst_mode : unsigned, default = 1, 0: burst1 1:burst2 2:burst4
0	R/W	0	reg_fmt444_comb : unsigned, default = 0, 0: 444 8bit comb mode

**Table 9-3189 AFBCE\_SIZE\_IN 0x41a2**

Bit(s)	R/W	Default	Description
28:16	R/W	1920	hsize_in : unsigned, default = 1920 , pic size in unit: pixel
12:0	R/W	1080	vsize_in : unsigned, default = 1080 , pic size in unit: pixel

**Table 9-3190 AFBCE\_BLK\_SIZE\_IN 0x41a3**

Bit(s)	R/W	Default	Description
28:16	R/W	60	hblk_size : unsigned, default = 60 , blk horz size out unit: 32*4 block
12:0	R/W	270	vblk_size : unsigned, default = 270, blk vert size out unit: 32*4 block

**Table 9-3191 AFBCE\_HEAD\_BADDR 0x41a4**

Bit(s)	R/W	Default	Description
31:0	R/W	0x00	head_baddr : unsigned, default = 32'h00,head_baddr

**Table 9-3192 AFBCE\_MIF\_SIZE 0x41a5**

Bit(s)	R/W	Default	Description
29:28	R/W	32	ddr_blk_size : unsigned, default = 32'd1,ddr_blk_size
26:24	R/W	32	cmd_blk_size : unsigned, default = 32'd3,cmd_blk_size
20:16	R/W	32	uncmp_size : unsigned, default = 32'd20 ,uncmp_size of a uncompressed 32*4 block
15:0	R/W	32	mmu_page_size : unsigned, default = 32'd4096,mmu_page_size 4096/8192



**Table 9-3193 AFBCE\_PIXEL\_IN\_HOR\_SCOPE 0x41a6**

Bit(s)	R/W	Default	Description
28:16	R/W	1919	enc_win_end_h : unsigned, default = 1919 ,pic scope in end,should be a integer multiple of 32 //
12:0	R/W	0	enc_win_bgn_h : unsigned, default = 0 ,pic scope in bgn,should be a integer multiple of 32 //

**Table 9-3194 AFBCE\_PIXEL\_IN\_VER\_SCOPE 0x41a7**

Bit(s)	R/W	Default	Description
28:16	R/W	1079	enc_win_end_v : unsigned, default = 1079 ,pic scope in end,should be a integer multiple of 4 //
12:0	R/W	0	enc_win_bgn_v : unsigned, default = 0 ,pic scope in bgn,should be a integer multiple of 4 //

**Table 9-3195 AFBCE\_CONV\_CTRL 0x41a8**

Bit(s)	R/W	Default	Description
28:16	R/W	2048	fmt_ybuf_depth : unsigned, default = 2048,fmt_ybuf_depth,fixed
11: 0	R/W	256	lbuf_depth : unsigned, default = 256, lbuf_depth,fixed

**Table 9-3196 AFBCE\_MIF\_HOR\_SCOPE 0x41a9**

Bit(s)	R/W	Default	Description
25:16	R/W	0	blk_end_h : unsigned, default = 0 ,blk scope out end,should be a integer multiple of 4 //
9:0	R/W	59	blk_bgn_h : unsigned, default = 59 ,blk scope out bgn,should be a integer multiple of 4 //

**Table 9-3197 AFBCE\_MIF\_VER\_SCOPE 0x41aa**

Bit(s)	R/W	Default	Description
27:16	R/W	0	blk_end_v : unsigned, default = 0 ,blk scope out end,should be a integer multiple of 4 //
11:0	R/W	269	blk_bgn_v : unsigned, default = 269 ,blk scope out bgn,should be a integer multiple of 4 //

**Table 9-3198 AFBCE\_STAT1 0x41ab**

Bit(s)	R/W	Default	Description
31	R.O	0	ro_frm_end_pulse1 : unsigned, RO,default = 0 ;frame end status
30:0	R.O	0	ro_dbg_top_info1 : unsigned, RO,default = 0 ,ro_dbg_top_info1

**Table 9-3199 AFBCE\_STAT2 0x410c**

Bit(s)	R/W	Default	Description
30:0	R.O	0	ro_dbg_top_info2 : unsigned, RO,default = 0 ,ro_dbg_top_info2

**Table 9-3200 AFBCE\_FORMAT 0x41ad**

Bit(s)	R/W	Default	Description
9: 8	R/W	2	reg_format_mode : // unsigned , RW, default = 2 data format;0 : YUV444, 1: YUV422, 2:YUV420, 3:RGB
7: 4	R/W	10	reg_compbits_c : // unsigned , RW, default = 10 chroma bitwidth 8 or 10
3: 0	R/W	10	reg_compbits_y : // unsigned , RW, default = 10 luma bitwidth 8 or 10

**Table 9-3201 AFBCE\_MODE\_EN 0x41ae**

Bit(s)	R/W	Default	Description
25	R/W	0	reg_adpt_interleave_ymode : // unsigned , RW, default = 0 force 0 to disable it: no HW implementation
24	R/W	0	reg_adpt_interleave_cmode : // unsigned , RW, default = 0 force 0 to disable it: not HW implementation
23	R/W	1	reg_adpt_yinterleave_luma_ride : // unsigned , RW, default = 1 vertical interleave piece luma reorder ride; 0: no reorder ride; 1: w/4 as ride
22	R/W	1	reg_adpt_yinterleave_chrm_ride : // unsigned , RW, default = 1 vertical interleave piece chroma reorder ride; 0: no reorder ride; 1: w/2 as ride
21	R/W	1	reg_adpt_xinterleave_luma_ride : // unsigned , RW, default = 1 vertical interleave piece luma reorder ride; 0: no reorder ride; 1: w/4 as ride
20	R/W	1	reg_adpt_xinterleave_chrm_ride : // unsigned , RW, default = 1 vertical interleave piece chroma reorder ride; 0: no reorder ride; 1: w/2 as ride
18	R/W	0	reg_disable_order_mode_i_6 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
17	R/W	0	reg_disable_order_mode_i_5 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
16	R/W	0	reg_disable_order_mode_i_4 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
15	R/W	0	reg_disable_order_mode_i_3 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
14	R/W	0	reg_disable_order_mode_i_2 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
13	R/W	0	reg_disable_order_mode_i_1 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
12	R/W	0	reg_disable_order_mode_i_0 : // unsigned , RW, default = 0 disable order mode0~6: each mode with one disable bit: 0: no disable, 1: disable
10	R/W	0	reg_minval_yenc_en : // unsigned , RW, default = 0 force disable, final decision to remove this ws 1% performance loss
9	R/W	0	reg_16x4block_enable : // unsigned , RW, default = 0 block as mission, but permit 16x4 block
8	R/W	0	reg_uncompress_split_mode : // unsigned , RW, default = 0 0: no split; 1: split

Bit(s)	R/W	Default	Description
5	R/W	0	reg_input_padding_uv128 : // unsigned , RW, default = 0 input picture 32x4 block gap mode: 0: pad uv=0; 1: pad uv=128
4	R/W	0	reg_dwds_padding_uv128 : // unsigned , RW, default = 0 downsampled image for double write 32x_gap mode: 0: pad uv=0; 1: pad uv=128
3: 1	R/W	0	reg_force_order_mode_value : // unsigned , RW, default = 0 force order mode 0~7
0	R/W	0	reg_force_order_mode_en : // unsigned , RW, default = 0 force order mode enable: 0: no force; 1: forced to force_value

Table 9-3202 AFBCE\_DWSCALAR 0x41af

Bit(s)	R/W	Default	Description
7: 6	R/W	3	reg_dwscalar_w0 : // unsigned , RW, default = 3 horizontal 1st step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg
5: 4	R/W	0	reg_dwscalar_w1 : // unsigned , RW, default = 0 horizontal 2nd step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg
3: 2	R/W	2	reg_dwscalar_h0 : // unsigned , RW, default = 2 vertical 1st step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg
1: 0	R/W	3	reg_dwscalar_h1 : // unsigned , RW, default = 3 vertical 2nd step scalar mode: 0: 1:1 no scalar; 1: 2:1 data drop (0,2,4, 6) pixel kept; 2: 2:1 data drop (1, 3, 5,7..) pixels kept; 3: avg

Table 9-3203 AFBCE\_DEFCOLOR\_1 0x41b0

Bit(s)	R/W	Default	Description
23:12	R/W	4095	reg_enc_defalutcolor_3 : // unsigned , RW, default = 4095 Picture wise default color value in [Y Cb Cr]
11: 0	R/W	4095	reg_enc_defalutcolor_0 : // unsigned , RW, default = 4095 Picture wise default color value in [Y Cb Cr]

Table 9-3204 AFBCE\_DEFCOLOR\_2 0x41b1

Bit(s)	R/W	Default	Description
23:12	R/W	2048	reg_enc_defalutcolor_2 : // unsigned , RW, default = 2048 wise default color value in [Y Cb Cr]
11: 0	R/W	2048	reg_enc_defalutcolor_1 : // unsigned , RW, default = 2048 wise default color value in [Y Cb Cr]

**Table 9-3205 AFBCE\_QUANT\_ENABLE 0x41b2**

Bit(s)	R/W	Default	Description
11	R/W	0	reg_quant_expand_en_1 : // unsigned , RW, default = 0 enable for quantization value expansion
10	R/W	0	reg_quant_expand_en_0 : // unsigned , RW, default = 0 enable for quantization value expansion
9: 8	R/W	0	reg_bcleav_ofst : // signed , RW, default = 0 bcleave ofset to get lower range, especially under lossy, for v1/v2, x=0 is equivalent, default = -1;
4	R/W	0	reg_quant_enable_1 : // unsigned , RW, default = 0 enable for quant to get some lossy
0	R/W	0	reg_quant_enable_0 : // unsigned , RW, default = 0 enable for quant to get some lossy

**Table 9-3206 AFBCE\_IQUANT\_LUT\_1 0x41b3**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_iquant_yclut_0_11 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
26:24	R/W	1	reg_iquant_yclut_0_10 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
22:20	R/W	2	reg_iquant_yclut_0_9 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
18:16	R/W	3	reg_iquant_yclut_0_8 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
14:12	R/W	4	reg_iquant_yclut_0_7 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	5	reg_iquant_yclut_0_6 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	5	reg_iquant_yclut_0_5 : // unsigned , RW, default = 5 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	4	reg_iquant_yclut_0_4 : // unsigned , RW, default = 4 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

**Table 9-3207 AFBCE\_IQUANT\_LUT\_2 0x41b4**

Bit(s)	R/W	Default	Description
14:12	R/W	3	reg_iquant_yclut_0_3 : // unsigned , RW, default = 3 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
10: 8	R/W	2	reg_iquant_yclut_0_2 : // unsigned , RW, default = 2 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
6: 4	R/W	1	reg_iquant_yclut_0_1 : // unsigned , RW, default = 1 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)
2: 0	R/W	0	reg_iquant_yclut_0_0 : // unsigned , RW, default = 0 quantization lut for mintree leavs, iquant=2^lut(bc_leav_q+1)

**Table 9-3208 AFBCE\_IQUANT\_LUT\_3 0x41b5**

Bit(s)	R/W	Default	Description
30:28	R/W	0	reg_iquant_yclut_1_11 : // unsigned , RW, default = 0 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
26:24	R/W	1	reg_iquant_yclut_1_10 : // unsigned , RW, default = 1 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
22:20	R/W	2	reg_iquant_yclut_1_9 : // unsigned , RW, default = 2 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
18:16	R/W	3	reg_iquant_yclut_1_8 : // unsigned , RW, default = 3 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
14:12	R/W	4	reg_iquant_yclut_1_7 : // unsigned , RW, default = 4 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
10: 8	R/W	5	reg_iquant_yclut_1_6 : // unsigned , RW, default = 5 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
6: 4	R/W	5	reg_iquant_yclut_1_5 : // unsigned , RW, default = 5 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
2: 0	R/W	4	reg_iquant_yclut_1_4 : // unsigned , RW, default = 4 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$

**Table 9-3209 AFBCE\_IQUANT\_LUT\_4 0x41b6**

Bit(s)	R/W	Default	Description
14:12	R/W	3	reg_iquant_yclut_1_3 : // unsigned , RW, default = 3 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
10: 8	R/W	2	reg_iquant_yclut_1_2 : // unsigned , RW, default = 2 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
6: 4	R/W	1	reg_iquant_yclut_1_1 : // unsigned , RW, default = 1 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$
2: 0	R/W	0	reg_iquant_yclut_1_0 : // unsigned , RW, default = 0 quantization lut for mintree leavs, $iquant=2^{lut}(bc\_leav\_q+1)$

**Table 9-3210 AFBCE\_RQUANT\_LUT\_1 0x41b7**

Bit(s)	R/W	Default	Description
30:28	R/W	5	reg_rquant_yclut_0_11 : // unsigned , RW, default = 5 quantization lut for bctree leavs, $quant=2^{lut}(bc\_leav\_r+1)$ , can be calculated from $iquant\_yclut(fw\_setting)$
26:24	R/W	5	reg_rquant_yclut_0_10 : // unsigned , RW, default = 5
22:20	R/W	4	reg_rquant_yclut_0_9 : // unsigned , RW, default = 4
18:16	R/W	4	reg_rquant_yclut_0_8 : // unsigned , RW, default = 4
14:12	R/W	3	reg_rquant_yclut_0_7 : // unsigned , RW, default = 3
10: 8	R/W	3	reg_rquant_yclut_0_6 : // unsigned , RW, default = 3
6: 4	R/W	2	reg_rquant_yclut_0_5 : // unsigned , RW, default = 2
2: 0	R/W	2	reg_rquant_yclut_0_4 : // unsigned , RW, default = 2

**Table 9-3211 AFBCE\_RQUANT\_LUT\_2 0x41b8**

Bit(s)	R/W	Default	Description
14:12	R/W	1	reg_rquant_yclut_0_3 : // unsigned , RW, default = 1
10: 8	R/W	1	reg_rquant_yclut_0_2 : // unsigned , RW, default = 1
6: 4	R/W	0	reg_rquant_yclut_0_1 : // unsigned , RW, default = 0
2: 0	R/W	0	reg_rquant_yclut_0_0 : // unsigned , RW, default = 0

**Table 9-3212 AFBCE\_RQUANT\_LUT\_3 0x41b9**

Bit(s)	R/W	Default	Description
30:28	R/W	5	reg_rquant_yclut_1_11 : // unsigned , RW, default = 5 quantization lut for bctree leavs, quant=2^lut(bc_leav_r+1), can be calculated from iquant_yclut(fw_setting)
26:24	R/W	5	reg_rquant_yclut_1_10 : // unsigned , RW, default = 5
22:20	R/W	4	reg_rquant_yclut_1_9 : // unsigned , RW, default = 4
18:16	R/W	4	reg_rquant_yclut_1_8 : // unsigned , RW, default = 4
14:12	R/W	3	reg_rquant_yclut_1_7 : // unsigned , RW, default = 3
10: 8	R/W	3	reg_rquant_yclut_1_6 : // unsigned , RW, default = 3
6: 4	R/W	2	reg_rquant_yclut_1_5 : // unsigned , RW, default = 2
2: 0	R/W	2	reg_rquant_yclut_1_4 : // unsigned , RW, default = 2

**Table 9-3213 AFBCE\_RQUANT\_LUT\_4 0x41ba**

Bit(s)	R/W	Default	Description
14:12	R/W	1	reg_rquant_yclut_1_3 : // unsigned , RW, default = 1
10: 8	R/W	1	reg_rquant_yclut_1_2 : // unsigned , RW, default = 1
6: 4	R/W	0	reg_rquant_yclut_1_1 : // unsigned , RW, default = 0
2: 0	R/W	0	reg_rquant_yclut_1_0 : // unsigned , RW, default = 0

**Table 9-3214 AFBCE\_YUV\_FORMAT\_CONV\_MODE 0x41bb**

Bit(s)	R/W	Default	Description
6: 4	R/W	0	reg_444to422_mode : // unsigned , RW, default = 0
2: 0	R/W	0	reg_422to420_mode : // unsigned , RW, default = 0

**Table 9-3215 AFBCE\_DUMMY\_DATA 0x41bc**

Bit(s)	R/W	Default	Description
29: 0	R/W	0	reg_dummy_data : // unsigned , default = 0 ,bit[9:0] v bit[19:10] U bit[29:20] Y

**Table 9-3216 AFBCE\_CLR\_FLAG 0x41bd**

Bit(s)	R/W	Default	Description
1	R/W	0	enc_error_clr : // unsigned, default = 0 ;
0	R/W	0	frm_end_clr : // unsigned, default = 0 ;

**Table 9-3217 AFBCE\_STA\_FLAGT 0x41be**

Bit(s)	R/W	Default	Description
31:26	R.O	0	reverse : // unsigned, RO,default = 0 ;
25:24	R.O	0	enc_st 0:idle 1:enc 2:hold 3:wait
23:21	R.O	0	emit_st 0:idle 1:tx_bcrot 2:tx_bc_lev 3:tx_zcd 4:tx_tree 5:tx_uncmp 6:tx_redu 7:tx_check
20:15	R.O	0	ro_cmd2bresp_cnt: // unsigned, RO,default = 0 ;
14:9	R.O	0	ro_enc2body_cnt: // unsigned, RO,default = 0 ;
8:4	R.O	0	ro_cmd2wd_cnt: // unsigned, RO,default = 0 ;
3:2	R.O	0	ro_wr_abort_flag: // unsigned, RO,default = 0 ;
1	R.O	0	ro_enc_error_flag: // unsigned, RO,default = 0 ;
0	R.O	0	ro_frm_end_flag: // unsigned, RO,default = 0 ;

**Table 9-3218 AFBCE\_MMU\_NUM 0x41bf**

Bit(s)	R/W	Default	Description
15: 0	R.O	0	ro_frm_mmu_num : // unsigned, RO,default = 0 ,mmu addr have been used in a frame

**Table 9-3219 AFBCE\_MMU\_RMIF\_CTRL1 0x41c0**

Bit(s)	R/W	Default	Description
25:24	R/W	0	reg_sync_sel : // unsigned , default = 0, axi canvas id sync with frm rst
23:16	R/W	0	reg_canvas_id : // unsigned , default = 0, axi canvas id num
14:12	R/W	1	reg_cmd_intr_len : // unsigned , default = 1, interrupt send cmd when how many series axi cmd, 0=12 1=16 2=24 3=32 4=40 5=48 6=56 7=64
11:10	R/W	1	reg_cmd_req_size : // unsigned , default = 1, how many room fifo have, then axi send series req, 0=16 1=32 2=24 3=64
9:8	R/W	2	reg_burst_len : // unsigned , default = 2, burst type: 0-single 1-bst2 2-bst4

Bit(s)	R/W	Default	Description
7	R/W	0	reg_swap_64bit : // unsigned , default = 0, 64bits of 128bit swap enable
6	R/W	0	reg_little_endian : // unsigned , default = 0, big endian enable
5	R/W	0	reg_y_rev : // unsigned , default = 0, vertical reverse enable
4	R/W	0	reg_x_rev : // unsigned , default = 0, horizontal reverse enable
2:0	R/W	3	reg_pack_mode : // unsigned , default = 3, 0:4bit 1:8bit 2:16bit 3:32bit 4:64bit 5:128bit

Table 9-3220 AFBCE\_MMU\_RMIF\_CTRL2 0x41c1

Bit(s)	R/W	Default	Description
31:30	R/W	0	reg_sw_rst : // unsigned , default = 0,
23:18	R/W	0x0	reg_gclk_ctrl :
16:0	R/W	0	reg_urgent_ctrl : // unsigned , default = 0, urgent control reg : 16 reg_ugt_init : urgent initial value 15 reg_ugt_en : urgent enable 14 reg_ugt_type : 1= wrmif 0=rdmif 7 :4 reg_ugt_top_th: urgent top threshold 3 :0 reg_ugt_bot_th: urgent bottom threshold

Table 9-3221 AFBCE\_MMU\_RMIF\_CTRL3 0x41c2

Bit(s)	R/W	Default	Description
16	R/W	1	reg_acc_mode : // unsigned , default = 1,
12:0	R/W	4096	reg_stride : // unsigned , default = 4096,

Table 9-3222 AFBCE\_MMU\_RMIF\_CTRL4 0x41c3

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_baddr : // unsigned , default = 0, reg_mmu_baddr

Table 9-3223 AFBCE\_MMU\_RMIF\_SCOPE\_X 0x41c4

Bit(s)	R/W	Default	Description
28:16	R/W	4095	reg_x_end : // unsigned , default = 4095, the canvas hor end pixel position
12: 0	R/W	0	reg_x_start : // unsigned , default = 0, the canvas hor start pixel position

Table 9-3224 AFBCE\_MMU\_RMIF\_SCOPE\_Y 0x41c5

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_y_end : // unsigned , default = 0, the canvas ver end pixel position
12: 0	R/W	0	reg_y_start : // unsigned , default = 0, the canvas ver start pixel position



**Table 9-3225 AFBCE\_MMU\_RMIF\_RO\_STAT 0x41c6**

Bit(s)	R/W	Default	Description
15:0	R/W	0x0	reg_status : // unsigned , reg_status

**9.2.3.49 VPP\_PIP\_ALPH Registers****Table 9-3226 VD1\_PIP\_ALPH\_CTRL 0x5880**

Bit(s)	R/W	Default	Description
28	R/W	0	reg_alph_gen_dbg: //unsigned default =0 ,dbg_mode: output pic with alph_256 to check function
27:12	R/W	0	reg_pic_en: //unsigned default=0, 16bits decide which pics are enabled
11:3	R/W	0	reg_alph0: //unsigned default=0, alph value of those points which are out of windows' ranges
2:1	R/W	0	reg_alph_gen_mode ://unsigned, default = 0, from out of window to border to inside :0:original ,1: alph change as 0/0.5/1 2: alph_change as 0/0.25/0.5/0.75/1
0	R/W	1	reg_alph_gen_bypas: // unsigned, default = 1 bypass mode

**Table 9-3227 VD1\_PIP\_ALPH\_SCP\_H\_0 0x5881**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h0: //unsigned default=0, 13bits pic_0 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h0: //unsigned default=0, 13bits pic_0 window horizontal begin

**Table 9-3228 VD1\_PIP\_ALPH\_SCP\_H\_1 0x5882**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h1: //unsigned default=0, 13bits pic_1 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h1: //unsigned default=0, 13bits pic_1 window horizontal begin

**Table 9-3229 VD1\_PIP\_ALPH\_SCP\_H\_2 0x5883**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h2: //unsigned default=0, 13bits pic_2 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h2: //unsigned default=0, 13bits pic_2 window horizontal begin

**Table 9-3230 VD1\_PIP\_ALPH\_SCP\_H\_3 0x5884**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h3: //unsigned default=0, 13bits pic_3 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h3: //unsigned default=0, 13bits pic_3 window horizontal begin

**Table 9-3231 VD1\_PIP\_ALPH\_SCP\_H\_4 0x5885**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h4: //unsigned default=0, 13bits pic_4 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h4: //unsigned default=0, 13bits pic_4 window horizontal begin

**Table 9-3232 VD1\_PIP\_ALPH\_SCP\_H\_5 0x5886**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h5: //unsigned default=0, 13bits pic_5 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h5: //unsigned default=0, 13bits pic_5 window horizontal begin

**Table 9-3233 VD1\_PIP\_ALPH\_SCP\_H\_6 0x5887**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h6: //unsigned default=0, 13bits pic_6 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h6: //unsigned default=0, 13bits pic_6 window horizontal begin

**Table 9-3234 VD1\_PIP\_ALPH\_SCP\_H\_7 0x5888**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h7: //unsigned default=0, 13bits pic_7 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h7: //unsigned default=0, 13bits pic_7 window horizontal begin

**Table 9-3235 VD1\_PIP\_ALPH\_SCP\_H\_8 0x5889**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h8: //unsigned default=0, 13bits pic_8 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h8: //unsigned default=0, 13bits pic_8 window horizontal begin

**Table 9-3236 VD1\_PIP\_ALPH\_SCP\_H\_9 0x588a**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h9: //unsigned default=0, 13bits pic_9 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h9: //unsigned default=0, 13bits pic_9 window horizontal begin

**Table 9-3237 VD1\_PIP\_ALPH\_SCP\_H\_10 0x588b**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h10: //unsigned default=0, 13bits pic_10 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h10: //unsigned default=0, 13bits pic_10 window horizontal begin

**Table 9-3238 VD1\_PIP\_ALPH\_SCP\_H\_11 0x588c**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h11: //unsigned default=0, 13bits pic_11 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h11: //unsigned default=0, 13bits pic_11 window horizontal begin

**Table 9-3239 VD1\_PIP\_ALPH\_SCP\_H\_12 0x588d**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h12: //unsigned default=0, 13bits pic_12 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h12: //unsigned default=0, 13bits pic_12 window horizontal begin

**Table 9-3240 VD1\_PIP\_ALPH\_SCP\_H\_13 0x588e**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h13: //unsigned default=0, 13bits pic_13 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h13: //unsigned default=0, 13bits pic_13 window horizontal begin

**Table 9-3241 VD1\_PIP\_ALPH\_SCP\_H\_14 0x588f**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h14: //unsigned default=0, 13bits pic_14 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h14: //unsigned default=0, 13bits pic_14 window horizontal begin

**Table 9-3242 VD1\_PIP\_ALPH\_SCP\_H\_15 0x5890**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h15: //unsigned default=0, 13bits pic_15 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h15: //unsigned default=0, 13bits pic_15 window horizontal begin

**Table 9-3243 VD1\_PIP\_ALPH\_SCP\_V\_0 0x5891**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v0: //unsigned default=0, 13bits pic_0 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v0: //unsigned default=0, 13bits pic_0 window vertical begin

**Table 9-3244 VD1\_PIP\_ALPH\_SCP\_V\_1 0x5892**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v1: //unsigned default=0, 13bits pic_1 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v1: //unsigned default=0, 13bits pic_1 window vertical begin

**Table 9-3245 VD1\_PIP\_ALPH\_SCP\_V\_2 0x5893**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v2: //unsigned default=0, 13bits pic_2 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v2: //unsigned default=0, 13bits pic_2 window vertical begin

**Table 9-3246 VD1\_PIP\_ALPH\_SCP\_V\_3 0x5894**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v3: //unsigned default=0, 13bits pic_3 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v3: //unsigned default=0, 13bits pic_3 window vertical begin

**Table 9-3247 VD1\_PIP\_ALPH\_SCP\_V\_4 0x5895**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v4: //unsigned default=0, 13bits pic_4 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v4: //unsigned default=0, 13bits pic_4 window vertical begin

**Table 9-3248 VD1\_PIP\_ALPH\_SCP\_V\_5 0x5896**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v5: //unsigned default=0, 13bits pic_5 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v5: //unsigned default=0, 13bits pic_5 window vertical begin

**Table 9-3249 VD1\_PIP\_ALPH\_SCP\_V\_6 0x5897**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v6: //unsigned default=0, 13bits pic_6 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v6: //unsigned default=0, 13bits pic_6 window vertical begin

**Table 9-3250 VD1\_PIP\_ALPH\_SCP\_V\_7 0x5898**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v7: //unsigned default=0, 13bits pic_7 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v7: //unsigned default=0, 13bits pic_7 window vertical begin

**Table 9-3251 VD1\_PIP\_ALPH\_SCP\_V\_8 0x5899**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v8: //unsigned default=0, 13bits pic_8 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v8: //unsigned default=0, 13bits pic_8 window vertical begin

**Table 9-3252 VD1\_PIP\_ALPH\_SCP\_V\_9 0x589a**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v9: //unsigned default=0, 13bits pic_9 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v9: //unsigned default=0, 13bits pic_9 window vertical begin

**Table 9-3253 VD1\_PIP\_ALPH\_SCP\_V\_10 0x589b**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v10: //unsigned default=0, 13bits pic_10 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v10: //unsigned default=0, 13bits pic_10 window vertical begin

**Table 9-3254 VD1\_PIP\_ALPH\_SCP\_V\_11 0x589c**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v11: //unsigned default=0, 13bits pic_11 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v11: //unsigned default=0, 13bits pic_11 window vertical begin

**Table 9-3255 VD1\_PIP\_ALPH\_SCP\_V\_12 0x589d**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v12: //unsigned default=0, 13bits pic_12 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v12: //unsigned default=0, 13bits pic_12 window vertical begin

**Table 9-3256 VD1\_PIP\_ALPH\_SCP\_V\_13 0x589e**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v13: //unsigned default=0, 13bits pic_13 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v13: //unsigned default=0, 13bits pic_13 window vertical begin

**Table 9-3257 VD1\_PIP\_ALPH\_SCP\_V\_14 0x589f**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v14: //unsigned default=0, 13bits pic_14 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v14: //unsigned default=0, 13bits pic_14 window vertical begin

**Table 9-3258 VD1\_PIP\_ALPH\_SCP\_V\_15 0x58a0**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v15: //unsigned default=0, 13bits pic_15 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v15: //unsigned default=0, 13bits pic_15 window vertical begin

**Table 9-3259 VD2\_PIP\_ALPH\_CTRL 0x58b0**

Bit(s)	R/W	Default	Description
28	R/W	0	reg_alph_gen_dbg: //unsigned default =0 ,dbg_mode: output pic with alph_256 to check function
27:12	R/W	0	reg_pic_en: //unsigned default=0, 16bits decide which pics are enabled
11:3	R/W	0	reg_alph0: //unsigned default=0, alph value of those points which are out of windows' ranges
2:1	R/W	0	reg_alph_gen_mode ://unsigned, default = 0, from out of window to border to inside :0:original ,1: alph change as 0/0.5/1 2: alph_change as 0/0.25/0.5/0.75/1
0	R/W	1	reg_alph_gen_byps: // unsigned, default = 1 bypass mode

**Table 9-3260 VD2\_PIP\_ALPH\_SCP\_H\_0 0x58b1**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h0: //unsigned default=0, 13bits pic_0 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h0: //unsigned default=0, 13bits pic_0 window horizontal begin

**Table 9-3261 VD2\_PIP\_ALPH\_SCP\_H\_1 0x58b2**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h1: //unsigned default=0, 13bits pic_1 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h1: //unsigned default=0, 13bits pic_1 window horizontal begin

**Table 9-3262 VD2\_PIP\_ALPH\_SCP\_H\_2 0x58b3**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h2: //unsigned default=0, 13bits pic_2 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h2: //unsigned default=0, 13bits pic_2 window horizontal begin

**Table 9-3263 VD2\_PIP\_ALPH\_SCP\_H\_3 0x58b4**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h3: //unsigned default=0, 13bits pic_3 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h3: //unsigned default=0, 13bits pic_3 window horizontal begin

**Table 9-3264 VD2\_PIP\_ALPH\_SCP\_H\_4 0x58b5**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h4: //unsigned default=0, 13bits pic_4 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h4: //unsigned default=0, 13bits pic_4 window horizontal begin

**Table 9-3265 VD2\_PIP\_ALPH\_SCP\_H\_5 0x58b6**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h5: //unsigned default=0, 13bits pic_5 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h5: //unsigned default=0, 13bits pic_5 window horizontal begin

**Table 9-3266 VD2\_PIP\_ALPH\_SCP\_H\_6 0x58b7**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h6: //unsigned default=0, 13bits pic_6 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h6: //unsigned default=0, 13bits pic_6 window horizontal begin

**Table 9-3267 VD2\_PIP\_ALPH\_SCP\_H\_7 0x58b8**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h7: //unsigned default=0, 13bits pic_7 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h7: //unsigned default=0, 13bits pic_7 window horizontal begin

**Table 9-3268 VD2\_PIP\_ALPH\_SCP\_H\_8 0x58b9**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h8: //unsigned default=0, 13bits pic_8 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h8: //unsigned default=0, 13bits pic_8 window horizontal begin

**Table 9-3269 VD2\_PIP\_ALPH\_SCP\_H\_9 0x58ba**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h9: //unsigned default=0, 13bits pic_9 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h9: //unsigned default=0, 13bits pic_9 window horizontal begin

**Table 9-3270 VD2\_PIP\_ALPH\_SCP\_H\_10 0x58bb**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h10: //unsigned default=0, 13bits pic_10 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h10: //unsigned default=0, 13bits pic_10 window horizontal begin

**Table 9-3271 VD2\_PIP\_ALPH\_SCP\_H\_11 0x58bc**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h11: //unsigned default=0, 13bits pic_11 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h11: //unsigned default=0, 13bits pic_11 window horizontal begin

**Table 9-3272 VD2\_PIP\_ALPH\_SCP\_H\_12 0x58bd**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h12: //unsigned default=0, 13bits pic_12 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h12: //unsigned default=0, 13bits pic_12 window horizontal begin

**Table 9-3273 VD2\_PIP\_ALPH\_SCP\_H\_13 0x58be**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h13: //unsigned default=0, 13bits pic_13 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h13: //unsigned default=0, 13bits pic_13 window horizontal begin

**Table 9-3274 VD2\_PIP\_ALPH\_SCP\_H\_14 0x58bf**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h14: //unsigned default=0, 13bits pic_14 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h14: //unsigned default=0, 13bits pic_14 window horizontal begin

**Table 9-3275 VD2\_PIP\_ALPH\_SCP\_H\_15 0x58c0**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_h15: //unsigned default=0, 13bits pic_15 window horizontal end
12:0	R/W	0	reg_scpxn_bgn_h15: //unsigned default=0, 13bits pic_15 window horizontal begin

**Table 9-3276 VD2\_PIP\_ALPH\_SCP\_V\_0 0x58c1**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v0: //unsigned default=0, 13bits pic_0 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v0: //unsigned default=0, 13bits pic_0 window vertical begin



**Table 9-3277 VD2\_PIP\_ALPH\_SCP\_V\_1 0x58c2**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v1: //unsigned default=0, 13bits pic_1 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v1: //unsigned default=0, 13bits pic_1 window vertical begin

**Table 9-3278 VD2\_PIP\_ALPH\_SCP\_V\_2 0x58c3**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v2: //unsigned default=0, 13bits pic_2 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v2: //unsigned default=0, 13bits pic_2 window vertical begin

**Table 9-3279 VD2\_PIP\_ALPH\_SCP\_V\_3 0x58c4**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v3: //unsigned default=0, 13bits pic_3 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v3: //unsigned default=0, 13bits pic_3 window vertical begin

**Table 9-3280 VD2\_PIP\_ALPH\_SCP\_V\_4 0x58c5**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v4: //unsigned default=0, 13bits pic_4 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v4: //unsigned default=0, 13bits pic_4 window vertical begin

**Table 9-3281 VD2\_PIP\_ALPH\_SCP\_V\_5 0x58c6**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v5: //unsigned default=0, 13bits pic_5 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v5: //unsigned default=0, 13bits pic_5 window vertical begin

**Table 9-3282 VD2\_PIP\_ALPH\_SCP\_V\_6 0x58c7**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v6: //unsigned default=0, 13bits pic_6 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v6: //unsigned default=0, 13bits pic_6 window vertical begin

**Table 9-3283 VD2\_PIP\_ALPH\_SCP\_V\_7 0x58c8**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v7: //unsigned default=0, 13bits pic_7 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v7: //unsigned default=0, 13bits pic_7 window vertical begin

**Table 9-3284 VD2\_PIP\_ALPH\_SCP\_V\_8 0x58c9**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v8: //unsigned default=0, 13bits pic_8 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v8: //unsigned default=0, 13bits pic_8 window vertical begin

**Table 9-3285 VD2\_PIP\_ALPH\_SCP\_V\_9 0x58ca**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v9: //unsigned default=0, 13bits pic_9 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v9: //unsigned default=0, 13bits pic_9 window vertical begin

**Table 9-3286 VD2\_PIP\_ALPH\_SCP\_V\_10 0x58cb**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v10: //unsigned default=0, 13bits pic_10 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v10: //unsigned default=0, 13bits pic_10 window vertical begin

**Table 9-3287 VD2\_PIP\_ALPH\_SCP\_V\_11 0x58cc**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v11: //unsigned default=0, 13bits pic_11 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v11: //unsigned default=0, 13bits pic_11 window vertical begin

**Table 9-3288 VD2\_PIP\_ALPH\_SCP\_V\_12 0x58cd**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v12: //unsigned default=0, 13bits pic_12 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v12: //unsigned default=0, 13bits pic_12 window vertical begin

**Table 9-3289 VD2\_PIP\_ALPH\_SCP\_V\_13 0x58ce**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v13: //unsigned default=0, 13bits pic_13 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v13: //unsigned default=0, 13bits pic_13 window vertical begin

**Table 9-3290 VD2\_PIP\_ALPH\_SCP\_V\_14 0x58cf**

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v14: //unsigned default=0, 13bits pic_14 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v14: //unsigned default=0, 13bits pic_14 window vertical begin

Table 9-3291 VD2\_PIP\_ALPH\_SCP\_V\_15 0x58d0

Bit(s)	R/W	Default	Description
28:16	R/W	0	reg_scpxn_end_v15: //unsigned default=0, 13bits pic_15 window vertical end
12:0	R/W	0	reg_scpxn_bgn_v15: //unsigned default=0, 13bits pic_15 window vertical begin

## 9.2.3.50 HDMITX Registers

### 9.2.3.50.1 Register Access

Accessing HDMITX Top-Level registers is by directly accessing memory addresses. The top-level registers are 4-byte wide, below list the correct ways to access Top-level registers.

#### Top-level Registers Access

```
#define HDMITX_TOP_OFFSET0xfe300000

void hdmix_wr_only_TOP (uint8_t int_ext, uint32_t addr, uint32_t data)
{
    *(volatile uint32_t*)( HDMITX_TOP_OFFSET +addr) = (data);
} /* hdmix_wr_only_TOP */

uint32_t hdmix_rd_TOP (uint8_t int_ext, uint32_t addr)
{
    uint32_t data;
    data = *(volatile uint32_t*)( HDMITX_TOP_OFFSET +addr);
    return (data);
} /* hdmix_rd_TOP */
```

### 9.2.3.50.2 Top-Level Register Description

#### Register Address

For below registers the base address is 0xfe380000.

Each register final address = BASE + address \* 4.

The following lists describe the mapping between each register and its address.

Address	Register	RW	Function
0x000<<2	HDMITX_TOP_SW_RESET	RW	Software reset sub-modules.
0x001<<2	HDMITX_TOP_CLK_CNTL	RW	Clock gating and inversion.
0x002<<2	HDMITX_TOP_HPD_FILTER	RW	HPD and RxSense input glitch filter.
0x003<<2	HDMITX_TOP_INTR_MASKN	RW	Interrupt mask.
0x004<<2	HDMITX_TOP_INTR_STAT	RW	Interrupt status.
0x005<<2	HDMITX_TOP_INTR_STAT_CLR	W	Interrupt clear.
0x006<<2	HDMITX_TOP_BIST_CNTL	RW	Build-In Self Test(BIST) control.
0x007<<2	HDMITX_TOP_SHIFT_PTTN_012	RW	Shift pattern for BIST.
0x008<<2	HDMITX_TOP_SHIFT_PTTN_345	RW	Shift pattern for BIST.

Address	Register	RW	Function
0x009<<2	HDMITX_TOP_SHIFT_PTTN_67	RW	Shift pattern for BIST.
0x00A<<2	HDMITX_TOP_TMDS_CLK_PTTN_01	RW	TMDS clock pattern for generating /10 or /40 rate clock.
0x00B<<2	HDMITX_TOP_TMDS_CLK_PTTN_23	RW	TMDS clock pattern for generating /10 or /40 rate clock.
0x00C<<2	HDMITX_TOP_TMDS_CLK_PTTN_CNTL	RW	TMDS clock pattern for generating /10 or /40 rate clock.
0x00E<<2	HDMITX_TOP_STAT0	RW	Status.
0x01D<<2	HDMITX_TOP_INFILTER	RW	DDC input glitch filter control.
0x01E<<2	HDMITX_TOP_NSEC_SCRATCH	RW	Scratch register for non-secure access.
0x01F<<2	HDMITX_TOP_SEC_SCRATCH	RW	Scratch register for secure access.
0x021<<2	HDMITX_TOP_HS_INTR_CNTL	RW	HS_INTR pulse control
0x022<<2	HDMITX_TOP_AUD_ID	RW	Audio input ID
0x023<<2	HDMITX_TOP_HV_ACTIVE	RW	Video size for secure video
0x028<<2	HDMITX_TOP_I2C_BUSY_CNT_MAX	RW	Max I2C idle time after I2C Start.
0x029<<2	HDMITX_TOP_I2C_BUSY_CNT_STAT	RW	I2C idle time status.
0x02B<<2	HDMITX_TOP_DDC_CNTL	RW	DDC pull down by SW.
0x02C<<2	HDMITX_TOP_FIX_DISABLE_0	RW	Disable IP internal sync crossing bug fix.
0x02D<<2	HDMITX_TOP_FIX_DISABLE_1	RW	Disable IP internal sync crossing bug fix.
0x033<<2	HDMITX_TOP_SEC_VIDEO_OVR	RW	Secure video protect.
0x034<<2	HDMITX_TOP_HDCP14_MIN_SIZE	RW	Secure video protect.
0x035<<2	HDMITX_TOP_HDCP22_MIN_SIZE	RW	Secure video protect.
0x040<<2	HDMITX_TOP_SECURE_INDEX	RW	Secure core register bank start index.
0x041<<2	HDMITX_TOP_SECURE_DATA	RW	Secure core register bank content.

## Register Description

**Table 9-3292 HDMITX\_TOP\_SW\_RESET**

Bit	R/W	Default	Description
31:8	R	0	Reserved
15:7	RW	0x1ff	Reserved

Bit	R/W	Default	Description
6	RW	1	sw_reset_ft: to reset DDC&CEC input glitch filter. 0=Release from reset; 1=Apply reset.
4	RW	1	sw_reset_phyif: to reset PHY interface. 0=Release from reset; 1=Apply reset.
3	RW	1	sw_reset_intr: to reset interrupt block. 0=Release from reset; 1=Apply reset.
0	RW	1	sw_reset_core: To reset TX Controller IP. 0=Release from reset; 1=Apply reset.

Table 9-3293 HDMITX\_TOP\_CLK\_CNTL

Bit	R/W	Default	Description
31	RW	0	free_clk_en: 0= Enable clock gating for power saving; 1= Disable clock gating, enable free-run clock.
30:14	RW	0	Reserved
13	RW	0	aud_mclk_sel: Select to use which clock for ACR measurement. 0= Use i2s_mclk; 1=Use spdif_clk.
12	RW	0	i2s_ws_inv: 1= Invert i2s_ws.
11	RW	0	i2s_clk_inv: 1= Invert i2s_clk.
9	RW	0	tmads_clk_inv: 1= Invert tmads_clk.
8	RW	0	pixel_clk_inv: 1= Invert pixel_clk.
4	RW	0	Reserved.
3	RW	0	i2s_clk_en: 1= Enable i2s_clk.
1	RW	0	tmads_clk_en: 1= Enable tmads_clk.
0	RW	0	pixel_clk_en: 1= Enable pixel_clk.

Table 9-3294 HDMITX\_TOP\_HPD\_FILTER

Bit	R/W	Default	Description
31:16	R	0	Reserved
31:28	RW	0	rxsense_glitch_width: Filter out glitch <= rxsense_glitch_width.
27:16	RW	0	rxsense_valid_width: Filter out width <= rxsense_valid_width * 1024.

Bit	R/W	Default	Description
15:12	RW	0	hpd_glitch_width: Filter out glitch $\leq$ hpd_glitch_width.
11:0	RW	0	hpd_valid_width: Filter out width $\leq$ hpd_valid_width * 1024.

**Table 9-3295 HDMITX\_TOP\_INTR\_MASKN**

Interrupt MASKN, one bit per interrupt source.

0= Disable interrupt source;

1= Enable interrupt source.

Bit	R/W	Default	Description
31:9	R	0	Reserved
8	RW	0	hdcp_topology_err
7	RW	0	rxsense_fall
6	RW	0	rxsense_rise
5	RW	0	err_i2c_timeout
4	RW	0	HS_intr
3	RW	0	TX Controller aon_intr interrupt
2	RW	0	hpd_fall
1	RW	0	hpd_rise
0	RW	0	TX Controller pwd_intr interrupt.

**Table 9-3296 HDMITX\_TOP\_INTR\_STAT**

Interrupt status. For each bit of bit[8:0], write 1 to manually set the interrupt bit, read back the interrupt status.

Bit	R/W	Default	Description
31	R	0	Shadowing TX Controller pwd_intr status flag.
30	R	0	Shadowing TX Controller aon_intr status flag.
29:9	R	0	Reserved
8	RW	0	hdcp_topology_err
7	RW	0	rxsense_fall
6	RW	0	rxsense_rise
5	RW	0	err_i2c_timeout
4	RW	0	HS_intr
3	RW	0	TX Controller aon_intr interrupt
2	RW	0	hpd_fall

Bit	R/W	Default	Description
1	RW	0	hpd_rise
0	RW	0	TX Controller pwd_intr interrupt.

**Table 9-3297 HDMITX\_TOP\_INTR\_STAT\_CLR**

Interrupt status clear. For each bit, write 1 to clear the interrupt bit.

Bit	R/W	Default	Description
31:9	R	0	Reserved
8	W	0	hdcp_topology_err
7	W	0	rxsense_fall
6	W	0	rxsense_rise
5	W	0	err_i2c_timeout
4	W	0	HS_intr
3	W	0	TX Controller aon_intr interrupt
2	W	0	hpd_fall
1	W	0	hpd_rise
0	W	0	TX Controller pwd_intr interrupt.

**Table 9-3298 HDMITX\_TOP\_BIST\_CNTL**

Bit	R/W	Default	Description
31:16	R	0	Reserved
15	RW	0	Reserved
14:12	RW	0	tmnds_sel: 3'b000=Output zero; 3'b001=Output normal TMDS data; 3'b010=Output PRBS data; 3'b100=Output shift pattern.
11: 9	RW	0	shift_pttn_repeat: 0=New pattern every clk cycle; 1=New pattern every 2 clk cycles; ...; 7=New pattern every 8 clk cycles.
8	RW	0	shift_pttn_en: 1= Eanble shift pattern generator; 0=Disable.
7:5	RW	0	Reserved
4: 3	RW	0	prbs_pttn_mode: 0=PRBS11; 1=PRBS15; 2=PRBS7; 3=PRBS31.

Bit	R/W	Default	Description
2:1	RW	0	prbs_pttn_width: 0=Idle; 1=Output 8-bit pattern; 2=Output 1-bit pattern; 3=Output 10-bit pattern.
0	RW	0	prbs_pttn_en: 1=Enable PRBS generator; 0=Disable.

Table 9-3299 HDMITX\_TOP\_SHIFT\_PTTN\_012

Bit	R/W	Default	Description
31:30	R	0	Reserved
29:20	RW	0	shift_pttn_data[59:50].
19:10	RW	0	shift_pttn_data[69:60].
9:0	RW	0	shift_pttn_data[79:70].

Table 9-3300 HDMITX\_TOP\_SHIFT\_PTTN\_345

Bit	R/W	Default	Description
31:30	R	0	Reserved
29:20	RW	0	shift_pttn_data[29:20].
19:10	RW	0	shift_pttn_data[39:30].
9:0	RW	0	shift_pttn_data[49:40].

Table 9-3301 HDMITX\_TOP\_SHIFT\_PTTN\_67

Bit	R/W	Default	Description
31:20	R	0	Reserved
19:10	RW	0	shift_pttn_data[9:0].
9:0	RW	0	shift_pttn_data[19:10].

Table 9-3302 HDMITX\_TOP\_TMDS\_CLK\_PTTN\_01

Bit	R/W	Default	Description
31:26	R	0	Reserved
25:16	RW	0	tmcls_clk_pttn[19:10].
15:10	R	0	Reserved
9:0	RW	0	tmcls_clk_pttn[9:0].



**Table 9-3303 HDMITX\_TOP\_TMDS\_CLK\_PTTN\_23**

Bit	R/W	Default	Description
31:26	R	0	Reserved
25:16	RW	0	tmds_clk_pttn[39:30].
15:10	R	0	Reserved
9:0	RW	0	tmds_clk_pttn[29:20].

**Table 9-3304 HDMITX\_TOP\_TMDS\_CLK\_PTTN\_CNTL**

Bit	R/W	Default	Description
31:2	R	0	Reserved
1	RW	0	shift_tmds_clk_pttn: 1=Enable shifting clk pattern, used when TMDS CLK rate = TMDS character rate /4.
0	W	0	load_tmds_clk_pttn: Write this bit to 1 to load tmds_clk_pttn to HW. Always read back 0.

**Table 9-3305 HDMITX\_TOP\_STAT0**

Bit	R/W	Default	Description
31:2	R	0	Reserved
1	R	0	filtered RxSense status: 0= RxSense low; 1= RxSense high.
0	R	0	filtered HPD status: 0= HPD low; 1= HPD high.

**Table 9-3306 HDMITX\_TOP\_INFILTER**

Bit	R/W	Default	Description
31:27	RW	0	Reserved
26:24	RW	0	For DDC infilter: filter internal clock divider. 0=No divide; 1=Divide by 2; 2=Divide by 3; ... 7=Divide by 8.
23:16	RW	0	For DDC infilter: sampling clock divider. 0=No divide; 1=Divide the filter sampling clock by 2; 2=Divide the filter sampling clock by 3; ... 255=Divide the filter sampling clock by 256;
15:0	RW	0	Reserved

**Table 9-3307 HDMITX\_TOP\_NSEC\_SCRATCH**

Bit	R/W	Default	Description
31:0	RW	0	Scratch register that can be used for either secure or non-secure reg access.

**Table 9-3308 HDMITX\_TOP\_SEC\_SCRATCH**

Bit	R/W	Default	Description
31:1	RW	0	Scratch register that can be used for secure reg access only.
1	RW	0	cor_reg_access_en: 0=Cannot access Controller core registers; 1=Enable access Controller core register.

**Table 9-3309 HDMITX\_TOP\_HS\_INTR\_CNTL**

Bit	R/W	Default	Description
17	RW	0	HS_polarity. 0=Use HS rise edge as HS pulse; 1=Use HS fall edge as HS pulse.
16	RW	0	VS_polarity. 0=Use VS rise edge as VS pulse; 1=Use VS fall edge as VS pulse.
15:0	RW	0	HS_latency: assert an HS_INTR after seeing VS pulse, and after seeing HS_latency number of HS pulses.

**Table 9-3310 HDMITX\_TOP\_AUD\_ID**

Bit	R/W	Default	Description
4:0	RW	0	Indicate to core of which audio source is being selected. 5'b00001 = SPDIF; 5'b00010 = I2S; 5'b01000 = HBR audio.

**Table 9-3311 HDMITX\_TOP\_HV\_ACTIVE**

Bit	R/W	Default	Description
30:16	RW	0	Vactive: active video height..
14:0	RW	0	Hactive: active video width.

**Table 9-3312 HDMITX\_TOP\_I2C\_BUSY\_CNT\_MAX**

Bit	R/W	Default	Description
31:0	RW	0xffffffff	i2c_busy_cnt_max. After I2C Start bit, if I2C bus is static for more than i2c_busy_cnt_max number of cycles, an error interrupt will happen.

**Table 9-3313 HDMITX\_TOP\_I2C\_BUSY\_CNT\_STAT**

Bit	R/W	Default	Description
31:0	R	0	i2c_busy_cnt. Reflect the number of idle cycles after the I2C Start bit, for the latest I2C transaction.

**Table 9-3314 HDMITX\_TOP\_DDC\_CNTL**

Bit	R/W	Default	Description
31:2	R	0	Reserved.
1	RW	1	DSDA pull down. 0=Pull down DSDA. 1=No pull down.
0	RW	1	DSCL pull down. 0=Pull down DSCL. 1=No pull down.

**Table 9-3315 HDMITX\_TOP\_FIX\_DISABLE\_0**

Bit	R/W	Default	Description
31:0	RW	0	fix_disable[31:0]. Each bit to control whether to fix a internal time crossing bug. 0=Enable fix; 1=Go back to original behavior.

**Table 9-3316 HDMITX\_TOP\_FIX\_DISABLE\_1**

Bit	R/W	Default	Description
31:0	RW	0	fix_disable[63:32]. Each bit to control whether to fix a internal time crossing bug. 0=Enable fix; 1=Go back to original behavior.

**Table 9-3317 HDMITX\_TOP\_SEC\_VIDEO\_OVR**

There is a HW bit from VPU to indicate if the video content need secure protection, at the input of HDMITX.

When this bit=1, HDMITX\_TOP\_SEC\_VIDEO\_OVR, HDMITX\_TOP\_HDCP14\_MIN\_SIZE, HDMITX\_TOP\_HDCP22\_MIN\_SIZE will be used.

When this bit=0, then these three registers are don't care.

When this bit=1, the rule for HW to judge if the override secure video with HDMITX\_TOP\_SEC\_VIDEO\_OVR is as follows:

If the video size < HDMITX\_TOP\_HDCP14\_MIN\_SIZE, then no need override, regardless of authentication status.

If the video size >= HDMITX\_TOP\_HDCP14\_MIN\_SIZE,

but < HDMITX\_TOP\_HDCP22\_MIN\_SIZE, then override until HDCP1.4 or HDCP2.2 is authenticated.

If the video size >= HDMITX\_TOP\_HDCP22\_MIN\_SIZE, then override until HDCP2.2 is authenticated.

Bit	R/W	Default	Description
29:20	RW	0	Video data for tmds channel 2, when override.
19:10	RW	0	Video data for tmds channel 1, when override.
9:0	RW	0	Video data for tmds channel 0, when override.

Table 9-3318 HDMITX\_TOP\_HDCP14\_MIN\_SIZE

Bit	R/W	Default	Description
31	RW	0	hdcp14_min_size_v_en
30:16	RW	0	hdcp14_min_size_v[14:0]
15	RW	0	hdcp14_min_size_h_en
14:0	RW	0	hdcp14_min_size_h[14:0]

Table 9-3319 HDMITX\_TOP\_HDCP22\_MIN\_SIZE

Bit	R/W	Default	Description
31	RW	0	hdcp22_min_size_v_en
30:16	RW	0	hdcp22_min_size_v[14:0]
15	RW	0	hdcp22_min_size_h_en
14:0	RW	0	hdcp22_min_size_h[14:0]

Table 9-3320 HDMITX\_TOP\_SECURE\_INDEX

Bit	R/W	Default	Description
15:0	RW	0	Start pointer of the 32-deep secure core register table

Table 9-3321 HDMITX\_TOP\_SECURE\_DATA

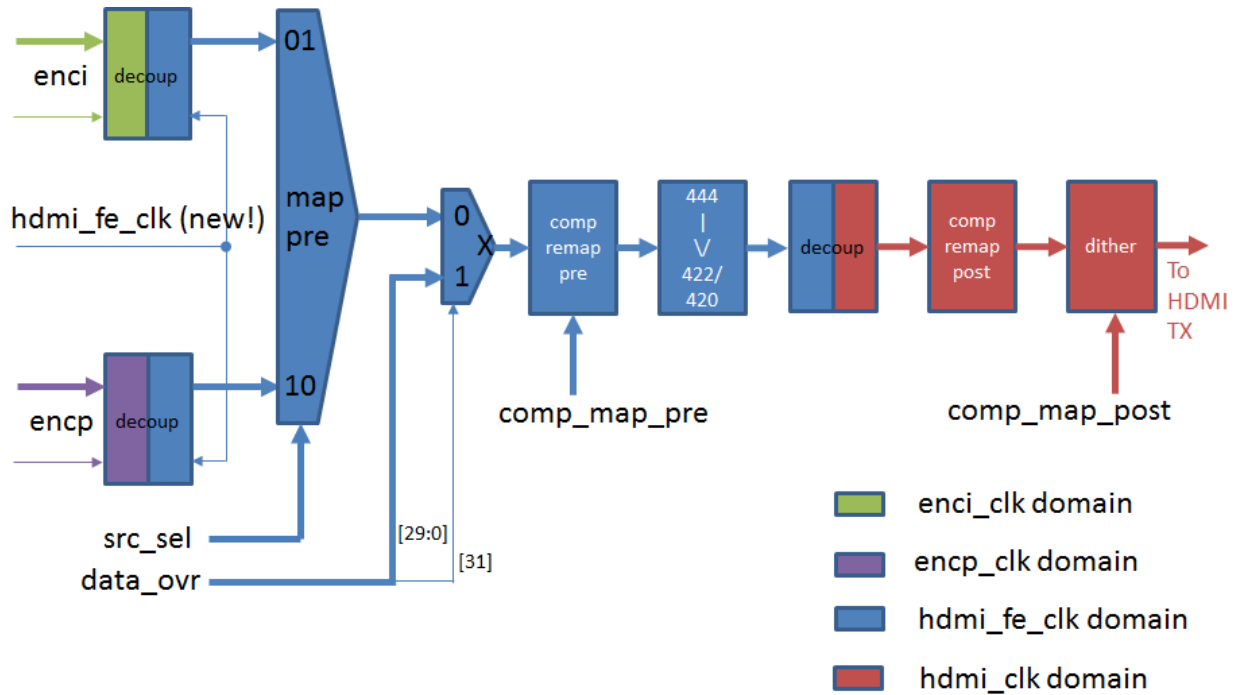
Bit	R/W	Default	Description
15:0	RW	0	The core register address value to be programmed into the 32-deep table.

### 9.2.3.50.3 VPU\_HDMI\_IF Module

#### Function

VPU\_HDMI\_IF is a sub-module inside the VPU. It takes video data from either ENCI or ENCP, performs functions such as video component remapping, 444 to 422 or 420 conversion and dither. Below is the block diagram of the module.

Figure 9-5 VPU\_HDMI\_IF Function



Register Description

Table 9-3322 vpu\_hdmi\_setting

Bit	R/W	Default	Description
27:24	RW	0	rd_data_post (New): Read rate to the async FIFO between HDMI_FE_CLK domain and HDMI_CLK domain. 0=One read every hdmi_clk; 1=One read every 2 hdmi_clk; 2=One read every 3 hdmi_clk; ... 15=One read every 16 hdmi_clk.
23:20	RW	0	wr_data_post (New): Write rate to the async FIFO between HDMI_FE_CLK domain and HDMI_CLK domain. 0=One write every hdmi_fe_clk; 1=One write every 2 hdmi_fe_clk; 2=One write every 3 hdmi_fe_clk; ... 15=One write every 16 hdmi_fe_clk.
18:16	RW	0	comp_map_pre (New): In case the input data is not CrYCb, map the data to CrYCb format, before 444->422/420 conversion. E.g, if input data format is {2, 1, 0} 0=Output {2, 1, 0}; 1=Output {1, 0, 2}; 2=Output {1, 2, 0}; 3=Output {0, 2, 1}; 4=Output {0, 1, 2}; 5=Output {2, 0, 1}; 6,7=Reserved.
15:12	RW	0	rd_data_pre (Original): Read rate to the async FIFO between VENC and HDMI_FE_CLK domain. 0=One read every hdmi_fe_clk;

Bit	R/W	Default	Description
			1=One read every 2 hdmi_fe_clk; 2=One read every 3 hdmi_fe_clk; ... 15=One read every 16 hdmi_fe_clk.
11:8	RW	0	wr_data_pre (Original): Write rate to the async FIFO between VENC and HDMI_FE_CLK domain. 0=One write every enc_clk; 1=One write every 2 enc_clk; 2=One write every 3 enc_clk; ... 15=One write every 16 enc_clk.
7:5	RW	0	comp_map_post (Original): Input data is CrYCb(BRG), map the output data to desired format. 0=Output {Cr, Y, Cb} or {B, R, G}; 1=Output {Y, Cb, Cr} or {R, G, B}; 2=Output {Y, Cr, Cb} or {R, B, G}; 3=Output {Cb, Cr, Y} or {G, B, R}; 4=Output {Cb, Y, Cr} or {G, R, B}; 5=Output {Cr, Cb, Y} or {B, G, R}; 6,7=Reserved.
4	RW	0	inv_dvi_clk. If true, invert the polarity of clock output to external DVI interface. (NOT internal HDMI).
3	RW	0	inv_vsync: If true, invert the polarity of VSync input from VENC.
2	RW	0	inv_hsync: If true, invert the polarity of HSync input from VENC.
1:0	RW	0	src_sel: Select which HDMI source from between ENCI and ENCP. 2'b00: Disable HDMI source; 2'b01: Select ENCI data to HDMI; 2'b10: Select ENCP data to HDMI; 2'b11: Not allowed.

Table 9-3323 vpu\_hdmi\_data\_ovr

Bit	R/W	Default	Description
31	RW	0	data_ovr_en: Control if override HDMI input data with data_ovr[29:0], for display e.g. black or blue screen. 0: No override; 1: Enable override.
29:20	RW	0	data_ovr[29:20]: Programmable pixel data value for override.
19:10	RW	0	data_ovr[19:10]: Programmable pixel data value for override.
9:0	RW	0	data_ovr[9:0]: Programmable pixel data value for override.

Table 9-3324 vpu\_hdmi\_fmt\_ctrl

Bit	R/W	Default	Description
27:24	RW	0	pix_repeat. Need to program to be the same as HDMI IP internal pixel repeat rate. 0 = No repeat; 1 = Repeat once; ....
23:22	RW	0	chroma_dnsmv: Vertical chroma down sample mode when convert to 422 or 420. 0 = use line 0; 1 = use line 1;

Bit	R/W	Default	Description
			2 = use average.
21:19	RW	0	Frame count offset for B.
18:16	RW	0	Frame count offset for G.
15	RW	0	hcnt hold when de valid.
14	RW	0	RGB frame count separate.
13	RW	0	dith4x4: frame random enable
12	RW	0	dith4x4 enable
11	RW	0	tunnel_en: tunnel enable for DOLBY.
10	RW	0	hdmi_round_en
9:6	RW	0	hdmi_dith10
5	RW	0	hdmi_dith_md
4	RW	0	hdmi_dith_en
3:2	RW	0	chroma_dnsmpl_h: Horizontal chroma down sample mode when convert to 422 or 420. 0 = use pixel 0; 1 = use pixel 1; 2 = use average.
1:0	RW	0	hdmi_vid_fmt: Control whether to convert ENC's 444 data to 422 or 420. 0 = No conversion; 1 = Convert to 422; 2 = Convert to 420.

#### 9.2.3.50.4 Operating Procedures

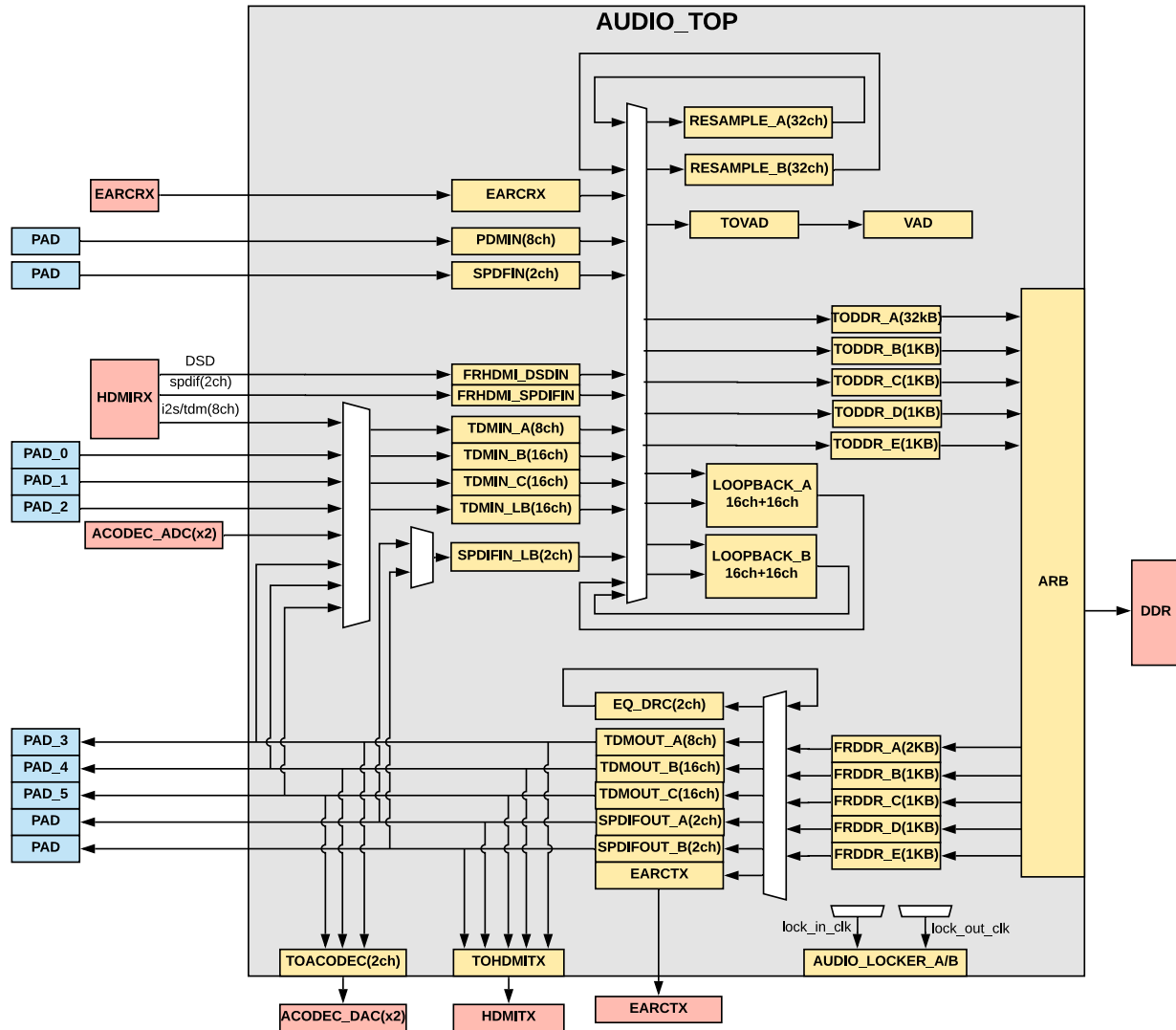
This section describes the procedure of protecting a selection of core registers to be secure mode access only.

- After reset, none of the IP registers are accessible, regardless secure mode or not.
- Via secure mode, program HDMITX\_TOP\_SECURE\_INDEX to an appropriate value, e.g. = 0
- Repetitively use secure write to HDMITX\_TOP\_SECURE\_DATA with the list of IP register addresses that you'd like to set to be secure-access only. Only up to 32 IP registers can be selected.

# 10 Audio Path

The SoC integrates 3 input/ 3 output TDM interface, 2 SPDIF output interface, 5 TODDR (FIFO) for transferring input data to DDR, 5 FRDDR (FIFO) for transferring data from DDR to output, 1 TDM LB and 1 Loopback for AEC, 1 HW resample for clock synchronization, 1 HW resample for AEC, 1 VAD voice wake up, and 2 clock locker detect difference of two clock. Below is the diagram for audio path.

Figure 10-1 Audio Path



## 10.1 Audio Clock Tree

Audio clocks are described in the following table.

Clock Source	Diagram
6 master mclk	Figure 10-2
3 different phase sclk/lrclk generated by each mclk	
<ul style="list-style-type: none"> <li>● phase0 for pad (device);</li> <li>● phase1 for tdm;in;</li> <li>● phase2 for tdm;out;</li> </ul>	



Clock Source	Diagram
<ul style="list-style-type: none"> <li>● generate master mclk to pad;</li> <li>● generate mst*_clk to pad by mst*_ph0;</li> <li>● generate tadmin*_clk by mst*_ph1 and slv*_clk;</li> <li>● generate tdmout*_clk by mst*_ph2 and slv*_clk;</li> <li>● generate tadmin_lb_clk by mst*_ph2 and slv*_clk;</li> </ul>	<p><a href="#">Figure 10-3</a></p>
<p>other clocks</p>	<p><a href="#">Figure 10-4</a></p>
<p>slv clocks are from PINMUX, which are defined as:</p> <ul style="list-style-type: none"> <li>● slv_sclk_a = SLV_SCLK_0</li> <li>● slv_sclk_b = SLV_SCLK_1</li> <li>● slv_sclk_c = SLV_SCLK_2</li> <li>● slv_sclk_d = SLV_SCLK_3</li> <li>● slv_sclk_e = SLV_SCLK_4</li> <li>● slv_sclk_f = HDMIRX_I2S_SCLK</li> <li>● slv_sclk_g = WORLD_SYNC</li> <li>● others = 0</li> <li>● slv_lrclk_a = SLV_LRCLK_0</li> <li>● slv_lrclk_b = SLV_LRCLK_1</li> <li>● slv_lrclk_c = SLV_LRCLK_2</li> <li>● slv_lrclk_d = SLV_LRCLK_3</li> <li>● slv_lrclk_e = SLV_LRCLK_4</li> <li>● slv_lrclk_f = HDMIRX_I2S_LRCLK</li> <li>● others = 0</li> </ul>	<p>-</p>

Figure 10-2 Audio Clock Tree (1)

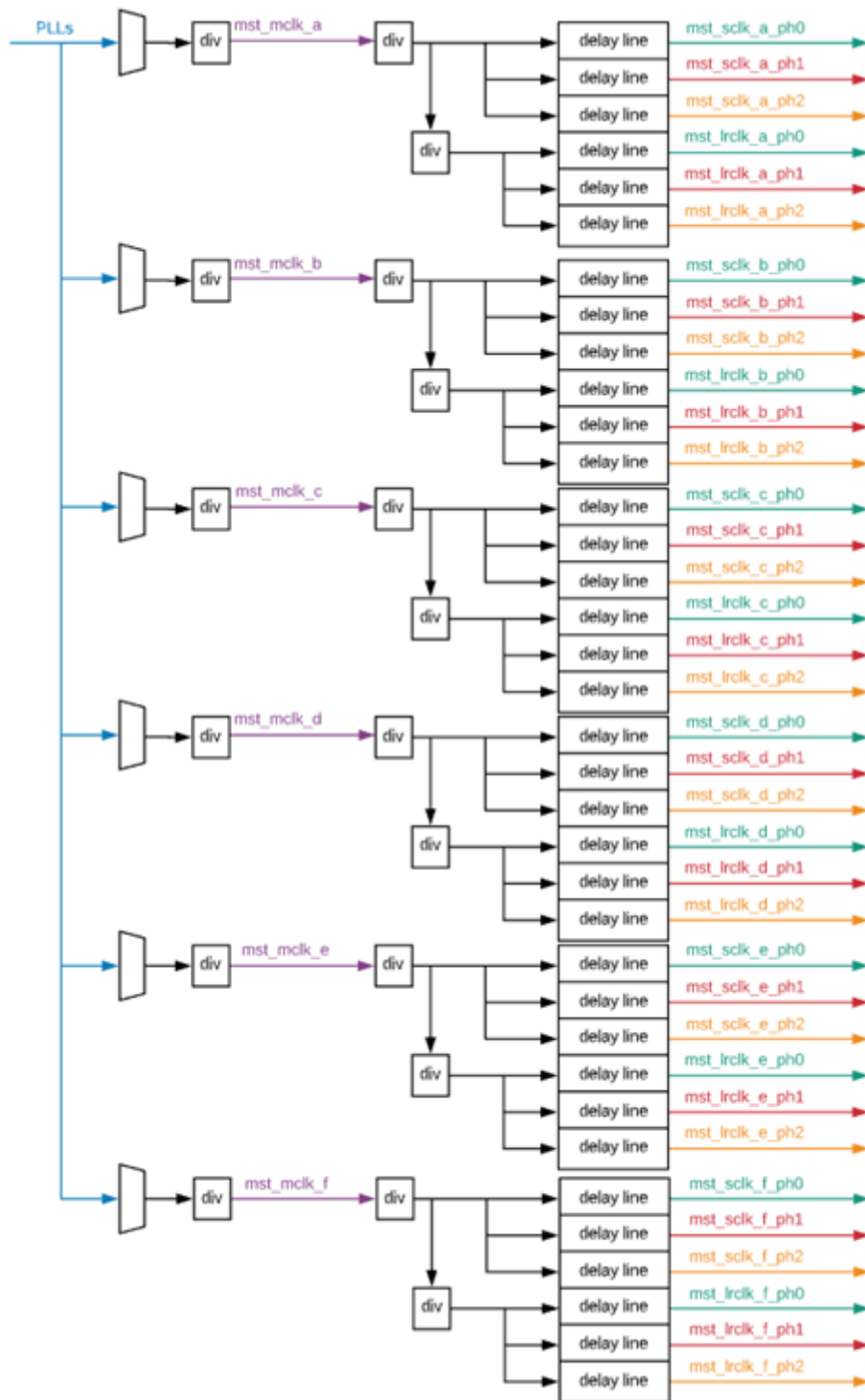


Figure 10-3 Audio Clock Tree (2)

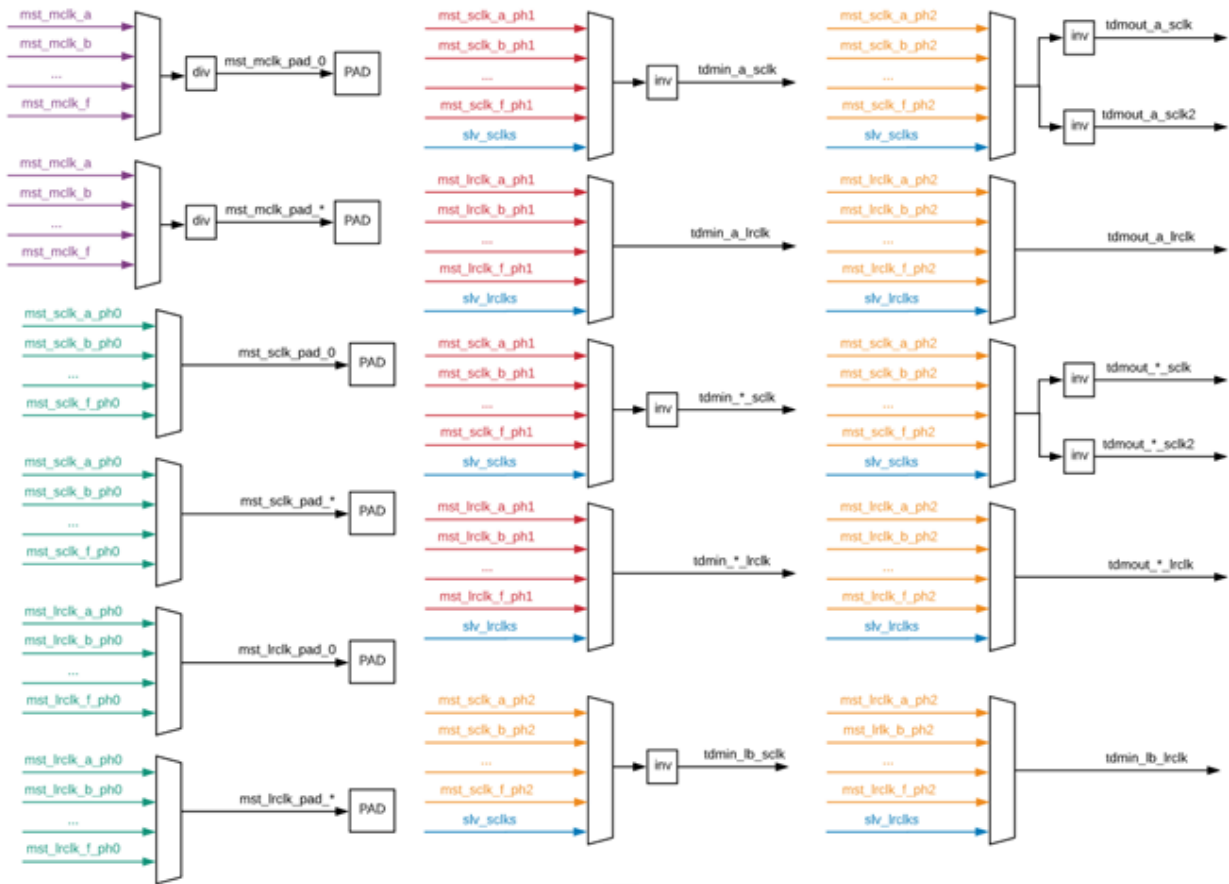


Figure 10-4 Audio Clock Tree (3)



## 10.2 Audio Input

## 10.2.1 Overview

This section describes TDM input interface, SPDIF input interface and PDM input interface.

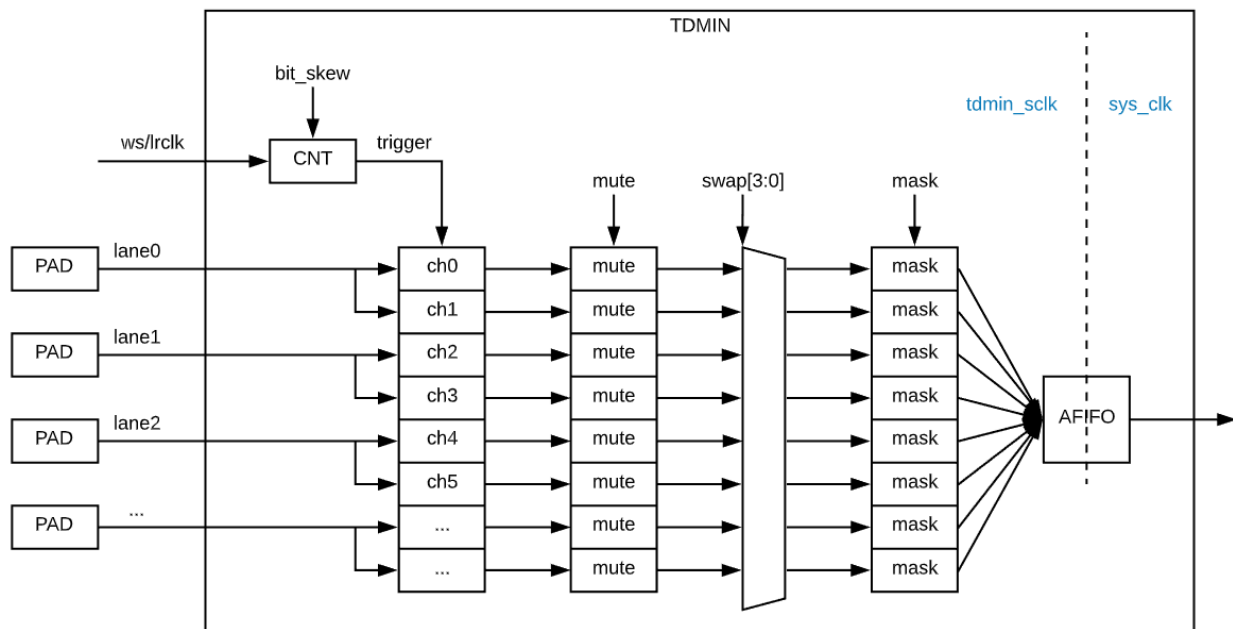
### 10.2.2 TDM Input Interface

TDM input interface works in the following way:

- All worked at tdm\_sclk before AFIFO,
- Output is worked at sysclk;
- Add skew control between lrclock and data;
- Detect rise or fall edge of lrclock and clear bit\_cnt/slot\_cnt;
- Shift serial data in to 8 parallel register (max 32bits) ;
- Swap parallel register to sample data;
- Send sample data out in serials by mask/mute configure;

Below is the diagram of TDM input interface.

Figure 10-5 TDM Input



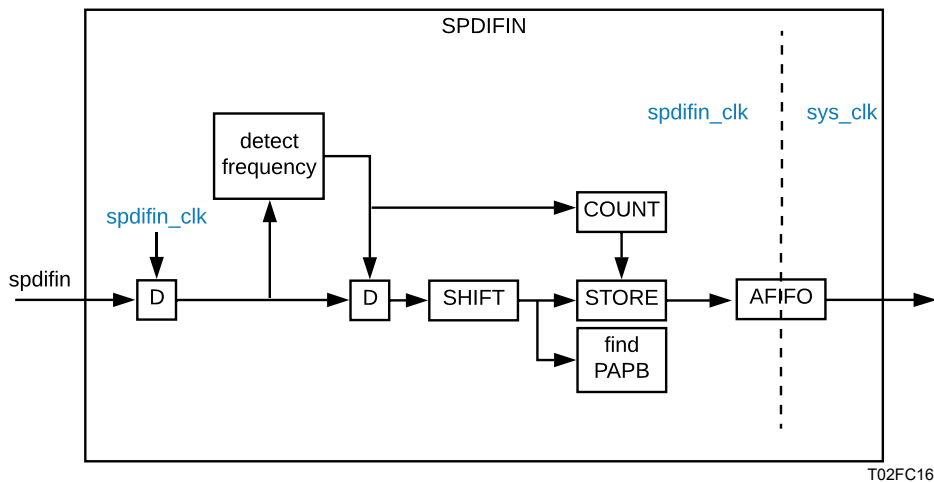
### 10.2.3 SPDIF Input Interface

SPDIF input interface works in the following way:

- All worked at spdifin\_clk before AFIFO;
- After AFIFO worked at sysclk;
- Detect frequency by rise or fall edge;
- Capture data to shift reg;
- Store data to sample L channel or R channel;
- Detect PaPb if it's IEC60937;

Below is the diagram of SPDIF input interface.

Figure 10-6 SPDIF Input



### 10.2.4 PDM

PDM input Decimation Filter is a highly programmable multistage decimation filter. It supports 4 inputs or 8 channel PDM digital microphone interface. Each channel contains one 9-stage CIC filter, 3 low power filters and a high pass filter. The PDM input bit rate is calculated with  $fs \cdot OSR$ . OSR can be 64, 128, 192 or 256.

CIC filter may have 3 to 9 CIC stages and the down sample rate also can be programable. There is a multiplier and shifter to adjust the result to match the accuracy and CIC bit width.

There are 3 low pass filters which shared a 336x24 Coefficient memory and a 336x28 data memory. That means these 3 filters can be configured up to 336 taps together. The coefficient memory needs to be programmed through APB bus. For example, if filter 1 contains 140 taps, filter 2 contains 32 taps, filter 3 contain 146 steps, filter 1 coefficient will use 0~139 of the coefficient memory address; filter 2 coefficient will use 140~171 of coefficient memory address and filter 3 coefficient will use 172~317 of the coefficient memory address. The filter stage controller will be based on the down sample rate, filter taps and the rounding mode of each filter to arrange the filter.

The final high pass filter is used to filter the DC current.

Figure 10-7 PDM Decimation Filter

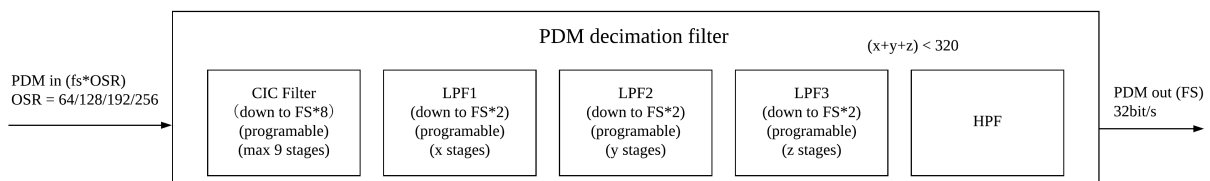
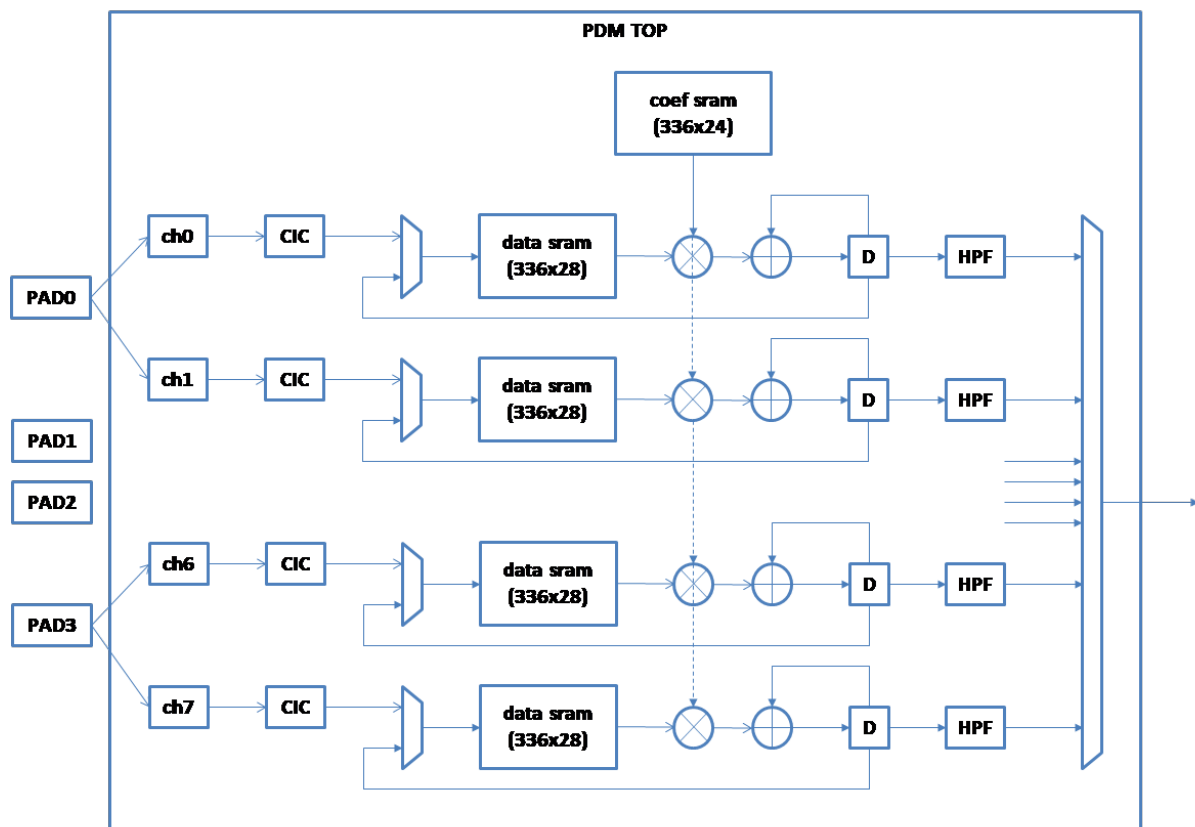


Figure 10-8 PDM Structure

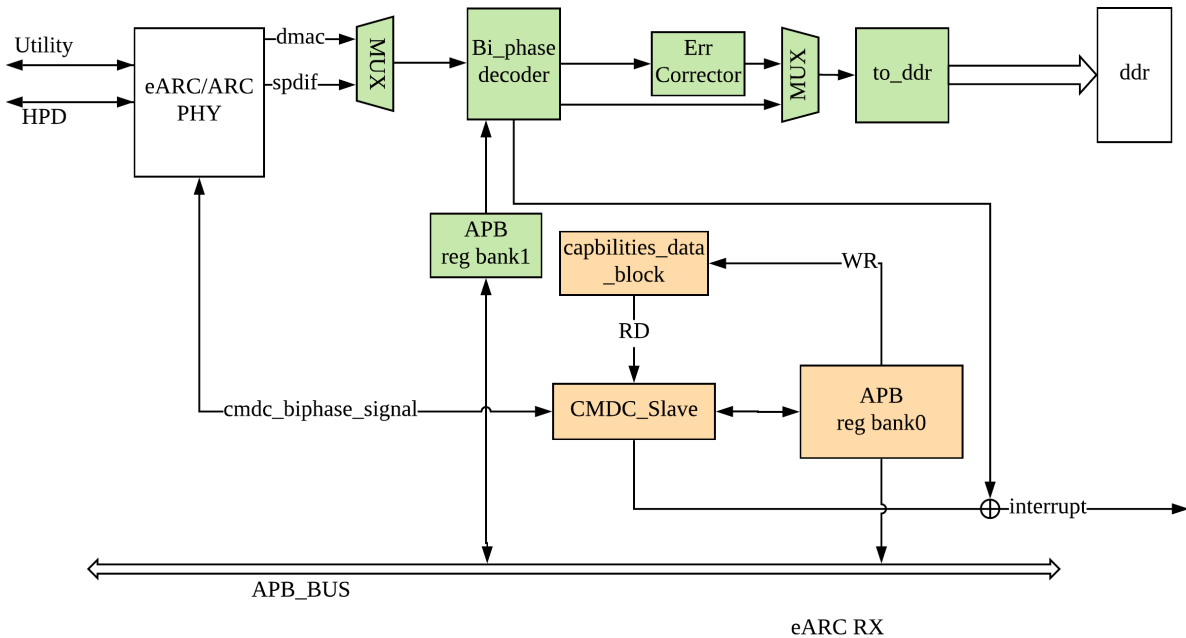


## 10.2.5 eARC RX

### 10.2.5.1 Overview

#### Structure

Figure 10-9 eARC RX Structure



#### eARC RX CMDC

The CMDC supports following features.

- Establish eARC mode, ARC mode, or mode switching between them.
- Status updates of eARC connection.
- Transmit audio channel delay information.
- Transmit eARC channel audio format compatibility information.

#### eARC RX DMAC

The DMAC supports both ARC and eARC functions. In eARC mode, the dmac path data is selected. In the ARC mode, the SPDIF path data is selected.

In eARC mode, dmac data and clock are from eARC RX analog phy. After bi\_phase decoding, Error corrector processing is performed according to whether the audio is compressed or not, and data is written into DDR.

In ARC mode, spdif data comes from eARC RX analog phy, which is bi\_phase decoded after down-sampling and data is written to DDR.

The user bit is supported to transmit data, and the received data can trigger the interrupt notification software to read.

## 10.2.5.2 Register Description

### 10.2.5.2.1 eARC RX CMDC Registers

Baseaddr : 0xfe333800 + offset\*4

**Table 10-1 EARC\_RX\_CMDC\_TOP\_CTRL0 0x0**

Bit(s)	R/W	Default	Description
31	R/W	0	idle2_int 1: enable
30	R/W	0	idle1_int 1: enable
29	R/W	0	disc2_int 1: enable
28	R/W	0	disc1_int 1: enable
27	R/W	0	earc_int 1: enable
26	R/W	0	hb_status_int 1: enable
25	R/W	0	losthb_int 1: enable
24	R/W	0	timeout_int 1: enable
23	R/W	0	status_ch_int 1: enable
22	R/W	0	int_rec_invalid_id 1: enable
21	R/W	0	int_rec_invalid_offset 1: enable
20	R/W	0	int_rec_unexp 1: enable
19	R/W	0	int_rec_ecc_err 1: enable
18	R/W	0	int_rec_parity_err 1: enable
17	R/W	0	int_rcv_packet 1: enable
16	R/W	0	int_rec_time_out 1: enable
15	R/W	0	cmdc_debug0 1: enable
14	R/W	0	cmdc_debug1 1: enable
13	R/W	0	cmdc_debug2 1: enable
12~7	R/W		reserved
6	R/W	0	mute_select 1: use bit5, 0: earc off
5	R/W	0	mute_contrl value of mannul mute control
4~0	R/W		reserved

**Table 10-2 EARC\_RX\_CMDC\_TOP\_CTRL1 0x1**

Bit(s)	R/W	Default	Description
31~13	R/W		reserved
12~8	R/W	RW, 0	reg_scan_reg
7~5	R/W		reserved
4~0	R/W	RW, 0	reg_top_soft_rst



**Table 10-3 EARC\_RX\_CMDC\_TOP\_CTRL2 0x2**

Bit(s)	R/W	Default	Description
31	R/W	0	reset_idle2_int
30	R/W	0	reset_idle1_int
29	R/W	0	reset_disc2_int
28	R/W	0	reset_disc1_int
27	R/W	0	reset_earc_int
26	R/W	0	reset_hb_status_int
25	R/W	0	reset_losthb_int
24	R/W	0	reset_timeout_int
23	R/W	0	reset_status_ch_int
22	R/W	0	reset_int_rec_invalid_id
21	R/W	0	reset_int_rec_invalid_offset
20	R/W	0	reset_int_rec_unexp
19	R/W	0	reset_int_rec_ecc_err
18	R/W	0	reset_int_rec_parity_err
17	R/W	0	reset_int_recv_packet
16	R/W	0	reset_int_rec_time_out
15~00	R/W		reserved

**Table 10-4 EARC\_RX\_CMDC\_TIMER\_CTRL0 0x3**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status0

**Table 10-5 EARC\_RX\_CMDC\_TIMER\_CTRL1 0x4**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status0

**Table 10-6 EARC\_RX\_CMDC\_TIMER\_CTRL2 0x5**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status0

**Table 10-7 EARC\_RX\_CMDC\_TIMER\_CTRL3 0x6**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status0

Table 10-8 EARC\_RX\_CMDC\_VSM\_CTRL0 0x7

Bit(s)	R/W	Default	Description
31	R/W	0	sw_state_update
30~28	R/W	0	sw_state
27	R/W	0	arc_initiated
26	R/W	0	arc_terminated
25	R/W	0	arc_enable
24	R/W	0	man_hpd
23~22	R/W	0	hpd_sel
21~20	R/W	0	hpd_sel_earc
19	R/W	0	comma_cnt_rst
18	R/W	0	timeout_status_rst
17	R/W	0	losthb_status_rst
16	R/W	0	force_rst
15	R/W	0	auto_state
14	R/W	0	cmdc_state_en
13~00	R/W		reserved

Table 10-9 EARC\_RX\_CMDC\_VSM\_CTRL1 0x8

Bit(s)	R/W	Default	Description
31~12	R/W	0	max_count_th idle done timing
11~8	R/W		reserved
7	R/W	0	reg_soft_rst idle done timing
6~4	R/W	0	time_sel idle done timing
3~2	R/W	0	soft_rst_sel idle done timing
1~0	R/W	0	enable_ctrl idle done timing

Table 10-10 EARC\_RX\_CMDC\_VSM\_CTRL2 0x9

Bit(s)	R/W	Default	Description
31~12	R/W	0	max_count_th comma off done timing
11~8	R/W		reserved
7	R/W	0	reg_soft_rst comma off done timing
6~4	R/W	0	time_sel comma off done timing
3~2	R/W	0	soft_rst_sel comma off done timing
1~0	R/W	0	enable_ctrl comma off done timing

Table 10-11 EARC\_RX\_CMDC\_VSM\_CTRL3 0xa

Bit(s)	R/W	Default	Description
31~12	R/W	0	max_count_th earc_time out timing
11~8	R/W		reserved
7	R/W	0	reg_soft_rst earc_time out timing
6~4	R/W	0	time_sel earc_time out timing
3~2	R/W	0	soft_rst_sel earc_time out timing
1~0	R/W	0	enable_ctrl earc_time out timing

Table 10-12 EARC\_RX\_CMDC\_VSM\_CTRL4 0xb

Bit(s)	R/W	Default	Description
31~12	R/W	0	max_count_th heartbeat lost timing
11~8	R/W		reserved
7	R/W	0	reg_soft_rst heartbeat lost timing
6~4	R/W	0	time_sel heartbeat lost timing
3~2	R/W	0	soft_rst_sel heartbeat lost timing
1~0	R/W	0	enable_ctrl heartbeat lost timing

Table 10-13 EARC\_RX\_CMDC\_VSM\_CTRL5 0xc

Bit(s)	R/W	Default	Description
31~16	R/W		reserved
15~8	R/W	0	status_soft in earc heartbeat det timing
7	R/W	0	reg_soft_rst in earc heartbeat det timing
6	R/W	0	status_rst in earc heartbeat det timing
5~4	R/W		reserved
3~2	R/W	0	soft_rst_sel in earc heartbeat det timing
1~0	R/W	0	enable_ctrl in earc heartbeat det timing

Table 10-14 EARC\_RX\_CMDC\_VSM\_CTRL6 0xd

Bit(s)	R/W	Default	Description
31~17	R/W		reserved
16	R/W	0	cntl_hpd_sel in earc heartbeat det timing
15~4	R/W	0	cntl_hpd_valid_width in earc heartbeat det timing
3~0	R/W	0	cntl_hpd_glitch_width in earc heartbeat det timing

**Table 10-15 EARC\_RX\_CMDC\_VSM\_CTRL7 0xe**

Bit(s)	R/W	Default	Description
31~00	R/W	0	vsm_ctrl7

**Table 10-16 EARC\_RX\_CMDC\_VSM\_CTRL8 0xf**

Bit(s)	R/W	Default	Description
31~00	R/W	0	vsm_ctrl8

**Table 10-17 EARC\_RX\_CMDC\_VSM\_CTRL9 0x10**

Bit(s)	R/W	Default	Description
31~00	R/W	0	vsm_ctrl9

**Table 10-18 EARC\_RX\_CMDC\_SENDER\_CTRL0 0x11**

Bit(s)	R/W	Default	Description
31~02	R/W		reserved
1	R/W	0	hb_chg_conf_auto
0	R/W	1,	hb_chg_auto

**Table 10-19 EARC\_RX\_CMDC\_PACKET\_CTRL0 0x12**

Bit(s)	R/W	Default	Description
31	R/W	0	packet_mode_enable packet control
30	R/W	0	free_enable packet control
29	R/W	0	soft_rst_man packet control
28~24	R/W	0	ready_th packet control
23~20	R/W		reserved
19~8	R/W	0	send_pre_th packet control
7~5	R/W		reserved
4	R/W	0	sw_state_update packet control
3~0	R/W	0	sw_state packet control

**Table 10-20 EARC\_RX\_CMDC\_PACKET\_CTRL1 0x13**

Bit(s)	R/W	Default	Description
31	R/W	0	ecc_endian send
30	R/W	0	pre_reg_st send
29~21	R/W		reserved
20~16	R/W	0	post_th send

Bit(s)	R/W	Default	Description
15~14	R/W		reserved
13~8	R/W	0	pre_th
7~0	R/W	0	post_flag

Table 10-21 EARC\_RX\_CMDC\_PACKET\_CTRL2 0x14

Bit(s)	R/W	Default	Description
31~0	R/W		pre_flag unsigned,

Table 10-22 EARC\_RX\_CMDC\_PACKET\_CTRL3 0x15

Bit(s)	R/W	Default	Description
31	R/W	0	cmdc_en
30	R/W	0	cmdc_parity_mask
29	R/W	0	imeout_en w
28	R/W	0	ecc_check_en
27	R/W	0	rev_debug_en
26~16	R/W		reserved
15~0	R/W	0	timeout_th

Table 10-23 EARC\_RX\_CMDC\_PACKET\_CTRL4 0x16

Bit(s)	R/W	Default	Name
31	R/W	0	ack_ignore
30	R/W	0	cmdc_tail_check_mask
29~20	R/W		reserved
19~0	R/W	0	cmdc_packet_head

Table 10-24 EARC\_RX\_CMDC\_PACKET\_CTRL5 0x17

Bit(s)	R/W	Default	Name
31~24	R/W	0	rev_debug_mask
23~20	R/W		reserved
19~0	R/W	0	cmdc_packet_head_mask

**Table 10-25 EARC\_RX\_CMDC\_PACKET\_CTRL6 0x18**

Bit(s)	R/W	Default	Name
31~20	R/W	0	recv_pre_threshold
19~9	R/W		reserved
8	R/W	0	rec_packet_d
7	R/W	0	rec_parity_err_cnt
6	R/W	0	rec_ecc_err_cnt
5	R/W	0	rec_unexp_cnt
4	R/W	0	rec_invalid_offset_cnt
3	R/W	0	rec_invalid_id_cnt
2	R/W	0	rec_timeout_cnt
1	R/W	0	rec_w_cnt
0	R/W	0	rec_r_cnt

**Table 10-26 EARC\_RX\_CMDC\_BIPHASE\_CTRL0 0x19**

Bit(s)	R/W	Default	Name
31~24	R/W	7, xx	reg_tns
23~16	R/W	0	delay_th
15~10	R/W		reserved
9	R/W	0	send_ack_en
8	R/W	0	sq_val_en
7	R/W	0	biphase_send_soft_rst
6	R/W	0	comma_soft_rst
5	R/W	0	fifo_rst
4	R/W	0	receiver_no_sender
3	R/W	0	sender_free
2	R/W	0	receiver_send
1	R/W	0	receiver_earc
0	R/W	0	receiver_free

**Table 10-27 EARC\_RX\_CMDC\_BIPHASE\_CTRL1 0x1a**

Bit(s)	R/W	Default	Name
31~16	R/W		reserved
15	R/W	send	ack_val_en
14~8	R/W		reserved
7~0	R/W	0	width send

**Table 10-28 EARC\_RX\_CMDC\_BIPHASE\_CTRL2 0x1b**

Bit(s)	R/W	Default	Description
31	R/W	0	ack_val_en send
30~20	R/W		reserved
19~16	R/W	0	ack_rate comma send
15~00	R/W	0	width comma sen

**Table 10-29 EARC\_RX\_CMDC\_BIPHASE\_CTRL3 0x1c**

Bit(s)	R/W	Default	Description
31~0	R/W	0	biphase_ctrl3

**Table 10-30 EARC\_RX\_CMDC\_DEVICE\_ID\_CTRL 0x1d**

Bit(s)	R/W	Default	Description
31	R/W	0	apb_write apb bus wr/read
30	R/W	0	apb_read apb bus wr/read
29	R/W	0	apb_w_r_done apb bus wr/read
28	R/W	0	apb_w_r_reset apb bus wr/read
27~16	R/W		reserved
15~8	R/W	0	apb_w_r_id apb bus wr/read
7~0	R/W	0	apb_w_r_start_addr apb bus wr/read

**Table 10-31 EARC\_RX\_CMDC\_DEVICE\_WDATA 0x1e**

Bit(s)	R/W	Default	Description
31~08	R/W		reserved
7~00	R/W	0	apb_write_data apb bus wr/rea

**Table 10-32 EARC\_RX\_CMDC\_DEVICE\_RDATA 0x1f**

Bit(s)	R/W	Default	Description
31~08	R/W		reserved
7~00	R/W	0	apb_read_data apb bus wr/rea

**Table 10-33 EARC\_RX\_ANA\_CTRL0 0x20**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl0

**Table 10-34 EARC\_RX\_ANA\_CTRL1 0x21**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl1

**Table 10-35 EARC\_RX\_ANA\_CTRL2 0x22**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl2

**Table 10-36 EARC\_RX\_ANA\_CTRL3 0x23**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl3

**Table 10-37 EARC\_RX\_ANA\_CTRL4 0x24**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl4

**Table 10-38 EARC\_RX\_ANA\_CTRL5 0x25**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl5

**Table 10-39 EARC\_RX\_ANA\_STAT0 0x26**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_ANA_status0

**Table 10-40 EARC\_RX\_CMDC\_STATUS0 0x27**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status0

**Table 10-41 EARC\_RX\_CMDC\_STATUS1 0x28**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status1

**Table 10-42 EARC\_RX\_CMDC\_STATUS2 0x29**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status2



**Table 10-43 EARC\_RX\_CMDC\_STATUS3 0x2a**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status3

**Table 10-44 EARC\_RX\_CMDC\_STATUS4 0x2b**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status4

**Table 10-45 EARC\_RX\_CMDC\_STATUS5 0x2c**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status5

**Table 10-46 EARC\_RX\_CMDC\_STATUS6 0x2d**

Bit(s)	R/W	Default	Description
31	R	0	ro_idle2_int
30	R	0	ro_idle1_int
29	R	0	ro_disc2_int
28	R	0	ro_disc1_int
27	R	0	ro_earc_int
26	R	0	ro_hb_status_int
25	R	0	ro_losthb_int
24	R	0	ro_timeout_int
23	R	0	ro_status_ch_int
22	R	0	ro_int_rec_invalid_id
21	R	0	ro_int_rec_invalid_offset
20	R	0	ro_int_rec_unexp
19	R	0	ro_int_rec_ecc_err
18	R	0	ro_int_rec_parity_err
17	R	0	ro_int_recv_packet
16	R	0	ro_int_rec_time_out
15~0	R		reserved

### 10.2.5.2.2 eARC RX DMAC Registers

Baseaddr : 0xfe333c00 + offset\*4



**Table 10-49 EARCRX\_DMAC\_SYNC\_CTRL1 0x23**

Bit(s)	R/W	default	Description
31:19	RW	0	reserved
18	RW	0	i_earcrx_pll_dmac_valid_auto neg irq enable
17	RW	0	i_earcrx_pll_dmac_valid_auto stable state clear
16	RW	0	i_earcrx_pll_dmac_valid_auto stable check enable
15:0	RW	0	i_earcrx_pll_dmac_valid_auto stable set thread hold

**Table 10-50 EARCRX\_DMAC\_SYNC\_CTRL3 0x25**

Bit(s)	R/W	default	Description
31	RW	0	unstable_t0_err_clr
19	RW	0	unstable_t0_check_en
18:16	RW	0	unstable_t0_tick_sel
15:0	RW	0	unstable_t0_thd

**Table 10-51 EARCRX\_DMAC\_SYNC\_CTRL4 0x26**

Bit(s)	R/W	default	Description
31	RW	0	unstable_t1_err_clr
19	RW	0	unstable_t1_check_en
18:16	RW	0	unstable_t1_tick_sel
15:0	RW	0	unstable_t1_thd

**Table 10-52 EARCRX\_DMAC\_SYNC\_CTRL5 0x27**

Bit(s)	R/W	default	Description
31	RW	0	unstable_t2_err_clr
19	RW	0	unstable_t2_check_en
18:16	RW	0	unstable_t2_tick_sel
15:0	RW	0	unstable_t2_thd

**Table 10-53 EARCRX\_DMAC\_SYNC\_STAT1 0x28**

Bit(s)	R/W	default	Description
31	RO	0	r_unstatble_t0_check
15:0	RO	0	r_unstatble_t0_cnt



**Table 10-59 EARCRX\_SPDIFIN\_SAMPLE\_CTRL3 0x06**

Bit(s)	R/W	default	Description
31:24	RW	0	mode0 sample time
23:16	RW	0	mode1 sample time
15:8	RW	0	mode2 sample time
7:0	RW	0	mode3 sample time

**Table 10-60 EARCRX\_SPDIFIN\_SAMPLE\_CTRL4 0x07**

Bit(s)	R/W	default	Description
31:24	RW	0	mode4 sample time
23:16	RW	0	mode5 sample time
15:8	RW	0	mode6 sample time
7:0	RW	0	reserved

**Table 10-61 EARCRX\_SPDIFIN\_SAMPLE\_CTRL5 0x08**

Bit(s)	R/W	default	Description
31	RW	0	dmac_sqout filter enable
30	RW	0	dmac_sqout invert
29:27	RW	0	dmac_sqout filter tick select,0:sys_clk 1:1us 2:10us 3:100us 4:1ms
26:24	RW	0	dmac_sqout filter select
23:20	RW	0	reserved
19:0	RW	0	dmac_sqout filter tick

**Table 10-62 EARCRX\_SPDIFIN\_SAMPLE\_CTRL6 0x24**

Bit(s)	R/W	default	Description
31:18	RW	0	reserved
17	RW	0	Hold_tri_sample:hold sample after 000 or 111
16	RW	0	Sample_mode_filter_en:use average value of min/max width
15:8	RW	0	Stable cycle min percent :N/256
7:0	RW	0	Stable cycle max percent :N/256

**Table 10-63 EARCRX\_SPDIFIN\_SAMPLE\_STAT0 0x09**

Bit(s)	R/W	default	Description
31	RO	0	reserved
30:28	RO	0	sample mode
27:18	RO	0	min width timer

Bit(s)	R/W	default	Description
17: 8	RO	0	max width timer
7	RO	0	spdif_sqout buf 2
6	RO	0	spdif_sqout
5:0	RO	0	reserved

Table 10-64 EARCRX\_SPDIFIN\_SAMPLE\_STAT1 0x0a

Bit(s)	R/W	default	Description
31:27	RO	0	reserved
26:16	RO	0	r_width_min when debug_en valid
15:9	RO	0	reserved
10:0	RO	0	r_width_max when debug_en valid

Table 10-65 EARCRX\_SPDIFIN\_MUTE\_VAL 0x0b

Bit(s)	R/W	default	Description
31:0	RW	0	spdif in mute value

Table 10-66 EARCRX\_SPDIFIN\_CTRL0 0x0c

Bit(s)	R/W	default	Description
31	RW	0	spdifin work enable
30	RW	0	0 ch_num = 0~383 1 ch_num = 0~1
29:28	RW	0	reserved
27	RW	0	debug enable
26	RW	0	star add ch_cnt to ch_num
25	RW	0	papb check enable
24	RW	0	nonpcm2pcm_th enable
23:12	RW	0	if long time didn't detect PaPb again, will generate irq
11:8	RW	0	for stat1/stat2 select
7	RW	0	mute channel l
6	RW	0	mute channel r
5:4	RW	0	reserved
3	RW	0	valid check enable
2	RW	0	parity check enable
1	RW	0	spdif data invert
0	RW	0	reserved

**Table 10-67 EARCRX\_SPDIFIN\_CTRL1 0x0d**

Bit(s)	R/W	default	Description
31:24	RW	0	reserved
31:24	RW	0	internal irq status clear
23:12	RW	0	mute block check time thd
11:9	RW	0	mute block check tick sel
8	RW	0	ext 0 sync check for papb
7:0	RW	0	sync 0 mask

**Table 10-68 EARCRX\_SPDIFIN\_CTRL2 0x0e**

Bit(s)	R/W	default	Description
31:24	RW	0	mute bit in channel st
23:19	RW	0	mute min block number to declare
18	RW	0	mute bit in channel st L or R
17	RW	0	mute block number check enable
16	RW	0	auto clear compress mode when channel status not compress
15	RW	0	auto clear compress mode when nonpcm2pcm
14	RW	0	auto change earc/arc
13	RW	0	user l or r channel status to check papb
12	RW	0	0:data valid after 1 block;1: in 1st block if exit papb ,data valid after papb
11	RW	0	start write toddr 1:from papb check,0 from preamble Z, valid when reg_earcin_check_papb set
10	RW	0	auto reset will detect format change
9	RW	0	compress B pcpd select : 1:next 4th subframe data 0:next sub frame data
8:4	RW	0	papb msb position in data
3	RW	0	when in arc mode,spdif on force enable
2	RW	0	force value
1	RW	0	earc mode force enable
0	RW	0	force value

**Table 10-69 EARCRX\_SPDIFIN\_CTRL3 0x0f**

Bit(s)	R/W	default	Description
31:16	RW	0	earc mode pa value
15:0	RW	0	earc mode pb value

**Table 10-70 EARCRX\_SPDIFIN\_CTRL4 0x20**

Bit(s)	R/W	default	Description
31	RW	0	Reserved
30	RW	0	Add_ch_r : add right channel when stop at left channel send out
29	RW	0	reg_bc_val0_en,when auto add last R channle data, flag in chnum
28:20	RW	0	Unstable mask enable: 8:xz_err :when find z,subframe!=191 or findx subfram ==191 7:subframe cnt err2,subframe>384 in one block 6:subframe cnt err1,subframe!=384 in one block 5:cycle time err2:cycle time <= min 4:cycle time err2:cycle time >= max 3:parity err 2:bit_cnt err2:bit cnt > 64 in one subframe 1:bit_cnt err1:bit cnt !=64 in one subframe 0:xyz_err:frame structure error, xyz follow error
19:16	RW	0	Stable zcnt:stable after received N preamble Z
15:0	RW	0	Reserved

**Table 10-71 EARCRX\_SPDIFIN\_CTRL5 0x21**

Bit(s)	R/W	default	Description
31	RW	0	Stable timeout status clr
27:16	RW	0	Stable timeout check thd
14:12	RW	0	Stable timeout thd tick sel
11	RW	0	Stable timeout check en
8:0	RW	0	Unstable int mask

**Table 10-72 EARCRX\_SPDIFIN\_CTRL6 0x22**

Bit(s)	R/W	default	Description
31:17	RW	0	reserved
16	RW	0	Check time enable:count the bit number in set check time, can read the count value in STAT2
15:0	RW	0	Check time thd

**Table 10-73 EARCRX\_SPDIFIN\_STAT0 0x10**

Bit(s)	R/W	default	Description
31	RO	0	r_valid_bit[0]
30	RO	0	r_spdifin_cps
29	RO	0	r_spdif_out_valid_mask
28:19	RO	0	reserved
18	RO	0	r_spdif_lr_flag
17	RO	0	r_spdifin_v



Bit(s)	R/W	default	Description
16	RO	0	r_chst_mute
15:13	RO	0	r_data_rdy[2:0]
12	RO	0	c_spdifin_sel
11	RO	0	c_spdifin_valid_sel
10	RO	0	c_sample_stable_sel
9	RO	0	c_earc_mode
8	RO	0	r_spdifin_en
7	RO	0	r_dmacrx_en
6	RO	0	c_find_papb
5	RO	0	c_valid_change
4	RO	0	c_find_nonpcm2pcm
3	RO	0	c_pcpd_change
2	RO	0	c_ch_status_change
1	RO	0	i_sample_mode_change
0	RO	0	r_parity_err

Table 10-74 EARCRX\_SPDIFIN\_STAT1 0x11

Bit(s)	R/W	default	Description
31:0	RO	0	0: reg_spdifin_stat1 = r_ch_status_l[31:0]; 1: reg_spdifin_stat1 = r_ch_status_l[63:32]; 2: reg_spdifin_stat1 = r_ch_status_l[95:64]; 3: reg_spdifin_stat1 = r_ch_status_l[127:96]; 4: reg_spdifin_stat1 = r_ch_status_l[159:128]; 5: reg_spdifin_stat1 = r_ch_status_l[191:160]; 6: reg_spdifin_stat1 = {r_pc_data,r_pd_data}; 7: reg_spdifin_stat1 = 32'd0; 8: reg_spdifin_stat1 = r_ch_status_r[31:0]; 9: reg_spdifin_stat1 = r_ch_status_r[63:32]; 10: reg_spdifin_stat1 = r_ch_status_r[95:64]; 11: reg_spdifin_stat1 = r_ch_status_r[127:96]; 12: reg_spdifin_stat1 = r_ch_status_r[159:128]; 13: reg_spdifin_stat1 = r_ch_status_r[191:160]; 14: reg_spdifin_stat1 = {r_pc_data,r_pd_data}; 15: reg_spdifin_stat1 = 32'd0;

Table 10-75 EARCRX\_SPDIFIN\_STAT2 0x12

Bit(s)	R/W	default	Description
31:0	RW	0	0: r_z_width 1: {16'd0,r_frame_cnt_min,r_frame_cnt_max} 2: bit number in set check time

Table 10-76 EARCRX\_DMAC\_UBIT\_CTRL0 0x13

Bit(s)	R/W	default	Description
31	RW	0	dmac user bit decode enable
30:24	RW	0	iu sync value
23:16	RW	0	generate irq when fifo level pass some threthold
15	RW	0	max distance between IUs to set lost

Bit(s)	R/W	default	Description
14	RW	0	iu sync code enable 0 : all iu to fifo 1 only sync iu packet to fifo
13:12	RW	0	00 off 01 use l channel userbit 10 use r channel userbit 11 user lr channel userbit
11:8	RW	0	max distance bewteen IUs value
7	RW	0	fifo_thd irq enable
6	RW	0	when lost, initial fifo
5	RW	0	fifo initial
4:0	RW	0	user bit position in data

Table 10-77 EARCRX\_IU\_RDATA 0x14

Bit(s)	R/W	default	Description
31:8	RW	0	reserved
7:0	RW	0	iu data,read only

Table 10-78 EARCRX\_ERR\_CORRECT\_CTRL0 0x16

Bit(s)	R/W	default	Description
31	RW	0	err correct work enable
30	RW	0	reserved
29	RW	0	reset afifo out side
28	RW	0	reset afifo in side
27:7	RW	0	reserved
6	RW	0	bch output 16bit data msb is 27 or 19
5	RW	0	bch output data revers
4	RW	0	bch input ecc msb/lbs
3	RW	0	bch input ecc revers
2	RW	0	bch input data revers
1	RW	0	0 off 1 compress audio mode
0	RW	0	force work mode enable

Table 10-79 EARCRX\_ANA\_RST\_CTRL0 0x18

Bit(s)	R/W	default	Description
31	RW	0	analog reset check work enable
30	RW	0	analog reset from register enable
29	RW	0	soft reset value
28	RW	0	analog reset work enable 0: from bit31 1: from bit31 & top_work_en
27	RW	0	reserved
26:23	RW	0	when new format data in, hold reset after N posedge

Bit(s)	R/W	default	Description
22:20	RW	0	earcrx_div2 hold threshold tick select
19:0	RW	0	earcrx_div2 hold threshold

Table 10-80 EARCRX\_ANA\_RST\_CTRL1 0x19

Bit(s)	R/W	default	Description
31	RW	0	filter enable
30:28	RW	0	filter select
27:25	RW	0	filter tick sel,0:sys_clk 1:1us 2:10us 3:100us 4:1ms
24:16	RW	0	filter tick time
15	RW	0	filter enable
14:12	RW	0	filter select
11:9	RW	0	filter tick sel,0:sys_clk 1:1us 2:10us 3:100us 4:1ms
8:0	RW	0	filter tick time

### 10.2.5.2.3 eARC RX Top Registers

Baseaddr : 0xfe333e00 + offset\*4

Table 10-81 EARCRX\_TOP\_CTRL0 0x00

Bit(s)	R/W	default	Description
31:10	RW	0	reserved
9:8	RW	0	top debug select
7	RW	0	force spdif_rx_en to reg_spdif_rx_en_force_value
6	RW	0	value
5	RW	0	force spdif_rx_sqen to reg_spdif_rx_sqe
4	RW	0	value
3	RW	0	force dmacrx_en to reg_dmacrx_en_force_value
2	RW	0	value
1	RW	0	force dmacrx_sqen to reg_dmacrx_sqen_force_value
0	RW	0	value

Table 10-82 EARCRX\_DMAC\_INT\_MASK 0x01

Bit(s)	R/W	default	Description
31:30	RW	0	reserved
29:0	RW	0	dmac int mask

**Table 10-83 EARCRX\_DMAC\_INT\_PENDING 0x02**

Bit(s)	R/W	default	Description
31:30	RW	0	reserved
29:0	RW	0	dmac int pending, 29:biphase stable timeout int_set 28:dmac_pll_unstable 27:xz_err :when find z,subframe!=191 or findx subfram ==191 26:xyz_err:frame structure error,xyz follow error 25:bit_cnt_err1:bit cnt !=64 in one subframe 24:bit_cnt_err2:bit cnt > 64 in one subframe 23:parity_err 22:cycle time_err2:cycle time >= max 21:cycle time_err2:cycle time <= min 20:subframe cnt_err1,subframe!=384 in one block 19:subframe cnt_err2,subframe>384 in one block 18:err_corr_lr_err 17 :earcrx_ana_rst_c_new_format_set 16 :earcrx_ana_rst_c_earcrx_div2_hold_set 15 :earcrx_err_correct_c_bcherr_int_set 14 :earcrx_err_correct_r_afifo_overflow_set 13 :earcrx_err_correct_r_fifo_overflow_set 12 :earcrx_user_bit_check_r_fifo_overflow 11 :earcrx_user_bit_check_c_fifo_thd_pass 10 :earcrx_user_bit_check_c_u_pk_lost_int_set 9 :earcrx_user_bit_check_c_iu_pk_end 8 :earcrx_biphase_decode_c_chst_mute_clr 7 :earcrx_biphase_decode_c_find_papb 6 :earcrx_biphase_decode_c_valid_change 5 :earcrx_biphase_decode_c_find_nonpcm2pcm 4 :earcrx_biphase_decode_c_pcpd_change 3 :earcrx_biphase_decode_c_ch_status_change 2 :earcrx_biphase_decode_i_sample_mode_change 1 :earcrx_biphase_decode_r_parity_err 0 :earcrx_dmac_sync_afifo_overflow

**Table 10-84 EARCRX\_CMDC\_INT\_MASK 0x03**

Bit(s)	R/W	default	Description
31:16	RW	0	reserved
15:0	RW	0	cmdc int mask

**Table 10-85 EARCRX\_CMDC\_INT\_PENDING 0x04**

Bit(s)	R/W	default	Description
31:16	RW	0	reserved
15:0	RO	0	cmdc int pending, 15 : idle2_int 14 : idle1_int 13 : disc2_int 12 : disc1_int 11 : earc_int 10 : hb_status_int 9 : losthb_int 8 : timeout_int 7 : status_ch_int 6 : int_rec_invalid_id 5 : int_rec_invalid_offset 4 : int_rec_unexp 3 : int_rec_ecc_err 2 : int_rec_parity_err 1 : int_recv_packet 0 : int_rec_time_out

**Table 10-86 EARCRX\_ANA\_CTRL0 0x05**

Bit(s)	R/W	default	Description
31	RW	0	earcrx_en_d2a
30:29	RW	0	reserved
28:24	RW	0	earcrx_cmdcrx_reftrim
23:20	RW	0	earcrx_idr_trim
19:15	RW	0	earcrx_rterm_trim

Bit(s)	R/W	default	Description
14:12	RW	0	earcrx_cmdctx_ack_hystrim
11:7	RW	0	earcrx_cmdctx_ack_reftrim
6	RW	0	earcrx_cmdcrx_vrefon_sel
5:4	RW	0	earcrx_cmdcrx_rcfilter_sel
2:0	RW	0	earcrx_cmdcrx_hystrim

**Table 10-87 EARCRX\_ANA\_CTRL1 0x06**

Bit(s)	R/W	default	Description
23:16	RW	0	earcrx_reserv
15:12	RW	0	reserved
11	RW	0	earcrx_rxcom_on_sel
10	RW	0	earcrx_idc_sel
9	RW	0	earcrx_cmdcrx_spdif_dat_invsel
8	RW	0	earcrx_dmac_out_invsel
7:4	RW	0	earcrx_cmdcrx_spdif_sqcon
3:0	RW	0	earcrx_dmacrx_sqcon

**Table 10-88 EARCRX\_PLL\_CTRL0 0x08**

Bit(s)	R/W	default	Description
31:30	RW	0	reserved
29	RW	0	earcrx_pll_self_reset
28	RW	0	earcrx_pll_en
27:25	RW	0	reserved
24	RW	0	earcrx_pll_digital_rstn_sel
23	RW	0	earcrx_pll_dmacrx_sqout_rstn_sel
22:15	RW	0	reserved
14:10	RW	0	earcrx_pll_n
9:0	RW	0	reserved

**Table 10-89 EARCRX\_PLL\_CTRL1 0x09**

Bit(s)	R/W	default	Description
31	RW	0	earcrx_pll_afc_bypass
30:29	RW	0	reserved
28:24	RW	0	earcrx_pll_afc_in
23:22	RW	0	reserved

Bit(s)	R/W	default	Description
21:20	RW	0	earcrx_pll_adj_ldo
19	RW	0	reserved
18:16	RW	0	earcrx_pll_alpha
15:14	RW	0	reserved
13:12	RW	0	earcrx_pll_bb_mode
11	RW	0	reserved
10:8	RW	0	earcrx_pll_data_sel
7	RW	0	earcrx_pll_dco_clk_sel
6	RW	0	earcrx_pll_dco_m_en
5	RW	0	earcrx_pll_fast_lock
4	RW	0	earcrx_pll_filter_mode
3	RW	0	earcrx_pll_fix_en
2:0	RW	0	reserved

Table 10-90 EARCRX\_PLL\_CTRL2 0x0a

Bit(s)	R/W	default	Description
31:28	RW	0	earcrx_pll_filter_pvt1
27:24	RW	0	earcrx_pll_filter_pvt2
23	RW	0	reserved
22:20	RW	0	earcrx_pll_lambda0
19	RW	0	reserved
18:16	RW	0	earcrx_pll_lambda1
15:14	RW	0	reserved
13:8	RW	0	earcrx_pll_lk_s
7:4	RW	0	earcrx_pll_lk_w
3:2	RW	0	reserved
1:0	RW	0	earcrx_pll_lkw_sel

Table 10-91 EARCRX\_PLL\_CTRL3 0x0b

Bit(s)	R/W	default	Description
31	RW	0	earcrx_pll_lock_f
30	RW	0	reserved
29:28	RW	0	earcrx_pll_lock_long
27:26	RW	0	reserved
25:24	RW	0	earcrx_pll_pfd_gain
23:22	RW	0	reserved

Bit(s)	R/W	default	Description
21:20	RW	0	earcrx_pll_bias_adj
19	RW	0	reserved
18:16	RW	0	earcrx_pll_rou
15	RW	0	earcrx_pll_tdc_mode
14	RW	0	earcrx_pll_acq_range
13	RW	0	earcrx_pll_dco_sdm_en
12:6	RW	0	reserved
5:0	RW	0	earcrx_pll_reve

Table 10-92 EARCRX\_PLL\_STAT0 0x0c

Bit(s)	R/W	default	Description
31	RO	0	earcrx_pll_dmac_valid
30	RO	0	earcrx_pll_dmac_valid_auto
29	RO	0	earcrx_pll_afc_done_a2d
28:10	RO	0	reserved
9:0	RO	0	earcrx_pll_reg_out

## 10.3 Audio Output

### 10.3.1 Overview

This section describes TDM output interface, SPDIF output interface and audio REQ\_FRDDR sub-module.

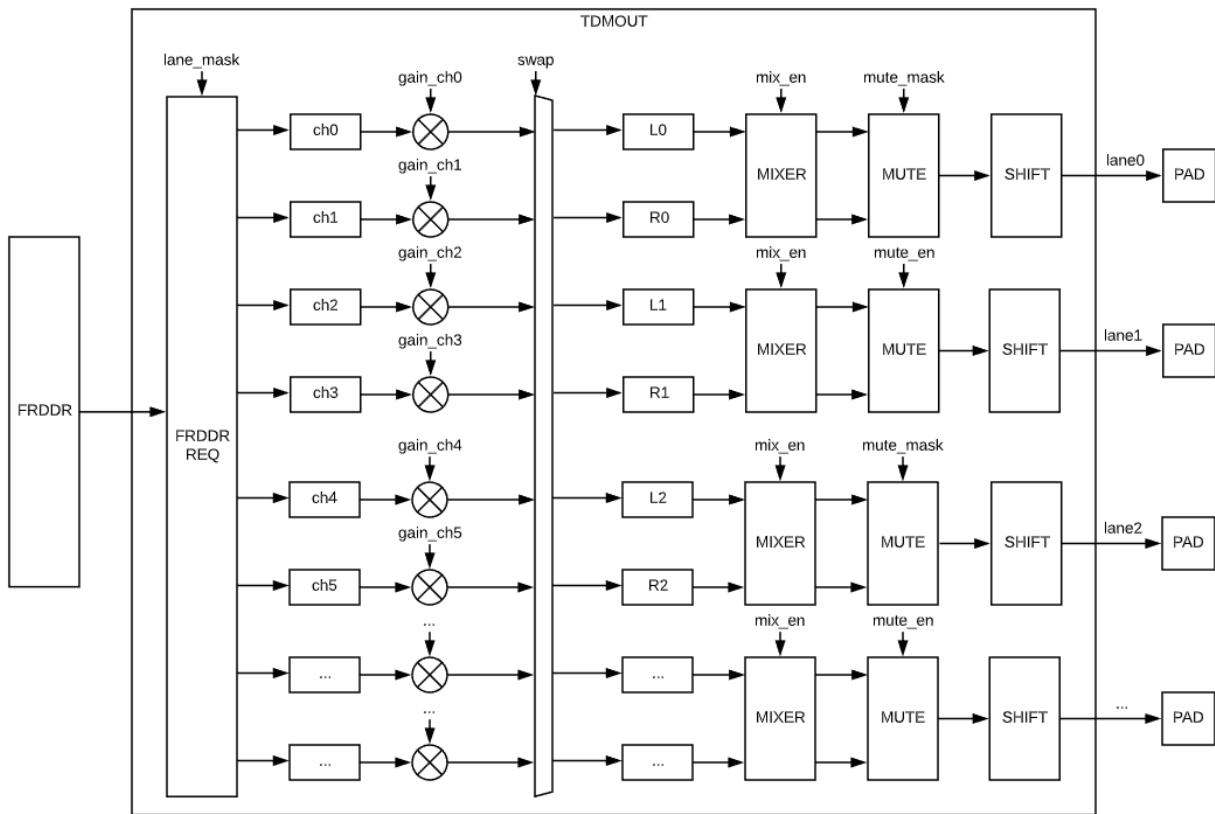
### 10.3.2 TDM Output Interface

TDM output interface works in the following way:

- All work at tdm\_sclk;
- Detect sample valid by tdm\_lrcclk rise edge and clear bit\_cnt/slot\_cnt;
- Request data from FRDDR and store to sample register;
- Swap sample register;
- Shift send out data;

Below is the diagram of TDM output interface.

Figure 10-10 TDM Output



### 10.3.3 SPDIF Output Interface

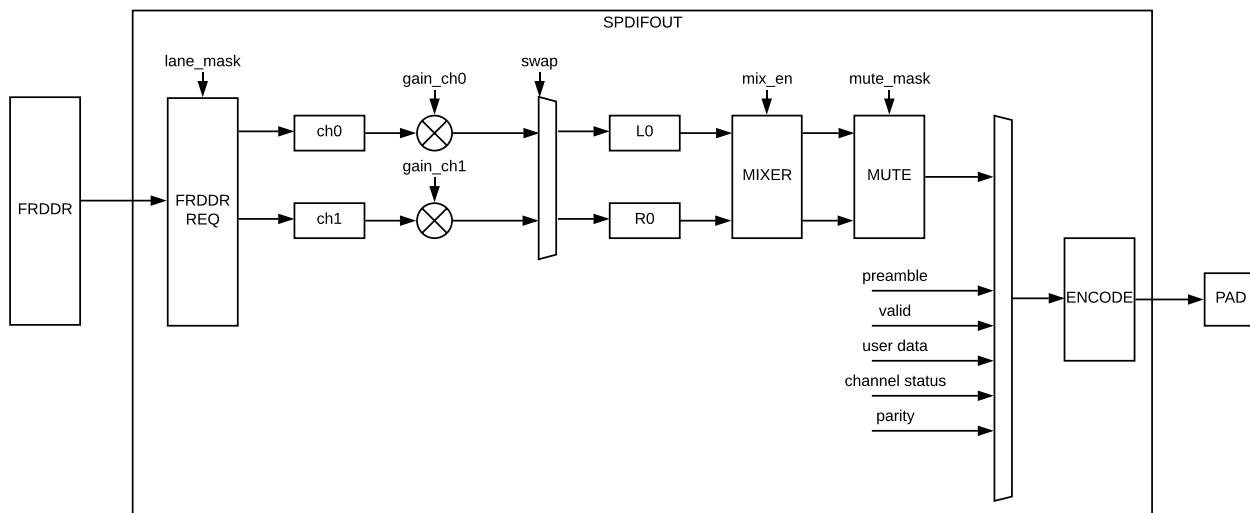
The SPDIF output interface works in the following way:

- All work at spdifout clk;
- Request data from FRDDR and store to sample L0/R0
- Select data by bit\_cnt/slot\_cnt and send out;

Below is the diagram of SPDIF output interface.



Figure 10-11 SPDIF Output Interface



T02FC20

The SPDIF (encode) add same source select. The SPDIF out can be the same with i2s 0/1/2/..., and can select before or after eq/drc.

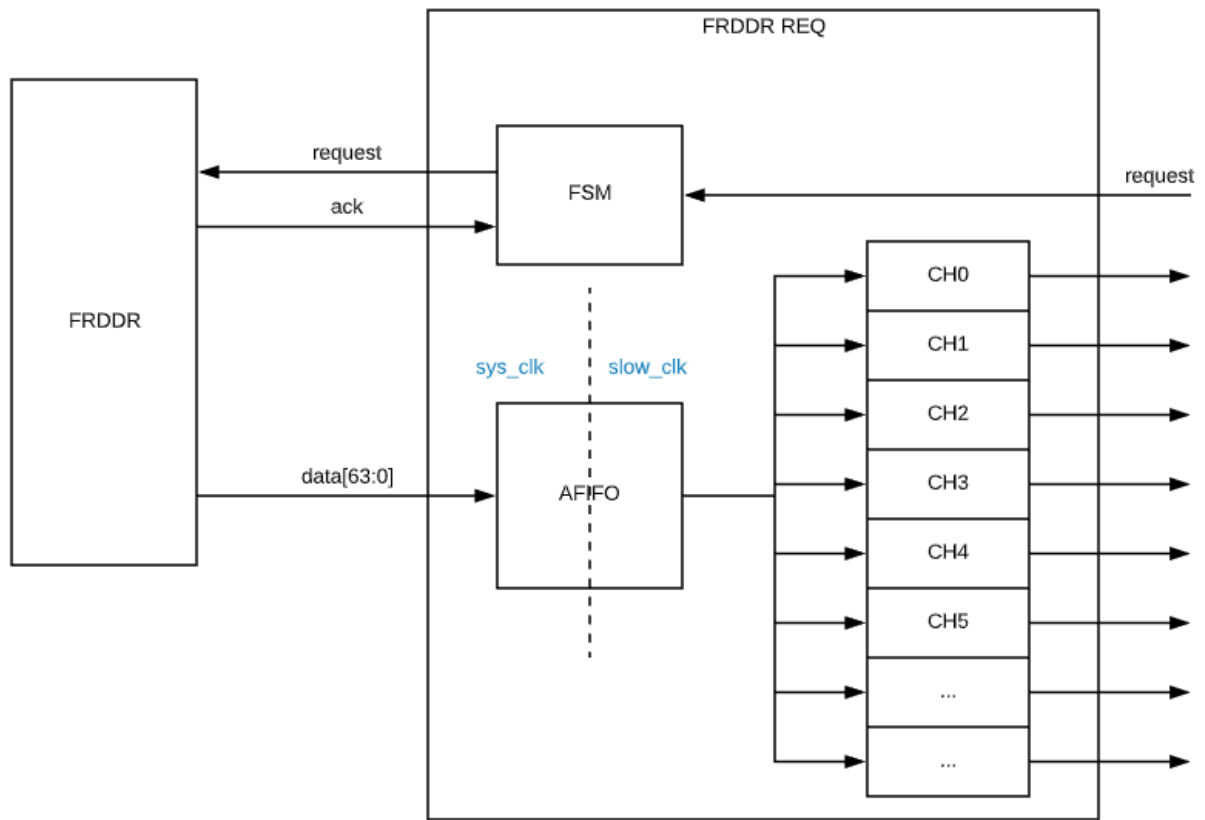
### 10.3.4 Audio REQ\_FRDDR

Audio Req\_Frddr transfers data from sys clock to out clock and change format in the following way;

- Wait for OUT enable and FRDDR initial done;
- Request first time and fill sample;
- When received update data, it will send request to FRDDR and update sample;

Below is a diagram of Audio REQ\_FRDDR.

Figure 10-12 Audio REQ\_FRDDR

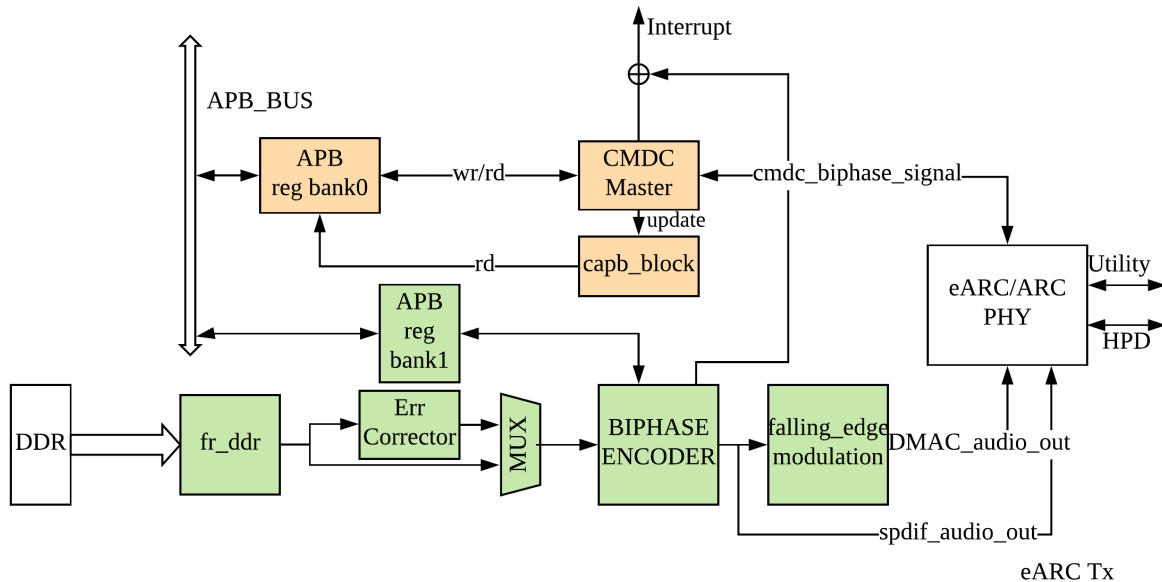


## 10.3.5 eARC TX

### 10.3.5.1 Overview

#### Structure

Figure 10-13 eARC TX Structure



#### eARC TX CMDC

The CMDC supports following features.

- Establish eARC mode, ARC mode, or mode switching between them.
- Status updates of eARC connection.
- Transmit audio channel delay information.
- Transmit eARC channel audio format compatibility information.

#### eARC TX DMAC

The DMAC supports both ARC and eARC functions.

In ARC mode, normal SPDIF signals can be transmitted.

In the eARC mode, after the audio data is taken out from the DDR, the error correction processing can be performed according to the register configuration; after the bi\_phase encoding, the falling\_edge modulation is also required.

In eARC mode, data can be transmitted via user bit.

In the eARC mode, the mute function is supported. You can manually turn Mute on or off. You can also Mute N blocks and then reset them to resume transmission or hold bus to switch audio formats.

A piece of invalid 01 data will be output before the transmission starts, to reduce the noise caused by the sudden change of the analog voltage.

### 10.3.5.2 Register Description

#### 10.3.5.2.1 EARC TX CMDC Registers

Baseaddr : 0xfe333000 + offset\*4

**Table 10-93 EARC\_TX\_CMDC\_TOP\_CTRL0 0x30**

Bit(s)	R/W	Default	Name
31	RW	0	idle2_int 1: enable
30	RW	0	idle1_int 1: enable
29	RW	0	disc2_int 1: enable
28	RW	0	disc1_int 1: enable
27	RW	0	earc_int 1: enable
26	RW	0	hb_status_int 1: enable
25	RW	0	losthb_int 1: enable
24	RW	0	timeout_int 1: enable
23	RW	0	status_ch_int 1: enable
22	RW	0	int_rcv_finished 1: enable
21	RW	0	int_rdata 1: enable
20	RW	0	int_rcv_nack 1: enable
19	RW	0	int_rcv_norsp 1: enable
18	RW	0	int_rcv_unexp 1: enable
17	RW	0	int_rcv_data 1: enable
16	RW	0	int_rcv_ack 1: enable
15	RW	0	int_rcv_ecc_err 1: enable
14	RW	0	int_rcv_packet 1: enable
13:0	RW		reserved 1: enable

**Table 10-94 EARC\_TX\_CMDC\_TOP\_CTRL1 0x31**

Bit(s)	R/W	Default	Name
31:0	RW		cmdc_top_ctrl1 0

**Table 10-95 EARC\_TX\_CMDC\_TOP\_CTRL2 0x32**

Bit(s)	R/W	Default	Name
31	RW	0	reset_idle2_int 1: enable
30	RW	0	reset_idle1_int 1: enable
29	RW	0	reset_disc2_int 1: enable
28	RW	0	reset_disc1_int 1: enable
27	RW	0	reset_earc_int 1: enable

Bit(s)	R/W	Default	Name
26	RW	0	reset_hb_status_int 1: enable
25	RW	0	reset_losthb_int 1: enable
24	RW	0	reset_timeout_int 1: enable
23	RW	0	reset_status_ch_int 1: enable
22	RW	0	reset_int_rcv_finished 1: enable
21	RW	0	reset_int_rdata 1: enable
20	RW	0	reset_int_rcv_nack 1: enable
19	RW	0	reset_int_rcv_norsp 1: enable
18	RW	0	reset_int_rcv_unexp 1: enable
17	RW	0	reset_int_rcv_data 1: enable
16	RW	0	reset_int_rcv_ack 1: enable
15	RW	0	reset_int_rcv_ecc_err 1: enable
14	RW	0	reset_int_rcv_packet 1: enable
13:00	RW		reserved

Table 10-96 EARC\_TX\_CMDC\_TIMER\_CTRL0 0x33

Bit(s)	R/W	Default	Name
31:0	RW		cmdc_timer_ctrl0 0

Table 10-97 EARC\_TX\_CMDC\_TIMER\_CTRL1 0x34

Bit(s)	R/W	Default	Name
31:0	RW		cmdc_timer_ctrl1 0

Table 10-98 EARC\_TX\_CMDC\_TIMER\_CTRL2 0x35

Bit(s)	R/W	Default	Name
31:0	RW		cmdc_timer_ctrl2 0

Table 10-99 EARC\_TX\_CMDC\_TIMER\_CTRL3 0x36

Bit(s)	R/W	Default	Name
31:0	RW		cmdc_timer_ctrl3 0

Table 10-100 EARC\_TX\_CMDC\_VSM\_CTRL0 0x37

Bit(s)	R/W	Default	Name
31	RW	0	sw_state_update
30:28	RW	0	sw_state

Bit(s)	R/W	Default	Name
27	RW	0	arc_initiated
26	RW	0	arc_terminated
25	RW	1	arc_enable
24	RW	0	man_hpd
23:22	RW	0	hpd_sel
21:20	RW	0	hpd_sel_earc
19	RW	0	comma_cnt_rst
18	RW	0	timeout_status_rst
17	RW	0	losthb_status_rst
16	RW	0	force_rst
15	RW	0	auto_state_en
14	RW	0	cmdc_state_en
13	RW	0	noack_repeat_en

Table 10-101 EARC\_TX\_CMDC\_VSM\_CTRL1 0x38

Bit(s)	R/W	Default	Name
8	RW	0	cntl_hpd_sel 0:sel hd_hpd,1:sel hdmirx_hpd
7:0	RW	0	comma_cnt_th should bigger than 3 and small than 10

Table 10-102 EARC\_TX\_CMDC\_VSM\_CTRL2 0x39

Bit(s)	R/W	Default	Name
11:8	RW		reserved disc1 hpd_val timing
7	RW	0	reg_soft_rst disc1 hpd_val timing
6:4	RW	0	time_sel disc1 hpd_val timing
3:2	RW	0	soft_rst_sel disc1 hpd_val timing
1:0	RW	0	enable_ctrl disc1 hpd_val timing

Table 10-103 EARC\_TX\_CMDC\_VSM\_CTRL3 0x3a

Bit(s)	R/W	Default	Name
11:8	RW		reserved disc2 heartbeat act timing
7	RW	0	reg_soft_rst

Bit(s)	R/W	Default	Name
			disc2 heartbeat act timing
6:4	RW	0	time_sel disc2 heartbeat act timing
3:2	RW	0	soft_rst_sel disc2 heartbeat act timing
1:0	RW	0	enable_ctrl disc2 heartbeat act timing

Table 10-104 EARC\_TX\_CMDC\_VSM\_CTRL4 0x3b

Bit(s)	R/W	Default	Name
31:12	RW	0	max_count_th in disc2, no heartbeat ack timing
11:8	RW		reserved
7	RW	0	reg_soft_rst in disc2, no heartbeat ack timing
6:4	RW	0	time_sel in disc2, no heartbeat ack timing
3:2	RW	0	soft_rst_sel in disc2, no heartbeat ack timing
1:0	RW	0	enable_ctrl in disc2, no heartbeat ack timing

Table 10-105 EARC\_TX\_CMDC\_VSM\_CTRL5 0x3c

Bit(s)	R/W	Default	Name
31:12	RW	0	max_count_th in disc1 and disc2 timing out
11:8	RW		reserved
7	RW	0	reg_soft_rst in disc1 and disc2 timing out
6:4	RW	0	time_sel in disc1 and disc2 timing out
3:2	RW	0	soft_rst_sel in disc1 and disc2 timing out
1:0	RW	0	enable_ctrl in disc1 and disc2 timing out

Table 10-106 EARC\_TX\_CMDC\_VSM\_CTRL6 0x3d

Bit(s)	R/W	Default	Name
11:8	RW		reserved
7	RW	0	reg_soft_rst in earc heartbeat act timing

Bit(s)	R/W	Default	Name
6:4	RW	0	time_sel in earc heartbeat act timing
3:2	RW	0	soft_rst_sel in earc heartbeat act timing
1:0	RW	0	enable_ctrl in earc heartbeat act timing

Table 10-107 EARC\_TX\_CMDC\_VSM\_CTRL7 0x3e

Bit(s)	R/W	Default	Name
15:8	RW	0	status_soft_val in earc heartbeat det timing
7	RW	0	reg_soft_rst in earc heartbeat det timing
6	RW	0	status_rst in earc heartbeat det timing
5:4	RW		reserved
3:2	RW	0	soft_rst_sel in earc heartbeat det timing
1:0	RW	0	enable_ctrl in earc heartbeat det timing

Table 10-108 EARC\_TX\_CMDC\_VSM\_CTRL8 0x3f

Bit(s)	R/W	Default	Name
31:0	RW	0	cmdc_vsm_ctrl8

Table 10-109 EARC\_TX\_CMDC\_VSM\_CTRL9 0x41

Bit(s)	R/W	Default	Name
31:0	RW	0	cmdc_vsm_ctrl9

Table 10-110 EARC\_TX\_CMDC\_SENDER\_CTRL0 0x42

Bit(s)	R/W	Default	Name
31:2		0	sender_ctrl0
1		1	hb_chg_conf_auto
0		0	hb_chg_auto



**Table 10-111 EARC\_TX\_CMDC\_PACKET\_CTRL0 0x43**

Bit(s)	R/W	Default	Name
31	RW	0	packet_mode_enable packet control
30	RW	0	free_enable packet control
29	RW	0	soft_rst packet control
28:24	RW	0	tx_ready_threshold packet control
23:20	RW		reserved
19:8	RW	0	send_pre_threshold packet control
5	RW	0	state_auto_en packet control
4	RW	0	sw_state_update_en packet control

**Table 10-112 EARC\_TX\_CMDC\_PACKET\_CTRL1 0x44**

Bit(s)	R/W	Default	Name
31	RW	0	ecc_endian send
30	RW	0	pre_start_value send
29:21	RW		reserved
20:16	RW	0	post_threshold send
15:14	RW		reserved
13:8	RW	0	pre_threshold

**Table 10-113 EARC\_TX\_CMDC\_PACKET\_CTRL2 0x45**

Bit(s)	R/W	Default	Name
31:0	RW	0	pre_flag

**Table 10-114 EARC\_TX\_CMDC\_PACKET\_CTRL3 0x46**

Bit(s)	R/W	Default	Name
31	RW	0	recv_en
30	RW	0	recv_parity_mask
29	RW	0	recv_timeout_en
28	RW	0	bch_ecc_en

Bit(s)	R/W	Default	Name
27:16	RW		reserved
15:0	RW	0	recv_timeout_threshold

**Table 10-115 EARC\_TX\_CMDC\_PACKET\_CTRL4 0x47**

Bit(s)	R/W	Default	Name
31:20	RW		reserved
19:0	RW	0	recv_packet_head

**Table 10-116 EARC\_TX\_CMDC\_PACKET\_CTRL5 0x48**

Bit(s)	R/W	Default	Name
31:20	RW		reserved
19:0	RW	0	recv_packet_head_mask

**Table 10-117 EARC\_TX\_CMDC\_PACKET\_CTRL6 0x49**

Bit(s)	R/W	Default	Name
31:20	RW	0	recv_pre_threshold packet control
19:7	RW		reserved
6	RW	0	recv_finished_int
5	RW	0	recv_ecc_err_int
4	RW	0	recv_ack_int
3	RW	0	recv_data_int
2	RW	0	recv_unexp_int
1	RW	0	recv_norosp_int
0	RW	0	recv_nack_int

**Table 10-118 EARC\_TX\_CMDC\_BIPHASE\_CTRL0 0x4a**

Bit(s)	R/W	Default	Name
31:24	RW		reserved
23:16	RW	0	ack delay threshold
15:10	RW		reserved
9	RW	0	send_ack_en
8	RW	0	sq_val_en
7	RW	0	biphase_send_soft_rst
6	RW	0	comma_soft_rst

Bit(s)	R/W	Default	Name
5	RW	0	fifo_rst
4	RW	0	receiver_no_sender
3	RW	0	sender_free
2	RW	0	receiver_send
1	RW	0	receiver_earc
0	RW	0	receiver_free

**Table 10-119 EARC\_TX\_CMDC\_BIPHASE\_CTRL1 0x4b**

Bit(s)	R/W	Default	Name
31:16	RW		reserved
15	RW	0	ack_enable send
14:8	RW		reserved
7:0	RW	0	wait_threshold before ack send

**Table 10-120 EARC\_TX\_CMDC\_BIPHASE\_CTRL2 0x4c**

Bit(s)	R/W	Default	Name
31	RW	0	comma_detection_enable comma detection
30	RW	0	manual_reset_enable comma detection
29	RW	0	manual_reset_value comma detection
28:16	RW		reserved
15:0	RW	0	comma_detection_threshold

**Table 10-121 EARC\_TX\_CMDC\_BIPHASE\_CTRL3 0x4d**

Bit(s)	R/W	Default	Name
31:0	RW	0	cmdc_biphase_ctrl3

**Table 10-122 EARC\_TX\_CMDC\_DEVICE\_ID\_CTRL 0x4e**

Bit(s)	R/W	Default	Name
31	RW	0	apb_write apb bus wr/read
30	RW	0	apb_read apb bus wr/read
29	RW	0	apb_rw_done apb bus wr/read

Bit(s)	R/W	Default	Name
28	RW	0	apb_rw_reset apb bus wr/read
27:17	RW		reserved
16	RW	1	hpb_rst_enable hpd rst enable
15:8	RW	0	apb_rwid apb bus wr/read
7:0	RW	0	apbrw_start_addr apb bus wr/read

Table 10-123 EARC\_TX\_CMDC\_DEVICE\_WDATA 0x4f

Bit(s)	R/W	Default	Name
31:8	RW		reserved
7:0	RW	0	apb_write_data apb bus wr/read

Table 10-124 EARC\_TX\_CMDC\_DEVICE\_RDATA 0x50

Bit(s)	R/W	Default	Name
31:8	RW		reserved
7:0	RW	0	apb_read_data apb bus wr/read

Table 10-125 EARC\_TX\_CMDC\_MASTER\_CTRL 0x51

Bit(s)	R/W	Default	Name
31	RW	0	master_cmd_rw 1 write 0 read
30	RW	0	master_hb_ignore 0:wait hb issued before pkt cmd, 1: pkt cmd issued immediately
29	RW	0	master_idle master status
28	RW	0	master_cmd_soft_rst
27:24	RW	0	hb_cmd_cal_th
23:16	RW	0	master_cmd_count cmd count -1
15:8	RW	0	master_cmd_id

Table 10-126 EARC\_TX\_ANA\_CTRL0 0x52

Bit(s)	R/W	Default	Name
31:0	RW		reg_earctx_ana_ctrl0

**Table 10-127 EARC\_TX\_ANA\_CTRL1 0x53**

Bit(s)	R/W	Default	Name
31:0	RW		reg_earctx_ana_ctrl1 0

**Table 10-128 EARC\_TX\_ANA\_CTRL2 0x54**

Bit(s)	R/W	Default	Name
31:0	RW		reg_earctx_ana_ctrl2 0

**Table 10-129 EARC\_TX\_ANA\_CTRL3 0x55**

Bit(s)	R/W	Default	Name
31:0	RW		reg_earctx_ana_ctrl3 0

**Table 10-130 EARC\_TX\_ANA\_CTRL4 0x56**

Bit(s)	R/W	Default	Name
31:0	RW		reg_earctx_ana_ctrl4 0

**Table 10-131 EARC\_TX\_ANA\_CTRL5 0x57**

Bit(s)	R/W	Default	Name
31:0	RW		reg_earctx_ana_ctrl5 0

**Table 10-132 EARC\_TX\_ANA\_STAT0 0x58**

Bit(s)	R/W	Default	Name
31:0	RO		ro_ANA_status0 0

**Table 10-133 EARC\_TX\_CMDC\_STATUS0 0x59**

Bit(s)	R/W	Default	Name
31:0	RO		ro_cmdc_status0 0

**Table 10-134 EARC\_TX\_CMDC\_STATUS1 0x5a**

Bit(s)	R/W	Default	Name
31:0	RO		ro_cmdc_status1 0

**Table 10-135 EARC\_TX\_CMDC\_STATUS2 0x5b**

Bit(s)	R/W	Default	Name
31:0	RO		ro_cmdc_status2 0

**Table 10-136 EARC\_TX\_CMDC\_STATUS3 0x5c**

Bit(s)	R/W	Default	Name
31:0	RO		ro_cmdc_status3 0

**Table 10-137 EARC\_TX\_CMDC\_STATUS4 0x5d**

Bit(s)	R/W	Default	Name
31:0	RO		ro_cmdc_status4 0

**Table 10-138 EARC\_TX\_CMDC\_STATUS5 0x5e**

Bit(s)	R/W	Default	Name
31:0	RO		ro_cmdc_status5 0

**Table 10-139 EARC\_TX\_CMDC\_STATUS6 0x5f**

Bit(s)	R/W	Default	Name
31	RO		ro_idle2_int
30	RO		ro_idle1_int
29	RO		ro_disc2_int
28	RO		ro_disc1_int
27	RO		ro_earc_int
26	RO		ro_hb_status_int
25	RO		ro_losthb_int
24	RO		ro_timeout_int
23	RO		ro_status_ch_int
22	RO		ro_int_rcv_finished
21	RO		ro_int_rdata
20	RO		ro_int_rcv_nack
19	RO		ro_int_rcv_norsp
18	RO		ro_int_rcv_unexp
17	RO		ro_int_rcv_data

Bit(s)	R/W	Default	Name
16	RO		ro_int_rcv_ack
15	RO		ro_int_rcv_ecc_err
14	RO		ro_int_rcv_packet
13:00			reserved

### 10.3.5.2.2 EARC TX DMAC Registers

Baseaddr : 0xfe333400 + offset\*4

**Table 10-140 EARCTX\_DMACE\_TOP\_CTRL0 0x00**

Bit(s)	R/W	default	Description
31	RW	0	reserved
30	RW	0	dmac top soft reset
29:26	RW	0	reserved
25:24	RW	0	dmac debug bus sel 3: fe_debug 2: err_debug
23:21	RW	0	reserved
20	RW	0	reg_slow_sync_scan_reg
19	RW	0	reg_fe_sf_scan_reg
18	RW	0	reg_fe_slow_sync_scan_reg
17	RW	0	reg_top_sf_scan_reg
16	RW	0	reg_top_slow_sync_scan_reg
15:4	RW	0	reserved
3	RW	0	spdif_tx_en force enable
2	RW	0	spdif_tx_en force value
1	RW	0	dmac_tx_en force enable
0	RW	0	dmac_tx_en force value

**Table 10-141 EARCTX\_MUTE\_VAL 0x01**

Bit(s)	R/W	default	Description
31:0	RW	0	when biahpase encode mute,the channel value,with reg_mute_l/reg_mute_r

**Table 10-142 EARCTX\_SPDIFOUT\_GAIN0 0x02**

Bit(s)	R/W	default	Description
31:24	RW	0	channel 3 gain
23:16	RW	0	channel 2 gain
15:8	RW	0	channel 1 gain
7:0	RW	0	channel 0 gain

**Table 10-143 EARCTX\_SPDIFOUT\_GAIN1 0x03**

Bit(s)	R/W	default	Description
31:24	RW	0	channel 7 gain
23:16	RW	0	channel 6 gain
15:8	RW	0	channel 5 gain
7:0	RW	0	channel 4 gain

**Table 10-144 EARCTX\_SPDIFOUT\_GAIN2 0x1e**

Bit(s)	R/W	default	Description
31:24	RW	0	channel 11 gain
23:16	RW	0	channel 10 gain
15:8	RW	0	channel 9 gain
7:0	RW	0	channel 8 gain

**Table 10-145 EARCTX\_SPDIFOUT\_GAIN3 0x1f**

Bit(s)	R/W	default	Description
31:24	RW	0	channel 15 gain
23:16	RW	0	channel 14 gain
15:8	RW	0	channel 13 gain
7:0	RW	0	channel 12 gain

**Table 10-146 EARCTX\_SPDIFOUT\_GAIN4 0x20**

Bit(s)	R/W	default	Description
31:16	RW	0	reserved
15:0	RW	0	Reqfrddr :reg_gain_en

**Table 10-147 EARCTX\_SPDIFOUT\_GAIN5 0x21**

Bit(s)	R/W	default	Description
31	RW	0	Reqfrddr :reg_gain_step_chg
30:26	RW	0	reserved
25:24	RW	0	Reqfrddr :reg_gain_val_sel
23:16	RW	0	Reqfrddr :reg_gain_step
15:0	RW	0	Reqfrddr :reg_gain_updt_timer



Table 10-148 EARCTX\_SPDIFOUT\_CTRL0 0x04

Bit(s)	R/W	default	Description
31	RW	0	biphase work start,pluse
30	RW	0	biphase work clear,pluse
29	RW	0	affo out reset
28	RW	0	affo in reset
27	RW	0	add delay to mathc TDM out when share buff
26	RW	0	user Bit select : 0 from reg_userdata_set 1 from data[29]
25	RW	0	value
24	RW	0	0 :from reg_chstst 1 : from data[30]
23	RW	0	reserved
22	RW	0	r channel mute ,with reg_mute_val
21	RW	0	l channel mute ,with reg_mute_val
20	RW	0	0 data from 31Bit 1 data from 27bit
19	RW	0	0 lsb first 1 msb first
18	RW	0	biphase encode valid Bit value sel : 0 from data 1 from reg_valid_set
17	RW	0	biphase encode valid Bit value
16	RW	0	when c_mute_hold_last_err_corr valid,clear work enable, initial biphase encode
15	RW	0	Req_frddr_rst
14:12	RW	0	reserved
0	RW	0	Bit 0 is initial parity value

Table 10-149 EARCTX\_SPDIFOUT\_CTRL1 0x05

Bit(s)	R/W	default	Description
31	RW	0	set 1 select eq_drc data
30	RW	0	Keep_req_ddr_init
30:28	RW	0	reserved
26:24	RW	0	from ddr selet
23:16	RW	0	wait some time when enable set to 1
15:13	RW	0	reserved
12:8	RW	0	msb position of data
7	RW	0	set 1 no need ack from frddr to transmit channel status
6:4	RW	0	reg_frddr_type 0: 8bit 1:16bit 2:16bit with shift 3:32bit 4 32bit with shift
3	RW	0	reserved
2	RW	0	Stat_sel: 1 cur_gain_val 0 debug singles
1:0	RW	0	reserved

**Table 10-150 EARCTX\_SPDIFOUT\_CTRL2 0x1d**

Bit(s)	R/W	default	Description
31:24	RW	0	reserved
23:16	RW	0	Req_frddr : reg_clr_by_init
15:0	RW	0	Reg_mask

**Table 10-151 EARCTX\_SPDIFOUT\_PREAMB 0x06**

Bit(s)	R/W	default	Description
31	RW	0	user 8'b11101000 1 user 7:0
30	RW	0	user 8'b11100100 1 user 15:8
29	RW	0	user 8'b11100010 1 user 23:16
28:24	RW	0	reserved
23:16	RW	0	value
15:8	RW	0	value
7:0	RW	0	value

**Table 10-152 EARCTX\_SPDIFOUT\_SWAP 0x07**

Bit(s)	R/W	default	Description
31:16	RW	0	hold start cnt ,valid when reg_hold_for_tdm set 1
15	RW	0	send 01 sequence some times after initial done from frddr set
14:0	RW	0	send 01 sequence time ,valid when reg_init_send_en set 1

**Table 10-153 EARCTX\_ERR\_CORRT\_CTRL0 0x08**

Bit(s)	R/W	default	Description
31:24	RW	0	reserved
23	RW	0	bch input data generate in 24bit data reverse
22	RW	0	bch output ecc reverse
21	RW	0	bch output data reverse
20	RW	0	bch output ecc position
19:17	RW	0	reserved
16	RW	0	fifo initial
15	RW	0	r channel select[3]
14	RW	0	l channel select[3]
13:12	RW	0	gain x 1/2/4/8
11	RW	0	l/r mix
10:8	RW	0	r channel select[2:0]
7	RW	0	reserved

Bit(s)	R/W	default	Description
6:4	RW	0	l channel select[2:0]
3:0	RW	0	iu transmit interval

**Table 10-154 EARCTX\_ERR\_CORRT\_CTRL1 0x09**

Bit(s)	R/W	default	Description
31	RW	0	iu data write enable,pluse,auto clr in reg.v
30:8	RW	0	reserved
7:0	RW	0	iu transmute data

**Table 10-155 EARCTX\_ERR\_CORRT\_CTRL2 0x0a**

Bit(s)	R/W	default	Description
31	RW	0	mute clear,pluse,auto clr in reg.v
30	RW	0	mute start,pluse,auto clr in reg.v
29:28	RW	0	reserved
27:16	RW	0	mute block number
15:8	RW	0	mute bit at channel statue which bit
7:3	RW	0	reserved
2	RW	0	mute data sel: 0 data 1 reg_mute_data_value
1:0	RW	0	0:always mute 1:mute block number and dis mute 2:mute bolck number and hold bus

**Table 10-156 EARCTX\_ERR\_CORRT\_CTRL3 0x0b**

Bit(s)	R/W	default	Description
31:30	RW	0	reserved
29	RW	0	bch generate enable
28:24	RW	0	bch data msb position in audio data
23:0	RW	0	mute value,only for audio data part

**Table 10-157 EARCTX\_ERR\_CORRT\_CTRL4 0x0c**

Bit(s)	R/W	default	Description
31	RW	0	user Bit lr swap
30	RW	0	l/r channel use same Bit user bit
29:25	RW	0	audio data msb position in input data
24	RW	0	audio data msb/lbs
23	RW	0	user Bit value
22	RW	0	valid Bit value

Bit(s)	R/W	default	Description
21	RW	0	data sel: 0 data 1 reg_mute_data_value
20:19	RW	0	userBit sel: 0 data 1 reg_value 2 fifo data
18	RW	0	validBit sel: 0 data 1 reg_value
17	RW	0	chanel status sel: 0 data 1 reg_value
16	RW	0	fifo_less_thd irq enable
15:8	RW	0	start transmit iu after fifo level greater than this value
7:0	RW	0	generate irq, when fifo level less than this value

**Table 10-158 EARCTX\_SPDIFOUT\_CHSTS0 0x0e**

Bit(s)	R/W	default	Description
31:0	RW	0	channel A status[31:0]

**Table 10-159 EARCTX\_SPDIFOUT\_CHSTS1 0x0f**

Bit(s)	R/W	default	Description
31:0	RW	0	channel A status[63:32]

**Table 10-160 EARCTX\_SPDIFOUT\_CHSTS2 0x10**

Bit(s)	R/W	default	Description
31:0	RW	0	channel A status[95:64]

**Table 10-161 EARCTX\_SPDIFOUT\_CHSTS3 0x11**

Bit(s)	R/W	default	Description
31:0	RW	0	channel A status[127:96]

**Table 10-162 EARCTX\_SPDIFOUT\_CHSTS4 0x12**

Bit(s)	R/W	default	Description
31:0	RW	0	channel A status[159:128]

**Table 10-163 EARCTX\_SPDIFOUT\_CHSTS5 0x13**

Bit(s)	R/W	default	Description
31:0	RW	0	channel A status[191:160]

**Table 10-164 EARCTX\_SPDIFOUT\_CHSTS6 0x14**

Bit(s)	R/W	default	Description
31:0	RW	0	channel B status[31:0]

**Table 10-165 EARCTX\_SPDIFOUT\_CHSTS7 0x15**

Bit(s)	R/W	default	Description
31:0	RW	0	channel B status[63:32]

**Table 10-166 EARCTX\_SPDIFOUT\_CHSTS8 0x16**

Bit(s)	R/W	default	Description
31:0	RW	0	channel B status[95:64]

**Table 10-167 EARCTX\_SPDIFOUT\_CHSTS9 0x17**

Bit(s)	R/W	default	Description
31:0	RW	0	channel B status[127:96]

**Table 10-168 EARCTX\_SPDIFOUT\_CHSTSA 0x18**

Bit(s)	R/W	default	Description
31:0	RW	0	channel B status[159:128]

**Table 10-169 EARCTX\_SPDIFOUT\_CHSTSB 0x19**

Bit(s)	R/W	default	Description
31:0	RW	0	channel B status[191:160]

**Table 10-170 EARCTX\_FE\_CTRL0 0x1a**

Bit(s)	R/W	default	Description
31	RW	0	reserved
30	RW	0	falling edge modulation work enable
29	RW	0	fe out invent
28	RW	0	hold min time enable
27	RW	0	0 auto clear hold at next valid 1 clear hold with reg_mute_hold_clr
26:24	RW	0	hold min time tick select
23:0	RW	0	hold min time

### 10.3.5.2.3 EARC TX Top Registers

Baseaddr : 0xfe333600 + offset\*4

**Table 10-171 EARCTX\_TOP\_CTRL0 0x00**

Bit(s)	R/W	default	Description
31:18	RW	0	reserved
17:16	RW	0	debug mux,3:analog debug 2:dmac debug 1:cmdc debug
15	RW	0	reg_slow_sync_scan_reg
14:12	RW	0	reserved
11	RW	0	hdmi_hpd invent
10	RW	0	hdmi_hpd mux = 3,register value
9:8	RW	0	hdmi_hpd mux ,3:register value 2:hdmi hdp2 1:hdmi hdp1 0:hdmi hdp0
7	RW	0	earctx_hd_hdp invent
6	RW	0	earctx_hd_hdp mux = 3,register value
5:4	RW	0	earctx_hd_hdp mux,3:register value 2:gpiow_5 1:gpiow_9 0:gpiow_1
3:2	RW	0	reserved
1	RW	0	force mode enale
0	RW	0	force mode value

**Table 10-172 EARCTX\_DMAC\_INT\_MASK 0x01**

Bit(s)	R/W	default	Description
31:6	RW	0	reserved
5:0	RW	0	dmac int mask

**Table 10-173 EARCTX\_DMAC\_INT\_PENDING 0x02**

Bit(s)	R/W	default	Description
31:5	RO	0	reserved
4:0	RO	0	dmac int pending,4 :earctx_fe_modulation c_hold_clr 3 :earctx_fe_modulation c_hold_start 2 :earctx_err_correct c_fifo_thd_less_pass 1 :earctx_err_correct r_fifo_overflow_set 0 :earctx_err_correct c_fifo_empty_set

**Table 10-174 EARCTX\_CMDC\_INT\_MASK 0x03**

Bit(s)	R/W	default	Description
31:18	RW	0	reserved
17:0	RW	0	cmdc int mask

**Table 10-175 EARCTX\_CMDC\_INT\_PENDING 0x04**

Bit(s)	R/W	default	Description
31:18	RO	0	reserved
17:0	RO	0	cmdc int pending, 17 : hpd_high_int 16 : hpd_low_int 15 : idle2_int 14 : idle1_int 13 : disc2_int 12 : disc1_int 11 : earc_int 10 : hb_status_int 9 : losthb_int 8 : timeout_int 7 : status_ch_int 6 : int_recv_finished 5 : int_recv_nack 4 : int_recv_norsp 3 : int_recv_unexp 2 : int_recv_data 1 : int_recv_ack 0 : int_recv_ecc_err

**Table 10-176 EARCTX\_ANA\_CTRL0 0x05**

Bit(s)	R/W	default	Description
31	RW	0	earctx_en_d2a
30	RW	0	earctx_cmdcrx_vrefon_sel
29:28	RW	0	earctx_cmdcrx_rcfilter_sel
27	RW	0	reserved
26:24	RW	0	earctx_cmdcrx_hystrim
23:20	RW	0	earctx_idr_trim
19:15	RW	0	reserved
16:12	RW	0	earctx_rterm_trim
11	RW	0	reserved
10:8	RW	0	earctx_dmac_slew_con
7:5	RW	0	earctx_cmdctx_ack_hystrim
4:0	RW	0	earctx_cmdctx_ack_reftrim

**Table 10-177 EARCTX\_ANA\_CTRL1 0x06**

Bit(s)	R/W	default	Description
23:16	RW	0	earctx_reg
15:14	RW	0	reserved
13	RW	0	earctx_arctx_comsel
12	RW	0	earctx_idc_sel
11	RW	0	earctx_arctx_selc
10	RW	0	earctx_arctx_seli
9	RW	0	earctx_cmdcrx_dat_invsel
8:4	RW	0	earctx_cmdcrx_reftrim
3:0	RW	0	earctx_cmdcrx_sqcon

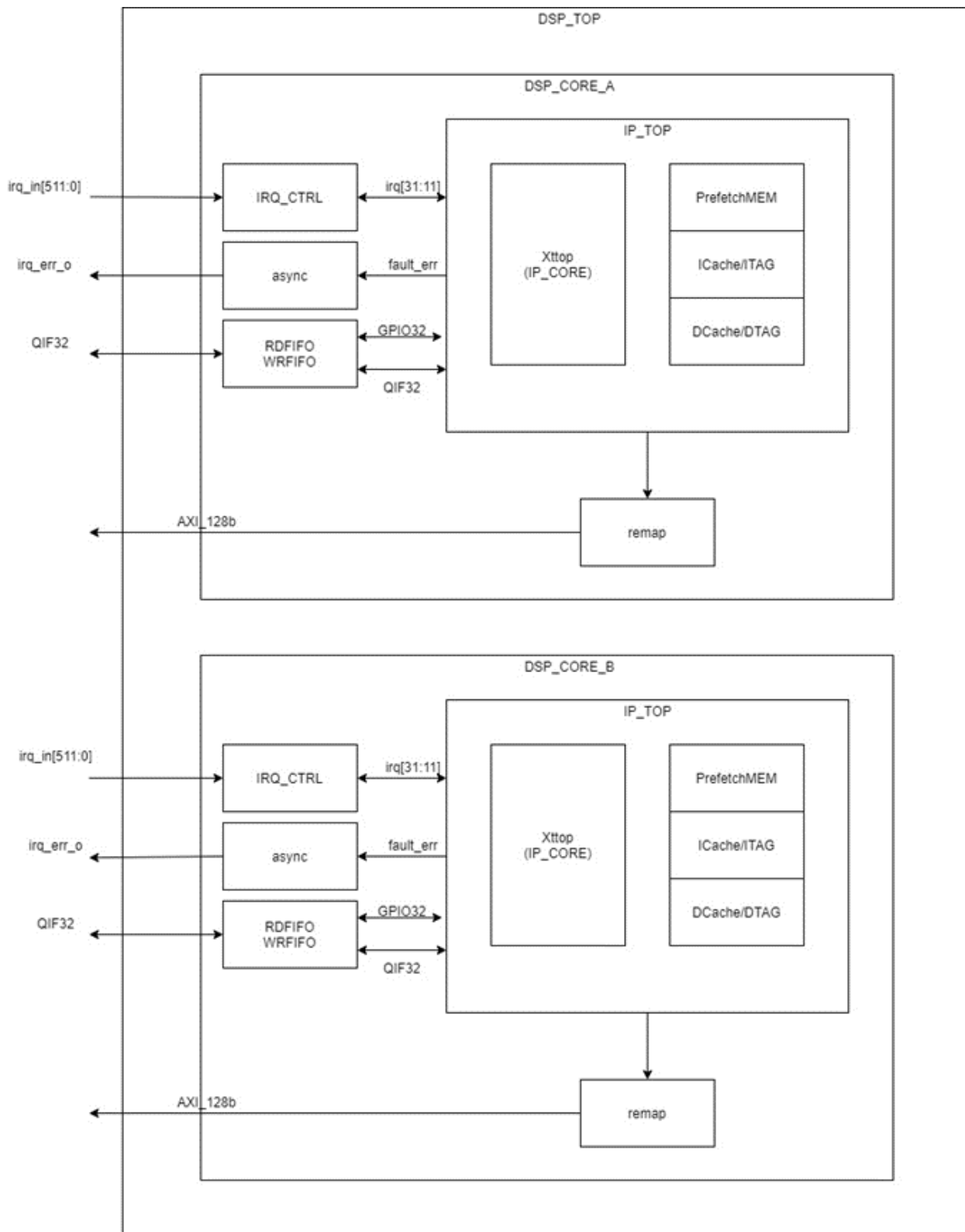
## 10.4 DSP

### 10.4.1 DSP\_TOP

- DSP\_TOP contains a DSP\_CORE\_A and DSP\_CORE\_B.
- Each DSP\_CORE contains 2 clocks, one SYSCLK and one DSP\_CLK;
- Each DSP\_CORE contains 2 resets, one for the debug part, and one for others;
- SYSCLK and reset for debug of the two DSP\_COREs are shared; DSP\_CLK and reset for others are two independent signals;
- Each DSP\_CORE has a separate APB bus for controlling and observing the contents of the DSP;
- Each DSP\_IP contains 21 external irqs, which are generated by IRQ\_CTRL.
- Each DSP\_IP has 2 faultERR outputs, which will be converted into irq and sent out;
- Each DSP\_CORE contains a set of RDFIFOs and a set of WRFIFOs that can be used to transmit data, especially between two DSP\_COREs;
- The AXI bus has three remap operations, corresponding to three kinds of step sizes;



Figure 10-14 DSP Path



### 10.4.2 DSP IRQ CTRL

Each IP\_TOP is configured with 32 irq sources, IRQ0~IRQ10 are IP's internal IRQ, and IRQ11~IRQ31 are from external.

For the 21 external irq sources:

- Select one from the 512 system level interrupt.
- The IRQ31 can be switched to audio\_irq. This audio IRQ has its own 20bit enable and statistic read only registers. And 20 bit clr\_sts registers. (these audio IRQ are also in the 512 system level interrupt).

**Figure 10-15 DSP IRQ CTRL Diagram**

**Table 10-178 DSP IRQ CTRL**

Interrupt NO.	Type	Level	Description
0	sw	1	DSP IP Internal
1	sw	1	DSP IP Internal
2	sw	2	DSP IP Internal
3	sw	3	DSP IP Internal
4	sw	3	DSP IP Internal
5	sw	4	DSP IP Internal
6	writeerr	3	DSP IP Internal
7	timer.0	1	DSP IP Internal
8	timer.1	3	DSP IP Internal
9	timer.2	4	DSP IP Internal
10	Profiling	1	DSP IP Internal
11	edge	1	External src 0
12	edge	1	External src 1
13	edge	1	External src 2
14	edge	2	External src 3
15	edge	2	External src 4
16	edge	2	External src 5
17	edge	2	External src 6
18	edge	2	External src 7
19	edge	2	External src 8
20	edge	2	External src 9
21	edge	2	External src 10
22	edge	2	External src 11
23	edge	2	External src 12
24	edge	2	External src 13
25	edge	3	External src 14
26	edge	3	External src 15
27	edge	3	External src 16

Interrupt NO.	Type	Level	Description
28	edge	3	External src 17
29	edge	4	External src 18
30	edge	2	External src 19
31	edge	3	External src 20 or External audio irq

There are also two irq outputs, which are the fault\_error and fault\_double\_error interfaces of the IP\_TOP output.

### 10.4.3 GPIO 32/QIF 32 Interface

The DSP comes in with GPIO32/QIF32 interface for transmitting data, you can use them to perform slow data transfer between 2 CPUs and 2 DSPs.

There're two kinds of solutions, as shown in the following figure.

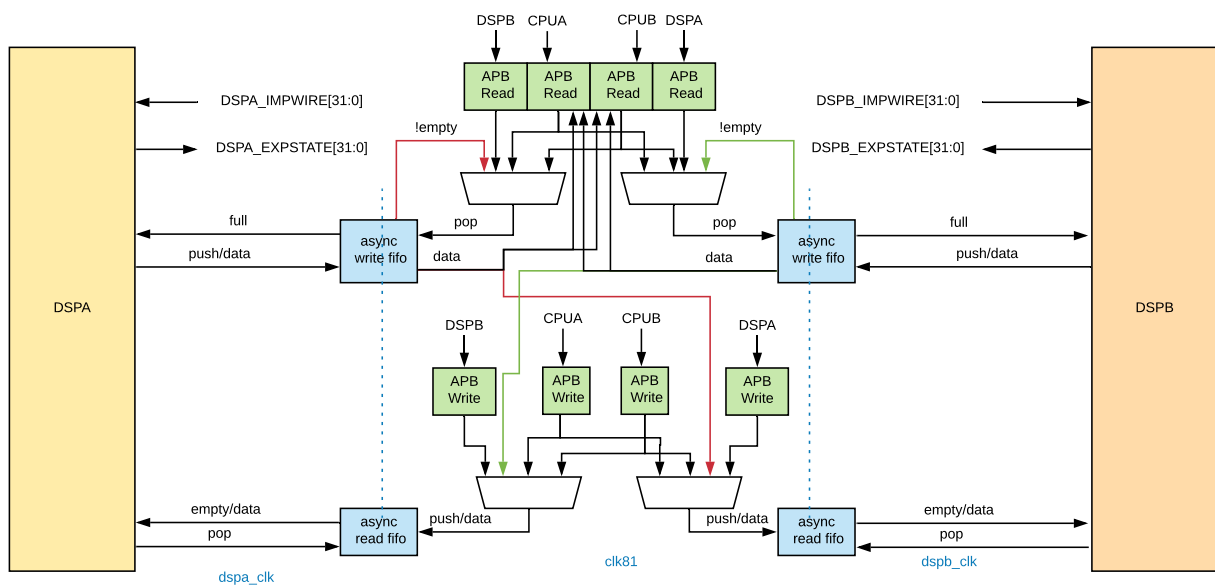
- (A) Black connections, CPUs and DSPs use the APB register to read and write the register access FIFO, and the DSPs use the QIF32 interfaces to access the FIFO to complete the data transmission.
- (B) Red or green connections, the two DSPs complete the data transmission directly through the QIF32 interfaces.

**Note**

The sel of the mux is controlled by the reg, and it can also be controlled by GPIO32(via the TIE\_EXPSTATE command), and the GPIO32 has a higher priority.

- TIE\_EXPSTATE[7]: force\_rdfifo\_sel;
- TIE\_EXPSTATE[5:4]: force\_rdfifo\_sel\_val;
- TIE\_EXPSTATE[3]: force\_wrfifo\_sel;
- TIE\_EXPSTATE[1:0]: force\_wrfifo\_sel\_val;

**Figure 10-16 GPIO 32/QIF 32 Interface Transmission**



T045T08

### 10.4.4 DSP Registers

Base address for following registers: 0xfe340000

**Table 10-179 DSP\_CFG0 0x00**

Bits	R/W	Default	Description
31	R/W	1	IP ports: DReset
30	R/W	1	IP ports: BReset
29	R/W	0	irq_clken
25:24	R/W	0	Dsp_irq_level_clr
23	R/W	0	clr_faultinfo
22	R/W	0	clr_double_irq_sts
15:0	R/W	0	IP ports: PRID

**Table 10-180 DSP\_IMPWARE 0x03**

Bits	R/W	Default	Description
31:0	R/W	0	IP ports: TIE_IMPWARE

**Table 10-181 DSP\_IRQ\_CTRL0 0x10**

Bits	R/W	Default	Description
31	R/W	0	Irq20_sel: for DSP_IRQ31, select audio IRQ.
19:0	R/W	0	audio_irq_mask; for DSP_IRQ31, each bit set 1 will enable audio irq src

**Table 10-182 DSP\_IRQ\_CTRL1 0x11**

Bits	R/W	Default	Description
19:0	R/W	0	audio_irq_clr; for DSP_IRQ31, each bit set 1 will clr audio irq sts

**Table 10-183 DSP\_IRQ\_CTRL2 0x12**

Bits	R/W	Default	Description
28:20	R/W	0	irq_sel; for DSP_IRQ13, from 512 interrupt sources
18:10	R/W	0	irq_sel; for DSP_IRQ12, from 512 interrupt sources
8:0	R/W	0	irq_sel; for DSP_IRQ11, from 512 interrupt sources

**Table 10-184 DSP\_IRQ\_CTRL3 0x13**

Bits	R/W	Default	Description
28:20	R/W	0	irq_sel; for DSP_IRQ16
18:10	R/W	0	irq_sel; for DSP_IRQ15
8:0	R/W	0	irq_sel; for DSP_IRQ14

**Table 10-185 DSP\_IRQ\_CTRL4 0x14**

Bits	R/W	Default	Description
28:20	R/W	0	irq_sel; for DSP_IRQ19
18:10	R/W	0	irq_sel; for DSP_IRQ18
8:0	R/W	0	irq_sel; for DSP_IRQ17

**Table 10-186 DSP\_IRQ\_CTRL5 0x15**

Bits	R/W	Default	Description
28:20	R/W	0	irq_sel; for DSP_IRQ22
18:10	R/W	0	irq_sel; for DSP_IRQ21
8:0	R/W	0	irq_sel; for DSP_IRQ20

**Table 10-187 DSP\_IRQ\_CTRL6 0x16**

Bits	R/W	Default	Description
28:20	R/W	0	irq_sel; for DSP_IRQ25
18:10	R/W	0	irq_sel; for DSP_IRQ24
8:0	R/W	0	irq_sel; for DSP_IRQ23

**Table 10-188 DSP\_IRQ\_CTRL7 0x17**

Bits	R/W	Default	Description
28:20	R/W	0	irq_sel; for DSP_IRQ28
18:10	R/W	0	irq_sel; for DSP_IRQ27
8:0	R/W	0	irq_sel; for DSP_IRQ26

**Table 10-189 DSP\_IRQ\_CTRL8 0x18**

Bits	R/W	Default	Description
28:20	R/W	0	irq_sel; for DSP_IRQ31
18:10	R/W	0	irq_sel; for DSP_IRQ30
8:0	R/W	0	irq_sel; for DSP_IRQ29

**Table 10-190 DSP\_IRQ\_STS 0x1F**

Bits	R/W	Default	Description
19:0	R	0	audio_irq_sts; for DSP_IRQ31;

**Table 10-191 DSP\_REMAP0 0x20**

Bits	R/W	Default	Description
31:16	R/W	0	remap0_replace[31:16];
15:0	R/W	0	remap0_match[31:16]

**Table 10-192 DSP\_REMAP1 0x21**

Bits	R/W	Default	Description
31:18	R/W	0	remap1_match[31:18]
15:2	R/W	0	remap1_replace[31:18]

**Table 10-193 DSP\_REMAP2 0x22**

Bits	R/W	Default	Description
31:20	R/W	0	remap2_match[31:20]
15:4	R/W	0	remap2_replace[31:20]

**Table 10-194 DSP\_STS0 0x40**

Bits	R/W	Default	Description
31:0	R/W	0	IP Ports: PFaultInfo[31:0]

**Table 10-195 DSP\_STS1 0x41**

Bits	R/W	Default	Description
31:0	R/W	0	IP Ports: TIE_EXPSTATE

**Table 10-196 DSP\_STS2 0x42**

Bits	R/W	Default	Description
31:0	R/W	0	reserved

**Table 10-197 DSP\_STS3 0x43**

Bits	R/W	Default	Description
31	R/W	0	IP Ports: DoubleExceptionError
30	R/W	0	IP Ports: PFatalError
29	R/W	0	IP Ports: PFaultInfoValid

Bits	R/W	Default	Description
28	R/W	0	IP Ports: XOCMode
27	R/W	0	IP Ports: DebugMode
26	R/W	0	IP Ports: BreakInAck
25	R/W	0	IP Ports: BreakOut
24	R/W	0	IP Ports: PWaitMode

**Table 10-198 DSP\_STS4 0x44**

Bits	R/W	Default	Description
31:0	R/W	0	PfaultInfo[63:32]

**Table 10-199 DSP\_STS5 0x45**

Bits	R/W	Default	Description
31:0	R/W	0	reserved

**Table 10-200 DSP\_QIF\_CTRL 0x80**

Bits	R/W	Default	Description
31	R/W	0	dsp_qif_en
30	R/W	0	qif_clk_en
27	R/W	0	rdfifo_rstn_in
26	R/W	0	rdfifo_rstn_out
25	R/W	0	wrfifo_rstn_in
24	R/W	0	wrfifo_rstn_out
23	R/W	0	rdfifo_clr_ov
22	R/W	0	wrfifo_clr_ov
5:4	R/W	0	rdfifo_sel 0:reg_frcpua; 1:reg_frcpub; 2:reg_frdsdp; 3:wrfifo_frdsdp;
1:0	R/W	0	wrfifo_sel 0:reg_tocpua; 1:reg_tocpub; 2:reg_todsp; 3:rdfifo_todsp;

**Table 10-201 DSP\_QIF\_STS 0x81**

Bits	R/W	Default	Description
27	R	0	dsp_rdfifo_outvalid
26	R	0	dsp_rdfifo_inready

Bits	R/W	Default	Description
25	R	0	dsp_rdfifo_overflow1
24	R	0	dsp_rdfifo_overflow2
20:16	R	0	dsp_rdfifo_cnt
10	R	0	dsp_wrfifo_outvalid
9	R	0	dsp_wrfifo_inready
8	R	0	dsp_wrfifo_overflow1
7	R	0	dsp_wrfifo_overflow2
4:0	R	0	dsp_wrfifo_cnt_sync

Table 10-202 DSP\_WRFIFO\_TOCPUA 0x82

Bits	R/W	Default	Description
31:0	R/W	0	pop one data from wrfifo when read

Table 10-203 DSP\_WRFIFO\_TOCPUB 0x83

Bits	R/W	Default	Description
31:0	R/W	0	pop one data from wrfifo when read

Table 10-204 DSP\_WRFIFO\_TODSP 0x84

Bits	R/W	Default	Description
31:0	R/W	0	pop one data from wrfifo when read

Table 10-205 DSP\_RDFIFO\_FRCPUA 0x88

Bits	R/W	Default	Description
31:0	R/W	0	push one data to rdfifo when write

Table 10-206 DSP\_RDFIFO\_FRCPUB 0x89

Bits	R/W	Default	Description
31:0	R/W	0	push one data to rdfifo when write

Table 10-207 DSP\_RDFIFO\_FRDSP 0x8A

Bits	R/W	Default	Description
31:0	R/W	0	push one data to rdfifo when write



**Table 10-208 DSP\_PM\_CTRL 0x90**

Bits	R/W	Default	Description
1	R/W	0	Select 1T latch of Pdebugenable.
0	R/W	0	Pdebugenable. Performance counter

**Table 10-209 DSP\_PDEBUGDATA\_STS 0x91**

Bits	R/W	Default	Description
31:0	R/W	0	pdebugdata

**Table 10-210 DSP\_PDEBUGINST\_STS 0x92**

Bits	R/W	Default	Description
31:0	R/W	0	pdebuginst

**Table 10-211 DSP\_PDEBUGLS0STAT\_STS 0x93**

Bits	R/W	Default	Description
31:0	R/W	0	Pdebugls0stat

**Table 10-212 DSP\_PDEBUGLS1STAT\_STS 0x94**

Bits	R/W	Default	Description
31:0	R/W	0	Pdebugls1stat

**Table 10-213 DSP\_PDEBUGOUTPIF\_STS 0x95**

Bits	R/W	Default	Description
31:0	R/W	0	Pdebugoutpif

**Table 10-214 DSP\_PDEBUGPC\_STS 0x96**

Bits	R/W	Default	Description
31:0	R/W	0	Pdebugpc_sts

**Table 10-215 DSP\_PDEBUGPREFETCHL1FILL\_STS 0x97**

Bits	R/W	Default	Description
31:0	R/W	0	Pdebugprefetchl1fill

**Table 10-216 DSP\_PDEBUGPREFETCHLOOKUP\_STS 0x98**

Bits	R/W	Default	Description
31:0	R/W	0	Pdebugprefetchlookup

**Table 10-217 DSP\_PDEBUGSTATUS\_STS 0x99**

Bits	R/W	Default	Description
31:0	R/W	0	Pdebugstatus

**Table 10-218 CHIP TOP DSP related IRQ sources**

IRQ NO.	IRQ source
351	reg_soft_irq[7]
350	reg_soft_irq[6]
349	reg_soft_irq[5]
348	reg_soft_irq[4]
347	reg_soft_irq[3]
346	reg_soft_irq[2]
345	reg_soft_irq[1]
344	reg_soft_irq[0]
343	1'b0
342	ddr1_phy_irq
341	ddr_phy_irq
340	eth_phy_irq_or
339	dmc1_sec_irq
338	dmc1_test_irq
337	dmc1_prot_irq
336	dmc1_mon_irq
335	dmc_sec_irq
334	dmc_test_irq
333	dmc_prot_irq
332	dmc_mon_irq
331	i2c_m_ao_b_irq
330	i2c_m_ao_a_irq
329	eth_qos_irq[9]
328	eth_qos_irq[8]
327	eth_qos_irq[7]
326	eth_qos_irq[6]

IRQ NO.	IRQ source
325	eth_qos_irq[5]
324	eth_qos_irq[4]
323	eth_qos_irq[3]
322	eth_qos_irq[2]
321	eth_qos_irq[1]
320	eth_qos_irq[0]
319	mipi_isp_irq[31]
318	mipi_isp_irq[30]
317	mipi_isp_irq[29]
316	mipi_isp_irq[28]
315	mipi_isp_irq[27]
314	mipi_isp_irq[26]
313	mipi_isp_irq[25]
312	mipi_isp_irq[24]
311	mipi_isp_irq[23]
310	mipi_isp_irq[22]
309	mipi_isp_irq[21]
308	mipi_isp_irq[20]
307	mipi_isp_irq[19]
306	mipi_isp_irq[18]
305	mipi_isp_irq[17]
304	mipi_isp_irq[16]
303	mipi_isp_irq[15]
302	mipi_isp_irq[14]
301	mipi_isp_irq[13]
300	mipi_isp_irq[12]
299	mipi_isp_irq[11]
298	mipi_isp_irq[10]
297	mipi_isp_irq[9]
296	mipi_isp_irq[8]
295	mipi_isp_irq[7]
294	mipi_isp_irq[6]
293	mipi_isp_irq[5]
292	mipi_isp_irq[4]
291	mipi_isp_irq[3]

IRQ NO.	IRQ source
290	mipi_isp_irq[2]
289	mipi_isp_irq[1]
288	mipi_isp_irq[0]
287	1'b0
286	1'b0
285	1'b0
284	ldim_done_int
283	venc1_vx1_int
282	venc0_vx1_int
281	usb_iddig_irq1
280	usb_vbusdig_irq1
279	a73_nIRQOUT[3]
278	a73_nIRQOUT[2]
277	a73_nIRQOUT[1]
276	a73_nIRQOUT[0]
275	a73_nFIQOUT[3]
274	a73_nFIQOUT[2]
273	a73_nFIQOUT[1]
272	a73_nFIQOUT[0]
271	1'b0
270	A73_STANDBYWFI2
269	A73_L2FLUSHDONE
268	A73IRQ[8]
267	A73IRQ[7]
266	A73IRQ[6]
265	A73IRQ[5]
264	A73IRQ[4]
263	A73IRQ[3]
262	A73IRQ[2]
261	A73IRQ[1]
260	A73IRQ[0]
259	A73_STANDBYWFI[3]
258	A73_STANDBYWFI[2]
257	A73_STANDBYWFI[1]
256	A73_STANDBYWFI[0]
255	assist_mbox_irq_ee[3]

IRQ NO.	IRQ source
254	assist_mbox_irq_ee[2]
253	assist_mbox_irq_ee[1]
252	assist_mbox_irq_ee[0]
251	mbox_irq[3]
250	mbox_irq[2]
249	mbox_irq[1]
248	mbox_irq[0]
247	nIRQOUT[3:0][3]
246	nIRQOUT[3:0][2]
245	nIRQOUT[3:0][1]
244	nIRQOUT[3:0][0]
243	nFIQOUT[3:0][3]
242	nFIQOUT[3:0][2]
241	nFIQOUT[3:0][1]
240	nFIQOUT[3:0][0]
239	mbox_irq[4]
238	STANDBYWFL2
237	L2FLUSHDONE
236	A53IRQ[8]
235	A53IRQ[7]
234	A53IRQ[6]
233	A53IRQ[5]
232	A53IRQ[4]
231	A53IRQ[3]
230	A53IRQ[2]
229	A53IRQ[1]
228	A53IRQ[0]
227	STANDBYWFI[3]
226	STANDBYWFI[2]
225	STANDBYWFI[1]
224	STANDBYWFI[0]
223	gdc_irq
222	edptx_int_1
221	edptx_int_0
220	vpu_sec_int

IRQ NO.	IRQ source
219	mute_irq
218	1'b0
217	ge2d_int
216	cusad_interrupt
215	rdma_done_int
214	vid1_wr_irq
213	vid0_wr_irq
212	vdin1_vsync_int
211	vdin1_hsync_int
210	vdin0_vsync_int
209	vdin0_hsync_int
208	viu1_mail_afbc_int[2]
207	lc_curve_int
206	vpu_crash_int
205	viu1_mail_afbc_int[1]
204	hdmitx_interrupt
203	deint_pre_irq
202	deint_post_irq
201	viff_empty_int_cpu
200	viu1_mail_afbc_int[0]
199	viu1_wm_int
198	viu1_dolby_int
197	viu1_vsync_int
196	viu1_hsync_int
195	viu1_line_n_int
194	viu2_vsync_int
193	viu2_hsync_int
192	viu2_line_n_int
191	spicc_5_int
190	spicc_4_int
189	spicc_3_int
188	anakin_irq[3]
187	anakin_irq[2]
186	anakin_irq[1]
185	anakin_irq[0]

IRQ NO.	IRQ source
184	spicc_1_int
183	spicc_0_int
182	spi_int
181	sar_adc_irq
180	cecb_irq
179	ceca_irq
178	sd_emmc_C_irq
177	sd_emmc_B_irq
176	sd_emmc_A_irq
175	nand_irq
174	smartcard_irq
173	uart_f_irq
172	uart_e_irq
171	uart_d_irq
170	uart_c_irq
169	uart_b_irq
168	uart_a_irq
167	spicc_2_int
166	i2c_s_a_irq
165	i2c_m_f_irq
164	i2c_m_e_irq
163	i2c_m_d_irq
162	i2c_m_c_irq
161	i2c_m_b_irq
160	i2c_m_a_irq
159	dspb_error_irq[1]
158	dspb_error_irq[0]
157	dspa_error_irq[1]
156	dspa_error_irq[0]
155	ts_demux_irq
154	mali_irq_pp3
153	mali_irq_ppmmu2
152	mali_irq_pp2
151	mali_irq_ppmmu1
150	mali_irq_pp1

IRQ NO.	IRQ source
149	mali_irq_ppmmu0
148	mali_irq_pp0
147	mali_irq_pmu
146	mali_irq_pp
145	mali_irq_gpmmu
144	mali_irq_gp
143	pcie_A_irq[7]
142	pcie_A_irq[6]
141	pcie_A_irq[5]
140	pcie_A_irq[4]
139	pcie_A_irq[3]
138	pcie_A_irq[2]
137	pcie_A_irq[1]
136	pcie_A_irq[0]
135	pcie_A_edma_int[1]
134	pcie_A_edma_int[0]
133	sectop_irq[1]
132	sectop_irq[0]
131	u2drd_interrupt
130	u3drd_interrupt
129	usb_iddig_irq0
128	usb_vbusdig_irq0
127	amrisc_int[31]
126	amrisc_int[30]
125	amrisc_int[29]
124	amrisc_int[28]
123	amrisc_int[27]
122	amrisc_int[26]
121	amrisc_int[25]
120	amrisc_int[24]
119	amrisc_int[23]
118	amrisc_int[22]
117	amrisc_int[21]
116	amrisc_int[20]
115	amrisc_int[19]



IRQ NO.	IRQ source
114	amrisc_int[18]
113	amrisc_int[17]
112	amrisc_int[16]
111	amrisc_int[15]
110	amrisc_int[14]
109	amrisc_int[13]
108	amrisc_int[12]
107	amrisc_int[11]
106	amrisc_int[10]
105	amrisc_int[9]
104	amrisc_int[8]
103	amrisc_int[7]
102	amrisc_int[6]
101	amrisc_int[5]
100	amrisc_int[4]
99	amrisc_int[3]
98	amrisc_int[2]
97	amrisc_int[1]
96	amrisc_int[0]
95	wave521_vpu_idle_irq
94	wave521_irq
93	dos_mbox_slow_irq[2]
92	dos_mbox_slow_irq[1]
91	dos_mbox_slow_irq[0]
90	demod_irq
89	sp_mbox_irq[1]
88	sp_mbox_irq[0]
87	mipi_dsi1_phy_irq
86	ts_irq_hevc
85	ts_irq_vpu
84	ts_irq_nna
83	ts_irq_gpu
82	viu3_vsync_int
81	viu3_hsync_int
80	viu3_line_n_int

IRQ NO.	IRQ source
79	cci_event_irq
78	cci_irq
77	aucpu_irq
76	eth_pmt_intr_o
75	mipi_dsi0_phy_irq
74	eth_gmac_int
73	eth_lpi_intr_o
72	eth_phy_irq[8]
71	eth_phy_irq[7]
70	eth_phy_irq[6]
69	eth_phy_irq[5]
68	eth_phy_irq[4]
67	eth_phy_irq[3]
66	eth_phy_irq[2]
65	eth_phy_irq[1]
64	eth_phy_irq[0]
63	sec_watchdog_irq
62	1'b0
61	1'b0
60	vdin2_vsync_int
59	dwap_irq
58	hdmirx_audmeas_int
57	hdmirx_interrupt
56	mipi_dsi_a_err_intr
55	mipi_dsi_a_intr
54	mipi_dsi_b_err_intr
53	mipi_dsi_b_intr
52	c_pwr_ctrl_irq
51	audio_irq[19]
50	audio_irq[18]
49	audio_irq[17]
48	audio_irq[16]
47	audio_irq[15]
46	audio_irq[14]
45	audio_irq[13]

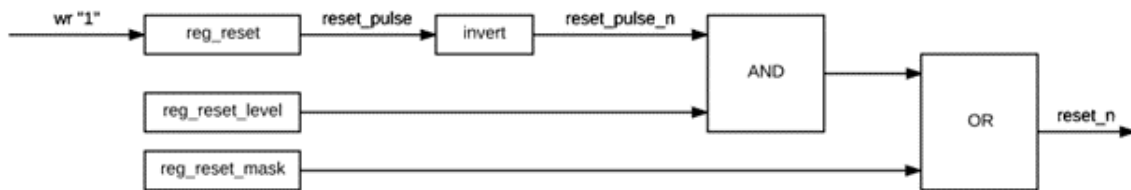
IRQ NO.	IRQ source
44	audio_irq[12]
43	audio_irq[11]
42	audio_irq[10]
41	audio_irq[9]
40	audio_irq[8]
39	audio_irq[7]
38	audio_irq[6]
37	audio_irq[5]
36	audio_irq[4]
35	audio_irq[3]
34	audio_irq[2]
33	audio_irq[1]
32	audio_irq[0]
31	ts_irq_a73
30	ts_irq_a53
29	dma_irq[5]
28	dma_irq[4]
27	dma_irq[3]
26	dma_irq[2]
25	dma_irq[1]
24	dma_irq[0]
23	ir_tx_irq
22	ir_rx_irq
21	gpio_irq[11]
20	gpio_irq[10]
19	gpio_irq[9]
18	gpio_irq[8]
17	gpio_irq[7]
16	gpio_irq[6]
15	gpio_irq[5]
14	gpio_irq[4]
13	gpio_irq[3]
12	gpio_irq[2]
11	gpio_irq[1]
10	gpio_irq[0]

IRQ NO.	IRQ source
9	watchdog_irq
8	sec_timerA_irq
7	timerJ_irq
6	timerI_irq
5	timerH_irq
4	timerG_irq
3	timerD_irq
2	timerC_irq
1	timerB_irq
0	timerA_irq

TOP DSP related reset registers.

- each reset includes 3 control register: reg\_reset, reg\_reset\_level, reg\_reset\_mask;
- when write “1” to reg\_reset, it will generate a signal: reset\_pulse;
- most of module’s reset is negative, so reset\_pulse will invert to reset\_pulse\_n;
- the usage of reg\_reset\_level is hold reset\_pulse\_n to “0”;
- the usage of reg\_reset\_mask is hold reset\_pulse\_n to “1”;

Figure 10-17 TOP DSP Reset



dspa normal reset bit11

dpsb normal reset bit10

RESETCTRL\_RESET1 0xfe002004

RESETCTRL\_RESET1\_LEVEL 0xfe002044

RESETCTRL\_RESET1\_MASK 0xfe002084

CHIP TOP DSP related GPIOs

Table 10-219

GPIOC_0	JTAG_B_TDO(oe)
GPIOC_1	JTAG_B_TDI(in)
GPIOC_4	JTAG_B_CLK(in)

GPIOC_5	JTAG_B_TMS(in)
GPIOD_6	JTAG_A_CLK(in)
GPIOD_7	JTAG_A_TMS(in)
GPIOD_8	JTAG_A_TDI(in)
GPIOD_9	JTAG_A_TDO(oe)

## 10.5 DDR Datapath

### 10.5.1 Overview

This part describes the datapath between audio module and DDR.

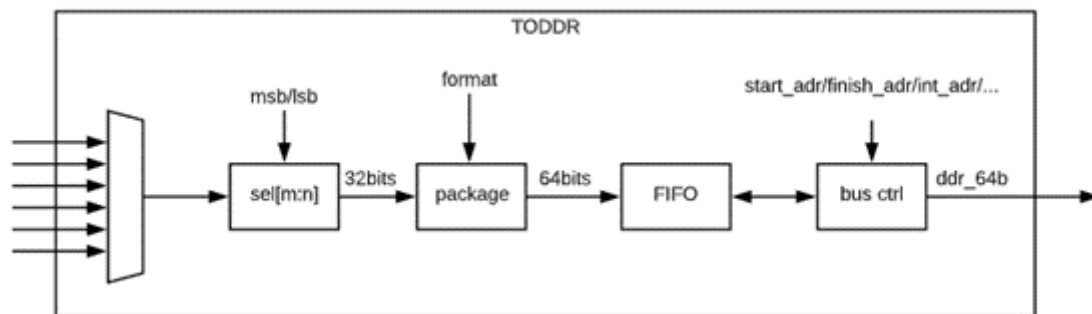
### 10.5.2 Audio TODDR

The SoC has 5 TODDR(FIFO), TODDR\_A's FIFO depth is 4096 x 64; B/C/D/E are 128 x 64. TODDR module works in the following way:

- All TODDR work at sysclk;
- Resample if need;
- Change format and package to 64 bits data by configuration;
- Write to fifo;
- Read data from fifo and send to DDR automatically by configuration;

Below is the diagram of Audio TODDR.

**Figure 10-18 Audio TODDR**



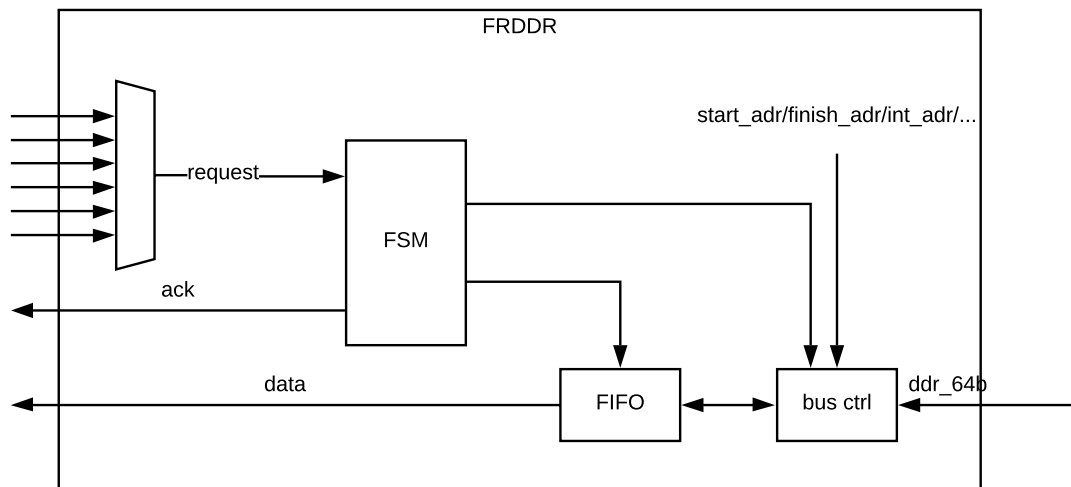
### 10.5.3 Audio FRDDR

The SoC has 5 FRDDR(FIFO), FRDDR\_A's FIFO depth is 256 x 64, B/C/D/E are 128 x 64;

- All FRDDR work at sysclk;
- When enable FRDDR, it will fill FIFO from DDR first;
- When FRDDR receive request, it will read data from fifo and send out;
- FRDDR will fill FIFO automatically by configuration;

Below is the Diagram of Audio FRDDR.

**Figure 10-19 Audio FRDDR**



T02FC23

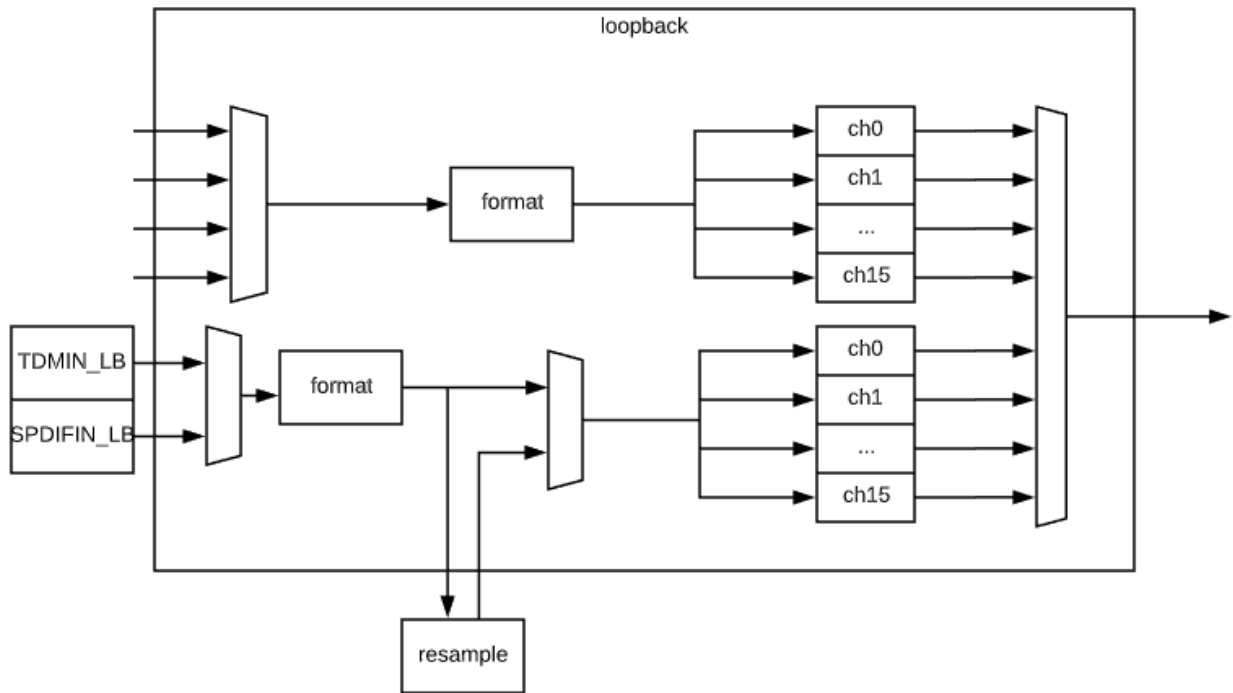
## 10.6 Audio Loopback

TDMIN\_LB can receive one TDMOUT, it can merge two sources in the following way:

- Store one source to temp register;
- When another source arrived, send it out directly;
- When finished, send out temp register;

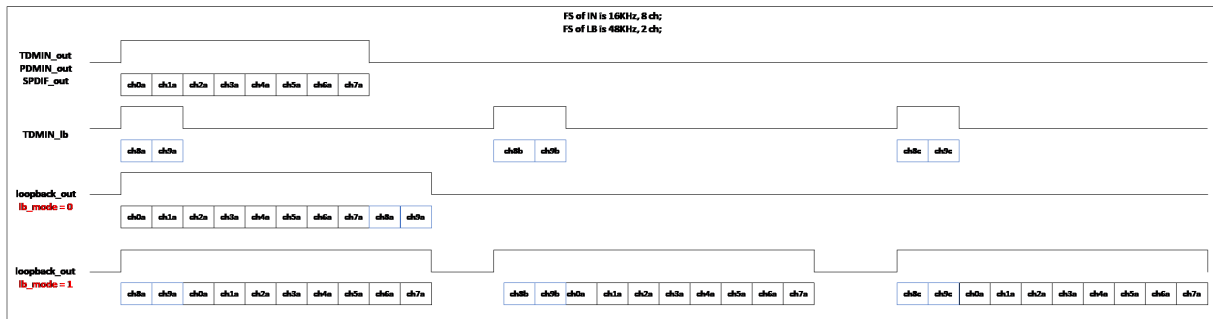
Below is the diagram of audio loopback datapath.

Figure 10-20 Audio Loopback Datapath



Audio loopback wave form is shown in the figure below.

Figure 10-21 Audio Loopback Wave Form

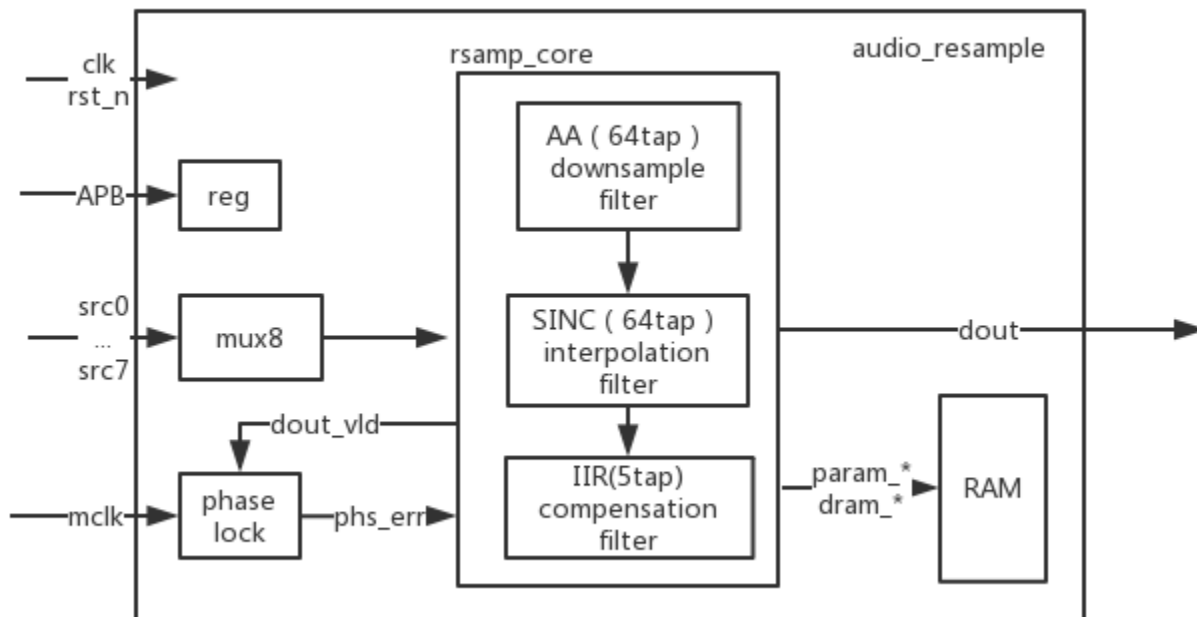


### 10.7 Audio Resample A/B

Audio resample changes data from one symbol rate to another symbol rate in the following steps:

1. Max 8 audio sources is supported, should be muxed from 1 source to audio\_resample core;
2. MSB/LSB selection is supported after source mux;
3. resample core performs the symbol rate convert, max 8/32 channel is supported.;
  - a. If down sample is needed, aa filter should be on, max 64 tap is supported. 1/2 and 1/4 down sample is supported;
  - b. Audio re-sample use sinc filter, max 64tap, 1/2^28 phase precision interpolation;
  - c. IIR compensation filter is used for compensate high-frequency damping;
4. Phase lock module is used for input-clock and output-clock symbol rate synchronization.

Figure 10-22 Audio Resample

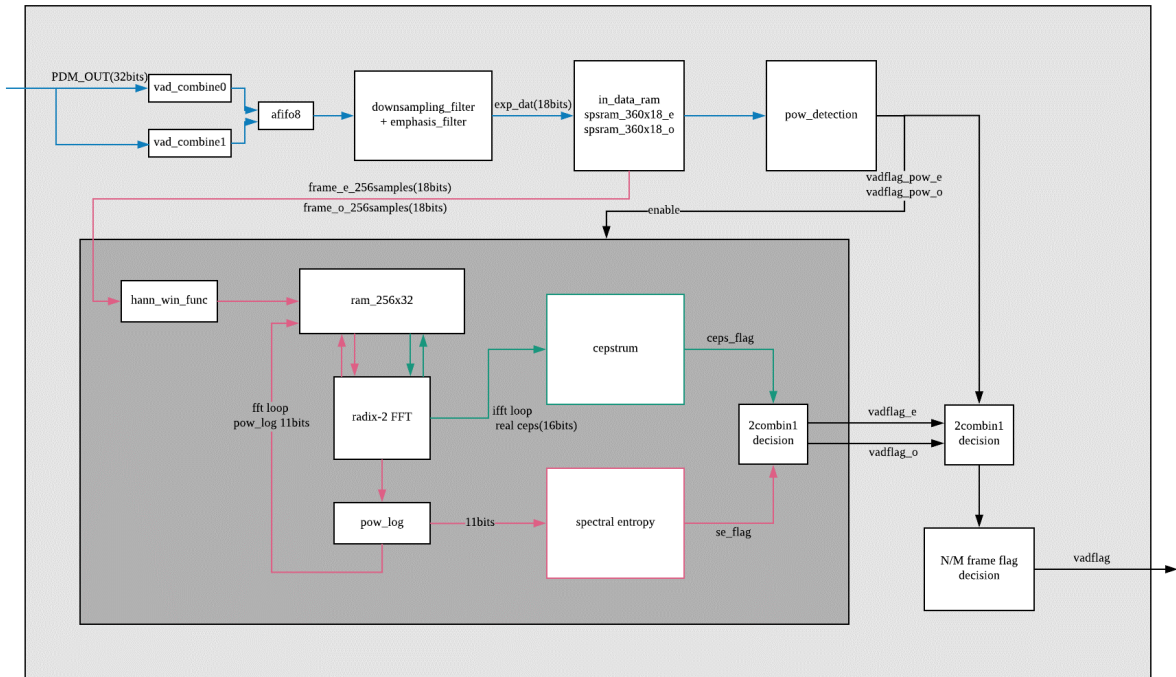


## 10.8 Audio VAD

VAD will wake up DSP when it determines if voice is present in a particular audio signal. It supports 2 channels audio streams searching which combined from PDM filter output audio stream. The audio stream will be down-sampled to 8K/s rate and de-noised before detection processing. VAD can provide power detection, cepstral detection and spectral entropy detection. It will do the detection per 10ms. It contains one 256p FFT function for cepstral and spectral entropy calculation.



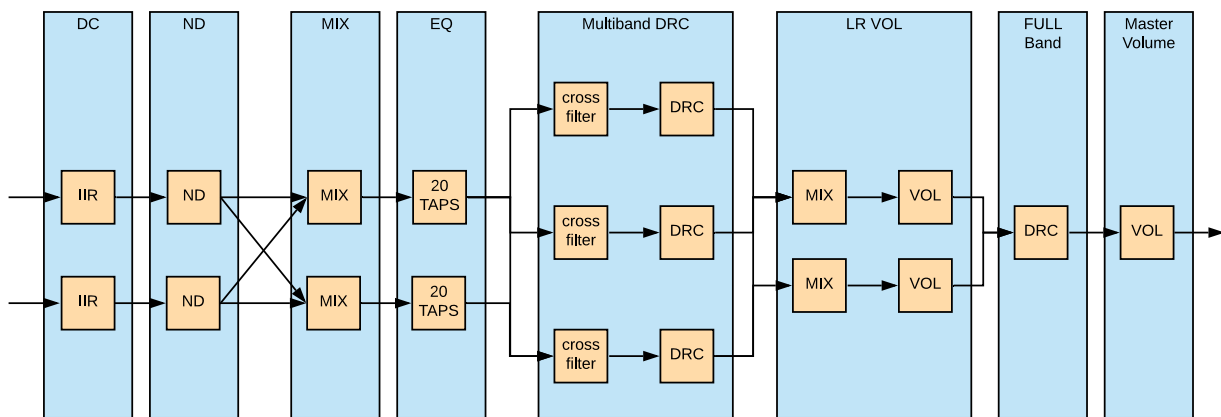
Figure 10-23 Audio VAD

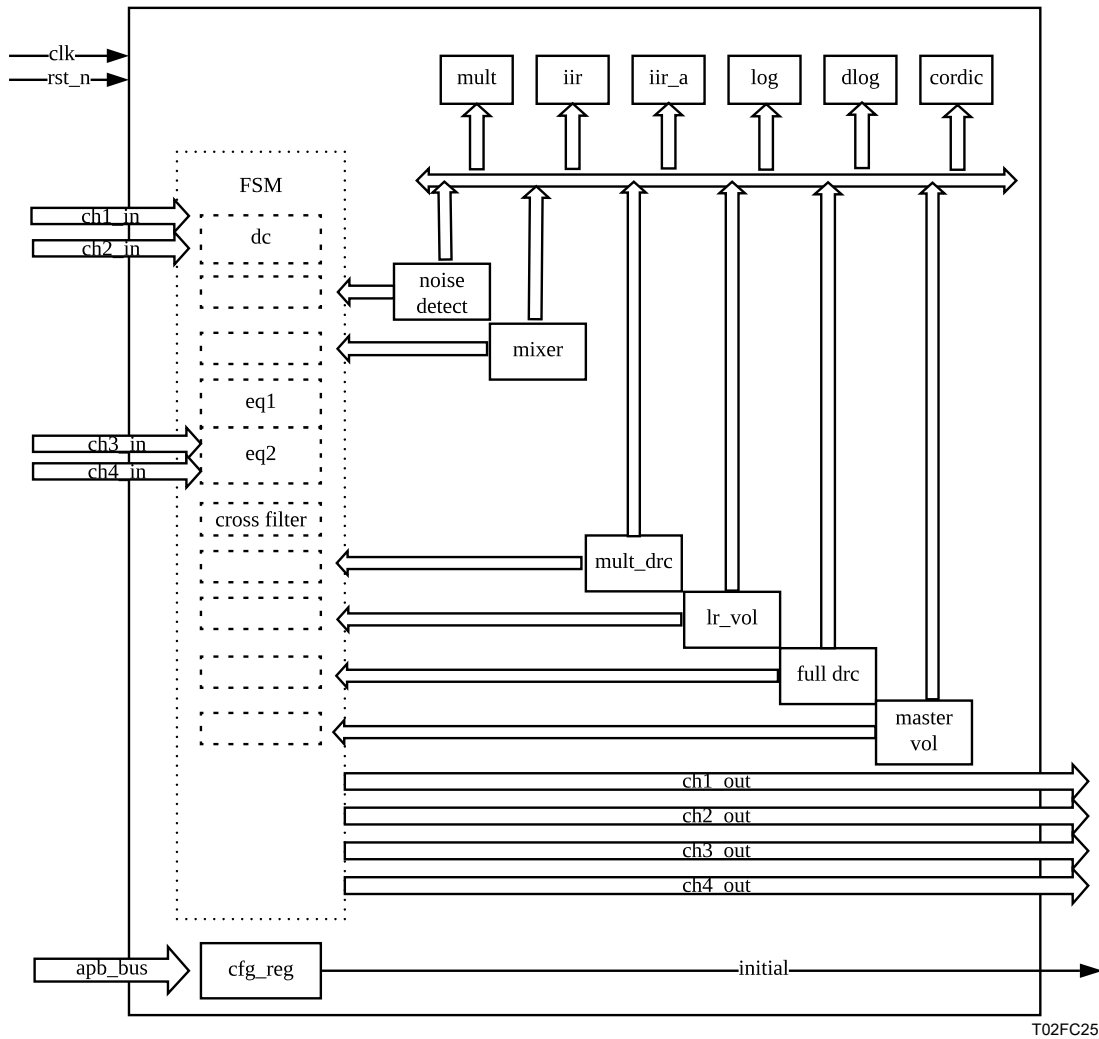


## 10.9 Audio EQDRC

Audio EQDRC supports 4 channels audio streams. Two channels do eq and drc, the others only do eq.

The basic structure diagram is shown below.





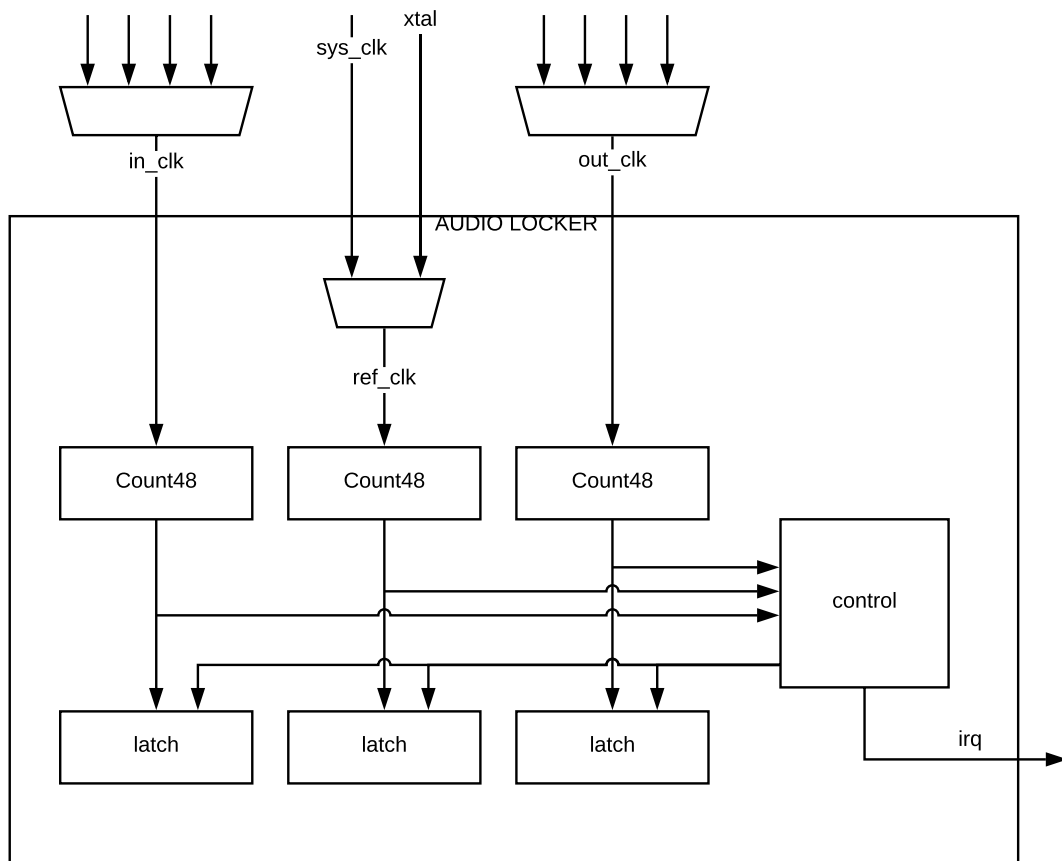
T02FC25

### 10.10 Audio Locker

The audio input clock and output clock are in different power domain and may have little difference, the difference leads to FIFO problem, the SoC uses Audio Locker module to fix this problem, the basic method is to measure the difference between different clocks and adjust (software dynamically) the clock frequency to make them match.

The diagram of Audio Locker module is shown below.

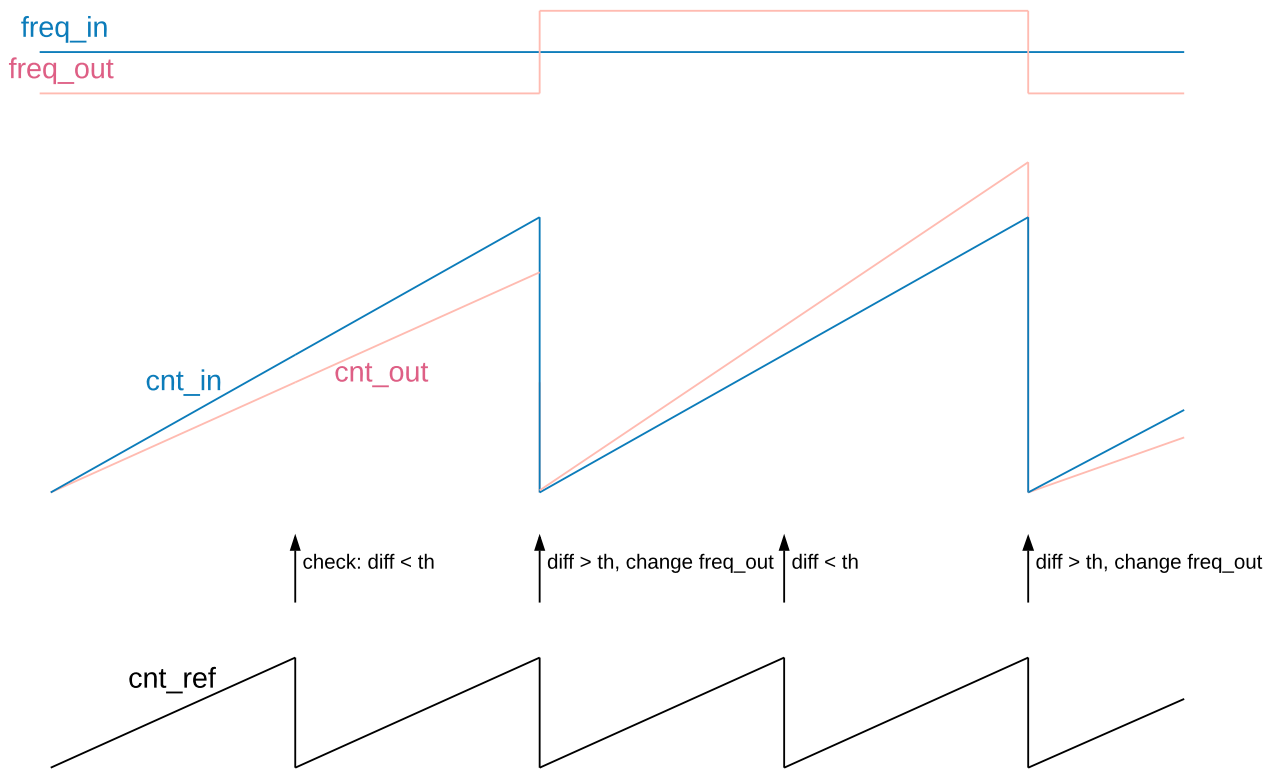
Figure 10-24 Audio Locker



T02FC26

For example: when the difference of two freq creator than threshold is detected, change frequency of `clock_out` by SW, will keep `clock_out` close to `clock_in`.

Figure 10-25 Audio Locker Example



T02FC27

## 10.11 Register Description

### 10.11.1 CLK/RESET Registers

For below registers the base address is 0xfe330000.

Each register final address = BASE + address \* 4.

Table 10-220 EE\_AUDIO\_CLK\_GATE\_EN0 0x00

Bits	R/W	Default	Description
31	R/W	0x0000-0000	frhdmirx, 0:disable; 1: enable;
30	R/W		fratv, 0:disable; 1: enable;
29	R/W		Spdifin_lb, 0:disable; 1: enable;
28	R/W		Audio locker, 0:disable; 1: enable;
27	R/W		tovad, 0:disable; 1: enable;
26	R/W		resampleB, 0:disable; 1: enable;
25	R/W		reserved
24	R/W		reserved
23	R/W		reserved
22	R/W		eqdrc, 0:disable; 1: enable;

Bits	R/W	Default	Description
21	R/W		spdifoutB, 0:disable; 1: enable;
20	R/W		reserved
19	R/W		reserved
18	R/W		resampleA, 0:disable; 1: enable;
17	R/W		spdifout, 0:disable; 1: enable;
16	R/W		spdifin, 0:disable; 1: enable;
15	R/W		loopbackA, 0:disable; 1: enable;
14	R/W		toddrC, 0:disable; 1: enable;
13	R/W		toddrB, 0:disable; 1: enable;
12	R/W		toddrA, 0:disable; 1: enable;
11	R/W		frddrC, 0:disable; 1: enable;
10	R/W		frddrB, 0:disable; 1: enable;
9	R/W		frddrA, 0:disable; 1: enable;
8	R/W		tdmoutC, 0:disable; 1: enable;
7	R/W		tdmoutB, 0:disable; 1: enable;
6	R/W		tdmoutA, 0:disable; 1: enable;
5	R/W		tdminb, 0:disable; 1: enable;
4	R/W		tdminc, 0:disable; 1: enable;
3	R/W		tdminb, 0:disable; 1: enable;
2	R/W		tdmina, 0:disable; 1: enable;
1	R/W		pdm, 0:disable; 1: enable;
0	R/W		ddr_arb, 0:disable; 1: enable;

Table 10-221 EE\_AUDIO\_CLK\_GATE\_EN1 0x01

Bits	R/W	Default	Description
31:3	R/W	0x0000-0000	reserved
8	R/W		locker_b
7	R/W		resampleb_old
6	R/W		earcrx
5	R/W		earctx
4	R/W		toddrE
3	R/W		frddrE
2	R/W		loopbackB, 0:disable; 1: enable;
1	R/W		toddrd, 0:disable; 1: enable;
0	R/W		frddrd, 0:disable; 1: enable;

**Table 10-222 EE\_AUDIO\_MCLK\_A\_CTRL 0x02**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
26:24	R/W		clk_sel, 0:mp0_pll; 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:cts_oscin(24M);
15:0	R/W		clk_div, the frequency of mclk = pll/(clk_div+1);

EE\_AUDIO\_MCLK\_B\_CTRL            0x03

Same as EE\_AUDIO\_MCLK\_A\_CTRL.

EE\_AUDIO\_MCLK\_C\_CTRL            0x04

Same as EE\_AUDIO\_MCLK\_A\_CTRL.

EE\_AUDIO\_MCLK\_D\_CTRL            0x05

Same as EE\_AUDIO\_MCLK\_A\_CTRL.

EE\_AUDIO\_MCLK\_E\_CTRL            0x06

Same as EE\_AUDIO\_MCLK\_A\_CTRL.

EE\_AUDIO\_MCLK\_F\_CTRL            0x07

Same as EE\_AUDIO\_MCLK\_A\_CTRL.

**Table 10-223 EE\_AUDIO\_SW\_RESET0 0x0A**

Bits	R/W	Default	Description
31	R/W	0	Frhdmirx
30	R/W	0	Fratv
29	R/W	0	Spdifin_lb
28	R/W	0	Locker
27	R/W	0	Tovad
26	R/W	0	resampleB
25	R/W	0	clk tree
24	R/W	0	tohdmitx
23	R/W	0	toacodec
20	R/W	0	ddrarb
19	R/W	0	resample
18	R/W	0	eqdrc
17	R/W	0	spdifin

Bits	R/W	Default	Description
16	R/W	0	spdifoutB
15	R/W	0	spdifout
14	R/W	0	tdmoutc
13	R/W	0	tdmoutb
12	R/W	0	tdmouta
11	R/W	0	frddrc
10	R/W	0	frddrb
9	R/W	0	frddra
8	R/W	0	toddrC
7	R/W	0	toddrb
6	R/W	0	toddra
5	R/W	0	loopback
4	R/W	0	tdmin_lb
3	R/W	0	tdminc
2	R/W	0	tdminb
1	R/W	0	tdmina
0	R/W	0	pdm

Table 10-224 EE\_AUDIO\_SW\_RESET1 0x0B

Bits	R/W	Default	Description
8	R/W	0	lockerB
7	R/W	0	resampleB_old
6	R/W	0	earcRX
5	R/W	0	earcTX
4	R/W	0	frddrE
3	R/W	0	toddrE
2	R/W	0	loopbackB
1	R/W	0	toddrd
0	R/W	0	frddrd

Table 10-225 EE\_AUDIO\_MST\_A\_SCLK\_CTRL0 0x10

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_in_en, 0:disable; 1: enable;
30	R/W		clk_out_en, 0:disable; 1: enable;
29:20	R/W		sclk_divN, the frequency of sclk = mclk/(sclk_divN+1);

Bits	R/W	Default	Description
19:10	R/W		lrclk_hi, duty cycle of LRCLK, less than lrclk_div; example 0: lrclk_hi = 1, LRCLK will only keep one cycle; example 1: lrclk_hi = lrclk_div/2, LRCLK will be 50/50 duty cycle;
9:0	R/W		lrclk_divN, the frequency of lrclk = sclk/(lrclk_divN+1);

Table 10-226 EE\_AUDIO\_MST\_A\_SCLK\_CTRL1 0x11

Bits	R/W	Default	Description
31:28	R/W	0x0000-0000	sclk_ph0_sel, select from sclk_delay_line(depth is 16)
27:24	R/W		lrclk_ph0_sel, select from lrclk_delay_line(depth is 16)
23:20	R/W		sclk_ph1_sel, select from sclk_delay_line(depth is 16)
19:16	R/W		lrclk_ph1_sel, select from lrclk_delay_line(depth is 16)
15:12	R/W		sclk_ph2_sel, select from sclk_delay_line(depth is 16)
11:8	R/W		lrclk_ph2_sel, select from lrclk_delay_line(depth is 16)
5:0	R/W		clk_inv, invert clk; [5]: lrclk_ph2; [4]: sclk_ph2; [3]: lrclk_ph1; [2]: sclk_ph1; [1]: lrclk_ph0; [0]: sclk_ph0;

EE\_AUDIO\_MST\_B\_SCLK\_CTRL0 0x12

Same as EE\_AUDIO\_MST\_A\_SCLK\_CTRL0

EE\_AUDIO\_MST\_B\_SCLK\_CTRL1 0x13

Same as EE\_AUDIO\_MST\_A\_SCLK\_CTRL1

EE\_AUDIO\_MST\_C\_SCLK\_CTRL0 0x14

Same as EE\_AUDIO\_MST\_A\_SCLK\_CTRL0

EE\_AUDIO\_MST\_C\_SCLK\_CTRL1 0x15

Same as EE\_AUDIO\_MST\_A\_SCLK\_CTRL1

EE\_AUDIO\_MST\_D\_SCLK\_CTRL0 0x16

Same as EE\_AUDIO\_MST\_A\_SCLK\_CTRL0

EE\_AUDIO\_MST\_D\_SCLK\_CTRL1 0x17

Same as EE\_AUDIO\_MST\_A\_SCLK\_CTRL1

EE\_AUDIO\_MST\_E\_SCLK\_CTRL0 0x18

Same as EE\_AUDIO\_MST\_A\_SCLK\_CTRL0

EE\_AUDIO\_MST\_E\_SCLK\_CTRL1 0x19

Same as EE\_AUDIO\_MST\_A\_SCLK\_CTRL1

EE\_AUDIO\_MST\_F\_SCLK\_CTRL0 0x1a

Same as EE\_AUDIO\_MST\_A\_SCLK\_CTRL0



EE\_AUDIO\_MST\_F\_SCLK\_CTRL1 0x1b

Same as EE\_AUDIO\_MST\_A\_SCLK\_CTRL1

**Table 10-227 EE\_AUDIO\_MST\_DLY\_CTRL0 0x1c**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	reserved
23:20	R/W		mst_sclk_f_ph1_dly_sel;
19:16	R/W		mst_sclk_e_ph1_dly_sel;
15:12	R/W		mst_sclk_d_ph1_dly_sel;
11:8	R/W		mst_sclk_c_ph1_dly_sel;
7:4	R/W		mst_sclk_b_ph1_dly_sel;
3:0	R/W		mst_sclk_a_ph1_dly_sel;

**Table 10-228 EE\_AUDIO\_MST\_DLY\_CTRL1 0x1d**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	reserved
23:20	R/W		mst_lrclk_f_ph1_dly_sel;
19:16	R/W		mst_lrclk_e_ph1_dly_sel;
15:12	R/W		mst_lrclk_d_ph1_dly_sel;
11:8	R/W		mst_lrclk_c_ph1_dly_sel;
7:4	R/W		mst_lrclk_b_ph1_dly_sel;
3:0	R/W		mst_lrclk_a_ph1_dly_sel;

**Table 10-229 EE\_AUDIO\_CLK\_TDMIN\_A\_CTRL 0x20**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_in_en, 0:disable; 1: enable;
30	R/W		clk_out_en, 0:disable; 1: enable;
29	R/W		sclk_inv, 0:not revert; 1: revert clock;

Bits	R/W	Default	Description
27:24	R/W		sclk_sel, 0:mst_a_sclk_ph1; 1:mst_b_sclk_ph1; 2:mst_c_sclk_ph1; 3:mst_d_sclk_ph1; 4:mst_e_sclk_ph1; 5:mst_f_sclk_ph1; 6:i_slv_sclk_a ; 7:i_slv_sclk_b ; 8:i_slv_sclk_c ; 9:i_slv_sclk_d ; 10:i_slv_sclk_e ; 11:i_slv_sclk_f ; 12:i_slv_sclk_g ; 13:i_slv_sclk_h ; 14:i_slv_sclk_i ; 15:i_slv_sclk_j ;
23:20	R/W		lrclk_sel, 0: mst_a_lrclk_ph1; 1:mst_b_lrclk_ph1; 2:mst_c_lrclk_ph1; 3:mst_d_lrclk_ph1; 4:mst_e_lrclk_ph1; 5:mst_f_lrclk_ph1; 6:i_slv_lrclk_a ; 7:i_slv_lrclk_b ; 8:i_slv_lrclk_c ; 9:i_slv_lrclk_d ; 10:i_slv_lrclk_e ; 11:i_slv_lrclk_f ; 12:i_slv_lrclk_g ; 13:i_slv_lrclk_h ; 14:i_slv_lrclk_i ; 15:i_slv_lrclk_j ;

EE\_AUDIO\_CLK\_TDMIN\_B\_CTRL 0x21

Same as EE\_AUDIO\_CLK\_TDMIN\_A\_CTRL

EE\_AUDIO\_CLK\_TDMIN\_C\_CTRL 0x22

Same as EE\_AUDIO\_CLK\_TDMIN\_A\_CTRL

EE\_AUDIO\_CLK\_TDMIN\_LB\_CTRL 0x23

Same as EE\_AUDIO\_CLK\_TDMIN\_A\_CTRL

**Table 10-230 EE\_AUDIO\_CLK\_TDMOUT\_A\_CTRL 0x24**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_in_en, 0:disable; 1: enable;
30	R/W		clk_out_en, 0:disable; 1: enable;
29	R/W		sclk_inv, 0:not revert; 1: revert clock;
28	R/W		Sclk_ws_inv, for the capture ws sclk; 0: not revert; 1: revert clock;

Bits	R/W	Default	Description
27:24	R/W		sclk_sel, 0:mst_a_sclk_ph2; 1:mst_b_sclk_ph2; 2:mst_c_sclk_ph2; 3:mst_d_sclk_ph2; 4:mst_e_sclk_ph2; 5:mst_f_sclk_ph2; 6:i_slv_sclk_a ; 7:i_slv_sclk_b ; 8:i_slv_sclk_c ; 9:i_slv_sclk_d ; 10:i_slv_sclk_e ; 11:i_slv_sclk_f ; 12:i_slv_sclk_g ; 13:i_slv_sclk_h ; 14:i_slv_sclk_i ; 15:i_slv_sclk_j ;
23:20	R/W		lrclk_sel, 0:mst_a_lrclk_ph2; 1:mst_b_lrclk_ph2; 2:mst_c_lrclk_ph2; 3:mst_d_lrclk_ph2; 4:mst_e_lrclk_ph2; 5:mst_f_lrclk_ph2; 6:i_slv_lrclk_a ; 7:i_slv_lrclk_b ; 8:i_slv_lrclk_c ; 9:i_slv_lrclk_d ; 10:i_slv_lrclk_e ; 11:i_slv_lrclk_f ; 12:i_slv_lrclk_g ; 13:i_slv_lrclk_h ; 14:i_slv_lrclk_i ; 15:i_slv_lrclk_j ;

EE\_AUDIO\_CLK\_TDMOUT\_B\_CTRL 0x25

Same as EE\_AUDIO\_CLK\_TDMOUT\_A\_CTRL

EE\_AUDIO\_CLK\_TDMOUT\_C\_CTRL 0x26

Same as EE\_AUDIO\_CLK\_TDMOUT\_A\_CTRL

**Table 10-231 EE\_AUDIO\_CLK\_SPDIFIN\_CTRL 0x27**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);

Bits	R/W	Default	Description
27:24	R/W		clk_sel, 0:mp0_pll; 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:fclk_div5(400M); 8:mst_a_sclk_ph1; 9:mst_b_sclk_ph1; 10:mst_c_sclk_ph1; 11:mst_d_sclk_ph1; 12:mst_e_sclk_ph1; 13:mst_f_sclk_ph1;
7:0	R/W		clk_div, set 7 if need div 8

Table 10-232 EE\_AUDIO\_CLK\_SPDIFOUT\_CTRL 0x28

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
27:24	R/W		clk_sel, 0:mp0_pll; 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:cts_oscin(24M) 8:mst_a_sclk_ph2; 9:mst_b_sclk_ph2; 10:mst_c_sclk_ph2; 11:mst_d_sclk_ph2; 12:mst_e_sclk_ph2; 13:mst_f_sclk_ph2;
9:0	R/W		clk_div, set 7 if need div 8

Table 10-233 EE\_AUDIO\_CLK\_RESAMPLEA\_CTRL 0x29

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);

Bits	R/W	Default	Description
27:24	R/W		sclk_sel, 0:mst_a_mclk; 1:mst_b_mclk; 2:mst_c_mclk; 3:mst_d_mclk; 4:mst_e_mclk; 5:mst_f_mclk; 6:i_slv_sclk_a ; 7:i_slv_sclk_b ; 8:i_slv_sclk_c ; 9:i_slv_sclk_d ; 10:i_slv_sclk_e ; 11:i_slv_sclk_f ; 12:i_slv_sclk_g ; 13:i_slv_sclk_h ; 14:i_slv_sclk_i ; 15:i_slv_sclk_j ;
7:0	R/W		clk_div, set 7 if need div 8

Table 10-234 EE\_AUDIO\_CLK\_LOCKER\_CTRL 0x2a

Bits	R/W	Default	Description
31	R/W	0x0000-0000	lock_out_clk; 0:disable; 1:enable;
30	R/W		Force_oscin for lock_out_clk, 0:disable; 1: force clock source as oscin(24M);
27:24	R/W		sclk_sel, 0:mst_a_mclk; 1:mst_b_mclk; 2:mst_c_mclk; 3:hdmirx_i2s_sclk; 4:world_sync; 5:hdmirx_find_z; 6:i_slv_sclk_a ; 7:i_slv_sclk_b ; 8:i_slv_sclk_c ; 9:i_slv_sclk_d ; 10:i_slv_sclk_e ; 11:earcrx_pll_dmac_ck ; 12:resample_b_vld; 13:resample_a_vld ; 14:earcrx find x/y/z ; 15:spdifin find x/y/z ;
23:16	R/W		clk_div, lock_out_clk, set 7 if need div 8
15	R/W		clk_en, lock_in_clk; 0:disable; 1:enable;
14	R/W		Force_oscin for lock_in_clk, 0:disable; 1: force clock source as oscin(24M);

Bits	R/W	Default	Description
11:8	R/W		clk_sel, lock_in_clk; 0:mst_a_mclk; 1:mst_b_mclk; 2:mst_c_mclk; 3:mst_d_mclk; 4:mst_e_mclk; 5:mst_f_mclk; 6:i_slv_sclk_a ; 7:i_slv_sclk_b ; 8:i_slv_sclk_c ; 9:i_slv_sclk_d ; 10:i_slv_sclk_e ; 11:earcrx_pll_dmac_ck ; 12:resample_b_vld; 13:resample_a_vld ; 14:earcrx find x/y/z ; 15:spdifin find x/y/z ;
7:0	R/W		clk_div, lock_in_clk, set 7 if need div 8

Table 10-235 EE\_AUDIO\_CLK\_PDMIN\_CTRL0 0x2b

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
26:24	R/W		clk_sel, 0:mp0_pll; 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:cts_oscin(24M);
15:0	R/W		clk_div, set 7 if need div 8

Table 10-236 EE\_AUDIO\_CLK\_PDMIN\_CTRL1 0x2c

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
26:24	R/W		clk_sel, 0:cts_oscin(24M); 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:fclk_div5(400M);
15:0	R/W		clk_div, set 7 if need div 8

Table 10-237 EE\_AUDIO\_CLK\_SPDIFOUT\_B\_CTRL 0x2d

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
27:24	R/W		clk_sel, 0:mp0_pll; 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:cts_oscin(24M); 8:mst_a_sclk_ph1; 9:mst_b_sclk_ph1; 10:mst_c_sclk_ph1; 11:mst_d_sclk_ph1; 12:mst_e_sclk_ph1; 13:mst_f_sclk_ph1;
9:0	R/W		clk_div, set 7 if need div 8

Table 10-238 EE\_AUDIO\_CLK\_RESAMPLEB\_CTRL 0x2E

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		sclk_sel,
27:24	R/W		0:mst_a_mclk; 1:mst_b_mclk; 2:mst_c_mclk; 3:mst_d_mclk; 4:mst_e_mclk; 5:mst_f_mclk; 6:i_slv_sclk_a ; 7:i_slv_sclk_b ; 8:i_slv_sclk_c ; 9:i_slv_sclk_d ; 10:i_slv_sclk_e ; 11:i_slv_sclk_f ; 12:i_slv_sclk_g ; 13:i_slv_sclk_h ; 14:i_slv_sclk_i ; 15:i_slv_sclk_j ;
7:0	R/W		clk_div, set 7 if need div 8

Table 10-239 EE\_AUDIO\_CLK\_SPDIFIN\_LB\_CTRL 0x2F

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Clk_sel: 0: spdif_out_a clk; 1: spdif_out_b clk;
29	R/W		Clk_inv

**Table 10-240 EE\_AUDIO\_CLK\_EQDRC\_CTRL 0x30**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
26:24	R/W		0:cts_oscin(24M); 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:fclk_div5(400M)
15:0	R/W		clk_div, set 7 if need div 8

**Table 10-241 EE\_AUDIO\_CLK\_VAD\_CTRL 0x31**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
26:24	R/W		0:cts_oscin(24M); 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:fclk_div5(400M)
15:0	R/W		clk_div, set 7 if need div 8

**Table 10-242 EE\_AUDIO\_EARCTX\_CMDC\_CLK\_CTRL 0x32**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
26:24	R/W		0:cts_oscin(24M); 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:fclk_div5(400M)
15:0	R/W		clk_div, set 7 if need div 8



**Table 10-243 EE\_AUDIO\_EARCTX\_DMAC\_CLK\_CTRL 0x33**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
26:24	R/W		0:cts_oscin(24M); 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:fclk_div5(400M)
15:0	R/W		clk_div, set 7 if need div 8

**Table 10-244 EE\_AUDIO\_EARCRX\_CMDC\_CLK\_CTRL 0x34**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
26:24	R/W		0:cts_oscin(24M); 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:fclk_div5(400M)
15:0	R/W		clk_div, set 7 if need div 8

**Table 10-245 EE\_AUDIO\_EARCRX\_DMAC\_CLK\_CTRL 0x35**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	clk_en, 0:disable; 1: enable;
30	R/W		Force_oscin, 0:disable; 1: force clock source as oscin(24M);
26:24	R/W		0:cts_oscin(24M); 1:mp1_pll; 2:mp2_pll; 3:mp3_pll; 4:hifi_pll; 5:fclk_div3(666M); 6:fclk_div4(500M); 7:fclk_div5(400M)
15:0	R/W		clk_div, set 7 if need div 8

**Table 10-246 EE\_AUDIO\_CLK\_LOCKERB\_CTRL 0x36**

Bits	R/W	Default	Description
Bits	From	Description	
31	R/W	0x0000-0000	lockb_out_clk; 0:disable; 1:enable;
30	R/W		Force_oscin for lock_out_clk, 0:disable; 1: force clock source as oscin(24M);
27:24	R/W		sclk_sel, 0:mst_a_mclk; 1:mst_b_mclk; 2:mst_c_mclk; 3:hdmirx_i2s_sclk; 4:world_sync; 5:hdmirx_find_z; 6:i_slv_sclk_a ; 7:i_slv_sclk_b ; 8:i_slv_sclk_c ; 9:i_slv_sclk_d ; 10:i_slv_sclk_e ; 11:earcrx_pll_dmac_ck ; 12:resample_b_vld; 13:resample_a_vld ; 14:earcrx find x/y/z ; 15:spdifin find x/y/z ;
23:16	R/W		clk_div, lock_out_clk, set 7 if need div 8
15	R/W		clk_en, lock_in_clk; 0:disable; 1:enable;
14	R/W		Force_oscin for lock_in_clk, 0:disable; 1: force clock source as oscin(24M);
11:8	R/W		clk_sel, lock_inb_clk; 0:mst_a_mclk; 1:mst_b_mclk; 2:mst_c_mclk; 3:mst_d_mclk; 4:mst_e_mclk; 5:mst_f_mclk; 6:i_slv_sclk_a ; 7:i_slv_sclk_b ; 8:i_slv_sclk_c ; 9:i_slv_sclk_d ; 10:i_slv_sclk_e ; 11:earcrx_pll_dmac_ck ; 12:resample_b_vld; 13:resample_a_vld ; 14:earcrx find x/y/z ; 15:spdifin find x/y/z ;
7:0	R/W		clk_div, lock_in_clk, set 7 if need div 8

### 10.11.2 OTHER Registers

The audio module includes 64 exception irq, see the following table.

**Table 10-247 Audio IRQs**

Bits	From	Description
63	TODDR_A	CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
62	TODDR_A	CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)

Bits	From	Description
61	TODDR_B	CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
60	TODDR_B	CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)
59	TODDR_C	CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
58	TODDR_C	CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)
57	TODDR_D	CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
56	TODDR_D	CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)
55	TODDR_E	CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
54	TODDR_E	CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)
53		
52		
51		
50		
49	PDMIN	HPF NOT FINISH
48	PDMIN	HCIC NOT FINISH
47		
46	TDMIN_A	AFIFO OVERFLOW
45	TDMIN_A	SLOT_CNT_ERR, SLOT CNT between two WS_RISE didn't equal to REG_SLOT_NUM
44	TDMIN_A	BIT_CNT_ERR, BIT CNT between two SLOT didn't equal to REG_BIT_NUM
43		
42	TDMIN_B	AFIFO OVERFLOW
41	TDMIN_B	SLOT_CNT_ERR, SLOT CNT between two WS_RISE didn't equal to REG_SLOT_NUM
40	TDMIN_B	BIT_CNT_ERR, BIT CNT between two SLOT didn't equal to REG_BIT_NUM
39		
38	TDMIN_C	AFIFO OVERFLOW
37	TDMIN_C	SLOT_CNT_ERR, SLOT CNT between two WS_RISE didn't equal to REG_SLOT_NUM
36	TDMIN_C	BIT_CNT_ERR, BIT CNT between two SLOT didn't equal to REG_BIT_NUM
35	DDR_ARB	locked for one request for long time (> time_out_cnt)
34	TDMIN_LB	AFIFO OVERFLOW
33	TDMIN_LB	SLOT_CNT_ERR, SLOT CNT between two WS_RISE didn't equal to REG_SLOT_NUM
32	TDMIN_LB	BIT_CNT_ERR, BIT CNT between two SLOT didn't equal to REG_BIT_NUM
31	LOOPBACK_A	ORIG DATA,CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
30	LOOPBACK_A	ORIG DATA,CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)
29	LOOPBACK_A	INSERT DATA,CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
28	LOOPBACK_A	INSERT DATA,CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)

Bits	From	Description
27	LOOPBACK_B	ORIG DATA,CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
26	LOOPBACK_B	ORIG DATA,CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)
25	LOOPBACK_B	INSERT DATA,CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
24	LOOPBACK_B	INSERT DATA,CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)
23	RESAMPLE_A	CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
22	RESAMPLE_A	CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)
21	RESAMPLE_B	CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
20	RESAMPLE_B	CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)
19		
18		
17	SPDIFIN	Reserved
16	SPDIFIN	afifo overflow
15	SPDIFIN	cyc_cnt err, < min
14	SPDIFIN	cyc_cnt err, > max
13	SPDIFIN	parity_err
12	SPDIFIN	bit_cnt err, > 64
11	SPDIFIN	bit_cnt err, !=64
10	SPDIFIN	xyz_err
9	VAD	CHNUM_SYNC_ERR1: detect write new ch when pop buffer;
8	VAD	CHNUM_SYNC_ERR0: miss CH (cur_chnum_id != reg_chnum_id)
7		
6		
5		
4		
3		
2		
1		
0		

Table 10-248 EE\_AUDIO\_EXCEPTION\_IRQ\_STS0 0x380

Bits	R/W	Default	Description
31	R	0x0000-0000	change to high by irq; change to low by clr;

**Table 10-249 EE\_AUDIO\_EXCEPTION\_IRQ\_STS1 0x381**

Bits	R/W	Default	Description
31	R	0x0000-0000	change to high by irq; change to low by clr;

**Table 10-250 EE\_AUDIO\_EXCEPTION\_IRQ\_MASK0 0x382**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	

**Table 10-251 EE\_AUDIO\_EXCEPTION\_IRQ\_MASK1 0x383**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	

**Table 10-252 EE\_AUDIO\_EXCEPTION\_IRQ\_MODE0 0x384**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	each bit: 0: pulse irq; 1: level irq;

**Table 10-253 EE\_AUDIO\_EXCEPTION\_IRQ\_MODE1 0x385**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	each bit: 0: pulse irq; 1: level irq;

**Table 10-254 EE\_AUDIO\_EXCEPTION\_IRQ\_CLR0 0x386**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	each bit can clear each sts bit

**Table 10-255 EE\_AUDIO\_EXCEPTION\_IRQ\_CLR1 0x387**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	each bit can clear each sts bit

**Table 10-256 EE\_AUDIO\_EXCEPTION\_IRQ\_INV0 0x388**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	each bit can invert irq

**Table 10-257 EE\_AUDIO\_EXCEPTION\_IRQ\_INV1 0x389**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	each bit can invert irq

**Table 10-258 EE\_AUDIO\_RSAMP\_A\_CHNUM\_ID0 0x350**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch3
23:16	R/W		ID of ch2
15:8	R/W		ID of ch1
7:0	R/W		ID of ch0

**Table 10-259 EE\_AUDIO\_RSAMP\_A\_CHNUM\_ID1 0x351**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch7
23:16	R/W		ID of ch6
15:8	R/W		ID of ch5
7:0	R/W		ID of ch4

**Table 10-260 EE\_AUDIO\_RSAMP\_A\_CHNUM\_ID2 0x352**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch11
23:16	R/W		ID of ch10
15:8	R/W		ID of ch9
7:0	R/W		ID of ch8

**Table 10-261 EE\_AUDIO\_RSAMP\_A\_CHNUM\_ID3 0x353**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch15
23:16	R/W		ID of ch14
15:8	R/W		ID of ch13
7:0	R/W		ID of ch12

**Table 10-262 EE\_AUDIO\_RSAMP\_A\_CHNUM\_ID4 0x354**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch19
23:16	R/W		ID of ch18
15:8	R/W		ID of ch17
7:0	R/W		ID of ch16

**Table 10-263 EE\_AUDIO\_RSAMP\_A\_CHNUM\_ID5 0x355**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch23
23:16	R/W		ID of ch22
15:8	R/W		ID of ch21
7:0	R/W		ID of ch20

**Table 10-264 EE\_AUDIO\_RSAMP\_A\_CHNUM\_ID6 0x356**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch27
23:16	R/W		ID of ch26
15:8	R/W		ID of ch25
7:0	R/W		ID of ch24

**Table 10-265 EE\_AUDIO\_RSAMP\_A\_CHNUM\_ID7 0x357**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch31
23:16	R/W		ID of ch30
15:8	R/W		ID of ch29
7:0	R/W		ID of ch28

**Table 10-266 EE\_AUDIO\_RSAMP\_A\_CHNUM\_MASK 0x35E**

Bits	R/W	Default	Description
23:16	R/W	0x0000-0000	for resample_C(B_old) stable mask, if set 1, will pull stable signal to low by different conditions bit3: resample_out_en changed(from 1 to 0 or from 0 to 1); bit2: reg_soft_reset = 1; bit1: resample_en_r changed(from 1 to 0 or from 0 to 1); bit0: resample_en changed(from 1 to 0 or from 0 to 1);
15:8	R/W		for resample_B stable mask, if set 1, will pull stable signal to low by different conditions bit3: reg_rsampl_en = 0; bit2: reg_rsamp_rst = 1; bit1: reg_output_en = 0; bit0: reg_module_bypass changed(from 0 to 1 or from 1 to 0)
7:0	R/W		for resample_A stable mask, if set 1, will pull stable signal to low by different conditions bit3: reg_rsampl_en = 0; bit2: reg_rsamp_rst = 1; bit1: reg_output_en = 0; bit0: reg_module_bypass changed(from 0 to 1 or from 1 to 0)

**Table 10-267 EE\_AUDIO\_RSAMP\_A\_CHSYNC\_CTRL 0x35F**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	bypass_n; 0: out=in; 1: sync data by chnum info;
30	R/W		sw_reset; 1: reset chnum_sync module
29	R/W		mode; 0: work as BUFF_MODE; 1: work as DROP_MODE;
28	R/W		id_se; 0: ID = input valid count; 1: ID= reg_id_***
27	R/W		fix_stable; 1: ignore stable info of input; 0: compensate dat by stable info;
26	R/W		debug_en0, for BUF_MODE only; 1: send chnum info from input; 0: generate new chnum info;
7:0	R/W		chnum_max; if input is 8ch, set this to 7;

EE\_AUDIO\_RSAMP\_B\_CHNUM\_ID0 0x360  
 EE\_AUDIO\_RSAMP\_B\_CHNUM\_ID1 0x361  
 EE\_AUDIO\_RSAMP\_B\_CHNUM\_ID2 0x362  
 EE\_AUDIO\_RSAMP\_B\_CHNUM\_ID3 0x363  
 EE\_AUDIO\_RSAMP\_B\_CHNUM\_ID4 0x364  
 EE\_AUDIO\_RSAMP\_B\_CHNUM\_ID5 0x365  
 EE\_AUDIO\_RSAMP\_B\_CHNUM\_ID6 0x366  
 EE\_AUDIO\_RSAMP\_B\_CHNUM\_ID7 0x367  
 EE\_AUDIO\_RSAMP\_B\_CHSYNC\_CTRL 0x36F



### 10.11.3 TODDR Registers

Table 10-268 EE\_AUDIO\_TODDR\_A\_CTRL0 0x40

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_toddr_en, 0: disable; 1: enable;
29	R/W		reg_ext_signed, 0: select write to only one buff (start_addr,finish_addr); 1: select write to two buff (start_addr,finish_addr) (start_addrb, finish_addrb);
28	R/W		reg_toddr_endian
27	R/W		Enable_sync_chnum; 1: start store data when first ch ;
26:24	R/W		reg_toddr_int_en
23:16	R/W		[23] : reserved; [22] : reserved; [21] : fifo overflow, write when fifocnt = depth; [20] : fifo overflow, read when fifocnt = 0; [19] : when write to ddr "int_addr" data (only once); [18] : when write to ddr "int_addr" data (repeat); [17] : when write to ddr address match "int_addr"; [16] : when write to ddr address match "finish_addr";
15:13	R/W		reg_toddr_sel, 0: combined data[m:n] without gap; like S0[m:n],S1[m:n],S2[m:n], ... 1: combined data[m:n] as 16bits; like {S0[11:0],4'd0},{S1[11:0],4'd0}... 2: combined data[m:n] as 16bits; like {4'd0,S0[11:0]},{4'd0,{S1[11:0]}... 3: combined data[m:n] as 32bits; like {S0[27:4],8'd0},{S1[27:4],8'd0}... 4: combined data[m:n] as 32bits; like {8'd0,S0[27:4]},{8'd0,{S1[27:4]}...
12:8	R/W		reg_toddr_m_sel, the msb position in data
7:3	R/W		reg_toddr_n_sel, the lsb position in data
2	R/W		Clear adr/cnt value which captured by vad frame sync
1	R/W		Clear adr/cnt value which captured by vadflag
0	R/W		Ddr bus ugt bit

Table 10-269 EE\_AUDIO\_TODDR\_A\_CTRL1 0x41

Bits	R/W	Default	Description
30:26	R/W	0x0000-0000	Src select: 0: tadmin_a; 1: tadmin_b; 2: tadmin_c; 3: spdifin; 4: pdmin; 5: fratv; 6: tadmin_lb; 7: loopback_a; 8: frhdmirx_spdif; 9: loopback_b; 10: spdifin_lb; 11:earc_rx_dmac 12:frhdmirx_dsd 13:resample_a 14:resample_b 15: vad; 16~31:reserved
25	R/W		force_finish; the value from 0-> 1: force finish by current address and jump to start_address;
23:12	R/W		reg_fifo_start_rd_th, each time, when fifo_cnt greater than this register, control will start read data from fifo and write to DDR; write length is "reg_fifo_start_rd_th + 1" * 8bytes;
11:8	R/W		reg_status_sel, control status2 source;
7:0	R/W		reg_int_status_clr,clear each bits of int_status register

Table 10-270 EE\_AUDIO\_TODDR\_A\_START\_ADDR 0x42

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	start_addr, buff_A start address, ignore [2:0]

Table 10-271 EE\_AUDIO\_TODDR\_A\_FINISH\_ADDR 0x43

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	finish_addr, buff_A finish address, ignore [2:0]

Table 10-272 EE\_AUDIO\_TODDR\_A\_INT\_ADDR 0x44

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	int_addr, usage A : as an address of interrupt; usage B : as a count of interrupt;

Table 10-273 EE\_AUDIO\_TODDR\_A\_STATUS1 0x45

Bits	R/W	Default	Description
25:23	R	0x0000-0000	stop_dds_status
22	R		stop_dds_done

Bits	R/W	Default	Description
21	R		sel_b_true
20	R/W		sel_b
19:8	R/W		fifo count, the num in fifo
7:0	R/W		int_status, when irq generate, related bit will changed to 1 and can only clear by reg_int_status_clr

Table 10-274 EE\_AUDIO\_TODDR\_A\_STATUS2 0x46

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	status2, by reg_status_sel: 0: current ddr write address; 1: next finish address; 2: count by ddr reply, current ddr write address; 3: count by ddr reply, next finish address; 4: ddr address captured by vad flag; 5: ddr address captured by vad frame sync; 6: [31:16]: fifo_cnt captured by vad frame sync; [15:0]: fifo_cnt captured by vad flag; 8: ddr address captured by first vad flag; 9: ddr address captured by first vad frame sync;

Table 10-275 EE\_AUDIO\_TODDR\_A\_START\_ADDRB 0x47

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	istart_addrb, buff_B start address, ignore [2:0]

Table 10-276 EE\_AUDIO\_TODDR\_A\_FINISH\_ADDRB 0x48

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	finish_addrb, buff_B finish address, ignore [2:0]

Table 10-277 EE\_AUDIO\_TODDR\_A\_INIT\_ADDR 0x49

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	initial address, the first ddr address after enable

Table 10-278 EE\_AUDIO\_TODDR\_A\_CTRL2 0x4a

Bits	R/W	Default	Description
31	R/W	0x0000-0000	Reserved for hold read. need set 1 before toddr_en = 1, then TODDR will hold all data in FIFO. after sometime (etc. VAD detect), set it to 0, TODDR will start flush all data in FIFO to ddr.
30	R/W		reg_stop_ddr; if set from 0 to 1, will: step1: stop write data to FIFO; step2: stop sending request to DDR; step3: keep receiving data from DDR;

Bits	R/W	Default	Description
			step4: compare request count and receive count; step5: done if two count matched;
29:28	R/W		Insert_chnum; 3: replace data[9:0] to chnum[9:0]; 2: replace data[3:0] to chnum[3:0]; 1: replace data[0] to chnum[0]; 0: not replace;
23:16	R/W		each bits will set to one interrupt, 0: pulse mode; 1: level mode;
11:0	R/W		Reserved for hold read start offset. when release hold read (bit31), the read point of fifo will equal to current write point + this reg +1. fifo width is 8 Bytes. so if input is 8ch*32bits, this register need set to 3/7/11/... if input is 32ch*32bits, this register need set to 15/31/47/...

Table 10-279 EE\_AUDIO\_TODDR\_A\_CHNUM\_ID0 0x300

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch3
23:16	R/W		ID of ch2
15:8	R/W		ID of ch1
7:0	R/W		ID of ch0

Table 10-280 EE\_AUDIO\_TODDR\_A\_CHNUM\_ID1 0x301

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch7
23:16	R/W		ID of ch6
15:8	R/W		ID of ch5
7:0	R/W		ID of ch4

Table 10-281 EE\_AUDIO\_TODDR\_A\_CHNUM\_ID2 0x302

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch11
23:16	R/W		ID of ch10
15:8	R/W		ID of ch9
7:0	R/W		ID of ch8

**Table 10-282 EE\_AUDIO\_TODDR\_A\_CHNUM\_ID3 0x303**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch15
23:16	R/W		ID of ch14
15:8	R/W		ID of ch13
7:0	R/W		ID of ch12

**Table 10-283 EE\_AUDIO\_TODDR\_A\_CHNUM\_ID4 0x304**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch19
23:16	R/W		ID of ch18
15:8	R/W		ID of ch17
7:0	R/W		ID of ch16

**Table 10-284 EE\_AUDIO\_TODDR\_A\_CHNUM\_ID5 0x305**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch23
23:16	R/W		ID of ch22
15:8	R/W		ID of ch21
7:0	R/W		ID of ch20

**Table 10-285 EE\_AUDIO\_TODDR\_A\_CHNUM\_ID6 0x306**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch27
23:16	R/W		ID of ch26
15:8	R/W		ID of ch25
7:0	R/W		ID of ch24

**Table 10-286 EE\_AUDIO\_TODDR\_A\_CHNUM\_ID7 0x307**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ID of ch31
23:16	R/W		ID of ch30
15:8	R/W		ID of ch29
7:0	R/W		ID of ch28

Table 10-287 EE\_AUDIO\_TODDR\_A\_CHSYNC\_CTRL 0x30F

Bits	R/W	Default	Description
31	R/W	0x0000-0000	bypass_n; 0: out=in; 1: sync data by chnum info;
30	R/W		sw_reset; 1: reset chnum_sync module
29	R/W		mode; 0: work as BUFF_MODE; 1: work as DROP_MODE;
28	R/W		id_se; 0: ID = input valid count; 1: ID= reg_id_***
27	R/W		fix_stable; 1: ignore stable info of input; 0: compensate dat by stable info;
26	R/W		debug_en0, for BUF_MODE only; 1: send chnum info from input; 0: generate new chnum info;
7:0	R/W		chnum_max; if input is 8ch, set this to 7;

EE\_AUDIO\_TODDR\_B\_CTRL0        0x50  
 EE\_AUDIO\_TODDR\_B\_CTRL1        0x51  
 EE\_AUDIO\_TODDR\_B\_START\_ADDR   0x52  
 EE\_AUDIO\_TODDR\_B\_FINISH\_ADDR   0x53  
 EE\_AUDIO\_TODDR\_B\_INT\_ADDR     0x54  
 EE\_AUDIO\_TODDR\_B\_STATUS1      0x55  
 EE\_AUDIO\_TODDR\_B\_STATUS2      0x56  
 EE\_AUDIO\_TODDR\_B\_START\_ADDRB   0x57  
 EE\_AUDIO\_TODDR\_B\_FINISH\_ADDR   0x58  
 EE\_AUDIO\_TODDR\_B\_INIT\_ADDR    0x59  
 EE\_AUDIO\_TODDR\_B\_CTRL2        0x5a  
 EE\_AUDIO\_TODDR\_B\_CHNUM\_ID0    0x310  
 EE\_AUDIO\_TODDR\_B\_CHNUM\_ID1    0x311  
 EE\_AUDIO\_TODDR\_B\_CHNUM\_ID2    0x312  
 EE\_AUDIO\_TODDR\_B\_CHNUM\_ID3    0x313  
 EE\_AUDIO\_TODDR\_B\_CHNUM\_ID4    0x314  
 EE\_AUDIO\_TODDR\_B\_CHNUM\_ID5    0x315  
 EE\_AUDIO\_TODDR\_B\_CHNUM\_ID6    0x316  
 EE\_AUDIO\_TODDR\_B\_CHNUM\_ID7    0x317  
 EE\_AUDIO\_TODDR\_B\_CHSYNC\_CTRL   0x31F

EE\_AUDIO\_TODDR\_C\_CTRL0        0x60  
 EE\_AUDIO\_TODDR\_C\_CTRL1        0x61  
 EE\_AUDIO\_TODDR\_C\_START\_ADDR    0x62  
 EE\_AUDIO\_TODDR\_C\_FINISH\_ADDR   0x63

---

EE_AUDIO_TODDR_C_INT_ADDR	0x64
EE_AUDIO_TODDR_C_STATUS1	0x65
EE_AUDIO_TODDR_C_STATUS2	0x66
EE_AUDIO_TODDR_C_START_ADDRB	0x67
EE_AUDIO_TODDR_C_FINISH_ADDR	0x68
EE_AUDIO_TODDR_C_INIT_ADDR	0x69
EE_AUDIO_TODDR_C_CTRL2	0x6a
EE_AUDIO_TODDR_C_CHNUM_ID0	0x320
EE_AUDIO_TODDR_C_CHNUM_ID1	0x321
EE_AUDIO_TODDR_C_CHNUM_ID2	0x322
EE_AUDIO_TODDR_C_CHNUM_ID3	0x323
EE_AUDIO_TODDR_C_CHNUM_ID4	0x324
EE_AUDIO_TODDR_C_CHNUM_ID5	0x325
EE_AUDIO_TODDR_C_CHNUM_ID6	0x326
EE_AUDIO_TODDR_C_CHNUM_ID7	0x327
EE_AUDIO_TODDR_C_CHSYNC_CTRL	0x32F
EE_AUDIO_TODDR_D_CTRL0	0x210
EE_AUDIO_TODDR_D_CTRL1	0x211
EE_AUDIO_TODDR_D_START_ADDR	0x212
EE_AUDIO_TODDR_D_FINISH_ADDR	0x213
EE_AUDIO_TODDR_D_INT_ADDR	0x214
EE_AUDIO_TODDR_D_STATUS1	0x215
EE_AUDIO_TODDR_D_STATUS2	0x216
EE_AUDIO_TODDR_D_START_ADDRB	0x217
EE_AUDIO_TODDR_D_FINISH_ADDR	0x218
EE_AUDIO_TODDR_D_INIT_ADDR	0x219
EE_AUDIO_TODDR_D_CTRL2	0x21a
EE_AUDIO_TODDR_D_CHNUM_ID0	0x330
EE_AUDIO_TODDR_D_CHNUM_ID1	0x331
EE_AUDIO_TODDR_D_CHNUM_ID2	0x332
EE_AUDIO_TODDR_D_CHNUM_ID3	0x333
EE_AUDIO_TODDR_D_CHNUM_ID4	0x334
EE_AUDIO_TODDR_D_CHNUM_ID5	0x335
EE_AUDIO_TODDR_D_CHNUM_ID6	0x336
EE_AUDIO_TODDR_D_CHNUM_ID7	0x337
EE_AUDIO_TODDR_D_CHSYNC_CTRL	0x33F

### 10.11.4 FRDDR Registers

**Table 10-288 EE\_AUDIO\_FRDDR\_A\_CTRL0 0x70**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_frddr_en, 0: disable; 1: enable;
30	R/W		reg_frddr_pp_mode, 0: select write to only one buff (start_addr,finish_addr); 1: select write to two buff (start_addr,finish_addr) (start_addrb, finish_addrb);
26:24	R/W		reg_frdd_endian
23:16	R/W		reg_frddr_int_en, [23] : reserved; [22] : reserved; [21] : fifo overflow, write when fifocnt = depth; [20] : fifo overflow, read when fifocnt = 0; [19] : first time when read from ddr "int_addr" data(only once); [18] : once time when read from ddr "int_addr" data(repeat); [17] : when read from ddr address match "int_addr"; [16] : when read from ddr address match "finish_addr";
15:12	R/W		reg_frddr_ack_dly, add delay to frddr ack
0	R/W		Ddr bus ugt setting.

**Table 10-289 EE\_AUDIO\_FRDDR\_A\_CTRL1 0x71**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	reg_fifo_depth, the max depth of fifo; for high bit rates like 384k*32bits*8ch, set this register higher; for low bit rates like 48k*32bits*2ch, set this register lower;
23:16			reg_fifo_start_wr_th, when the fifo cnt less than "reg_fifo_depth - reg_fifo_start_wr_th", start request and read data from DDR; each time "reg_fifo_start_wr_th" * 8 bytes data;
12			force finish; when the value changed from 0 to 1; will finished by current address and jump to start address;
11:8			reg_status_sel, control status2 source;
7:0			reg_int_status_clr, clear each bits of int_status register

**Table 10-290 EE\_AUDIO\_FRDDR\_A\_START\_ADDR 0x72**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	start_addr,buff_B start address

**Table 10-291 EE\_AUDIO\_FRDDR\_A\_FINISH\_ADDR 0x73**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	finish_addr,buff_B finish address



**Table 10-292 EE\_AUDIO\_FRDDR\_A\_INT\_ADDR 0x74**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	int_addr, usage A : as an address of interrupt; usage B : as a count of interrupt;

**Table 10-293 EE\_AUDIO\_FRDDR\_A\_STATUS1 0x75**

Bits	R/W	Default	Description
22:20	R	0x0000-0000	stop_ddr status
19	R		r_selb_true
18	R		r_selb
17	R		stop_ddr_done
16:8	R/W		fifo count, the num in fifo
7:0	R/W		int_status, when irq generate, related bit will changed to 1 and can only clear by reg_int_status_clr

**Table 10-294 EE\_AUDIO\_FRDDR\_A\_STATUS2 0x76**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	status2, by reg_status_sel: 0: current ddr write address; 1: next finish address; 2: count by ddr reply, current ddr write address; 3: count by ddr reply, next finish address;

The same design as TODDR:

We can generate 8 irq and add them together to CPU.

Then can read int\_status to know which irq it is.

EE\_AUDIO\_FRDDR\_A\_START\_ADDRB 0x77

EE\_AUDIO\_FRDDR\_A\_FINISH\_ADDRB 0x78

The same design as TODDR:

We have an internal register to store reg\_start\_adr and reg\_finish\_adr.

We call them as r\_start\_adr and r\_finish\_adr;

Each time, when curr\_adr match r\_finish\_adr, curr\_adr will jump to r\_start\_adr, then update r\_start\_adr and r\_finish\_adr;

That means SW can write new start adr and finish adr before curr\_adr match old finish adr.

EE\_AUDIO\_FRDDR\_A\_INIT\_ADDR 0x79

**Table 10-295 EE\_AUDIO\_FRDDR\_A\_CTRL2 0x7a**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	ch num for share buffer feature; if needn't share buffer, set it as 0; if need share buffer , set it as TDMOUT/SPDIFOUT ch number;
21	R/W		reg_stop_ddr; if set from 0 to 1, will: step1: stop write data to FIFO; step2: stop sending to DDR; step3: keep receiving data from DDR; step4: compare count and receive count; step5: done if two count matched;
20	R/W		Src_sel2_en
19	R/W		Src_sel2_eq; 1: select from EQDRC; 0: select from TDMOUT/SPDIFOUT;
18:16	R/W		Src_sel2; 0: tdmout_a; 1: tdmout_b; 2: tdmout_c; 3: spdifout; 4: spdifout_b; 5: reserved; 6: reserved; 7: reserved;
12	R/W		Src_sel1_en
11	R/W		Src_sel1_eq; 1: select from EQDRC; 0: select from TDMOUT/SPDIFOUT;
10:8	R/W		Src_sel1; 0: tdmout_a; 1: tdmout_b; 2: tdmout_c; 3: spdifout; 4: spdifout_b; 5: reserved; 6: reserved; 7: reserved;
4	R/W		Src_sel0_en
3	R/W		Src_sel0_eq; 1: select from EQDRC; 0: select from TDMOUT/SPDIFOUT;
2:0	R/W		Src_sel0; 0: tdmout_a; 1: tdmout_b; 2: tdmout_c; 3: spdifout; 4: spdifout_b; 5: reserved; 6: reserved; 7: reserved;

EE\_AUDIO\_FRDDR\_B\_CTRL0      0x80  
EE\_AUDIO\_FRDDR\_B\_CTRL1      0x81  
EE\_AUDIO\_FRDDR\_B\_START\_ADDR    0x82  
EE\_AUDIO\_FRDDR\_B\_FINISH\_ADDR   0x83  
EE\_AUDIO\_FRDDR\_B\_INT\_ADDR      0x84  
EE\_AUDIO\_FRDDR\_B\_STATUS1      0x85  
EE\_AUDIO\_FRDDR\_B\_STATUS2      0x86

EE\_AUDIO\_FRDDR\_B\_START\_ADDRB 0x87  
 EE\_AUDIO\_FRDDR\_B\_FINISH\_ADDR 0x88  
 EE\_AUDIO\_FRDDR\_B\_INIT\_ADDR 0x89  
 EE\_AUDIO\_FRDDR\_B\_CTRL2 0x8a  
  
 EE\_AUDIO\_FRDDR\_C\_CTRL0 0x90  
 EE\_AUDIO\_FRDDR\_C\_CTRL1 0x91  
 EE\_AUDIO\_FRDDR\_C\_START\_ADDR 0x92  
 EE\_AUDIO\_FRDDR\_C\_FINISH\_ADDR 0x93  
 EE\_AUDIO\_FRDDR\_C\_INT\_ADDR 0x94  
 EE\_AUDIO\_FRDDR\_C\_STATUS1 0x95  
 EE\_AUDIO\_FRDDR\_C\_STATUS2 0x96  
 EE\_AUDIO\_FRDDR\_C\_START\_ADDRB 0x97  
 EE\_AUDIO\_FRDDR\_C\_FINISH\_ADDR 0x98  
 EE\_AUDIO\_FRDDR\_C\_INIT\_ADDR 0x99  
 EE\_AUDIO\_FRDDR\_C\_CTRL2 0x9a  
  
 EE\_AUDIO\_FRDDR\_D\_CTRL0 0x220  
 EE\_AUDIO\_FRDDR\_D\_CTRL1 0x221  
 EE\_AUDIO\_FRDDR\_D\_START\_ADDR 0x222  
 EE\_AUDIO\_FRDDR\_D\_FINISH\_ADDR 0x223  
 EE\_AUDIO\_FRDDR\_D\_INT\_ADDR 0x224  
 EE\_AUDIO\_FRDDR\_D\_STATUS1 0x225  
 EE\_AUDIO\_FRDDR\_D\_STATUS2 0x226  
 EE\_AUDIO\_FRDDR\_D\_START\_ADDRB 0x227  
 EE\_AUDIO\_FRDDR\_D\_FINISH\_ADDR 0x228  
 EE\_AUDIO\_FRDDR\_D\_INIT\_ADDR 0x229  
 EE\_AUDIO\_FRDDR\_D\_CTRL2 0x22a

### 10.11.5 DDR ARB Registers

Table 10-296 EE\_AUDIO\_ARB\_CTRL0 0xa0

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_arb_en, 0:disable; 1: enable;
30	R/W		reg_clr_lock_clk_cnt, 1: clear clk_cnt of lock status;
29	R/W		reg_unlock; 1: will reset the status of DDR arbiter to IDLE;

Bits	R/W	Default	Description
28	R/W		reg_unlock_no_async; 0: will clear status of DDR ASYNC when set reg_unlock to 1;
27	R/W		reg_reset_async; 1: will reset DDR_ASYNC module;
22:20	R/W		reg_sts_sel, refer to ARB_STS;
15:0	R/W		reg_arb_mask, [9]: frddr_e; [8]: toddr_e; [7]: frddr_d; [6]: frddr_c; [5]: frddr_b; [4]: frddr_a; [3]: toddr_d; [2]: toddr_c; [1]: toddr_b; [0]: toddr_a;

Table 10-297 EE\_AUDIO\_ARB\_CTRL1 0xa1

Bits	R/W	Default	Description
15:0	R/W	0x0000-0000	reg_lock_timeout_cnt; will generate exception irq if clk_cnt of locked sts greater than this register;

Table 10-298 EE\_AUDIO\_ARB\_STS 0xa8

Bits	R/W	Default	Description
31:0	R	0x0000-0000	reg_sts_sel = 0: bit31: r_locked; mean arbiter is locked; bit30: c_usr_dc_rdy; mean DDR ASYNC is ready; bit27:22: rd_cnt; mean DDR ASYNC stored read command count; bit21:16: wr_cnt; mean DDR ASYNC stored write command count; bit15:0: r_req_sel; mean arbiter grant to which request; reg_sts_sel = 1: bit31:16: c_req_sel; mean request status; bit15:0 : r_lock_clk_cnt; mean how long locked this time; reg_sts_sel = 2: bit31:16: lock_clk_cnt_max; bit15:0: lock_clk_cnt_min; reg_sts_sel = 3: bit31:28: cnt_id0; bit27:24: cnt_id1; bit23:20: cnt_id2; bit19:16: cnt_id3; bit15:12: cnt_id4; bit11:8: cnt_id5; bit7:4: cnt_id6; bit3:0: cnt_id7; reg_sts_sel = 4: bit31:28: cnt_id8; bit27:24: cnt_id9; bit23:20: cnt_id10; bit19:16: cnt_id11; bit15:12: cnt_id12; bit11:8: cnt_id13; bit7:4: cnt_id14; bit3:0: cnt_id15;

## 10.11.6 LoopBack Registers

Table 10-299 EE\_AUDIO\_LB\_A\_CTRL0 0xb0

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_lb_en, 0:disable; 1: enable;
30	R/W		reg_lb_mode, 0: out rate = in data rate; 1: out rate = loopback data rate;
29	R/W		reg_ext_signed, 0: extend bits as "0"; 1: extend bits as "msb";
28	R/W		Enable_sync_chnum; 1: start store data when ch_num can match ID;
27	R/W		chnum_sel; 0: chnum[4:0] is fixed value: 0-1-2-...31; 1: chnum[4:0] is valid_cnt; the different is bace mask set;
17:16	R/W		Reg_sts_sel, refer to REG_LB_STATUS.
15:13	R/W		reg_dat_sel, shift [m:n] to [31:0]; 0: right justified, out = { ext,[m:n]}; 1: left justified, out = {[m:n],all0}; 2: right justified, out = { ext,[m:n]}; 3: left justified, out = {[m:n],all0}; 4: right justified, out = { ext,[m:n]};
12:8	R/W		reg_dat_m_sel, the msb position in data
7:3	R/W		reg_dat_n_sel, the lsb position in data

Table 10-300 EE\_AUDIO\_LB\_A\_CTRL1 0xb1

Bits	R/W	Default	Description
29	R/W		reg_lb_ext_signed, 0: extend bits as "0"; 1: extend bits as "msb";
28	R/W		Enable_sync_chnum; 1: start store data when ch_num can match ID;
27	R/W		hold_insertion 1: hold insert data store(update) and insert 0 after original data;
26	R/W		mute_insertern 1: set insert data to mute_insert_val;
24:16			the error condition which can pull stable signal down; 24: orig_sotre_cnt_err; start store with cnt!=0; 23: insert_store_cnt_err; start store with cnt!=0; 22: orig_store_err1; start store but not first ch 21: orig_store_err2; storing but receive first ch 20: insert_store_err1; start store but not first ch 19: insert_store_err2; storing but receive first ch 18: output_en_err; start send when output is not finish; 17: pipe_cnt_err1; cnt> 3; 16: pipe_cnt_err2; cnt< 0;
15:13			reg_lb_sel, shift [m:n] to [31:0]; 0: right justified, out = {ext,[m:n]}; 1: left justified, out = {[m:n],all0}; 2: right justified, out = { ext,[m:n]}; 3: left justified, out = {[m:n],all0};

Bits	R/W	Default	Description
			4: right justified, out = { ext,[m:n]};
12:8			reg_lb_m_sel, the msb position loopback data
7:3			reg_lb_n_sel, the lsb position loopback data

Table 10-301 EE\_AUDIO\_LB\_A\_CTRL2 0xb2

Bits	R/W	Default	Description
31:28	R/W		insert_fifo_thd insert fifo start work thd
27	R/W		insert_fifo_mode 1.enable insert path fifo mode
26	R/W		insert_fifo_init 1.initial insert fifo
25	R/W		fifo_noclk_gate
24:20	R/W	0x0000-0000	data Src select: 0: tadmin_a; 1: tadmin_b; 2: tadmin_c; 3: spdifin; 4: pdmin; 5: fratv; 6: tadmin_lb; 7: loopback_a; 8: frhdmirx_spdif; 9: loopback_b; 10: spdifin_lb; 11:earc_rx_dmac 12:frhdmirx_dsd 13:resample_a 14:resample_b 15: vad; 16~31:reserved
19:16	R/W		Dat_ch_num, max channel number of data source; max is 15 (equal 16 ch)
15:0	R/W		Dat_ch_mask, 16bits match 16 ch, set 1 will sending out, set0 will drop off this ch;

Table 10-302 EE\_AUDIO\_LB\_A\_CTRL3 0xb3

Bits	R/W	Default	Description
24:20	R/W	0x0000-0000	LB Src select: 0: tadmin_a; 1: tadmin_b; 2: tadmin_c; 3: spdifin; 4: pdmin; 5: fratv; 6: tadmin_lb; 7: loopback_a; 8: frhdmirx_spdif; 9: loopback_b; 10: spdifin_lb; 11:earc_rx_dmac 12:frhdmirx_dsd 13:resample_a 14:resample_b 15: vad; 16~31:reserved
19:16	R/W		Lb_ch_num, max channel number of data source; max is 15 (equal 16 ch)
15:0	R/W		Lb_ch_mask, 16bits match 16 ch, set 1 will sending out, set0 will drop off this ch;

Table 10-303 EE\_AUDIO\_LB\_A\_DAT\_ID0 0xb4

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	data_ch7_id
23:16	R/W		data_ch6_id
15:8	R/W		data_ch5_id
7:0	R/W		data_ch4_id

Table 10-304 EE\_AUDIO\_LB\_A\_DAT\_ID1 0xb5

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	data_ch3_id
23:16	R/W		data_ch2_id
15:8	R/W		data_ch1_id
7:0	R/W		data_ch0_id

Table 10-305 EE\_AUDIO\_LB\_A\_DAT\_ID2 0xb6

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	data_ch11_id
23:16	R/W		data_ch10_id
15:8	R/W		data_ch9_id
7:0	R/W		data_ch8_id

**Table 10-306 EE\_AUDIO\_LB\_A\_DAT\_ID3 0xb7**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	data_ch15_id
23:16	R/W		data_ch14_id
15:8	R/W		data_ch13_id
7:0	R/W		data_ch12_id

**Table 10-307 EE\_AUDIO\_LB\_A\_ID0 0xb8**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	lb_data_ch7_id
23:16	R/W		lb_data_ch6_id
15:8	R/W		lb_data_ch5_id
7:0	R/W		lb_data_ch4_id

**Table 10-308 EE\_AUDIO\_LB\_A\_ID1 0xb9**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	lb_data_ch3_id
23:16	R/W		lb_data_ch2_id
15:8	R/W		lb_data_ch1_id
7:0	R/W		lb_data_ch0_id

**Table 10-309 EE\_AUDIO\_LB\_A\_ID2 0xba**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	lb_data_ch11_id
23:16	R/W		lb_data_ch10_id
15:8	R/W		lb_data_ch9_id
7:0	R/W		lb_data_ch8_id

**Table 10-310 EE\_AUDIO\_LB\_A\_ID3 0xbb**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	lb_data_ch15_id
23:16	R/W		lb_data_ch14_id



Bits	R/W	Default	Description
15:8	R/W		lb_data_ch13_id
7:0	R/W		lb_data_ch12_id

Table 10-311 EE\_AUDIO\_LB\_A\_STS 0xbc

Bits	R/W	Default	Description
31:0	R	0x0000-0000	<p>reg_sts_sel = 0: [31:0] = 0;</p> <p>reg_sts_sel = 1: [31]:pipe_cnt_err2; mean pipe cnt decr when it's 0; [30]:pipe_cnt_err1; mean pipe cnt incr when it's 3; [29:28]:pipe_cnt; current pipe cnt value; [27]:output_en_err; when output, received a new start; [26]:insert_storing_err2; it it's receiving, but received first ch; [25]:insert_storing_err1, if it's not receiving, but received data [24]:orig_storing_err2; it it's receiving, but received first ch; [23]:orig_storing_err1, if it's not receiving, but received data [22]:1st_send_done, mean finished first time send insert data; [21]:insert_storing, mean it's receiving insert data now; [20]:orig_storing, mean it's receiving insert data now; [19]:insert_store_cnt_err, when first_ch, if store_cnt is not 0, mean error, missed ch or another reason; [9]:orig_store_cnt_err, when first_ch, if store_cnt is not 0, mean error, missed ch or another reason;</p> <p>reg_sts_sel = 2: [31:0]:insert_store_cnt_debug, when start sending data, stored insert dat ch number;</p> <p>reg_sts_sel = 3: [31:0]:orig_store_cnt_debug, when start sending data, stored orig dat ch number;</p>

Table 10-312 EE\_AUDIO\_LB\_A\_CHSYNC\_CTRL\_INSERT 0xbd

Bits	R/W	Default	Description
31	R/W	0x0000-0000	bypass_n; 0: out=in; 1: enable sync mode;
30	R/W		sw_reset; 1: reset status;
29	R/W		mode_sel; 0: buf sync mode; 1: drop sync mode;
28	R/W		id_sel; 0: by int_cnt; 1: by reg_chnum_id;
27	R/W		fix_stable; 1: will not pull stable low if detect any error;
26	R/W		debug_en0; for buf sync mode only; 0: send new chnum which generated by chnum_sync; 1: send old chnum from buff;
7:0	R/W		chnum_max; if receive 8 ch data, set it to 7;

Table 10-313 EE\_AUDIO\_LB\_A\_CHSYNC\_CTRL\_ORIG 0xbe

Bits	R/W	Default	Description
31	R/W	0x0000-0000	bypass_n; 0: out=in; 1: enable sync mode;
30	R/W		sw_reset; 1: reset status;
29	R/W		mode_sel; 0: buf sync mode; 1: drop sync mode;
28	R/W		id_sel; 0: by int_cnt; 1: by reg_chnum_id;
27	R/W		fix_stable; 1: will not pull stable low if detect any error;
26	R/W		debug_en0; for buf sync mode only; 0: send new chnum which generated by chnum_sync; 1: send old chnum from buff;
7:0	R/W		chnum_max; if receive 8 ch data, set it to 7;

Table 10-314 EE\_AUDIO\_LB\_A\_CTRL4 0xbf

Bits	R/W	Default	Description
24:20	R/W	0x0000-0000	mute_insert_val

EE_AUDIO_LB_B_CTRL0	0x230
EE_AUDIO_LB_B_CTRL1	0x231
EE_AUDIO_LB_B_CTRL2	0x232
EE_AUDIO_LB_B_CTRL3	0x233
EE_AUDIO_LB_B_DAT_ID0	0x234
EE_AUDIO_LB_B_DAT_ID1	0x235
EE_AUDIO_LB_B_DAT_ID2	0x236
EE_AUDIO_LB_B_DAT_ID3	0x237
EE_AUDIO_LB_B_ID0	0x238
EE_AUDIO_LB_B_ID1	0x239
EE_AUDIO_LB_B_ID2	0x23a
EE_AUDIO_LB_B_ID3	0x23b
EE_AUDIO_LB_B_STS	0x23c
EE_AUDIO_LB_B_CHSYNC_CTRL_INSERT	0x23d
EE_AUDIO_LB_B_CHSYNC_CTRL_ORIG	0x23e
EE_AUDIO_LB_B_CTRL4	0x2bf

### 10.11.7 TDM Registers

TDMin\_B/C are the same as TDMin\_A.

TDMout\_B/C are the same as TDMout\_A.

TDMin\_LB is the same as TDMin\_A.

TDM IN & OUT are independent , so need select MCLK/SCLK/LRCLK/DATIN/DATOUT in audio after set CHIP\_PIN\_MUX.

These pins are from/to CHIP\_PIN\_MUX module.

MCLK	2	output
SCLK_MST/SLV	3	input/output
LRCLK_MST/SLV	3	input/output
DAT	32	input/output

Sources of pad\_aud\_dat\_out are:

bit	src	bit	src	bit	src	bit	src
31	0	23	TDMOUT_C bit7	15	TDMOUT_B bit7	7	TDMOUT_A bit7
30	0	22	TDMOUT_C bit6	14	TDMOUT_B bit6	6	TDMOUT_A bit6
29	0	21	TDMOUT_C bit5	13	TDMOUT_B bit5	5	TDMOUT_A bit5
28	0	20	TDMOUT_C bit4	12	TDMOUT_B bit4	4	TDMOUT_A bit4
27	0	19	TDMOUT_C bit3	11	TDMOUT_B bit3	3	TDMOUT_A bit3
26	0	18	TDMOUT_C bit2	10	TDMOUT_B bit2	2	TDMOUT_A bit2
25	0	17	TDMOUT_C bit1	9	TDMOUT_B bit1	1	TDMOUT_A bit1
24	0	16	TDMOUT_C bit0	8	TDMOUT_B bit0	0	TDMOUT_A bit0

**Table 10-315 EE\_AUDIO\_MCLK\_PAD\_CTRL0 0x3A0**

Bits	R/W	Default	Description
31	R/W	0	mclk_pad_1_en;
30	R/W	0	mclk_pad_1, force to xtal;
26:24	R/W	0	mclk_pad_1_sel: 0: mclk_a; 1: mclk_b; 2: mclk_c; 3: mclk_d; 4: mclk_e; 5: mclk_f;

Bits	R/W	Default	Description
23:16	R/W	0	Mclk_pad_1_div: the mclk_pad will div by mclk source; set 7 if need div8
15	R/W	0	mclk_pad_0_en
14	R/W	0	mclk_pad_0, force to xtal;
10:8	R/W	0	mclk_pad_0_sel: 0: mclk_a; 1: mclk_b; 2: mclk_c; 3: mclk_d; 4: mclk_e; 5: mclk_f;
7:0	R/W	0	Mclk_pad_0_div: the mclk_pad will div by mclk source; set 7 if need div8

Table 10-316 EE\_AUDIO\_MCLK\_PAD\_CTRL1 0x3A1

Bits	R/W	Default	Description
15	R/W	0	mclk_pad_2_en
14	R/W	0	mclk_pad_2, force to xtal;
10:8	R/W	0	mclk_pad_2_sel: 0: mclk_a; 1: mclk_b; 2: mclk_c; 3: mclk_d; 4: mclk_e; 5: mclk_f;
7:0	R/W	0	Mclk_pad_2_div: the mclk_pad will div by mclk source; set 7 if need div8

Table 10-317 EE\_AUDIO\_SCLK\_PAD\_CTRL0 0x3A2

Bits	R/W	Default	Description
23	R/W	0	sclk_pad_4_oen: 0: pad work as output; 1: pad work as input;
22:20	R/W	0	sclk_pad_4_sel(when it's output): 0: mst_sclk_a; 1: mst_sclk_b; 2: mst_sclk_c; 3: mst_sclk_d; 4: mst_sclk_e; 5: mst_sclk_f;
19	R/W	0	sclk_pad_3_oen: 0: pad work as output; 1: pad work as input;
18:16	R/W	0	sclk_pad_3_sel(when it's output): 0: mst_sclk_a; 1: mst_sclk_b; 2: mst_sclk_c; 3: mst_sclk_d;

Bits	R/W	Default	Description
			4: mst_sclk_e; 5: mst_sclk_f;
11	R/W	0	sclk_pad_2_oen: 0: pad work as output; 1: pad work as input;
10:8	R/W	0	sclk_pad_2_sel(when it's output): 0: mst_sclk_a; 1: mst_sclk_b; 2: mst_sclk_c; 3: mst_sclk_d; 4: mst_sclk_e; 5: mst_sclk_f;
7	R/W	0	sclk_pad_1_oen: 0: pad work as output; 1: pad work as input;
6:4	R/W	0	sclk_pad_1_sel(when it's output): 0: mst_sclk_a; 1: mst_sclk_b; 2: mst_sclk_c; 3: mst_sclk_d; 4: mst_sclk_e; 5: mst_sclk_f;
3	R/W	0	sclk_pad_0_oen: 0: pad work as output; 1: pad work as input;
2:0	R/W	0	sclk_pad_0_sel(when it's output): 0: mst_sclk_a; 1: mst_sclk_b; 2: mst_sclk_c; 3: mst_sclk_d; 4: mst_sclk_e; 5: mst_sclk_f;

**Table 10-318 EE\_AUDIO\_SCLK\_PAD\_CTRL1 0x3A3**

Bits	R/W	Default	Description
23	R/W	0	lrclk_pad_4_oen: 0: pad work as output; 1: pad work as input;
22:20	R/W	0	lrclk_pad_4_sel(when it's output): 0: mst_lrclk_a; 1: mst_lrclk_b; 2: mst_lrclk_c; 3: mst_lrclk_d; 4: mst_lrclk_e; 5: mst_lrclk_f;
19	R/W	0	lrclk_pad_3_oen: 0: pad work as output; 1: pad work as input;
18:16	R/W	0	lrclk_pad_3_sel(when it's output): 0: mst_lrclk_a; 1: mst_lrclk_b; 2: mst_lrclk_c; 3: mst_lrclk_d; 4: mst_lrclk_e; 5: mst_lrclk_f;

Bits	R/W	Default	Description
11	R/W	0	lrclk_pad_2_oen: 0: pad work as output; 1: pad work as input;
10:8	R/W	0	lrclk_pad_2_sel(when it's output): 0: mst_lrclk_a; 1: mst_lrclk_b; 2: mst_lrclk_c; 3: mst_lrclk_d; 4: mst_lrclk_e; 5: mst_lrclk_f;
7	R/W	0	lrclk_pad_1_oen: 0: pad work as output; 1: pad work as input;
6:4	R/W	0	lrclk_pad_1_sel(when it's output): 0: mst_lrclk_a; 1: mst_lrclk_b; 2: mst_lrclk_c; 3: mst_lrclk_d; 4: mst_lrclk_e; 5: mst_lrclk_f;
3	R/W	0	lrclk_pad_0_oen: 0: pad work as output; 1: pad work as input;
2:0	R/W	0	lrclk_pad_0_sel(when it's output): 0: mst_lrclk_a; 1: mst_lrclk_b; 2: mst_lrclk_c; 3: mst_lrclk_d; 4: mst_lrclk_e; 5: mst_lrclk_f;

**Table 10-319 EE\_AUDIO\_DAT\_PAD\_CTRL0 0x390**

Bits	R/W	Default	Description
28:24	R/W	0	tdmin_a dat_in bit3 sel
20:16	R/W	0	tdmin_a dat_in bit2 sel
12:8	R/W	0	tdmin_a dat_in bit1 sel
4:0	R/W	0	tdmin_a dat_in bit0 sel 0: sel TDM_D0; 1: sel TDM_D1; 2: sel TDM_D2; ... 31: sel TDM_D31;

**Table 10-320 EE\_AUDIO\_DAT\_PAD\_CTRL1 0x391**

Bits	R/W	Default	Description
28:24	R/W	0	tdmin_a dat_in bit7 sel
20:16	R/W	0	tdmin_a dat_in bit6 sel
12:8	R/W	0	tdmin_a dat_in bit5 sel
4:0	R/W	0	tdmin_a dat_in bit4 sel

**Table 10-321 EE\_AUDIO\_DAT\_PAD\_CTRL2 0x392**

Bits	R/W	Default	Description
28:24	R/W	0	tdmin_b dat_in bit3 sel
20:16	R/W	0	tdmin_b dat_in bit2 sel
12:8	R/W	0	tdmin_b dat_in bit1 sel
4:0	R/W	0	tdmin_b dat_in bit0 sel

**Table 10-322 EE\_AUDIO\_DAT\_PAD\_CTRL3 0x393**

Bits	R/W	Default	Description
28:24	R/W	0	tdmin_b dat_in bit7 sel
20:16	R/W	0	tdmin_b dat_in bit6 sel
12:8	R/W	0	tdmin_b dat_in bit5 sel
4:0	R/W	0	tdmin_b dat_in bit4 sel

**Table 10-323 EE\_AUDIO\_DAT\_PAD\_CTRL4 0x394**

Bits	R/W	Default	Description
28:24	R/W	0	tdmin_c dat_in bit3 sel
20:16	R/W	0	tdmin_c dat_in bit2 sel
12:8	R/W	0	tdmin_c dat_in bit1 sel
4:0	R/W	0	tdmin_c dat_in bit0 sel

**Table 10-324 EE\_AUDIO\_DAT\_PAD\_CTRL5 0x395**

Bits	R/W	Default	Description
28:24	R/W	0	tdmin_c dat_in bit7 sel
20:16	R/W	0	tdmin_c dat_in bit6 sel
12:8	R/W	0	tdmin_c dat_in bit5 sel
4:0	R/W	0	tdmin_c dat_in bit4 sel

**Table 10-325 EE\_AUDIO\_DAT\_PAD\_CTRL6 0x396**

Bits	R/W	Default	Description
28:24	R/W	0	TDM_D3 out src sel:
20:16	R/W	0	TDM_D2 out src sel:

Bits	R/W	Default	Description
12:8	R/W	0	TDM_D1 out src sel:
4:0	R/W	0	TDM_D0 out src sel: 0: tdmout_a lane0; 1: tdmout_a lane1; ... 7: tdmout_b lane7; 8: tdmout_b lane0; 9: tdmout_b lane1; ... 15: tdmout_b lane7; 16: tdmout_c lane0; 17: tdmout_c lane1; ... 23: tdmout_c lane7;

Table 10-326 EE\_AUDIO\_DAT\_PAD\_CTRL7 0x397

Bits	R/W	Default	Description
28:24	R/W	0	TDM_D7 out src sel:
20:16	R/W	0	TDM_D6 out src sel:
12:8	R/W	0	TDM_D5 out src sel:
4:0	R/W	0	TDM_D4 out src sel:

Table 10-327 EE\_AUDIO\_DAT\_PAD\_CTRL8 0x398

Bits	R/W	Default	Description
28:24	R/W	0	TDM_D11 out src sel:
20:16	R/W	0	TDM_D10 out src sel:
12:8	R/W	0	TDM_D9 out src sel:
4:0	R/W	0	TDM_D8 out src sel:

Table 10-328 EE\_AUDIO\_DAT\_PAD\_CTRL9 0x399

Bits	R/W	Default	Description
28: 24	R/W	0	TDM_D15 out src sel:
20:16	R/W	0	TDM_D14 out src sel:
12:8	R/W	0	TDM_D13 out src sel:
4:0	R/W	0	TDM_D12 out src sel:

Table 10-329 EE\_AUDIO\_DAT\_PAD\_CTRLA 0x39a

Bits	R/W	Default	Description
28:24	R/W	0	TDM_D19 out src sel:
20:16	R/W	0	TDM_D18 out src sel:



Bits	R/W	Default	Description
12:8	R/W	0	TDM_D17 out src sel:
4:0	R/W	0	TDM_D16 out src sel:

Table 10-330 EE\_AUDIO\_DAT\_PAD\_CTRLB 0x39b

Bits	R/W	Default	Description
28:24	R/W	0	TDM_D23 out src sel:
20:16	R/W	0	TDM_D22 out src sel:
12:8	R/W	0	TDM_D21 out src sel:
4:0	R/W	0	TDM_D20 out src sel:

Table 10-331 EE\_AUDIO\_DAT\_PAD\_CTRLC 0x39c

Bits	R/W	Default	Description
28:24	R/W	0	TDM_D27 out src sel:
20:16	R/W	0	TDM_D26 out src sel:
12:8	R/W	0	TDM_D25 out src sel:
4:0	R/W	0	TDM_D24 out src sel:

Table 10-332 EE\_AUDIO\_DAT\_PAD\_CTRLD 0x39d

Bits	R/W	Default	Description
28:24	R/W	0	TDM_D31 out src sel:
20:16	R/W	0	TDM_D30 out src sel:
12:8	R/W	0	TDM_D29 out src sel:
4:0	R/W	0	TDM_D28 out src sel:

Table 10-333 EE\_AUDIO\_DAT\_PAD\_CTRL E 0x39e

Bits	R/W	Default	Description
31	R/W	0	TDM_D31 oen sel: 0: reg_oen_val bit31; 1: from TDMOUT;
30	R/W	0	TDM_D30 oen sel: 0: reg_oen_val bit30; 1: from TDMOUT;
...	...	...	...
0	R/W	0	TDM_D0 oen sel: 0: reg_oen_val bit0; 1: from TDMOUT;

Table 10-334 EE\_AUDIO\_DAT\_PAD\_CTRLF 0x39f

Bits	R/W	Default	Description
31:0	R/W	0	reg_oen_val; 0: output; 1: input; bit31: TDM_D31 bit30: TDM_D30 ... bit1: TDM_D1 bit0: TDM_D0

Table 10-335 EE\_AUDIO\_TDMIN\_A\_CTRL 0xc0

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_work_enable 0:disable; 1:enable;
30	R/W		reg_i2s_mode,0:tdm mode; 1: i2s mode;
29	R/W		reg_rst_afifo_out_n, reset afifo out side; need set to 1 before reg_rst_afifo_in_n;
28	R/W		reg_rst_afifo_in_n, reset afifo in side; need set 1 after set reg_rst_afifo_out_n to 1;
26	R/W		reg_tdm_in_resync; force TDMIN stable signal to low, then TDMIN need resync to stable; the condition of stable to high is : detected rise edge of WS
25	R/W		reg_tdm_in_rev_ws, revert ws(lrclk); 0 :disable; 1: enable;
24	R/W		reg_tdm_in_rev_dat, revert data; 0:disable; 1:enable;
23:20	R/W		select tdm src; 0 tdm_a dat_in 1 tdm_b dat_in 2 tdm_c dat_in 3 0 4 hdmix 5 acodec_adc 6 0 7 0 8 0 9 0 10 0 11 0 12 tdmout_a dat_out 13 tdmout_b dat_out 14 tdmout_c dat_out 15 0
18:16	R/W		reg_tdm_in_bit_skew, add delay to ws or data for skew modification;
15:13	R/W		reg_stable_mask, each bit can enable one condition to pull stable signal to low. bit15: if afifo overflow, will pull stable to low; bit14: if the slot_cnt != reg_tdm_slot_num, will pull stable to low; bit13: if the bit_cnt != reg_tdm_bit_num, will pull stable to low;
12:8	R/W		reg_tdm_slot_num for I2S, it's fixed 0x1. for TDM, the max is 31 which mean 32 slots
7:6	R/W		reg_tdm_chnum_mode; 0:chnum[7:0] = valid_cnt[7:0] 1:chnum[7:0] = cyc_cnt[3:0] 2:chnum[7:0] = slot_cnt[4:1],cyc_cnt[0] 3:chnum[7:0] = slot_cnt[4:1],cyc_cnt[3:0]

Bits	R/W	Default	Description
5	R/W		reg_lsb_first, 0: store first bit received to data_store[0]; 1: store first bit received to data_store[31];
4:0	R/W		reg_tadmin_bit_num, bitwidth of each slot, if slot is 16bits, set this register to 15;

Table 10-336 EE\_AUDIO\_TDMIN\_A\_SWAP0 0xc1

Bits	R/W	Default	Description
31:28	R/W	0x0000-0000	ch7_sel 0: lane0 left channel; 1: lane0 right channel; 2: lane1 left channel; 3: lane1 right channel; 4: lane2 left channel; 5: lane2 right channel; 6: lane3 left channel; 7: lane3 right channel; 8: lane4 left channel; 9: lane4 right channel; 10: lane5 left channel; 11: lane5 right channel; 12: lane6 left channel; 13: lane6 right channel; 14: lane7 left channel; 15: lane7 right channel;
27:24	R/W		ch6_sel
23:20	R/W		ch5_sel
19:16	R/W		ch4_sel
15:12	R/W		ch3_sel
11:8	R/W		ch2_sel
7:4	R/W		ch1_sel
3:0	R/W		ch0_sel

Table 10-337 EE\_AUDIO\_TDMIN\_A\_SWAP1 0x260

Bits	R/W	Default	Description
31:28	R/W	0x0000-0000	ch15_sel
27:24	R/W		ch14_sel
23:20	R/W		ch13_sel
19:16	R/W		ch12_sel
15:12	R/W		ch11_sel
11:8	R/W		ch10_sel
7:4	R/W		ch9_sel
3:0	R/W		ch8_sel

**Table 10-338 EE\_AUDIO\_TDMIN\_A\_MASK0 0xc2**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane0_mask, mask each channel in lane0, max is 32 ch;

**Table 10-339 EE\_AUDIO\_TDMIN\_A\_MASK1 0xc3**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane1_mask, mask each channel in lane1, max is 32 ch;

**Table 10-340 EE\_AUDIO\_TDMIN\_A\_MASK2 0xc4**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane2_mask, mask each channel in lane2, max is 32 ch;

**Table 10-341 EE\_AUDIO\_TDMIN\_A\_MASK3 0xc5**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane3_mask, mask each channel in lane3, max is 32 ch;

**Table 10-342 EE\_AUDIO\_TDMIN\_A\_MASK4 0x261**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane4_mask, mask each channel in lane4, max is 32 ch;

**Table 10-343 EE\_AUDIO\_TDMIN\_A\_MASK5 0x262**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane5_mask, mask each channel in lane5, max is 32 ch;

**Table 10-344 EE\_AUDIO\_TDMIN\_A\_MASK6 0x263**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane6_mask, mask each channel in lane6, max is 32 ch;

**Table 10-345 EE\_AUDIO\_TDMIN\_A\_MASK7 0x264**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane7_mask, mask each channel in lane7, max is 32 ch;

**Table 10-346 EE\_AUDIO\_TDMIN\_A\_STAT 0xc6**

Bits	R/W	Default	Description
31:29	R	0x0000-0000	afifo_cnt[2:0]
28:24	R		r_input_slot_cnt_max, the maxnum of slot_cnt
23:16	R		r_input_slot_cnt_min, the minnum of slot_cnt
14	R		overflow_flag, overflow flag of afifo
13:10	R		max_fifo_cnt, the maxnum of afifo_cnt
9:5	R		r_input_bit_cnt_max, the maxnum of bit_cnt
4:0	R		r_input_bit_cnt_min, the minnum of bit_cnt

**Table 10-347 EE\_AUDIO\_TDMIN\_A\_MUTE\_VAL 0xc7**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_tmdin_a_mute_val, will replace received data if set mute = 1 , the channel value

**Table 10-348 EE\_AUDIO\_TDMIN\_A\_MUTE0 0xc8**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane0_mute, mute each channel in lane0, max is 32 ch;

**Table 10-349 EE\_AUDIO\_TDMIN\_A\_MUTE1 0xc9**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane1_mute, mute each channel in lane1, max is 32 ch;

**Table 10-350 EE\_AUDIO\_TDMIN\_A\_MUTE2 0xca**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane2_mute, mute each channel in lane2, max is 32 ch;

**Table 10-351 EE\_AUDIO\_TDMIN\_A\_MUTE3 0xcb**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane3_mute, mute each channel in lane3, max is 32 ch;

**Table 10-352 EE\_AUDIO\_TDMIN\_A\_MUTE4 0x265**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane4_mute, mute each channel in lane4, max is 32 ch;

**Table 10-353 EE\_AUDIO\_TDMIN\_A\_MUTE5 0x266**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane5_mute, mute each channel in lane5, max is 32 ch;

**Table 10-354 EE\_AUDIO\_TDMIN\_A\_MUTE6 0x267**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane6_mute, mute each channel in lane6, max is 32 ch;

**Table 10-355 EE\_AUDIO\_TDMIN\_A\_MUTE7 0x268**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane7_mute, mute each channel in lane7, max is 32 ch;

EE_AUDIO_TDMIN_B_CTRL	0xd0
EE_AUDIO_TDMIN_B_SWAP0	0xd1
EE_AUDIO_TDMIN_B_SWAP1	0x270
EE_AUDIO_TDMIN_B_MASK0	0xd2
EE_AUDIO_TDMIN_B_MASK1	0xd3
EE_AUDIO_TDMIN_B_MASK2	0xd4
EE_AUDIO_TDMIN_B_MASK3	0xd5
EE_AUDIO_TDMIN_B_MASK4	0x271
EE_AUDIO_TDMIN_B_MASK5	0x272
EE_AUDIO_TDMIN_B_MASK6	0x273
EE_AUDIO_TDMIN_B_MASK7	0x274
EE_AUDIO_TDMIN_B_STAT	0xd6
EE_AUDIO_TDMIN_B_MUTE_VAL	0xd7
EE_AUDIO_TDMIN_B_MUTE0	0xd8
EE_AUDIO_TDMIN_B_MUTE1	0xd9
EE_AUDIO_TDMIN_B_MUTE2	0xda
EE_AUDIO_TDMIN_B_MUTE3	0xdb
EE_AUDIO_TDMIN_B_MUTE4	0x275
EE_AUDIO_TDMIN_B_MUTE5	0x276
EE_AUDIO_TDMIN_B_MUTE6	0x277
EE_AUDIO_TDMIN_B_MUTE7	0x278

---

EE_AUDIO_TDMIN_C_CTRL	0xe0
EE_AUDIO_TDMIN_C_SWAP0	0xe1
EE_AUDIO_TDMIN_C_SWAP1	0x280
EE_AUDIO_TDMIN_C_MASK0	0xe2
EE_AUDIO_TDMIN_C_MASK1	0xe3
EE_AUDIO_TDMIN_C_MASK2	0xe4
EE_AUDIO_TDMIN_C_MASK3	0xe5
EE_AUDIO_TDMIN_C_MASK4	0x281
EE_AUDIO_TDMIN_C_MASK5	0x282
EE_AUDIO_TDMIN_C_MASK6	0x283
EE_AUDIO_TDMIN_C_MASK7	0x284
EE_AUDIO_TDMIN_C_STAT	0xe6
EE_AUDIO_TDMIN_C_MUTE_VAL	0xe7
EE_AUDIO_TDMIN_C_MUTE0	0xe8
EE_AUDIO_TDMIN_C_MUTE1	0xe9
EE_AUDIO_TDMIN_C_MUTE2	0xea
EE_AUDIO_TDMIN_C_MUTE3	0xeb
EE_AUDIO_TDMIN_C_MUTE4	0x285
EE_AUDIO_TDMIN_C_MUTE5	0x286
EE_AUDIO_TDMIN_C_MUTE6	0x287
EE_AUDIO_TDMIN_C_MUTE7	0x288
EE_AUDIO_TDMIN_LB_CTRL	0xf0
EE_AUDIO_TDMIN_LB_SWAP0	0xf1
EE_AUDIO_TDMIN_LB_SWAP1	0x290
EE_AUDIO_TDMIN_LB_MASK0	0xf2
EE_AUDIO_TDMIN_LB_MASK1	0xf3
EE_AUDIO_TDMIN_LB_MASK2	0xf4
EE_AUDIO_TDMIN_LB_MASK3	0xf5
EE_AUDIO_TDMIN_LB_MASK4	0x291
EE_AUDIO_TDMIN_LB_MASK5	0x292
EE_AUDIO_TDMIN_LB_MASK6	0x293
EE_AUDIO_TDMIN_LB_MASK7	0x294
EE_AUDIO_TDMIN_LB_STAT	0xf6
EE_AUDIO_TDMIN_LB_MUTE_VAL	0xf7
EE_AUDIO_TDMIN_LB_MUTE0	0xf8
EE_AUDIO_TDMIN_LB_MUTE1	0xf9
EE_AUDIO_TDMIN_LB_MUTE2	0xfa

EE_AUDIO_TDMIN_LB_MUTE3	0xfb
EE_AUDIO_TDMIN_LB_MUTE4	0x295
EE_AUDIO_TDMIN_LB_MUTE5	0x296
EE_AUDIO_TDMIN_LB_MUTE6	0x297
EE_AUDIO_TDMIN_LB_MUTE7	0x298

Table 10-356 EE\_AUDIO\_TDMOUT\_A\_CTRL0 0x140

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_work_enable, 0:disable; 1:enable;
30	R/W		reg_rst_req_frddr, reset ref_frddr in TDMOUT;
29	R/W		reg_rst_afifo_out_n, reset afifo out side;
28	R/W		reg_rst_afifo_in_n, reset afifo in side; need set 1 after set reg_rst_afifo_out_n to 1;
19:15	R/W		reg_tdm_init_bitnum, initial count value of bitcnt
14:10	R/W		reg_tdm_init_slotnum, initial count value of slotcnt
9:5	R/W		reg_tdmout_slot_num , max value of slotcnt; if each frame has 8 slots, set it to 7;
4:0	R/W		reg_tdmout_bit_num ,max value of bitcnt; if each slot has 16bits, set it to 15;

Table 10-357 EE\_AUDIO\_TDMOUT\_A\_CTRL1 0x141

Bits	R/W	Default	Description
31	R/W	0x0000-0000	eq_drc_sel; 1: select eq_drc output;
30	R/W		reg_debug_en,0:disable debug feature; 1: enable;
29	R/W		reg_out_lsb_first, 0: msb first; 1: lsb first;
28	R/W		reg_rev_ws_in, revert ws; 0: disable; 1: enable;
27	R/W		reg_rev_dat, revert data; 0: disable; 1: enable;
26:24	R/W		reg_frddr_sel, 0:frddr_A;1:frddr_B;2:frddr_C; 3:frddr_D;4:frddr_E
23:16	R/W		reg_wait_cnt, wait some time when enable set to 1;then start request data from frddr;
15:14	R/W		reg_gain_shift; 0: data * 1; 1: data * 2; 2: data * 4; 3: data * 8;
12:8	R/W		reg_frddr_msb, msb position of data
6:4	R/W		reg_frddr_type, 0: split 64bits ddr data to 8 sample, each sample need 8 bits; if bitwidth < 8, left-justified; 1: split 64bits ddr data to 4 sample, each sample need 16 bits; if bitwidth < 16, left-justified ; 2: split 64bits ddr data to 4 sample, each sample need 16 bits; if bitwidth < 16, right-justified ; 3: split 64bits ddr data to 2 sample, each sample need 32 bits; if bitwidth < 32, left-justified; 4: split 64bits ddr data to 2 sample, each sample need 32 bits; if bitwidth < 32, right-justified;



Bits	R/W	Default	Description
3	R/W		reg_keep_req_ddr_init; set 0 will stop sending data if req_ddr stopped by init_done signal pull down.(refer to reg_clr_by_init)
2	R/W		reg_sts_sel;

For position of sample in DDR:

ddr_data [7:0] (byte)								
type	byte7	byte6	byte5	byte4	byte3	byte2	byte1	byte0
0	s7	s6	s5	s4	s3	s2	s1	s0
	s15	s14	s13	s12	s11	s10	s9	s8
1/2	s3		s2		s1		s0	
	s7		s6		s5		s4	
3/4	s1				s0			
	s3				s2			

For encoder, it will start output[31] first, then output[30], finished at output[32-bit\_num];

If lsb\_first, will start s[0], then s[1], finished at s[bit\_num+1];

output [31:0] (bit)																																											
type	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0											
0	s[7]	s[6]	s[5]	s[4]	s[3]	s[2]	s[1]	s[0]	0																																		
1	s[15]	s[14]	s[13]	s[12]	s[11]	s[10]	s[9]	s[8]	s[7]	s[6]	s[5]	s[4]	s[3]	s[2]	s[1]	s[0]	0															0											
2	s[msb]	s[msb-1]	--		s[1]	s[0]	0																																				
3	s[31]	s[30]	s[29]	s[28]	s[27]	s[26]	s[25]	s[24]	s[23]	s[22]	s[21]	s[20]	s[19]	s[18]	s[17]	s[16]	s[15]	s[14]	s[13]	s[12]	s[11]	s[10]	s[9]	s[8]	s[7]	s[6]	s[5]	s[4]	s[3]	s[2]	s[1]	s[0]											
4	s[msb]	s[msb-1]	--																		s[1]	s[0]	0																				

Table 10-358 EE\_AUDIO\_TDMOUT\_A\_CTRL2 0x2a0

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	reg_clr_by_init; when FRDDR is disable, its init_done will pull down, then need clear the status of REQ_FRDDR. 31: clear init_finish; 30:clear data_buff; 29:clear gain related; 28:clear DDR_REQ; 27:clear ack_masked(mean received first init_done); 26:clear afifo in; 25:clear afifo out ; 24:clear bit_cnt/slot_cnt/fit_num/wait_cnt;
7:0	R/W		reg_tdm_lr_mix, [7]: mix l7 and r7; [6]: mix l6 and r6; [5]: mix l5 and r5; [4]: mix l4 and r4; [3]: mix l3 and r3; [2]: mix l2 and r2; [1]: mix l1 and r1; [0]: mix l0 and r0;

Table 10-359 EE\_AUDIO\_TDMOUT\_A\_SWAP0 0x142

Bits	R/W	Default	Description
31:28	R/W	0x0000-0000	ch7_sel 0: lane0 left channel; 1: lane0 right channel; 2: lane1 left channel; 3: lane1 right channel; 4: lane2 left channel; 5: lane2 right channel; 6: lane3 left channel; 7: lane3 right channel; 8: lane4 left channel; 9: lane4 right channel; 10: lane5 left channel; 11: lane5 right channel; 12: lane6 left channel; 13: lane6 right channel; 14: lane7 left channel; 15: lane7 right channel;
27:24	R/W		ch6_sel
23:20	R/W		ch5_sel
19:16	R/W		ch4_sel
15:12	R/W		ch3_sel
11:8	R/W		ch2_sel
7:4	R/W		ch1_sel
3:0	R/W		ch0_sel

Table 10-360 EE\_AUDIO\_TDMOUT\_A\_SWAP1 0x2a1

Bits	R/W	Default	Description
31:28	R/W	0x0000-0000	ch15_sel
27:24	R/W		ch14_sel
23:20	R/W		ch13_sel
19:16	R/W		ch12_sel
15:12	R/W		ch11_sel
11:8	R/W		ch10_sel
7:4	R/W		ch9_sel
3:0	R/W		ch8_sel

Table 10-361 EE\_AUDIO\_TDMOUT\_A\_MASK0 0x143

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane0_mask, mask each channel in lane0, max is 32 ch;

**Table 10-362 EE\_AUDIO\_TDMOUT\_A\_MASK1 0x144**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane1_mask, mask each channel in lane1, max is 32 ch;

**Table 10-363 EE\_AUDIO\_TDMOUT\_A\_MASK2 0x145**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane2_mask, mask each channel in lane2, max is 32 ch;

**Table 10-364 EE\_AUDIO\_TDMOUT\_A\_MASK3 0x146**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane3_mask, mask each channel in lane3, max is 32 ch;

**Table 10-365 EE\_AUDIO\_TDMOUT\_A\_MASK4 0x2a4**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane4_mask, mask each channel in lane4, max is 32 ch;

**Table 10-366 EE\_AUDIO\_TDMOUT\_A\_MASK5 0x2a5**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane5_mask, mask each channel in lane5, max is 32 ch;

**Table 10-367 EE\_AUDIO\_TDMOUT\_A\_MASK6 0x2a6**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane6_mask, mask each channel in lane6, max is 32 ch;

**Table 10-368 EE\_AUDIO\_TDMOUT\_A\_MASK7 0x2a7**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane7_mask, mask each channel in lane7, max is 32 ch;

**Table 10-369 EE\_AUDIO\_TDMOUT\_A\_MASK\_VAL 0x14f**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_tdmout_mask_val, when masked, the channel value

**Table 10-370 EE\_AUDIO\_TDMOUT\_A\_STAT 0x147**

Bits	R/W	Default	Description
31:0	R	0x0000-0000	if reg_sts_sel = 1, it's reg_cur_gain_val; if reg_sts_sel = 0, it's another sts as blow: bit31:29 : fifo_cnt, afifo cnt bit28 : up_error, change to 1 if overflow bit27:24 : req_frdd_fsm_stat bit21 : r_slot_cnt_err bit20 : r_bit_cnt_err bit16:12 : r_max_slot_cnt bit8:4 : r_max_bit_cnt bit3 : r_out_en bit2 : r_out_en_pre bit1 : r_first_fs bit0 : c_frdd_init_finish

**Table 10-371 EE\_AUDIO\_TDMOUT\_A\_GAIN0 0x148**

Bits	R/W	Default	Description
31:24	R/W	0xFF	gain_ch3
23:16	R/W	0xFF	gain_ch2
15:8	R/W	0xFF	gain_ch1
7:0	R/W	0xFF	gain_ch0

**Table 10-372 EE\_AUDIO\_TDMOUT\_A\_GAIN1 0x149**

Bits	R/W	Default	Description
31:24	R/W	0xFF	gain_ch7
23:16	R/W	0xFF	gain_ch6
15:8	R/W	0xFF	gain_ch5
7:0	R/W	0xFF	gain_ch4

**Table 10-373 EE\_AUDIO\_TDMOUT\_A\_GAIN2 0x2a2**

Bits	R/W	Default	Description
31:24	R/W	0xFF	gain_ch11
23:16	R/W	0xFF	gain_ch10
15:8	R/W	0xFF	gain_ch9
7:0	R/W	0xFF	gain_ch8

**Table 10-374 EE\_AUDIO\_TDMOUT\_A\_GAIN3 0x2a3**

Bits	R/W	Default	Description
31:24	R/W	0xFF	gain_ch15
23:16	R/W	0xFF	gain_ch14

Bits	R/W	Default	Description
15:8	R/W	0xFF	gain_ch13
7:0	R/W	0xFF	gain_ch12

**Table 10-375 EE\_AUDIO\_TDMOUT\_A\_MUTE\_VAL 0x14a**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_tmdin_a_mute_val, when mute , the channel value

**Table 10-376 EE\_AUDIO\_TDMOUT\_A\_MUTE0 0x14b**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane0_mute, mute each channel in lane0, max is 32 ch;

**Table 10-377 EE\_AUDIO\_TDMOUT\_A\_MUTE1 0x14c**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane1_mute, mute each channel in lane1, max is 32 ch;

**Table 10-378 EE\_AUDIO\_TDMOUT\_A\_MUTE2 0x14d**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane2_mute, mute each channel in lane2 max is 32 ch;

**Table 10-379 EE\_AUDIO\_TDMOUT\_A\_MUTE3 0x14e**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane3_mute, mute each channel in lane3, max is 32 ch;

**Table 10-380 EE\_AUDIO\_TDMOUT\_A\_MUTE4 0x2a8**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane4_mute, mute each channel in lane4, max is 32 ch;

**Table 10-381 EE\_AUDIO\_TDMOUT\_A\_MUTE5 0x2a9**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane5_mute, mute each channel in lane5, max is 32 ch;

**Table 10-382 EE\_AUDIO\_TDMOUT\_A\_MUTE6 0x2aa**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane6_mute, mute each channel in lane6 max is 32 ch;

**Table 10-383 EE\_AUDIO\_TDMOUT\_A\_MUTE7 0x2ab**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_lane7_mute, mute each channel in lane7, max is 32 ch;

**Table 10-384 EE\_AUDIO\_TDMOUT\_A\_GAIN\_EN 0x2ac**

Bits	R/W	Default	Description
31:16	R/W	0x0000-0000	
15	R/W	0x0000-0000	ch15 gain_en; 1: out = in * gain_ch15; 0: out = in;
14	R/W	0x0000-0000	ch14 gain_en; 1: out = in * gain_ch14; 0: out = in;
13	R/W	0x0000-0000	ch13 gain_en; 1: out = in * gain_ch13; 0: out = in;
12	R/W	0x0000-0000	ch12 gain_en; 1: out = in * gain_ch12; 0: out = in;
11	R/W	0x0000-0000	ch11 gain_en; 1: out = in * gain_ch11; 0: out = in;
10	R/W	0x0000-0000	ch10 gain_en; 1: out = in * gain_ch10; 0: out = in;
9	R/W	0x0000-0000	ch9 gain_en; 1: out = in * gain_ch9; 0: out = in;
8	R/W	0x0000-0000	ch8 gain_en; 1: out = in * gain_ch8; 0: out = in;
7	R/W	0x0000-0000	ch7 gain_en; 1: out = in * gain_ch7; 0: out = in;
6	R/W	0x0000-0000	ch6 gain_en; 1: out = in * gain_ch6; 0: out = in;
5	R/W	0x0000-0000	ch5 gain_en; 1: out = in * gain_ch5; 0: out = in;
4	R/W	0x0000-0000	ch4 gain_en; 1: out = in * gain_ch4; 0: out = in;
3	R/W	0x0000-0000	ch3 gain_en; 1: out = in * gain_ch3; 0: out = in;
2	R/W	0x0000-0000	ch2 gain_en; 1: out = in * gain_ch2; 0: out = in;
1	R/W	0x0000-0000	ch1 gain_en; 1: out = in * gain_ch1; 0: out = in;
0	R/W	0x0000-0000	ch0 gain_en; 1: out = in * gain_ch0; 0: out = in;

Table 10-385 EE\_AUDIO\_TDMOUT\_A\_GAIN\_CTRL 0x2ad

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_gain_step_chg; 0: the current_gain will follow direct to reg_gain change; 1: the current_gain will follow step by step to reg_gain change; etc: change reg_gain from 40 to 50, and gain_step = 3; the current_gain will change as: 40-> 43-> 46-> 49-> 50;
25:24	R/W		reg_cur_gain_sel; 0: cur_gain_sts = ch3,ch2,ch1,ch0; 1: cur_gain_sts = ch7,ch6,ch5,ch4; 2: cur_gain_sts = ch11,ch10,ch9,ch8; 3: cur_gain_sts = ch15,ch14,ch13,ch12;
23:16	R/W		reg_gain_step; modified gain value each time;
15:0	R/W		reg_gain_updt_timer; the period of each change, unit is FS;

### 10.11.8 SPDIF Registers

Table 10-386 EE\_AUDIO\_SPDIFIN\_CTRL0 0x100

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_work_enable, 0: disable; 1:enable;
30	R/W		reg_chnum_sel, 0: ch_num = 0~383(include frame cnt); 1: ch_num = 0~1(only L/R);
29	R/W		reg_rst_afifo_out_n, reset afifo out side;
28	R/W		reg_rst_afifo_in_n, reset afifo in side; need set 1 after set reg_rst_afifo_out_n to 1;
27	R/W		reg_debug_en, 0:disable debug; 1: enable;
26	R/W		reg_exception_sts_clr;
25	R/W		reg_findpapb_en, 0: disable NonPCM mode; 1: enable;
24	R/W		reg_width_sel, 0: detect sample mode by max_width;1: detect sample mode by min_width;
23:12	R/W		reg_nonpcm2pcm_th, when detected NonPcm mode;if long time (z_cnt > = th) didn't detect PaPb again, will generate interrupt to SW: now changed to PCM mode;
11:8	R/W		reg_ch_status_sel, For EE_AUDIO_SPDIFIN_STAT1
7	R/W		reg_mute_l, 0: disable ; 1: mute channel L;
6	R/W		reg_mute_r, 0: disable ; 1: mute channel R;
5:4	R/W		reg_spdifin_src_sel, 0: PAD of spdifin;1: spdifout;
3	R/W		reg_check_valid, 0: disable valid check ; 1: enable;
2	R/W		reg_check_parity, 0: disable parity check ; 1: enable;

Bits	R/W	Default	Description
1	R/W		reg_invert_data, 0: disable; 1: invert [27:4] to [4:27];
0	R/W		reg_spdifin_phase, 0: disable invert; 1: enable;

**Table 10-387 EE\_AUDIO\_SPDIFIN\_CTRL1 0x101**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_force_sample_mode, 0: auto detect sample mode; 1: force a fixed sample mode;
30:28	R/W		reg_sample_mode
27:20	R/W		reg_interrupt_mask, mask each interrupt;
19:0	R/W		reg_base_timer, define a base timer to detect sample mode changed; need *256;

**Table 10-388 EE\_AUDIO\_SPDIFIN\_CTRL2 0x102**

Bits	R/W	Default	Description
29:20	R/W	0x0000-0000	reg_sample_mode0_timer_th
19:10	R/W		reg_sample_mode1_timer_th
9:0	R/W		reg_sample_mode2_timer_th

**Table 10-389 EE\_AUDIO\_SPDIFIN\_CTRL3 0x103**

Bits	R/W	Default	Description
29:20	R/W	0x0000-0000	reg_sample_mode3_timer_th
19:10	R/W		reg_sample_mode4_timer_th
9:0	R/W		reg_sample_mode5_timer_th

**Table 10-390 EE\_AUDIO\_SPDIFIN\_CTRL4 0x104**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	reg_sample_mode0_timer
23:16	R/W		reg_sample_mode1_timer
15:8	R/W		reg_sample_mode2_timer
7:0	R/W		reg_sample_mode3_timer



Table 10-391 EE\_AUDIO\_SPDIFIN\_CTRL5 0x105

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	reg_sample_mode4_timer
23:16	R/W		reg_sample_mode5_timer
15:8	R/W		reg_sample_mode6_timer

Table 10-392 EE\_AUDIO\_SPDIFIN\_CTRL6 0x106

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	Reg_clr_internal_sts[7:0]; 7: clear valid bit status; 6: clear parity bit status; 5: clear ch status; 4: clear z_cnt status; 3: clear find_nonpcm status; 2: clear pd_data status; 1: clear pc_data status; 0: clear find_papb status;
23:16	R/W		Reg_clr_interrupt[7:0] for each bit of irq_status[7:0];
14	R/W		enable send out find_z
13	R/W		irq_mode; 0: pulse; 1: level;
12	R/W		update_sample_mode_filter_en 0: by 1 width; 1: by average of 4 width;
8	R/W		reg_papb_ext_sync, 1: add ext "0" sync check for papb; 0: disable;
7:0	R/W		reg_papb_ext_mask, mask 8 channel "0" sync

Table 10-393 EE\_AUDIO\_SPDIFIN\_STAT0 0x107

Bits	R/W	Default	Description
31	R/W	0x0000-0000	Valid bit
30:28	R/W		r_sample_mode, current sample mode;
27:18	R/W		r_width_min, the min width of two edge;
17:8	R/W		r_width_max, the max width of two edge;
7:0	R/W		r_interrupt_status, interrupt status, need clear by reg_clk_interrupt; [7]: find PaPb; [6]: valid changed; [5]: find nonpcm to pcm ( reg_nonpcm2pcm_th ) ; [4]: find Pc or Pd changed; [3]: find CH status changed; [2]: find sample mode changed; [1]: find parity error; [0]: find overflow;

Table 10-394 EE\_AUDIO\_SPDIFIN\_STAT1 0x108

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel status, by reg_ch_status_sel; reg_ch_status_sel[3]: 0: channel A; 1: channel B; reg_ch_status_sel[2:0]: 0: ch_status[31:0]; 1: ch_status[63:32]; 2: ch_status[95:64]; 3: ch_status[127:96]; 4: ch_status[159:128]; 5: ch_status[191:160]; 6: pc[15:0],pd[15:0];

Table 10-395 EE\_AUDIO\_SPDIFIN\_STAT2 0x109

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	debug status, by reg_ch_status_sel; 0: r_z_width, the width of two preamble Z; 1: {16'd0, frame_cnt_min[7:0], frame_cnt_max[7:0]}; 2: {6'd0,width_min[9:0],2'd0,afifo_cnt[3:0], width_max[9:0]};

Table 10-396 EE\_AUDIO\_SPDIFIN\_MUTE\_VAL 0x10a

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_spdifin_mute_val, when muted, the channel value

Table 10-397 EE\_AUDIO\_SPDIFIN\_CTRL7 0x10b

Bits	R/W	Default	Description
31	R/W	0x0000-0000	hold_tri_sample 0: stop capture if received triple 0 or 1;
30	R/W		add_ch_r; when unstable, spdifin will stop send data; set this register to 1 will check current status, if missed ch_r, will add one more ch_r which value is 0;
27:20	R/W		stable_mask; condition of pull stable low; 27: xz_err; 26: afifo overflow; 25: cyc_err2; sample width < reg_stable_cyc_min 24: cyc_err1; sample width > reg_stable_cyc_max; 23: parity_err; 22: bit_cnt_err2; > 64 21: bit_cnt_err1; != 64 20: xyz_err
19:16	R/W		stable_zcnt; the condition to pull stable high
15:8	R/W		stable_cyc_min
7:0	R/W		stable_cyc_max

**Table 10-398 EE\_AUDIO\_SPDIFOUT\_STAT 0x120**

Bits	R/W	Default	Description
31:0	R	0x0000-0000	reg_sts_sel = 0: bit7:5: fifo_cnt, afifo cnt; bit4: up_error, change to 1 if overflow bit3:0: req_frdd_fsm_stat reg_sts_sel = 1: cur_gain_val from REQ_FRDDR;

**Table 10-399 EE\_AUDIO\_SPDIFOUT\_GAIN0 0x121**

Bits	R/W	Default	Description
31:24	R/W	0xFF	gain_ch3
23:16	R/W	0xFF	gain_ch2
15:8	R/W	0xFF	gain_ch1
7:0	R/W	0xFF	gain_ch0

**Table 10-400 EE\_AUDIO\_SPDIFOUT\_GAIN1 0x122**

Bits	R/W	Default	Description
31:24	R/W	0xFF	gain_ch7
23:16	R/W	0xFF	gain_ch6
15:8	R/W	0xFF	gain_ch5
7:0	R/W	0xFF	gain_ch4

**Table 10-401 EE\_AUDIO\_SPDIFOUT\_CTRL0 0x123**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_work_enable, 0: disable ; 1: enable;
30	R/W		reg_req_frddr_rst; 1: reset REQ_FRDDR in SPDIFOUT;
29	R/W		reg_rst_afifo_out_n, reset afifo out side;
28	R/W		reg_rst_afifo_in_n, reset afifo in side; need set 1 after set reg_rst_afifo_out_n to 1;
27	R/W		reg_hold_start_en; 1: add delay to match TDM out when share buff;
26	R/W		reg_userdata_sel, 0: "user data" = reg_userdata_set; 1: "user data" = data [29];
25	R/W		reg_userdata_set
24	R/W		reg_chdata_sel, 0: "ch status" = reg_chsts0~B ;1: "ch status" = data [30];
23	R/W		reg_mix_lr, 0: disable; 1: L = (L+R)/2; R = (L+R)/2;
22	R/W		reg_mute_l, 0: disable; 1: ch_l_data = reg_mute_val;
21	R/W		reg_mute_r, 0: disable; 1: ch_r_data = reg_mute_val;
20	R/W		reg_data_sel, 0: insert data from 31bits;1: insert data from 27bits;
19	R/W		reg_out_msb_first, 0: lsb first; 1: msb first;

Bits	R/W	Default	Description
18	R/W		reg_valid_sel, 0: "valid flag" = reg_valid_set; 1: "valid flag" = data [28];
16	R/W		reg_parity_mask; initial parity value
15:0	R/W		reg_mask, [15:14]: mask lane7 L/R; [13:12]: mask lane6 L/R; [11:10]: mask lane5 L/R; [9:8]: mask lane4 L/R; [7:6]: mask lane3 L/R; [5:4]: mask lane2 L/R; [3:2]: mask lane1 L/R; [1:0]: mask lane0 L/R;

Table 10-402 EE\_AUDIO\_SPDIFOUT\_CTRL1 0x124

Bits	R/W	Default	Description
21	R/W	0x0000-0000	eq_drc_sel; 1: select eq_drc data;
26:24	R/W		reg_frddr_sel, 0:frddr_A; 1:frddr_B; 2:frddr_C; 3:frddr_D; 4:frddr_E;
23:16	R/W		reg_wait_cnt, wait some time when enable set to 1; then start request data from frddr;
15:14	R/W		reg_gain_shift; 0: data * 1; 1: data * 2; 2: data * 4; 3: data * 8;
12:8	R/W		reg_frddr_msb, msb position of data
7	R/W		Reg_force_start; 1: needn't ack from FRDDR; for only transmit "ch status" usage.
6:4	R/W		reg_frddr_type, 0: split 64bits ddr data to 8 sample, each sample need 8 bits; if bitwidth < 8, left-justified; 1: split 64bits ddr data to 4 sample, each sample need 16 bits; if bitwidth < 16, left-justified ; 2: split 64bits ddr data to 4 sample, each sample need 16 bits; if bitwidth < 16, right-justified ; 3: split 64bits ddr data to 2 sample, each sample need 32 bits; if bitwidth < 32, left-justified; 4: split 64bits ddr data to 2 sample, each sample need 32 bits; if bitwidth < 32, right-justified;
3	R/W		reg_keep_req_ddr_init; set 0 will stop sending data if req_ddr stopped by init_done signal pull down.(refer to reg_clr_by_init)
2	R/W		reg_sts_sel;

**Table 10-403 EE\_AUDIO\_SPDIFOUT\_PREAMB 0x125**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	set preamble Z, 0: preamble Z = 8'b11101000; 1: reg[7:0];
20	R/W		set preamble Y, 0: preamble Z = 8'b11100100; 1: reg[15:8];
29	R/W		set preamble X, 0: preamble Z = 8'b11100010; 1: reg[23:16];
23:16	R/W		preamble X
15:8	R/W		preamble Y
7:0	R/W		preamble Z

**Table 10-404 EE\_AUDIO\_SPDIFOUT\_SWAP 0x126**

Bits	R/W	Default	Description
31:16	R/W	0x0000-0000	hold start cnt, work when CTRL0[27] = 1;
15:8	R/W		reg_clr_by_init; when FRDDR is disable, its init_done will pull down, then need clear the status of REQ_FRDDR. 15: clear init_finish; 14: clear data_buff; 13: clear gain related; 12: clear DDR_REQ; 11: clear ack_masked(mean received first init_done); 10: clear afifo in; 9: clear afifo out ; 8: clear bit_cnt/slot_cnt/fit_num/wait_cnt;
7:4	R/W		lane0 right ch sel,
3:0	R/W		lane0 left ch sel,

**Table 10-405 EE\_AUDIO\_SPDIFOUT\_CHSTS0 0x127**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel A status[31:0]

**Table 10-406 EE\_AUDIO\_SPDIFOUT\_CHSTS1 0x128**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel A status[63:32]

**Table 10-407 EE\_AUDIO\_SPDIFOUT\_CHSTS2 0x129**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel A status[95:64]

**Table 10-408 EE\_AUDIO\_SPDIFOUT\_CHSTS3 0x12a**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel A status[127:96]

**Table 10-409 EE\_AUDIO\_SPDIFOUT\_CHSTS4 0x12b**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel A status[159:128]

**Table 10-410 EE\_AUDIO\_SPDIFOUT\_CHSTS5 0x12cv**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel A status[191:160]

**Table 10-411 EE\_AUDIO\_SPDIFOUT\_CHSTS6 0x12d**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel B status[31:0]

**Table 10-412 EE\_AUDIO\_SPDIFOUT\_CHSTS7 0x12e**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel B status[63:32]

**Table 10-413 EE\_AUDIO\_SPDIFOUT\_CHSTS8 0x12f**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel B status[95:64]

**Table 10-414 EE\_AUDIO\_SPDIFOUT\_CHSTS9 0x130**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel B status[127:96]

**Table 10-415 EE\_AUDIO\_SPDIFOUT\_CHSTSA 0x131**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel B status[159:128]

**Table 10-416 EE\_AUDIO\_SPDIFOUT\_CHSTSB 0x132**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel B status[191:160]

EE\_AUDIO\_SPDIFOUT\_MUTE\_VAL 0x133

**Table 10-417 EE\_AUDIO\_SPDIFOUT\_GAIN2 0x134**

Bits	R/W	Default	Description
31:24	R/W	0xFF	gain_ch11
23:16	R/W	0xFF	gain_ch10
15:8	R/W	0xFF	gain_ch9
7:0	R/W	0xFF	gain_ch8

**Table 10-418 EE\_AUDIO\_SPDIFOUT\_GAIN3 0x135**

Bits	R/W	Default	Description
31:24	R/W	0xFF	gain_ch15
23:16	R/W	0xFF	gain_ch14
15:8	R/W	0xFF	gain_ch13
7:0	R/W	0xFF	gain_ch12

**Table 10-419 EE\_AUDIO\_SPDIFOUT\_GAIN\_EN 0x136**

Bits	R/W	Default	Description
31:16	R/W	0x0000-0000	
15	R/W	0x0000-0000	ch15 gain_en; 1: out = in * gain_ch15; 0: out = in;
14	R/W	0x0000-0000	ch14 gain_en; 1: out = in * gain_ch14; 0: out = in;
13	R/W	0x0000-0000	ch13 gain_en; 1: out = in * gain_ch13; 0: out = in;
12	R/W	0x0000-0000	ch12 gain_en; 1: out = in * gain_ch12; 0: out = in;
11	R/W	0x0000-0000	ch11 gain_en; 1: out = in * gain_ch11; 0: out = in;
10	R/W	0x0000-0000	ch10 gain_en; 1: out = in * gain_ch10; 0: out = in;
9	R/W	0x0000-0000	ch9 gain_en; 1: out = in * gain_ch9; 0: out = in;
8	R/W	0x0000-0000	ch8 gain_en; 1: out = in * gain_ch8; 0: out = in;
7	R/W	0x0000-0000	ch7 gain_en; 1: out = in * gain_ch7; 0: out = in;

Bits	R/W	Default	Description
6	R/W	0x0000-0000	ch6 gain_en; 1: out = in * gain_ch6; 0: out = in;
5	R/W	0x0000-0000	ch5 gain_en; 1: out = in * gain_ch5; 0: out = in;
4	R/W	0x0000-0000	ch4 gain_en; 1: out = in * gain_ch4; 0: out = in;
3	R/W	0x0000-0000	ch3 gain_en; 1: out = in * gain_ch3; 0: out = in;
2	R/W	0x0000-0000	ch2 gain_en; 1: out = in * gain_ch2; 0: out = in;
1	R/W	0x0000-0000	ch1 gain_en; 1: out = in * gain_ch1; 0: out = in;
0	R/W	0x0000-0000	ch0 gain_en; 1: out = in * gain_ch0; 0: out = in;

Table 10-420 EE\_AUDIO\_SPDIFOUT\_GAIN\_CTRL 0x137

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_gain_step_chg; 0: the current_gain will follow direct to reg_gain change; 1: the current_gain will follow step by step to reg_gain change; etc: change reg_gain from 40 to 50, and gain_step = 3; the current_gain will change as: 40-> 43-> 46-> 49-> 50;
25:24	R/W		reg_cur_gain_sel; 0: cur_gain_sts = ch3,ch2,ch1,ch0; 1: cur_gain_sts = ch7,ch6,ch5,ch4; 2: cur_gain_sts = ch11,ch10,ch9,ch8; 3: cur_gain_sts = ch15,ch14,ch13,ch12;
23:16	R/W		reg_gain_step; modified gain value each time;
15:0	R/W		reg_gain_updt_timer; the period of each change, unit is FS;

EE\_AUDIO\_SPDIFOUT\_B\_STAT      0x1a0  
 EE\_AUDIO\_SPDIFOUT\_B\_GAIN0    0x1a1  
 EE\_AUDIO\_SPDIFOUT\_B\_GAIN1    0x1a2  
 EE\_AUDIO\_SPDIFOUT\_B\_CTRL0    0x1a3  
 EE\_AUDIO\_SPDIFOUT\_B\_CTRL1    0x1a4  
 EE\_AUDIO\_SPDIFOUT\_B\_PREAMB   0x1a5  
 EE\_AUDIO\_SPDIFOUT\_B\_SWAP     0x1a6  
 EE\_AUDIO\_SPDIFOUT\_B\_CHSTS0   0x1a7  
 EE\_AUDIO\_SPDIFOUT\_B\_CHSTS1   0x1a8  
 EE\_AUDIO\_SPDIFOUT\_B\_CHSTS2   0x1a9  
 EE\_AUDIO\_SPDIFOUT\_B\_CHSTS3   0x1aa  
 EE\_AUDIO\_SPDIFOUT\_B\_CHSTS4   0x1ab



EE_AUDIO_SPDIFOUT_B_CHSTS5	0x1ac
EE_AUDIO_SPDIFOUT_B_CHSTS6	0x1ad
EE_AUDIO_SPDIFOUT_B_CHSTS7	0x1ae
EE_AUDIO_SPDIFOUT_B_CHSTS8	0x1af
EE_AUDIO_SPDIFOUT_B_CHSTS9	0x1b0
EE_AUDIO_SPDIFOUT_B_CHSTSA	0x1b1
EE_AUDIO_SPDIFOUT_B_CHSTSB	0x1b2
EE_AUDIO_SPDIFOUT_B_MUTE_VAL	0x1b3
EE_AUDIO_SPDIFOUT_GAIN2	0x1b4
EE_AUDIO_SPDIFOUT_GAIN3	0x1b5
EE_AUDIO_SPDIFOUT_GAIN_EN	0x1b6
EE_AUDIO_SPDIFOUT_GAIN_CTRL	0x1b7

**Table 10-421 EE\_AUDIO\_SPDIFIN\_LB\_CTRL0 0x1f0**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_work_enable, 0: disable; 1:enable;
30	R/W		reg_chnum_sel, 0: ch_num = 0~383(include frame cnt); 1: ch_num = 0~1(only L/R);
29	R/W		reg_rst_afifo_out_n, reset afifo out side;
28	R/W		reg_rst_afifo_in_n, reset afifo in side; need set 1 after set reg_rst_afifo_out_n to 1;
26	R/W		reg_chunm_en; 1: start add ch_cnt to ch_num;
25	R/W		reg_findpapb_en, 0: disable NonPCM mode; 1: enable;
24	R/W		reg_clk_inv; 0: sample spdif by posedge; 1: sample spdif by negedge;
23:12	R/W		reg_nonpcm2pcm_th, when detected NonPcm mode;if long time (z_cnt >= th) didn't detect PaPb again, will generate interrupt to SW: now changed to PCM mode;
11:8	R/W		reg_ch_status_sel, For EE_AUDIO_SPDIFIN_STAT1
7	R/W		reg_mute_l, 0: disable ; 1: mute channel L;
6	R/W		reg_mute_r, 0: disable ; 1: mute channel R;
5:4	R/W		reg_spdifin_src_sel, 0: PAD of spdifin;1: spdifout;
3	R/W		reg_check_valid, 0: disable valid check ; 1: enable;
2	R/W		reg_check_parity, 0: disable parity check ; 1: enable;
1	R/W		reg_invert_data, 0: disable; 1: invert [27:4] to [4:27];
0	R/W		reg_spdifin_phase, 0: disable invert; 1: enable;

**Table 10-422 EE\_AUDIO\_SPDIFIN\_LB\_CTRL1 0x1f1**

Bits	R/W	Default	Description
27:20	R/W		reg_interrupt_mask, mask each interrupt;

Table 10-423 EE\_AUDIO\_SPDIFIN\_LB\_CTRL6 0x1f6

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	Reg_clr_internal_sts[7:0]; 7: clear valid bit status; 6: clear parity bit status; 5: clear ch status; 4: clear z_cnt status; 3: clear find_nonpcm status; 2: clear pd_data status; 1: clear pc_data status; 0: clear find_papb status;
23:16	R/W		Reg_clr_interrupt[7:0] for each bit of irq_status[7:0];
13	R/W		reg_irq_mode; 1: level mode; 0: pulse mode;
8	R/W		reg_papb_ext_sync, 1: add ext "0" sync check for papb; 0: disable;
7:0	R/W		reg_papb_ext_mask, mask 8 channel "0" sync

Table 10-424 EE\_AUDIO\_SPDIFIN\_LB\_STAT0 0x1f7

Bits	R/W	Default	Description
31	R/W	0x0000-0000	Valid bit
11:8	R/W		affo_cnt[3:0]
7:0	R/W		r_interrupt_status, interrupt status, need clear by reg_clk_interrupt; [7]: find PaPb; [6]: valid changed; [5]: find nonpcm to pcm ( reg_nonpcm2pcm_th ); [4]: find Pc or Pd changed; [3]: find CH status changed; [2]: find sample mode changed; [1]: find parity error; [0]: find overflow;

Table 10-425 EE\_AUDIO\_SPDIFIN\_LB\_STAT1 0x1f8

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	channel status, by reg_ch_status_sel; reg_ch_status_sel[3]: 0: channel A; 1: channel B; reg_ch_status_sel[2:0]: 0: ch_status[31:0]; 1: ch_status[63:32]; 2: ch_status[95:64]; 3: ch_status[127:96]; 4: ch_status[159:128]; 5: ch_status[191:160]; 6: pc[15:0],pd[15:0];

Table 10-426 EE\_AUDIO\_SPDIFIN\_LB\_MUTE\_VAL 0x1fa

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	reg_spdifin_mute_val, when muted, the channel value

### 10.11.9 ResampleA Registers

Base Address: 0xFE33\_1C00

Each register final address = module base address+ address \* 4

**Table 10-427 AUDIO\_RSAMP\_CTRL0 0x000**

Bit(s)	R/W	Default	Description
2	R/W	0	reg_lock_rst, phase_lock module software reset
1	R/W	0	reg_rsamp_rst, resample_core module software reset
0	R/W	0	reg_sw_rst, resample_top module software reset

**Table 10-428 AUDIO\_RSAMP\_CTRL1 0x001**

Bit(s)	R/W	Default	Description
25	R/W	0	reg_rsamp_rst_sel 1:resample_core use hardware as reset 0:resample_core use software reg_rsamp_rst as reset
24	R/W	0	reg_module_bypas 1: bypass resamp_top, input source direct output 0: resamp_top work
23:18	R/W	0	reg_gclk_ctrl, clk gate
17:13	R/W	23	reg_in_msb
12	R/W	0	reg_output_en, is used for source switch
11	R/W	0	reg_rsamp_en : It is used to enable the resample module. 1=enable, 0 = disable
10	R/W	0	reg_filt_en : It is used to enable the AA filter to downsample the input data. 1=enable, 0 = disable
9	R/W	0	reg_post_en : It is used to enable the compensation filter. 1= enable , 0= disable
8	R/W	0	reg_inp_mux_mode
3:0	R/W	0	reg_inp_mux

**Table 10-429 AUDIO\_RSAMP\_CTRL2 0x002**

Bit(s)	R/W	Default	Description
29:24	R/W	2	reg_chx_size : It is used to set the channel number.
17:16	R/W	0	reg_scl_step 0: 1/1 1: 1/2 2: 1/4
15:8	R/W	63	reg_filt_tap : It is used to set the taps of the AA filter.
7:0	R/W	63	reg_intp_tap : It is used to set the tap of the interpolation filter.

**Table 10-430 AUDIO\_RSAMP\_PHSINIT 0x003**

Bit(s)	R/W	Default	Description
27:0	R/W	0	reg_init_phs : It is used to set the initial phase for accumulator.

**Table 10-431 AUDIO\_RSAMP\_PHSSTEP 0x004**

Bit(s)	R/W	Default	Description
30:0	R/W	0x8000-000	reg_rsamp_step: It is used to set the step size for accumulator. It can be calculated by: $Fs\_in/down\_ratio/fs\_out*2^{28}$ , where fs_in is the frequency of input data, fs_out is the frequency of output data, down_ratio is the down sample ratio.

**Table 10-432 AUDIO\_RSAMP\_SHIFT 0x005**

Bit(s)	R/W	Default	Description
31:24	R/W	23	reg_rsft_iir : It is used to set the shift bit for compensation IIR filter.
23:16	R/W	21	reg_rsft_blnd : It is used to set the shift bit for blending.
15:8	R/W	31	reg_rsft_sinc : It is used to set the shift bit for interpolation filter.
7:0	R/W	31	reg_rsft_aa □ It is used to set the shift bit for anti-alias filter.

**Table 10-433 AUDIO\_RSAMP\_ADJ\_CTRL0 0x006**

Bit(s)	R/W	Default	Description
2	R/W	0	reg_rsamp_adj_out_inv : It is used to set the direction for adjusting.
1	R/W	0	reg_rsamp_adj_force_en : It is used to enable forcing a adjustment value for accumulator step. 1= enable,0=disable
0	R/W	0	reg_rsamp_adj_en : It is used to enable the auto adjusting module. 1=enable, 0=disable

**Table 10-434 AUDIO\_RSAMP\_ADJ\_CTRL1 0x007**

Bit(s)	R/W	Default	Description
31:16	R/W	8	reg_rsamp_adj_odet_step
15:0	R/W	32768	reg_rsamp_adj_kmax : It is used to set the maximum value after loop filter.

**Table 10-435 AUDIO\_RSAMP\_ADJ\_SFT 0x008**

Bit(s)	R/W	Default	Description
29	R/W	0	reg_rsamp_adj_dif_sel : It is used to select the input data for adjustment module.
28:24	R/W	9	reg_rsamp_adj_ki : It is used to set the ki of loop filter.
20:16	R/W	1	reg_rsamp_adj_kp : It is used to set the kp of loop filter
12:8	R/W	6	reg_rsamp_adj_ki_sft : It is used to set the shift value after ki multiplier.
5:0	R/W	12	reg_rsamp_adj_out_sft : It is used to set the adjustment output shift value.

**Table 10-436 AUDIO\_RSAMP\_ADJ\_IDET\_LEN 0x009**

Bit(s)	R/W	Default	Description
31:0	R/W	10000	reg_rsamp_adj_idet_len : It is used to set the detection length for adjustment.

**Table 10-437 AUDIO\_RSAMP\_ADJ\_FORCE 0x00a**

Bit(s)	R/W	Default	Description
31:0	R/W	8	reg_rsamp_adj_force_err: It is used to set the forcing err.

**Table 10-438 AUDIO\_RSAMP\_ADJ\_KI\_FORCE 0x00b**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_rsamp_adj_ki_force : It is used to set a forcing value for ki adjustment.

**Table 10-439 AUDIO\_RSAMP\_RO\_STATUS 0x010**

Bit(s)	R/W	Default	Description
31:0	R.O	0x0	ro_rsamp_stat : din_chx_chk_err, is_idle_st, rsamp_fifo_over_cnt[7:0]} : It is used report the working state for resample.

**Table 10-440 AUDIO\_RSAMP\_RO\_ADJ\_FREQ 0x011**

Bit(s)	R/W	Default	Description
31:0	R.O	0x0	ro_rsamp_adj_freq : It is used to report the the adjustment frequency.

**Table 10-441 AUDIO\_RSAMP\_RO\_ADJ\_DIFF\_BAK 0x012**

Bit(s)	R/W	Default	Description
31:0	R.O	0x0	ro_det_diff_bak : It is used to report the difference for comparator.

**Table 10-442 AUDIO\_RSAMP\_RO\_ADJ\_DIFF\_DLT 0x013**

Bit(s)	R/W	Default	Description
31:0	R.O	0x0	ro_det_diff_dlt : It is used to report the difference delta.

**Table 10-443 AUDIO\_RSAMP\_RO\_ADJ\_PHS\_ERR 0x014**

Bit(s)	R/W	Default	Description
31:0	R.O	0x0	ro_det_phase_err : It is used to report the detection phase error.

**Table 10-444 AUDIO\_RSAMP\_RO\_ADJ\_KI\_OUT 0x015**

Bit(s)	R/W	Default	Description
31:0	R.O	0x0	ro_rsamp_ki_out : It is used to report the out of integration branch.

**Table 10-445 AUDIO\_RSAMP\_RO\_IN\_CNT 0x016**

Bit(s)	R/W	Default	Description
31:0	R.O	0x0	ro_rsamp_in_cnt : It is used to report the counter of input data .

**Table 10-446 AUDIO\_RSAMP\_RO\_OUT\_CNT 0x017**

Bit(s)	R/W	Default	Description
31:0	R.O	0x0	ro_rsamp_out_cnt : It is used to report the counter of output data.

**Table 10-447 AUDIO\_RSAMP\_POST\_COEF0 0x020**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_post_coef0 : It is used to set the coefficient of compensation filter.

**Table 10-448 AUDIO\_RSAMP\_POST\_COEF1 0x021**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_post_coef1 : It is used to set the coefficient of compensation filter.

**Table 10-449 AUDIO\_RSAMP\_POST\_COEF2 0x022**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_post_coef2 : It is used to set the coefficient of compensation filter.

**Table 10-450 AUDIO\_RSAMP\_POST\_COEF3 0x023**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_post_coef3 : It is used to set the coefficient of compensation filter.

**Table 10-451 AUDIO\_RSAMP\_POST\_COEF4 0x024**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_post_coef4 : It is used to set the coefficient of compensation filter.

**Table 10-452 AUDIO\_RSAMP\_AA\_COEF\_ADDR 0x030**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_aa_coef_addr

**Table 10-453 AUDIO\_RSAMP\_AA\_COEF\_DATA 0x031**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_aa_coef_data

**Table 10-454 AUDIO\_RSAMP\_SINC\_COEF\_ADDR 0x040**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_sinc_coef_addr

**Table 10-455 AUDIO\_RSAMP\_SINC\_COEF\_DATA 0x041**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reg_sinc_coef_data

### 10.11.10 TOACODEC Registers

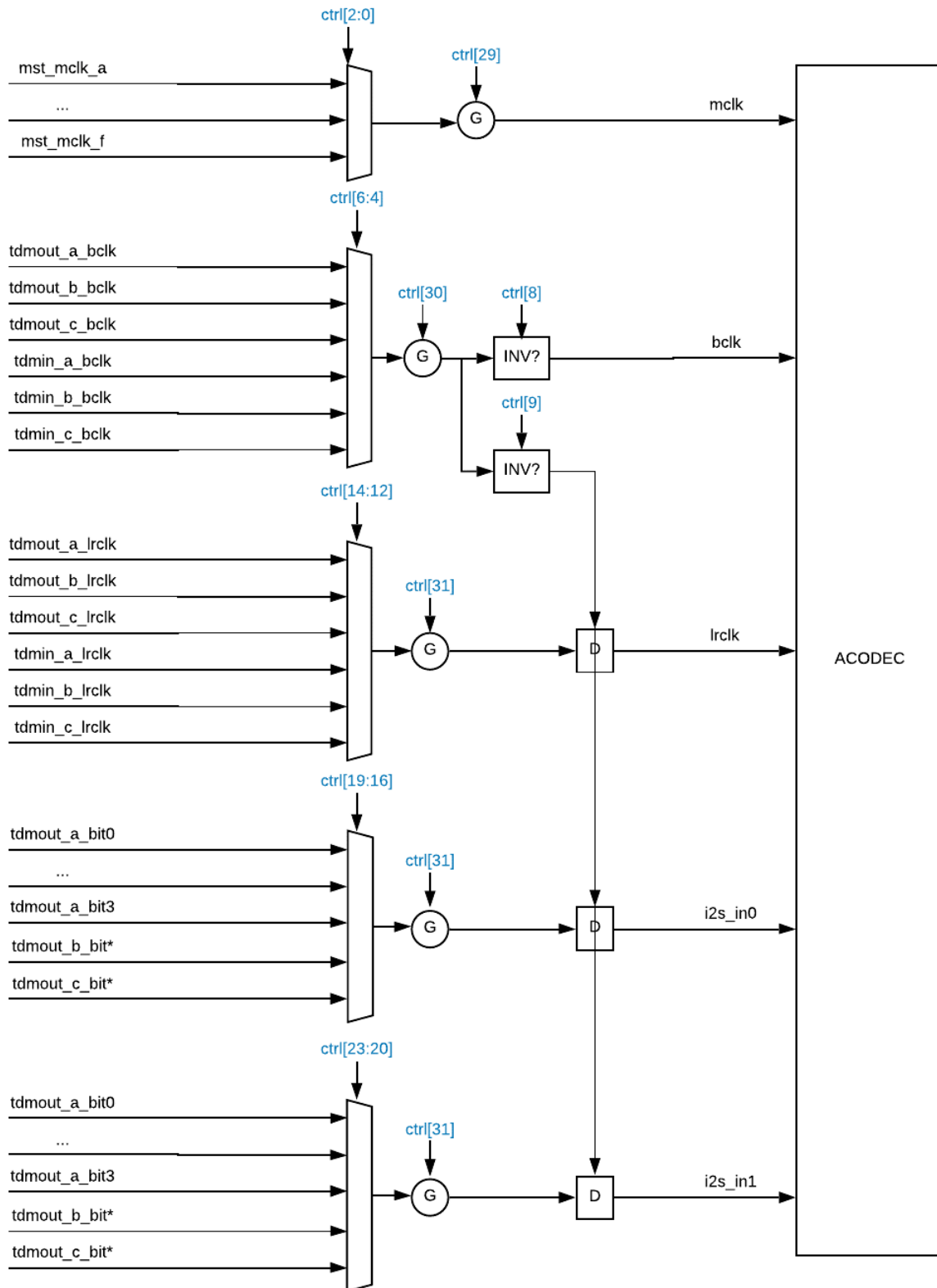




Table 10-456 EE\_AUDIO\_TOACODEC\_CTRL0 0x1d0

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_toacodec_en_dat, 0: disable; 1: enable; step1: enable/disable mclk step2: enable/disable bclk step3: enable/disable dat
30	R/W		reg_toacodec_en_bclk, 0: disable; 1: enable;
29	R/W		reg_toacodec_en_mclk, 0: disable; 1: enable;
26:22	R/W		Dat1_sel: 0: tdmout_a_dat[0]; 1: tdmout_a_dat[1]; 2: tdmout_a_dat[2]; 3: tdmout_a_dat[3]; 4: tdmout_a_dat[4]; 5: tdmout_a_dat[5]; 6: tdmout_a_dat[6]; 7: tdmout_a_dat[7]; 8: tdmout_b_dat[0]; 9: tdmout_b_dat[1]; 10: tdmout_b_dat[2]; 11: tdmout_b_dat[3]; 12: tdmout_b_dat[4]; 13: tdmout_b_dat[5]; 14: tdmout_b_dat[6]; 15: tdmout_b_dat[7]; 16: tdmout_c_dat[0]; 17: tdmout_c_dat[1]; 18: tdmout_c_dat[2]; 19: tdmout_c_dat[3]; 20: tdmout_c_dat[4]; 21: tdmout_c_dat[5]; 22: tdmout_c_dat[6]; 23: tdmout_c_dat[7];
20:16	R/W		Dat0_sel: 0: tdmout_a_dat[0]; 1: tdmout_a_dat[1]; 2: tdmout_a_dat[2]; 3: tdmout_a_dat[3]; 4: tdmout_a_dat[4]; 5: tdmout_a_dat[5]; 6: tdmout_a_dat[6]; 7: tdmout_a_dat[7]; 8: tdmout_b_dat[0]; 9: tdmout_b_dat[1]; 10: tdmout_b_dat[2]; 11: tdmout_b_dat[3]; 12: tdmout_b_dat[4]; 13: tdmout_b_dat[5]; 14: tdmout_b_dat[6]; 15: tdmout_b_dat[7]; 16: tdmout_c_dat[0]; 17: tdmout_c_dat[1]; 18: tdmout_c_dat[2]; 19: tdmout_c_dat[3]; 20: tdmout_c_dat[4]; 21: tdmout_c_dat[5]; 22: tdmout_c_dat[6]; 23: tdmout_c_dat[7];
14:12	R/W		lrcclk_sel: 0: tdmout_a_lrcclk; 1: tdmout_b_lrcclk; 2: tdmout_c_lrcclk; 4: tdmout_a_lrcclk; 5: tdmout_b_lrcclk;

Bits	R/W	Default	Description
			6: tdm_in_c_lrclk;
10	R/W		Lrclk_inv; 1: invert lrclk;
9	R/W		Bclk_cap_inv: The dat_o and lrclk_o will captured for timing balance after select; If this bit set to 1, will use invert bclk to capture;
8	R/W		Bclk_o_inv: if set 1, the final bclk connect to acodec will invert;
6:4	R/W		Bclk_sel: 0: tdmout_a_bclk; 1: tdmout_b_bclk; 2: tdmout_c_bclk; 4: tdm_in_a_bclk; 5: tdm_in_b_bclk; 6: tdm_in_c_bclk;
2:0	R/W		Mclk_sel: 0:mst_mclk_a; 1:mst_mclk_b; 2:mst_mclk_c; 3:mst_mclk_d; 4:mst_mclk_e; 5:mst_mclk_f;

### 10.11.11 TOHDMITX Registers

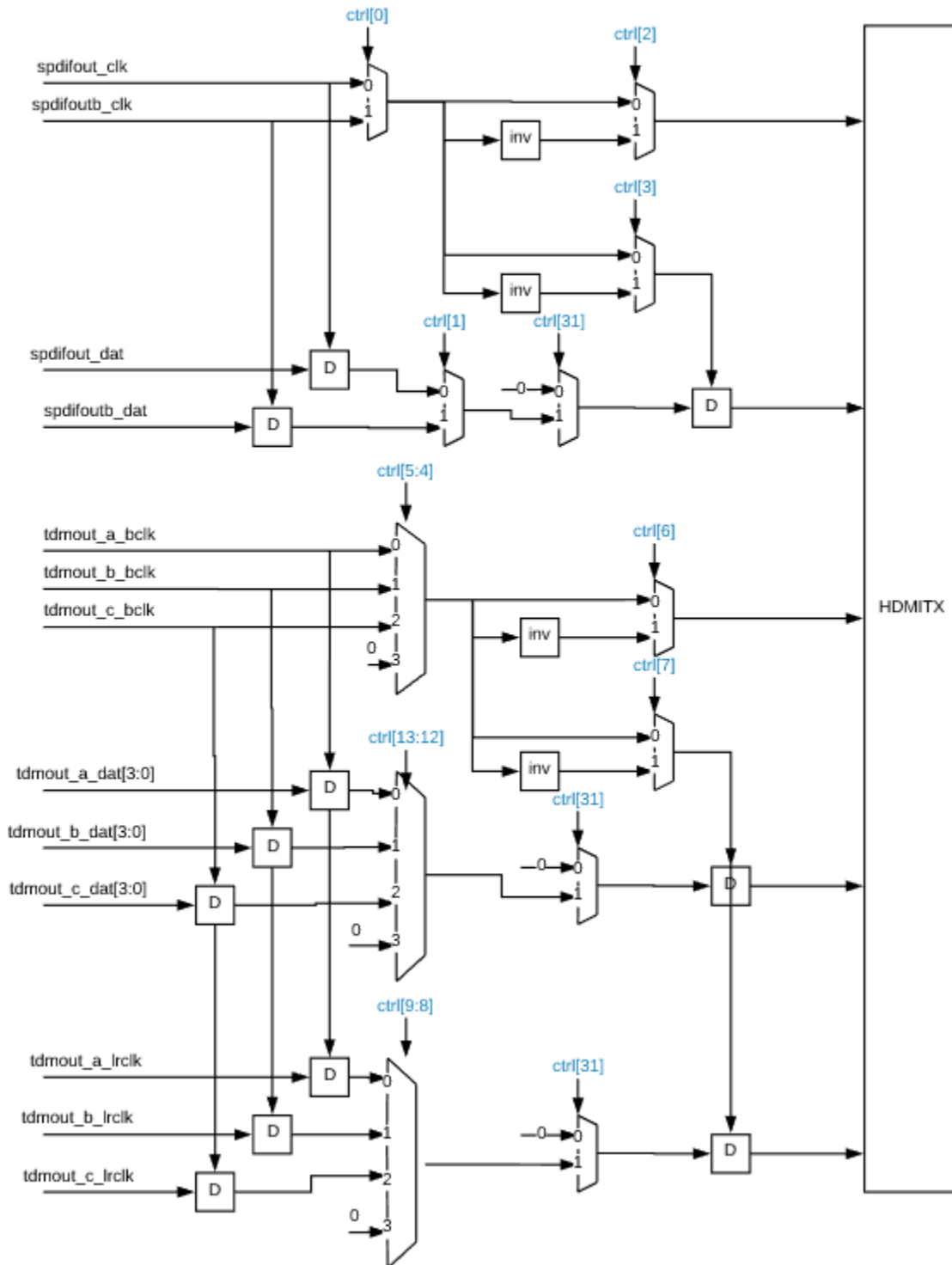


Table 10-457 EE\_AUDIO\_TOHDMITX\_CTRL0 0x1d1

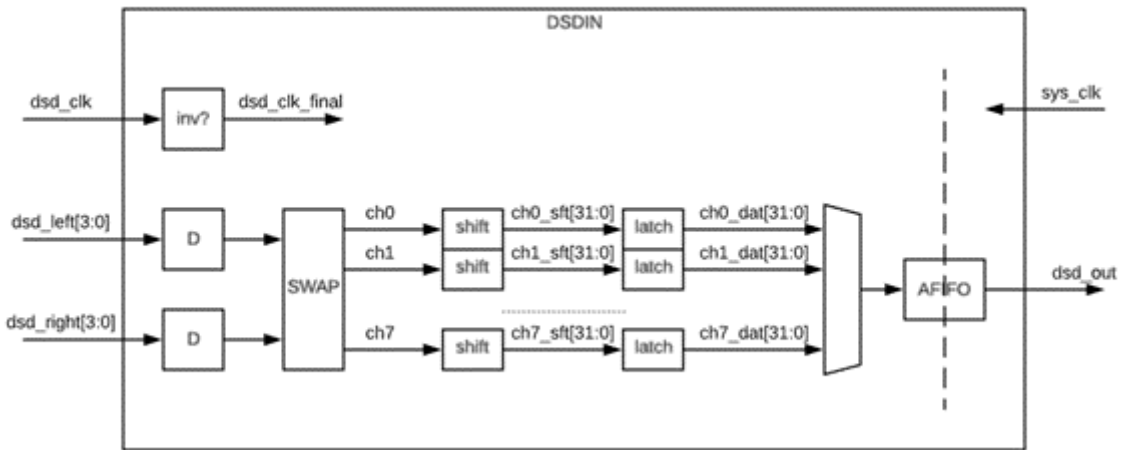
Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_tohdmix_spdif_en_dat, 0: disable; 1: enable; step1: enable/disable clk; step2: enable/disable dat;
30	R/W		reg_tohdmix_spdif_en_clk, 0: disable; 1: enable;
29	R/W		reg_tohdmix_i2s_en_dat, 0: disable; 1: enable; step1: enable/disable clk; step2: enable/disable dat;
28	R/W		reg_tohdmix_i2s_en_clk, 0: disable; 1: enable;
27:20	R/W		reg_tohdmix_mclk_div, if need div8, set to 7
19	R/W		reg_tohdmix_mclk_en;
18:16	R/W		reg_tohdmix_mclk_sel: 0: mst_mclk_a; 1: mst_mclk_b; 2: mst_mclk_c; 3: mst_mclk_d; 4: mst_mclk_e; 5: mst_mclk_f;
13:12	R/W		dat_sel: 0: tdmout_a_dat; 1: tdmout_b_dat; 2: tdmout_c_dat;
10	R/W		Lrclk_inv;
9:8	R/W		lrclk_sel: 0: tdmout_a_lrclk; 1: tdmout_b_lrclk; 2: tdmout_c_lrclk;
7	R/W		Bclk_cap_inv: The dat_o and lrclk_o will captured for timing balance after select; If this bit set to 1, will use invert bclk to capture;
6	R/W		Bclk_o_inv: if set 1, the final bclk connect to acodec will invert;
5:4	R/W		Bclk_sel: 0: tdmout_a_bclk; 1: tdmout_b_bclk; 2: tdmout_c_bclk;
3	R/W		Spdif_clk_cap_inv: The spdif_dat will captured for timing balance after select; If this bit set to 1, will use invert spdif_clk to capture;
2	R/W		Spdif_clk_o_inv: if set 1, the final bclk connect to hdmix will invert;
1	R/W		Spdif_sel: 0: spdif_out; 1: spdif_out_b;
0	R/W		Spdif_clk_sel: 0: spdif_clk; 1: spdif_clk_b;

### 10.11.12 FRHDMIRX

HDMIRX can send 3 audio signals:

- I2S, connect to TDMIN;
- SPDIF, connect to spdifin;
- DSD, include 8 lanes;

Figure 10-26 FRHDMIRX



### 10.11.13 TOVAD Registers

Table 10-458 EE\_AUDIO\_TOVAD\_CTRL0 0x1d2

Bits	R/W	Default	Description
31	R/W	0x0000-0000	reg_tovad_en, 0: disable; 1: enable;
30	R/W		reg_tovad_v_sel, 0: level; 1:pulse;
14:12	R/W		Data_sel: 0: tadmin_a; 1: tadmin_b; 2: tadmin_c; 3: spdifin; 4: pdmin; 5: loopback_b; 6: tadmin_lb; 7: loopback_a;

### 10.11.14 PDM Registers

Base Address: 0xFE331000

Each register final address = module base address+ address \* 4

**Table 10-459 PDM\_CTRL 0x00**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	PDM enable
30	R/W		invert the PDM_DCLK
29	R/W		output mode: 1: 24bits. 0: 32 bits
28	R/W		bypass mode. 1: bypass all filter. directly output the PDM input to DDR. 0: normal mode.
27:20	R/W		pdm_mute_mask[7:0]; [7] mute pdm ch7; ... [0] mute pdm ch0;
19	R/W		train_en; 1: check value of capture data;
18	R/W		train_clr; 1: clear training status;
17	R/W		chnum_sel. 0: valid_cnt; 1: mask_cnt;
16	R/W		PDM Asynchronous FIFO soft reset. write 1 to soft reset AFIFO
15:8	R/W		PDM channel reset. 0: to reset each PDM channel. 1: normal mode
7:0	R/W		PDM channel enable. each bit for one channel

**Table 10-460 PDM\_HCIC\_CTRL1 0x01**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	hcic filter enable. 1 use sinc filter. 0 bypass input to output
29:24	R/W		hcic final gain shift parameter
23:16	R/W		hcic final gain multiplier
8:4	R/W		hcic down sample rate
3:0	R/W		hcic stage number. must be between 3 to 9

PDM\_HCIC\_CTRL2 0x02

**Table 10-461 PDM\_F1\_CTRL 0x03**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	filter 1 enable
17:16	R/W		f1 round mode. 2'b00 : sign bit at bit 49. 28bits output [49:22] round at bit 21. 32bits output [49:18]. 24bits output [49:26] 2'b01 : sign bit at bit 50. 28bits output [50:23] round at bit 22. 32bits output [49:18]. 24bits output [49:26] 2'b10 : sign bit at bit 51. 28bits output [51:24] round at bit 23 32bits output [49:18]. 24bits output [49:26].
15:12	R/W		filter 1 down sample rate
8:0	R/W		filter 1 stage number

**Table 10-462 PDM\_F2\_CTRL 0x04**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	filter 2 enable
17:16	R/W		F2 round mode. 2'b00 : round at bit 21. 2'b01 : round at bit 22. 2'b10 : round at bit 23
15:12	R/W		filter 2 down sample rate
8:0	R/W		filter 2 stage number

**Table 10-463 PDM\_F3\_CTRL 0x05**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	filter 3 enable
17:16	R/W		F3 round mode. 2'b00 : round at bit 21. 2'b01 : round at bit 22. 2'b10 : round at bit 23
15:12	R/W		filter 3 down sample rate
8:0	R/W		filter 3 stage number

**Table 10-464 PDM\_HPF\_CTRL 0x06**

Bits	R/W	Default	Description
31	R/W	0x0000-0000	High pass filter enable
20:16	R/W		high pass filter shift steps. 6~19 steps
15:0	R/W		high pass filter output factor

**Table 10-465 PDM\_CHAN\_CTRL 0x07**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	Chan3 data sample pointer vs rise edge of the PDM_DCLK
23:16	R/W		Chan2 data sample pointer vs rise edge of the PDM_DCLK
15:8	R/W		Chan1 data sample pointer vs rise edge of the PDM_DCLK
7:0	R/W		Chan0 data sample pointer vs rise edge of the PDM_DCLK

**Table 10-466 PDM\_CHAN\_CTRL1 0x08**

Bits	R/W	Default	Description
31:24	R/W	0x0000-0000	Chan7 data sample pointer vs rise edge of the PDM_DCLK
23:16	R/W		Chan6 data sample pointer vs rise edge of the PDM_DCLK
15:8	R/W		Chan5 data sample pointer vs rise edge of the PDM_DCLK
7:0	R/W		Chan4 data sample pointer vs rise edge of the PDM_DCLK

**Table 10-467 PDM\_COEFF\_ADDR 0x09**

Bits	R/W	Default	Description
8:0	R/W	0x0000-0000	address of the write/read of coeff data

**Table 10-468 PDM\_COEFF\_DATA 0x0A**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	write/read data to coeff memory

**Table 10-469 PDM\_CLKG\_CTRL 0x0B**

Bits	R/W	Default	Description
6	R/W	0x0000-0000	filt_ctrl module auto clock gating control
5	R/W		sinc fifo module auto clock gating control
4	R/W		filter module auto clock gating control
3	R/W		apb module auto clock gating control
2	R/W		coeff memory module auto clock gating control
1	R/W		each channel module auto clock gating control
0	R/W		cts_pdm_clk auto clock gating control

**Table 10-470 PDM\_STS 0x0C**

Bits	R/W	Default	Description
15:12	R	0x0000-0000	afffo count value
11:4	R		train result of 8 ch; 1: diff; 0: same;
1	R		HPF filter output overflow. means the PCLK is too slow
0	R		HCIC filter output overflow. means the CTS_PDM_CLK is too slow. can't finished the filter function.

**Table 10-471 PDM\_MUTE\_VALUE 0x0D**

Bits	R/W	Default	Description
31:0	R/W	0x0000-0000	mute value if mute_mask = 1;



**Table 10-472 PDM\_MASK\_NUM 0x0E**

Bits	R/W	Default	Description
15:0	R/W	0x0000-0000	When PDM power on, there are some invalid data because filter initialize. So set this register can mask these invalid data. If PDM out frequency is 48K, set MASK_NUM = 49 will mask $50/48k = 1.04ms$ .

**Table 10-473 PDM\_CHAN\_CTRL2 0x0F**

Bits	R/W	Default	Description
7:0	R/W	0x0000-0000	start filter pointer vs rise edge of the PDM_DCLK need avoid close to data sample pointer which defined by PDM_CHAN_CTRL0/1

### 10.11.15 LOCKER Registers

Base Address: 0xFE331400

Each register final address = module base address+ address \* 4

**Table 10-474 AUD\_LOCK\_EN 0X00**

Bit(s)	R/W	Default	Description
31:1			reserved
0	R/W	0x0	Audio_lock_en: 0 = disable; 1 = enable.

**Table 10-475 AUD\_LOCK\_SW\_RESET 0X01**

Bit(s)	R/W	Default	Description
31:1			reserved
0	R/W	0x0	Audio_lock_soft_reset: 1=generate soft reset pulse

**Table 10-476 AUD\_LOCK\_SW\_LATCH 0X02**

Bit(s)	R/W	Default	Description
31:4			reserved
3	R/W	0x0	omclk2ref software latch: 1= generate the latch pulse
2	R/W	0x0	Ref2omclk software latch: 1= generate the latch pulse
1	R/W	0x0	Ref2imclk software latch: 1= generate the latch pulse
0	R/W	0x0	lmclk2ref software latch: 1= generate the latch pulse

**Table 10-477 AUD\_LOCK\_HW\_LATCH 0X03**

Bit(s)	R/W	Default	Description
31:4			reserved
3	R/W	0x0	omclk2ref hardware latch enable: 1= enable 0= disable

Bit(s)	R/W	Default	Description
2	R/W	0x0	Ref2omclk hardware latch enable: 1= enable 0= disable
1	R/W	0x0	Ref2imclk hardware latch enable: 1= enable 0= disable
0	R/W	0x0	Imclk2ref hardware latch enable: 1= enable 0= disable

**Table 10-478 AUD\_LOCK\_REFCLK\_SRC 0X04**

Bit(s)	R/W	Default	Description
31:2			reserved
1:0	R/W	0x0	Ref clk source sel: 0= pclk; 1=oscinclk; 2,3,reserved

**Table 10-479 AUD\_LOCK\_REFCLK\_LAT\_INT 0X05**

Bit(s)	R/W	Default	Description
31:0			U32 number of reference clock cycles to latch the imclk and omclk

**Table 10-480 AUD\_LOCK\_IMCLK\_LAT\_INT 0X06**

Bit(s)	R/W	Default	Description
31:0			U32 number of imclk clock cycles to latch the reference clock

**Table 10-481 AUD\_LOCK\_OMCLK\_LAT\_INT 0X07**

Bit(s)	R/W	Default	Description
31:0			U32 number of omclk clock cycles to latch the reference clock

**Table 10-482 AUD\_LOCK\_REFCLK\_DS\_INT 0X08**

Bit(s)	R/W	Default	Description
31:10			reserved
9:0	R/W	0x0	U10 downsample step of reference clock for the counter to be measured, module = x+1

**Table 10-483 AUD\_LOCK\_IMCLK\_DS\_INT 0X09**

Bit(s)	R/W	Default	Description
31:10			reserved
9:0	R/W	0x0	U10 downsample step of imclk for the counter to be measured, module = x+1

**Table 10-484 AUD\_LOCK\_OMCLK\_DS\_INT 0X0a**

Bit(s)	R/W	Default	Description
31:10			reserved
9:0	R/W	0x0	U10 downsample step of omclk for the counter48ds to be measured, module = x +1

**Table 10-485 AUD\_LOCK\_INT\_CLR 0X0b**

Bit(s)	R/W	Default	Description
31:2			reserved
1	R/W	0x0	It is used to generate pulse to clear the interrupt status
0	R/W	0x0	It is used to generate pulse to clear the interrupt

**Table 10-486 AUD\_LOCK\_GCLK\_CTRL 0X0c**

Bit(s)	R/W	Default	Description
31:4			reserved
3:2	R/W	0x0	It is used to gate the module clock
1	R/W	0x0	It is used to gate the register clock
0			reserved

**Table 10-487 AUD\_LOCK\_INT\_CTRL 0X0d**

Bit(s)	R/W	Default	Description
31:4			reserved
3	R/W	0x0	It is used to mask interrupt for omclk_state
2	R/W	0x0	It is used to mask interrupt for imclk_state
1	R/W	0x0	It is used to mask interrupt for refclk_state1
0	R/W	0x0	It is used to mask interrupt for refclk_state0

**Table 10-488 RO\_REF2IMCLK\_CNT\_L 0X10**

Bit(s)	R/W	Default	Description
31:0	R	0x0	U48 latched imclk counter48ds of each reg_refclk_latch_interval; Will start from 0 reaching 2^48

**Table 10-489 RO\_REF2IMCLK\_CNT\_H 0X11**

Bit(s)	R/W	Default	Description
31:16			reserved
15:0	R	0x0	U48 latched imclk counter48ds of each reg_refclk_latch_interval; Will start from 0 reaching 2^48

**Table 10-490 RO\_REF2OMCLK\_CNT\_L 0X12**

Bit(s)	R/W	Default	Description
31:0	R	0x0	U48 latched omclk counter48ds of each reg_refclk_latch_interval; Will start from 0 reaching 2^48

**Table 10-491 RO\_REF2OMCLK\_CNT\_H 0X13**

Bit(s)	R/W	Default	Description
31:16			reserved
15:0	R	0x0	U48 latched omclk counter48ds of each reg_refclk_latch_interval; Will start from 0 reaching 2^48

**Table 10-492 RO\_IMCLK2REF\_CNT\_L 0X14**

Bit(s)	R/W	Default	Description
31:0	R	0x0	U48 latched reference clock counter48ds of each reg_imclk_latch_interval; Will start from 0 reaching 2^48

**Table 10-493 RO\_IMCLK2REF\_CNT\_H 0X15**

Bit(s)	R/W	Default	Description
31:16			reserved
15:0	R	0x0	U48 latched reference clock counter48ds of each reg_imclk_latch_interval; Will start from 0 reaching 2^48

**Table 10-494 RO\_OMCLK2REF\_CNT\_L 0X16**

Bit(s)	R/W	Default	Description
31:0	R	0x0	U48 latched reference clock counter48ds of each reg_omclk_latch_interval; Will start from 0 reaching 2^48

**Table 10-495 RO\_OMCLK2REF\_CNT\_H 0X17**

Bit(s)	R/W	Default	Description
31:16			reserved
15:0	R	0x0	U48 latched reference clock counter48ds of each reg_omclk_latch_interval; Will start from 0 reaching 2^48

**Table 10-496 RO\_REFCLK\_PKG\_CNT 0X18**

Bit(s)	R/W	Default	Description
31:16	R	0x0	U16,number of reference clk latch interval period conter for imclk latch
15:0	R	0x0	U16,number of reference clk latch interval period conter for imclk latch

**Table 10-497 RO\_IMCLK\_PKG\_CNT 0X19**

Bit(s)	R/W	Default	Description
31:16			reserved
15:0	R	0x0	U16, number of imclk latch interval period counter

**Table 10-498 RO\_OMCLK\_PKG\_CNT 0X1a**

Bit(s)	R/W	Default	Description
31:16			reserved
15:0	R	0x0	U16, number of omclk latch interval period counter

**Table 10-499 RO\_AUD\_LOCK\_INT\_STATUS 0X1b**

Bit(s)	R/W	Default	Description
31:4			reserved
3	R	0x0	It is used to report interrupt status for omclk_state
2	R	0x0	It is used to report interrupt status for imclk_state
1	R	0x0	It is used to report interrupt status for refclk_state1
0	R	0x0	It is used to report interrupt status for refclk_state0

### 10.11.16 VAD Registers

Base Address: 0xFE331800

Each register final address = module base address+ address \* 4

**Table 10-500 VAD\_TOP\_CTRL0 0x0**

Bits	R/W	Default	Description
31	R/W	0	vad_en
30	R/W	0	dec_fir_en
29	R/W	0	pre_emp_en
28	R/W	0	pre_ram_en
27	R/W	0	frame_his_en
26	R/W	0	frame_his_save_en
23	R/W	0	ceps_ceps_en
22	R/W	0	ceps_spec_en
21	R/W	0	post_dec_en
20	R/W	0	two_channel_en
10	R/W	0	soft_rst
9	R/W	0	dec_fir_soft_rst

Bits	R/W	Default	Description
8	R/W	0	pre_emp_soft_rst
7	R/W	0	proc_soft_rst
6	R/W	0	frame_his_soft_rst
5	R/W	0	frame_his_save_soft_rst
4	R/W	0	ceps_win_soft_rst
3	R/W	0	ceps_fft_soft_rst
2	R/W	0	ceps_ceps_soft_rst
1	R/W	0	ceps_spec_soft_rst
0	R/W	0	post_dec_soft_rst

Table 10-501 VAD\_TOP\_CTRL1 0x1

Bits	R/W	Default	Description
31:30	R/W	0	vad_clk_gate
29:28	R/W	0	prepare_clk_gate
27:26	R/W	0	proc_clk_gate
25:24	R/W	0	frame_his_clk_gate
23:22	R/W	0	ceps_ceps_clk_gate
21:20	R/W	0	ceps_spec_clk_gate
19:18	R/W	0	ch_sel1_clk_gate
17:16	R/W	0	ch_sel0_clk_gate
7	R/W	0	vad_sw_reset
6	R/W	0	prepare_sw_reset
5	R/W	0	proc_sw_reset
4	R/W	0	frame_his_sw_reset
3	R/W	0	ceps_win_sw_reset
2	R/W	0	ceps_ceps_sw_reset
1	R/W	0	ceps_spec_sw_reset
0	R/W	0	post_dec_sw_reset

Table 10-502 VAD\_TOP\_CTRL2 0x2

Bits	R/W	Default	Description
25:24	R/W	0	ch1_shift □0:0, 1:1, 2:2, 3:3
9:8	R/W	0	ch0_shift□0:0, 1:1, 2:2, 3:3

**Table 10-503 VAD\_FIR\_CTRL 0x3**

Bits	R/W	Default	Description
29:24	R/W	0	dec_filter_order
15:8	R/W	0	dec_filter_depthm1 ;> = filter pipe+ order
3:0	R/W	0	dec_rate;decimation_rate-1(0~11)

**Table 10-504 VAD\_FIR\_EMP 0x4**

Bits	R/W	Default	Description
9:0	R/W	0	pre_emp_coef :float unsigned

**Table 10-505 VAD\_FIR\_COEF 0 0x5**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_0 : 8.1.1.7
15:0	R/W	0	dec_coef_1 : 8.1.1.7

**Table 10-506 VAD\_FIR\_COEF 1 0x6**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_2 : 8.1.1.7
15:0	R/W	0	dec_coef_3 : 8.1.1.7

**Table 10-507 VAD\_FIR\_COEF 2 0x7**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_4 : 8.1.1.7
15:0	R/W	0	dec_coef_5 : 8.1.1.7

**Table 10-508 VAD\_FIR\_COEF 3 0x8**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_6 : 8.1.1.7
15:0	R/W	0	dec_coef_7 : 8.1.1.7

**Table 10-509 VAD\_FIR\_COEF 4 0x9**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_8 : 8.1.1.7
15:0	R/W	0	dec_coef_9 : 8.1.1.7

**Table 10-510 VAD\_FIR\_COEF 5 0xa**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_10 : 8.1.1.7
15:0	R/W	0	dec_coef_11 : 8.1.1.7

**Table 10-511 VAD\_FIR\_COEF 6 0xb**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_12 : 8.1.1.7
15:0	R/W	0	dec_coef_13 : 8.1.1.7

**Table 10-512 VAD\_FIR\_COEF 7 0xc**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_14 : 8.1.1.7
15:0	R/W	0	dec_coef_15 : 8.1.1.7

**Table 10-513 VAD\_FIR\_COEF 8 0xd**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_16 : 8.1.1.7
15:0	R/W	0	dec_coef_17 : 8.1.1.7

**Table 10-514 VAD\_FIR\_COEF 9 0xe**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_18 : 8.1.1.7
15:0	R/W	0	dec_coef_19 : 8.1.1.7

**Table 10-515 VAD\_FIR\_COEF 10 0xf**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_20 : 8.1.1.7
15:0	R/W	0	dec_coef_21 : 8.1.1.7

**Table 10-516 VAD\_FIR\_COEF 11 0x10**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_22 : 8.1.1.7
15:0	R/W	0	dec_coef_23 : 8.1.1.7



**Table 10-517 VAD\_FIR\_COEF 12 0x11**

Bits	R/W	Default	Description
31:16	R/W	0	dec_coef_24 : 8.1.1.7

**Table 10-518 VAD\_FRAME\_CTRL0 0x12**

Bits	R/W	Default	Description
31:30	R/W	0	frame_len 0: 64, 1: 128, 2: 256
26:24	R/W	0	pow_diff_dist:0:4, 1:8, 3:16, 5:24
17:16	R/W	0	pow_old_avglen:0:4, 1:8,3:16

**Table 10-519 VAD\_FRAME\_CTRL1 0x13**

Bits	R/W	Default	Description
11:0	R/W	0	pow_rampup_thr: 8.16.0.13, threshold for power ramp up detect

**Table 10-520 VAD\_FRAME\_CTRL2 0x14**

Bits	R/W	Default	Description
31:16	R/W	0	pow_cur_thr : 10.0.0.52
15:0	R/W	0	pow_cur_thr_high: 10.0.0.52

**Table 10-521 VAD\_CEP\_CTRL0 0x15**

Bits	R/W	Default	Description
29:28	R/W	0	ceps_decision_sel: 0 :   , 1: &&, 2:naxm 3:maxmin
25:24	R/W	0	decision_sel_each:0 :   , 1: &&, 2:spectral_entropy 3:ceps
20:16	R/W	0	login_min: $2^{-(\text{logn\_min})}$ ,5.5.0, [0~20]
10:0	R/W	0	ceps_log_blk: 11.6.1

**Table 10-522 VAD\_CEP\_CTRL1 0x16**

Bits	R/W	Default	Description
8:0	R/W	0	ceps_weight_step : 9,-3,0

**Table 10-523 VAD\_CEP\_CTRL2 0x17**

Bits	R/W	Default	Description
31:16	R/W	0	ceps_max_thr :16.9.1
15:0	R/W	0	ceps_maxmin_thr :16.9.1

**Table 10-524 VAD\_CEP\_CTRL3 0x18**

Bits	R/W	Default	Description
31:24	R/W	0	ceps_det_start: [0~128]
23:16	R/W	0	ceps_det_end: [0~128], end> start
14:8	R/W	0	ceps_freq_blk_low: 0: no blk, 1:blk[0], 2 blank [255 0 1]...
6:0	R/W	0	ceps_freq_blk_high: 0:no blk, 1:blk[128],2 blk[127:129]...

**Table 10-525 VAD\_CEP\_CTRL4 0x19**

Bits	R/W	Default	Description
30:24	R/W	0	spectral_entropy_start: [0~127]
22:16	R/W	0	spectral_entropy_end: [0~127]

**Table 10-526 VAD\_CEP\_CTRL5 0x1a**

Bits	R/W	Default	Description
27:16	R/W	0	spectral_entropy_thr_max: 12.7.1
11:0	R/W	0	spectral_entropy_thr_min: 12.7.1

**Table 10-527 VAD\_DEC\_CTRL 0x1b**

Bits	R/W	Default	Description
29:28	R/W	0	decision_sel_combine: 0 is &, 1 is  , 2 is +
20:16	R/W	0	vadflag_confirm_m: < 32
5:0	R/W	0	vadflag_confirm_n

**Table 10-528 VAD\_TOP\_STS0 0x1c**

Bits	R/W	Default	Description
4	R	0	Afifo_err
3:0	R	0	laffifo_cnt

**Table 10-529 VAD\_TOP\_STS1 0x1d**

Bits	R/W	Default	Description
20	R	0	Pre_overflow
19	R	0	Pre_fst
18	R	0	Pre_val
17:0	R	0	Pre_dat

**Table 10-530 VAD\_TOP\_STS2 0x1e**

Bits	R/W	Default	Description
18	R	0	Ceps_val
17:0	R	0	Ceps_dat

**Table 10-531 VAD\_FIR\_STS0 0x1f**

Bits	R/W	Default	Description
31:0	R	0	reserved

**Table 10-532 VAD\_FIR\_STS1 0x20**

Bits	R/W	Default	Description
31:-	R	0	reserved

**Table 10-533 VAD\_POW\_STS0 0x21**

Bits	R/W	Default	Description
31:16	R	0	reserved
15:0	R	0	Pow_o

**Table 10-534 VAD\_POW\_STS1 0x22**

Bits	R/W	Default	Description
31:0	R	0	reserved

**Table 10-535 VAD\_POW\_STS2 0x23**

Bits	R/W	Default	Description
31:0	R	0	reserved

**Table 10-536 VAD\_FFT\_STS0 0x24**

Bits	R/W	Default	Description
31	R	0	Ceps_done_latch
30	R	0	Spec_entropy_done
29	R	0	Cepstrum_done

**Table 10-537 VAD\_FFT\_STS1 0x25**

Bits	R/W	Default	Description
11:0	R	0	Se_o

**Table 10-538 VAD\_SPE\_STS0 0x26**

Bits	R/W	Default	Description
31:16	R	0	Cepsw_o_h
15:0	R	0	Cepsw_o_l

**Table 10-539 VAD\_SPE\_STS1 0x27**

Bits	R/W	Default	Description
31:16	R	0	Cur_pow

**Table 10-540 VAD\_SPE\_STS2 0x28**

Bits	R/W	Default	Description
31:0	R	0	Calc_q_pre

**Table 10-541 VAD\_SPE\_STS3 0x29**

Bits	R/W	Default	Description
31:0	R	0	Reserved

**Table 10-542 VAD\_DEC\_STS0 0x2a**

Bits	R/W	Default	Description
31:0	R	0	Reserved

**Table 10-543 VAD\_DEC\_STS1 0x2b**

Bits	R/W	Default	Description
31:0	R	0	Reserved

**Table 10-544 VAD\_LUT\_CTRL 0x2c**

Bits	R/W	Default	Description
31	R/W	0	win_lut_w
30	R/W	0	win_lut_r
6:0	R/W	0	win_lut_addr

**Table 10-545 VAD\_LUT\_WR 0x2d**

Bits	R/W	Default	Description
31:0	R/W	0	Win_lut_wr:12.0.0.7

**Table 10-546 VAD\_LUT\_RD 0x2e**

Bits	R/W	Default	Description
31:0	R	0	Win_lut_rd:12.0.0.7

**Table 10-547 VAD\_IN\_SEL0 0x2f**

Bits	R/W	Default	Description
31:0	R/W	0	ch0_sel: reg_vad.gen_vadin_coeff, bit[n] for ch_n enable

**Table 10-548 VAD\_IN\_SEL1 0x30**

Bits	R/W	Default	Description
31:0	R/W	0	Ch1_sel: reg_vad.gen_vadin_coeff, bit[n] for ch_n enable

**Table 10-549 VAD\_TO\_DDR 0x31**

Bits	R/W	Default	Description
31	R/W	0	vad_to_ddr_en
30	R/W	0	vad_to_ddr_rst
29	R/W	0	vad_to_ddr_eo
27:16	R/W	0	vad_to_ddr_cur_cnt
11:0	R/W	0	vad_to_ddr_max_cnt

### 10.11.17 EQDRC Registers

EQDRC Registers Base Address: 0xFE332000

Each register final address = module base address+ address \* 4

**Table 10-550 EE\_AED\_COEF\_RAM\_CNTL 0x0**

Bits	R/W	Default	Description
31-10	R/W	0x0	reserved;
9-2	R/W	0x0	addr: read or write the addr of coef ram
1	R/W	0x0	rd/wr: 0 read from coef ram; 1 write to coef ram
0	R/W	0x0	valid: 0 the command had been finish; 1 the command wait to be done

**Table 10-551 EE\_AED\_COEF\_RAM\_DATA 0x1**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	data: read from coef ram or write to coef ram

**Table 10-552 EE\_AED\_EQ\_EN 0x2**

Bits	R/W	Default	Description
31-2	R/W	0x0	reserved;
1	R/W	0x0	eq2_en: It is used to enable EQ for ch3 and ch4. 1= enable, 0 = bypass.
0	R/W	0x0	eq1_en: It is used to enable EQ for ch1 and ch2. 1= enable, 0 = bypass.

**Table 10-553 EE\_AED\_TAP\_CNTL 0x3**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
4-0	R/W	0xa	eq2_tap: It is used to set the tap for EQ2.
4-0	R/W	0xa	eq1_tap: It is used to set the tap for EQ1.

**Table 10-554 EE\_AED\_EQ\_VOLUME 0x4**

Bits	R/W	Default	Description
31-30	R/W	0x2	volume_step: It is used to set volume adjust step.
29-26	R/W	0x0	reserved;
25-16	R/W	0x3ff	volume_master: It is used to set master volume.
15-8	R/W	0x30	volume_ch2: It is used to set channel2 volume.
7-0	R/W	0x30	volume_ch1: It is used to set channel1 volume.

**Table 10-555 EE\_AED\_EQ\_VOLUME\_SLEW\_CNT 0x5**

Bits	R/W	Default	Description
31-12	R/W	0x0	reserved;
11-0	R/W	0x0	volume_slew_cnt: It is used to control volume change and mute ramp rate.

**Table 10-556 EE\_AED\_MUTE 0x6**

Bits	R/W	Default	Description
31	R/W	0x0	mute_master: It is used to mute all.
30-2	R/W	0x0	reserved;
1	R/W	0x0	mute_ch2: It is used to mute channel1.
0	R/W	0x0	mute_ch1: It is used to mute channel1.

**Table 10-557 EE\_AED\_DRC\_CNTL 0x7**

Bits	R/W	Default	Description
31-6	R/W	0x0	reserved;
5-3	R/W	0x5	drc_tap: It is used to set the power tap for DRC.

Bits	R/W	Default	Description
2	R/W	0x0	drc_pow_sel: It is used to select the input of gain alpha filter. 0 = dB domain, 1 = decimal domain
1	R/W	0x0	drc_rms_mode: It is used to select the mode of pow calculation. 0 = RMS, 1= peaking
0	R/W	0x0	drc_en: It is used to enable the post DRC. 1= enable, 0 = bypass.

Table 10-558 EE\_AED\_DRC\_RMS\_COEF0 0x8

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x34ebb	It is used to set the coefficient for power calculation filter.

Table 10-559 EE\_AED\_DRC\_RMS\_COEF1 0x9

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7c-b145	It is used to set the coefficient for power calculation filter.

Table 10-560 EE\_AED\_DRC\_THD0 0x0a

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold0 for DRC.

Table 10-561 EE\_AED\_DRC\_THD1 0x0b

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold1 for DRC.

Table 10-562 EE\_AED\_DRC\_THD2 0x0c

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold2 for DRC.

Table 10-563 EE\_AED\_DRC\_THD3 0x0d

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold3 for DRC.

Table 10-564 EE\_AED\_DRC\_THD4 0x0e

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold4 for DRC.

**Table 10-565 EE\_AED\_DRC\_K0 0x0f**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x40000	It is used to set the k-slope of gain for tap 0.

**Table 10-566 EE\_AED\_DRC\_K1 0x10**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x40000	It is used to set the k-slope of gain for tap 1.

**Table 10-567 EE\_AED\_DRC\_K2 0x11**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x40000	It is used to set the k-slope of gain for tap 2.

**Table 10-568 EE\_AED\_DRC\_K3 0x12**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x40000	It is used to set the k-slope of gain for tap 3.

**Table 10-569 EE\_AED\_DRC\_K4 0x13**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x40000	It is used to set the k-slope of gain for tap 4.

**Table 10-570 EE\_AED\_DRC\_K5 0x14**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x40000	It is used to set the k-slope of gain for tap 5.

**Table 10-571 EE\_AED\_DRC\_THD\_OUT0 0x15**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold_out0 for DRC.



**Table 10-572 EE\_AED\_DRC\_THD\_OUT1 0x16**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold_out1 for DRC.

**Table 10-573 EE\_AED\_DRC\_THD\_OUT2 0x17**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold_out2 for DRC.

**Table 10-574 EE\_AED\_DRC\_THD\_OUT3 0x18**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold_out3 for DRC.

**Table 10-575 EE\_AED\_DRC\_OFFSET 0x19**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power offset of DRC.

**Table 10-576 EE\_AED\_DRC\_RELEASE\_COEF00 0x1a**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x5188	It is used to set the coefficient for gain release filter.

**Table 10-577 EE\_AED\_DRC\_RELEASE\_COEF01 0x1b**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7fae78	It is used to set the coefficient for gain release filter.

**Table 10-578 EE\_AED\_DRC\_RELEASE\_COEF10 0x1c**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x5188	It is used to set the coefficient for gain release filter.

**Table 10-579 EE\_AED\_DRC\_RELEASE\_COEF11 0x1d**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7fae78	It is used to set the coefficient for gain release filter.

**Table 10-580 EE\_AED\_DRC\_RELEASE\_COEF20 0x1e**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x5188	It is used to set the coefficient for gain release filter.

**Table 10-581 EE\_AED\_DRC\_RELEASE\_COEF21 0x1f**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7fae78	It is used to set the coefficient for gain release filter.

**Table 10-582 EE\_AED\_DRC\_RELEASE\_COEF30 0x20**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x5188	It is used to set the coefficient for gain release filter.

**Table 10-583 EE\_AED\_DRC\_RELEASE\_COEF31 0x21**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7fae78	It is used to set the coefficient for gain release filter.

**Table 10-584 EE\_AED\_DRC\_RELEASE\_COEF40 0x22**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x5188	It is used to set the coefficient for gain release filter.

**Table 10-585 EE\_AED\_DRC\_RELEASE\_COEF41 0x23**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7fae78	It is used to set the coefficient for gain release filter.

**Table 10-586 EE\_AED\_DRC\_RELEASE\_COEF50 0x24**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x5188	It is used to set the coefficient for gain release filter.

**Table 10-587 EE\_AED\_DRC\_RELEASE\_COEF51 0x25**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7fae78	It is used to set the coefficient for gain release filter.

**Table 10-588 EE\_AED\_DRC\_ATTACK\_COEF00 0x26**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x3263a	It is used to set the coefficient for gain attack filter.

**Table 10-589 EE\_AED\_DRC\_ATTACK\_COEF01 0x27**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7cd9-c6	It is used to set the coefficient for gain attack filter.

**Table 10-590 EE\_AED\_DRC\_ATTACK\_COEF10 0x28**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x3263a	It is used to set the coefficient for gain attack filter.

**Table 10-591 EE\_AED\_DRC\_ATTACK\_COEF11 0x29**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7cd9-c6	It is used to set the coefficient for gain attack filter.

**Table 10-592 EE\_AED\_DRC\_ATTACK\_COEF20 0x2a**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x3263a	It is used to set the coefficient for gain attack filter.

**Table 10-593 EE\_AED\_DRC\_ATTACK\_COEF21 0x2b**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7cd9-c6	It is used to set the coefficient for gain attack filter.

**Table 10-594 EE\_AED\_DRC\_ATTACK\_COEF30 0x2c**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x3263a	It is used to set the coefficient for gain attack filter.

**Table 10-595 EE\_AED\_DRC\_ATTACK\_COEF31 0x2d**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7cd9-c6	It is used to set the coefficient for gain attack filter.

**Table 10-596 EE\_AED\_DRC\_ATTACK\_COEF40 0x2e**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x3263a	It is used to set the coefficient for gain attack filter.

**Table 10-597 EE\_AED\_DRC\_ATTACK\_COEF41 0x2f**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7cd9-c6	It is used to set the coefficient for gain attack filter.

**Table 10-598 EE\_AED\_DRC\_ATTACK\_COEF50 0x30**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x3263a	It is used to set the coefficient for gain attack filter.

**Table 10-599 EE\_AED\_DRC\_ATTACK\_COEF51 0x31**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7cd9-c6	It is used to set the coefficient for gain attack filter.

**Table 10-600 EE\_AED\_DRC\_LOOPBACK\_CNTL 0x32**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
17	R/W	0x0	dis_fifo_rst: 0 if drc_loopback_cnt change, then reset fifo; 1 don't reset

Bits	R/W	Default	Description
16	R/W	0x0	drc_loopback_bypass: It is used to set bypass look-ahead mode. 1= bypass.
15-9	R/W	0x0	reserved;
8-0	R/W	0x90	drc_loopback_cnt: It is used to set the look-ahead count.

Table 10-601 EE\_AED\_MDRC\_CNTL 0x33

Bits	R/W	Default	Description
31-17	R/W	0x0	reserved;
16	R/W	0x0	mdrc_pow_sel: It is used to select the input of gain alpha filter. 0 = dB domain, 1 = decimal domain
15-9	R/W	0x0	reserved;
8	R/W	0x0	mdrc_all_en: It is used to enable the entire multi-band DRC. 1= enable, 0 = bypass.
7-6	R/W	0x0	reserved;
5-3	R/W	0x0	[3] mdrc_rms_mode: It is used to select the band0 mode of pow calculation. 0 = RMS, 1= peaking It is used to select the band1 mode of pow calculation. 0 = RMS, 1= peaking It is used to select the band2 mode of pow calculation. 0 = RMS, 1= peaking
2-0	R/W	0x0	[0] mdrc_en: It is used to enable the band 0 of MDRC. 1= enable, 0 = bypass. It is used to enable the band 1 of MDRC. 1= enable, 0 = bypass. It is used to enable the band 2 of MDRC. 1= enable, 0 = bypass.

Table 10-602 EE\_AED\_MDRC\_RMS\_COEF00 0x34

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x34ebb	It is used to set the coefficient for power calculation filter.

Table 10-603 EE\_AED\_MDRC\_RMS\_COEF01 0x35

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7c-b145	It is used to set the coefficient for power calculation filter.

Table 10-604 EE\_AED\_MDRC\_RELEASE\_COEF00 0x36

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x5188	It is used to set the coefficient for gain release filter.

**Table 10-605 EE\_AED\_MDRC\_RELEASE\_COEF01 0x37**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7fae78	It is used to set the coefficient for gain release filter.

**Table 10-606 EE\_AED\_MDRC\_ATTACK\_COEF00 0x38**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x3263a	It is used to set the coefficient for gain attack filter.

**Table 10-607 EE\_AED\_MDRC\_ATTACK\_COEF01 0x39**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7cd9-c6	It is used to set the coefficient for gain attack filter.

**Table 10-608 EE\_AED\_MDRC\_THD0 0x3a**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold for DRC.

**Table 10-609 EE\_AED\_MDRC\_K0 0x3b**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the k-slope of gain.

**Table 10-610 EE\_AED\_MDRC\_LOW\_GAIN 0x3c**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the gain for DRC band 0.

**Table 10-611 EE\_AED\_MDRC\_OFFSET0 0x3d**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power offset of DRC.

**Table 10-612 EE\_AED\_MDRC\_RMS\_COEF10 0x3e**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x34ebb	It is used to set the coefficient for power calculation filter.

**Table 10-613 EE\_AED\_MDRC\_RMS\_COEF11 0x3f**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7c-b145	It is used to set the coefficient for power calculation filter.

**Table 10-614 EE\_AED\_MDRC\_RELEASE\_COEF10 0x40**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x5188	It is used to set the coefficient for gain release filter.

**Table 10-615 EE\_AED\_MDRC\_RELEASE\_COEF11 0x41**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7fae78	It is used to set the coefficient for gain release filter.

**Table 10-616 EE\_AED\_MDRC\_ATTACK\_COEF10 0x42**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x3263a	It is used to set the coefficient for gain attack filter.

**Table 10-617 EE\_AED\_MDRC\_ATTACK\_COEF11 0x43**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7cd9-c6	It is used to set the coefficient for gain attack filter.

**Table 10-618 EE\_AED\_MDRC\_THD1 0x44**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold for DRC.

**Table 10-619 EE\_AED\_MDRC\_K1 0x45**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the k-slope of gain.

**Table 10-620 EE\_AED\_MDRC\_OFFSET1 0x46**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power offset of DRC.

**Table 10-621 EE\_AED\_MDRC\_MID\_GAIN 0x47**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the gain for DRC band 1.

**Table 10-622 EE\_AED\_MDRC\_RMS\_COEF20 0x48**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x34ebb	It is used to set the coefficient for power calculation filter.

**Table 10-623 EE\_AED\_MDRC\_RMS\_COEF21 0x49**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7c-b145	It is used to set the coefficient for power calculation filter.

**Table 10-624 EE\_AED\_MDRC\_RELEASE\_COEF20 0x4a**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x5188	It is used to set the coefficient for gain release filter.

**Table 10-625 EE\_AED\_MDRC\_RELEASE\_COEF21 0x4b**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7fae78	It is used to set the coefficient for gain release filter.



**Table 10-626 EE\_AED\_MDRC\_ATTACK\_COEF20 0x4c**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x3263a	It is used to set the coefficient for gain attack filter.

**Table 10-627 EE\_AED\_MDRC\_ATTACK\_COEF21 0x4d**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x7cd9-c6	It is used to set the coefficient for gain attack filter.

**Table 10-628 EE\_AED\_MDRC\_THD2 0x4e**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power threshold for DRC.

**Table 10-629 EE\_AED\_MDRC\_K2 0x4f**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the k-slope of gain.

**Table 10-630 EE\_AED\_MDRC\_OFFSET2 0x50**

Bits	R/W	Default	Description
31-0	R/W	0x0	It is used to set the power offset of DRC.

**Table 10-631 EE\_AED\_MDRC\_HIGH\_GAIN 0x51**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the gain for DRC band 1.

**Table 10-632 EE\_AED\_ED\_CNTL 0x52**

Bits	R/W	Default	Description
31-2	R/W	0x0	reserved;
1	R/W	0x0	ed_sign:
0	R/W	0x0	ed_int: soft reset

**Table 10-633 EE\_AED\_DC\_EN 0x53**

Bits	R/W	Default	Description
31-1	R/W	0x0	reserved;
0	R/W	0x0	It is used to enable the noise detection. 1= enable, 0 = bypass.

**Table 10-634 EE\_AED\_ND\_LOW\_THD 0x54**

Bits	R/W	Default	Description
31-24	R/W	0x0	reserved;
23-0	R/W	0x100	It is used to set the low threshold for detection.

**Table 10-635 EE\_AED\_ND\_HIGH\_THD 0x55**

Bits	R/W	Default	Description
31-24	R/W	0x0	reserved;
23-0	R/W	0x200	It is used to set the high threshold for detection. If the signal amplitude is bigger than it, the signal is not noise.

**Table 10-636 EE\_AED\_ND\_CNT\_THD 0x56**

Bits	R/W	Default	Description
31-16	R/W	0x0	reserved;
15-0	R/W	0x100	It is used to set the counter threshold for amplitude detection. If the counter for which the signal is smaller than reg_nd_low_thd continuously is bigger than it, the signal is noise.

**Table 10-637 EE\_AED\_ND\_SUM\_NUM 0x57**

Bits	R/W	Default	Description
31-16	R/W	0x0	reserved;
15-0	R/W	0x200	It is used to set the statistical number for sum of signal amplitude.

**Table 10-638 EE\_AED\_ND\_CZ\_SUM 0x58**

Bits	R/W	Default	Description
31-16	R/W	0x0	reserved;
15-0	R/W	0x800	It is used to set the statistical number for zero-crossing.

**Table 10-639 EE\_AED\_ND\_SUM\_THD0 0x59**

Bits	R/W	Default	Description
31-0	R/W	0x20000	It is used to set the low threshold for sum of amplitude.

**Table 10-640 EE\_AED\_ND\_SUM\_THD1 0x5a**

Bits	R/W	Default	Description
31-0	R/W	0x30000	It is used to set the high threshold for sum of amplitude.

**Table 10-641 EE\_AED\_ND\_CZ\_THD0 0x5b**

Bits	R/W	Default	Description
31-16	R/W	0x0	reserved;
15-0	R/W	0x200	It is used to set the low threshold for zero-crossing number.

**Table 10-642 EE\_AED\_ND\_CZ\_THD1 0x5c**

Bits	R/W	Default	Description
31-16	R/W	0x0	reserved;
15-0	R/W	0x100	It is used to set the high threshold for zero-crossing number.

**Table 10-643 EE\_AED\_ND\_COND\_CNTL 0x5d**

Bits	R/W	Default	Description
31-6	R/W	0x0	reserved;
5	R/W	0x0	Reg_nd_audio_cond2_en: It is used to enable the audio detection for amplitude.
4	R/W	0x0	Reg_nd_audio_cond1_en: It is used to enable the audio detection for amplitude.
3	R/W	0x0	Reg_nd_audio_cond0_en: It is used to enable the audio detection for amplitude.
2	R/W	0x0	Reg_nd_noise_cond2_en: It is used to enable the noise detection for amplitude.
1	R/W	0x0	Reg_nd_noise_cond1_en: It is used to enable the noise detection for amplitude.
0	R/W	0x0	Reg_nd_noise_cond0_en: It is used to enable the noise detection for amplitude.

**Table 10-644 EE\_AED\_ND\_RELEASE\_COEF0 0x5e**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the coefficients for release filter when gain adjusting.

**Table 10-645 EE\_AED\_ND\_RELEASE\_COEF1 0x5f**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the coefficients for release filter when gain adjusting.

**Table 10-646 EE\_AED\_ND\_ATTACK\_COEF0 0x60**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the coefficients for attack filter when gain adjusting.

**Table 10-647 EE\_AED\_ND\_ATTACK\_COEF1 0x61**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the coefficients for attack filter when gain adjusting.

**Table 10-648 EE\_AED\_ND\_ATTACK\_COEF1 0x61**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x0	It is used to set the coefficients for attack filter when gain adjusting.

**Table 10-649 EE\_AED\_ND\_CNTL 0x62**

Bits	R/W	Default	Description
31-3	R/W	0x0	reserved;
2-1	R/W	0x0	nd_gain_sel: It is used to set the gain when noise is detected. 0 = 1/2 1 = 1/4 2 = 1/8 3 = 0
0	R/W	0x0	It is used to enable the dc cut module. 1= enable, 0 = bypass.

**Table 10-650 EE\_AED\_MIX0\_LL 0x63**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x40000	It is used to set the ch1 gain to ch1 mixer.

**Table 10-651 EE\_AED\_MIX0\_RL 0x64**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x40000	It is used to set the ch2 gain to ch1 mixer.

**Table 10-652 EE\_AED\_MIX0\_LR 0x65**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x40000	It is used to set the ch1 gain to ch2 mixer.

**Table 10-653 EE\_AED\_MIX0\_RR 0x66**

Bits	R/W	Default	Description
31-26	R/W	0x0	reserved;
25-0	R/W	0x40000	It is used to set the ch2 gain to ch2 mixer.

**Table 10-654 EE\_AED\_CLIP\_THD 0x67**

Bits	R/W	Default	Description
31-23	R/W	0x0	reserved;
22-0	R/W	0x7ffff	It is used to set the clipping threshold.

**Table 10-655 EE\_AED\_CH1\_ND\_SUM\_OUT 0x68**

Bits	R/W	Default	Description
31-0	R	0x0	It is used to report the sum of amplitude of ch1.

**Table 10-656 EE\_AED\_CH2\_ND\_SUM\_OUT 0x69**

Bits	R/W	Default	Description
31-0	R	0x0	It is used to report the sum of amplitude of ch2.

**Table 10-657 EE\_AED\_CH1\_ND\_CZ\_OUT 0x6a**

Bits	R/W	Default	Description
31-16	R	0x0	reserved;
15-0	R	0x0	It is used to report the sum of zero-crossing of ch1.

**Table 10-658 EE\_AED\_CH2\_ND\_CZ\_OUT 0x6b**

Bits	R/W	Default	Description
31-16	R	0x0	reserved;
15-0	R	0x0	It is used to report the sum of zero-crossing of ch2.

**Table 10-659 EE\_AED\_NOISE\_STATUS 0x6c**

Bits	R/W	Default	Description
31-7	R	0x0	reserved;
6	R	0x0	It is used to report the noise flag.
5-3	R	0x0	It is used to report the audio state.
2-0	R	0x0	It is used to report the noise state.

**Table 10-660 EE\_AED\_POW\_CURRENT\_S0 0x6d**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the mdrc0 current power.

**Table 10-661 EE\_AED\_POW\_CURRENT\_S1 0x6e**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the mdrc1 current power.

**Table 10-662 EE\_AED\_POW\_CURRENT\_S2 0x6f**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the mdrc2 current power.

**Table 10-663 EE\_AED\_POW\_OUT0 0x70**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the mdrc0 destination power.

**Table 10-664 EE\_AED\_POW\_OUT1 0x71**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the mdrc1 destination power.

**Table 10-665 EE\_AED\_POW\_OUT2 0x72**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the mdrc2 destination power.

**Table 10-666 EE\_AED\_ADJ\_INDEX0 0x73**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R/W	0x0	It is used to report the mdrc0 adjusting power.

**Table 10-667 EE\_AED\_ADJ\_INDEX1 0x74**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the mdrc1 adjusting power.

**Table 10-668 EE\_AED\_ADJ\_INDEX2 0x75**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the mdrc2 adjusting power.

**Table 10-669 EE\_AED\_DRC\_GAIN\_INDEX0 0x76**

Bits	R/W	Default	Description
31-26	R	0x0	reserved;
25-0	R	0x0	It is used to report mdrc0 the gain.

**Table 10-670 EE\_AED\_DRC\_GAIN\_INDEX1 0x77**

Bits	R/W	Default	Description
31-26	R	0x0	reserved;
25-0	R	0x0	It is used to report mdrc0 the gain.

**Table 10-671 EE\_AED\_DRC\_GAIN\_INDEX2 0x78**

Bits	R/W	Default	Description
31-26	R	0x0	reserved;
25-0	R	0x0	It is used to report mdrc0 the gain.

**Table 10-672 EE\_AED\_CH1\_VOLUME\_STATE 0x79**

Bits	R/W	Default	Description
31-11	R	0x0	reserved;
10-0	R	0x0	It is used to report the volume state of ch1.

**Table 10-673 EE\_AED\_CH2\_VOLUME\_STATE 0x7a**

Bits	R/W	Default	Description
31-11	R	0x0	reserved;
10-0	R	0x0	It is used to report the volume state of ch2.

**Table 10-674 EE\_AED\_CH1\_VOLUME\_GAIN 0x7b**

Bits	R/W	Default	Description
31-25	R	0x0	reserved;
25-0	R	0x0	It is used to report the volume gain of ch1.

**Table 10-675 EE\_AED\_CH2\_VOLUME\_GAIN 0x7c**

Bits	R/W	Default	Description
31-25	R	0x0	reserved;
25-0	R	0x0	It is used to report the volume gain of ch2.

**Table 10-676 EE\_AED\_FULL\_POW\_CURRENT 0x7d**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the current power.

**Table 10-677 EE\_AED\_FULL\_POW\_OUT 0x7e**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the destination power.

**Table 10-678 EE\_AED\_FULL\_POW\_ADJ 0x7f**

Bits	R/W	Default	Description
-	R	0x0	reserved;
31-0	R	0x0	It is used to report the adjusting power.

**Table 10-679 EE\_AED\_FULL\_DRC\_GAIN 0x80**

Bits	R/W	Default	Description
31-26	R	0x0	reserved;
25-0	R	0x0	It is used to report the gain.



**Table 10-680 EE\_AED\_MASTER\_VOLUME\_STATE 0x81**

Bits	R/W	Default	Description
31-11	R	0x0	reserved;
10-0	R	0x0	It is used to report the volume state of master.

**Table 10-681 EE\_AED\_MASTER\_VOLUME\_GAIN 0x82**

Bits	R/W	Default	Description
31-26	R	0x0	reserved;
25-0	R	0x0	It is used to report the volume gain of master.

**Table 10-682 EE\_AED\_AED\_TOP\_CTL0 0x83**

Bits	R/W	Default	Description
31	R/W	0x0	reg_start_en:write 1 to start eqdrc function,will auto clear to 0
30	R/W	0x0	reg_sf_rst
29	R/W	0x0	reg_cfg_load
28-14	R/W	0x0	reserved;
13-11	R/W	0x0	reg_data_type:0: split 64bits ddr data to 8 sample, each sample need 8 bits; if bitwidth < 8, left-justified; 1: split 64bits ddr data to 4 sample, each sample need 16 bits; if bitwidth < 16, left-justified ; 2: split 64bits ddr data to 4 sample, each sample need 16 bits; if bitwidth < 16, right-justified ; 3: split 64bits ddr data to 2 sample, each sample need 32 bits; if bitwidth < 32, left-justified; 4: split 64bits ddr data to 2 sample, each sample need 32 bits; if bitwidth < 32, right-justified;
10-6	R/W	0x0	reg_data_msb:msb position of data
5-3	R/W	0x0	reg_frddr_source: 0:frddr_A; 1:frddr_B; 2:frddr_C; 3:frddr_D; 4:frddr_E;
2	R/W	0x0	reg_afifo_in_rst: reset afifo in side;
1	R/W	0x0	reg_afifo_out_rst:reset afifo out side;
0	R/W	0x0	req_eqdrc_en:eqdrc module enable

**Table 10-683 EE\_AED\_AED\_TOP\_CTL1 0x84**

Bits	R/W	Default	Description
23-20	R/W	0x0	reg_ch2_sel:channel 2 data source select
19-16	R/W	0x0	reg_ch1_sel:channel 1 data source select
5	R/W	0x0	reg_eq_ch2_en :channel 2 eq enable

Bits	R/W	Default	Description
4	R/W	0x0	reg_eq_ch1_en :channel 1 eq enable
3-0	R/W	0x0	reg_channel_valid: Valid channel number -1

Table 10-684 EE\_AED\_AED\_TOP\_CTL2 0x85

Bits	R/W	Default	Description
31	R/W	0x0	reg_ini_done_rst,when enable will auto rst at init_done negedge
30-20	R/W	0x0	reserved;
19-12	R/W	0x0	reg_ack_num: if needn't share buffer, set it as 0; if need share buffer , set it as TDMOUT/SPDIFOUT ch number;
11	R/W	0x0	reg_req_sel2_en
10-8	R/W	0x0	reg_req_sel2: 0: tdmout_a; 1: tdmout_b; 2: tdmout_c; 3: spdifout; 4: spdifout_b; 5: reserved; 6: reserved; 7: reserved;
7	R/W	0x0	reg_req_sel1_en
6-4	R/W	0x0	reg_req_sel1: 0: tdmout_a; 1: tdmout_b; 2: tdmout_c; 3: spdifout; 4: spdifout_b; 5: reserved; 6: reserved; 7: reserved;
3	R/W	0x0	reg_req_sel0_en
2-0	R/W	0x0	req_req_sel0: 0: tdmout_a; 1: tdmout_b; 2: tdmout_c; 3: spdifout; 4: spdifout_b; 5: reserved; 6: reserved; 7: reserved;

Table 10-685 EE\_AED\_AED\_TOP\_ST 0x86

Bits	R/W	Default	Description
29-20	RO	0x0	tofifo_dat_buf_valid_num
19-10	RO	0x0	frddr_dat_buf_valid_num
9	RO	0x0	r_frddr_init_done
8	RO	0x0	frddr_start

Bits	R/W	Default	Description
7-4	RO	0x0	affo_in_cnt
3-0	RO	0x0	affo_out_cnt

### 10.11.18 EARC RX Registers

#### EARCRX\_CMDC Registers

Base Address: 0xFF663800

Each register final address = module base address+ address \* 4

**Table 10-686 EARC\_RX\_CMDC\_TOP\_CTRL0 0x0**

Bit(s)	R/W	Default	description
31	R/W	0	idle2_int 1: enable
30	R/W	0	idle1_int 1: enable
29	R/W	0	disc2_int 1: enable
28	R/W	0	disc1_int 1: enable
27	R/W	0	earc_int 1: enable
26	R/W	0	hb_status_int 1: enable
25	R/W	0	losthb_int 1: enable
24	R/W	0	timeout_int 1: enable
23	R/W	0	status_ch_int 1: enable
22	R/W	0	int_rec_invalid_id 1: enable
21	R/W	0	int_rec_invalid_offset 1: enable
20	R/W	0	int_rec_unexp 1: enable
19	R/W	0	int_rec_ecc_err 1: enable
18	R/W	0	int_rec_parity_err 1: enable
17	R/W	0	int_rcv_packet 1: enable
16	R/W	0	int_rec_time_out 1: enable
15	R/W	0	cmdc_debug0 1: enable
14	R/W	0	cmdc_debug1 1: enable
13	R/W	0	cmdc_debug2 1: enable
12~7	R/W		reserved
6	R/W	0	mute_select 1: use bit5, 0: earc off
5	R/W	0	mute_ctrl value of mannul mute control
4~0	R/W		reserved

**Table 10-687 EARC\_RX\_CMDC\_TOP\_CTRL1 0x1**

Bit(s)	R/W	Default	Description
31~13	R/W		reserved
12~8	R/W	RW, 0	reg_scan_reg
7~5	R/W		reserved
4~0	R/W	RW, 0	reg_top_soft_rst

**Table 10-688 EARC\_RX\_CMDC\_TOP\_CTRL2 0x2**

Bit(s)	R/W	Default	Description
31	R/W	0	reset_idle2_int
30	R/W	0	reset_idle1_int
29	R/W	0	reset_disc2_int
28	R/W	0	reset_disc1_int
27	R/W	0	reset_earc_int
26	R/W	0	reset_hb_status_int
25	R/W	0	reset_losthb_int
24	R/W	0	reset_timeout_int
23	R/W	0	reset_status_ch_int
22	R/W	0	reset_int_rec_invalid_id
21	R/W	0	reset_int_rec_invalid_offset
20	R/W	0	reset_int_rec_unexp
19	R/W	0	reset_int_rec_ecc_err
18	R/W	0	reset_int_rec_parity_err
17	R/W	0	reset_int_rcv_packet
16	R/W	0	reset_int_rec_time_out
15~00	R/W		reserved

**Table 10-689 EARC\_RX\_CMDC\_TIMER\_CTRL0 0x3**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status0

**Table 10-690 EARC\_RX\_CMDC\_TIMER\_CTRL1 0x4**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status0

**Table 10-691 EARC\_RX\_CMDC\_TIMER\_CTRL2 0x5**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status0

**Table 10-692 EARC\_RX\_CMDC\_TIMER\_CTRL3 0x6**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status0

**Table 10-693 EARC\_RX\_CMDC\_VSM\_CTRL0 0x7**

Bit(s)	R/W	Default	Description
31	R/W	0	sw_state_update
30~28	R/W	0	sw_state
27	R/W	0	arc_initiated
26	R/W	0	arc_terminated
25	R/W	0	arc_enable
24	R/W	0	man_hpd
23~22	R/W	0	hpd_sel
21~20	R/W	0	hpd_sel_earc
19	R/W	0	comma_cnt_rst
18	R/W	0	timeout_status_rst
17	R/W	0	losthb_status_rst
16	R/W	0	force_rst
15	R/W	0	auto_state
14	R/W	0	cmdc_state_en
13~00	R/W		reserved

**Table 10-694 EARC\_RX\_CMDC\_VSM\_CTRL1 0x8**

Bit(s)	R/W	Default	Description
31~12	R/W	0	max_count_th idle done timing
11~8	R/W		reserved
7	R/W	0	reg_soft_rst idle done timing
6~4	R/W	0	time_sel idle done timing
3~2	R/W	0	soft_rst_sel idle done timing
1~0	R/W	0	enable_ctrl idle done timing

**Table 10-695 EARC\_RX\_CMDC\_VSM\_CTRL2 0x9**

Bit(s)	R/W	Default	Description
31~12	R/W	0	max_count_th comma off done timing
11~8	R/W		reserved
7	R/W	0	reg_soft_rst comma off done timing
6~4	R/W	0	time_sel comma off done timing
3~2	R/W	0	soft_rst_sel comma off done timing
1~0	R/W	0	enable_ctrl comma off done timing

**Table 10-696 EARC\_RX\_CMDC\_VSM\_CTRL3 0xa**

Bit(s)	R/W	Default	Description
31~12	R/W	0	max_count_th earc_time out timing
11~8	R/W		reserved
7	R/W	0	reg_soft_rst earc_time out timing
6~4	R/W	0	time_sel earc_time out timing
3~2	R/W	0	soft_rst_sel earc_time out timing
1~0	R/W	0	enable_ctrl earc_time out timing

**Table 10-697 EARC\_RX\_CMDC\_VSM\_CTRL4 0xb**

Bit(s)	R/W	Default	Description
31~12	R/W	0	max_count_th heartbeat lost timing
11~8	R/W		reserved
7	R/W	0	reg_soft_rst heartbeat lost timing
6~4	R/W	0	time_sel heartbeat lost timing
3~2	R/W	0	soft_rst_sel heartbeat lost timing
1~0	R/W	0	enable_ctrl heartbeat lost timing

**Table 10-698 EARC\_RX\_CMDC\_VSM\_CTRL5 0xc**

Bit(s)	R/W	Default	Description
31~16	R/W		reserved
15~8	R/W	0	status_soft in earc heartbeat det timing
7	R/W	0	reg_soft_rst in earc heartbeat det timing
6	R/W	0	status_rst in earc heartbeat det timing
5~4	R/W		reserved
3~2	R/W	0	soft_rst_sel in earc heartbeat det timing
1~0	R/W	0	enable_ctrl in earc heartbeat det timing

**Table 10-699 EARC\_RX\_CMDC\_VSM\_CTRL6 0xd**

Bit(s)	R/W	Default	Description
31~17	R/W		reserved
16	R/W	0	cntl_hpd_sel in earc heartbeat det timing
15~4	R/W	0	cntl_hpd_valid_width in earc heartbeat det timing
3~0	R/W	0	cntl_hpd_glitch_width in earc heartbeat det timing

**Table 10-700 EARC\_RX\_CMDC\_VSM\_CTRL7 0xe**

Bit(s)	R/W	Default	Description
31~00	R/W	0	vsm_ctrl7

**Table 10-701 EARC\_RX\_CMDC\_VSM\_CTRL8 0xf**

Bit(s)	R/W	Default	Description
31~00	R/W	0	vsm_ctrl8

**Table 10-702 EARC\_RX\_CMDC\_VSM\_CTRL9 0x10**

Bit(s)	R/W	Default	Description
31~00	R/W	0	vsm_ctrl9

**Table 10-703 EARC\_RX\_CMDC\_SENDER\_CTRL0 0x11**

Bit(s)	R/W	Default	Description
31~02	R/W		reserved
1	R/W	0	hb_chg_conf_auto
0	R/W	1,	hb_chg_auto

**Table 10-704 EARC\_RX\_CMDC\_PACKET\_CTRL0 0x12**

Bit(s)	R/W	Default	Description
31	R/W	0	packet_mode_enable packet control
30	R/W	0	free_enable packet control
29	R/W	0	soft_rst_man packet control
28~24	R/W	0	ready_th packet control
23~20	R/W		reserved
19~8	R/W	0	send_pre_th packet control
7~5	R/W		reserved

Bit(s)	R/W	Default	Description
4	R/W	0	sw_state_update packet control
3~0	R/W	0	sw_state packet control

**Table 10-705 EARC\_RX\_CMDC\_PACKET\_CTRL1 0x13**

Bit(s)	R/W	Default	Description
31	R/W	0	ecc_endian send
30	R/W	0	pre_reg_st send
29~21	R/W		reserved
20~16	R/W	0	post_th send
15~14	R/W		reserved
13~8	R/W	0	pre_th
7~0	R/W	0	post_flag

**Table 10-706 EARC\_RX\_CMDC\_PACKET\_CTRL2 0x14**

Bit(s)	R/W	Default	Description
31~0	R/W	X	pre_flag unsigned,

**Table 10-707 EARC\_RX\_CMDC\_PACKET\_CTRL3 0x15**

Bit(s)	R/W	Default	Description
31	R/W	0	cmdc_en
30	R/W	0	cmdc_parity_mask
29	R/W	0	imeout_en w
28	R/W	0	ecc_check_en
27	R/W	0	rev_debug_en
26~16	R/W		reserved
15~0	R/W	0	timeout_th X

**Table 10-708 EARC\_RX\_CMDC\_PACKET\_CTRL4 0x16**

Bit(s)	R/W	Default	Description
31	R/W	0	ack_ignore
30	R/W	0	cmdc_tail_check_mask
29~20	R/W		reserved
19~0	R/W	0	cmdc_packet_head



**Table 10-709 EARC\_RX\_CMDC\_PACKET\_CTRL5 0x17**

Bit(s)	R/W	Default	Description
31~24	R/W	0	rev_debug_mask
23~20	R/W		reserved
19~0	R/W	0	cmdc_packet_head_mask

**Table 10-710 EARC\_RX\_CMDC\_PACKET\_CTRL6 0x18**

Bit(s)	R/W	Default	Description
31~20	R/W	0	recv_pre_threshold packet control
19~9	R/W		reserved
8	R/W	0	rec_packet_d
7	R/W	0	rec_parity_err_cnt
6	R/W	0	rec_ecc_err_cnt
5	R/W	0	rec_unexp_cnt
4	R/W	0	rec_invalid_offset_cnt
3	R/W	0	rec_invalid_id_cnt
2	R/W	0	rec_timeout_cnt
1	R/W	0	rec_w_cnt
0	R/W	0	rec_r_cnt X

**Table 10-711 EARC\_RX\_CMDC\_BIPHASE\_CTRL0 0x19**

Bit(s)	R/W	Default	Description
31~24	R/W	7,	reg_tns
23~16	R/W	0	delay_th
15~10	R/W		reserved
9	R/W	0	send_ack_en
8	R/W	0	sq_val_en
7	R/W	0	biphase_send_soft_rst
6	R/W	0	comma_soft_rst
5	R/W	0	fifo_rst
4	R/W	0	receiver_no_sender
3	R/W	0	sender_free
2	R/W	0	receiver_send
1	R/W	0	receiver_earc
0	R/W	0	receiver_free

**Table 10-712 EARC\_RX\_CMDC\_BIPHASE\_CTRL1 0x1a**

Bit(s)	R/W	Default	Description
31~16	R/W		reserved
15	R/W	send	ack_val_en
14~8	R/W		reserved
7~0	R/W	0	width send

**Table 10-713 EARC\_RX\_CMDC\_BIPHASE\_CTRL2 0x1b**

Bit(s)	R/W	Default	Description
31	R/W	0	ack_val_en send
30~20	R/W		reserved
19~16	R/W	0	ack_rate comma send
15~00	R/W	0	width comma sen

**Table 10-714 EARC\_RX\_CMDC\_BIPHASE\_CTRL3 0x1c**

Bit(s)	R/W	Default	Description
31~0	R/W	0	biphase_ctrl3

**Table 10-715 EARC\_RX\_CMDC\_DEVICE\_ID\_CTRL 0x1d**

Bit(s)	R/W	Default	Description
31	R/W	0	apb_write apb bus wr/read
30	R/W	0	apb_read apb bus wr/read
29	R/W	0	apb_w_r_done apb bus wr/read
28	R/W	0	apb_w_r_reset apb bus wr/read
27~16	R/W		reserved
15~8	R/W	0	apb_w_r_id apb bus wr/read
7~0	R/W	0	apb_w_r_start_addr apb bus wr/read

**Table 10-716 EARC\_RX\_CMDC\_DEVICE\_WDATA 0x1e**

Bit(s)	R/W	Default	Description
31~08	R/W		reserved
7~00	R/W	0	apb_write_data apb bus wr/rea

**Table 10-717 EARC\_RX\_CMDC\_DEVICE\_RDATA 0x1f**

Bit(s)	R/W	Default	Description
31~08	R/W		reserved
7~00	R/W	0	apb_read_data apb bus wr/rea

**Table 10-718 EARC\_RX\_ANA\_CTRL0 0x20**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl0

**Table 10-719 EARC\_RX\_ANA\_CTRL1 0x21**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl1

**Table 10-720 EARC\_RX\_ANA\_CTRL2 0x22**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl2

**Table 10-721 EARC\_RX\_ANA\_CTRL3 0x23**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl3

**Table 10-722 EARC\_RX\_ANA\_CTRL4 0x24**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl4

**Table 10-723 EARC\_RX\_ANA\_CTRL5 0x25**

Bit(s)	R/W	Default	Description
31~00	R/W	0	ana_ctrl5

**Table 10-724 EARC\_RX\_ANA\_STAT0 0x26**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_ANA_status0

**Table 10-725 EARC\_RX\_CMDC\_STATUS0 0x27**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status0

**Table 10-726 EARC\_RX\_CMDC\_STATUS1 0x28**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status1

**Table 10-727 EARC\_RX\_CMDC\_STATUS2 0x29**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status2

**Table 10-728 EARC\_RX\_CMDC\_STATUS3 0x2a**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status3

**Table 10-729 EARC\_RX\_CMDC\_STATUS4 0x2b**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status4

**Table 10-730 EARC\_RX\_CMDC\_STATUS5 0x2c**

Bit(s)	R/W	Default	Description
31~00	R	0	ro_cmdc_status5

**Table 10-731 EARC\_RX\_CMDC\_STATUS6 0x2d**

Bit(s)	R/W	Default	Description
31	R	0	ro_idle2_int
30	R	0	ro_idle1_int
29	R	0	ro_disc2_int
28	R	0	ro_disc1_int
27	R	0	ro_earc_int
26	R	0	ro_hb_status_int
25	R	0	ro_losthb_int
24	R	0	ro_timeout_int
23	R	0	ro_status_ch_int
22	R	0	ro_int_rec_invalid_id

Bit(s)	R/W	Default	Description
21	R	0	ro_int_rec_invalid_offset
20	R	0	ro_int_rec_unexp
19	R	0	ro_int_rec_ecc_err
18	R	0	ro_int_rec_parity_err
17	R	0	ro_int_recv_packet
16	R	0	ro_int_rec_time_out
15~0	R		reserved

## EARCRX\_DMAC Registers

Base Address: 0xFF663C00

Each register final address = module base address+ address \* 4

**Table 10-732 EARCRX\_DMAC\_TOP\_CTRL0 0x00**

Bit(s)	R/W	default	Description
31	RW	0	top work enable
30	RW	0	top soft reset
29:23	RW	0	reserved
22:20	RW	0	dmac debug select
19:18	RW	0	reserved
17	RW	0	dmac sync without clk
16	RW	0	dmac sync without clk
15	RW	0	rst_n soft reset scan reg
14	RW	0	reserved
13	RW	0	rst_n sync clk_slow scan reg
12	RW	0	rst_n sync clk_analog scan reg
11	RW	0	clk_slow auto gate
10	RW	0	clk_analog auto gate
9:0	RW	0	reserved

**Table 10-733 EARCRX\_DMAC\_SYNC\_CTRL0 0x01**

Bit(s)	R/W	default	Description
31	RW	0	dmac sync module work enable
30	RW	0	affo out reset
29	RW	0	affo in reset
28:17	RW	0	reserved
16	RW	0	data from analog delay enable

Bit(s)	R/W	default	Description
15	RW	0	reserved
14:12	RW	0	delay cycles
11	RW	0	reserved
10:8	RW	0	valid last how many 0 will clear
7	RW	0	reserved
6:4	RW	0	valid last how may 1 will set
3:1	RW	0	reserved
0	RW	0	dmac data invert

Table 10-734 EARCRX\_SPDIFIN\_SAMPLE\_CTRL0 0x03

Bit(s)	R/W	default	Description
31	RW	0	spdif in sample enable
30	RW	0	spdif in invert
29	RW	0	debug single enable
28	RW	0	0 detect by max_width 1 detect by min_width
27:23	RW	0	reserved
22:20	RW	0	value
19:0	RW	0	base timer to detect sample mode change

Table 10-735 EARCRX\_SPDIFIN\_SAMPLE\_CTRL1 0x04

Bit(s)	R/W	default	Description
31	RW	0	0 auto detect sample mode 1 force a fixed sample mode with reg_sample_mode
30	RW	0	reserved
29:20	RW	0	mode0 threathold time
19:10	RW	0	mode1 threathold time
9:0	RW	0	mode2 threathold time

Table 10-736 EARCRX\_SPDIFIN\_SAMPLE\_CTRL2 0x05

Bit(s)	R/W	default	Description
31:30	RW	0	reserved
29:20	RW	0	mode3 threathold time
19:10	RW	0	mode4 threathold time
9:0	RW	0	mode5 threathold time

**Table 10-737 EARCRX\_SPDIFIN\_SAMPLE\_CTRL3 0x06**

Bit(s)	R/W	default	Description
31:24	RW	0	mode0 sample time
23:16	RW	0	mode1 sample time
15:8	RW	0	mode2 sample time
7:0	RW	0	mode3 sample time

**Table 10-738 EARCRX\_SPDIFIN\_SAMPLE\_CTRL4 0x07**

Bit(s)	R/W	default	Description
31:24	RW	0	mode4 sample time
23:16	RW	0	mode5 sample time
15:8	RW	0	mode6 sample time
7:0	RW	0	reserved

**Table 10-739 EARCRX\_SPDIFIN\_SAMPLE\_CTRL5 0x08**

Bit(s)	R/W	default	Description
31	RW	0	dmac_sqout filter enable
30	RW	0	dmac_sqout invert
29:27	RW	0	dmac_sqout filter tick select,0:sys_clk 1:1us 2:10us 3:100us 4:1ms
26:24	RW	0	dmac_sqout filter select
23:20	RW	0	reserved
19:0	RW	0	dmac_sqout filter tick

**Table 10-740 EARCRX\_SPDIFIN\_SAMPLE\_STAT0 0x09**

Bit(s)	R/W	default	Description
31	RO	0	reserved
30:28	RO	0	sample mode
27:18	RO	0	min width timer
17: 8	RO	0	max width timer
7	RO	0	spdif_sqout buf 2
6	RO	0	spdif_sqout
5:0	RO	0	reserved

**Table 10-741 EARCRX\_SPDIFIN\_SAMPLE\_STAT1 0x0a**

Bit(s)	R/W	default	Description
31:27	RO	0	reserved
26:16	RO	0	r_width_min when debug_en valid
15:9	RO	0	reserved
10:0	RO	0	r_width_max when debug_en valid

**Table 10-742 EARCRX\_SPDIFIN\_MUTE\_VAL 0x0b**

Bit(s)	R/W	default	Description
31:0	RW	0	spdif in mute value

**Table 10-743 EARCRX\_SPDIFIN\_CTRL0 0x0c**

Bit(s)	R/W	default	Description
31	RW	0	spdifin work enable
30	RW	0	0 ch_num = 0~383 1 ch_num = 0~1
29:28	RW	0	reserved
27	RW	0	debug enable
26	RW	0	star add ch_cnt to ch_num
25	RW	0	papb check enable
24	RW	0	nonpcm2pcm_th enable
23:12	RW	0	if long time didn't detect PaPb again,will generate irq
11:8	RW	0	for stat1/stat2 select
7	RW	0	mute channel l
6	RW	0	mute channel r
5:4	RW	0	reserved
3	RW	0	valid check enable
2	RW	0	parity check enable
1	RW	0	spdif data invert
0	RW	0	reserved

**Table 10-744 EARCRX\_SPDIFIN\_CTRL1 0x0d**

Bit(s)	R/W	default	Description
31:24	RW	0	reserved
31:24	RW	0	internal irq status clear
23:12	RW	0	mute block check time thd
11:9	RW	0	mute block check tick sel



Bit(s)	R/W	default	Description
8	RW	0	ext 0 sync check for papb
7:0	RW	0	sync 0 mask

Table 10-745 EARCRX\_SPDIFIN\_CTRL2 0x0e

Bit(s)	R/W	default	Description
31:24	RW	0	mute bit in channel st
23:19	RW	0	mute min block number to declare
18	RW	0	mute bit in channel st L or R
17	RW	0	mute block number check enable
16	RW	0	auto clear compress mode when channel status not compress
15	RW	0	auto clear compress mode when nonpcm2pcm
14	RW	0	auto change earc/arc
13	RW	0	user l or r channle status to check papb
12	RW	0	0:data valid after 1 block;1: in 1st block if exit papb ,data valid after papb
11	RW	0	start write toddr 1:from papb check,0 from preamble Z,valid when reg_earcin_check_papb set
10	RW	0	auto reset will detect format change
9	RW	0	compress B pcpd select : 1:next 4th subframe data 0:next sub frame data
8:4	RW	0	papb msb position in data
3	RW	0	when in arc mode,spdif on force enable
2	RW	0	force value
1	RW	0	earc mode force enable
0	RW	0	force value

Table 10-746 EARCRX\_SPDIFIN\_CTRL3 0x0f

Bit(s)	R/W	default	Description
31:16	RW	0	earc mode pa value
15:0	RW	0	earc mode pb value

Table 10-747 EARCRX\_SPDIFIN\_STAT0 0x10

Bit(s)	R/W	default	Description
31	RO	0	r_valid_bit[0]
30	RO	0	r_spdifin_cps
29	RO	0	r_spdif_out_valid_mask
28:19	RO	0	reserved
18	RO	0	r_spdif_lr_flag

Bit(s)	R/W	default	Description
17	RO	0	r_spdifin_v
16	RO	0	r_chst_mute
15:13	RO	0	r_data_rdy[2:0]
12	RO	0	c_spdifin_sel
11	RO	0	c_spdifin_valid_sel
10	RO	0	c_sample_stable_sel
9	RO	0	c_earc_mode
8	RO	0	r_spdifin_en
7	RO	0	r_dmacrx_en
6	RO	0	c_find_papb
5	RO	0	c_valid_change
4	RO	0	c_find_nonpcm2pcm
3	RO	0	c_pcpd_change
2	RO	0	c_ch_status_change
1	RO	0	i_sample_mode_change
0	RO	0	r_parity_err

Table 10-748 EARCRX\_SPDIFIN\_STAT1 0x11

Bit(s)	R/W	default	Description
31:0	RO	0	0: reg_spdifin_stat1 = r_ch_status_l[31:0]; 1: reg_spdifin_stat1 = r_ch_status_l[63:32]; 2: reg_spdifin_stat1 = r_ch_status_l[95:64]; 3: reg_spdifin_stat1 = r_ch_status_l[127:96]; 4: reg_spdifin_stat1 = r_ch_status_l[159:128]; 5: reg_spdifin_stat1 = r_ch_status_l[191:160]; 6: reg_spdifin_stat1 = {r_pc_data,r_pd_data}; 7: reg_spdifin_stat1 = 32'd0; 8: reg_spdifin_stat1 = r_ch_status_r[31:0]; 9: reg_spdifin_stat1 = r_ch_status_r[63:32]; 10: reg_spdifin_stat1 = r_ch_status_r[95:64]; 11: reg_spdifin_stat1 = r_ch_status_r[127:96]; 12: reg_spdifin_stat1 = r_ch_status_r[159:128]; 13: reg_spdifin_stat1 = r_ch_status_r[191:160]; 14: reg_spdifin_stat1 = {r_pc_data,r_pd_data}; 15: reg_spdifin_stat1 = 32'd0;

Table 10-749 EARCRX\_SPDIFIN\_STAT2 0x12

Bit(s)	R/W	default	Description
31:0	RW	0	0: r_z_width 1: {16'd0,r_frame_cnt_min,r_frame_cnt_max}

Table 10-750 EARCRX\_DMAC\_UBIT\_CTRL0 0x13

Bit(s)	R/W	default	Description
31	RW	0	dmac user bit decode enable
30:24	RW	0	iu sync value
23:16	RW	0	generate irq when fifo level pass some threthold
15	RW	0	max distance bewteen IUs to set lost

Bit(s)	R/W	default	Description
14	RW	0	iu sync code enable 0 : all iu to fifo 1 only sync iu packet to fifo
13:12	RW	0	00 off 01 use l channel userbit 10 use r channel userbit 11 user lr channel userbit
11:8	RW	0	max distance bewteen IUs value
7	RW	0	fifo_thd irq enable
6	RW	0	when lost,initial fifo
5	RW	0	fifo initial
4:0	RW	0	user bit position in data

Table 10-751 EARCRX\_IU\_RDATA 0x14

Bit(s)	R/W	default	Description
31:8	RW	0	reserved
7:0	RW	0	iu data,read only

Table 10-752 EARCRX\_ERR\_CORRECT\_CTRL0 0x16

Bit(s)	R/W	default	Description
31	RW	0	err correct work enable
30	RW	0	reserved
29	RW	0	reset afifo out side
28	RW	0	reset afifo in side
27:7	RW	0	reserved
6	RW	0	bch output 16bit data msb is 27 or 19
5	RW	0	bch output data revers
4	RW	0	bch input ecc msb/lbs
3	RW	0	bch input ecc revers
2	RW	0	bch input data revers
1	RW	0	0 off 1 compress audio mode
0	RW	0	force work mode enable

Table 10-753 EARCRX\_ANA\_RST\_CTRL0 0x18

Bit(s)	R/W	default	Description
31	RW	0	analog reset check work enable
30	RW	0	analog reset from register enable
29	RW	0	soft reset value
28	RW	0	analog reset work enable 0: from bit31 1: from bit31 & top_work_en
27	RW	0	reserved
26:23	RW	0	when new format data in, hold reset after N posedge

Bit(s)	R/W	default	Description
22:20	RW	0	earcrx_div2 hold threshold tick select
19:0	RW	0	earcrx_div2 hold threshold

Table 10-754 EARCRX\_ANA\_RST\_CTRL1 0x19

Bit(s)	R/W	default	Description
31	RW	0	filter enable
30:28	RW	0	filter select
27:25	RW	0	filter tick sel,0:sys_clk 1:1us 2:10us 3:100us 4:1ms
24:16	RW	0	filter tick time
15	RW	0	filter enable
14:12	RW	0	filter select
11:9	RW	0	filter tick sel,0:sys_clk 1:1us 2:10us 3:100us 4:1ms
8:0	RW	0	filter tick time

## EARCRX\_TOP Registers

Base Address: 0xFF663E00

Each register final address = module base address+ address \* 4

Table 10-755 EARCRX\_TOP\_CTRL0 0x00

Bit(s)	R/W	default	Description
31:10	RW	0	reserved
9:8	RW	0	top debug select
7	RW	0	force spdif_rx_en to reg_spdif_rx_en_force_value
6	RW	0	value
5	RW	0	force spdif_rx_sqen to reg_spdif_rx_sqe
4	RW	0	value
3	RW	0	force dmacrx_en to reg_dmacrx_en_force_value
2	RW	0	value
1	RW	0	force dmacrx_sqen to reg_dmacrx_sqen_force_value
0	RW	0	value

Table 10-756 EARCRX\_DMAC\_INT\_MASK 0x01

Bit(s)	R/W	default	Description
31:18	RW	0	reserved
17:0	RW	0	dmac int mask

**Table 10-757 EARCRX\_DMAC\_INT\_PENDING 0x02**

Bit(s)	R/W	default	Description
31:18	RW	0	reserved
17:0	RW	0	dmac int pending, 17 :earcrx_ana_rst_c_new_format_set 16 :earcrx_ana_rst_c_earcrx_div2_hold_set 15 :earcrx_err_correct_c_bcherr_int_set 14 :earcrx_err_correct_r_afifo_overflow_set 13 :earcrx_err_correct_r_fifo_overflow_set 12 :earcrx_user_bit_check_r_fifo_overflow 11 :earcrx_user_bit_check_c_fifo_thd_pass 10 :earcrx_user_bit_check_c_u_pk_lost_int_set 9 :earcrx_user_bit_check_c_iu_pk_end 8 :earcrx_biphase_decode_c_chst_mute_clr 7 :earcrx_biphase_decode_c_find_papb 6 :earcrx_biphase_decode_c_valid_change 5 :earcrx_biphase_decode_c_find_nonpcm2pcm 4 :earcrx_biphase_decode_c_pcpd_change 3 :earcrx_biphase_decode_c_ch_status_change 2 :earcrx_biphase_decode_i_sample_mode_change 1 :earcrx_biphase_decode_r_parity_err 0 :earcrx_dmac_sync_afifo_overflow

**Table 10-758 EARCRX\_CMDC\_INT\_MASK 0x03**

Bit(s)	R/W	default	Description
31:16	RW	0	reserved
15:0	RW	0	cmdc int mask

**Table 10-759 EARCRX\_CMDC\_INT\_PENDING 0x04**

Bit(s)	R/W	default	Description
31:16	RW	0	reserved
15:0	RO	0	cmdc int pending, 15 : idle2_int 14 : idle1_int 13 : disc2_int 12 : disc1_int 11 : earc_int 10 : hb_status_int 9 : losthb_int 8 : timeout_int 7 : status_ch_int 6 : int_rec_invalid_id 5 : int_rec_invalid_offset 4 : int_rec_unexp 3 : int_rec_ecc_err 2 : int_rec_parity_err 1 : int_recv_packet 0 : int_rec_time_out

**Table 10-760 EARCRX\_ANA\_CTRL0 0x05**

Bit(s)	R/W	default	Description
31	RW	0	earcrx_en_d2a
30:29	RW	0	reserved
28:24	RW	0	earcrx_cmdcrx_reftrim
23:20	RW	0	earcrx_idr_trim
19:15	RW	0	earcrx_rterm_trim
14:12	RW	0	earcrx_cmdctx_ack_hystrim
11:7	RW	0	earcrx_cmdctx_ack_reftrim
6	RW	0	earcrx_cmdcrx_vrefon_sel
5:4	RW	0	earcrx_cmdcrx_rcfilter_sel
2:0	RW	0	earcrx_cmdcrx_hystrim

**Table 10-761 EARCRX\_ANA\_CTRL1 0x06**

Bit(s)	R/W	default	Description
31	RW	0	earcrx_rterm_cal_rstn
30	RW	0	earcrx_rterm_cal_en
29	RW	0	reserved
28	RW	0	earcrx_rterm_cal_pd
27:24	RW	0	earcrx_rterm_cal_reg
23:16	RW	0	earcrx_reserv
15:12	RW	0	reserved
11	RW	0	earcrx_rxcom_on_sel
10	RW	0	earcrx_idc_sel
9	RW	0	earcrx_cmdcrx_spdif_dat_invsel
8	RW	0	earcrx_dmac_out_invsel
7:4	RW	0	earcrx_cmdcrx_spdif_sqcon
3:0	RW	0	earcrx_dmacrx_sqcon

**Table 10-762 EARCRX\_ANA\_STAT0 0x07**

Bit(s)	R/W	default	Description
31	RO	0	earcrx_rterm_cal_done
30:5	RO	0	reserved
4:0	RO	0	earcrx_rterm_cal_code

**Table 10-763 EARCRX\_PLL\_CTRL0 0x08**

Bit(s)	R/W	default	Description
31:30	RW	0	reserved
29	RW	0	earcrx_pll_self_reset
28	RW	0	earcrx_pll_en
27:25	RW	0	reserved
24	RW	0	earcrx_pll_digital_rstn_sel
23	RW	0	earcrx_pll_dmacrx_sqout_rstn_sel
22:15	RW	0	reserved
14:10	RW	0	earcrx_pll_n
9:0	RW	0	reserved

**Table 10-764 EARCRX\_PLL\_CTRL1 0x09**

Bit(s)	R/W	default	Description
31	RW	0	earcrx_pll_afc_bypass
30:29	RW	0	reserved
28:24	RW	0	earcrx_pll_afc_in
23:22	RW	0	reserved
21:20	RW	0	earcrx_pll_adj_ldo
19	RW	0	reserved
18:16	RW	0	earcrx_pll_alpha
15:14	RW	0	reserved
13:12	RW	0	earcrx_pll_bb_mode
11	RW	0	reserved
10:8	RW	0	earcrx_pll_data_sel
7	RW	0	earcrx_pll_dco_clk_sel
6	RW	0	earcrx_pll_dco_m_en
5	RW	0	earcrx_pll_fast_lock
4	RW	0	earcrx_pll_filter_mode
3	RW	0	earcrx_pll_fix_en
2:0	RW	0	reserved

**Table 10-765 EARCRX\_PLL\_CTRL2 0x0a**

Bit(s)	R/W	default	Description
31:28	RW	0	earcrx_pll_filter_pvt1
27:24	RW	0	earcrx_pll_filter_pvt2
23	RW	0	reserved
22:20	RW	0	earcrx_pll_lambda0
19	RW	0	reserved
18:16	RW	0	earcrx_pll_lambda1
15:14	RW	0	reserved
13:8	RW	0	earcrx_pll_lk_s
7:4	RW	0	earcrx_pll_lk_w
3:2	RW	0	reserved
1:0	RW	0	earcrx_pll_lkw_sel

**Table 10-766 EARCRX\_PLL\_CTRL3 0x0b**

Bit(s)	R/W	default	Description
31	RW	0	earcrx_pll_lock_f
30	RW	0	reserved
29:28	RW	0	earcrx_pll_lock_long
27:26	RW	0	reserved
25:24	RW	0	earcrx_pll_pfd_gain
23:22	RW	0	reserved
21:20	RW	0	earcrx_pll_bias_adj
19	RW	0	reserved
18:16	RW	0	earcrx_pll_rou
15	RW	0	earcrx_pll_tdc_mode
14	RW	0	earcrx_pll_acq_range
13	RW	0	earcrx_pll_dco_sdm_en
12:6	RW	0	reserved
5:0	RW	0	earcrx_pll_reve

**Table 10-767 EARCRX\_PLL\_STAT0 0x0c**

Bit(s)	R/W	default	Description
31	RO	0	earcrx_pll_dmac_valid
30	RO	0	earcrx_pll_dmac_valid_auto
29	RO	0	earcrx_pll_afc_done_a2d
28:10	RO	0	reserved
9:0	RO	0	earcrx_pll_reg_out



# 11 AFIFO

## 11.1 Register Description

For below registers the base address is 0xfe046000.

Each register final address = BASE + address \* 4.

The following lists describe the mapping between each AFIFO register and its address.

AIU_AIFIFO_CTRL	0xfe046000
AIU_AIFIFO_STATUS	0xfe046004
AIU_AIFIFO_GBIT	0xfe046008
AIU_AIFIFO_CLB	0xfe04600c
AIU_MEM_AIFIFO_START_PTR	0xfe046010
AIU_MEM_AIFIFO_CURR_PTR	0xfe046014
AIU_MEM_AIFIFO_END_PTR	0xfe046018
AIU_MEM_AIFIFO_BYTES_AVAIL	0xfe04601c
AIU_MEM_AIFIFO_CONTROL	0xfe046020
AIU_MEM_AIFIFO_MAN_WP	0xfe046024
AIU_MEM_AIFIFO_MAN_RP	0xfe046028
AIU_MEM_AIFIFO_LEVEL	0xfe04602c
AIU_MEM_AIFIFO_BUF_CNTL	0xfe046030
AIU_MEM_AIFIFO_BUF_WRAP_COUNT	0xfe046034
AIU_MEM_AIFIFO_MEM_CTL	0xfe04603c
AIFIFO_TIME_STAMP_CNTL	0xfe046040
AIFIFO_TIME_STAMP_SYNC_0	0xfe046044
AIFIFO_TIME_STAMP_SYNC_1	0xfe046048
AIFIFO_TIME_STAMP_0	0xfe04604c
AIFIFO_TIME_STAMP_1	0xfe046050
AIFIFO_TIME_STAMP_2	0xfe046054
AIFIFO_TIME_STAMP_3	0xfe046058
AIFIFO_TIME_STAMP_LENGTH	0xfe04605c

**Table 11-1 AIU\_AIFIFO\_CTRL**

Bit(s)	R/W	Default	Description
3	R/W	0	CRC pop aififo enable
2	R/W	0	writing to this bit to 1 causes CRC module reset
1	R/W	0	enable aififo
0	R/W	0	writing to this bit to 1 causes aififo soft reset

**Table 11-2 AIU\_AIFIFO\_STATUS**

Bit(s)	R/W	Default	Description
13	R	0	aififo request to dcu status
12	R	0	dcu select status
11:5	R	0	aififo word counter number
4:0	R	0	how many bits left in the first pop register

**Table 11-3 AIU\_AIFIFO\_GBIT**

Bit(s)	R/W	Default	Description
31	R/W	0	Same function as the AIGBIT of AIFIFO in CDROM module write to this register how many bits wanna pop, and reading this register gets the corresponding bits data

**Table 11-4 AIU\_AIFIFO\_CLB**

Bit(s)	R/W	Default	Description
31	R/W	0	Same function as the AICLB of AIFIFO in CDROM module return the leading zeros by reading this registers

**Table 11-5 AIU\_MEM\_AIFIFO\_START\_PTR**

Bit(s)	R/W	Default	Description
31	R/W	0	The AIFIFO start pointer into DDR memory is a 32-bit number The Start pointer will automatically be truncated to land on an 8-byte boundary. That is, bits [2:0] = 0;

**Table 11-6 AIU\_MEM\_AIFIFO\_CURR\_PTR**

Bit(s)	R/W	Default	Description
31	R/W	0	The current pointer points so some location between the START and END pointers. The current pointer is a BYTE pointer. That is, you can point to any BYTE address within the START/END range

**Table 11-7 AIU\_MEM\_AIFIFO\_END\_PTR**

Bit(s)	R/W	Default	Description
31	R/W	0	

**Table 11-8 AIU\_MEM\_AIFIFO\_BYTES\_AVAIL**

Bit(s)	R/W	Default	Description
31	R/W	0	

**Table 11-9 AIU\_MEM\_AIFIFO\_CONTROL**

Bit(s)	R/W	Default	Description
10	R/W	0	use_level Set this bit to 1 to enable filling of the FIFO controlled by the buffer level control. If this bit is 0, then use bit[1] to control the enabling of filling
9	R/W	0	Data Ready. This bit is set when data can be popped
8	R/W	0	fill busy This bit will be high when we're fetching data from the DDR memory To reset this module, set cntl_enable = 0, and then wait for busy = 0. After that you can pulse cntl_init to start over
7	R/W	0	cntl_endian_jic Just in case endian. last minute byte swap of the data out of the FIFO to getbit
5:3	R/W	0	ddr_endian
2	R/W	0	cntl_empty_en Set to 1 to enable reading the DDR memory FIFO and filling the pipeline to get-bit Set cntl_empty_en = cntl_fill_en = 0 when pulsing cntl_init
1	R/W	0	cntl_fill_en Set to 1 to enable reading data from DDR memory
0	R/W	0	cntl_init: After setting the read pointers, sizes, channel masks and read masks, set this bit to 1 and then to 0 NOTE: You don't need to pulse cntl_init if only the start address is being changed

**Table 11-10 AIU\_MEM\_AIFIFO\_MAN\_WP**

Bit(s)	R/W	Default	Description
31	R/W	0	manual write pointer

**Table 11-11 AIU\_MEM\_AIFIFO\_MAN\_RP**

Bit(s)	R/W	Default	Description
31	R/W	0	manual read pointer

**Table 11-12 AIU\_MEM\_AIFIFO\_LEVEL**

Bit(s)	R/W	Default	Description

**Table 11-13 AIU\_MEM\_AIFIFO\_BUF\_CNTL**

Bit(s)	R/W	Default	Description
1	R/W	0	manual mode Set to 1 for manual write pointer mode
0	R/W	0	Init Set high then low after everything has been initialized

**Table 11-14 AIU\_MEM\_AIFIFO\_BUF\_WRAP\_COUNT**

Bit(s)	R/W	Default	Description
31	R/W	0	

**Table 11-15 AIU\_MEM\_AIFIFO\_MEM\_CTL**

Bit(s)	R/W	Default	Description
29:24	R/W	0	A_brst_num
21:16	R/W	0	A_id
15:0	R/W	0	level_hold

**Table 11-16 AIFIFO\_TIME\_STAMP\_CNTL**

Bit(s)	R/W	Default	Description
31:16	R/W	0	drop_bytes
15:14	R	0	drop_status
13:12	R	0	sync_match_position
5:4	R/W	0	TIME_STAMP_NUMBER, 0-32bits, 1-64bits, 2-96bits, 3-128bits
3	R/W	0	stamp_soft_reset
2	R/W	0	TIME_STAMP_length_enable
1	R/W	0	TIME_STAMP_sync64_enable
0	R/W	0	TIME_STAMP_enable

**Table 11-17 AIFIFO\_TIME\_STAMP\_SYNC\_0**

Bit(s)	R/W	Default	Description
31	R/W	0	TIME_STAMP_SYNC_CODE_0

**Table 11-18 AIFIFO\_TIME\_STAMP\_SYNC\_1**

Bit(s)	R/W	Default	Description
31	R/W	0	TIME_STAMP_SYNC_CODE_1

**Table 11-19 AIFIFO\_TIME\_STAMP\_0/1/2/3**

Bit(s)	R/W	Default	Description
31	R/W	0	

**Table 11-20 AIFIFO\_TIME\_STAMP\_LENGTH**

Bit(s)	R/W	Default	Description
31	R/W	0	

# 12 Memory Interface

## 12.1 DDR

### 12.1.1 Overview

DDR subsystem consists of four parts: two DDR memory controllers (DMC) and two DDR PHYs. The main features are:

- Supports up to DDR4-3200, LPDDR4- and LPDDR4/LPDDR4x-4266 with an interleaved 64 bit data bus
- Theoretical memory bandwidth of up to 34.128 GB/s
- Up to 8 GBs of DDR memory space, 16 GBs total address space (34 bit address)
- All DDR data can be optionally obfuscated
- Allows an asymmetric DDR memory configuration with only part of the memory space interleaved to reduce cost
- Secure DDR access control based on AXI bus ARUSER/AWUSER bit field
- Quality of Service (QoS) support using AXI bus ARQOS/AWQOS priority bit field
- Each DMC uses a different clock phase to reduce EMI

**Figure 12-1 DDR Interface**

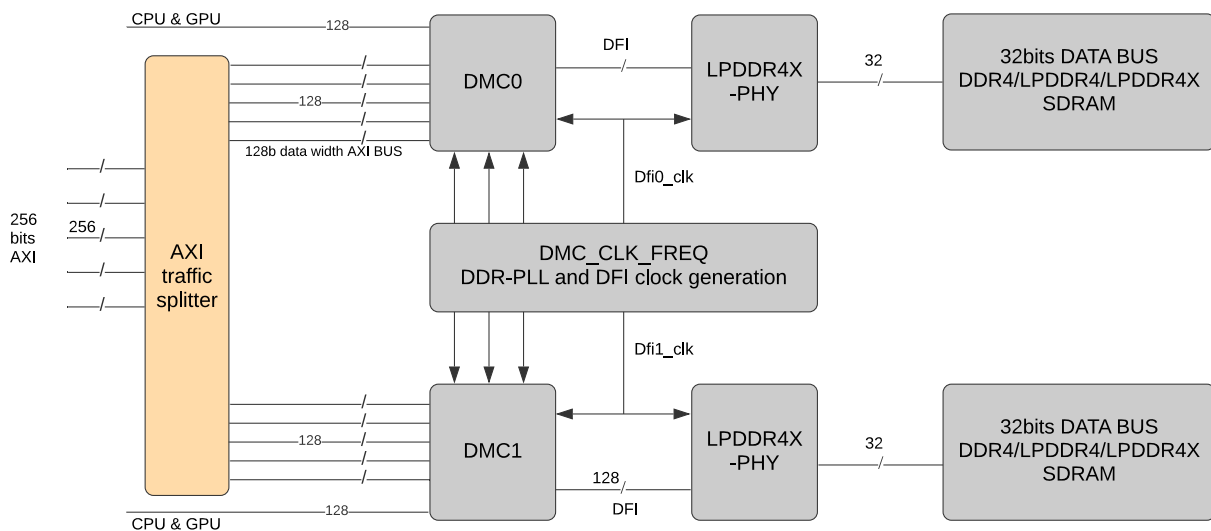
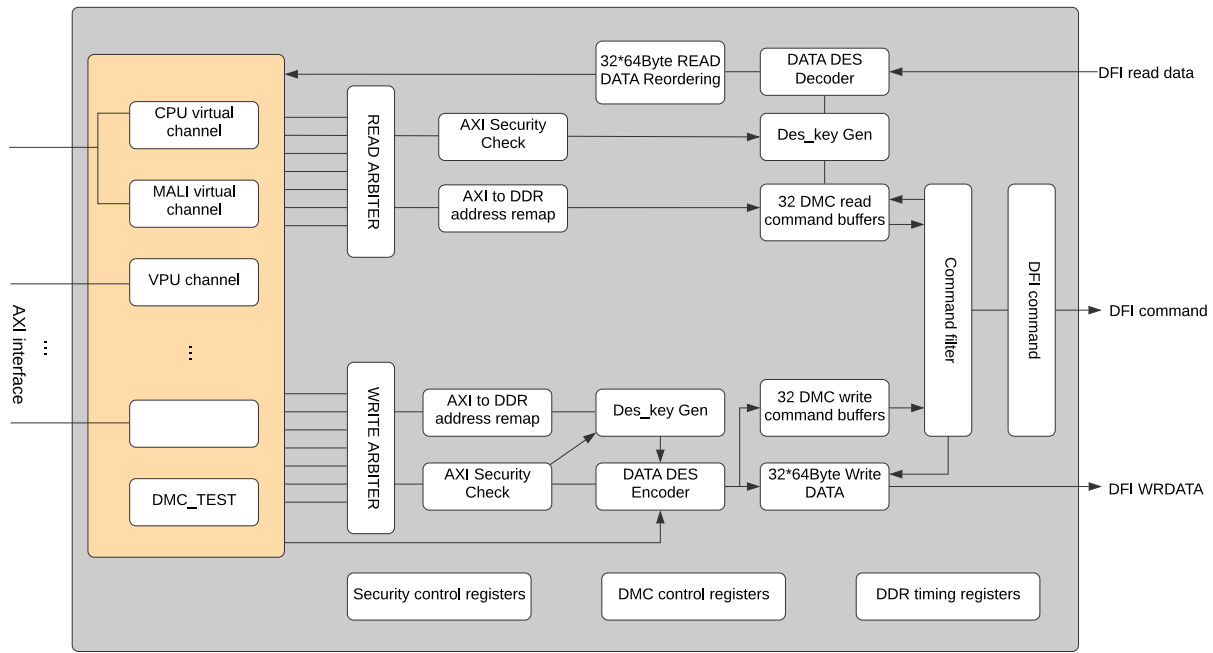


Figure 12-2 DMC Architecture



### 12.1.2 AXI to DDR Address Remapping

16 Gbytes of total AXI address space is remapped into up to 4 Gbytes of DDR address space per DMC controller for a total of up to 8 Gbytes of DDR. Each 512 Mbytes of interleaved DDR memory can be mapped to a different AXI address. The 32 (16 GB/512 MB) remap registers are defined below:

DMC\_AXI2DDR<sub>x</sub>, x = 0x00 to 0x1F

Bit	Register
15:8	Rank_base_addr
7:6	Not used
5	Rank_sel
4	Pure_16b
3	16b_in32
2	Not used
1:0	Map_mode

- **Rank\_base\_addr:** Remaps 256 Mbytes for each DMC controller for a total of 512 Mbytes if interleaved.
- **RANK\_SEL:**
  - 1: this address region is in DDR SDRAM RANK1.
  - 0: this address region is in DDR SDRAM RANK0.
- **Pure\_16b:** this region is a DDR 16bits data bus area in a 16bits DDR SDRAM.
- **16b\_in32:** this region is a DDR 16bits data bus area in the 32bits DDR DATA bus. (2 DDR SDRAMs with different size).

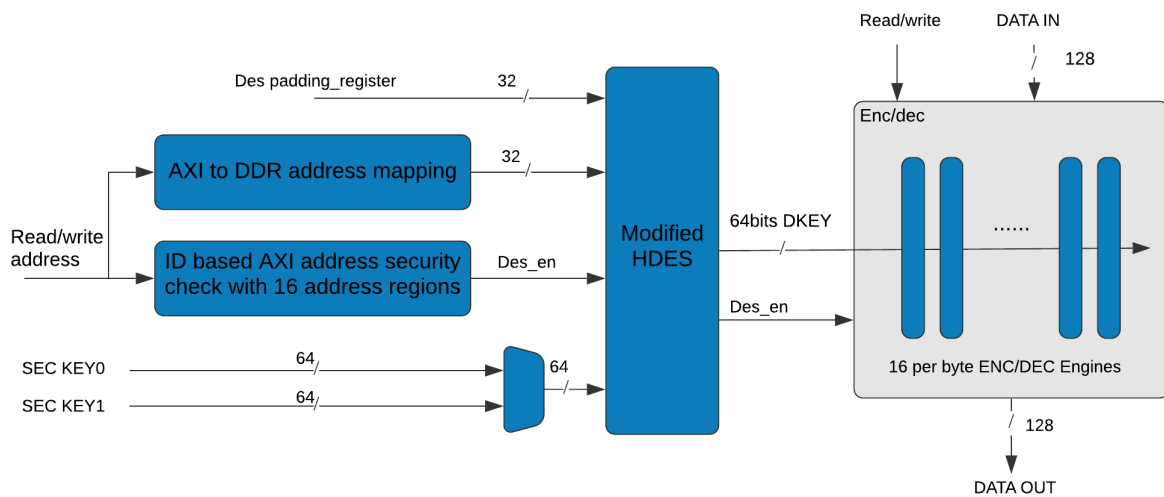
● **Map\_mode:**

- 2'b00, 2'b10: address region is not used by this DMC.
- 2'b01: address region is mapped to only this DMC in non-interleave mode.
- 2'b10: region is mapped to both DMC controller using interleaving.

**12.1.3 DMC AXI Address Security Region Checking and Data Scrambling**

The 16 GB AXI address can be partitioned into 16 security regions inside the DMC. The first 15 regions are defined with start and end addresses with 4KB boundary. The last region is all the remaining address space not defined or enabled in the first 15 regions. Each region has an 8 bit control register and 320 AXI ARUSER/AWUSER master ID-based access control registers to control AXI master access. Each region address space can be overlapped. The security priority is highest for region 0 and decreases all the way down to region 15. For example, if an AXI transaction address exists in security region 0 and region 4, the region 0 security control is used for the transaction.

**Figure 12-3 AXI address security checking and data scrambling**



**Region\_x Control Register**

**Table 12-1 Register Definition**

Bit	Register
7:5	Security_level
4	monitor_en
3	local_des_en
2	Lock
1	Key_sel
0	Enable

- **Security\_level:** Region security levels for DMA/DEMUX.
- **monitor\_en:** Region monitored by DMC\_PROT function. 1= monitored. 0= not monitored.



- **Local\_des\_en:** 1 = Enable DES scrambling for this region. NOTE: The Regionx\_des\_policy and Global\_des\_en bits below affects whether scrambling is actually enabled.
- **Lock:** 1 = All region related registers are locked and cannot be changed (includes this control register, 10 ID based control registers and address range register)
- **Key\_sel:** Selects security key (0 or 1) when scrambling is enabled
- **Enable:** Region enable bit. (NOTE: Region 15 is always enabled and this bit is ignored)

## Global DES Control Register

Table 12-2 Register Definition

Bit	Register
31:16	Regionx_des_policy
15:2	Reserved
1	Global_des_en
0	Lock

- **Lock:** 1 = This register and des\_padding register are locked and cannot be changed
- **Global\_des\_en:** 1 = Enable global DES data scrambling
- **Regionx\_des\_policy:** Each bit decides the scrambling policy for one security region.
  - 1: Region is DES scrambled if region Local\_des\_en && Global\_des\_en.
  - 0: Region is DES scrambled if region Local\_des\_en || Global\_des\_en.

## ARUSER/AWUSER Based AXI Bus Security Information

Domain	Master	Bit Assignment							
		bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7:0 (sys level)
BUS MA-TRIX 0 (10b)	A53						AXI-PROT[2]	AXIPROT [1]	1
	A73						AXI-PROT[2]	AXIPROT [1]	2
	Mali0							PROT-MODE	3
	Mali1							PROT-MODE	4
	BUS MATRIX 0 internal							AXIPROT [1]	5
BUS MA-TRIX 1 (10b)	NNA0								8
	NNA1								9
	NNA2								10
	NNA3								11
BUS MA-TRIX 2	HDMI Rx								16
	usb2 controller								17
	usb3 controller								18

Domain	Master	Bit Assignment								
		bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7:0 (sys level)	
	PCIE controller								19	
	dev0 p0 -SPICC0								24	
	dev0 p1 -SPICC1								25	
	dev0 p2 -SDEMMCA								26	
	dev0 p3 -SDEMMCB								27	
	dev0 p4 -SDEMMCC								28	
	dev0 p5 - SPICC2								29	
	dev0 p6 - reserved								30	
	dev0 p7 - reserved								31	
	dev1 p0 - reserved								32	
	dev1 p1-reserved								33	
	dev1 p2 -ETH								34	
	dev1 p3 -AIFIFO								35	
	dev1 p4 -AUDMA								36	
	dev1 p5 - SPICC3								37	
	dev1 p6 - SPICC4								38	
	dev1 p7 - SPICC5								39	
BUS MA-TRIX 3	GE2D							AxSEC	48	
	GDC								49	
	dewarp						AxID[1:0]	AxSEC	50	
	vdec	AxUSER[6:0]								51
	hevc f	AxUSER[6:0]								52
	hevc b	AxUSER[6:0]								53
	hcodec	AxUSER[6:0]								54
	wave521							AXI-PROT[2]	AXIPROT [1]	55
BUS MA-TRIX 4 (20b)	demux(awuser)						wr_level[2:0]		63	
	demux(ruser)						rd_level[2:0]		63	
	ISP0								64	
	ISP1								65	

Domain	Master	Bit Assignment								
		bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7:0 (sys level)	
	ISP2									66
	ISP3									67
	DSP A						0	0		68
	DSP B						0	0		69
	AO-CPU						0	!HPROT [0]		70
	jtag						0	0		71
	dev2 p0 - AUDIO					AUDIO_SUBID[3:0]				72
	dev2 p1 - DMA	wr_level[2:0]			DMA_SUBID[3:0]				73	
	dev2 p1 - DMA (ruser)				RD-ERR	rd_level[2:0]				73
	dev2 p1 - DMA (buser)							WRERR	73	
	dev2 p2 - reserved								74	
	dev2 p3 - reserved								75	
	dev2 p4 - reserved								76	
	dev2 p5 - reserved								77	
	dev2 p6 - reserved								78	
dev2 p7 - reserved								79		
BUS MATRIX 5 (20b)	VPU0			AxID[3:0]				AxSEC	80	
	VPU1			AxID[3:0]				AxSEC	81	
	VPU2			AxID[3:0]				AxSEC	82	

```

//the 128 read/write control bits defines as below.
//bit 11:0. for BUS MATRIX 0 domain.
//bit [11:0] = {Bus Matrix 0_non_sec, Bus Matrix 0_sec, mali_non_prot, mali_prot,
//a73_non_sec_inst, a73_sec_inst, a73_non_sec_data, a73_sec_data,
//a53_non_sec_inst, a53_sec_inst, a53_non_sec_data, a53_sec_data };

//bit 15:12 for BUS MATRIX 1 nna domain.
// 15:12 = {nna3, nna2, nna1, nna0}

//bit 37:16 for BUS MATRIX 2 device0
// 37:16 = { not_used, spicc5, spicc4, spicc3, audma, aififo,
// eth, dev0_p1, dev0_p0, dev0_p7, dev0_p6, spicc2, sdeimmcc, sdeimmcb,
// sdeimmca, spicc1, spicc0, not used, pcie, usb3, usb2, hdmi_rx };

//bit 78:38 for BUS MATRIX 3 ge2d,gdc, hcodec, vdec, hevc, wave521 etc.
//[78:38] Bus Matrix 3_sid_sel = { wave11, wave10, wave01, wave00,
//dewap_nonsec, dewap_sec, gdc, ge2d_nonsec, ge2d_sec,
// hevcf_7, hevcf_6, hevcf_5, hevcf_4, hevcf_3, hevcf_2, hevcf_1, hevcf_0,
//hevcb_7, hevcb_6, hevcb_5, hevcb_4, hevcb_3, hevcb_2, hevcb_1, hevcb_0,
//hcodec_7, hcodec_6, hcodec_5, hcodec_4, hcodec_3, hcodec_2, hcodec_1, hcodec_0,
//vdec_7, vdec_6, vdec_5, vdec_4, vdec_3, vdec_2, vdec_1, vdec_0};
//bit 119:80 for BUS MATRIX 4 device 2 aocpu dsp isp.
    
```

```
// { demux, jtag, dev2_p7, dev2_p6, dev2_p5, dev2_p4, dev2_p3, dev2_p2,
// dma_7, dma_6, dma_5, dma_4, dma_3, dma_2, dma_1, dma_0,
// audio_15, audio_14, audio_13, audio_12, audio_11, audio_10, audio_9, audio_8,
// audio_7, audio_6, audio_5, audio_4, audio_3, audio_2, audio_1, audio_0,
// aocpu_1, aocpu_0, dsp_b, dsp_a, isp3, isp2, isp1, isp0};
//bit 126:120. for Bus Matrix 5 VPU
// { dmc_test, vpu2_nosec, vpu2_sec, vpu1_nosec, vpu1_sec, vpu0_nosec, vpu0_sec}
```

### 12.1.4 DMC Security Register Bit Assignment

Register Control Bit	Master IDs Security Item	Special Requirement
0	a53_sec_data	
1	a53_non_sec_data	
2	a53_sec_inst	
3	a53_non_sec_inst	
4	a73_sec_data	
5	a73_non_sec_data	
6	a73_sec_inst	
7	a73_non_sec_inst	
8	mali_prot	
9	mali_non_prot	
10	Bus Matrix 0_sec	
11	Bus Matrix 0_non_sec	
12	nna0	
13	nna1	
14	nna2	
15	nna3	
16	hdmi_rx	
17	usb2	
18	usb3	
19	pcie	
20	not used	
21	spicc0	
22	spicc1	
23	sdemmcA	
24	sdemmcB	
25	sdemmcC	
26	spicc2	
27	not used	
28	not used	
29	not used	

Register Control Bit	Master IDs Security Item	Special Requirement
30	not used	
31	ETHNET	
32	aiffo	
33	audma	
34	spicc3	
35	spicc4	
36	spicc5	
37	not used	
38	vdec_lvl0	
39	vdec_lvl1	
40	vdec_lvl2	
41	vdec_lvl3	
42	vdec_lvl4	
43	vdec_lvl5	
44	vdec_lvl6	
45	vdec_lvl7	
46	hcodec_lvl0	
47	hcodec_lvl1	
48	hcodec_lvl2	
49	hcodec_lvl3	
50	hcodec_lvl4	
51	hcodec_lvl5	
52	hcodec_lvl6	
53	hcodec_lvl7	
54	hevcb_lvl0	
55	hevcb_lvl1	
56	hevcb_lvl2	
57	hevcb_lvl3	
58	hevcb_lvl4	
59	hevcb_lvl5	
60	hevcb_lvl6	
61	hevcb_lvl7	
62	hevcb_lvl0	
63	hevcb_lvl1	
64	hevcb_lvl2	
65	hevcb_lvl3	

Register Control Bit	Master IDs Security Item	Special Requirement
66	hevcf_lvl4	
67	hevcf_lvl5	
68	hevcf_lvl6	
69	hevc_f_lv7	
70	ge2d_sec	
71	ge2d_nonsec	
72	gdc	
73	dewap_sec	
74	dewap_nonsec	
75	wave_sid0	
76	wave_sid1	
77	wave_sid2	
78	wave_sid3	
79	not used	
80	isp0	
81	isp1	
82	isp2	
83	isp3	
84	dsp_a	
85	dsp_b	
86	aocpu_0	
87	aocpu_1	
88	audio_sid0	
89	audio_sid1	
90	audio_sid2	
91	audio_sid3	
92	audio_sid4	
93	audio_sid5	
94	audio_sid6	
95	audio_sid7	
96	audio_sid8	
97	audio_sid9	
98	audio_sid10	
99	audio_sid11	
100	audio_sid12	

Register Control Bit	Master IDs Security Item	Special Requirement
101	audio_sid13	
102	audio_sid14	
103	audio_sid15	
104	dma_thread0	sec_lvl <= region_lvl for write
105	dma_thread1	sec_lvl <= region_lvl for write
106	dma_thread2	sec_lvl <= region_lvl for write
107	dma_thread3	sec_lvl <= region_lvl for write
108	dma_thread4	sec_lvl <= region_lvl for write
109	dma_thread5	sec_lvl <= region_lvl for write
110	dma_thread6	sec_lvl <= region_lvl for write
111	dma_thread7	sec_lvl <= region_lvl for write
112	dev_port2	
113	dev_port3	
114	dev_port4	
115	dev_port5	
116	dev_prot6	
117	dev_prot7	
118	JTAG	
119	DEMUX	sec_lvl <= region_lvl for write
120	vpu0_sec	
121	vpu0_non-sec	
122	vpu1_sec	
123	vpu1_non_sec	
124	vpu2_sec	
125	vpu2_non_sec	
126	dmc_test	

### 12.1.5 AXI0 CPU & GPU Virtual Channel

AXI0 channel is from BUS MATRIX 0 for both CPU and GPU. But CPU and GPU have different QOS controls. CPU and GPU traffic are separated into two different virtual channels inside the DMC. The CPU traffic goes directly to AXI port arbiter. The GPU traffic goes through a 16 deep command FIFO before the AXI port arbiter. Inside the DMC, there are different QOS control systems for CPU and GPU channels.

**Table 12-3 DMC AXI Port Mapping**

AXI0	BUS MATRIX 0	A73, A53, GPU
AXI2	BUS MATRIX 5	3 VPU

AXI3	BUS MATRIX 1	4 NNA
AXI4	BUS MATRIX 2	DEVICE0, DEVICE1, USB, PCIe, HDMI Rx
AXI5	DMC_TEST	DMC internal
AXI6	BUS MATRIX 3	DOS, GE2d
AXI7	BUS MATRIX 4	DEVICE2, ISP, DEMUX, DSP

### 12.1.6 DMC AXI Port QoS Control

DMC uses ARQOS[3:0]/AWQOS[3:0] bit fields as the default QoS control input. Inside each AXI port interface, there are several control registers to give additional QoS control. The QoS value is used for AXI port arbitration to decide which AXI request to put in the DMC command buffer. The QoS value of each AXI transaction is also used for the DDR command filter to balance the DDR performance and DDR request latency. There are standard QoS control features for all AXI ports and additional special QoS control features for the CPU, GPU, VPU and ISP.

1. Fixed QoS: Forces a fixed QoS value for each AXI port instead of the AXI ARQOS/AWQOS bit field input.
2. Transaction Grouping: Each AXI port can be programmed with transaction grouping that controls how many AXI transactions can be granted together when port wins the arbiter. NOTE: Also called arbitration weight
3. Improve Wait Time: If an AXI port cannot win the arbitration because of higher priority QoS AXI requests, the QoS value can be programmed to increase after waiting a certain number of clock cycles.
4. CPU IRQ/FIQ QoS Control: DMC can set the CPU channel QoS value to a fixed value if the CPU is inside an IRQ or FIQ interrupt.
5. CPU Auto QoS Control: DMC can set the CPU channel QoS value to a fixed value if there is a VPU request or other request inside the DMC.
6. VPU/ISP Urgent Control: DMC can set the VPU/ISP channel QoS value to a fixed value if the urgent control signal is set.
7. VPU Sleep: DMC can use this signal to trigger LPDDR4 PHY retraining.
8. AXI port status registers

### 12.1.7 DMC TEST Module

DMC test modules features:

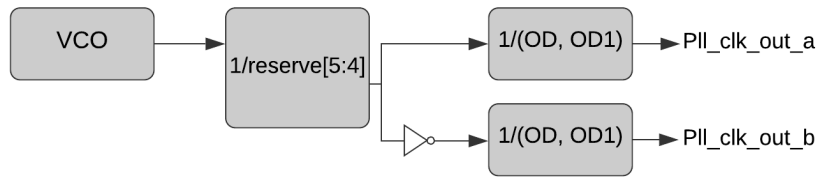
- Increase test data pattern size to 128bursts (32x128).
- Error counters.
- Meets DRAM validation hooks requirements.
- Address use AXI linear address.
- Must use security control to identify the address space to be used.

### 12.1.8 Clock, Refresh, Retraining and Frequency Change for 2 DMCs

- `dmc_clk_freq` generates 2 DMC clocks with the same frequency but phase differs and controlled by different PLL control registers.
- Timer inside `dmc_clk_freq` to trigger both DMCs for auto refresh/retraining/frequency change at the same time.



**Figure 12-4 Changes for 2 DMCs**



### 12.1.9 Register Description

The two DMC modules, including the registers, are exactly the same.

**Table 12-4 DMC Registers Base Address**

Registers Domain	Register Base Address	Range
DMC0	0xfe036000	8KB
DMC1	0xfe034000	8KB
DDRPHY0	0xfc000000	16MB
DDRPHY1	0xfb000000	16MB
DMC_CLK_FREQ	0xfe0a0000	8KB

DMC\_CLK\_FREQ registers address = Base Address + offset.

**Table 12-5 AM\_DDR\_PLL\_CNTL0 0x0000**

Bit(s)	R/W	Default	Description
31	Read only	0	dpll_lock. 1: lock. 0: unlock.
30	R/W	0	Not used.
29	R/W	1	dpll_reset.
28	R/W	0	dpll_en.
27:26	R/W	0	dpll_clk_en. 2'b10: pll_clock output enable. 4xclk output disable.. 2'b11, pll_clock and 4xclk output enable.
25	R/W	0	dpll_inv_sel
21:19	R/W	0	od1.
18:16	R/W	0	OD.
14:10	R/W	0	dpll_ref_div_n
8:0	R/W	0	dpll_int_num

**Table 12-6 AM\_DDR\_PLL\_CNTL1 0x0001**

Bit(s)	R/W	Default	Description
18:0	R/W	0	ddr_dpll_frac

**Table 12-7 AM\_DDR\_PLL\_CNTL2 0x0002**

Bit(s)	R/W	Default	Description
22:20	R/W	0	fref_sel
17:16	R/W	0	os_ssc
15:12	R/W	0	ssc_str_m
8	R/W	0	ssc_en
7:4	R/W	0	ssc_dep_sel
1:0	R/W	0	dpll ss_mode.

**Table 12-8 AM\_DDR\_PLL\_CNTL3 0x0003**

Bit(s)	R/W	Default	Description
31	R/W	0	afc bypass
30	R/W	0	afc clk sel
29	R/W	0	code new
28	R/W	0	dco_m_en
27	R/W	0	dco_sdm_en
26	R/W	0	div2
25	R/W	0	div mode
24	R/W	0	fast_lock mode
23	R/W	0	fb_pre_div
22	R/W	0	filter_mode
21	R/W	0	fix_en
20	R/W	0	freq_shift_en
19	R/W	0	load
18	R/W	0	load_en
17	R/W	0	lock_f
16	R/W	0	pulse_width_en
15	R/W	0	sdmnc_en
14	R/W	0	sdmnc_mode
13	R/W	0	sdmnc_range
12	R/W	0	tdc_en
11	R/W	0	tdc_mode_sel
10	R/W	0	Ddr_dpll_wait_en
9:0	R/W	0	Not used.

**Table 12-9 AM\_DDR\_PLL\_CNTL4 0x0004**

Bit(s)	R/W	Default	Description
1:0	R/W	0	pf_d_gain
7:4	R/W	0	filter_pvt1
11:8	R/W	0	filter pvt2
13:12	R/W	0	acq_gain
18:16	R/W	0	lambda0
22:20	R/W	0	lambda1
26:24	R/W	0	rou
30:28	R/W	0	alpha

**Table 12-10 AM\_DDR\_PLL\_CNTL5 0x0005**

Bit(s)	R/W	Default	Description
15:0	R/W	0	reve
21:16	R/W	0	lm_s
27:24	R/W	0	lm_w
30:28	R/W	0	adj_vco_ldo

**Table 12-11 AM\_DDR\_PLL\_CNTL6 0x0006**

Bit(s)	R/W	Default	Description
30:28	R/W	0	adj_vco_ldo
27:24	R/W	0	lm_w
21:16	R/W	0	lm_s
15:6	R/W	0	Reve[15:6]
5:4	R/W	0	reve[5:4] for chan_A, chan_B phase control. if freq =1, it would replaced by AM_DDR_FREQ_CTRL bit[11:10];
3:0	R/W	0	Reve[3:0]

**Table 12-12 AM\_DDR\_PLL\_STS 0x0007**

Bit(s)	R/W	Default	Description
31	Read only	0	DDR_PLL_LOCK
30	Read only	0	Lock_a
29	Read only	0	DDR_AFC_DONE

Bit(s)	R/W	Default	Description
22:16	Read only	0	sdmnc_monitor
9:0	Read only		out_rsv

**Table 12-13 DDR\_CLK\_CNTL 0x0008**

Bit(s)	R/W	Default	Description
31	R/W	0	Channel 0 ddr_pll_clk enable. enable the clock from DDR_PLL to clock generation.
30.	R/W	0	Channel 0 ddr_pll_prod_test_en. enable the clock to clock/32 which to clock frequency measurement and production test pin.
29.	R/W	0	Not used
28.	R/W	0	clock generation logic soft reset. 0 = reset.
27.	R/W	0	channel 0 phy_4xclk phase inverter..
25	R/W	0	channel 0 DDRPHY DfiClk/DMC clock selection. 1: AM_PLL clk output /2. 0: directly output from AM_PLL .
24	R/W	0	channel 0 enable AM_PLL CLK output /2 function. 1: enable. 0: disable. if try to use this clkoutput/2 function.
23.	R/W	0	Channel 1 ddr_pll_clk enable. enable the clock from DDR_PLL to clock generation.
22	R/W	0	Channel 1 ddr_pll_prod_test_en. enable the clock to clock/32 which to clock frequency measurement and production test pin.
21:20	R/W	0	Not used.
19	R/W	0	channel 1 phy_4xclk phase inverter.
17	R/W	0	Channel 1 DDRPHY DfiClk/DfiCtiClk/DMC clock selection. 1: AM_PLL clk output /2. 0: directly output from AM_PLL
16	R/W	0	Channel 1 AM_PLL CLK output /2 function. 1: enable. 0: disable. if try to use this clkoutput/2 function.
12	R/W	1	Channel 1 def_clk_sel 1: select reference clock as LPDDR4-PHY clock. 0: normal.
11	R/W	1	Channel 1 default clock enable. enable PCLK as LPDDR4_PHY clock. 0: disable.
10	R/W	0	enable ddr channel 1 dmc_clk.
9	R/W	1	enable LPDDR4-PHY channel 1 DfiClk.
8	R/W	1	enable LPDDR4-PHY channel 1 DfiCtiClk.
4	R/W	1	ddr0 def_clk_sel 1: select reference clock as LPDDR4-PHY clock. 0: normal.
3	R/W	1	ddr0 default clock enable. enable PCLK as LPDDR4_PHY clock. 0: disable.
2.	R/W	0	enable ddr channel 0 dmc_clk.
1.	R/W	1	enable LPDDR4-PHY channel 0 DfiClk.
0.	R/W	1	enable LPDDR4-PHY channel 0 DfiCtiClk.

LPDDR4 power on reset need to special combination of PwrOkIn and phy\_reset\_n. Please check the PHY PUB data book for detail.

**Table 12-14 DDR\_PHY\_CTRL 0x0009**

Bit(s)	R/W	Default	Description
31:8	R/W	0	Not used.
7	R/W	0	use dmc_clk_freq to control DDR-PHY channel 1 refresh/retraining/frequency change/phy power etc.
6	R/W	0	use dmc_clk_freq to control DDR-PHY channel 0 refresh/retraining/frequency change/phy power etc
5	R/W	0	DDR_PHY 1 PwrOkIn pin.
4	R/W	0	DDR_PHY 0 PwrOkIn pin.
3	R/W	0	DDR_PHY 1 APB soft reset_n.
2	R/W	0	DDR_PHY 1 phy_reset_n.
1.	R/W	0	DDR_PHY 0 APB soft reset_n.
0.	R/W	0	DDR_PHY 0 phy_reset_n.

**Table 12-15 AM\_DDR\_FREQ\_CTRL 0x000c**

Bit(s)	R/W	Default	Description
31	R/W	0	write trigger the DDR frequency change procedure. read =0 the frequency change done.
30	R/W	0	current FREQ selection. it can forced to change to select which frequency to select, or it can auto changed by FREQ change hardware.
29	R/W	0	next freq for frequency change.
12	R/W	0	ddr_dpll_inv_sel in frequency1 for 4xclk inverter.
11:10	R/W	0	ddr_dpll_clk_en in frequency1 for 4xclk and clock output.
9:8	R/W	0	pll_reve[5:4] in frequency1.
6:4	R/W	0	OD1 number in frequency 1.
2:0	R/W	0	OD number in frequency 1.

Frequency 0 auto refresh timing.

**Table 12-16 AM\_DDR\_TIMING\_F0 0x000d**

Bit(s)	R/W	Default	Description
15:8	R/W	0	refresh period in 100ns.
7:0	R/W	0	T100ns. how many clock cycle for 100ns.

Frequency 1 auto refresh timing.

**Table 12-17 AM\_DDR\_TIMING\_F1 0x000e**

Bit(s)	R/W	Default	Description
15:8	R/W	0	refresh period in 100ns.
7:0	R/W	0	T100ns. how many clock cycle for 100ns.

**Table 12-18 AM\_DDR\_TIMING\_CFG 0x000f**

Bit(s)	R/W	Default	Description
31	R/W	0	cfg ddr timing. write 1 to save the timing parameters to ddr clock domain.
30	R/W	0	dmc_clk_freq control the LPDDR4 retraining.
39	R/W	0	dmc_clk_freq control the auto refresh timing.
28:24	R/W	0	Not used.
23:0	R/W	0	lpddr4 phy retraining timer counter in 100ns.

Register address= dmc0/1\_register\_base\_addr + ( offset << 2).

**Table 12-19 DMC\_REQ\_CTRL 0x0000**

Bit(s)	R/W	Default	Description
31:8	R/W	0	Not used.
7.	R/W	0	Enable axi port 7
6.	R/W	0	Enable axi port 6
5.	R/W	0	Not used.
4.	R/W	0	Enable axi port 4.
3.	R/W	0	Enable axi port 3
2.	R/W	0	enable axi port 2
1.	R/W	0	Not used.
0.	R/W	0	Enable axi port0.

**Table 12-20 DMC\_SOFT\_RST 0x0001**

Bit(s)	R/W	Default	Description
31~ 8	R/W	0	reserved for future.
7:0.	R/W	0	Reset_n for AXI ports dmc clock domain interface. One bit for one AXI port . 0: reset. 1: normal working mode.

**Table 12-21 DMC\_SOFT\_RST1 0x0002**

Bit(s)	R/W	Default	Description
31~ 1	R/W	0	not used.
0.	R/W	0	AXI0 master clock domain reset_n signal. 0: reset. 1: normal working mode.

**Table 12-22 DMC\_SOFT\_RST2 0x0003**

Bit(s)	R/W	Default	Description
31~ 11.	R/W	0	reserved for future.
10	R/W	0	DMC DFI cmd soft reset_n. 0: reset. 1: normal working mode.
9	R/W	0	DMC DFI MISC soft reset_n. 0: reset. 1: normal working mode.
8	R/W	0	DMC DFI data soft reset_n. 0: reset. 1: normal working mode.
7	R/W	0	DMC DFI dcu soft reset_n. 0: reset. 1: normal working mode.
6	R/W	0	DMC siu soft reset_n. 0: reset. 1: normal working mode.
5.	R/W	0	DMC test soft reset_n.0: reset. 1: normal working mode.
4.	R/W	0	DMC low power control moudle soft reset_n. 0: reset. 1: normal working mode.
3.	R/W	0	DMC QoS monitor module soft reset_n.0: reset. 1: normal working mode.
2.	R/W	0	DMC register modle soft reset_n. 0: reset. 1: normal working mode.
1.	R/W	0	Not used.
0.	R/W	0	DMC command buffers and command generation modules soft reset_n.0 = re- set. 1: normal working mode.

**Table 12-23 DMC\_RST\_STS1 0x0004**

Bit(s)	R/W	Default	Description
31~1.	Read only	0	not used.
0.	Read only	0	RO. the DMC_SOFT_RST1 signal in n_clk domain. the purpose of this register is when one of the 2 clocks is too slow or too fast, we can read this regis- ter to make sure another clock domain reset is done.

**Table 12-24 DMC\_CLKG\_CTRL0 0x0006**

Bit(s)	R/W	Default	Description
31:8	R/W	0xffffffff	Not used.
7:0	R/W	0xff	enable the 8 axi interfaces both main and n_clk auto clock gating function. each 1 bit for one interface.

**Table 12-25 DMC\_CLKG\_CTRL1 0x0007**

Bit(s)	R/W	Default	Description
31:8	R/W	0	Not used.
7:0	R/W	0	force to disable the 8 axi interfaces both main and n_clk. each 1 bit for one interface.

**Table 12-26 DMC\_CLKG\_CTRL2 0x0008**

Bit(s)	R/W	Default	Description
12	R/W	1	enable auto clock gating for awcmdfifo.
11	R/W	1	enable auto clock gating for arcmdfifo.
10	R/W	1	enable auto clock gating for dfi command generation
9	R/W	1	enable auto clock gating for dram controller
8	R/W	1	enable auto clock gating for dfi data path.
7.	R/W	1	enable auto clock gating for write rsp generation.
6.	R/W	1	enable auto clock gating for read rsp generation.
5.	R/W	1	enable auto clock gating for ddr0 command filter.
4.	R/W	1	enable auto clock gating for ddr0 write reorder buffer.
3.	R/W	1	enable auto clock gating for ddr0 write data buffer.
2.	R/W	1	enable auto clock gating for ddr0 read reorder buffer.
1.	R/W		Not used
0.	R/W		Not used.

**Table 12-27 DMC\_CLKG\_CTRL3 0x0009**

Bit(s)	R/W	Default	Description
12	R/W	0	force to disable the clock of awcmdfifo.
11	R/W	0	force to disable the clock of arcmdfifo.
10	R/W	0	force to disable the clock of dfi command generation
9	R/W	0	force to disable the clock of dram controller
8	R/W	0	force to disable the clock of dfi data path.
7.	R/W	0	force to disable the clock of write rsp generation.
6.	R/W	0	force to disable the clock of read rsp generation.
5.	R/W	0	force to disable the clock of command filter.
4.	R/W	0	force to disable the clock of write reorder buffer.
3.	R/W	0	force to disable the clock of write data buffer.
2.	R/W	0	force to disable the clock of read reorder buffer.
1.	R/W	0	Not used.
0.	R/W	0	Not used.

**Table 12-28 DMC\_MON\_CTRL0 0x0010**

Bit(s)	R/W	Default	Description
31.	R/W	0	QoS_mon_en. write 1 to trigger the enable. polling this bit 0, means finished.or use interrupt to check finish.
30.	R/W	0	QoS_mon interrupt clear. clear the QoS monitor result.read 1 = QoS mon finish interrupt.



**Table 12-29 DMC\_MON\_TIMER 0x0011**

Bit(s)	R/W	Default	Description
31:0	R/W	0	timer for the monitor period.

**Table 12-30 DMC\_MON\_ALL\_IDLE\_CNT 0x0012**

Bit(s)	R/W	Default	Description
31:0	Read only	0	at the test period, the whole MMC all channel IDLE time. unit, dmc clock.

**Table 12-31 DMC\_MON\_ALL\_BW 0x0013**

Bit(s)	R/W	Default	Description
31:0	Read only	0	at the test period, the whole MMC granted data cycles. units one data clock cycle = 16bytes.

**Table 12-32 DMC\_MON\_ALL\_BW 0x0014**

Bit(s)	R/W	Default	Description
31:0	Read only	0	at the test period, the whole MMC granted data cycles which goes to 16bits ddr. units one data clock cycle = 16byte.

**Table 12-33 DMC\_MON0\_STA 0x0020**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon0 start address unit 4KB.

**Table 12-34 DMC\_MON0\_EDA 0x0021**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon0 end address unit 4KB.

**Table 12-35 DMC\_MON0\_CTRL 0x0022**

Bit(s)	R/W	Default	Description
31.	R/W	0	bandwidth monitor 0 enable.
30.	R/W	0	check write bandwidth 1: check. 0 not check.
29	R/W	0	check read bandwidth 1: check. 0: not check.
28:27	R/W	0	not used.
26	R/W	0	ld_num 2 enable.
25	R/W	0	ld_num 1 enable.

Bit(s)	R/W	Default	Description
24	R/W	0	Id_num 0 enable.
23:16	R/W	0	master ID number 2.
15:8	R/W	0	master ID number 1.
7:0	R/W	0	master ID number 0.

**Table 12-36 DMC\_MON0\_BW 0x0023**

Bit(s)	R/W	Default	Description
31:0	Read only	0	Monitor0 bandwidth report. read only. unit: 1 clock cycle data transfer = 16byte.

**Table 12-37 DMC\_MON1\_STA 0x0024**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon0 start address unit 4KB.

**Table 12-38 DMC\_MON1\_EDA 0x0025**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon1 end address unit 4KB.

**Table 12-39 DMC\_MON1\_CTRL 0x0026**

Bit(s)	R/W	Default	Description
31.	R/W	0	bandwidth monitor 1 enable.
30.	R/W	0	check write bandwidth 1: check. 0 not check.
29	R/W	0	check read bandwidth 1: check. 0: not check.
28:27	R/W	0	not used.
26	R/W	0	Id_num 2 enable.
25	R/W	0	Id_num 1 enable.
24	R/W	0	Id_num 0 enable.
23:16	R/W	0	master ID number 2.
15:8	R/W	0	master ID number 1.
7:0	Read only	0	master ID number 0.

**Table 12-40 DMC\_MON1\_BW 0x0027**

Bit(s)	R/W	Default	Description
31:0	RO		Monitor1 bandwidth report. read only. unit: 1 clock cycle data transfer = 16byte.

**Table 12-41 DMC\_MON2\_STA 0x0028**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon2 start address unit 4KB.

**Table 12-42 DMC\_MON2\_EDA 0x0029**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon2 end address unit 4KB.

**Table 12-43 DMC\_MON2\_CTRL 0x002a**

Bit(s)	R/W	Default	Description
31.	R/W	0	bandwidth monitor 2 enable.
30.	R/W	0	check write bandwidth 1: check. 0 not check.
29	R/W	0	check read bandwidth 1: check. 0: not check.
28:27	R/W	0	not used.
26	R/W	0	ld_num 2 enable.
25	R/W	0	ld_num 1 enable.
24	R/W	0	ld_num 0 enable.
23:16	R/W	0	master ID number 2.
15:8	R/W	0	master ID number 1.
7:0	R/W	0	master ID number 0.

**Table 12-44 DMC\_MON2\_BW 0x002b**

Bit(s)	R/W	Default	Description
31:0	Read only	0	Monitor 2 bandwidth report. read only. unit: 1 clock cycle data transfer = 16byte.

**Table 12-45 DMC\_MON3\_STA 0x002c**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon3 start address unit 4KB.

**Table 12-46 DMC\_MON3\_EDA 0x002d**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon3 end address unit 4KB.

**Table 12-47 DMC\_MON3\_CTRL 0x002e**

Bit(s)	R/W	Default	Description
31.	R/W	0	bandwidth monitor 3 enable.
30.	R/W	0	check write bandwidth 1: check. 0 not check.
29	R/W	0	check read bandwidth 1: check. 0: not check.
28:27	R/W	0	not used.
26	R/W	0	ld_num 2 enable.
25	R/W	0	ld_num 1 enable.
24	R/W	0	ld_num 0 enable.
23:16	R/W	0	master ID number 2.
15:8	R/W	0	master ID number 1.
7:0	R/W	0	master ID number 0.

**Table 12-48 DMC\_MON3\_BW 0x002f**

Bit(s)	R/W	Default	Description
31:0	Read only	0	Monitor 3 bandwidth report. read only. unit: 1 clock cycle data transfer = 16byte.

**Table 12-49 DMC\_MON4\_STA 0x0030**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon4 start address unit 4KB.

**Table 12-50 DMC\_MON4\_EDA 0x0031**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon4 end address unit 4KB.

**Table 12-51 DMC\_MON4\_CTRL 0x0032**

Bit(s)	R/W	Default	Description
31.	R/W	0	bandwidth monitor 4 enable.
30.	R/W	0	check write bandwidth 1: check. 0 not check.
29	R/W	0	check read bandwidth 1: check. 0: not check.

Bit(s)	R/W	Default	Description
28:27	R/W	0	not used.
26	R/W	0	Id_num 2 enable.
25	R/W	0	Id_num 1 enable.
24	R/W	0	Id_num 0 enable.
23:16	R/W	0	master ID number 2.
15:8	R/W	0	master ID number 1.
7:0	R/W	0	master ID number 0.

**Table 12-52 DMC\_MON4\_BW 0x0033**

Bit(s)	R/W	Default	Description
31:0	Read only	0	Monitor 4 bandwidth report. read only. unit: 1 clock cycle data transfer = 16byte.

**Table 12-53 DMC\_MON5\_STA 0x0034**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon5 start address unit 4KB.

**Table 12-54 DMC\_MON5\_EDA 0x0035**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon5 end address unit 4KB.

**Table 12-55 DMC\_MON5\_CTRL 0x0036**

Bit(s)	R/W	Default	Description
31.	R/W	0	bandwidth monitor 5 enable.
30.	R/W	0	check write bandwidth 1: check. 0 not check.
29	R/W	0	check read bandwidth 1: check. 0: not check.
28:27	R/W	0	not used.
26	R/W	0	Id_num 2 enable.
25	R/W	0	Id_num 1 enable.
24	R/W	0	Id_num 0 enable.
23:16	R/W	0	master ID number 2.
15:8	R/W	0	master ID number 1.
7:0	R/W	0	master ID number 0.

**Table 12-56 DMC\_MON5\_BW 0x0037**

Bit(s)	R/W	Default	Description
31:0	Read only	0	Monitor 5 bandwidth report. read only. unit: 1 clock cycle data transfer = 16byte.

**Table 12-57 DMC\_MON6\_STA 0x0038**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon6 start address unit 4KB.

**Table 12-58 DMC\_MON6\_EDA 0x0039**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon6 end address unit 4KB.

**Table 12-59 DMC\_MON6\_CTRL 0x003a**

Bit(s)	R/W	Default	Description
31.	R/W	0	bandwidth monitor 6 enable.
30.	R/W	0	check write bandwidth 1: check. 0 not check.
29	R/W	0	check read bandwidth 1: check. 0: not check.
28:27	R/W	0	not used.
26	R/W	0	ld_num 2 enable.
25	R/W	0	ld_num 1 enable.
24	R/W	0	ld_num 0 enable.
23:16	R/W	0	master ID number 2.
15:8	R/W	0	master ID number 1.
7:0	R/W	0	master ID number 0.

**Table 12-60 DMC\_MON6\_BW 0x003b**

Bit(s)	R/W	Default	Description
31:0	Read only	0	Monitor 6 bandwidth report. read only. unit: 1 clock cycle data transfer = 16byte.

**Table 12-61 DMC\_MON7\_STA 0x003c**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon7 start address unit 4KB.

**Table 12-62 DMC\_MON7\_EDA 0x003d**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:0.	R/W	0	Mon7 end address unit 4KB.

**Table 12-63 DMC\_MON7\_CTRL 0x003e**

Bit(s)	R/W	Default	Description
31.	R/W	0	bandwidth monitor 7 enable.
30.	R/W	0	check write bandwidth 1: check. 0 not check.
29	R/W	0	check read bandwidth 1: check. 0: not check.
28:27	R/W	0	not used.
26	R/W	0	ld_num 2 enable.
25	R/W	0	ld_num 1 enable.
24	R/W	0	ld_num 0 enable.
23:16	R/W	0	master ID number 2.
15:8	R/W	0	master ID number 1.
7:0	R/W	0	master ID number 0.

**Table 12-64 DMC\_MON7\_BW 0x003f**

Bit(s)	R/W	Default	Description
31:0	Read only	0	Monitor 6 bandwidth report. read only. unit: 1 clock cycle data transfer = 16byte.

**Table 12-65 DMC\_CMD\_FILTER\_CTRL0 0x0040**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Not used.

**Table 12-66 DMC\_CMD\_FILTER\_CTRL2 0x0041**

Bit(s)	R/W	Default	Description
31:24.	R/W	0x50	keep the bank active if there's urgent level 3 read bank hit request.
23:16.	R/W	0x40	keep the bank active if there's urgent level 2 read bank hit request.
15:8.	R/W	0x30	keep the bank active if there's urgent level 1 read bank hit request.
7:0.	R/W	0x20	keep the bank active if there's urgent level 0 read bank hit request.

**Table 12-67 DMC\_CMD\_FILTER\_CTRL3 0x0042**

Bit(s)	R/W	Default	Description
31.	R/W	0	force wbuf empty.
30:26	R/W	28	wbuf high level number
25:21	R/W	16	wbuf midlevel number
20:16	R/W	12	wbuf low level number
14:10	R/W	20	rbuf high level number
9:5	R/W	12	rbuf middle level number
4:0	R/W	6	rbuf low level number

**Table 12-68 DMC\_CMD\_FILTER\_CTRL4 0x0043**

Bit(s)	R/W	Default	Description
29:25	R/W	8	tITW.long
24:20	R/W	4	tITW. short
19:12	R/W	30	tAP auto precharge the bank not used if idle that long time.
11:6	R/W	0x20	write to read accesses if there write hit request.
5:0	R/W	0x20	read to write accesses if there write hit request.

**Table 12-69 DMC\_CMD\_FILTER\_CTRL5 0x0044**

Bit(s)	R/W	Default	Description
31:24	R/W	0x3f	Once ddr data bus switch to read, the maximum read command number to give up the bus when there's write request pending for write buffer.
23:16	R/W	0x3f	Once ddr data bus switch to write, the maximum write command number to give up the bus when there's read request pending too long.
15:8	R/W	0x18	Once ddr data bus switch to read, the minimum read command number to transfer back to write stage if there's still pending read request.
7:0	R/W	0x18	Once ddr data bus switch to write, the minimum write command number to transfer back to read stage if there's still pending write request.

**Table 12-70 DMC\_CMD\_FILTER\_CTRL6 0x0045**

Bit(s)	R/W	Default	Description
31:24	R/W	8	write urgent 3 request pending hold num.
23:16	R/W	8	write urgent 2 request pending hold num.
15:8	R/W	8	write urgent 1 request pending hold num.
7:0	R/W	8	write urgent 0 request pending hold num.



**Table 12-71 DMC\_CMD\_FILTER\_CTRL7 0x0046**

Bit(s)	R/W	Default	Description
31:24	R/W	0x10	aw_req_pending signal assertion after wbuf full.
23:16	R/W	0x10	aw_req_pending signal hold how long if wbuf not full.
15:8	R/W	0x40	write to read waiting cycles if there write hit request.
7:0	R/W	0x40	read to write waiting cycles if there write hit request.

**Table 12-72 DMC\_CMD\_BUFFER\_CTRL8 0x0047**

Bit(s)	R/W	Default	Description
31:8	R/W	0	Reserved
7:0	R/W	0x20	rank limit to change to another rank.

**Table 12-73 DMC\_CMD\_BUFFER\_CTRL 0x0048**

Bit(s)	R/W	Default	Description
31:26	R/W	32	total write buffer number. default 32.
25:20	R/W	32	total read buffer number. default 32.
19:8	R/W	0	reserved.
7:0	R/W	0x30	aw_pending_inc_num. increase write urgent level 1 when write command waiting to in write buffer that long.

**Table 12-74 DMC\_CMD\_BUFFER\_CTRL1 0x0049**

Bit(s)	R/W	Default	Description
29:24	R/W	28	read buffer number in non-urgent request.
23:16	R/W	16	read buffer bank miss watch dog threshold.
15:12	R/W	12	read buffer urgent level 3 counter inc weight.
11:8	R/W	20	read buffer urgent level 2 counter inc weight.
7:4	R/W	12	read buffer urgent level 1 counter inc weight.
3:0	R/W	6	read buffer urgent level 0 counter inc weight.

**Table 12-75 DMC\_2ARB\_CTRL 0x004c**

Bit(s)	R/W	Default	Description
32:24.	R/W	0x80	Waiting limit to use the highest urgent level in the pipelines.
23:0	R/W	0	Not used.

**Table 12-76 DMC\_VERSION 0x004f**

Bit(s)	R/W	Default	Description
32:0	Read only	0x0100-0009	0x01000009

This AXI0 channel CTRL register is for CPU virtual channel QoS.

**Table 12-77 DMC\_AXI0\_CHAN\_CTRL 0x0080**

Bit(s)	R/W	Default	Description
31:30	R/W	0	Not used.
29:20	R/W	0x3f	write request pending cycle number to increment QoS level if not granted
19:16	R/W	0	Fixed QoS value
13:4	R/W	0x3f	read request pending cycle number to increment QoS level if not granted.
3:0	R/W	4	arbiter weight

This AXI0 hold ctrl is for AXI0(BUS MATRIX 0) interface outstanding traffic control.

**Table 12-78 DMC\_AXI0\_HOLD\_CTRL 0x0081**

Bit(s)	R/W	Default	Description
31:24	R/W	0x28	write hold num. max outstanding request number.
23:16	R/W	0x20	write hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.
15:8	R/W	0x28	read hold num. max outstanding request number.
7:0	R/W	0x20	read hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.

This AXI0 channel CTRL register is for CPU virtual channel QoS.

**Table 12-79 DMC\_AXI0\_CHAN\_CTRL1 0x0082**

Bit(s)	R/W	Default	Description
31:28	R/W	0xc	QoS value for FIQ QoS control enabled.
27:24	R/W	0xa	QoS value for IRQ QoS control enabled
23:20	R/W	0x4	AWQoS value for auto QoS control enable with VPU request.
19:16	R/W	0x6	AWQoS value for auto QoS control enable with other request.
15:12	R/W	0x7	AWQoS value for auto QoS control enable without any other request
11:8	R/W	0x4	ARQoS value for auto QoS control enable with VPU request.
7:4	R/W	0x6	ARQoS value for auto QoS control enable with other request.
3:0	R/W	0x7	ARQoS value for auto QoS control enable without any other request

This AXI0 channel CTRL register is for CPU virtual channel QoS.

**Table 12-80 DMC\_AXI0\_CHAN\_CTRL2 0x0083**

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used.
8	R/W	1	Not used.
7:4	R/W	4	Not used.
3	R/W	1	CPU FIQ QoS control enable.
2	R/W	1	CPU IRQ QoS control enable.
1	R/W	1	CPU read channel auto QoS enable (based on other traffic).
0	R/W	1	CPU write channel auto QoS enable(based on other traffic).

This AXI1 is for GPU virtual channel.

**Table 12-81 DMC\_AXI1\_CHAN\_CTRL 0x0084**

Bit(s)	R/W	Default	Description
31:30	R/W		Not used.
29:20	R/W	0x7f	write request pending cycle number to increment QoS level if not granted
19:16	R/W	0	Fixed QoS value
13:4	R/W	0x7f	read request pending cycle number to increment QoS level if not granted.
3:0	R/W	4	arbiter weight

This AXI1 is for GPU virtual channel.

**Table 12-82 DMC\_AXI1\_HOLD\_CTRL 0x0085**

Bit(s)	R/W	Default	Description
31:24	R/W	0x18	write hold num. max outstanding request number.
23:16	R/W	0x10	write hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.
15:8	R/W	0x18	read hold num. max outstanding request number.
7:0	R/W	0x10	read hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.

**Table 12-83 DMC\_AXI1\_CHAN\_CTRL2 0x0087**

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used.
8	R/W	1	Enable asymmetric QoS value if the transaction address is in the range of single/non-interleaved DMC transaction (asymmetric memory).
7:4	R/W	0	Fixed asymmetric QoS value if the transaction address is in the range of single/non-interleaved DMC transaction (asymmetric memory)
3:0	R/W	0	Not used.

This AXI2 channel CTRL register is for VPU channel

**Table 12-84 DMC\_AXI2\_CHAN\_CTRL 0x0088**

Bit(s)	R/W	Default	Description
31:30	R/W	0	Not used.
29:20	R/W	0x3f	write request pending cycle number to increment QoS level if not granted
19:16	R/W	0	Fixed QoS value
13:4	R/W	0x3f	read request pending cycle number to increment QoS level if not granted.
3:0	R/W	0x8	arbiter weight

This AXI2 channel CTRL register is for VPU channel.

**Table 12-85 DMC\_AXI2\_HOLD\_CTRL 0x0089**

Bit(s)	R/W	Default	Description
31:24	R/W	0x28	write hold num. max outstanding request number.
23:16	R/W	0x20	write hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.
15:8	R/W	0x28	read hold num. max outstanding request number.
7:0	R/W	0x20	read hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.

This AXI2 channel CTRL register is for VPU channel.

**Table 12-86 DMC\_AXI2\_CHAN\_CTRL1 0x008a**

Bit(s)	R/W	Default	Description
31:20	R/W		Not used
19	R/W	1	Enable sideband write urgent signal to force fixed AWQOS. 1: enable. 0: disable.
18	R/W	1	Enable sideband read urgent signal to force fixed ARQOS. 1: enable. 0: disable.
17	R/W	0	Enable sideband write urgent signal to block other masters 1: enable 0: disable.
16	R/W	0	Enable sideband read urgent signal to block other masters 1: enable 0: disable.
15:12	R/W	0xf	Fixed AWQOS value when sideband write urgent signal = 1 and bit 19 enabled.
11:8	R/W	0xf	Fixed ARQOS value when sideband read urgent signal = 1 and bit 18 enabled.
7:0	R/W	0	AXI masters to block when sideband read/write urgent signal = 1 and bit 16/17 enabled NOTE: Do not block ISP, VPU, CPU or other urgent request.

This AXI3 channel CTRL register is for NNA channel

**Table 12-87 DMC\_AXI3\_CHAN\_CTRL 0x008c**

Bit(s)	R/W	Default	Description
31:30	R/W		Not used.
29:20	R/W	0x7f	write request pending cycle number to increment QoS level if not granted
19:16	R/W	0	Fixed QoS value
13:4	R/W	0x7f	read request pending cycle number to increment QoS level if not granted.
3:0	R/W	4	arbiter weight

This AXI3 channel CTRL register is for NNA channel.

**Table 12-88 DMC\_AXI3\_HOLD\_CTRL 0x008d**

Bit(s)	R/W	Default	Description
31:24	R/W	0x18	write hold num. max outstanding request number.
23:16	R/W	0x10	write hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.
15:8	R/W	0x18	read hold num. max outstanding request number.
7:0	R/W	0x10	read hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.

This AXI4 channel CTRL register is for device channel

**Table 12-89 DMC\_AXI4\_CHAN\_CTRL 0x0090**

Bit(s)	R/W	Default	Description
31:30	R/W	0	Not used.
29:20	R/W	0x7f	write request pending cycle number to increment QoS level if not granted
19:16	R/W	0	Fixed QoS value
13:4	R/W	0x7f	read request pending cycle number to increment QoS level if not granted.
3:0	R/W	4	arbiter weight

This AXI4 channel CTRL register is for device channel.

**Table 12-90 DMC\_AXI4\_HOLD\_CTRL 0x0091**

Bit(s)	R/W	Default	Description
31:24	R/W	0x18	write hold num. max outstanding request number.
23:16	R/W	0x10	write hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.
15:8	R/W	0x18	read hold num. max outstanding request number.
7:0	R/W	0x10	read hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.

This AXI5 channel CTRL register is for dmc test channel

**Table 12-91 DMC\_AXI5\_CHAN\_CTRL 0x0094**

Bit(s)	R/W	Default	Description
31:20	R/W	0xff	Not used.
19:16	R/W	0	ARQoS/AWQoS for internal dmc_test.
13:4	R/W	0xff	Not used.
3:0	R/W	4	arbiter weight

This AXI5 channel CTRL register is for dmc test channel.

**Table 12-92 DMC\_AXI5\_HOLD\_CTRL 0x0095**

Bit(s)	R/W	Default	Description
31:24	R/W	7	write hold num. max outstanding request number.
23:16	R/W	4	write hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.
15:8	R/W	7	read hold num. max outstanding request number.
7:0	R/W	4	read hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.

This AXI6 channel CTRL register is for device channel

**Table 12-93 DMC\_AXI6\_CHAN\_CTRL 0x0098**

Bit(s)	R/W	Default	Description
31:30	R/W	0	Not used.
29:20	R/W	0x7f	write request pending cycle number to increment QoS level if not granted
19:16	R/W	0	Fixed QoS value
13:4	R/W	0x7f	read request pending cycle number to increment QoS level if not granted.
3:0	R/W	4	arbiter weight

This AXI6 channel CTRL register is for device channel.

**Table 12-94 DMC\_AXI6\_HOLD\_CTRL 0x0099**

Bit(s)	R/W	Default	Description
31:24	R/W	0x18	write hold num. max outstanding request number.
23:16	R/W	0x10	write hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.

Bit(s)	R/W	Default	Description
15:8	R/W	0x18	read hold num. max outstanding request number.
7:0	R/W	0x10	read hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.

This AXI7 channel CTRL register is for ISP/demux/other device channel

**Table 12-95 DMC\_AXI7\_CHAN\_CTRL 0x009c**

Bit(s)	R/W	Default	Description
31:30	R/W	0	Not used.
29:20	R/W	0x7f	write request pending cycle number to increment QoS level if not granted
19:16	R/W	0	Fixed QoS value
13:4	R/W	0x7f	read request pending cycle number to increment QoS level if not granted.
3:0	R/W	0	arbiter weight

This AXI7 channel CTRL register is for ISP/demux/other device channel.

**Table 12-96 DMC\_AXI7\_HOLD\_CTRL 0x009d**

Bit(s)	R/W	Default	Description
31:24	R/W	0x1f	write hold num. max outstanding request number.
23:16	R/W	0x10	write hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.
15:8	R/W	0x18	read hold num. max outstanding request number.
7:0	R/W	0x10	read hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.

This AXI7 channel CTRL register is for ISP/demux/other device channel.

**Table 12-97 DMC\_AXI7\_CHAN\_CTRL1 0x009e**

Bit(s)	R/W	Default	Description
31:20	R/W		Not used
19	R/W	1	Enable sideband write urgent signal to control AWQOS. 1: enable. 0: disable.
18	R/W	1	Enable sideband read urgent signal to control ARQOS. 1: enable. 0: disable.
17	R/W	0	Enable sideband write urgent signal to block other master request. 1: enable. 0 disable.
16	R/W	0	Enable sideband read urgent signal to block other master request. 1: enable. 0 disable.
15:12	R/W	0xf	AWQOS value when sideband write urgent signal = 1 and bit 19 enabled..

Bit(s)	R/W	Default	Description
11:8	R/W	0xf	ARQOS value when sideband read urgent signal = 1 and bit 18 enabled.
7:0	R/W	0	AXI masters to block when sideband read/write urgent signal = 1 and bit 16/17 enabled NOTE: Do not block ISP, VPU, CPU or other urgent request.

AXI0 Async interface status.

**Table 12-98 DMC\_AXI0\_CHAN\_STS 0x00a0**

Bit(s)	R/W	Default	Description
31:24	Read only	0	read command pending transaction number.
23:16	Read only	0	write command pending transaction number.
15:12	Read only	0	FIQ STS
11:8	Read only	0	IRQ STS
7:0	Read only	0	read back data waiting counter(RVALID high with RREADY low max time counter.

Mali virtual channel status.

**Table 12-99 DMC\_AXI1\_CHAN\_STS 0x00a1**

Bit(s)	R/W	Default	Description
31:24	Read only	0	read command pending transaction number.
23:16	Read only	0	write command pending transaction number.
15:8	Read only	0	write data fifo counter.
7:0	Read only	0	Mali master side wddata valid for AWVALID counter..

**Table 12-100 DMC\_AXI2\_CHAN\_STS 0x00a2**

Bit(s)	R/W	Default	Description
31:24	Read only	0	read command pending transaction number.
23:16	Read only	0	write command pending transaction number.
15:8	Read only	0	AWVALID without WDATA transaction number.
7:0	Read only	0	read back data waiting counter(RVALID high with RREADY low max time counter)



**Table 12-101 DMC\_AXI3\_CHAN\_STS 0x00a3**

Bit(s)	R/W	Default	Description
31:24	Read only	0	read command pending transaction number.
23:16	Read only	0	write command pending transaction number.
15:8	Read only	0	AWVALID without WDATA transaction number.
7:0	Read only	0	read back data waiting counter (RVALID high with RREADY low max time counter)

**Table 12-102 DMC\_AXI4\_CHAN\_STS 0x00a4**

Bit(s)	R/W	Default	Description
31:24	Read only	0	read command pending transaction number.
23:16	Read only	0	write command pending transaction number.
15:8	Read only	0	AWVALID without WDATA transaction number.
7:0	Read only	0	read back data waiting counter (RVALID high with RREADY low max time counter)

**Table 12-103 DMC\_AXI5\_CHAN\_STS 0x00a5**

Bit(s)	R/W	Default	Description
31:24	Read only	0	read command pending transaction number.
23:16	Read only	0	write command pending transaction number.
15:8	Read only	0	AWVALID without WDATA transaction number.
7:0	Read only	0	Not used

**Table 12-104 DMC\_AXI6\_CHAN\_STS 0x00a6**

Bit(s)	R/W	Default	Description
31:24	Read only	0	read command pending transaction number.
23:16	Read only	0	write command pending transaction number.
15:8	Read only	0	AWVALID without WDATA transaction number.
7:0	Read only	0	read back data waiting counter (RVALID high with RREADY low max time counter)

**Table 12-105 DMC\_AXI7\_CHAN\_STS 0x00a7**

Bit(s)	R/W	Default	Description
31:24	Read only	0	read command pending transaction number.
23:16	Read only	0	write command pending transaction number.
15:8	Read only	0	AWVALID without WDATA transaction number.
7:0	Read only	0	read back data waiting counter (RVALID high with RREADY low max time counter)

CPU virtual channel outstanding traffic control.

**Table 12-106 DMC\_CPU\_HOLD\_CTRL 0x00a8**

Bit(s)	R/W	Default	Description
31:24	R/W	0x20	write hold num. max outstanding request number.
23:16	R/W	0x18	write hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.
15:8	R/W	0x20	read hold num. max outstanding request number.
7:0	R/W	0x18	read hold release num. if the outstanding request == hold num, then hold this request unless the outstanding request number below the hold release number, then continue to request.

**Table 12-107 DMC\_CHAN\_STS 0x00bc**

Bit(s)	R/W	Default	Description
27	Read only	1	always 1
26	Read only	1	write data buffer idle. 1: idle 0: busy.
25	Read only	1	always 1.
24	Read only	1	Write command buffer idle. 1: idle 0: busy.
23:10	Read only	0x3fff	Not used.
9	Read only	1	Mali virtual channel idle. 1: idle 0: busy.
8	Read only	1	CPU Virtual channel idle. 1: idle 0: busy.
7:0.	Read only	0xff	Axi interface channel idle. 1: idle 0: busy.

**Table 12-108 DMC\_IRQ\_STS 0x00db**

Bit(s)	R/W	Default	Description
31:1	R	0	Not used.
0	R/W	0	QoS Monitor interrupt flag. 1: means there's QoS monitor interrupt. write 1 to clean this interrupt.

**Table 12-109 DMC\_PROT0\_STA 0x00e0**

Bit(s)	R/W	Default	Description
31:20	R	0	Not used.
19:0	R/W	0	protection 0 start address . unit 4KB.

**Table 12-110 DMC\_PROT0\_EDA 0x00e1**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used.
19:0	R/W	0	protection 0 end address . unit 4KB.

**Table 12-111 DMC\_PROT0\_CTRL 0x00e2**

Bit(s)	R/W	Default	Description
31:28	R/W	0	Not used.
27	R/W	0	protection 0 write monitor enable.
26	R/W	0	protection 0 read monitor enable.
25:9	R/W	0	Not used.
8	R/W	0	monior policy: 1 include the defined IDs. 0: exclude the defined IDs.
7:4	R/W	0	Not used.
3	R/W	0	ID3 enable.
2	R/W	0	ID2 enable.
1	R/W	0	ID1 enable.
0	R/W	0	ID0 enable.

**Table 12-112 DMC\_PROT0\_CTRL1 0x00e3**

Bit(s)	R/W	Default	Description
31:24	R/W	0	ID3 id number.
23:16	R/W	0	ID2 id number.
15:8	R/W	0	ID1 id number.
7:0	R/W	0	ID0 id number.

**Table 12-113 DMC\_PROT1\_STA 0x00e4**

Bit(s)	R/W	Default	Description
31:20	R	0	Not used.
19:0	R/W	0	protection 1 start address . unit 4KB.

**Table 12-114 DMC\_PROT1\_EDA 0x00e5**

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used.
19:0	R/W	0	protection 1 end address . unit 4KB.

**Table 12-115 DMC\_PROT1\_CTRL 0x00e6**

Bit(s)	R/W	Default	Description
31:28	R/W	0	Not used.
27	R/W	0	protection 1 write monitor enable.
26	R/W	0	protection 1 read monitor enable.
25:9	R/W	0	Not used.
8	R/W	0	monitor policy: 1 include the defined IDs. 0: exclude the defined IDs.
7:4	R/W	0	Not used.
3	R/W	0	ID3 enable.
2	R/W	0	ID2 enable.
1	R/W	0	ID1 enable.
0	R/W	0	ID0 enable.

**Table 12-116 DMC\_PROT1\_CTRL1 0x00e7**

Bit(s)	R/W	Default	Description
31:24	R/W	0	ID3 id number.
23:16	R/W	0	ID2 id number.
15:8	R/W	0	ID1 id number.
7:0	R/W	0	ID0 id number.

**Table 12-117 DMC\_PROT\_VIO\_0 0x00e8**

Bit(s)	R/W	Default	Description
31:0	Read only	0	protection write violation low 32bits address. AWADDR[31:0].

**Table 12-118 DMC\_PROT\_VIO\_1 0x00e9**

Bit(s)	R/W	Default	Description
31:19	R/W	0	Not used.
18	R/W	0	protection 1 violation.
17	R/W	0	protection 0 violation.
16:15	R/W	0	AWADDR[33:32].
14:0	R/W	0	AWUSER for this violated transaction.

**Table 12-119 DMC\_PROT\_VIO\_2 0x00ea**

Bit(s)	R/W	Default	Description
31:0	Read only	0	protection read violation low 32bits address. ARADDR[31:0].

**Table 12-120 DMC\_PROT\_VIO\_3 0x00eb**

Bit(s)	R/W	Default	Description
31:19	Read only	0	Not used.
18	Read only	0	protection 1 violation.
17	Read only	0	protection 0 violation.
16:15	Read only	0	ARADDR[33:32].
14:0	Read only	0	ARUSER for this violated transaction.

**Table 12-121 DMC\_PROT\_IRQ\_CTRL\_STS 0x00ec**

Bit(s)	R/W	Default	Description
31:3	Read only	0	Not used.
2	R/W	0	protection IRQ enable
1	R/W	0	write protection violation. write 1 to clean the write protection status and interrupt.
0	R/W	0	read protection violation. write 1 to clean the read protection status and interrupt.

### DMC DDR Timing Parameter Registers

There're 2 sets of timing DDR timing parameters for 2 sets of DDR frequency parameters.

When change frequency, the hardware would automatically select one of these two sets of timing parameters: DMC\_DRAM\_\* is for Frequency set 0. DMC\_NFQ\_\* is for Frequency set 1.

**Table 12-122 DMC\_DRAM\_TMRD 0x0100**

Bit(s)	R/W	Default	Description
4:0	R/W	0	tMRD. //MR command cycles, in DDR3/4.in LPDDR4, it should be value of tMRW

**Table 12-123 DMC\_DRAM\_TRFC 0x0101**

Bit(s)	R/W	Default	Description
9:0	R/W	0	tRFC

**Table 12-124 DMC\_DRAM\_TRP 0x0102**

Bit(s)	R/W	Default	Description
21:16	R/W	0	tRP for precharge all banks.
5:0	R/W	0	tRP for precharge one bank.

**Table 12-125 DMC\_DRAM\_TRTW 0x0103**

Bit(s)	R/W	Default	Description
5:0	R/W	0	DDR3/4 mode: tRTW. For LPDDR4 . The total read command -> write command = (RL + BL/2 - tWOD-TON) + TDQSCK_MAX + tWPRE +RD(tRPST). tRTW = TDQSCK_max + tWPRE + tRD(tRPST) + (delay margin)

**Table 12-126 DMC\_DRAM\_TCL 0x0104**

Bit(s)	R/W	Default	Description
5:0	R/W	0	tCL/tRL. read latency.

**Table 12-127 DMC\_DRAM\_TCWL 0x0105**

Bit(s)	R/W	Default	Description
5:0	R/W	0	CWL write latency.

**Table 12-128 DMC\_DRAM\_TRAS 0x0106**

Bit(s)	R/W	Default	Description
7:0	R/W	0	tRAS.minimum active to precharge time for same bank.

**Table 12-129 DMC\_DRAM\_TRC 0x0107**

Bit(s)	R/W	Default	Description
7:0	R/W	0	tRC.minimum active to active time for same bank.

**Table 12-130 DMC\_DRAM\_TRCD 0x0108**

Bit(s)	R/W	Default	Description
7:0	R/W	0	tRCDactive to read/write timing for same bank.

**Table 12-131 DMC\_DRAM\_TRRD 0x0109**

Bit(s)	R/W	Default	Description
21:16	R/W	0	tRRD_lactive bank A to active B in same band group for DDR4.
5:0	R/W	0	tRRD/tRRD_sactive bank A to active bank b time. tRRD_s: active bank A to active bank b in different bank groups for DDR4.

**Table 12-132 DMC\_DRAM\_TFAW 0x010a**

Bit(s)	R/W	Default	Description
8:0	R/W	0	tFAW.four active command windows

**Table 12-133 DMC\_DRAM\_TRTP 0x010b**

Bit(s)	R/W	Default	Description
5:0	R/W	0	tRTP.

**Table 12-134 DMC\_DRAM\_TWR 0x010c**

Bit(s)	R/W	Default	Description
5:0			tWR.

**Table 12-135 DMC\_DRAM\_TWTR 0x010d**

Bit(s)	R/W	Default	Description
21:16	R/W	0	tWTR_l For DDR4 WTR_l. FOR DDR3/LPDDR4 same as tWTR_s.
5:0	R/W	0	tWTR.

**Table 12-136 DMC\_DRAM\_TCCD 0x010e**

Bit(s)	R/W	Default	Description
19:16	R/W	0	tCCD/tCCD_l.
3:0	R/W	0	tCCD/tCCD_s read to read command time or write to write command time.

**Table 12-137 DMC\_DRAM\_TEXSR 0x010f**

Bit(s)	R/W	Default	Description
12:0	R/W	0	tEXSR.EXIT SELF-REFRESH to read/write command.

**Table 12-138 DMC\_DRAM\_TXS 0x0110**

Bit(s)	R/W	Default	Description
9:0	R/W	0	tXS.EXIT SELF_REFRESH to other command time

**Table 12-139 DMC\_DRAM\_TXP 0x0111**

Bit(s)	R/W	Default	Description
3:0	R/W	0	tXP.EXIT power down to other command time

**Table 12-140 DMC\_DRAM\_TXPDLL 0x0112**

Bit(s)	R/W	Default	Description
12:0	R/W	0	tXPDLL,EXIT power down to read/write command time(need to relock PLL).

**Table 12-141 DMC\_DRAM\_TZQCS 0x0113**

Bit(s)	R/W	Default	Description
11:0	R/W	0	ZQCS command to other command time.

**Table 12-142 DMC\_DRAM\_TCKSRE 0x0114**

Bit(s)	R/W	Default	Description
4:0	R/W	0	enter self refresh to disable clock time.

**Table 12-143 DMC\_DRAM\_TCKSRX 0x0115**

Bit(s)	R/W	Default	Description
4:0	R/W	0	enable clock to exit self refresh time.

**Table 12-144 DMC\_DRAM\_TCKE 0x0116**

Bit(s)	R/W	Default	Description
4:0	R/W	0	CKE high or low minimum time.

**Table 12-145 DMC\_DRAM\_TMOD 0x0117**

Bit(s)	R/W	Default	Description
4:0	R/W	0	tMOD.MRR/MRW to other command time. in LPDDR4, still use this register but it called tMRD.



**Table 12-146 DMC\_DRAM\_TDQS 0x0018**

Bit(s)	R/W	Default	Description
31:28	R/W	0	reserved
27:24	R/W	0	tDQS. the delay for write after read in different rank.
23:20	R/W	0	reserved
19:16	R/W	0	tDQS. the delay for read after write in different rank.
15:12	R/W	0	reserved
11:8	R/W	0	tDQS. the delay for write after write in different rank.
7:4	R/W	0	reserved
3:0	R/W	0	tDQS. the delay for read after read in different rank.

**Table 12-147 DMC\_DRAM\_TZQLAT 0x011a**

Bit(s)	R/W	Default	Description
5:0	R/W	0	ZQ LATCH command to other command timing in LPDDR4 mode.

**Table 12-148 DMC\_DRAM\_TMRR 0x011b**

Bit(s)	R/W	Default	Description
7:0	R/W	0	tMRR not used in DMC.

**Table 12-149 DMC\_DRAM\_TCKESR 0x011c**

Bit(s)	R/W	Default	Description
9:0	R/W	0	tCKESR.CKE low minimum pulse in self refresh mode.

**Table 12-150 DMC\_DRAM\_DFITCTRLDELAY 0x011e**

Bit(s)	R/W	Default	Description
3:0	R/W	0	DFI_t_ctrldealy

**Table 12-151 DMC\_DRAM\_DFITPHYWRDATA 0x011f**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_t_phy_wrdata.

**Table 12-152 DMC\_DRAM\_DFITPHYWRLAT 0x0120**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_t_phy_wrlat. in DDR3/4/LPDDR3 mode: WL -5. in LPDDR4 mode: WL -5 + 2.

**Table 12-153 DMC\_DRAM\_DFITRDDATAEN 0x0121**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_t_rddata_en. in DDR3/4/LPDDR3 mode: RL -5. in LPDDR4 mode: RL -5 + 1.

**Table 12-154 DMC\_DRAM\_DFITPHYRDLAT 0x0122**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_t_rdlat.

**Table 12-155 DMC\_DRAM\_DFITCTRLUPDMIN 0x0123**

Bit(s)	R/W	Default	Description
7:0	R/W	0	CTRLUPD_MIN minimum clock cycle to maintain CTRLUPD_REQ.

**Table 12-156 DMC\_DRAM\_DFITCTRLUPDMAX 0x0124**

Bit(s)	R/W	Default	Description
7:0	R/W	0	CTRLUPD_MAX.maximum clock cycle to maintain CTRLUPD_REQ if no CTRLUPD_ACK response.

**Table 12-157 DMC\_DRAM\_DFITDRAMCLK 0x0128**

Bit(s)	R/W	Default	Description
17	R/W	0	dram clk1 enable.
16	R/W	0	dram clk0 enable.
15:8	R/W	0	DRAM CLK disable waiting time
7:0	R/W	0	DRAM CLK enable timer

**Table 12-158 DMC\_DRAM\_DFITLPRESP 0x012a**

Bit(s)	R/W	Default	Description
3:0	R/W	0	dfi_lp_ctrl_req response time. after dfi_lp_ctrl_req asserted, and after response time if there's still no dfi_lp_ack response, then drop the dfi_lp_ctrl_req.

**Table 12-159 DMC\_DRAM\_TCKECK 0x012c**

Bit(s)	R/W	Default	Description
4:0	R/W	0	tCKECKfrom CKE low to assert dfi_dram_clk_disable time. this time + dfi_t_ctrl_delay

**Table 12-160 DMC\_DRAM\_TREFI 0x012d**

Bit(s)	R/W	Default	Description
31:24	R/W	0	tZQCI dmc send zqci period. unit is how much auto refresh period.
23:16	R/W	0	pvtidmc send dfi_ctrlupd_req period.unit is one auto refresh period.
15:8	R/W	0	tREFI.dmc send auto refresh command period. unit is 100ns.
7:0	R/W	0	t100ns period. unit is dmc clock cycles

**Table 12-161 DMC\_DRAM\_TSR 0x012e**

Bit(s)	R/W	Default	Description
5:0	R/W	0	tSR.self refresh enter to exit time.

**Table 12-162 DMC\_DRAM\_TCCDMW 0x012f**

Bit(s)	R/W	Default	Description
5:0	R/W	0	4*tCCD in LPDDR4 mask write.

**Table 12-163 DMC\_DRAM\_TESCKE 0x0130**

Bit(s)	R/W	Default	Description
5:0	R/W	0	tESCKE.enter self refresh to power time for LPDDR4.

**Table 12-164 DMC\_DRAM\_TREFI\_DDR3 0x0131**

Bit(s)	R/W	Default	Description
7:0.	R/W	0	DDR3 SDRAM tREFI single auto refresh time . the unit is t100ns.

**Table 12-165 DMC\_DRAM\_TZQCAL 0x0132**

Bit(s)	R/W	Default	Description
11:0.	R/W	0	ZQCAL for LPDDR4. ZQINIT/ZQoper for DDR3/4 ZQCL command.

**Table 12-166 DMC\_DRAM\_T10US 0x0133**

Bit(s)	R/W	Default	Description
11:0.	R/W	0	10us clock cycle number used for LP2 mode.

**Table 12-167 DMC\_DRAM\_TMRRI 0x0134**

Bit(s)	R/W	Default	Description
7:0	R/W	0	tMRRI for MRR

**Table 12-168 DMC\_DRAM\_TXS\_FAST 0x0135**

Bit(s)	R/W	Default	Description
9:0	R/W	0	DDR4 mode XS_FAST exit self_refrest to zqcl/zqcs/mrs command.

**Table 12-169 DMC\_DRAM\_DFIODTCFG 0x0136**

Bit(s)	R/W	Default	Description
12.	R/W	0	rank1 ODT default. default value for ODT[1] pins if there's no read/write activity.
11.	R/W	0	rank1 ODT write sel.enable ODT[1] if there's write occur in rank1.
10.	R/W	0	rank1 ODT write nsel. enable ODT[1] if there's write occur in rank0.
9.	R/W	0	rank1 odt read sel.enable ODT[1] if there's read occur in rank1.
8.	R/W	0	rank1 odt read nsel.enable ODT[1] if there's read occur in rank0.
4.	R/W	0	rank0 ODT default. default value for ODT[0] pins if there's no read/write activity.
3.	R/W	0	rank0 ODT write sel. enable ODT[0] if there's write occur in rank0.
2.	R/W	0	rank0 ODT write nsel. enable ODT[0] if there's write occur in rank1.
1.	R/W	0	rank0 odt read sel. enable ODT[0] if there's read occur in rank0.
0.	R/W	0	rank0 odt read nsel. enable ODT[0] if there's read occur in rank1.

**Table 12-170 DMC\_DRAM\_DFIODTCFG1 0x0137**

Bit(s)	R/W	Default	Description
27:24	R/W	0	ODT length for BL8 read transfer.
19:16	R/W	0	ODT length for BL8 write transfer.
12:8	R/W	0	ODT latency for reads. suppose to be 0.
4:0	R/W	0	ODT latency for writes. suppose to be 0.

**Table 12-171 DMC\_DRAM\_TWODTON 0x0138**

Bit(s)	R/W	Default	Description
5:0	R/W	0	DRAM write access ODT on time.

**Table 12-172 DMC\_DRAM\_TRETRAIN 0x0139**

Bit(s)	R/W	Default	Description
15:0	R/W	0	LPDDR4 OSCRING retraining timer(units = auto_refresh timer). sync with auto refresh

**Table 12-173 DMC\_DRAM\_PHYADDRON 0x013a**

Bit(s)	R/W	Default	Description
5:0	R/W	0	PHY_CTRL_ADDR_ON for new PHY. default: 12 unit: DfIClk

**Table 12-174 DMC\_DRAM\_PHYWRON 0x013b**

Bit(s)	R/W	Default	Description
6:0	R/W	0	$PHY\_CTRL\_WR\_ON. = 20 + WL + (tDQSS + tDQS2DQ + tWPST)/ddr\_clk + BL/2$

**Table 12-175 DMC\_DRAM\_PHYRDON 0x013c**

Bit(s)	R/W	Default	Description
6:0	R/W	0	$PHY\_CTRL\_RD\_ON. = 20 + RL + (tDQCK + tRPST + tDQSQ)/ddr\_clk + BL/2$

**Table 12-176 DMC\_DRAM\_ZQCTRL 0x013d**

Bit(s)	R/W	Default	Description
31:24.	R/W	0	in LPDDR4 case when to send ZQLAT command to both RANKs of LPDDR4 SDRAM
23:16.	R/W	0	when to send PHY ZQ UPDATE command.
15:8.	R/W	0	when to send ZQCS/ZQCAL to rank1 DDR SDRAM.
7:0.	R/W	0	when to send ZQCS/ZQCAL to rank0 DDR SDRAM.

DMC\_NFQ\_TMRD 0x0140  
 DMC\_NFQ\_TRFC 0x0141  
 DMC\_NFQ\_TRP 0x0142  
 DMC\_NFQ\_TRTW 0x0143  
 DMC\_NFQ\_TCL 0x0144  
 DMC\_NFQ\_TCWL 0x0145  
 DMC\_NFQ\_TRAS 0x0146  
 DMC\_NFQ\_TRC 0x0147  
 DMC\_NFQ\_TRCD 0x0148  
 DMC\_NFQ\_TRRD 0x0149  
 DMC\_NFQ\_TFAW 0x014a  
 DMC\_NFQ\_TRTP 0x014b  
 DMC\_NFQ\_TWR 0x014c  
 DMC\_NFQ\_TWTR 0x014d  
 DMC\_NFQ\_TCCD 0x014e  
 DMC\_NFQ\_TEXSR 0x014f  
 DMC\_NFQ\_TXS 0x0150  
 DMC\_NFQ\_TXP 0x0151  
 DMC\_NFQ\_TXPDLL 0x0152  
 DMC\_NFQ\_TZQCS 0x0153

---

DMC_NFQ_TCKSRE	0x0154
DMC_NFQ_TCKSRX	0x0155
DMC_NFQ_TCKE	0x0156
DMC_NFQ_TMOD	0x0157
DMC_NFQ_TDQS	0x0158
DMC_NFQ_TRSTL	0x0159
DMC_NFQ_TZQLAT	0x015a
DMC_NFQ_TMRR	0x015b
DMC_NFQ_TCKESR	0x015c
DMC_NFQ_TDPD	0x015d
DMC_NFQ_DFITCTRLDELAY	0x015e
DMC_NFQ_DFITPHYWRDATA	0x015f
DMC_NFQ_DFITPHYWRLAT	0x0160
DMC_NFQ_DFITRDDATAEN	0x0161
DMC_NFQ_DFITPHYRDLAT	0x0162
DMC_NFQ_DFITCTRLUPDMIN	0x0163
DMC_NFQ_DFITCTRLUPDMAX	0x0164
DMC_NFQ_DFITREFMSKI	0x0166
DMC_NFQ_DFITCTRLUPDI	0x0167
DMC_NFQ_DFITDRAMCLK	0x0168
DMC_NFQ_DFITLPRESP	0x016a
DMC_NFQ_TCKECK	0x016c
DMC_NFQ_TREFI	0x016d
DMC_NFQ_TSR	0x016e
DMC_NFQ_TCCDMW	0x016f
DMC_NFQ_TESCKE	0x0170
DMC_NFQ_TREFI_DDR3	0x0171
DMC_NFQ_TZQCAL	0x0172
DMC_NFQ_T10US	0x0173
DMC_NFQ_TMRRI	0x0174
DMC_NFQ_TXS_FAST	0x0175
DMC_NFQ_DFIODTCFG	0x0176
DMC_NFQ_DFIODTCFG1	0x0177
DMC_NFQ_TWODTON	0x0178
DMC_NFQ_TRETRAIN	0x0179
DMC_NFQ_PHYADDRON	0x017a
DMC_NFQ_PHYWRON	0x017b
DMC_NFQ_PHYRDON	0x017c

DMC\_NFQ\_ZQCTRL 0x017d

**Table 12-177 DMC\_DRAM\_DFITPHYUPDTYPE0 0x0180**

Bit(s)	R/W	Default	Description
31:0	R/W	0	dfi_phyupd_ack hold time for dfi_phyupd_req type = 0

**Table 12-178 DMC\_DRAM\_DFITPHYUPDTYPE1 0x0181**

Bit(s)	R/W	Default	Description
31:0	R/W	0	dfi_phyupd_ack hold time for dfi_phyupd_req type = 1

**Table 12-179 DMC\_DRAM\_DFITPHYUPDTYPE2 0x0182**

Bit(s)	R/W	Default	Description
31:0	R/W	0	dfi_phyupd_ack hold time for dfi_phyupd_req type = 2

**Table 12-180 DMC\_DRAM\_DFITPHYUPDTYPE3 0x0183**

Bit(s)	R/W	Default	Description
31:0	R/W	0	dfi_phyupd_ack hold time for dfi_phyupd_req type = 3

**Table 12-181 DMC\_DRAM\_MCFG 0x0186**

Bit(s)	R/W	Default	Description
31:30	R/W	0	not used.
29	R/W	0	freq_change request from dmc_clk_freq. 1: enable. 0 use local dmc freq_change request.
28	R/W	0	in LPDDR4 mode, retraining request from dmc_clk_freq. 1: enable. 0: use local dmc retraining request.
27	R/W	0	use auto_refr request from dmc_clk_freq. 1: enabled. 0 use local dmc auto refresh request.
26:24	R/W	0	rank1_d16_sel. 16bit mode, rank 1 byte selection.
23	R/W	0	not used.
22:20	R/W	0	rank0_d16_sel. 16bit mode, rank 0 byte selection.
19	R/W	0	send PHY ZQ command after siu self refresh exit. 1: enable. 0: disable.
18	R/W	0	send retrain command after siu self refresh exit. 1: enable. 0: disable.
17.	R/W	0	in DDR3/4 mode, send ZQCL command after exit register triggered self refresh.
16.	R/W	0	send refresh command after finish frequency change. 1 disable.
15.	R/W	0	send refresh command after finish LPDDR4 retraining. 1 disable.
14.	R/W	0	1 cke init high.
13	R/W	0	1 dbi write enable only for LPDDR4.
12.	R/W	0	1 dbi read high inversion.

Bit(s)	R/W	Default	Description
11.	R/W	0	1 dbi not enabled.
10	R/W	0	1 enable staggered chip select for 2 ranks DRAM.
9	R/W	0	1 enable send auto refresh command to DDR SDRAM when PCTL is in CFG/ STOP state.
8	R/W	0	send auto refr cmd before enter register triggered self refresh
7	R/W	0	send auto refr cmd after exit register triggered self refresh mode.
6	R/W	0	disable dram clock after enter register triggered self refresh.
5	R/W	0	send DFI_LP_REQ to PHY after enter register triggered elf refresh mode.
4	R/W	0	send DRAM to power down mode after enter self refresh. ONLY for LPDDR4.
3	R/W	0	send DFI_CTRLUPD_REQ after exit register triggered self refresh.
2	R/W	0	send ZQCS command after exit register triggered self refresh.
1	R/W	0	enable PHY triggered DFI_PHYUPD_REQ.
0	R/W	0	2T mode. always 1 in DDR3/4 mode.

Table 12-182 DMC\_DRAM\_ZQ\_CTRL 0x0187

Bit(s)	R/W	Default	Description
31:3	R/W	0	Not used
2	R/W	0	ZQ command must be send in different time.
1	R/W	0	ZQ command can be send at same time.
0	R/W	0	rank0 ZQ only

Table 12-183 DMC\_DRAM\_MCFG1 0x0188

Bit(s)	R/W	Default	Description
31:8	R/W	0	Not used
7:0	R/W	0xf8	LPDDR4 MRW DBI data mask pattern. //default 8'b11111000;

Table 12-184 DMC\_DRAM\_DFI\_CTRL 0x0189

Bit(s)	R/W	Default	Description
31	R/W	0	siu_dfi_lat err generation enable. 1 disable.
20	R/W	0	phy_ctrl_rd_on phy_ctrl_wr_on ignore 16bit mode. 1: sending 0xf in 16bit mode. 0: sending valid byte enable in 16bit mode
19	R/W	0	dfi_init_complete wait enable. 1: after dfi_init_complete, wait additional EXSR time for new command. 0: phy will handle all the timing after dfi_init_complete DMC can do everything they want.
18	R/W	0	dfi_rddata_cs_n polariy. 0: rank0 select = 2'b10. rank1 select = 2'b10. 1: rank0 select = 2'b01, rank1_select = 2'b10.
17	R/W	0	dfi_wrdata_cs_n polariy. 0: rank0 select = 2'b10. rank1 select = 2'b10. 1: rank0 select = 2'b01, rank1_select = 2'b10.



Bit(s)	R/W	Default	Description
16	R/W	0	force PHY ctrl_on if = 1, phy_ctrl_on signal will keep on.
15	R/W	0	siu_dfi1_lp_en
14	R/W	0	siu_dfi_lp_ack_and
13	R/W	0	siu_dfi_lp_ack_or
12	R/W	0	siu_dfi1_init_start_en
11	R/W	0	siu_dfi_init_com_and
10	R/W	0	siu_dfi_init_com_or
9	R/W	0	siu_dfi1_freq_en
8	R/W	0	siu_dfi1_dram_clk_dis_en
7	R/W	0	siu_dfi_phyupd_type_sel
6	R/W	0	siu_dfi1_phyupd_ack_en
5	R/W	0	siu_dfi_phyupd_req_and
4	R/W	0	siu_dfi_phyupd_req_or
3	R/W	0	siu_dfi_ctrlupd_ack_and
2	R/W	0	siu_dfi_ctrlupd_ack_or
1	R/W	0	siu_dfi1_ctrlupd_req_en
0	R/W	0	siu_dfi1_cmd_en

Table 12-185 DMC\_DRAM\_DFIINITCFG 0x018a

Bit(s)	R/W	Default	Description
31.	R/W	0	dfi_init_complete status. read only.
15:14	R/W	0	Frequency set 1 dfi_freq_ratio value.
12:8	R/W	0	Frequency set 1 dfi_freq value.
7:6	R/W	0	Frequency set 0 dfi_freq_ratio value.
5:1	R/W	0	Frequency set 0 dfi_freq value.
0.	R/W	0	dfi_init_start value can be use manually config dfi_init_start signal.

Table 12-186 DMC\_DRAM\_APD\_CTRL 0x018c

Bit(s)	R/W	Default	Description
19:16	R/W	0	16DFI_LP_WAKEUP value in APD DFI_LP_REQ mode
12	R/W	0	1 fast mode.
11	R/W	0	enable DFI_LP_REQ when enter Auto power down mode.
10	R/W	0	disable DFI_clk_disable when enter auto power down mode.
9:0	R/W	0	0 disable auto power down mode.

**Table 12-187 DMC\_DRAM\_ASR\_CTRL 0x018d**

Bit(s)	R/W	Default	Description
31	R/W	0	enable RETRAIN PHY after auto self refresh exit. for AM_PHY only.
30.	R/W	0	in DDR3/4 mode, send ZQCL command after exit from auto self refresh mode.
29.	R/W	0	enable PHY clock in LP2 mode.1 disable.
28.	R/W	0	enable dmc wait 10us after LP2 mode exit if it's long time the PHY in LP2 mode.
27:24	R/W	0	DFI_LP_WAKEUP value in self refresh DFI_LP_REQ mode.
23	R/W	0	send REFRESH command after exit from auto self referesh mode(ASR).
22	R/W	0	send REFERSH command before enter to Auto self refresh mode(ASR).
21	R/W	0	send ZQCS command after exit from Auto self refresh mode(ASR).
20	R/W	0	send dfi_ctrl_upd after exit from ASR mode
19	R/W	0	send power down command when enter ASR mode. //for LPDDR4 only.
18	R/W	0	set the PHY enter LP2 mode after enter ASR mode.
17	R/W	0	send DFI_LP_REQafter enter ASR mode.
16	R/W	0	set DFI_CLK_DISABLE after enter ASR mode.
15:0	R/W	0	0 disable auto ASR mode. Not 0 value enable ASR mode.

**Table 12-188 DMC\_DRAM\_REFR\_CTRL 0x0192**

Bit(s)	R/W	Default	Description
22	R/W	0	dmc to control DFI_CTRLUPD_REQ with zq generation together.
21	R/W	0	dmc to control PHY ZQ generation enable.
20	R/W	0	dmc to control zqlat(in LPDDR4 mode) generation enable.
19	R/W	0	dmc to control zqcs1 generation enable.
18	R/W	0	dmc to control zqcs0 generation enable.
17:8	R/W	0	auto refresh request pending cnt if there's page hit request.
7	R/W	0	retrain enable for auto refresh request. 1: enable. 0: disable.
6	R/W	0	Disabled auto refresh command if over 16 auto refresh command sent in 2 TRE-FI_DDR3 period
5	R/W	0	enable dmc send ZQCS command .
4.	R/W	0	enable dmc send DFI_CTRUPD_REQ.
3:1	R/W	0	how many refresh command send for one period. = this number + 1
0.	R/W	0	enable dmc send auto refresh command.

**Table 12-189 DMC\_DRAM\_FREQ\_CTRL 0x0093**

Bit(s)	R/W	Default	Description
31	R/W	0	write 1 to change frequency read 0: finished.
30	R/W	0	waiting for software to send some manual command. 1: waiting. 0: not ready yet.

Bit(s)	R/W	Default	Description
29:22	R/W	0	not used.
21	R/W	0	after Freqchange send PHY ZQ update.
20	R/W	0	send CTRLUPD_REQ to PHY after freq_change finished.
19:16	R/W	0	how many cycles to send PLL change req after init_complete signal to low.
15.	R/W	0	freq pre config_en. Before freq enter stop state let DMC configure DDR SDRAM.
14.	R/W	0	freq post config_en. After freq enter stop state let DMC configure DDR SDRAM.
13.	R/W	0	send zqcl after freq change in DDR3/4 mode.
12.	R/W	0	send zqcs after freq change. 1 enable. 0 not send.
11.	R/W	0	in AUTO MRW function MRW format.
10.	R/W	0	AUTO MRW function don't do auto MRW.
9.	R/W	0	1 FREQ MRW done. let FREQ change machine continue.
8	R/W	0	FREQ WAIT. 1 when freq change finishes, state machine stop at self refresh state in case there's something need to handle. 0 after freq change finishes the state machine go back to access state.
7	R/W	0	when change PLL setting, disable dmc clock
6	R/W	0	when change PLL setting, disable PHY DfiClk and DfiCtlClk.
5	R/W	0	check vpu_sleep_en ==1 when do FREQ change.if vpu_sleep_en == 0, just wait.
4	R/W	0	nxt frequency selection.1 = freq1. 0 = freq0.
3:1	R/W	0	not used.
0.	R/W	0	current frequency selection.

Table 12-190 DMC\_DRAM\_SCFG 0x0194

Bit(s)	R/W	Default	Description
2	R/W	0	1:to ask PCTL enter ACCESS STATE. 0:desert the request.
1	R/W	0	1:to ask PCTL enter SELF REFRESH STATE. 0:desert the request.
0	R/W	0	1:to ask PCTL enter STOP/CONFIG STATE. 0:desert the request.

Table 12-191 DMC\_DRAM\_STAT 0x0195

Bit(s)	R/W	Default	Description
31	Read only	0	rd latency error. 1 means after dfiphytrdlat cycles, the read data still not back.
28:24	Read only	0	dram_sr_state
23:20	Read only	0	stop_st
19:15	Read only	0	sleep_st
14:12	Read only	0	2 ACCESS STATUS 0: ACCESS is in normal working mode. 1: ACCESS sending precharge command.

Bit(s)	R/W	Default	Description
			2: ACCESS sending AUTO REFRESH command. 3: ACCESS sending DIF_CTRLUPD_REQ command. 4: ACCESS sending ZQCS command to DDR DRAM(ZQCAL for LPDDR4). 5: ACCESS sending ZQLATCH command to LPDDR4 only.
11:8	Read only	0	5 APD STATUS APD_IDLE 1: APD sending PRECHARGE command 2: APD sending CKE low command 3: APD sending DISABLE DRAM CLOCK command 4: APD sending DFI_LP_CTRL_REQ : APD in Auto Power down mode. : APD desert DFI_LP_CTRL_REQ : APD sending enable DRAM CLOCK command 8: APD sending out CKE high command.
7:4	Read only	0	DRAM IDLE DRAM_STOP/DRAM_CFG DRAM_ACCESS DRAM_SLEEP DRAM APD(AUTO POWER DOWN). IDLE -> STOP/CONFIG STOP -> SLEEP STOP -> ACCESS ACCESS -> SLEEP. ACCESS -> STOP A: ACCESS -> APD B: SLEEP -> STOP C: SLEEP -> ACCESS D: APD -> ACCESS
3	Read only	0	reserved.
2	Read only	0	1 DRAM enter normal working state.
1	Read only	0	1 DRAM enter sleep state. self refresh state.
0	Read only	0	1 dram enter cfg state.

Table 12-192 DMC\_DRAM\_STAT1 0x0196

Bit(s)	R/W	Default	Description
11:8	Read only	0	freq_st. 4'h0: IDLE 4'h1: FREQ_CHECK_VPU 4'h2: FREQ_WAITING_STOP 4'h3: FREQ_DRAM_CFG. waiting DMC/software to send special MRW/MPC command to configure DDR SDRAM either before Freq change or after freq change. 4'h4: FREQ_SELF_REFR_ST. 4'h5: FREQ_SET_DFI_FREQ. 4'h6: FREQ_DFI_INIT_START_HIGH. 4'h7: FREQ_CHANGE_PLL_ST. 4'h8: FREQ_UPDATA_REG. 4'h9: FREQ_DFI_INIT_START_LOW. 4'ha: FREQ_WAITING_FINISH 4'hb: FREQ_ZQCS 4'hc: FREQ_ZQ_LATCH 4'hd: FREQ_PVT
7:5	Read only	0	train_st
4:0	Read only	0	dram_phy_st

**Table 12-193 DMC\_PHY\_RETRAINING\_CTRL 0x0197**

Bit(s)	R/W	Default	Description
31	R/W	0	phy_retraining enable.
30	R/W	0	checkvpu sleep_en.
25:24	R/W	0	retraining dfi_freq[4:3], the [2:0] bit still use the dfi_freq bits to keep the frequency.
23:0	R/W	0	retraining period unit:100ns.

**Table 12-194 DMC\_DFI\_ERR\_STAT 0x0198**

Bit(s)	R/W	Default	Description
31:0	Read only	0	20. not used.
9:0	Read only	0	ddr0_dfi_error
8:5	Read only	0	ddr0_dfi_error_info.
4:0	Read only	0	ddr1_dfi_error.
3:0	Read only	0	ddr1_dfi_error_info.

**Table 12-195 DMC\_LP2\_TIMER 0x019a**

Bit(s)	R/W	Default	Description
15:0			timer setting to measure how long the chip is entered LP2 mode.

**Table 12-196 DMC\_DRAM\_COMPLETE\_WAIT 0x019a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	how long to waiting for DFI_INIT_COMPLETE become high after DFI_INIT_START triggered.

**Table 12-197 DMC\_DRAM\_DFI\_SWAP\_0 0x01a0**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_act_ndfi_act_n function select 6'h00 :the function itself. 6'h01 :act_n. 6'h02 : ras_n. 6'h03 :cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 : ba[1]. 6'h09 :ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a [11]. 6'h16 :a[12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a [17]. function select 6'h00 :the function itself. 6'h01 :act_n. 6'h02 :ras_n. 6'h03 : cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 :ba[1]. 6'h09 : ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a[11]. 6'h16 :a [12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a[17].

**Table 12-198 DMC\_DRAM\_DFI\_SWAP\_1 0x01a1**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_ras_n function select 6'h00 :the function itself 6'h01 :act_n. 6'h02 :ras_n. 6'h03 :cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 :ba[1]. 6'h09 :ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a[11]. 6'h16 :a[12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a[17].

**Table 12-199 DMC\_DRAM\_DFI\_SWAP\_2 0x01a2**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_cas_n function select 6'h00 :the function itself. 6'h01 :act_n. 6'h02 :ras_n. 6'h03 :cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 :ba[1]. 6'h09 :ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a[11]. 6'h16 :a[12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a[17].

**Table 12-200 DMC\_DRAM\_DFI\_SWAP\_3 0x01a3**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_we_n function select 6'h00 :the function itself. 6'h01 :act_n. 6'h02 :ras_n. 6'h03 :cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 :ba[1]. 6'h09 :ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a[11]. 6'h16 :a[12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a[17].

**Table 12-201 DMC\_DRAM\_DFI\_SWAP\_4 0x01a4**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_bg0 function select 6'h00 :the function itself. 6'h01 :act_n. 6'h02 :ras_n. 6'h03 :cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 :ba[1]. 6'h09 :ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a[11]. 6'h16 :a[12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a[17].

**Table 12-202 DMC\_DRAM\_DFI\_SWAP\_5 0x01a5**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_bg[1] function select 6'h00 :the function itself. 6'h01 :act_n. 6'h02 :ras_n. 6'h03 :cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 :ba[1]. 6'h09 :ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a[11]. 6'h16 :a[12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a[17].

**Table 12-203 DMC\_DRAM\_DFI\_SWAP\_6 0x01a6**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_ba[0] function select 6'h00 :the function itself. 6'h01 :act_n. 6'h02 :ras_n. 6'h03 :cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 :ba[1]. 6'h09 :ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a[11]. 6'h16 :a[12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a[17].

**Table 12-204 DMC\_DRAM\_DFI\_SWAP\_7 0x01a7**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_ba[1] function select 6'h00 :the function itself. 6'h01 :act_n. 6'h02 :ras_n. 6'h03 :cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 :ba[1]. 6'h09 :ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a[11]. 6'h16 :a[12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a[17].

**Table 12-205 DMC\_DRAM\_DFI\_SWAP\_8 0x01a8**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_ba[2] function select 6'h00 :the function itself. 6'h01 :act_n. 6'h02 :ras_n. 6'h03 :cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 :ba[1]. 6'h09 :ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a[11]. 6'h16 :a[12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a[17].

**Table 12-206 DMC\_DRAM\_DFI\_SWAP\_9 0x01a9**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[0] function select 6'h00 :the function itself. 6'h01 :act_n. 6'h02 :ras_n. 6'h03 :cas_n. 6'h04 :we_n. 6'h05 :bg[0]. 6'h06 :bg[1]. 6'h07 :ba[0]. 6'h08 :ba[1]. 6'h09 :ba[2]. 6'h0a :a[0]. 6'h0b :a[1]. 6'h0c :a[2]. 6'h0d :a[3]. 6'h0e :a[4]. 6'h0f :a[5]. 6'h10 :a[6]. 6'h11 :a[7]. 6'h12 :a[8]. 6'h13 :a[9]. 6'h14 :a[10]. 6'h15 :a[11]. 6'h16 :a[12]. 6'h17 :a[13]. 6'h18 :a[14]. 6'h19 :a[15]. 6'h1a :a[16]. 6'h1b :a[17].

**Table 12-207 DMC\_DRAM\_DFI\_SWAP\_10 0x01aa**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[1] function select

**Table 12-208 DMC\_DRAM\_DFI\_SWAP\_11 0x01ab**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[2] function select

**Table 12-209 DMC\_DRAM\_DFI\_SWAP\_12 0x01ac**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[3] function select

**Table 12-210 DMC\_DRAM\_DFI\_SWAP\_13 0x01ad**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[4] function select

**Table 12-211 DMC\_DRAM\_DFI\_SWAP\_14 0x01ae**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[5] function select

**Table 12-212 DMC\_DRAM\_DFI\_SWAP\_15 0x01af**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[6] function select

**Table 12-213 DMC\_DRAM\_DFI\_SWAP\_16 0x01b0**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[7] function select

**Table 12-214 DMC\_DRAM\_DFI\_SWAP\_17 0x01b1**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[8] function select

**Table 12-215 DMC\_DRAM\_DFI\_SWAP\_18 0x01b2**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[9] function select

**Table 12-216 DMC\_DRAM\_DFI\_SWAP\_19 0x01b3**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[10] function select

**Table 12-217 DMC\_DRAM\_DFI\_SWAP\_20 0x01b4**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[11] function select

**Table 12-218 DMC\_DRAM\_DFI\_SWAP\_21 0x01b5**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[12] function select

**Table 12-219 DMC\_DRAM\_DFI\_SWAP\_22 0x01b6**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[13] function select



**Table 12-220 DMC\_DRAM\_DFI\_SWAP\_23 0x01b7**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[14] function select

**Table 12-221 DMC\_DRAM\_DFI\_SWAP\_24 0x01b8**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[15] function select

**Table 12-222 DMC\_DRAM\_DFI\_SWAP\_25 0x01b9**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[16] function select

**Table 12-223 DMC\_DRAM\_DFI\_SWAP\_26 0x01ba**

Bit(s)	R/W	Default	Description
5:0	R/W	0	dfi_a[17] function select

**Table 12-224 DMC\_DRAM\_CMD 0x01d0**

Bit(s)	R/W	Default	Description
31.	R/W	0	cmd done. write 0 to clean.
30.	R/W	0	data done. write 0 to clean.
5.	R/W	0	user defined command.
4.	R/W	0	LPDDR4 MPC write data command (MPC WR FIFO).
3.	R/W	0	LPDDR4 MPC read data command (MPC RD Calibration and RD FIFO).
2.	R/W	0	LPDDR4 MPC-1 command (NOP, start DQS interval )
1.	R/W	0	mrr command.
0.	R/W	0	mrw command.

**Table 12-225 DMC\_DRAM\_CMD\_CODE 0x01d1**

Bit(s)	R/W	Default	Description
31:28	R/W	0	user command case = {act_n, ras_n, cas_n, we_n}
27:26.	R/W	0	128bits data cycles . 0:1 clock cycles; 1:2 clock cycles; 2:3 clock cycles; 3:4 clock cycles. LPDDR4: 4 clock cycles; DDR3/4/LPDDR3: 2 clock cycles.
25	R/W	0	MRW/MRR/MPC command rank 1 select. 1:select. 0:not select.
24.	R/W	0	MRW/MRR/MPC command rank 0 select. 1:select. 0:not select.

Bit(s)	R/W	Default	Description
23:16	R/W	0	MR addr. DDR4 case: 18:16 ba[2:0]. 20:19 BG[1:0].
15:0	R/W	0	opcode.

**Table 12-226 DMC\_DRAM\_CMD\_TIME 0x01d2**

Bit(s)	R/W	Default	Description
31:16	R/W	0	PRECMD timer. //delay how many cycle to start the command.
15:0	R/W	0	POST CMD timer//delay how many cycle after the command execute.

MPC WR FIFO command DBI WRITE data

**Table 12-227 DMC\_DRAM\_WSTRB0 0x01d3**

Bit(s)	R/W	Default	Description
31:16	R/W	0	the second cycle.
15:0	R/W	0	the first cycle.

MPC WR FIFO command DBI WRITE data

**Table 12-228 DMC\_DRAM\_WSTRB1 0x01d4**

Bit(s)	R/W	Default	Description
31:16	R/W	0	the second cycle.
15:0	R/W	0	the first cycle.

MPC RD FIFO command DBI read back data

**Table 12-229 DMC\_DRAM\_RDBI0 0x01d5**

Bit(s)	R/W	Default	Description
31:16	R/W	0	the Forth cycle.
15:0	R/W	0	the third cycle.

**Table 12-230 DMC\_DRAM\_RDBI1 0x01d6**

Bit(s)	R/W	Default	Description
31:16	R/W	0	the Forth cycle.
15:0	R/W	0	the third cycle.

DMC\_DRAM\_WDx and DMC\_DRAM\_RDx can be used as MRW command as Frequency change .

DMC\_DRAM\_WDx is for Freq1 DRAM MR setting, it would send to DRAM right before FREQ0->FREQ1

DMC\_DRAM\_RDx is for Freq0 DRAM MR setting. it would send to DRAM right before FREQ1->FREQ0.

One register can be one MRW command. So total 16 MRW command can be sent to DRAM.

**Table 12-231 DMC\_DRAM\_WDx and DMC\_DRAM\_RDx**

Bit(s)	R/W	Default	Description
31	R/W	0	MRW/USER command enable. 1: enabled command. 0 not enabled.
30	R/W	0	Last MRW/USER command. if this bit =1, After send this command, the DRAM controller will continue to frequency next stage.
29:26	R/W	0	SER COMMAND parameter: in DDR3/DDR4. {act_n, ras_n, cas_n, we_n} value for user command //in LPDDR4. bit 16: 1 4 cycles command. 0 2 cycles command.
25	R/W	0	MRW/USER command rank 1 select. 1: select. 0: not select.
24	R/W	0	MRW/USER command rank 0 select. 1: select. 0: not select.
23:0	R/W	0	USER command. //DDR3: 18:16 bank address. 15:0: address. //DDR4: 20:19 bank group address. 18:16: bank address. 15:0 address. //LPDDR3: 9:0. rising edge address. 19:10. falling edge address. //LPDDR4. 5:0, first edge address, 11:6 second edge address, 17:12: third edge address. 23:18, forth edge address. //MRW command format: //bit 23:16 MR addr. DDR4 case: 18:16 ba[2:0]. 20:19 BG[1:0]. //bit 15:0 opcode

DMC\_DRAM\_WD0 0x01e0  
 DMC\_DRAM\_WD1 0x01e1  
 DMC\_DRAM\_WD2 0x01e2  
 DMC\_DRAM\_WD3 0x01e3  
 DMC\_DRAM\_WD4 0x01e4  
 DMC\_DRAM\_WD5 0x01e5  
 DMC\_DRAM\_WD6 0x01e6  
 DMC\_DRAM\_WD7 0x01e7  
 DMC\_DRAM\_WD8 0x01e8  
 DMC\_DRAM\_WD9 0x01e9  
 DMC\_DRAM\_WD10 0x01ea  
 DMC\_DRAM\_WD11 0x01eb  
 DMC\_DRAM\_WD12 0x01ec  
 DMC\_DRAM\_WD13 0x01ed  
 DMC\_DRAM\_WD14 0x01ee  
 DMC\_DRAM\_WD15 0x01ef  
 DMC\_DRAM\_RD0 0x01f0  
 DMC\_DRAM\_RD1 0x01f1  
 DMC\_DRAM\_RD2 0x01f2  
 DMC\_DRAM\_RD3 0x01f3  
 DMC\_DRAM\_RD4 0x01f4  
 DMC\_DRAM\_RD5 0x01f5

---

DMC_DRAM_RD6	0x01f6
DMC_DRAM_RD7	0x01f7
DMC_DRAM_RD8	0x01f8
DMC_DRAM_RD9	0x01f9
DMC_DRAM_RD10	0x01fa
DMC_DRAM_RD11	0x01fb
DMC_DRAM_RD12	0x01fc
DMC_DRAM_RD13	0x01fd
DMC_DRAM_RD14	0x01fe
DMC_DRAM_RD15	0x01ff

#### DMC\_STICKY\_REGISTERS

Each DMC contains 64 sticky registers. Those registers are used to save some temporary value for software to debug DDR PHY. And they are implemented with SRAM. it won't lose if DMC was reset.

DMC_STICKY_0	0x0200
DMC_STICKY_1	0x0201
DMC_STICKY_2	0x0202
DMC_STICKY_3	0x0203
DMC_STICKY_4	0x0204
DMC_STICKY_5	0x0205
DMC_STICKY_6	0x0206
DMC_STICKY_7	0x0207
DMC_STICKY_8	0x0208
DMC_STICKY_9	0x0209
DMC_STICKY_10	0x020a
DMC_STICKY_11	0x020b
DMC_STICKY_12	0x020c
DMC_STICKY_13	0x020d
DMC_STICKY_14	0x020e
DMC_STICKY_15	0x020f
DMC_STICKY_16	0x0210
DMC_STICKY_17	0x0211
DMC_STICKY_18	0x0212
DMC_STICKY_19	0x0213
DMC_STICKY_20	0x0214
DMC_STICKY_21	0x0215
DMC_STICKY_22	0x0216
DMC_STICKY_23	0x0217
DMC_STICKY_24	0x0218

---

DMC_STICKY_25	0x0219
DMC_STICKY_26	0x021a
DMC_STICKY_27	0x021b
DMC_STICKY_28	0x021c
DMC_STICKY_29	0x021d
DMC_STICKY_30	0x021e
DMC_STICKY_31	0x021f
DMC_STICKY_32	0x0220
DMC_STICKY_33	0x0221
DMC_STICKY_34	0x0222
DMC_STICKY_35	0x0223
DMC_STICKY_36	0x0224
DMC_STICKY_37	0x0225
DMC_STICKY_38	0x0226
DMC_STICKY_39	0x0227
DMC_STICKY_40	0x0228
DMC_STICKY_41	0x0229
DMC_STICKY_42	0x022a
DMC_STICKY_43	0x022b
DMC_STICKY_44	0x022c
DMC_STICKY_45	0x022d
DMC_STICKY_46	0x022e
DMC_STICKY_47	0x022f
DMC_STICKY_48	0x0230
DMC_STICKY_49	0x0231
DMC_STICKY_50	0x0232
DMC_STICKY_51	0x0233
DMC_STICKY_52	0x0234
DMC_STICKY_53	0x0235
DMC_STICKY_54	0x0236
DMC_STICKY_55	0x0237
DMC_STICKY_56	0x0238
DMC_STICKY_57	0x0239
DMC_STICKY_58	0x023a
DMC_STICKY_59	0x023b
DMC_STICKY_60	0x023c
DMC_STICKY_61	0x023d
DMC_STICKY_62	0x023e

DMC\_STICKY\_63 0x023f

**Table 12-232 DMC\_TEST\_STA 0x0350**

Bit(s)	R/W	Default	Description
31:0	R/W	0	TEST start address low 32bits. low 6 bits should be 0.

**Table 12-233 DMC\_TEST\_STA\_HI 0x0351**

Bit(s)	R/W	Default	Description
1:0	R/W	0	TEST start address higher 2 bits.

**Table 12-234 DMC\_TEST\_EDA 0x0352**

Bit(s)	R/W	Default	Description
31:0	R/W	0	TEST end address low 32bits. low 6 bits should be 0.

**Table 12-235 DMC\_TEST\_EDA\_HI 0x0353**

Bit(s)	R/W	Default	Description
1:0	R/W	0	TEST end address higher 2 bits.

**Table 12-236 DMC\_TEST\_CTRL 0x0354**

Bit(s)	R/W	Default	Description
31.	R/W	0	enable test.
30.	R/W	0	when enable test, enable the write to DDR function.
29.	R/W	0	when enable test, enable the read from DDR function.
27.	R/W	0	enable to compare data. when do the read enable to enable the error comparison. suppose the read data should be same as the data in the write buffer.
26.	R/W	0	max_err_stop enable. 1: stop the test if err_cnt over max_err configure. 0: disable.
24.	R/W	0	done type.0 use the DMC_TEST_NUM register as the counter of test numbers. for write if the write command number == the DMC_TEST_NUM, the write is done. for read if the read command number == the DMC TEST_num, the read id done. for one read command can be repeated repeat number times. 1:finished at end address.
22:20	R/W	0	read repeat times. for non-sha function, we can define multi times of the read. the test module would repeat the same address repeat times.
19.	R/W	0	limit write.0 no outstanding write request limitation. 1:limit the outstanding write commands to the number of bits [15:8]
18.	R/W	0	limit read.0. no outstanding read request limitation. 1. limit the read outstanding request to the number of bits[7:0].
15:8	R/W	0	write outstanding commands limit.
7:0	R/W	0	read outstanding commands limit.

**Table 12-237 DMC\_TEST\_NUM 0x0355**

Bit(s)	R/W	Default	Description
31:0.	R/W	0	how many test command to end for the test if the DMC_TEST_CTRL bit 24 is 0.

**Table 12-238 DMC\_TEST\_WDG 0x0356**

Bit(s)	R/W	Default	Description
31:16.	R/W	0	write response watch dog.
15:0.	R/W	0	read response watch dog.

**Table 12-239 DMC\_TEST\_MAX\_ERR 0x0357**

Bit(s)	R/W	Default	Description
31:0	Read only	0	max err counter. if error cnt over this number, stop the test.

**Table 12-240 DMC\_TEST\_COMP\_MASK 0x0358**

Bit(s)	R/W	Default	Description
31:0	R/W	0	32bits for DMC TEST COMPARE bit enable. 1: to MASK this bit. 0: compare this bit.

**Table 12-241 DMC\_TEST\_WRDATA\_ADDR 0x0359**

Bit(s)	R/W	Default	Description
31:7	R/W	0	Not used.
6:0	R/W	0	WRDATA pattern APB configure address. total 128 32bits wrdata pattern.

**Table 12-242 DMC\_TEST\_WRDATA\_VALUE 0x035a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Write this register to configure the WRDATA pattern with DMC_TEST_WRDATA_ADDR address. Write this register would trigger WRDATA_ADDR + 1.

**Table 12-243 DMC\_TEST\_RDEXP\_ADDR 0x035b**

Bit(s)	R/W	Default	Description
31:7	R/W	0	Not used.
6:0	R/W	0	Expect data pattern APB configure address. Total 128 32bits read expected data pattern.

**Table 12-244 DMC\_TEST\_RDEXP\_VALUE 0x035c**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Write this register to configure the READ EXP data pattern with DMC_TEST_RDEXP_ADDR address. Write this register would trigger RDEXP_ADDR + 1..

**Table 12-245 DMC\_TEST\_ERR\_ADDR 0x0340**

Bit(s)	R/W	Default	Description
31:0	Read only	0	the first error address lower 32bits.

**Table 12-246 DMC\_TEST\_ERR\_ADDR\_HI 0x0341**

Bit(s)	R/W	Default	Description
31:0	Read only	0	the first error address higher 2 bits.

**Table 12-247 DMC\_TEST\_ERR\_CNT 0x0342**

Bit(s)	R/W	Default	Description
31:0	Read only	0	how many data error happens in the whole test period.

**Table 12-248 DMC\_TEST\_ERR\_DATA0 0x0343**

Bit(s)	R/W	Default	Description
31:0	Read only	0	the first error data low 32bit.

**Table 12-249 DMC\_TEST\_ERR\_DATA1 0x0344**

Bit(s)	R/W	Default	Description
31:0	Read only	0	the first error data low [63:32].

**Table 12-250 DMC\_TEST\_ERR\_DATA2 0x0345**

Bit(s)	R/W	Default	Description
31:0	Read only	0	the first error data low [95:64].

**Table 12-251 DMC\_TEST\_ERR\_DATA3 0x0346**

Bit(s)	R/W	Default	Description
31:0	Read only	0	the first error data low [127:96].



**Table 12-252 DMC\_TEST\_STS 0x0347**

Bit(s)	R/W	Default	Description
31,	R/W	0	test done bit. write 1 to clean.
30,	R/W	0	indicate address error
29:7	R/W	0	not used.
6.	R/W	0	read data resp error(caused by security or rd latency).
4,	R/W	0	Maximum error counter violation. write 1 to clean
3,	R/W	0	write done. write 1 to clean.
2,	R/W	0	read done. write 1 to clean
1,	R/W	0	write watchdog triggered. write 1 to clean
0,	R/W	0	read watchdog triggered. write 1 to clean.

**Table 12-253 DMC\_TEST\_WRCMD\_ADDR 0x0348**

Bit(s)	R/W	Default	Description
31:0	Read only	0	The current write address lower 32bits.

**Table 12-254 DMC\_TEST\_WRCMD\_ADDR\_HI 0x0349**

Bit(s)	R/W	Default	Description
1:0	Read only	0	The current write address higher 2 bits..

**Table 12-255 DMC\_TEST\_RDCMD\_ADDR 0x034a**

Bit(s)	R/W	Default	Description
31:0	Read only	0	The current write address lower 32bits.

**Table 12-256 DMC\_TEST\_RDCMD\_ADDR\_HI 0x034B**

Bit(s)	R/W	Default	Description
1:0	Read only	0	The current write address higher 2 bits.

**Table 12-257 DMC\_TEST\_RDRSP\_ADDR 0x034c**

Bit(s)	R/W	Default	Description
31:0	Read only	0	The current read response address lower 32bits.

**Table 12-258 DMC\_TEST\_RDRSP\_ADDR\_HI 0x034d**

Bit(s)	R/W	Default	Description
1:0	Read only	0	The current read response address higher 2 bits.

**Table 12-259 DMC\_SEC\_RANGE0\_STA 0x0400**

Bit(s)	R/W	Default	Description
31:22	R/W	0	Not used.
21:0	R/W	0	Region 0 start address AXI address [33:12] unit 4KB.

**Table 12-260 DMC\_SEC\_RANGE0\_EDA 0x0401**

Bit(s)	R/W	Default	Description
31:22	R/W	0	Not used.
21:0	R/W	0	Region 0 end address AXI address [33:12] unit 4KB.

**Table 12-261 DMC\_SEC\_RANGE0\_CTRL 0x0402**

Bit(s)	R/W	Default	Description
7:5	R/W	0	range0_security_level   range0 security level for DMC and DEMUX only.
4	R/W	0	range0_prot_en   range0 allow protect monitor function. 1:allowed. 0:not allowed.
3	R/W	0	range0_local_des_en   range0 data des enable. 1:enable. 0:disable. this bit works together with GLOBAL_DES_EN to decide this region data des enable or not.
2	R/W	0	range0_lock   lock range 0 registers. write 1 to lock range 0 register. can't change anymore.
1	R/W	0	range0_key_sel   range 0 data des key selection. 1:key1. 0:key0.
0	R/W	0	range0_en   range 0 enable bit. 1:enable, 0:disable.

**Table 12-262 DMC\_SEC\_RANGE0\_RID\_CTRL0 0x0403**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range0_rd_sid_en[31:0]

**Table 12-263 DMC\_SEC\_RANGE0\_RID\_CTRL1 0x0404**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range0_rd_sid_en[63:32]

**Table 12-264 DMC\_SEC\_RANGE0\_RID\_CTRL2 0x0405**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range0_rd_sid_en[95:64]

**Table 12-265 DMC\_SEC\_RANGE0\_RID\_CTRL3 0x0406**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range0_rd_sid_en[127:96]

**Table 12-266 DMC\_SEC\_RANGE0\_RID\_CTRL4 0x0407**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range0_rd_sid_en[159:128]

**Table 12-267 DMC\_SEC\_RANGE0\_WID\_CTRL0 0x0408**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range0_wr_sid_en[31:0]

**Table 12-268 DMC\_SEC\_RANGE0\_WID\_CTRL1 0x0409**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range0_wr_sid_en[63:32]

**Table 12-269 DMC\_SEC\_RANGE0\_WID\_CTRL2 0x040a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range0_wr_sid_en[95:64]

**Table 12-270 DMC\_SEC\_RANGE0\_WID\_CTRL3 0x040b**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range0_wr_sid_en[127:96]

**Table 12-271 DMC\_SEC\_RANGE0\_WID\_CTRL4 0x040c**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range0_wr_sid_en[159:128]

**Table 12-272 DMC\_SEC\_RANGE1\_STA 0x0410**

Bit(s)	R/W	Default	Description
31:22	R/W	0	Not used.
21:0	R/W	0	Region 1 start address AXI address [33:12] unit 4KB.

**Table 12-273 DMC\_SEC\_RANGE1\_EDA 0x0411**

Bit(s)	R/W	Default	Description
31:22	R/W	0	Not used.
21:0	R/W	0	Region 1 end address AXI address [33:12] unit 4KB.

**Table 12-274 DMC\_SEC\_RANGE1\_CTRL 0x0412**

Bit(s)	R/W	Default	Description
7:5	R/W	0	Range1_security_level   range0 security level for DMC and DEMUX only.
4	R/W	0	Range1_prot_en   range0 allow protect monitor function. 1:allowed. 0:not allowed.
3	R/W	0	Range1_local_des_en   range0 data des enable. 1:enable. 0:disable. this bit works together with GLOBAL_DES_EN to decide this region data des enable or not.
2	R/W	0	Range1_lock   lock range 0 registers. write 1 to lock range 0 register. can't change anymore.
1	R/W	0	Range1_key_sel   range 0 data des key selection. 1:key1. 0:key0.
0	R/W	0	Range1_en   range 0 enable bit. 1:enable, 0:disable.

**Table 12-275 DMC\_SEC\_RANGE1\_RID\_CTRL0 0x0413**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range1_rd_sid_en[31:0]

**Table 12-276 DMC\_SEC\_RANGE1\_RID\_CTRL1 0x0414**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range1_rd_sid_en[63:32]

**Table 12-277 DMC\_SEC\_RANGE1\_RID\_CTRL2 0x0415**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range1_rd_sid_en[95:64]

**Table 12-278 DMC\_SEC\_RANGE1\_RID\_CTRL3 0x0416**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range1_rd_sid_en[127:96]

**Table 12-279 DMC\_SEC\_RANGE1\_RID\_CTRL4 0x0417**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range1_rd_sid_en[159:128]

**Table 12-280 DMC\_SEC\_RANGE1\_WID\_CTRL0 0x0418**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range1_wr_sid_en[31:0]

**Table 12-281 DMC\_SEC\_RANGE1\_WID\_CTRL1 0x0419**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range1_wr_sid_en[63:32]

**Table 12-282 DMC\_SEC\_RANGE1\_WID\_CTRL2 0x041a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range1_wr_sid_en[95:64]

**Table 12-283 DMC\_SEC\_RANGE1\_WID\_CTRL3 0x041b**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range1_wr_sid_en[127:96]

**Table 12-284 DMC\_SEC\_RANGE1\_WID\_CTRL4 0x041c**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range1_wr_sid_en[159:128]

**Table 12-285 DMC\_SEC\_RANGE2\_STA 0x0420**

Bit(s)	R/W	Default	Description
31:22	R/W	0	Not used.
21:0	R/W	0	Region 2 start address AXI address [33:12] unit 4KB.

**Table 12-286 DMC\_SEC\_RANGE2\_EDA 0x0421**

Bit(s)	R/W	Default	Description
31:22	R/W	0	Not used.
21:0	R/W	0	Region 2 end address AXI address [33:12] unit 4KB.

**Table 12-287 DMC\_SEC\_RANGE2\_CTRL 0x0422**

Bit(s)	R/W	Default	Description
7:5	R/W	0	range2 security level for DMC and DEMUX only.
4	R/W	0	range2 allow protect monitor function. 1:allowed. 0:not allowed.
3	R/W	0	range2 data des enable. 1:enable. 0:disable. this bit works together with GLOBAL_DES_EN to decide this region data des enable or not.

Bit(s)	R/W	Default	Description
2	R/W	0	Range2_lock. lock range 2 registers. write 1 to lock range 0 register. can't change anymore.
1	R/W	0	Range2_key_sel. range 2 data des key selection. 1:key1. 0:key0.
0	R/W	0	Range2_en. range 2 enable bit. 1:enable, 0:disable.

**Table 12-288 DMC\_SEC\_RANGE2\_RID\_CTRL0 0x0423**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range2_rd_sid_en[31:0]

**Table 12-289 DMC\_SEC\_RANGE2\_RID\_CTRL1 0x0424**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range2_rd_sid_en[63:32]

**Table 12-290 DMC\_SEC\_RANGE2\_RID\_CTRL2 0x0425**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range2_rd_sid_en[95:64]

**Table 12-291 DMC\_SEC\_RANGE2\_RID\_CTRL3 0x0426**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range2_rd_sid_en[127:96]

**Table 12-292 DMC\_SEC\_RANGE2\_RID\_CTRL4 0x0427**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range2_rd_sid_en[159:128]

**Table 12-293 DMC\_SEC\_RANGE2\_WID\_CTRL0 0x0428**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range2_wr_sid_en[31:0]

**Table 12-294 DMC\_SEC\_RANGE2\_WID\_CTRL1 0x0429**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range2_wr_sid_en[63:32]

**Table 12-295 DMC\_SEC\_RANGE2\_WID\_CTRL2 0x042a**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range2_wr_sid_en[95:64]

**Table 12-296 DMC\_SEC\_RANGE2\_WID\_CTRL3 0x042b**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range2_wr_sid_en[127:96]

**Table 12-297 DMC\_SEC\_RANGE2\_WID\_CTRL4 0x042c**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range2_wr_sid_en[159:128]

Range3~14 is same as range 0, range 1 defines.

DMC\_SEC\_RANGE3\_STA 0x0430

DMC\_SEC\_RANGE3\_EDA 0x0431

DMC\_SEC\_RANGE3\_CTRL 0x0432

DMC\_SEC\_RANGE3\_RID\_CTRL0 0x433

DMC\_SEC\_RANGE3\_RID\_CTRL1 0x434

DMC\_SEC\_RANGE3\_RID\_CTRL2 0x435

DMC\_SEC\_RANGE3\_RID\_CTRL3 0x436

DMC\_SEC\_RANGE3\_RID\_CTRL4 0x437

DMC\_SEC\_RANGE3\_WID\_CTRL0 0x0438

DMC\_SEC\_RANGE3\_WID\_CTRL1 0x0439

DMC\_SEC\_RANGE3\_WID\_CTRL2 0x043a

DMC\_SEC\_RANGE3\_WID\_CTRL3 0x043b

DMC\_SEC\_RANGE3\_WID\_CTRL4 0x043c

DMC\_SEC\_RANGE4\_STA 0x0440

DMC\_SEC\_RANGE4\_EDA 0x0441

DMC\_SEC\_RANGE4\_CTRL 0x0442

DMC\_SEC\_RANGE4\_RID\_CTRL0 0x443

DMC\_SEC\_RANGE4\_RID\_CTRL1 0x0444

DMC\_SEC\_RANGE4\_RID\_CTRL2 0x0445

DMC\_SEC\_RANGE4\_RID\_CTRL3 0x0446

DMC\_SEC\_RANGE4\_RID\_CTRL4 0x0447

DMC\_SEC\_RANGE4\_WID\_CTRL0 0x0448

DMC\_SEC\_RANGE4\_WID\_CTRL1 0x0449

DMC\_SEC\_RANGE4\_WID\_CTRL2 0x044a

DMC\_SEC\_RANGE4\_WID\_CTRL3 0x044b  
DMC\_SEC\_RANGE4\_WID\_CTRL4 0x044c  
DMC\_SEC\_RANGE5\_STA 0x0450  
DMC\_SEC\_RANGE5\_EDA 0x0451  
DMC\_SEC\_RANGE5\_CTRL 0x0452  
DMC\_SEC\_RANGE5\_RID\_CTRL0 0x453  
DMC\_SEC\_RANGE5\_RID\_CTRL1 0x0454  
DMC\_SEC\_RANGE5\_RID\_CTRL2 0x0455  
DMC\_SEC\_RANGE5\_RID\_CTRL3 0x0456  
DMC\_SEC\_RANGE5\_RID\_CTRL4 0x0457  
DMC\_SEC\_RANGE5\_WID\_CTRL0 0x0458  
DMC\_SEC\_RANGE5\_WID\_CTRL1 0x0459  
DMC\_SEC\_RANGE5\_WID\_CTRL2 0x045a  
DMC\_SEC\_RANGE5\_WID\_CTRL3 0x045b  
DMC\_SEC\_RANGE5\_WID\_CTRL4 0x045c  
DMC\_SEC\_RANGE6\_STA 0x0460  
DMC\_SEC\_RANGE6\_EDA 0x0461  
DMC\_SEC\_RANGE6\_CTRL 0x0462  
DMC\_SEC\_RANGE6\_RID\_CTRL0 0x463  
DMC\_SEC\_RANGE6\_RID\_CTRL1 0x0464  
DMC\_SEC\_RANGE6\_RID\_CTRL2 0x0465  
DMC\_SEC\_RANGE6\_RID\_CTRL3 0x0466  
DMC\_SEC\_RANGE6\_RID\_CTRL4 0x0467  
DMC\_SEC\_RANGE6\_WID\_CTRL0 0x0468  
DMC\_SEC\_RANGE6\_WID\_CTRL1 0x0469  
DMC\_SEC\_RANGE6\_WID\_CTRL2 0x046a  
DMC\_SEC\_RANGE6\_WID\_CTRL3 0x046b  
DMC\_SEC\_RANGE6\_WID\_CTRL4 0x046c  
DMC\_SEC\_RANGE7\_STA 0x0470  
DMC\_SEC\_RANGE7\_EDA 0x0471  
DMC\_SEC\_RANGE7\_CTRL 0x0472  
DMC\_SEC\_RANGE7\_RID\_CTRL0 0x0473  
DMC\_SEC\_RANGE7\_RID\_CTRL1 0x0474  
DMC\_SEC\_RANGE7\_RID\_CTRL2 0x0475  
DMC\_SEC\_RANGE7\_RID\_CTRL3 0x0476  
DMC\_SEC\_RANGE7\_RID\_CTRL4 0x0477  
DMC\_SEC\_RANGE7\_WID\_CTRL0 0x0478  
DMC\_SEC\_RANGE7\_WID\_CTRL1 0x0479



DMC\_SEC\_RANGE7\_WID\_CTRL2 0x047a  
DMC\_SEC\_RANGE7\_WID\_CTRL3 0x047b  
DMC\_SEC\_RANGE7\_WID\_CTRL4 0x047c  
DMC\_SEC\_RANGE8\_STA 0x0480  
DMC\_SEC\_RANGE8\_EDA 0x0481  
DMC\_SEC\_RANGE8\_CTRL 0x0482  
DMC\_SEC\_RANGE8\_RID\_CTRL0 0x0483  
DMC\_SEC\_RANGE8\_RID\_CTRL1 0x0484  
DMC\_SEC\_RANGE8\_RID\_CTRL2 0x0485  
DMC\_SEC\_RANGE8\_RID\_CTRL3 0x0486  
DMC\_SEC\_RANGE8\_RID\_CTRL4 0x0487  
DMC\_SEC\_RANGE8\_WID\_CTRL0 0x0488  
DMC\_SEC\_RANGE8\_WID\_CTRL1 0x0489  
DMC\_SEC\_RANGE8\_WID\_CTRL2 0x048a  
DMC\_SEC\_RANGE8\_WID\_CTRL3 0x048b  
DMC\_SEC\_RANGE8\_WID\_CTRL4 0x048c  
DMC\_SEC\_RANGE9\_STA 0x0490  
DMC\_SEC\_RANGE9\_EDA 0x0491  
DMC\_SEC\_RANGE9\_CTRL 0x0492  
DMC\_SEC\_RANGE9\_RID\_CTRL0 0x0493  
DMC\_SEC\_RANGE9\_RID\_CTRL1 0x0494  
DMC\_SEC\_RANGE9\_RID\_CTRL2 0x0495  
DMC\_SEC\_RANGE9\_RID\_CTRL3 0x0496  
DMC\_SEC\_RANGE9\_RID\_CTRL4 0x0497  
DMC\_SEC\_RANGE9\_WID\_CTRL0 0x0498  
DMC\_SEC\_RANGE9\_WID\_CTRL1 0x0499  
DMC\_SEC\_RANGE9\_WID\_CTRL2 0x049a  
DMC\_SEC\_RANGE9\_WID\_CTRL3 0x049b  
DMC\_SEC\_RANGE9\_WID\_CTRL4 0x049c  
DMC\_SEC\_RANGE10\_STA 0x04a0  
DMC\_SEC\_RANGE10\_EDA 0x04a1  
DMC\_SEC\_RANGE10\_CTRL 0x04a2  
DMC\_SEC\_RANGE10\_RID\_CTRL0 0x04a3  
DMC\_SEC\_RANGE10\_RID\_CTRL1 0x04a4  
DMC\_SEC\_RANGE10\_RID\_CTRL2 0x04a5  
DMC\_SEC\_RANGE10\_RID\_CTRL3 0x04a6  
DMC\_SEC\_RANGE10\_RID\_CTRL4 0x04a7  
DMC\_SEC\_RANGE10\_WID\_CTRL0 0x04a8

---

DMC_SEC_RANGE10_WID_CTRL1	0x04a9
DMC_SEC_RANGE10_WID_CTRL2	0x04aa
DMC_SEC_RANGE10_WID_CTRL3	0x04ab
DMC_SEC_RANGE10_WID_CTRL4	0x04ac
DMC_SEC_RANGE11_STA	0x04b0
DMC_SEC_RANGE11_EDA	0x04b1
DMC_SEC_RANGE11_CTRL	0x04b2
DMC_SEC_RANGE11_RID_CTRL0	0x04b3
DMC_SEC_RANGE11_RID_CTRL1	0x04b4
DMC_SEC_RANGE11_RID_CTRL2	0x04b5
DMC_SEC_RANGE11_RID_CTRL3	0x04b6
DMC_SEC_RANGE11_RID_CTRL4	0x04b7
DMC_SEC_RANGE11_WID_CTRL0	0x04b8
DMC_SEC_RANGE11_WID_CTRL1	0x04b9
DMC_SEC_RANGE11_WID_CTRL2	0x04ba
DMC_SEC_RANGE11_WID_CTRL3	0x04bb
DMC_SEC_RANGE11_WID_CTRL4	0x04bc
DMC_SEC_RANGE12_STA	0x04c0
DMC_SEC_RANGE12_EDA	0x04c1
DMC_SEC_RANGE12_CTRL	0x04c2
DMC_SEC_RANGE12_RID_CTRL0	0x04c3
DMC_SEC_RANGE12_RID_CTRL1	0x04c4
DMC_SEC_RANGE12_RID_CTRL2	0x04c5
DMC_SEC_RANGE12_RID_CTRL3	0x04c6
DMC_SEC_RANGE12_RID_CTRL4	0x04c7
DMC_SEC_RANGE12_WID_CTRL0	0x04c8
DMC_SEC_RANGE12_WID_CTRL1	0x04c9
DMC_SEC_RANGE12_WID_CTRL2	0x04ca
DMC_SEC_RANGE12_WID_CTRL3	0x04cb
DMC_SEC_RANGE12_WID_CTRL4	0x04cc
DMC_SEC_RANGE13_STA	0x04d0
DMC_SEC_RANGE13_EDA	0x04d1
DMC_SEC_RANGE13_CTRL	0x04d2
DMC_SEC_RANGE13_RID_CTRL0	0x04d3
DMC_SEC_RANGE13_RID_CTRL1	0x04d4
DMC_SEC_RANGE13_RID_CTRL2	0x04d5
DMC_SEC_RANGE13_RID_CTRL3	0x04d6
DMC_SEC_RANGE13_RID_CTRL4	0x04d7

DMC\_SEC\_RANGE13\_WID\_CTRL0 0x04d8  
 DMC\_SEC\_RANGE13\_WID\_CTRL1 0x04d9  
 DMC\_SEC\_RANGE13\_WID\_CTRL2 0x04da  
 DMC\_SEC\_RANGE13\_WID\_CTRL3 0x04db  
 DMC\_SEC\_RANGE13\_WID\_CTRL4 0x04dc  
 DMC\_SEC\_RANGE14\_STA 0x04e0  
 DMC\_SEC\_RANGE14\_EDA 0x04e1  
 DMC\_SEC\_RANGE14\_CTRL 0x04e2  
 DMC\_SEC\_RANGE14\_RID\_CTRL0 0x04e3  
 DMC\_SEC\_RANGE14\_RID\_CTRL1 0x04e4  
 DMC\_SEC\_RANGE14\_RID\_CTRL2 0x04e5  
 DMC\_SEC\_RANGE14\_RID\_CTRL3 0x04e6  
 DMC\_SEC\_RANGE14\_RID\_CTRL4 0x04e7  
 DMC\_SEC\_RANGE14\_WID\_CTRL0 0x04e8  
 DMC\_SEC\_RANGE14\_WID\_CTRL1 0x04e9  
 DMC\_SEC\_RANGE14\_WID\_CTRL2 0x04ea  
 DMC\_SEC\_RANGE14\_WID\_CTRL3 0x04eb  
 DMC\_SEC\_RANGE14\_WID\_CTRL4 0x04ec

**Table 12-298 DMC\_SEC\_RANGE15\_CTRL 0x04f2**

Bit(s)	R/W	Default	Description
7:5	R/W	0	Range15 security level for DMC and DEMUX only.
4	R/W	0	Range15 allow protect monitor function. 1:allowed. 0:not allowed.
3	R/W	0	Range15 data des enable. 1:enable. 0:disable. this bit works together with GLOBAL_DES_EN to decide this region data des enable or not.
2	R/W	0	Range15_lock. lock range 15 registers. write 1 to lock range 0 register. can't change anymore.
1	R/W	0	Range15_key_sel. range 15 data des key selection. 1:key1. 0:key0.
0	R/W	0	Range15_en. range 15 enable bit. 1:enable, 0:disable.

**Table 12-299 DMC\_SEC\_RANGE15\_RID\_CTRL0 0x04f3**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range15_rd_sid_en[31:0]

**Table 12-300 DMC\_SEC\_RANGE15\_RID\_CTRL1 0x04f4**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range15_rd_sid_en[63:32]

**Table 12-301 DMC\_SEC\_RANGE15\_RID\_CTRL2 0x04f5**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range15_rd_sid_en[95:64]

**Table 12-302 DMC\_SEC\_RANGE15\_RID\_CTRL3 0x04f6**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range15_rd_sid_en[127:96]

**Table 12-303 DMC\_SEC\_RANGE15\_RID\_CTRL4 0x04f7**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range15_rd_sid_en[159:128]

**Table 12-304 DMC\_SEC\_RANGE15\_WID\_CTRL0 0x04f8**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range15_wr_sid_en[31:0]

**Table 12-305 DMC\_SEC\_RANGE15\_WID\_CTRL1 0x04f9**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range15_wr_sid_en[63:32]

**Table 12-306 DMC\_SEC\_RANGE15\_WID\_CTRL2 0x04fa**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range15_wr_sid_en[95:64]

**Table 12-307 DMC\_SEC\_RANGE15\_WID\_CTRL3 0x04fb**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range15_wr_sid_en[127:96]

**Table 12-308 DMC\_SEC\_RANGE15\_WID\_CTRL4 0x04fc**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Range15_wr_sid_en[159:128]

**Table 12-309 DMC\_DES\_PADDING 0x0500**

Bit(s)	R/W	Default	Description
31:0	R/W	0	des_pading. 32bits address padding used for DES dkey generation.

Table 12-310 DMC\_DES\_CTRL 0x0501

Bit(s)	R/W	Default	Description
31	R/W	0	range15_des_en policy:1 range15_des_en = range15_local_des_en && GLOBAL_DES_EN. 0:range15_des_en = range15_local_des_en    GLOBAL_DES_EN.
30	R/W	0	range14_des_en policy:1 range14_des_en = range14_local_des_en && GLOBAL_DES_EN. 0:range14_des_en = range14_local_des_en    GLOBAL_DES_EN.
29	R/W	0	range13_des_en policy:1 range13_des_en = range13_local_des_en && GLOBAL_DES_EN. 0:range13_des_en = range13_local_des_en    GLOBAL_DES_EN.
28	R/W	0	range12_des_en policy:1 range12_des_en = range12_local_des_en && GLOBAL_DES_EN. 0:range12_des_en = range12_local_des_en    GLOBAL_DES_EN.
27	R/W	0	range11_des_en policy:1 range11_des_en = range11_local_des_en && GLOBAL_DES_EN. 0:range11_des_en = range11_local_des_en    GLOBAL_DES_EN.
26	R/W	0	range10_des_en policy :1 range10_des_en = range10_local_des_en && GLOBAL_DES_EN. 0 :range10_des_en = range10_local_des_en    GLOBAL_DES_EN.
25	R/W	0	range9_des_en policy:1 range9_des_en = range9_local_des_en && GLOBAL_DES_EN. 0: range9_des_en = range9_local_des_en    GLOBAL_DES_EN.
24	R/W	0	range8_des_en policy:1 range8_des_en = range8_local_des_en && GLOBAL_DES_EN. 0: range8_des_en = range8_local_des_en    GLOBAL_DES_EN.
23	R/W	0	range7_des_en policy:1 range7_des_en = range7_local_des_en && GLOBAL_DES_EN. 0: range7_des_en = range7_local_des_en    GLOBAL_DES_EN.
22	R/W	0	range6_des_en policy:1 range6_des_en = range6_local_des_en && GLOBAL_DES_EN. 0: range6_des_en = range6_local_des_en    GLOBAL_DES_EN.
21	R/W	0	range5_des_en policy:1 range5_des_en = range5_local_des_en && GLOBAL_DES_EN. 0: range5_des_en = range5_local_des_en    GLOBAL_DES_EN.
20	R/W	0	range4_des_en policy:1 range4_des_en = range4_local_des_en && GLOBAL_DES_EN. 0: range4_des_en = range4_local_des_en    GLOBAL_DES_EN.
19	R/W	0	range3_des_en policy:1 range3_des_en = range3_local_des_en && GLOBAL_DES_EN. 0: range3_des_en = range3_local_des_en    GLOBAL_DES_EN.
18	R/W	0	range2_des_en policy:1 range2_des_en = range2_local_des_en && GLOBAL_DES_EN. 0: range2_des_en = range2_local_des_en    GLOBAL_DES_EN.
17	R/W	0	range1_des_en policy:1 range1_des_en = range1_local_des_en && GLOBAL_DES_EN. 0: range1_des_en = range1_local_des_en    GLOBAL_DES_EN.
16	R/W	0	range0_des_en policy:1 range0_des_en = range0_local_des_en && GLOBAL_DES_EN. 0: range0_des_en = range0_local_des_en    GLOBAL_DES_EN.
1	R/W	0	0   GLOBAL_DES_EN   1:Global DES enable. 0:global DES disable. default is 0.
0	R/W	0	0   DES_lock   one time lock bit. after write 1 to this bit, DMC_DES_CTRL, DMC_DES_PADDING can't be written or read.

**Table 12-311 DMC\_DES\_KEY0\_REG0 0x0502**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Key0[31:0]

**Table 12-312 DMC\_DES\_KEY0\_REG1 0x0503**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Key0[63:32]

**Table 12-313 DMC\_DES\_KEY0\_REG2 0x0504**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Key0[95:64] if 128bits des_key.

**Table 12-314 DMC\_DES\_KEY0\_REG3 0x0505**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Key0[127:96] if 128bits des_key.

**Table 12-315 DMC\_DES\_KEY1\_REG0 0x0506**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Key1[31:0]

**Table 12-316 DMC\_DES\_KEY1\_REG1 0x0507**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Key1[63:32]

**Table 12-317 DMC\_DES\_KEY1\_REG2 0x0508**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Key1[95:64] if 128bits des_key.

**Table 12-318 DMC\_DES\_KEY1\_REG3 0x0509**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Key1[127:96] if 128bits des_key.

APB access control for DMC DRAM timing parameter and DFI interface registers.

**Table 12-319 DMC\_APB\_SEC\_CTRL0 0x0510**

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used.
8	R/W	0	lock bit if this bit = 1, this register is locked and cannot modified anymore.
7:0	R/W	0x5	APB access enable for each APB user ID. one ID one bit. 1: enable. 0 disable.

APB access control for DMC DRAM timing parameter and DFI interface registers.

**Table 12-320 DMC\_APB\_SEC\_CTRL1 0x0511**

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used.
8	R/W	0	lock bit if this bit = 1, this register is locked and cannot modified anymore.
7:0	R/W	0x5	APB access enable for each APB user ID. one ID one bit. 1: enable. 0 disable.

APB access control for DMC PLL clock frequency control register.

**Table 12-321 DMC\_APB\_SEC\_CTRL2 0x0512**

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used.
8	R/W	0	lock bit if this bit = 1, this register is locked and cannot modified anymore.
7:0	R/W	0x5	APB access enable for each APB user ID. one ID one bit. 1: enable. 0 disable.

APB access control for DMC sticky control register.

**Table 12-322 DMC\_APB\_SEC\_CTRL3 0x0513**

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used.
8	R/W	0	lock bit if this bit = 1, this register is locked and cannot modified anymore.
7:0	R/W	0x5	APB access enable for each APB user ID. one ID one bit. 1: enable. 0 disable.

APB access control for DMC test control registers.

**Table 12-323 DMC\_APB\_SEC\_CTRL4 0x0514**

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used.
8	R/W	0	lock bit if this bit = 1, this register is locked and cannot modified anymore.
7:0	R/W	0x5	APB access enable for each APB user ID. one ID one bit. 1: enable. 0 disable.

APB access control for DMC clk reset control registers.

**Table 12-324 DMC\_APB\_SEC\_CTRL5 0x0515**

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used.
8	R/W	0	lock bit if this bit = 1, this register is locked and cannot modified anymore.
7:0	R/W	0x5	APB access enable for each APB user ID. one ID one bit. 1: enable. 0 disable.

APB access control for DMC clk reset control registers.

**Table 12-325 DMC\_APB\_SEC\_CTRL6 0x0516**

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used.
8	R/W	0	lock bit if this bit = 1, this register is locked and cannot modified anymore.
7:0	R/W	0x5	APB access enable for each APB user ID. one ID one bit. 1: enable. 0 disable.

APB access control for DMC normal control registers.

**Table 12-326 DMC\_APB\_SEC\_CTRL7 0x0517**

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used.
8	R/W	0	lock bit if this bit = 1, this register is locked and cannot modified anymore.
7:0	R/W	0x5	APB access enable for each APB user ID. one ID one bit. 1: enable. 0 disable.

APB access control for DDR-PHY registers.

**Table 12-327 DMC\_APB\_SEC\_CTRL8 0x0518**

Bit(s)	R/W	Default	Description
31:11	R/W	0	Not used.
10	R/W	0	PHY IMEM control 1: force PHY IMEM output 0. 0: normal working mode.
9	R/W	0	PHY DMEM control 1: force PHY DMEM output 0. 0: normal working mode.
8	R/W	0	lock bit if this bit = 1, this register is locked and cannot modified anymore.
7:0	R/W	0x5	APB access enable for each APB user ID. one ID one bit. 1: enable. 0 disable.

**Table 12-328 DMC\_SEC\_STATUS 0x051a**

Bit(s)	R/W	Default	Description
31:4		0	Not used.
3	R/W	0	1: security register APB write violation.( APB bus WDATA not enquire the register value when APB write). write 1 to clean this bit to 0.
2	R/W	0	Not used.
1	R/W	0	write security violation.
0	R/W		read security violation.



**Table 12-329 DMC\_VIO\_ADDR0 0x051b**

Bit(s)	R/W	Default	Description
31:0	Read only	0	ddr0 write secure violation address[31:0]

**Table 12-330 DMC\_VIO\_ADDR1 0x051c**

Bit(s)	R/W	Default	Description
31:19	Read only	0	not used.
18	Read only	0	ddr0 write address secure check error.
17	Read only	0	ddr0 write address overflow or not in this DDR channel.
16:15	Read only	0	dr0 write address bit 34:33.
14:0	Read only	0	ddr0_write violation USER BIT.

**Table 12-331 DMC\_VIO\_ADDR2 0x051b**

Bit(s)	R/W	Default	Description
31:0	Read only	0	ddr0 read secure check violation address[31:0]

**Table 12-332 DMC\_VIO\_ADDR3 0x051c**

Bit(s)	R/W	Default	Description
31:19	Read only	0	not used.
18	Read only	0	ddr0 read address secure check error.
17	Read only	0	ddr0 read address overflow or not in this DDR channel.
16:15	Read only	0	ddr0 read address bit 34:33.
14:0	Read only	0	ddr0 read violation USER BIT.

DMC\_AXI2DDR0 is for AXI address 0x0\_0000\_0000 ~ 0x0\_1fff\_ffff

**Table 12-333 DMC\_AXI2DDR0 0x0520**

Bit(s)	R/W	Default	Description
31:16	R/W	0	Not used.
15:8	R/W	0	DDR base address in 256 MB units.

Bit(s)	R/W	Default	Description
7:6	R/W	0	Not used.
5	R/W	0	Rank_sel. 1: rank1. 0: rank0.
4	R/W	0	Pure 16bits mode. this range is a pure 16 bits in a 16 bits only rank.
3	R/W	0	16bit mode in a 32bits bus, 1: enable. 0 not enabled.
1:0.	R/W	3	map mode. 2'b00, 2'b10: no mapped. 2'b01. this region is all mapped to this DMC channel. 2'b11. this region is mapped to both DMC in interleave mode.

DMC\_AXI2DDR1 is for AXI address 0x0\_2000\_0000 ~ 0x0\_3fff\_ffff

**Table 12-334 DMC\_AXI2DDR1 0x0521**

Bit(s)	R/W	Default	Description
31:16	R/W	0	Not used.
15:8	R/W	1	DDR base address in 256 MB units
7:6	R/W	0	Not used.
5	R/W	0	Rank_sel. 1 : rank1. 0 : rank0.
4	R/W	0	Pure 16bits mode. this range is a pure 16bits in a 16bits only rank.
3	R/W	0	16bit mode in a 32bits bus, 1: enable. 0: not enabled.
1:0.	R/W	3	map mode. 2'b00, 2'b10 : no mapped. 2'b01. this region is all mapped to this DMC channel. 2'b11. this region is mapped to both DMC in interleave mode.

DMC\_AXI2DDR2 to DMC\_AXI\_DDR1F are similar to DMC\_AXI2DDR0.

Each one is for AXI 512MB address:

DMC\_AXI2DDR2 0x0522  
 DMC\_AXI2DDR3 0x0523  
 DMC\_AXI2DDR4 0x0524  
 DMC\_AXI2DDR5 0x0525  
 DMC\_AXI2DDR6 0x0526  
 DMC\_AXI2DDR7 0x0527  
 DMC\_AXI2DDR8 0x0528  
 DMC\_AXI2DDR9 0x0529  
 DMC\_AXI2DDRA 0x052A  
 DMC\_AXI2DDRB 0x052B  
 DMC\_AXI2DDRC 0x052C  
 DMC\_AXI2DDRD 0x052D  
 DMC\_AXI2DDRE 0x052E  
 DMC\_AXI2DDRF 0x052F  
 DMC\_AXI2DDR10 0x0530

DMC_AXI2DDR11	0x0531
DMC_AXI2DDR12	0x0532
DMC_AXI2DDR13	0x0533
DMC_AXI2DDR14	0x0534
DMC_AXI2DDR15	0x0535
DMC_AXI2DDR16	0x0536
DMC_AXI2DDR17	0x0537
DMC_AXI2DDR18	0x0538
DMC_AXI2DDR19	0x0539
DMC_AXI2DDR1A	0x053A
DMC_AXI2DDR1B	0x053B
DMC_AXI2DDR1C	0x053C
DMC_AXI2DDR1D	0x053D
DMC_AXI2DDR1E	0x053E
DMC_AXI2DDR1F	0x053F

Table 12-335 DMC\_DDR\_CTRL 0x0540

Bit(s)	R/W	Default	Description
24:22	R/W	0	3'b000:ddr3 mode. 3'b001:ddr4 mode. 3'b010:lpddr3 mode. 3'b011:lpddr4 mode.
21.	R/W	0	rank1 enable bit. if 1, rank1 used the address map is as bit 5:3 defined.
20	R/W	0	DDR4 BG1 enable bit.
18:	R/W	0	ddrphy_16b, DDR PHY DDR SDRAM data bits width. 1:16bits. 0:32 bits.
16.	R/W	0	ddr_16b, 1:only use 16bits data in a 32bits DDR PHY data interface. 0:normal data interface.
7:4	R/W	0	ddr1_size.DDR rank1 size control. 4'b0000:DDR rank 1 is 128MB. 4'b0001:DDR rank 1 is 256MB. 4'b0010:DDR rank 1 is 512MB. 4'b0011:DDR rank 1 is 1GB. 4'b0100:DDR rank 1 is 2GB. 4'b0101:DDR rank 1 is 4GB. 4'b1000:DDR rank 1 is 192MB. 4'b1001:DDR rank 1 is 374MB. 4'b1010:DDR rank 1 is 768MB. 4'b1011:DDR rank 1 is 1.5GB. 4'b1100:DDR rank 1 is 3GB. others: reserved.
3:0	R/W	0	ddr0_size. DDR rank0 size control. 4'b0000:DDR rank 0 is 128MB. 4'b0001:DDR rank 0 is 256MB. 4'b0010:DDR rank 0 is 512MB. 4'b0011:DDR rank 0 is 1GB. 4'b0100:DDR rank 0 is 2GB. 4'b0101:DDR rank 0 is 4GB. 4'b1000:DDR rank 1 is 192MB. 4'b1001:DDR rank 1 is 374MB. 4'b1010:DDR rank 1 is 768MB. 4'b1011:DDR rank 1 is 1.5GB. 4'b1100:DDR rank 1 is 3GB. others: reserved.

Table 12-336 DMC\_DDR\_CTRL1 0x0541

Bit(s)	R/W	Default	Description
31:5		0	Not used.
4	R/W	0	DMC_DDR_LOCK. 1: LOCK DMC_DDR_CTRL, DMC_DDR_CTRL1, DMC_AXI2DDR <sub>x</sub> , DDR0/1_ADDRRMAP <sub>x</sub> registers. those register can't modified any more. 0: all these registers can be read/write by secure APB access.

Bit(s)	R/W	Default	Description
3	R/W	0	for the split address. 1: the strip address bit == 0 to this channel. 0: the strip address bit == 1 to this channel.
2:0	R/W	0	DDR strip size. if axi2ddr configuration is split to DMC. 3'b000: 64Byte. 3'b001: 128Byte. 3'b010: 256Byte. 3'b011: 512Byte. Other: reserved for future.

Table 12-337 DMC\_RANK0\_ADDRMAP\_0 0x0542

Bit(s)	R/W	Default	Description
29:25	R/W	0	ca8.
24:20	R/W	0	ca7.
19:15	R/W	0	ca6.
14:10	R/W	0	ca5.
9:5	R/W	0	ca4.
4:0	R/W	0	ca3.

Table 12-338 DMC\_RANK0\_ADDRMAP\_1 0x0543

Bit(s)	R/W	Default	Description
29:25	R/W	0	ra2.
24:20	R/W	0	ra1.
19:15	R/W	0	ra0.
14:10	R/W	0	ca11.
9:5	R/W	0	ca10.
4:0	R/W	0	ca9.

Table 12-339 DMC\_RANK0\_ADDRMAP\_2 0x0544

Bit(s)	R/W	Default	Description
29:25	R/W	0	ra8.
24:20	R/W	0	ra7.
19:15	R/W	0	ra6.
14:10	R/W	0	ra5.
9:5	R/W	0	ra4.
4:0	R/W	0	ra3.

Table 12-340 DMC\_RANK0\_ADDRMAP\_3 0x0545

Bit(s)	R/W	Default	Description
29:25	R/W	0	ra14
24:20	R/W	0	ra13

Bit(s)	R/W	Default	Description
19:15	R/W	0	ra12
14:10	R/W	0	ra11
9:5	R/W	0	ra10
4:0	R/W	0	ra9

Table 12-341 DMC\_RANK0\_ADDRMAP\_4 0x0546

Bit(s)	R/W	Default	Description
29:25	R/W	0	ra16 for DDR4 SDRAM
24:20	R/W	0	bg1 for DDR4 SDRAM.
19:15	R/W	0	ba2. or bg0 for DDR4.
14:10	R/W	0	ba1.
9:5	R/W	0	ba0.
4:0	R/W	0	ra15.

Table 12-342 DMC\_RANK1\_ADDRMAP\_0 0x0547

Bit(s)	R/W	Default	Description
29:25	R/W	0	ca8.
24:20	R/W	0	ca7.
19:15	R/W	0	ca6.
14:10	R/W	0	ca5.
9:5	R/W	0	ca4.
4:0	R/W	0	ca3.

Table 12-343 DMC\_RANK1\_ADDRMAP\_1 0x0548

Bit(s)	R/W	Default	Description
29:25	R/W	0	ra2.
24:20	R/W	0	ra1.
19:15	R/W	0	ra0.
14:10	R/W	0	ca11.
9:5	R/W	0	ca10.
4:0	R/W	0	ca9.

Table 12-344 DMC\_RANK1\_ADDRMAP\_2 0x0549

Bit(s)	R/W	Default	Description
29:25	R/W	0	ra8.
24:20	R/W	0	ra7.
19:15	R/W	0	ra6.

Bit(s)	R/W	Default	Description
14:10	R/W	0	ra5.
9:5	R/W	0	ra4.
4:0	R/W	0	ra3.

Table 12-345 DMC\_RANK1\_ADDRMAP\_3 0x054a

Bit(s)	R/W	Default	Description
29:25	R/W	0	ra14
24:20	R/W	0	ra13
19:15	R/W	0	ra12
14:10	R/W	0	ra11
9:5	R/W	0	ra10
4:0	R/W	0	ra9

Table 12-346 DMC\_RANK1\_ADDRMAP\_4 0x054b

Bit(s)	R/W	Default	Description
29:25	R/W	0	ra16 for DDR4 SDRAM
24:20	R/W	0	bg1 for DDR4 SDRAM.
19:15	R/W	0	ba2. or bg0 for DDR4.
14:10	R/W	0	ba1.
9:5	R/W	0	ba0.
4:0	R/W	0	ra15.

Table 12-347 DMC\_SEC\_TEST\_STA 0x0550

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used.
19:0	R/W	0	Start address reserved for DMC TEST. unit 4KB.

Table 12-348 DMC\_SEC\_TEST\_EDA 0x0551

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used.
19:0	R/W	0	End address reserved for DMC TEST. unit 4KB.

Table 12-349 DMC\_SEC\_TEST\_CTRL 0x0552

Bit(s)	R/W	Default	Description
7	R/W	0	lock bit. if this bit = 1, the DMC_SEC_TEST_* register can't changed.
6	R/W	0	Dmc_test_enable. 1: enable dmc_test in this address. 0: dmc_test can't be used.

Bit(s)	R/W	Default	Description
5	R/W	0	1: strip position = 0 for this DMC. 0: strip position == 1 for this DMC.
4:2	R/W	0	3'h0: 64Byte strip size in interleaved mode. 3'h1: 128Byte strip size in interleaved mode. 3'h2: 256Byte strip size in interleaved mode. 3'h3: 512Byte strip size in interleaved mode. Others: reserved.
1:0	R/W	0	2'b01 this TEST address range is for this DMC only 2'b11 this TEST address range is for DMC0 and DMC0 interleaved.

## 12.2 eMMC/SD

### 12.2.1 Overview

The SoC has the following features of eMMC/SD.

- Supports SDSC/SDHC/SDXC card
- Supports eMMC and MMC card specification version 5.1 up to HS400 with data content TDES crypto
- 1 bit, 4 Bits, 8 Bits data lines supported (8 Bits only for MMC)
- Descriptor chain architecture, timing tuning and adjustment
- Supports descriptor-based internal DMA controller

This module uses eMMC/SD CONTROLLERS to connect varied SD/MMC Card, or eMMC protocol compatible memory with high throughput.

### 12.2.2 Pin Description

**Table 12-350 Pin Description of eMMC/SD/SDIO Module**

Name	Type	Description	Speed (MHz)
CLK	Output	SD eMMC clock, 0~200MHz	200
DS	DS Data strobe for eMMC HS400 mode	-	-
DAT[7:0]	SD Card 4 Bits, eMMC 8 Bits Input/Output/Push-Pull Internal pull-up for pins not used	Data, 1,4,8 mode	200
CMD	Input/Output/Push-Pull/Open-Drain Open-drain for initialization Push-pull for fast command transfer ROD is connected when in open-drain mode.	Command Response	200
Rst_n	eMMC required	Hardware reset	Low
IRQ	SD Card or eMMC not used.	Device interrupt can be replaced by DAT[1]	Low

## 12.2.3 eMMC/SD Mode

### eMMC Mode

**Table 12-351 eMMC Mode**

Mode Name	Data Rate	IO Voltage	Bus Width	Frequency	Max Data Transfer
Legacy MMC card	Single	3/1.8V	1, 4, 8	0-26MHz	26MB/s
High Speed SDR	Single	3/1.8V	1,4, 8	0-52MHz	52MB/s
High Speed DDR	Dual	3/1.8V	4, 8	0-52MHz	104MB/s
HS200	Single	1.8V	4, 8	0-200MHz	200MB/s

The HS200 mode offers the following features:

- SDR Data sampling method
- CLK frequency up to 200MHz Data rate – up to 200MB/s
- 4 or 8-bits bus width supported
- Single ended signaling with 4 Drive Strengths
- Signaling levels of 1.8V
- Tuning concept for Read Operations

### SD Mode

**Table 12-352 SD Mode**

Mode Name	Data Rate	IO Voltage	Bus Width	Frequency	Max Data Transfer
Default Speed	Single	3.3V	1, 4	0-25MHz	12.5MB/s
High Speed	Single	3.3V	1, 4	0-50MHz	25MB/s
SDR12	Single	1.8V	1,4	0-25MHz	12.5MB/s
SDR25	Single	1.8V	1,4	0-50MHz	25MB/s
SDR50	Single	1.8V	1,4	0-100MHz	50MB/s
SDR104 (highest)	Single	1.8V	1,4	0-208MHz	104MB/s
DDR50	Dual	1.8V	4	0-50MHz	50MB/s

## 12.2.4 Descriptor

### Structure

The descriptor has a size of 4x32 Bits.



Table 12-353 Descriptor Structure

byte	7	6	5	4	3	2	1	0
0	length[7:0]							
1	Timeout 4 Bits				End of chain	R1b	block mode	length[8]
2	data num	resp num	resp 128	resp nocrc	Data wr	Data io	No cmd	No resp
3	owner	error	cmd index 6 Bits					
4	cmd argument 32 Bits							
5								
6								
7								
8	data address 32 Bits or data 0-4 bytes [1]Big Endian, [0]SRAM							
9								
10								
11								
12	response address 32 Bits or response irq en [0]SRAM							
13								
14								
15								

## Definition

Name	Bits	Description
Length	Cmd_cfg[8:0]	same as spec, copy the content from command argument into this field, different byte size and 512 bytes, different number of blocks and infinite blocks. If the command is operating on bytes, block mode = 0, this field contains the number of bytes to read or write, A value of 0 shall cause 512 bytes to be read to written, if the command is operating on blocks, block mode = 1, this field contains the number of blocks, a value of 0 is infinite number of blocks.
Block_mode	Cmd_cfg[9]	1: the read or write shall be performed on block basis. The block size is from SD/eMMC device, and saved in APB3 register in module. 0: the read or write is byte based.
R1b	Cmd_cfg[10]	1: check the DAT0 busy after received response R1 0: do not check the DAT0 busy state.
End_of_chain	Cmd_cfg[11]	1: it is the end of descriptor chain, the host stops and issues IRQ after this descriptor is done. 0: the host reads next descriptor and continues. The command chain execution is started by write an APB3 register and stopped by the "end of chain" or clear a APB3 start register, or found one descriptor with owner is set to 0.
Timeout	Cmd_cfg[15:12]	2timeout ms when timeout != 0, max timeout 32.768s, when over the timeout limit, error bit is set, IRQ is issued. When timeout is 0, no time limit.
No_resp	Cmd_cfg[16]	1: this command doesn't have response, used with command doesn't have response. 0: there is a response. The module waits for response, the response time-out setting is in APB3 register.

Name	Bits	Description
No_cmd	Cmd_cfg[17]	1: this descriptor doesn't have command in it, it does data DMA only, used with command to read or write SD/eMMC with data from multiple locations.
Data_io	Cmd_cfg[18]	1: there is data action in this descriptor, used with command have data process. 0: there is no data read/write action.
Data_wr	Cmd_cfg[19]	1: host writes data to SD/eMMC 0: host read data from SD/eMMC
Resp_nocrc	Cmd_cfg[20]	1: R3 response doesn't have CRC. 0: host does CRC check.
Resp_128	Cmd_cfg[21]	1: R3 response with 128 Bits information. 0: 32 Bits responses.
Resp_num	Cmd_cfg[22]	1: the resp_addr is the IRQ enable Bits, used to check the response error status, when there is an error, IRQ[14] is issued, the first 4 bytes of response is saved into resp_addr. 0: save response into SRAM or DDR location.
Data_num	Cmd_cfg[23]	1: save 4 bytes of data back into descriptor itself at bytes 8~11.
Cmd_index	Cmd_cfg[29:24]	The SD/eMMC command index. Desc REG wr: 4 reg44, 12 reg4c.
Error	Cmd_cfg[30]	Write back by host. The combined error from command, response, data, includes CRC error and timeout. When it is set the descriptor execution is stopped and an IRQ is issued. The CPU can read SD_EMMC_STATUS register to get detail information.
Owner	Cmd_cfg[31]	Programmed by CPU to 1, cleared by host to 0. 1: the descriptor is valid and owned by host, after it is done, even it has error, the owner bit is cleared, the descriptor is owned by CPU. In case of descriptor chain execution when host found a descriptor with "0" owner bit, it will stop.
Cmd_arg	Desc 4~7 bytes	32 Bits. The actual command argument some of the previous fields are copied from this command argument, the software need to make sure they are consistent. Desc REG wr: new value Data_addr: write mask, 1: change, 0: no change.
Data_addr	Desc 8~11 bytes	32 Bits. If the data_num is 0, the content is data address. If the data_num is 1, this content is 4 data bytes. When it is an address: Data_addr[0]: 1: SRAM address, 0: DDR address. If the data_addr[31:12] matches with SD_EMMC_BASE, it is SRAM address. Data_addr[1]: 1: 4 bytes big endian, 0: little endian(default).
Resp_addr	Desc 12~15 bytes	32 Bits If the resp_num is 0, the content is resp address. If the resp_num is 1, before execution, it is the response IRQ enable Bits, after execution, it is the first 4 response bytes. When it is an address: Resp_addr[0]: 1: SRAM address, 0: DDR address. If the resp_addr[31:12] matches with SD_EMMC_BASE, it is SRAM address.

## 12.2.5 Register Description

Each register final address = module base address+ address \* 4

Where module addresses are 0xfe08c000 for port C, 0xfe08a000 for port B, and 0xfe088000 for port A.

**Table 12-354 SD\_EMMC\_CLOCK 0x0**

Bit(s)	R/W	Default	Description
30	R/W	0	Cfg_irq_sdio_sleep_ds : 1: select DS as IRQ source during sleep.
29	R/W	0	Cfg_irq_sdio_sleep: 1: enable IRQ sdio when in sleep mode. When DAT1 IRQ, the controller uses PCLK to detect DAT1 level and starts core clock, the core initials IRQ_period and detect DAT1 IRQ.
28	R/W	0	Cfg_always_on: 1: Keep clock always on 0: Clock on/off controlled by activities. Any APB3 access or descriptor execution will turn clock on. Recommended value: 0
27:22	R/W	0	Cfg_rx_delay: RX clock delay line, 6 bits 0: no delay, n: delay n*50ps Maximum delay 3150ps.
21:16	R/W	0	Cfg_tx_delay: TX clock delay line, 6 bits 0: no delay, n: delay n*50ps Maximum delay 63*50ps =3150ps.
15:14	R/W	0	Cfg_sram_pd: Sram power down
13:12			Cfg_rx_phase: RX clock phase 0: 0 phase, 1: 90 phase, 2: 180 phase, 3: 270 phase. Recommended value: 0
11:10	R/W	0	Cfg_tx_phase: TX clock phase 0: 0 phase, 1: 90 phase, 2: 180 phase, 3: 270 phase. Recommended value: 2
9:8	R/W	0	Cfg_co_phase: Core clock phase 0: 0 phase, 1: 90 phase, 2: 180 phase, 3: 270 phase. Recommended value: 2
7:6	R/W	0	Cfg_src: Clock source 0: Crystal 24MHz or other frequencies selected by clock reset test control register. 1: Fix PLL, 1000MHz Recommended value: 1
5:0	R/W	0	Cfg_div: Clock divider Frequency = clock source/cfg_div Clock off: cfg_div==0, the clock is disabled Divider bypass: cfg_div==1, clock source is used as core clock without divider Maximum divider 63.

**Table 12-355 SD\_EMMC\_DELAY1 0x4**

Bit(s)	R/W	Default	Description
31:30	R/W	0	Reserved.
29:24	R/W	0	Dly[4]: Data 4 delay line
23:18	R/W	0	Dly[3]: Data 3 delay line
17:12	R/W	0	Dly[2]: Data 2 delay line
11:6	R/W	0	Dly[1]: Data 1 delay line
5:0	R/W	0	Dly[0]: Data 0 delay line Total delay = 50ps * Dly When Dly == 0, no delay. When Dly ==63, 3150ps delay. NOTE: the 50ps is typical delay, actually delay may vary from chip to chip, from different temperature.

**Table 12-356 SD\_EMMC\_DELAY2 0x8**

Bit(s)	R/W	Default	Description
31:30			Reserved.
29:24			Dly[9]: Data 9 delay line
23:18			Dly[8]: Data 8 delay line
17:12			Dly[7]: Data 7 delay line
11:6			Dly[6]: Data 6 delay line
5:0			Dly[5]: Data 5 delay line Total delay = 50ps * Dly When Dly == 0, no delay. When Dly ==63, 3150ps delay. NOTE: the 50ps is typical delay, actually delay may vary from chip to chip, from different temperature.

**Table 12-357 SD\_EMMC\_ADJUST 0xc**

Bit(s)	R/W	Default	Description
31:23			Unused
22	R/W	0	Adj_auto 1: Use cali_dut's first falling edge to adjust the timing, set cali_enable to 1 to use this function, simulation shows it can tracking 2.5ns range with 800ppm. 0: disable Working for HS200 mode, set Cali_enable to 1. Enhanced after gxlx project. Use RESP and DAT0 as reference, Separate RESP and DAT0, adjust the timing whenever there is a transition, insert a sample when there is no transition.
21:16	R/W	0	Adj_delay: Resample the input signals when clock index==adj_delay
15	R/W	0	Reserved
14	R/W	0	Cali_rise: 1: test the rising edge, recording rising edge location only. 0: test the falling edge
13	R/W	0	Adj_fixed: Adjust interface timing by resampling the input signals
12	R/W	0	Cali_enable: 1: Enable calibration 0: shut off to save power.
11:8	R/W	0	Cali_sel: Select one signal to be tested Signals are labeled from 0 to 9 the same as delay lines. Only one signal is tested at anytime. For example: Cali_sel == 9, test CMD line.

**Table 12-358 SD\_EMMC\_CALOUT 0x10**

Bit(s)	R/W	Default	Description
31:16			Unused
15:8	R		Cali_setup
7	R		Cali_vld: The reading is valid When there is no rising edge or falling edge event, the valid is low, this reading is not valid.
5:0	R		Cali_idx: The event happens at this index, The index starts from rising edge of core clock from 0, 1, 2, ...

**Table 12-359 SD\_EMMC\_ADJ\_IDX\_LOG 0x20**

Bit(s)	R/W	Default	Description
31:30			Current, Auto_adj mode, Current 6 bits adj_idx
29:24			Previous 1
23:18			Previous 2
17:12			Previous 3
11:6			Previous 4
5:0			Previous 5: Last two bits of previous 5

**Table 12-360 SD\_EMMC\_CLKTEST\_LOG 0x24**

Bit(s)	R/W	Default	Description
31			Clktest_done, Test done
30:0			Clktest_times, Test clock core for $2^{\wedge}$ Clktest_exp

**Table 12-361 SD\_EMMC\_CLKTEST\_OUT 0x28**

Bit(s)	R/W	Default	Description
31:0			Clktest_out, All $2^{\wedge}$ clktest_exp test results add up. Note: divided by $2^{\wedge}$ clktest_exp and get average clock core period length measured by 50ps delay cells.

**Table 12-362 SD\_EMMC\_EYETEST\_LOG 0x2c**

Bit(s)	R/W	Default	Description
31			eyetest_done, Test done
30:0			eyetest_times, Test eye for $2^{\wedge}$ eyetest_exp

**Table 12-363 SD\_EMMC\_EYETEST\_OUT0 0x30**

Bit(s)	R/W	Default	Description
31:0			All $2^{\wedge}$ eyetest_exp test results "OR" together.

**Table 12-364 SD\_EMMC\_EYETEST\_OUT1 0x34**

Bit(s)	R/W	Default	Description
31:0			eyetest_out1, All $2^{\wedge}$ eyetest_exp test results "OR" together. Total eyeout [62:0] = {Eyetest_out1[30:0], eyetest_out0[31:0]} EYEtest output changed to 64 bits after TXLX and A113 {eyetest_out1, eyetest_out0}

Table 12-365 SD\_EMMC\_INTF3 0x38

Bit(s)	R/W	Default	Description
26			Eyetest_sel, 0 : select core clock as eyetest clock. 1 : select DS after delay line as eyetest clock. Eyetest point is DS after delay line.
25:23			NAND_EDO, NAND Async interface EDO position after RE rising edge. [not for SD_eMMC]
22			Sd_intf3, Using SD interface 3
21:18			Ds_sht_exp, 0: DS shift setting never expires, always using the DS_sht_m as shift length. None-zero: 2 <sup>ds_sht_exp</sup> after 2 <sup>ds_sht_exp</sup> "ms", the setting expired The internal FSM will automatically change the Ds shift setting.
17:12			Ds_sht_m, Shift DS by number of 50ps delay cells. If using auto FSM mode, it is the initial value.
11			Eyetest_on, 1: Turn on eye test After eyetest_done, set this bit to 0 to reset eyetest internal registers.
10:6			Eyetest_exp, Repeat the eye test for 2 <sup>eyetest_exp</sup> times, Or the test results together, report the final results.
5			Clktest_on_m, Manual turn on clock test
4:0			Clktest_exp, Repeat the clock test for 2 <sup>clktest_exp</sup> times. Add the clock length together, report the sum.

Table 12-366 SD\_EMMC\_START 0x40

Bit(s)	R/W	Default	Description
31:2	R/W	0	Desc_addr[31:2]: Descriptor address, the last 2 Bits are 0, SRAM: 4 bytes aligned, the valid address range is from 0x200~0x3ff DDR: 8 bytes aligned the valid address is anywhere in DDR, the length of chain is unlimited. Desc_addr = ADDR>>2.
1	R/W	0	Desc_busy: Start/Stop 1: Start command chain execution process. 0: Stop Write 1 to this register starts execution. Write 0 to this register stops execution.
0	R/W	0	Desc_int: SRAM/DDR 1: Read descriptor from internal SRAM, limited to 32 descriptors. 0: Read descriptor from external DDR

Table 12-367 SD\_EMMC\_CFG 0x44

Bit(s)	R/W	Default	Description
31:28	R/W	0	Cfg_ip_txd_adj: Data 1 interrupt, when in TXD mode, the data 1 irq is a input signal, the round trip delay is uncertain factor, change this cfg to compensate the delay.
27	R/W	0	Cfg_err_abort: 1: abort current read/write and issue IRQ 0: continue on current read/write blocks.
26	R/W	0	Cfg_irq_ds: 1: Use DS pin as SDIO IRQ input, 0: Use DAT1 pin as SDIO IRQ input.
25	R/W	0	Cfg_txd_retry: When TXD CRC error, host sends the block again. The total number of retries of one descriptor is limited to 15, after 15 retries, the TXD_err is set to high.

Bit(s)	R/W	Default	Description
24	R/W	0	Cfg_txd_add_err: TXD add error test. Test feature, should not be used in normal condition. It will inverted the first CRC Bits of the 3rd block. Block index starts from 0, 1, 2, ...
23	R/W	0	Cfg_auto_clk: SD/eMMC Clock Control 1: when BUS is idle and no descriptor is available, automatically turn off clock, to save power. 0: whenever core clock is on the SD/eMMC clock is ON, it is still on/off during read data from SD/eMMC.
22	R/W	0	Cfg_stop_clk: SD/eMMC Clock Control 1: no clock for external SD/eMMC, used in voltage switch. 0: normal clock, the clock is automatically on/off during reading mode to back off reading in case of DDR slow response.
21	R/W	0	Cfg_cmd_low: Hold CMD as output Low eMMC boot mode.
20	R/W	0	Reserved
19	R/W	0	Cfg_ignore_owner: Use this descriptor even if its owner bit is "0".
18	R/W	0	Cfg_sdclk_always_on: 1: SD/eMMC clock is always ON 0: SD/eMMC clock is controlled by host. WARNING: Set SD/eMMC clock to always ON, host may lose data when DDR is slow.
17	R/W	0	Cfg_blk_gap_ip: 1: Enable SDIO data block gap interrupt period 0: Disabled.
16	R/W	0	Cfg_out_fall: DDR mode only The command and TXD start from rising edge. Set 1 to start from falling edge.
15:12	R/W	0	Cfg_rc_cc: Wait response-command, command-command gap before next command, 2cfg_rc_cc core clock cycles.
11:8	R/W	0	Cfg_resp_timeout: Wait response till 2cfg_resp_timeout core clock cycles. Maximum 32768 core cycles.
7:4	R/W	0	Cfg_bl_len: Block length 2cfg_bl_len, because internal buffer size is limited to 512 bytes, the cfg_bl_len <=9.
3	R/W	0	Cfg_dc_ugt: 1: DDR access urgent 0: DDR access normal
2	R/W	0	Cfg_ddr: 1: DDR mode 0: SDR mode
1:0	R/W	0	Cfg_bus_width: 0: 1 bit 1: 4 Bits 2: 8 Bits 3: 2 Bits (not supported)

Table 12-368 SD\_EMMC\_STATUS 0x48

Bit(s)	R/W	Default	Description
31	R		Core_busy: 1: core is busy, desc_busy or sd_emmc_irq or bus_fsm is not idle. 0: core is idle.
30	R		Desc_busy: 1: Desc input process is busy, more descriptors in chain. 0: no more descriptor in chain or desc_err.
29:26	R		Bus_fsm: BUS fsm
25	R		DS: Input data strobe
24	R		CMD_i: Input response signal
23:16	R		DAT_i: Input data signals

Bit(s)	R/W	Default	Description
15	R/W		IRQ_sdio: SDIO device uses DAT[1] to request IRQ
14	R/W		Resp_status: When resp_num is set to 1, the resp_addr is the response status IRQ enable Bits, if there is an error.
13	R/W		End_of_Chain: End of Chain IRQ, Normal IRQ
12	R/W		Desc_timeout: Descriptor execution time over time limit. The timeout limit is set by descriptor itself. Consider the multiple block read/write, set the proper timeout limits.
11	R/W		Resp_timeout: No response received before time limit. The timeout limit is set by cfg_resp_timeout.
10	R/W		Resp_err: Response CRC error
9	R/W		Desc_err: SD/eMMC controller doesn't own descriptor. The owner bit is "0", set cfg_ignore_owner to ignore this error.
8	R/W		Txd_err: TX data CRC error, For multiple block write, any one of blocks CRC error.
7:0	R/W		Rxd_err: RX data CRC error per wire, for multiple block read, the CRC errors are Ored together.

Table 12-369 SD\_EMMC\_IRQ\_EN 0x4c

Bit(s)	R/W	Default	Description
31:18			unused
17			Cfg_cmd_setup, 1: improve CMD setup time by half SD_CLK cycle. 0: CMD is two cycle aligned with DATA.
16	R/W	0	Cfg_secure: Data read/write with crypto DES
15	R/W	0	en_IRQ_sdio: Enable sdio interrupt.
14	R/W	0	En_resp_status: Response status error.
13	R/W	0	en_End_of_Chain: End of Chain IRQ
12	R/W	0	en_Desc_timeout: Descriptor execution time over time limit.
11	R/W	0	en_Resp_timeout: No response received before time limit.
10	R/W	0	en_Resp_err: Response CRC error
9	R/W	0	en_Desc_err: SD/eMMC controller doesn't own descriptor.
8	R/W	0	En_txd_err: TX data CRC error
7:0	R/W	0	en_Rxd_err: RX data CRC error per wire.

Table 12-370 Descriptor\_REG0 0x50

Bit(s)	R/W	Default	Description
31:0	R/W		SD_EMMC_CMD_CFG APB read wait Same as descriptor first word, resp_num = 1, response saved back into descriptor only. Read from this APB will hold APB bus



**Table 12-371 Descriptor\_REG1 0x54**

Bit(s)	R/W	Default	Description
31:0	R/W		SD_EMMC_CMD_ARG APB write start Same as descriptor second word. Write to this APB address starts execution. If the current desc is busy, it will be executed after current descriptor is done.

**Table 12-372 Descriptor\_REG2 0x58**

Bit(s)	R/W	Default	Description
31:0	R/W		SD_EMMC_CMD_DAT: Same as descriptor third word, 32 Bits data.

**Table 12-373 Descriptor\_REG3 0x5c**

Bit(s)	R/W	Default	Description
31:0	R/W		SD_EMMC_CMD_RSP: Write: response status IRQ enable Bits. Read: Response Bit 31:0

**Table 12-374 Descriptor\_REG4 0x60**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CMD_RSP1: Response bit 63:32

**Table 12-375 Descriptor\_REG5 0x64**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CMD_RSP2: Response bit 95:64

**Table 12-376 Descriptor\_REG6 0x68**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CMD_RSP3: Response bit 127:96

**Table 12-377 Descriptor\_REG7 0x6c**

Bit(s)	R/W	Default	Description
31:0			Reserved

**Table 12-378 Current\_Next\_Descriptor\_REG0 0x70**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CURR_CFG: Current descriptor under execution.

**Table 12-379 Current\_Next\_Descriptor\_REG1 0x74**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CURR_ARG

**Table 12-380 Current\_Next\_Descriptor\_REG2 0x78**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CURR_DAT

**Table 12-381 Current\_Next\_Descriptor\_REG3 0x7c**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CURR_RSP

**Table 12-382 Current\_Next\_Descriptor\_REG4 0x80**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_NEXT_CFG: Next descriptor waiting for execution, already read out from SRAM or DDR, can't be changed.

**Table 12-383 Current\_Next\_Descriptor\_REG5 0x84**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_NEXT_ARG

**Table 12-384 Current\_Next\_Descriptor\_REG6 0x88**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_NEXT_DAT

**Table 12-385 Current\_Next\_Descriptor\_REG7 0x8c**

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_NEXT_RSP

**Table 12-386 SD\_EMMC\_RXD 0x90**

Bit(s)	R/W	Default	Description
31:25	R		Unused
24:16	R		Data_blk: Rxd Blocks received from BUS Txd blocks received from DDR.
15:10			unused
9:0	R		Data_cnt: Rxd words received from BUS. Txd words received from DDR.

**Table 12-387 SD\_EMMC\_TXD 0x94**

Bit(s)	R/W	Default	Description
31:25			Unused
24:16	R		Txd_blk: Txd BUS block counter
15			unused
14:0	R		Txd_cnt: Txd BUS cycle counter

## 12.3 Serial Peripheral Interface Communication Controller

### 12.3.1 Overview

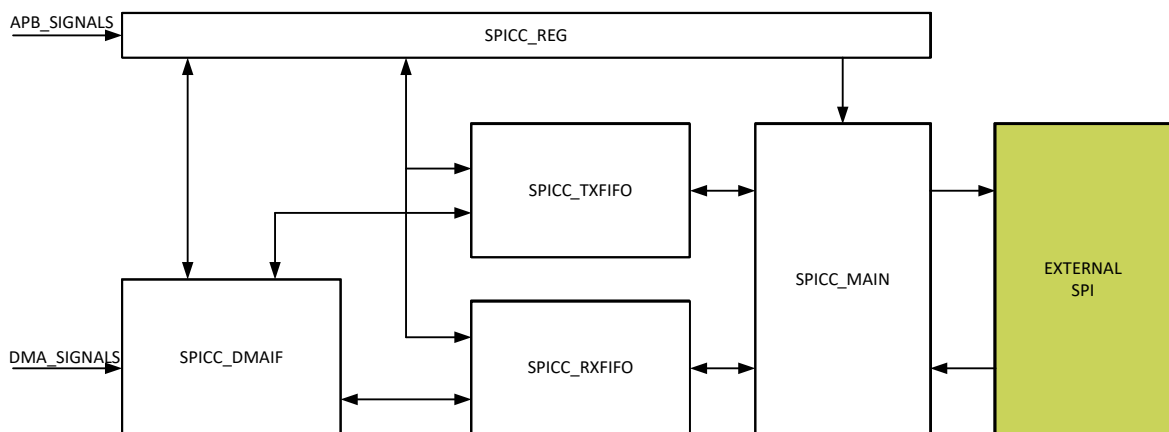
SPI Communication Controller is designed for connecting general SPI protocol compatible module. This controller allows rapid data communication with less software interrupts than conventional serial communications.

There are 5 sub-modules in spi controller, i.e. spicc\_reg, spicc\_dmaif, spicc\_txfifo, spicc\_rxfifo, and spicc\_main.

Transmitting and receiving are using different channel, that means they have different buffer.

- spicc\_reg is driven by host cpu, and spicc\_reg is responsible for configuring other modules.
- spicc\_dmaif is responsible for dealing with DMA transfer.
- spicc\_txfifo contains a transmission FIFO.
- spicc\_rxfifo contains a receiving FIFO.
- spicc\_main is responsible for main control of basic spi operation.

**Figure 12-5 SPICC**



T02FC29



**Note**

- spicc\_reg is driven by host cpu, and spicc\_reg is responsible for configuring other modules.
- spicc\_dmaif is responsible for dealing with DMA operators.
- spicc\_txfifo contains a transmission FIFO.
- spicc\_rxfifo contains a receiving FIFO.
- spicc\_main is responsible for main control of basic spi operation.

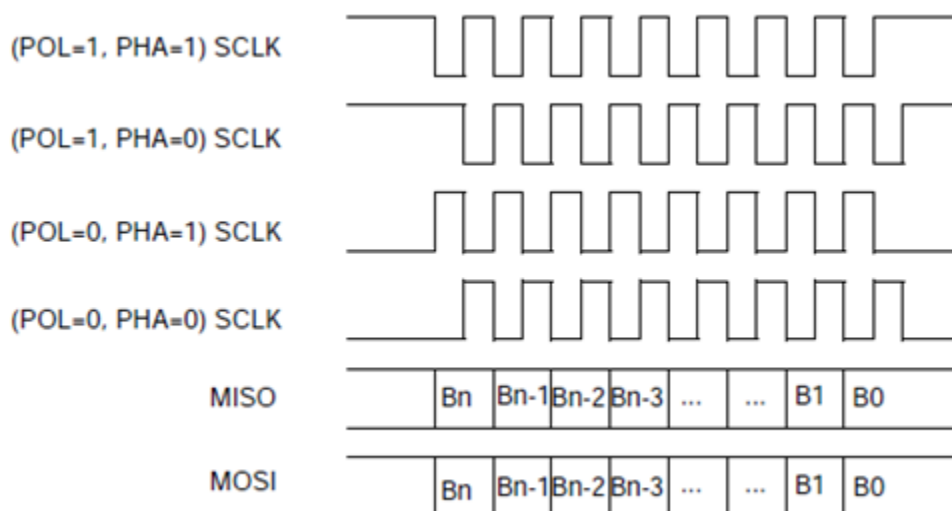
Here are SPI External Signals:

**Table 12-388 SPI External Signals**

Signal Name	I/O	Description
spicc_sclk	IO	SCLK, SPI Clock
spicc_miso	IO	MISO, Master Input Slave Output
spicc_mosi	IO	MOSI, Master Output Slave In
spicc_ss[3:0]	IO	SS, SPI chip Select, Supports up to 4 slaves.
Spicc_rdy_i	I	RDY input, Data Ready Input

And here is SPI Generic Timing:

**Figure 12-6 SPI Generic Timing**



**12.3.2 Features**

- Full-duplex synchronous serial interface
- Master/Slave configurable
- Four chip selects to support multiple peripherals
- Transfer continuation function allows unlimited length data transfers
- 64-bit wide by 16-entry FIFO for both transmit and receive data
- Polarity and phase of the Chip Select (SS) and SPI Clock (SCLK) are configurable

- Both PIO(Programming In/Out interface) and DMA(Direct Memory Access interface) supported

### 12.3.3 Function

The Serial Peripheral Interface Communication Controller(SPICC) module allows rapid data communication with fewer software interrupts than conventional serial communications. The SPICC module contains one 16x64 receive buffer (RXFIFO) and one 16 x 64 transmit buffer (TXFIFO).

The following are two SPICC modes of operation:

- **Master Mode:** When the SPICC module is configured as a master, it uses a serial link to transfer data between the SPICC and an external device. A chip-enable signal and a clock signal are used to transfer data between these two devices. If the external device is a transmit-only device, the SPICC master's output port can be ignored and used for other purposes. To use the internal TXFIFO and RXFIFO, two auxiliary output signals, SS and SPI\_RDY, are used for data transfer rate control. The user can also program the sample period control register to a fixed data transfer rate.
- **Slave Mode:** When the SPICC module is configured as a slave, the user can configure the SPICC Control register to match the external SPI master's timing. In this configuration, SS becomes an input signal, and is used to control data transfers through the Shift register, as well as to load/store the data FIFO.

### 12.3.4 Register Description

There are 6 sets of SPICC registers which are SPICC0, SPICC1, ... and SPICC5. The following list shows register addresses.

- SPICC0\_RXDATA 0xfe050000
- SPICC0\_TXDATA 0xfe050004
- SPICC0\_CONREG 0xfe050008
- SPICC0\_INTREG 0xfe05000c
- SPICC0\_DMAREG 0xfe050010
- SPICC0\_STATREG 0xfe050014
- SPICC0\_PERIODREG 0xfe050018
- SPICC0\_TESTREG 0xfe05001c
- SPICC0\_DRADDR 0xfe050020
- SPICC0\_DWADDR 0xfe050024
- SPICC0\_LD\_CNTL0 0xfe050028
- SPICC0\_LD\_CNTL1 0xfe05002c
- SPICC0\_LD\_RADDR 0xfe050030
- SPICC0\_LD\_WADDR 0xfe050034
- SPICC0\_ENHANCE\_CNTL 0xfe050038
- SPICC0\_ENHANCE\_CNTL1 0xfe05003c
- SPICC0\_ENHANCE\_CNTL2 0xfe050040
- SPICC1\_RXDATA 0xfe052000
- SPICC1\_TXDATA 0xfe052004
- SPICC1\_CONREG 0xfe052008
- SPICC1\_INTREG 0xfe05200c
- SPICC1\_DMAREG 0xfe052010
- SPICC1\_STATREG 0xfe052014
- SPICC1\_PERIODREG 0xfe052018

- SPICC1\_TESTREG 0xfe05201c
- SPICC1\_DRADDR 0xfe052020
- SPICC1\_DWADDR 0xfe052024
- SPICC1\_LD\_CNTL0 0xfe052028
- SPICC1\_LD\_CNTL1 0xfe05202c
- SPICC1\_LD\_RADDR 0xfe052030
- SPICC1\_LD\_WADDR 0xfe052034
- SPICC1\_ENHANCE\_CNTL 0xfe052038
- SPICC1\_ENHANCE\_CNTL1 0xfe05203c
- SPICC1\_ENHANCE\_CNTL2 0xfe052040
- SPICC2\_RXDATA 0xfe054000
- SPICC2\_TXDATA 0xfe054004
- SPICC2\_CONREG 0xfe054008
- SPICC2\_INTREG 0xfe05400c
- SPICC2\_DMAREG 0xfe054010
- SPICC2\_STATREG 0xfe054014
- SPICC2\_PERIODREG 0xfe054018
- SPICC2\_TESTREG 0xfe05401c
- SPICC2\_DRADDR 0xfe054020
- SPICC2\_DWADDR 0xfe054024
- SPICC2\_LD\_CNTL0 0xfe054028
- SPICC2\_LD\_CNTL1 0xfe05402c
- SPICC2\_LD\_RADDR 0xfe054030
- SPICC2\_LD\_WADDR 0xfe054034
- SPICC2\_ENHANCE\_CNTL 0xfe054038
- SPICC2\_ENHANCE\_CNTL1 0xfe05403c
- SPICC2\_ENHANCE\_CNTL2 0xfe054040
- SPICC3\_RXDATA 0xfe04a000
- SPICC3\_TXDATA 0xfe04a004
- SPICC3\_CONREG 0xfe04a008
- SPICC3\_INTREG 0xfe04a00c
- SPICC3\_DMAREG 0xfe04a010
- SPICC3\_STATREG 0xfe04a014
- SPICC3\_PERIODREG 0xfe04a018
- SPICC3\_TESTREG 0xfe04a01c
- SPICC3\_DRADDR 0xfe04a020
- SPICC3\_DWADDR 0xfe04a024
- SPICC3\_LD\_CNTL0 0xfe04a028
- SPICC3\_LD\_CNTL1 0xfe04a02c
- SPICC3\_LD\_RADDR 0xfe04a030
- SPICC3\_LD\_WADDR 0xfe04a034
- SPICC3\_ENHANCE\_CNTL 0xfe04a038
- SPICC3\_ENHANCE\_CNTL1 0xfe04a03c
- SPICC3\_ENHANCE\_CNTL2 0xfe04a040

- SPICC4\_RXDATA 0xfe04c000
- SPICC4\_TXDATA 0xfe04c004
- SPICC4\_CONREG 0xfe04c008
- SPICC4\_INTREG 0xfe04c00c
- SPICC4\_DMAREG 0xfe04c010
- SPICC4\_STATREG 0xfe04c014
- SPICC4\_PERIODREG 0xfe04c018
- SPICC4\_TESTREG 0xfe04c01c
- SPICC4\_DRADDR 0xfe04c020
- SPICC4\_DWADDR 0xfe04c024
- SPICC4\_LD\_CNTL0 0xfe04c028
- SPICC4\_LD\_CNTL1 0xfe04c02c
- SPICC4\_LD\_RADDR 0xfe04c030
- SPICC4\_LD\_WADDR 0xfe04c034
- SPICC4\_ENHANCE\_CNTL 0xfe04c038
- SPICC4\_ENHANCE\_CNTL1 0xfe04c03c
- SPICC4\_ENHANCE\_CNTL2 0xfe04c040
- SPICC5\_RXDATA 0xfe04e000
- SPICC5\_TXDATA 0xfe04e004
- SPICC5\_CONREG 0xfe04e008
- SPICC5\_INTREG 0xfe04e00c
- SPICC5\_DMAREG 0xfe04e010
- SPICC5\_STATREG 0xfe04e014
- SPICC5\_PERIODREG 0xfe04e018
- SPICC5\_TESTREG 0xfe04e01c
- SPICC5\_DRADDR 0xfe04e020
- SPICC5\_DWADDR 0xfe04e024
- SPICC5\_LD\_CNTL0 0xfe04e028
- SPICC5\_LD\_CNTL1 0xfe04e02c
- SPICC5\_LD\_RADDR 0xfe04e030
- SPICC5\_LD\_WADDR 0xfe04e034
- SPICC5\_ENHANCE\_CNTL 0xfe04e038
- SPICC5\_ENHANCE\_CNTL1 0xfe04e03c
- SPICC5\_ENHANCE\_CNTL2 0xfe04e040

SPICC0 to SPICC5 share the same registers description, the value of "n" in the name of each register is 0~5.

**Table 12-389 SPICCN\_RXDATA**

Bits	R/W	Defaults	Description
31:0	R	0	Rx Data

Note1: when PIO mode, programmer can get data from this register.

Table 12-390 SPICCN\_TXDATA

Bits	R/W	Defaults	Description
31:0	W	0	Tx Data

Note1: when PIO mode, programmer need send data to this register.

Table 12-391 SPICCN\_CONREG

Bits	R/W	Defaults	Description
31:19	RW	0	[13]burst_length ([5:0]bit number of one word/package, [12:6]burst length-1)
18:16	RW	0	[3]data_rate (sclk will be divided by system clock with equation: $2^{(data\_rate+2)}$ , Example: if system clock = 128MHz and data_rate=2, sclk's frequency equals 8MHz)
15:14			Reserved
13:12	RW	0	[2]chip_select (00:select ss_0, 01:select ss_1, 10:select ss_2, 11:select ss_3,)
11:10			Reserved
9:8	RW	0	[2]drctl (0:ignore RDY input, 1:Data ready using pin rdy_i's falling edge, 2:Data ready using pin rdy_i's low level, 3:reserved)
7	RW	0	[1]sspol (0:SS polarity Low active,1:High active)
6	RW	0	[1]ssctl (see details in Note1)
5	RW	0	[1]pha (clock/data phase control, see section 2.2)
4	RW	0	[1]pol (clock polarity control, see section 2.2)
3	RW	0	[1]smc (start mode control, see Note2)
2	RW	0	[1]xch(exchange bit, ATTN:will automatically cleared when burst finished, see Note3)
1	RW	0	[1]mode (0:slave,1:master)
0	RW	0	[1]en (0:spicc disable,1:enable)

Note1: In one burst of master mode, if ssctl ==1, ss will output 1 between each spi transition. And if ssctl ==0, ss will output 0.

Note2: smc is for start mode control. If smc ==0, burst will start when xch is set to 1'b1; if smc==1, burst will start when txfifo is not empty.



Note3: setting xch will issue a burst when smc==0, and this bit will be self cleared after burst is finished.

**Table 12-392 SPICCN\_INTREG**

Bits	R/W	Defaults	Description
31:8			Reserved
7	RW	0	[ 1]tcen(transfer completed erilog enable)
5	RW	0	[ 1]rfen(rxfifo full erilog enable)
3	RW	0	[ 1]rren(rxfifo ready erilog enable)
2	RW	0	[ 1]tfen(txfifo full erilog enable)
0	RW	0	[ 1]teen(txfifo empty erilog enable)

**Table 12-393 SPICCN\_DMAREG**

Bits	R/W	Defaults	Description
31:26	RW	0	[ 6]DMA Burst Number
25:20	RW	0	[ 6]DMA Thread ID
19	RW	0	[ 1]DMA Urgent
18:15	RW	0x7	[ 4]Number in one Write request burst(0:1,1:2...)
14:11	RW	0x7	[ 4]Number in one Read request burst(0:1,1:2...)
10:6	RW	0x8	[ 5]RxFIFO threshold(RxFIFO's count>thres, will request write)
5:1	RW	0	[ 5]TxFIFO threshold(TxFIFO's count
0	RW	0	[ 1]DMA Enable

**Table 12-394 SPICCN\_STATREG**

Bits	R/W	Defaults	Description
31:8			Reserved
7	RW	0	[ 1]tc(transfer completed, w1c, see Note1)
5	R	0	[ 1]rf(rxfifo full)
3	R	0	[ 1]rr(rxfifo ready)

Bits	R/W	Defaults	Description
2	R	0	[ 1]tf(txfifo full)
0	R	0	[ 1]te(txfifo empty)

**Table 12-395 SPICCN\_PERIODREG**

Bits	R/W	Defaults	Description
31:15			Reserved
14:0	RW	0	[15]period(wait cycles, see Note1)

Note1: Programmer can add wait cycles through this register if transmission rate need to be controlled.

**Table 12-396 SPICCN\_TESTREG**

Bits	R/W	Defaults	Description
31:23	RW	0	Reserved
23:22	RW	0	[ 2]ffirst(fifo soft reset)
21:16	RW	0x15	[ 6]dlyctl(delay control)
15	RW	0	[ 1]swap(data swap for reading rxfifo)
14	RW	0	[ 1]lbc(loop back control)
12:10	R	0	[ 3]smstatus(internal state machine status)
9:5	R	0	[ 5]rxcnt(internal RxFIFO counter)
4:0	R	0	[ 5]txcnt(internal TxFIFO counter)

Note1: Programmer can only use the TESTREG[9:0], rxcnt(internal RxFIFO counter) and txcnt(internal TxFIFO counter) , and other bits just for test.

**Table 12-397 SPICCN\_DRADDR**

Bits	R/W	Defaults	Description
31:0	RW	0	Read Address of DMA

**Table 12-398 SPICCN\_DWADDR**

Bits	R/W	Defaults	Description
31:0	RW	0	Write Address of DMA

**Table 12-399 SPICCN\_LD\_CNTL0**

Bits	R/W	Defaults	Description
31:9	RW	0	Reserved
8	RW	0	dma raddr/waddr load by dma_enable signal
7	RW	0	dma waddr load by vsync irq

Bits	R/W	Defaults	Description
6	RW	0	dma raddr load by vsync irq
5	RW	0	dma write counter enable
4	RW	0	dma read counter enable
3	RW	0	xch enable set by vsync irq
2	RW	0	dma enable set by vsync irq
1	RW	0	Reserved
0	RW	0	Vsync irq source select

Table 12-400 SPICCN\_LD\_CNTL1

Bits	R/W	Defaults	Description
31:16	RW	0	dma write counter
15:0	RW	0	dma read counter

Table 12-401 SPICCN\_LD\_RADDR

Bits	R/W	Defaults	Description
31:0	RW	0	shadow dma read address for load

Table 12-402 SPICCN\_LD\_WADDR

Bits	R/W	Defaults	Description
31:0	RW	0	shadow dma write address for load

Table 12-403 SPICCN\_ENHANCE\_CNTL

Bits	R/W	Defaults	Description
31:30	R	0	Reserved
29	R/W	0	main clock always on
28	R/W	0	clk-cs delay enable
27	R/W	0	cs_oen enhance enable
26	R/W	0	clk_oen enhance enable
25	R/W	0	mosi_oen enhance enable
24	R/W	0	spi clk select 0: controlled by data_rate in CONREG 1: controlled by enhance_clk_div in ENHANCE_CNTL
23:16	R/W	0	enhance_clk_div
15:0	R/W	0	clk-cs delay value

Table 12-404 SPICCN\_ENHANCE\_CNTL1

Bits	R/W	Defaults	Description
31:29	R/W	0	enhance_fclk_mosi_oen_dlyctl: mosi_oen delay control in fclk
28	R/W	0	enhance_fclk_mosi_oen_dlyctl_en: enable dlyctl
27:25	R/W	0	enhance_fclk_mosi_o_dlyctl: mosi_o delay control in fclk
24	R/W	0	enhance_fclk_mosi_o_dlyctl_en: enable dlyctl
23:21	R/W	0	enhance_fclk_miso_i_dlyctl: miso_i delay control in fclk
20	R/W	0	enhance_fclk_miso_i_dlyctl_en: enable dlyctl
19:17	R/W	0	enhance_fclk_mosi_i_dlyctl: mosi_i delay control in fclk
16	R/W	0	enhance_fclk_mosi_i_dlyctl_en: enable dlyctl
15	R/W	0	enhance_fclk_en: fclk gate enable
14	R/W	0	enhance_mosi_i_capture_en: enable, 1=select enhance capture function for mosi_i (slave mode)
9:1	R/W	0	enhance_clk_tcnt: adjust capturing timing for miso_i data (master mode). when clk_cnt=enhance_clk_tcnt, capture input data. the value of enhance_clk_tcnt must be less than the most value of clk_cnt. sclk will be divided by system clock with clk_cnt.
0	R/W	0	enhance_miso_i_capture_en: enable, 1=select enhance capture function for miso_i (master mode)

Table 12-405 SPICCN\_ENHANCE\_CNTL2

Bits	R/W	Defaults	Description
31	R/W	0	clk_cs_tt delay enable
30:16	R/W	0	clk_cs_tt delay value
15	R/W	0	clk_cs_ti delay enable
14:0	R/W	0	clk_cs_ti delay value

## 12.4 Serial Peripheral Interface Flash Controller

### 12.4.1 Overview

SPI Flash Controller is designed for connecting varied SPI Flash memory.

### 12.4.2 Features

- Support three operation modes, NOR Flash mode, Master mode, and Slave mode.
- Support read/write buffer up to 64bytes.
- Support no clock toggling during DUMMY state.
- Support hold by an external pin during a transition.
- AHB read support byte and halfword.
- Support bit-number rather than byte-number for each stage.

- Support 2/4 wire writing like fast reading
- Support both rising-edge and falling-edge for SPI slave sampling and SPI master sampling.
- Support 1 wire for SPI\_D and SPI\_Q.
- Support SPI\_CK setup and hold time by cycles
- Support 8 bit clock divider, so SPI\_CK can be low as 1/256 HCLK
- Support byte-order in a word
- Support no command state, so the command is sent/received in address state by 2/4 wires.
- Support both data input and data output in a transition. SPI\_DOUT->(SPI\_DUMMY)->SPI\_DIN

### 12.4.3 Register Description



#### Note

If the bit2 “Backward Compatible” in “SPI User Register” is 1, the registers and functions of this SPI controller are as same as Apollo SPI controller.

To use SPI Flash commands, please set the bit2 “Backward Compatible” in “SPI User Register” as “1”.

- SPI\_FLASH\_CMD 0xfe056000
- SPI\_FLASH\_ADDR 0xfe056004
- SPI\_FLASH\_CTRL 0xfe056008
- SPI\_FLASH\_CTRL1 0xfe05600c
- SPI\_FLASH\_STATUS 0xfe056010
- SPI\_FLASH\_CTRL2 0xfe056014
- SPI\_FLASH\_CLOCK 0xfe056018
- SPI\_FLASH\_USER 0xfe05601c
- SPI\_FLASH\_USER1 0xfe056020
- SPI\_FLASH\_USER2 0xfe056024
- SPI\_FLASH\_USER3 0xfe056028
- SPI\_FLASH\_USER4 0xfe05602c
- SPI\_FLASH\_SLAVE 0xfe056030
- SPI\_FLASH\_SLAVE1 0xfe056034
- SPI\_FLASH\_SLAVE2 0xfe056038
- SPI\_FLASH\_SLAVE3 0xfe05603c
- SPI\_FLASH\_C0 0xfe056040
- SPI\_FLASH\_C1 0xfe056044
- SPI\_FLASH\_C2 0xfe056048
- SPI\_FLASH\_C3 0xfe05604c
- SPI\_FLASH\_C4 0xfe056050
- SPI\_FLASH\_C5 0xfe056054
- SPI\_FLASH\_C6 0xfe056058
- SPI\_FLASH\_C7 0xfe05605c
- SPI\_FLASH\_B8 0xfe056060
- SPI\_FLASH\_B9 0xfe056064
- SPI\_FLASH\_B10 0xfe056068

- SPI\_FLASH\_B11 0xfe05606c
- SPI\_FLASH\_B12 0xfe056070
- SPI\_FLASH\_B13 0xfe056074
- SPI\_FLASH\_B14 0xfe056078
- SPI\_FLASH\_B15 0xfe05607c

**Table 12-406 SPI\_FLASH\_CMD**

Bit(s)	R/W	Default	Description
31	R/W	0	READ command. 1 = read. When it becomes 0, the read command is finished. The READ command could be (0xEB, 0x6B, 0xBB, 0x3B, 0x0B, 0x03). By default is 0x0B. One read command will read erilog 32x8bits data. And saved in data cache.
30	R/W	0	WREN command. (0x06)
29	R/W	0	WRDI command. ( 0x04).
28	R/W	0	RDID command. (0x9f).
27	R/W	0	RDSR command. (0x05).
26	R/W	0	WRSR command. (0x01).
25	R/W	0	Page program command. (0xAD or 0x02).
24	R/W	0	SE command (0x20).
23	R/W	0	BE command ( 0xD8).
22	R/W	0	CE command.(0xC7).
21	R/W	0	Deep Power Down command(0xB9).
20	R/W	0	RES command. (0xAB).
19	R/W	0	HPM command.(0xA3). (Just For winbond SPI flash).
18	R/W	0	USER defined command.
17:0	R/W	0	Reserved for future.

**Table 12-407 SPI\_FLASH\_ADDR**

Bit(s)	R/W	Default	Description
31:0	R/W	0	The address[31:0] of the user command

**Table 12-408 SPI\_FLASH\_CTRL**

Bit(s)	R/W	Default	Description
31:27	R/W	0	Reserved.
26	R/W	0	Write bit order. 1 = 0, 1, 2, 3, 4, 5, 6, 7. 0 = 7, 6, 5, 4, 3, 2, 1, 0.
25	R/W	0	Read bit order. . 1 = 0, 1, 2, 3, 4, 5, 6, 7. 0 = 7, 6, 5, 4, 3, 2, 1, 0.
24	R/W	0	Fast read QIO mode.
23	R/W	0	Fast read DIO mode.
22	R/W	0	Write 2 bytes status mode. For some of winbond SPI flash, the status register is 16bits.

Bit(s)	R/W	Default	Description
21	R/W	1	SPI flash WP pin value if use SPI flash WP pin as write protection.
20	R/W	0	Fast read QOUT mode.
19	R/W	1	1 = SPI share pins with SDRAM. 0 = doesn't share.
18	R/W	0	SPI hold mode. 1=SPI controller would use SPI hold function. 0 = SPI controller won't use hold function. The SPI flash hold pin can be tie high on the board. Or SPI controller can use hold pin as QIO/QOUT mode.
17	R/W	1	1 = enable AHB request. 0 = disable AHB request when you reconfigure SPI controller or running APB bus commands.
16	R/W	0	1 =enable SST SPI Flash aai command. The APB bus PP command will send AAI command.
15	R/W	1	1 = release from Deep Power-Down command is with read electronic signature.
14	R/W	0	Fast read DOUT mode.
13	R/W	1	Fast read mode. AHB bus read requirement and APB bus read command use the command 0x0Bh.
12:0	R/W	0	Reserved for future.

Table 12-409 SPI\_FLASH\_CTRL1

Bit(s)	R/W	Default	Description
31:28	R/W	5	SPI Clock cycles for SPI flash timing requirement tCSH.
27:16	R/W	0xff	SPI Clock cycles for SPI flash timing requirement tRES.
15:0	R/W	0x0120	System clock cycles for SPI bus timer. In SPI share bus and SPI hold function mode. SPI bus timer used , if SPI use the bus for a limit time, SPI controller will diassert SPI hold pin to halt the SPI Flash, and give the bus control to SDRAM.

Table 12-410 SPI\_FLASH\_STATUS

Bit(s)	R/W	Default	Description
Bit(s)	R/W	Default	Description
31:24	R/W	0	Reserved.
23:16	R/W	0	For winbond SPI flash, this 8 bits used for DIOMode M7~M0,
15:0	R/W	0	SPI status register value. WRSR command will write this value to SPI flash status. RDSR or RES command will save the read result to this register.
When SPI controller in the slave mode, this register are the status for the SPI master to read out.			
31:0	R/W	0	In SPI Slave mode, the read status of the user command

Table 12-411 SPI\_FLASH\_CTRL2

Bit(s)	R/W	Default	Description
31:28	R/W	0	Delay cycle number of SPI_CS input in SPI slave mode or SPI_CS output in SPI master mode 0= not delay, 1 = delayed by 1 cycle of system clock, 2 = delayed by 2 cycles of system clock , ...
27:26	R/W	0	delay mode of SPI_CS input in SPI slave mode or SPI_CS output in SPI master mode 0= not latched by the edges of SPI_CK 1= latched by the falling edges of SPI_CK 2 = latched by the rising edges of SPI_CK
25:23	R/W	0	Delay cycle number of SPI Data from SPI Master to SPI Slave In SPI master mode, it is for data outputs; in SPI slave mode, it is for data inputs. 0= not delay, 1 = delayed by 1 cycle of system clock, 2 = delayed by 2 cycles of system clock , ...
22:21	R/W	0	Delay mode of SPI Data from SPI Master to SPI Slave In SPI master mode, it is for data outputs; in SPI slave mode, it is for data inputs. 0= not latched by the edges of SPI_CK 1= latched by the falling edges of SPI_CK 2 = latched by the rising edges of SPI_CK
20:18	R/W	0	Delay cycle number of SPI Data from SPI Slave to SPI Master. In SPI master mode, it is for data inputs; in SPI slave mode, it is for data outputs. 0= not delay, 1 = delayed by 1 cycle of system clock, 2 = delayed by 2 cycles of system clock , ...
17:16	R/W	0	Delay mode of SPI Data from SPI Slave to SPI Master. In SPI master mode, it is for data inputs; in SPI slave mode, it is for data outputs. 0= not latched by the edges of SPI_CK 1= latched by the falling edges of SPI_CK 2 = latched by the rising edges of SPI_CK
15:12	R/W	0	In SPI master mode, SPI_CK rising edge mode 4'b1000 = later by 1/4 cycle of SPI_CK 4'b1001 = later by 1/8 cycle of SPI_CK 4'b1010 = later by 1/16 cycle of SPI_CK 4'b1011 = later by 1/32 cycle of SPI_CK 4'b1100 = earlier by 1/4 cycle of SPI_CK 4'b1101 = earlier by 1/8 cycle of SPI_CK 4'b1110 = earlier by 1/16 cycle of SPI_CK 4'b1111 = earlier by 1/32 cycle of SPI_CK Others = Normal
11:8	R/W	0	In SPI master mode, SPI_CK falling edge mode 4'b1000 = later by 1/4 cycle of SPI_CK 4'b1001 = later by 1/8 cycle of SPI_CK 4'b1010 = later by 1/16 cycle of SPI_CK 4'b1011 = later by 1/32 cycle of SPI_CK 4'b1100 = earlier by 1/4 cycle of SPI_CK 4'b1101 = earlier by 1/8 cycle of SPI_CK 4'b1110 = earlier by 1/16 cycle of SPI_CK 4'b1111 = earlier by 1/32 cycle of SPI_CK Others = Normal
7:4	R/W	1	In master mode, SPI clock cycles for SPI hold timing.
3:0	R/W	1	In master mode, SPI clock cycles for SPI setup timing. SPI setup time and SPI hold time is used to configure how soon the controller can enable spi_cs_n after the controller get the bus and how long the controller still keep the bus after the spi_cs_n become to be high.



**Table 12-412 SPI\_FLASH\_CLOCK**

Bit(s)	R/W	Default	Description
31	R/W	1	1=SPI clock frequency is same as system clock. 0 = SPI clock frequency will use clock divider.
30:18	R/W	0	Clock counter for Pre-scale divider: 0= not pre-scale divider, 1= pre-scale divided by 2, 2= pre-scale divided by 3, .....
17:12	R/W	0	Clock counter for clock divider.
11:6	R/W	0	Clock high counter in SPI master mode. In SPI slave mode, it is for the delay counter for the rising edges of spi_ck_i
5:0	R/W	0	Clock low counter, in SPI master mode. In SPI slave mode, it is for the delay counter for the falling edges of spi_ck_i If the SPI clock frequency = sys_clock_frequency / n. Then the clock divider counter = n - 1; the clock high counter = n / 2 - 1; the clock low counter = n - 1; For example, if you want to SPI clock erilog is divided by 2 of the system clock. The clock divider counter = 1, clock high counter = 0, clock low counter = 1. For SPI clock frequency = system clock / 4. The clock divider counter = 3, clock high counter = 1, clock low counter = 3.

**Table 12-413 SPI\_FLASH\_USER**

Bit(s)	R/W	Default	Description
31	R/W	1	USER command COMMAND bit. 1 = user command includes command. 0 = no command. If some SPI slaves may support 2/4 IO at the first cycle, clear this bit.
30	R/W	0	USER command ADDRESS bit. 1 = user command includes address. 0 = no address.
29	R/W	0	USER command DUMMY bit. 1= user command includes Dummy bytes.
28	R/W	0	USER command DIN bit. 1 = user command includes data in. 0 = no data in.
27	R/W	0	USER command DO bit. 1 = user command includes data output. 0 = no data output. If both DIN and DO are valid, SPI master is firstly in data output state and then in data input state. If all of DUMMY, DO and DIN are valid, SPI master is firstly in data output state and then in dummy state, finally in data input state.
26	R/W	0	USER command dummy idle bit. 1= no SPI clock toggling in dummy state. 0= normal
25	R/W	0	USER command highpart bit for SPI_DOUT stage. It is for data-output in spi master mode and for data-input in spi slave mode. 1 = only high half part of buffer are used. 0 = low half part or the whole 64bytes are used.
24	R/W	0	USER command highpart bit for SPI_DIN stage. It is for data-input in spi master mode and for data-output in spi slave mode. 1 = only high half part of buffer are used. 0 = low half part or the whole 64bytes are used.
23	R/W	0	User command external hold bit for prep. 1 = in prep state, SPI master controller can be hold by the external pin SPI_HOLD
22	R/W	0	User command external hold bit for command. 1 = in command state, SPI master controller can be hold by the external pin SPI_HOLD
21	R/W	0	User command external hold bit for address. 1 = in address state, SPI master controller can be hold by the external pin SPI_HOLD

Bit(s)	R/W	Default	Description
20	R/W	0	User command external hold bit for dummy. 1 = in dummy state, SPI master controller can be hold by the external pin SPI_HOLD
19	R/W	0	User command external hold bit for data input. 1 = in data input state, SPI master controller can be hold by the external pin SPI_HOLD
18	R/W	0	User command external hold bit for data output. 1 = in data output state, SPI master controller can be hold by the external pin SPI_HOLD
17	R/W	1	User command external hold polarity bit. 1 = high is valid for hold, 0 = low is valid for hold.
16	R/W	0	Single DIO mode: Data output and input apply only 1 wire.
15	R/W	0	Fast write QIO mode.
14	R/W	0	Fast write DIO mode.
13	R/W	0	Fast write QOUT mode.
12	R/W	0	Fast write DOUT mode.
11	R/W	0	Write byte order. 0 = d[7:0], d[15:8], d[23:16], d[31:24]. 1 = d[31:24], d[23:16], d[15:8], d[7:0]
10	R/W	0	Read byte order. . 0 = d[7:0], d[15:8], d[23:16], d[31:24]. 1 = d[31:24], d[23:16], d[15:8], d[7:0]
9:8	R/W	0	AHB endian mode: 0= little-endian; 1= big-endian; 2~3 reserved
7	R/W	0	In SPI master mode, the clock output edge bit: 0 = SPI_CK is inverted, 1 = SPI_CK is not inverted
6	R/W	1	In SPI slave mode, the clock input edge bit: 0 = SPI_CK_I is inverted, 1 = SPI_CK_I is not inverted
5	R/W	0	SPI CS setup bit: 1 = valid in prep state
4	R/W	0	SPI CS hold bit: 1 = valid in done state
3	R/W	0	AHB-read apply the configurations of user-command, such as command value, bit-length,...
2	R/W	1	Backward Compatible: 1 = compatible to Apollo SPI This bit affect the three registers: "SPI Flash Command Register", "SPI Address Register" and "SPI Control Register"
1	R/W	0	AHB-read support 4byte address, when AHB-read apply the configurations of user-command. 1 = 4byte address, 0 = 3byte address
0	R/W	0	In SPI master mode, Enable bit for Data input during SPI_DOUT stage. 1 = enable; 0 = disable This bit shall not be used in 2/4wire or SIO. When this bit is 1, during SPI_DOUT stage, data input are stored into cacheline/ buffer from address 0, i.e., Bit24 is not controlling the start address. The data output can be specified by bit25

Table 12-414 SPI\_FLASH\_USER1

Bit(s)	R/W	Default	Description
31:26	R/W	0	USER command bit number for address state 0 = 1 bit, 1= 2 bits, ...
25:17	R/W	0	USER command bit number for data output state 0 = 1 bit, 1= 2 bits, ...

Bit(s)	R/W	Default	Description
16:8	R/W	0	USER command bit number for data input state 0 = 1 bit, 1= 2 bits, ...
7:0	R/W	0	USER command cycle number for dummy state 0 = 1 cycle, 1= 2 cycles, ...

Table 12-415 SPI\_FLASH\_USER2

Bit(s)	R/W	Default	Description
31:28	R/W	0	USER command bit number for command state 0 = 1 bit, 1= 2 bits, ...
27:16	R/W	0	Reserved
15:0	R/W	0	The command content of the user command

Table 12-416 SPI\_FLASH\_USER3

Bit(s)	R/W	Default	Description
31:0	R/W	0	In SPI Master mode, the address[63:32] of the user command In SPI Slave mode, the write status of the user command

Table 12-417 SPI\_FLASH\_USER4

Bit(s)	R/W	Default	Description
31	R/W	0	Pin swap when it is in DIN stage and SPI data input are 4wire. This feature is for Ubec Zigbee chips. 1 = swap between {spi_q, spi_d_i} and {spi_hold_i, spi_wp_i} 0 = normal
30	R/W	0	In SPI Master mode, CS keep active after a transition. 1 = enable; 0 = disable
29	R/W	0	Idle edge of SPI_CK 0 = low when it is idle 1 = high when it is idle
28:24	R/W	0	Reserved
23	R/W	0	In the SPI slave mode, spi_cs_i polarity: 1= high voltage is active 0= low voltage is active
22:21	R/W	0	In the SPI slave mode, spi_ck_i and spi_cs_i source pins 0=SPI_CK and SPI_CS pins, respectively 1=SPI_CS2 and SPI_CS1 pins, respectively 2=SPI_HOLD and SPI_WP pins, respectively
20	R/W	0	SPI_CS2 and SPI_CS1 pin function MUX 0= spi_ck and spi_cs in the SPI master mode 1= input pads for spi_ck_i and spi_cs_i, respectively, in the SPI slave mode
19	R/W	0	SPI_CK and SPI_CS pin function MUX 0= spi_ck and spi_cs in the SPI master mode 1= input pads for spi_ck_i and spi_cs_i, respectively, in the SPI slave mode
18:17	R/W	0	SPI_HOLD and SPI_WP pin function MUX 0= normal 1= spi_q and spi_d, respectively 2= spi_cs3 and spi_cs2, respectively

Bit(s)	R/W	Default	Description
			3= input pads for spi_ck_i and spi_cs_i, respectively, in the SPI slave mode
16	R/W	0	SPI_D and SPI_Q switch 1= SPI_D and SPI_Q pin-functions are swapped 0= normal
15:11	R/W	0	In Master mode, these are spi_ck MUX bit[4:0] for spi_cs4 (SPI_HOLD pin), spi_cs3 (SPI_WP pin), spi_cs2, spi_cs1 and spi_cs, respectively 1= this pin is spi_ck, if this pin is not idle 0= this pin is spi_cs, if this pin is not idle
10:6	R/W	0	In Master mode, these are polarity bit[4:0] for spi_cs4 (SPI_HOLD pin), spi_cs3 (SPI_WP pin), spi_cs2, spi_cs1 and spi_cs, respectively 1= high voltage is active 0= low voltage is active
5:0	R/W	0x1E	In Master mode, these are idle bit[5:0] for SPI_CK, for spi_cs4 (SPI_HOLD pin), spi_cs3 (SPI_WP pin), spi_cs2, spi_cs1 and spi_cs, respectively 1= idle, i.e., the spi_ck signal is 0 or the spi_cs is at the inactive level 0= active if SPI controller is working

Table 12-418 SPI\_FLASH\_SLAVE

Bit(s)	R/W	Default	Description
31	R/W	0	SPI controller SW reset: 1 = reset, 0 = none
30	R/W	0	SPI slave mode: 1 = slave, 0 = master
29	R/W	0	In SPI slave mode, the enable bit for the command of write-buffer-and-read-buffer 0= disable, 1=enable
28	R/W	0	In SPI slave mode, the enable bit for the command of write-status-and-read-status 0= disable, 1=enable
27	R/W	0	SPI slave command define enable 0=Apply the last 3bits of Flash commands and two extra command, please refer to In SPI slave mode, SPI-Flash-Lite Commands are supported. 1=Apply the user defined command in SPI Slave register 3
26:4	R/W	0	Reserved
11:10	R/W	0	spi_cs_i recovery mode: 0= and 1= or 2= normal 3= delayed
9:5	R/W	0	Interrupt enable for bit4:0 1= enable, 0=disable
4	R/W	0	A SPI transition is done. (whatever it is in SPI master mode or SPI slave mode)
3	R/W	0	In SPI slave mode, a status write is done
2	R/W	0	In SPI slave mode, a status read is done
1	R/W	0	In SPI slave mode, a buffer write is done
0	R/W	0	In SPI slave mode, a buffer read is done

**Table 12-419 SPI\_FLASH\_SLAVE1**

Bit(s)	R/W	Default	Description
31:27	R/W	0	In SPI slave mode, status bit number 0 = 1 bit, 1= 2 bits, ...
26	R/W	0	In SPI slave mode, status fast read/write enable bit: 1 = enable, 0 = disable
25	R/W	0	In SPI slave mode, status read back enable bit: 1 = reading status is written status in SPI User register 3, 0 = reading status is in SPI Status register.
24:16	R/W	0	In SPI slave mode, buffer bit number 0 = 1 bit, 1= 2 bits, ...
15:10	R/W	0	In SPI slave mode, address bit number for reading buffer 0 = 1 bit, 1= 2 bits, ...
9:4	R/W	0	In SPI slave mode, address bit number for writing buffer 0 = 1 bit, 1= 2 bits, ...
3	R/W	0	In SPI slave mode, dummy enable bit for writing status 1=enable, 0=disable
2	R/W	0	In SPI slave mode, Dummy enable bit for reading status 1=enable, 0=disable
1	R/W	0	In SPI slave mode, Dummy enable bit for writing buffer 1=enable, 0=disable
0	R/W	0	In SPI slave mode, Dummy enable bit for reading buffer 1=enable, 0=disable

**Table 12-420 SPI\_FLASH\_SLAVE2**

Bit(s)	R/W	Default	Description
31:24	R/W	0	In SPI slave mode, Dummy cycle number for writing buffer 0 = 1 cycle, 1= 2 cycles, ...
23:16	R/W	0	In SPI slave mode, Dummy cycle number for reading buffer 0 = 1 cycle, 1= 2 cycles, ...
15:8	R/W	0	In SPI slave mode, Dummy cycle number for writing status 0 = 1 cycle, 1= 2 cycles, ...
7:0	R/W	0	In SPI slave mode, Dummy cycle number for reading status 0 = 1 cycle, 1= 2 cycles, ...

**Table 12-421 SPI\_FLASH\_SLAVE3**

Bit(s)	R/W	Default	Description
31:24	R/W	0	In SPI slave mode, Command value for writing status, when bit27 "SPI slave command define enable" in SPI Slave register is 1
23:16	R/W	0	In SPI slave mode, Command value for reading status, when bit27 "SPI slave command define enable" in SPI Slave register is 1
15:8	R/W	0	In SPI slave mode, Command value for writing buffer, when bit27 "SPI slave command define enable" in SPI Slave register is 1
7:0	R/W	0	In SPI slave mode, Command value for reading buffer, when bit27 "SPI slave command define enable" in SPI Slave register is 1

**Table 12-422 SPI\_FLASH\_C0~7**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Cache line Word 0~7. Cache is used to read data both for AHB or APB read command. Cache is also used for APB page programming etc.

**Table 12-423 SPI\_FLASH\_B8~15**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Buffer Word 8. Buffer is used to read/write data only for APB read/write user commands.

## 13 I/O Interface

---

### 13.1 Universal Serial Bus

#### 13.1.1 Overview

The chip integrates one USB 2.0 port which supports USB OTG, one USB3.0 DRD port and one PCIe 2.0 (Root Complex) interface up to 5Gbps.

#### 13.1.2 Features

The USB2.0 OTG controller features

- Support for the following speeds: High-Speed (HS, 480-Mbps), Full-Speed (FS, 12-Mbps) and Low-Speed (LS, 1.5-Mbps) modes
- Multiple DMA/non DMA mode access support on the application side
- Supports up to 16 bidirectional endpoints, including control endpoint 0.
- Supports Session Request Protocol (SRP) and Host Negotiation Protocol (HNP)
- Supports ACA ID detector

The USB2.0 Host controller features:

- Support for the following speeds: High-Speed (HS, 480-Mbps), Full-Speed (FS, 12-Mbps) and Low-Speed (LS, 1.5-Mbps) modes
- Multiple DMA/non DMA mode access support on the application side

The USB2.0 PHY features:

- Support for the following speeds: High-Speed (HS, 480-Mbps), Full-Speed (FS, 12-Mbps) and Low-Speed (LS, 1.5-Mbps) modes

The USB3.0 Host controller features:

- Support for the following speed: Super-Speed, High-Speed, Full-Speed and Low-Speed
- Compliant with the xHCI specification

The USB3.0 SS PHY features:

- 5-Gbps SuperSpeed data transmission rate over 3-m USB 3 cable
- Integrated PHY includes transmitter, receiver, spread spectrum clock (SSC) generation, PLL, digital core, and electrostatic discharge (ESD) protection circuits
- Supports legacy Half-rate mode for power-saving

#### 13.1.3 Register Description

Base address:

PHY 20: 0xfe03c000.

PHY 21: 0xfe03e000.

For the following registers, each register's final address= base address + offset \*4.

**Table 13-1 reg32\_00 0x00**

Bit(s)	R/W	Default	Description
31:15	R/W	0	reserved
14	R/W	1	reset_FS_LS_Clock_Divider
13	R/W	1	reset_HS_CDR
12	R/W	1	reset_FS_LS_CDR
11:10	R/W	0	reg32_00_11_10_reserved
9:8	R/W	0	reserved
7	R/W	0	usb2_cal_pd
6:4	R/W	0	reserved
3	R/W	0	usb2_rx_fsls_pd
2:1	R/W	0	reserved
0	R/W	0	usb2_tx_hs_pd

**Table 13-2 reg32\_01 0x04**

Bit(s)	R/W	Default	Description
31:24	R/W	0	TBD_3
23:16	R/W	0	TBD_2
15:10	R/W	0	TBD_1
9:2	R/W	0	tx_cfg[7:0]
1:0	R/W	0x0	reserved

**Table 13-3 reg32\_02 0x08**

Bit(s)	R/W	Default	Description
31	RO	0	Phy_status
30	RO	0	cal_en_flag
29:27	RO	0	reg32_02_29_27_reserved
26	RO	0	HS_Disconnect_Status
25	RO	0	HS_Squelch_Status
24	RO	0	PRBS_Sync_Out
23:16	RO	0	Calibration_code_Value_23_16
15:8	RO	0	Calibration_code_Value_15_8
7:0	RO	0	Calibration_code_Value_7_0



**Table 13-4 reg32\_03 0x0c**

Bit(s)	R/W	Default	Description
31:24	R/W	0	TBD_7
23:16	R/W	0	TBD_6
15:8	R/W	0	TBD_5
7:4	R/W	0x2	TBD_4
4:2	R/W	0x1	usb2_disc_trim
2:0	R/W	0	usb2_squelch_trim

**Table 13-5 reg32\_04 0x10**

Bit(s)	R/W	Default	Description
31:30	R/W	0	i_c2l_bias_trim_3_2
29:28	R/W	0	i_c2l_bias_trim_1_0
27	R/W	0	TEST_Bypass_mode_enable
26	RO	0	i_c2l_cal_done
25	R/W	0	i_c2l_cal_reset_n
24	R/W	0	i_c2l_cal_en
23:16	R/W	0	Calibration_code_Value_23_16
15:8	R/W	0	Calibration_code_Value_15_8
7:0	R/W	0	Calibration_code_Value_7_0

**Table 13-6 reg32\_05 0x14**

Bit(s)	R/W	Default	Description
31:24	R/W	0	i_c2l_obs_7_0
23:22	R/W	0	reg32_05_23_22_reserved
21:20	R/W	0	i_c2l_pll_perfcfg_21_20
19	R/W	0x1	i_c2l_pll_perfcfg_19
18:16	R/W	0	i_c2l_pll_perfcfg_18_16
15:12	R/W	0	i_c2l_pll_perfcfg_15_12
11:10	R/W	0	i_c2l_pll_perfcfg_11_10
9	R/W	0x1	i_c2l_pll_perfcfg_9 (pll_reset)
8	R/W	0x1	i_c2l_pll_perfcfg_8
7:4	R/W	0x7	i_c2l_pll_perfcfg_7_4
3:2	R/W	0x3	i_c2l_pll_perfcfg_3_2
1:0	R/W	0	i_c2l_pll_perfcfg_1_0

**Table 13-7 reg32\_06 0x18**

Bit(s)	R/W	Default	Description
31	R/W	0x1	hub_extra_bit_cntr
30:24	R/W	0	cntr_timeout
23	R/W	0	Internal_loopback
22	R/W	0	reg32_06_22_reserved
21	R/W	0	PCS_Reset_Transmit_State_machine
20	R/W	0	PCS_Reset_Receive_State_machine
19:16	R/W	0xf	fsls_farend_device_disconnect_micro_second_count_11_8
15:12	R/W	0xa	reg32_06_15_12_reserved
11:8	R/W	0	bypass_disc_cntr_3_0
7:0	R/W	0x17	PCS_microsecond_timer_done_count_value_7_0

**Table 13-8 reg32\_07 0x1c**

Bit(s)	R/W	Default	Description
31:24	RO	0	Prbs_Error_count
23:21	R/W	0	reg32_07_23_21_reserved
20:17	R/W	0xf	RX_ERROR_Turn_Around_Timer_Count
16	R/W	0	acceptable_bit_drops
15	R/W	0	host_tristate
14:12	R/W	0x4	fs_ls_minimum_count
11:8	R/W	0xf	cntr_done_value_7_4
7:4	R/W	0xf	cntr_done_value_3_0
3:0	R/W	0	HS_CDR_internal_tap_select

**Table 13-9 reg32\_08 0x20**

Bit(s)	R/W	Default	Description
31:24	R/W	0	Custom_Pattern_2
23:16	R/W	0	Custom_Pattern_1
15:8	R/W	0x4	Custom_Pattern_0
7	R/W	0	Enable_RX_ERROR_Timeout_Mode
6	R/W	0	reset_us_timer
5	R/W	0	PRBS_ERROR_Insert
4	R/W	0	PRBS_comparison_enable
3	R/W	0	PRBS_Enable
2:0	R/W	0	pattern

**Table 13-10 reg32\_09 0x24**

Bit(s)	R/W	Default	Description
31:24	R/W	0	Custom_Pattern_6
23:16	R/W	0	Custom_Pattern_5
15:8	R/W	0	Custom_Pattern_4
7:0	R/W	0	Custom_Pattern_3

**Table 13-11 reg32\_10 0x28**

Bit(s)	R/W	Default	Description
31:24	R/W	0	Custom_Pattern_10
23:16	R/W	0	Custom_Pattern_9
15:8	R/W	0	Custom_Pattern_8
7:0	R/W	0	Custom_Pattern_7

**Table 13-12 reg32\_11 0x2c**

Bit(s)	R/W	Default	Description
31:24	R/W	0	Custom_Pattern_14
23:16	R/W	0	Custom_Pattern_13
15:8	R/W	0	Custom_Pattern_12
7:0	R/W	0	Custom_Pattern_11

**Table 13-13 reg32\_12 0x30**

Bit(s)	R/W	Default	Description
31:24	R/W	0	Custom_Pattern_18
23:16	R/W	0	Custom_Pattern_17
15:8	R/W	0	Custom_Pattern_16
7:0	R/W	0	Custom_Pattern_15

**Table 13-14 reg32\_13 0x34**

Bit(s)	R/W	Default	Description
31	R/W	0	reg32_13_31_reserved
30	R/W	0	bypass_reg[6]:i_c2l_fsls_rx_en
29	R/W	0	bypass_reg[5]:i_c2l_hs_rx_en
28	R/W	0	bypass_reg[4]:i_c2l_fs_oe
27	R/W	0	bypass_reg[3]:i_c2l_hs_oe

Bit(s)	R/W	Default	Description
26	R/W	0	bypass_reg[2]:i_c2l_ls_en
25	R/W	0	bypass_reg[1]:i_c2l_fs_en
24	R/W	0	bypass_reg[0]:i_c2l_hs_en
23	R/W	0	Bypass_Host_Disconnect_Enable
22	R/W	0	Bypass_Host_Disconnect_Value
21	R/W	0	Clear_Hold_HS_disconnect
20:16	R/W	0x8	minimum_count_for_sync_detection
15	R/W	0	Update_PMA_signals
14	R/W	0	load_stat
13:8	R/W	0	reg32_13_13_8_reserved
7:0	R/W	0	Custom_Pattern_19

Table 13-15 reg32\_14 0x38

Bit(s)	R/W	Default	Description
31:24	R/W	0x03	usb2_reg_cfg
23:16	R/W	0	Bypass_ctrl_15_8 bypass_ctrl[8]: bypass i_rpd_en bypass_ctrl[9]: bypass i_rpu_sw2_en bypass_ctrl[10]: i_rpu_sw1_en bypass_ctrl[11]: reserved bypass_ctrl [12]: pg_rstn bypass_ctrl[13]: pg_data_16_8 bypass_ctrl[14]: assert_se0
15:8	R/W	0	Bypass_ctrl_7_0 bypass_ctrl[0]: hs bypass_ctrl[1]: fs bypass_ctrl[2]: ls bypass_ctrl[3]:hs_out_en bypass_ctrl[4]:fsls_out_en bypass_ctrl[5]:hs_rx_en bypass_ctrl [6]:hls_rx_en
7	R/W	0	reg32_14_7_reserved
6	R/W	0	i_c2l_assert_single_enable_zero
5	R/W	0	i_c2l_data_16_8
4	R/W	0	pg_rstn
3:2	R/W	0	bypass_reg[11:10]:i_rpu_sw1_en
1	R/W	0	bypass_reg[9]:i_rpu_sw2_en
0	R/W	0	bypass_reg[8]: i_rpd_en

Table 13-16 reg32\_15 0x3c

Bit(s)	R/W	Default	Description
31:29	R/W	0	reg32_15_31_29_reserved
28:16	R/W	0xfa0	ms_4_cntr
15:8	R/W	0x3c	non_se0_cntr
7:0	R/W	0x3c	se0_cntr

**Table 13-17 reg32\_16 0x40**

Bit(s)	R/W	Default	Description
31	RO	0	usb2_mppll_lock_dig
30	RO	0	usb2_mppll_lock
29	R/W	0	usb2_mppll_reset
28	R/W	0	usb2_mppll_en
27	R/W	0x1	usb2_mppll_fast_lock
26	R/W	0	usb2_mppll_lock_f
25:24	R/W	0x1	usb2_mppll_lock_long
23	R/W	0	usb2_mppll_dco_sdm_en
22	R/W	0x1	usb2_mppll_load
21	R/W	0	usb2_mppll_sdm_en
20	R/W	0	usb2_mppll_tdc_mode
19:15	R/W	0	reg32_16_19_15_reserved
14:10	R/W	0x1	usb2_mppll_n
9	R/W	0	reg32_16_9_reserved
8:0	R/W	0x14	usb2_mppll_m

**Table 13-18 reg32\_17 0x44**

Bit(s)	R/W	Default	Description
31:28	R/W	0x9	usb2_mppll_filter_pvt1
27:24	R/W	0x2	usb2_mppll_filter_pvt2
23	R/W	0	usb2_mppll_filter_mode
22:20	R/W	0x7	usb2_mppll_lambda0
19:17	R/W	0x7	usb2_mppll_lambda1
16	R/W	0	usb2_mppll_fix_en
15:14	R/W	0	reg32_17_15_14_reserved
13:0	R/W	0	usb2_mppll_frac_in

**Table 13-19 reg32\_18 0x48**

Bit(s)	R/W	Default	Description
31	R/W	0	usb2_mppll_acg_range
30:29	R/W	0x3	usb2_mppll_adj_ldo
28:26	R/W	0x3	usb2_mppll_alpha
25:24	R/W	0x1	usb2_mppll_bb_mode
23:22	R/W	0x1	usb2_mppll_bias_adj

Bit(s)	R/W	Default	Description
21:19	R/W	0x3	usb2_mppll_data_sel
18:16	R/W	0x3	usb2_mppll_rou
15:14	R/W	0	usb2_mppll_pfd_gain
13	R/W	0x1	usb2_mppll_dco_clk_sel
12	R/W	0	usb2_mppll_dco_m_en
11:6	R/W	0x27	usb2_mppll_lk_s
5:2	R/W	0x9	usb2_mppll_lk_w
1:0	R/W	0x1	usb2_mppll_lkw_sel

Table 13-20 reg32\_19 0x4c

Bit(s)	R/W	Default	Description
31	RO	0	usb2_mppll_lock_dig
30	RO	0	usb2_mppll_lock
29:10	RO	0	reg32_19_29_10_reserved
9:0	RO	0	usb2_mppll_reg_out

Table 13-21 reg32\_20 0x50

Bit(s)	R/W	Default	Description
31	R/W	0	bypass_cal_done_r5
30:29	R/W	0	usb2_bgr_dbg_1_0
28:24	R/W	0	usb2_bgr_vref_4_0
23:22	R/W	0	sqelch_sel: 01: debounce 1; 10: debounce 2; 00/11: no debounce
21	R/W	0	usb2_bgr_start
20:16	R/W	0	usb2_bgr_adj_4_0
15:14	R/W	0	usb2_edgedrv_trim_1_0
13	R/W	0	usb2_edgedrv_en
12:9	R/W	0xf	usb2_dmon_sel_3_0
8	R/W	0	usb2_dmon_en
7	R/W	0	bypass_otg_det
6	R/W	0	usb2_cal_code_r5
5	R/W	0	usb2_amon_en
4	R/W	0x0	usb2_otg_vbusdet_en
3:1	R/W	0x4	usb2_otg_vbus_trim_2_0
0	R/W	0	usb2_otg_iddet_en

Table 13-22 reg32\_21 0x54

Bit(s)	R/W	Default	Description
31:26	R/W	0	bypass_chirp_reg [30]: bypass hs chirpK reg [31]: bypass hs term_sel reg
29:28	R/W	0	bypass_chirp_ctrl [28]: bypass hs chirpK [29]: bypass hs term_sel
27:26	R/W		reg32_21_27_26_reserved
25:20	R/W	0	bypass_utmi_reg [21:20]: xcvr_select ctrl reg [22]: term_select ctrl reg [23]: suspend ctrl reg [25:24]: opmode ctrl reg
19:16	R/W	0	bypass_utmi_cntr [16]: bypass xcvr_select [17]: bypass term_select [18]: bypass suspend [19]: bypass opmode
15:8	R/W	0	hs cdr ctrl
7	R/W		hs cdr sel
6	R/W		reg32_21_6_reserved
5:4	R/W	0x2	usb2_otg_aca_trim_1_0
3	R/W	0	usb2_tx_strg_pd
2	R/W	0x0	usb2_otg_aca_en
1	R/W	0x1	usb2_cal_ack_en
0	R/W	0	usb2_bgr_force

Table 13-23 reg32\_22 0x58

Bit(s)	R/W	Default	Description
31:14	RO	0	reg32_22_31_14_reserved
[13:6]			hs cdr state
5:3	RO	0	usb2_otg_aca_iddig
2	RO	0	usb2_otg_vbus_vld
1	RO	0	usb2_otg_sess_vld
0	RO	0	usb2_otg_id_dig

Table 13-24 reg32\_23 0x5c

Bit(s)	R/W	Default	Description
31:16	RO	0	test_bus_data_int_15_0
15:13		0	ldo_trim
12		1	ldo_en
11		0	new_hs_disc_ctrl[1]
10		0	new_hs_disc_ctrl[0]
9		0	sel_cdr
8		0	pcs_sel
7	R/W	0	orw_test_bus_en

Bit(s)	R/W	Default	Description
6:1	R/W	0	orw_test_bus_sel_5_0
0	R/W	0	orw_usb2_bgr_en

## 13.2 PCIE

PCIE module includes PCIE controller and PCIE PHY.

## 13.3 Ethernet

### 13.3.1 Overview

The Ethernet MAC controller provides a complete ethernet interface from the chip to a Reduced Gigabit Media Independent Interface(RGMII) compliant Ethernet PHY.

### 13.3.2 Ethernet Top Registers

Base address: 0xfe024000

**Table 13-25 ETHTOP\_CNTL0 0x140**

Bit(s)	R/W	Default	Description
31	R/W	0	eth_urgent 1: set AHB to DDR interface as urgent
30	R/W	0	clock source select for rgmii_tx_clk 1: use rx_clk as tx_clk
29:27	R/W	0	cali_sel RMII & RGMII mode Select one signal from {RXDV, RXD3:0} to calibrate.
26	R/W	0	cali_rise RMII & RGMII mode 0: test falling edge 1: test rising edge
25	R/W	0	cali_start RMII & RGMII mode Start calibration logic
24:20	R/W	0	adj_skew RMII & RGMII mode 5 bits correspondent to {RXDV, RXD3:0}, set to 1 will delay the data capture by 1 cycle(fast clock).
19:15	R/W	0	adj_delay Set bit14 to 0. RMII & RGMII mode Capture input data at clock index equal to adj_delay.
14	R/W	0	adj_setup Set RXDV and RXD setup time, data is aligned with index 0. When set to 1, auto delay and skew
13	R/W	0	adj_enable RMII & RGMII mode Enable data delay adjustment and calibration logic.
12	R/W	0	clk_en



Bit(s)	R/W	Default	Description
			RMII & RGMII mode Enable TX_CLK and PHY_REF_CLK generator.
11	R/W	0	clk_rmii_i_invert RMII mode Use inverted internal clk_rmii_i to generate 25/2.5 tx_rx_clk.
9:7	R/W	0	rgmii_tx_clk_ratio RMII & RGMII mode, 000: invalid value. 001: clk_m250 is 250MHz. 010: clk_m250 is 500MHz. 100: clk_m250 is 1000MHz. clk_m250 is "ratio" *250MHz.
6:5	R/W	0	rgmii_tx_clk phase select RGMII mode, TX_CLK related to TXD 00: clock delay 0 cycle. 01: clock delay ¼ cycle. 10: clock delay ½ cycle. 11: clock delay ¾ cycle.
4	R/W	0	clock source select for rgmii_tx_clk 0: clk_m250_src0(fclk_div2) 1: clk_m250_src1(fclk_div4)
3	R/W	0	invert control for rx_clk_or_clk_rmii_i RMII mode CLK_RMII RGMII mode RX_CLK Use inverted signal when set to 1.
2:0	R/W	0	PHY Interface Select Function: These pins select one of the multiple PHY interfaces of MAC. This is sampled only during reset assertion and ignored after that. 3'b001: RGMII 3'b100: RMII 3'b000: MII

Table 13-26 EHTOP\_CNTL1 0x144

Bit(s)	R/W	Default	Description
25:21	R/W	0	cfg_txclk_dly Add delay to input TX_CLK for better timing
20:16	R/W	0	cfg_rxclk_dly Add delay to input RX_CLK for better timing
15	R		cali_vld the result is valid
14	R		cali_rise The results is rising edge test or falling edge test.
13:11	R		cali_sel The signal under test.
10	R		cali_act The Calibration logic is waiting for event.
9:5	R		cali_len The RX_CLK length in 1ns.
4:0	R		cali_idx Signal switch position in 1ns.

**Table 13-27 ETHTOP\_CNTL2 0x158**

Bit(s)	R/W	Default	Description
31:3	R/W	0	reserved
2	R/W	0	pwr_isolate_i
1	R/W	0	pwr_clamp_ctrl_i
0	R/W	0	pwr_down_ctrl_i

**Table 13-28 ETHTOP\_CNTL3 0x15c**

Bit(s)	R/W	Default	Description
31:0	R/W	0	reserved

**Table 13-29 ETHTOP\_STS0 0x160**

Bit(s)	R/W	Default	Description
31:0	R	0	ptp_timestamp_o31:0

**Table 13-30 ETHTOP\_STS1 0x164**

Bit(s)	R/W	Default	Description
31:0	R	0	ptp_timestamp_o63:32

**Table 13-31 ETHTOP\_STS2 0x168**

Bit(s)	R/W	Default	Description
31:5	R	0	reserved
4	R	0	sbd_pwr_down_ack_o
3	R	0	sbd_tx_clk_gating_ctrl_o LPI Tx clock gating control, high after MAC enter the Tx LPI mode
2:0	R	0	mdc_chid_o current transfer DMA channel ID

### 13.3.3 Ethernet MAC

#### 13.3.3.1 Features

Ethernet MAC has the following features:

- 10/100/1000 MAC 5.10a
- RGMII/RMII
- AHB 32 Bits internal bus
- RX FIFO 8KB, TX FIFO 8KB
- 2 MAC addresses
- EEE

- Power Management

### 13.3.4 Ethernet PHY

#### 13.3.4.1 Features

Ethernet PHY has the following features:

- Integrated IEEE 802.3/802.3u compliant 10/100Mbps Ethernet PHY
- Supporting both full and half-duplex for either 10 or 100 Mb/s data rate
- Auto MDIX capable
- Supports wake-on-LAN
- 100 Base-T support
- MII/RMII/SMII interface
- Supports auto-negotiation
- Full set of power down modes
- Interface available to 100Base-FX Fiber-PMD
- Serial Management Interface (SMI)
- Fix configurations for LED status indicators
- Supporting military temperature range -20°C to 80°C
- Perfect mix of analog and digital lends itself to robustness, portability, and performance
- Multiple input clock options
- Stand-alone core

#### 13.3.4.2 Register Description

Base Address: 0xfe028000

**Table 13-32 ETH\_PHY\_DBG\_CTL0 0x0**

Bit(s)	R/W	Default	Description
7:0	R/W	0	ETH digital debug registers

**Table 13-33 ETH\_PHY\_DBG\_CTL1 0x4**

Bit(s)	R/W	Default	Description
23:0	R/W	0	ETH digital debug registers

**Table 13-34 ETH\_PHY\_DBG\_CFG0 0x8**

Bit(s)	R/W	Default	Description
31:0	R/W	0	ETH digital debug registers

**Table 13-35 ETH\_PHY\_DBG\_CFG1 0xc**

Bit(s)	R/W	Default	Description
31:0	R/W	0	ETH digital debug registers

**Table 13-36 ETH\_PHY\_DBG\_CFG2 0x10**

Bit(s)	R/W	Default	Description
31:0	R/W	0	ETH digital debug registers

**Table 13-37 ETH\_PHY\_DBG\_CFG3 0x14**

Bit(s)	R/W	Default	Description
31:0	R/W	0	ETH digital debug registers

**Table 13-38 ETH\_PHY\_DBG\_CFG4 0x18**

Bit(s)	R/W	Default	Description
31:0	R/W	0	ETH digital debug registers

**Table 13-39 ETH\_PLL\_STS 0x40**

Bit(s)	R/W	Default	Description
31	R		eth_mppll_lock
30	R		eth_mppll_lock_dig
29:10	R		reserved
9:0	R		eth_mppll_reg_out

**Table 13-40 ETH\_PLL\_CTL0 0x44**

Bit(s)	R/W	Default	Description
31	R		eth_mppll_lock
30	R		eth_mppll_lock_dig
29	R/W		eth_mppll_reset
28	R/W		eth_mppll_en
27	R/W		eth_mppll_fast_lock
26	R/W		eth_mppll_lock_f
25:24	R/W		eth_mppll_lock_long
23	R/W		eth_mppll_sel_ref
22	R/W		eth_mppll_load
21	R/W		eth_mppll_sdm_en
20	R/W		eth_mppll_tdc_mode
19:15	R/W		reserved
14:10	R/W		eth_mppll_n
9	R/W		reserved
8:0	R/W		eth_mppll_m

**Table 13-41 ETH\_PLL\_CTL1 0x48**

Bit(s)	R/W	Default	Description
31:28	R/W		eth_mppll_filter_pvt1
27:24	R/W		eth_mppll_filter_pvt2
23	R/W		eth_mppll_filter_mode
22:20	R/W		eth_mppll_lambda0
19:17	R/W		eth_mppll_lambda1
16	R/W		eth_mppll_fix_en
15:14	R/W		reserved
13:0	R/W		eth_mppll_frac_in

**Table 13-42 ETH\_PLL\_CTL2 0x4C**

Bit(s)	R/W	Default	Description
31	R/W		eth_mppll_acq_range
30:29	R/W		eth_mppll_adj_ldo
28:26	R/W		eth_mppll_alpha
25:24	R/W		eth_mppll_bb_mode
23:22	R/W		eth_mppll_bias_adj
21:19	R/W		eth_mppll_data_sel
18:16	R/W		eth_mppll_rou
15:14	R/W		eth_mppll_pfd_gain
13	R/W		eth_mppll_dco_clk_sel
12	R/W		eth_mppll_dco_m_en
11:6	R/W		eth_mppll_lk_s
5:2	R/W		eth_mppll_lk_w
1:0	R/W		eth_mppll_lkw_sel

**Table 13-43 ETH\_PLL\_CTL3 0x50**

Bit(s)	R/W	Default	Description
31:1	R		reserved
0	R/W		eth_mppll_dco_sdm_en

**Table 13-44 ETH\_PLL\_CTL4 0x54**

Bit(s)	R/W	Default	Description
31:0	R/W		reserved

Table 13-45 ETH\_PLL\_CTL4 0x54

Bit(s)	R/W	Default	Description
31:1	R/W		reserved
0	R/W	0	led_cfg_1

Table 13-46 ETH\_PLL\_CTL5 0x58

Bit(s)	R/W	Default	Description
31:16	R/W	0	ETH_PHY_RXADC0
15:3	R/W	0	ETH_PHY_CTLIO
2:0	R/W	0	reserved

Table 13-47 ETH\_PLL\_CTL6 0x5C

Bit(s)	R/W	Default	Description
31:16	R/W	0	reserved
15:0	R/W	0	ETH_PHY_RXSQLD

Table 13-48 ETH\_PLL\_CTL7 0x60

Bit(s)	R/W	Default	Description
31:16	R/W	0	reserved
15:0	R/W	0	ETH_PHY_RXADC1

Table 13-49 ETH\_PHY\_CNTL0 0x80

Bit(s)	R/W	Default	Description
31:16	R/W	0	co_reg3_oui_in, SMI register 3 default value
15:0	R/W	0	co_reg2_oui_in, SMI register 2 default value

Table 13-50 ETH\_PHY\_CNTL1 0x84

Bit(s)	R/W	Default	Description
31:24	R/W	0	{led_cfg_1,led_cfg7:5}: co_activity_led output select 0: link_led & (~activity) 1: link_led 2: link_led & activity 3: link_led   activity 4: activity 5: rx_led 6: tx_led 7: duplex_led 8: link_led   (~activity) led_cfg4:2: co_link_speed_led output select 0: speed100_led 1: activity & speed100_led 2: link_led & speed100_led 3: link_led & activity & speed100_led 4: speed10_led 5: activity & speed10_led 6: link_led & speed10_led 7: link_led & activity & speed10_led led_cfg1: invert co_link_speed_led when output to gpio led_cfg0: invert all led signal from phy
23	R/W	0	invert co_activity_led invert co_activity_led when output to gpio
22	R/W	0	co_pwruprst_byp

Bit(s)	R/W	Default	Description
21	R/W	0	co_clk_ext
20	R/W	0	co_st_scan
19	R/W	0	co_rxclk_inv
18	R/W	0	co_phy_enb
17	R/W	0	co_clkfreq
16	R/W	0	eth_clk_enable
15:14	R/W	0	co_st_miimode1:0
13	R/W	0	co_smii_source_sync
12	R/W	0	co_st_pllbp
11	R/W	0	co_st_adcbp
10	R/W	0	co_st_fxmode
9	R/W	0	co_en_high
8	R/W	0	co_automdix_en
7:3	R/W	0	co_st_phyadd4:0
2:0	R/W	0	co_st_mode2:0

Table 13-51 ETH\_PHY\_CNTL2 0x88

Bit(s)	R/W	Default	Description
31	R/W	0	reserved
30	R/W	0	eth_phy_clk25min_en:
29:28	R/W	0	reserved
27:24	R/W	0	debug output bus select
23:13	R/W	0	reserved
12	R/W	0	analog production test mode enable
11	R/W	0	mdi source select
10	R/W	0	reserved
9	R/W	0	source select for rx_clk to mac
8	R/W	0	source select for tx_clk output to gpio
7	R/W	0	rx_dv/col switch for mac
6	R/W	0	ephy smi source select
5	R/W	0	use internal phy
4	R/W	0	ephy smi source select
3	R/W	0	co_clkin source select
2	R/W	0	co_mdclk source select
1	R/W	0	enable reset in test mode
0	R/W	0	ephy loopback mode enable, in/out through debug in/out gpio

**Table 13-52 ETH\_PHY\_CNTL3 0x8C**

Bit(s)	R/W	Default	Description
6	R/W	0	eth_txda_test_sel
5:0	R/W	0	eth_txda_test_data

**Table 13-53 ETH\_PHY\_STS0 0x94**

Bit(s)	R/W	Default	Description
31:28	R		reserved
27:19	R		co_int_vec8:0
19	R		reserved
18:0	R		ETH_PLL_STS18:0

**Table 13-54 ETH\_PHY\_STS1 0x98**

Bit(s)	R/W	Default	Description
31:0	R		eth_phy_dbg_prb

**Table 13-55 ETH\_PHY\_STS2 0x9C**

Bit(s)	R/W	Default	Description
31:26	R		eth_phy_rxda5:0
25	R		eth_phy_rxadcoflw
24	R		eth_phy_rxadcufw
23	R		eth_phy_rxprepga
22:20	R		eth_phy_rxgain2:0
19:0	R		eth_test_status

**Table 13-56 ETH\_PHY\_DBG\_REG 0xA0**

Bit(s)	R/W	Default	Description
31:16	R		eth_phy_dbg_reg
15:8			reserved
7:0	R/W	0	eth_phy_dbg_reg_mux

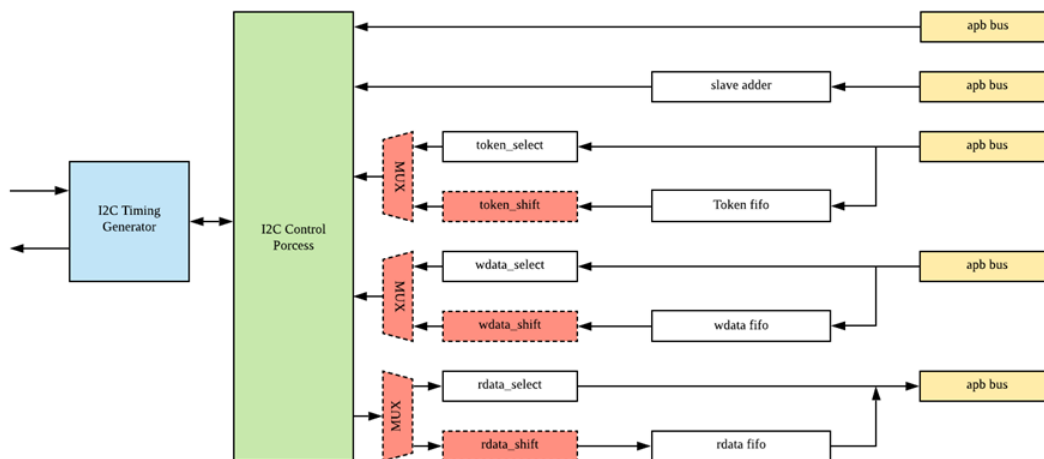
## 13.4 Inter-Integrated Circuit (I2C)



### 13.4.1 Overview

Inter-Integrated Circuit (IIC or I2C) is a multi-slave serial communication bus between ICs. The chip integrates the I2C interface and signals allowing communications with other I2C peripheral devices.

**Figure 13-1 I2C Diagram**



### 13.4.2 Features

The I2C Master Module has the following features:

- Support for 7-bit and 10-bit addressable devices
- Programmable bus speed including standard speed (100 KBits/s) and fast speed (400 KBits/s)
- Error transfer detection
- “Transfer complete” indication by polling or interrupt (Interrupts handled by the ISA module).
- Internal buffer holding up to 8 bytes for transfer (in either direction)
- Flexible architecture allowing the software to dictate the format of the I2C bit streams
- Manual setting of the I2C bus to accommodate a software only mode

### 13.4.3 Register Description

The following list shows address for each I2C registers.

- I2C\_M\_AO\_A\_CONTROL\_REG 0xfe076000
- I2C\_M\_AO\_A\_SLAVE\_ADDR 0xfe076004
- I2C\_M\_AO\_A\_TOKEN\_LIST0 0xfe076008
- I2C\_M\_AO\_A\_TOKEN\_LIST1 0xfe07600c
- I2C\_M\_AO\_A\_WDATA\_REG0 0xfe076010
- I2C\_M\_AO\_A\_WDATA\_REG1 0xfe076014
- I2C\_M\_AO\_A\_RDATA\_REG0 0xfe076018
- I2C\_M\_AO\_A\_RDATA\_REG1 0xfe07601c
- I2C\_M\_AO\_A\_TIMEOUT\_TH 0xfe076020
- I2C\_M\_AO\_A\_CNTL\_DELY1 0xfe076024
- I2C\_M\_AO\_A\_CNTL\_DELY2 0xfe076028

- I2C\_M\_AO\_A\_LOW\_DELY 0xfe07602c
- I2C\_M\_AO\_A\_HIGH\_DELY 0xfe076030
- I2C\_M\_AO\_A\_FIFO\_CTRL0 0xfe076034
- I2C\_M\_AO\_A\_FIFO\_CTRL1 0xfe076038
- I2C\_M\_AO\_A\_FIFO\_PENDING 0xfe07603c
- I2C\_M\_AO\_A\_FIFO\_PENDING\_MASK 0xfe076040
- I2C\_M\_AO\_A\_FIFO\_ST0 0xfe076044
- I2C\_M\_AO\_B\_CONTROL\_REG 0xfe086000
- I2C\_M\_AO\_B\_SLAVE\_ADDR 0xfe086004
- I2C\_M\_AO\_B\_TOKEN\_LIST0 0xfe086008
- I2C\_M\_AO\_B\_TOKEN\_LIST1 0xfe08600c
- I2C\_M\_AO\_B\_WDATA\_REG0 0xfe086010
- I2C\_M\_AO\_B\_WDATA\_REG1 0xfe086014
- I2C\_M\_AO\_B\_RDATA\_REG0 0xfe086018
- I2C\_M\_AO\_B\_RDATA\_REG1 0xfe08601c
- I2C\_M\_AO\_B\_TIMEOUT\_TH 0xfe086020
- I2C\_M\_AO\_B\_CNTL\_DELY1 0xfe086024
- I2C\_M\_AO\_B\_CNTL\_DELY2 0xfe086028
- I2C\_M\_AO\_B\_LOW\_DELY 0xfe08602c
- I2C\_M\_AO\_B\_HIGH\_DELY 0xfe086030
- I2C\_M\_AO\_B\_FIFO\_CTRL0 0xfe086034
- I2C\_M\_AO\_B\_FIFO\_CTRL1 0xfe086038
- I2C\_M\_AO\_B\_FIFO\_PENDING 0xfe08603c
- I2C\_M\_AO\_B\_FIFO\_PENDING\_MASK 0xfe086040
- I2C\_M\_AO\_B\_FIFO\_ST0 0xfe086044
- I2C\_M\_A\_CONTROL\_REG 0xfe066000
- I2C\_M\_A\_SLAVE\_ADDR 0xfe066004
- I2C\_M\_A\_TOKEN\_LIST0 0xfe066008
- I2C\_M\_A\_TOKEN\_LIST1 0xfe06600c
- I2C\_M\_A\_WDATA\_REG0 0xfe066010
- I2C\_M\_A\_WDATA\_REG1 0xfe066014
- I2C\_M\_A\_RDATA\_REG0 0xfe066018
- I2C\_M\_A\_RDATA\_REG1 0xfe06601c
- I2C\_M\_A\_TIMEOUT\_TH 0xfe066020
- I2C\_M\_A\_CNTL\_DELY1 0xfe066024
- I2C\_M\_A\_CNTL\_DELY2 0xfe066028
- I2C\_M\_A\_LOW\_DELY 0xfe06602c
- I2C\_M\_A\_HIGH\_DELY 0xfe066030
- I2C\_M\_A\_FIFO\_CTRL0 0xfe066034
- I2C\_M\_A\_FIFO\_CTRL1 0xfe066038
- I2C\_M\_A\_FIFO\_PENDING 0xfe06603c
- I2C\_M\_A\_FIFO\_PENDING\_MASK 0xfe066040
- I2C\_M\_A\_FIFO\_ST0 0xfe066044
- I2C\_M\_B\_CONTROL\_REG 0xfe068000

- I2C\_M\_B\_SLAVE\_ADDR 0xfe068004
- I2C\_M\_B\_TOKEN\_LIST0 0xfe068008
- I2C\_M\_B\_TOKEN\_LIST1 0xfe06800c
- I2C\_M\_B\_WDATA\_REG0 0xfe068010
- I2C\_M\_B\_WDATA\_REG1 0xfe068014
- I2C\_M\_B\_RDATA\_REG0 0xfe068018
- I2C\_M\_B\_RDATA\_REG1 0xfe06801c
- I2C\_M\_B\_TIMEOUT\_TH 0xfe068020
- I2C\_M\_B\_CNTL\_DELY1 0xfe068024
- I2C\_M\_B\_CNTL\_DELY2 0xfe068028
- I2C\_M\_B\_LOW\_DELY 0xfe06802c
- I2C\_M\_B\_HIGH\_DELY 0xfe068030
- I2C\_M\_B\_FIFO\_CTRL0 0xfe068034
- I2C\_M\_B\_FIFO\_CTRL1 0xfe068038
- I2C\_M\_B\_FIFO\_PENDING 0xfe06803c
- I2C\_M\_B\_FIFO\_PENDING\_MASK 0xfe068040
- I2C\_M\_B\_FIFO\_ST0 0xfe068044
- I2C\_M\_C\_CONTROL\_REG 0xfe06a000
- I2C\_M\_C\_SLAVE\_ADDR 0xfe06a004
- I2C\_M\_C\_TOKEN\_LIST0 0xfe06a008
- I2C\_M\_C\_TOKEN\_LIST1 0xfe06a00c
- I2C\_M\_C\_WDATA\_REG0 0xfe06a010
- I2C\_M\_C\_WDATA\_REG1 0xfe06a014
- I2C\_M\_C\_RDATA\_REG0 0xfe06a018
- I2C\_M\_C\_RDATA\_REG1 0xfe06a01c
- I2C\_M\_C\_TIMEOUT\_TH 0xfe06a020
- I2C\_M\_C\_CNTL\_DELY1 0xfe06a024
- I2C\_M\_C\_CNTL\_DELY2 0xfe06a028
- I2C\_M\_C\_LOW\_DELY 0xfe06a02c
- I2C\_M\_C\_HIGH\_DELY 0xfe06a030
- I2C\_M\_C\_FIFO\_CTRL0 0xfe06a034
- I2C\_M\_C\_FIFO\_CTRL1 0xfe06a038
- I2C\_M\_C\_FIFO\_PENDING 0xfe06a03c
- I2C\_M\_C\_FIFO\_PENDING\_MASK 0xfe06a040
- I2C\_M\_C\_FIFO\_ST0 0xfe06a044
- I2C\_M\_D\_CONTROL\_REG 0xfe06c000
- I2C\_M\_D\_SLAVE\_ADDR 0xfe06c004
- I2C\_M\_D\_TOKEN\_LIST0 0xfe06c008
- I2C\_M\_D\_TOKEN\_LIST1 0xfe06c00c
- I2C\_M\_D\_WDATA\_REG0 0xfe06c010
- I2C\_M\_D\_WDATA\_REG1 0xfe06c014
- I2C\_M\_D\_RDATA\_REG0 0xfe06c018
- I2C\_M\_D\_RDATA\_REG1 0xfe06c01c
- I2C\_M\_D\_TIMEOUT\_TH 0xfe06c020

- I2C\_M\_D\_CNTL\_DELY1 0xfe06c024
- I2C\_M\_D\_CNTL\_DELY2 0xfe06c028
- I2C\_M\_D\_LOW\_DELY 0xfe06c02c
- I2C\_M\_D\_HIGH\_DELY 0xfe06c030
- I2C\_M\_D\_FIFO\_CTRL0 0xfe06c034
- I2C\_M\_D\_FIFO\_CTRL1 0xfe06c038
- I2C\_M\_D\_FIFO\_PENDING 0xfe06c03c
- I2C\_M\_D\_FIFO\_PENDING\_MASK 0xfe06c040
- I2C\_M\_D\_FIFO\_ST0 0xfe06c044
- I2C\_M\_E\_CONTROL\_REG 0xfe06e000
- I2C\_M\_E\_SLAVE\_ADDR 0xfe06e004
- I2C\_M\_E\_TOKEN\_LIST0 0xfe06e008
- I2C\_M\_E\_TOKEN\_LIST1 0xfe06e00c
- I2C\_M\_E\_WDATA\_REG0 0xfe06e010
- I2C\_M\_E\_WDATA\_REG1 0xfe06e014
- I2C\_M\_E\_RDATA\_REG0 0xfe06e018
- I2C\_M\_E\_RDATA\_REG1 0xfe06e01c
- I2C\_M\_E\_TIMEOUT\_TH 0xfe06e020
- I2C\_M\_E\_CNTL\_DELY1 0xfe06e024
- I2C\_M\_E\_CNTL\_DELY2 0xfe06e028
- I2C\_M\_E\_LOW\_DELY 0xfe06e02c
- I2C\_M\_E\_HIGH\_DELY 0xfe06e030
- I2C\_M\_E\_FIFO\_CTRL0 0xfe06e034
- I2C\_M\_E\_FIFO\_CTRL1 0xfe06e038
- I2C\_M\_E\_FIFO\_PENDING 0xfe06e03c
- I2C\_M\_E\_FIFO\_PENDING\_MASK 0xfe06e040
- I2C\_M\_E\_FIFO\_ST0 0xfe06e044
- I2C\_M\_F\_CONTROL\_REG 0xfe070000
- I2C\_M\_F\_SLAVE\_ADDR 0xfe070004
- I2C\_M\_F\_TOKEN\_LIST0 0xfe070008
- I2C\_M\_F\_TOKEN\_LIST1 0xfe07000c
- I2C\_M\_F\_WDATA\_REG0 0xfe070010
- I2C\_M\_F\_WDATA\_REG1 0xfe070014
- I2C\_M\_F\_RDATA\_REG0 0xfe070018
- I2C\_M\_F\_RDATA\_REG1 0xfe07001c
- I2C\_M\_F\_TIMEOUT\_TH 0xfe070020
- I2C\_M\_F\_CNTL\_DELY1 0xfe070024
- I2C\_M\_F\_CNTL\_DELY2 0xfe070028
- I2C\_M\_F\_LOW\_DELY 0xfe07002c
- I2C\_M\_F\_HIGH\_DELY 0xfe070030
- I2C\_M\_F\_FIFO\_CTRL0 0xfe070034
- I2C\_M\_F\_FIFO\_CTRL1 0xfe070038
- I2C\_M\_F\_FIFO\_PENDING 0xfe07003c
- I2C\_M\_F\_FIFO\_PENDING\_MASK 0xfe070040

- I2C\_M\_F\_FIFO\_ST0 0xfe070044

The following content describes registers of I2C\_M\_A, and the other I2C's registers description is the same.

**Table 13-57 I2C\_M\_A\_CONTROL\_REG**

Bit(s)	R/W	Default	Description
31	R/W	0	CNTL_JIC: There is internal logic to dynamically enable the gated clocks. If this gated clock logic doesn't work, you can set this bit to always enable the clock. Setting this bit wastes power.
30	R/W	0	enhance mode, enhance mode enable timing adjust. =1,enable enhance mode =0, disable enhance mode
29-28	R/W	0	QTR_CLK_EXT: These two bits extend the clock divider to 12 bits: QTR_CLK = {[29:28],[21:12]}
26	R	0	Read back level of the SDA line
25	R	0	Read back level of the SCL line
24	R/W	0	Sets the level of the SDA line if manual mode is enabled. If this bit is '0', then the SDA line is pulled low. If this bit is '1' then the SDA line is tri-stated.
23	R/W	0	Sets the level of the SCL line if manual mode is enabled. If this bit is '0', then the SCL line is pulled low. If this bit is '1' then the SCL line is tri-stated.
22	R/W	0	This bit is used to enable manual mode. Manual I2C mode is controlled by bits 12,13,14 and 15 above.
21:12	R/W	0x142	QTR_CLK_DLY. This value corresponds to period of the SCL clock divided by 4 Quarter Clock Delay = * System Clock Frequency For example, if the system clock is 133Mhz, and the I2C clock period is 10uS (100khz), then Quarter Clock Delay = * 133 Mhz = 332
11:8	R	-	READ_DATA_COUNT: This value corresponds to the number of bytes READ over the I2C bus. If this value is zero, then no data has been read. If this value is 1, then bits [7:0] in TOKEN_RDATA_REG0 contains valid data. The software can read this register after an I2C transaction to get the number of bytes to read from the I2C device.
7:4	R	-	CURRENT_TOKEN: This value reflects the current token being processed. In the event of an error, the software can use this value to determine the error location.
3	R	-	ERROR: This read only bit is set if the I2C device generates a NACK during writing. This bit is cleared at on the clock cycle after the START bit is set to 1 indicating the start of list processing. Errors can be ignored by setting the ACK_IGNORE bit below. Errors will be generated on Writes to devices that return NACK instead of ACK. A NACK is returned by a device if it is unable to accept any more data (for example because it is processing some other real-time function). In the event of an ERROR, the I2C module will automatically generate a STOP condition on the bus.
2	R	-	STATUS: This bit reflects the status of the List processor: 0: IDLE 1: Running. The list processor will enter this state on the clock cycle after the START bit is set. The software can poll the status register to determine when processing is complete.

Bit(s)	R/W	Default	Description
1	R/W	0	ACK_IGNORE: Set to 1 to disable I2C ACK detection. The I2C bus uses an ACK signal after every byte transfer to detect problems during the transfer. Current Software implementations of the I2C bus ignore this ACK. This bit is for compatibility with the current Amlogic software. This bit should be set to 0 to allow NACK operations to abort I2C bus transactions. If a NACK occurs, the ERROR bit above will be set.
0	R/W	0	START: Set to 1 to start list processing. Setting this bit to 0 while the list processor is operating causes the list processor to abort the current I2C operation and generate an I2C STOP command on the I2C bus. Normally this bit is set to 1 and left high until processing is complete. To re-start the list processor with a new list (after a previous list has been exhausted), simply set this bit to zero then to one.

Table 13-58 I2C\_M\_A\_SLAVE\_ADDR

Bit(s)	R/W	Default	Description
31:29	R	0	Unused
28	R/W	0	USE_CNTL_SCL_LOW: If this bit is set to 1, then bits[27:16] control the SCL low time.
27:16	R/W	0	SCL Low delay. This is a new feature in M8baby. In the previous M8baby design, the SCL low time was controlled by bits[21:12] of the register above. In this design, the SCL delay is controlled independently by these bits.
15:14	R	0	Unused
13-11	R/W	0	SCL_FILTER: A filter was added in the SCL input path to allow for filtering of slow rise times. 0 = no filtering, 7 = max filtering
10:8	R/W	0	SDA_FILTER: A filter was added in the SDA input path to allow for filtering of slow rise times. 0 = no filtering, 7 = max filtering
7:0	R/W	0x00	SLAVE_ADDRESS. This is a 7-bit value for a 7-bit I2C device, or (0xF0   {A9, A8}) for a 10-bit I2C device. By convention, the slave address is typically stored in by first left shifting it so that it's MSB is D7 (The I2C bus assumes the 7-bit address is left shifted one). Additionally, since the SLAVE address is always a 7-bit value, D0 is always 0. NOTE: The I2C always transfers 8-bits even for address. The I2C hardware will use D0 to dictate the direction of the bus. Therefore, D0 should always be '0' when this register is set.

Table 13-59 I2C\_M\_A\_TOKEN\_LIST0

Bit(s)	R/W	Default	Description
31:28	R/W	0x00	8th token in the list to process
27:24	R/W	0x00	7th token in the list to process
23:20	R/W	0x00	6th token in the list to process
19:16	R/W	0x00	5th token in the list to process
15:12	R/W	0x00	4th token in the list to process
11:8	R/W	0x00	3rd token in the list to process
7:4	R/W	0x00	2nd token in the list to process
3:0	R/W	0x00	1st token in the list to process (See the table below for token definitions)

**Table 13-60 Token Definitions**

Command Token	Value	Data	Description
END	0x0	N/A	Used to tell the I <sup>2</sup> C module that this is the end of the Token list. This token is not associated with the I <sup>2</sup> C bus, but rather with the state-machine that drives the token list processor.
START	0x1	N/A	The START Token is used to tell an I <sup>2</sup> C device that this is the beginning of an I <sup>2</sup> C transfer
SLAVE_ADDR-WRITE	0x2	7-bits	This bit-sequence is used to address a device and tell the device it is being WRITTEN
SLAVE_ADDR-READ	0x3	7-bits	This bit sequence is used to address a device and tell the device it is being READ.
DATA	0x4	8-bits	This 8-bit byte sequence is a byte transfer (READ or WRITE). The DATA token corresponds to a WRITE if it follows a SLAVE_ADDR-WRITE token. The DATA token corresponds to a READ if it follows a SLAVE_ADDR-READ token.
DATA-LAST	0x5	8-bits	Used to indicate the last 8-bit byte transfer is a byte transfer of a READ.
STOP	0x6	N/A	This tells the I <sup>2</sup> C device it is no longer being addressed

Write data associated with the DATA token should be placed into the I2C\_TOKEN\_WDATA\_REG0 or I2C\_TOKEN\_WDATA\_REG1 registers.

Read data associated with the DATA or DATA-LAST token can be read from the I2C\_TOKEN\_RDATA\_REG0 or I2C\_TOKEN\_RDATA\_REG1 registers.

**Table 13-61 I2C\_M\_A\_TOKEN\_LIST1**

Bit(s)	R/W	Default	Description
31:28	R/W	0x00	16th token in the list to process
27:24	R/W	0x00	15th token in the list to process
23:20	R/W	0x00	14th token in the list to process
19:16	R/W	0x00	13th token in the list to process
15:12	R/W	0x00	12th token in the list to process
11:8	R/W	0x00	11th token in the list to process
7:4	R/W	0x00	10th token in the list to process
3:0	R/W	0x00	9th token in the list to process

**Table 13-62 I2C\_M\_A\_WDATA\_REG0**

Bit(s)	R/W	Default	Description
31:24	R/W	0x00	4th data byte written for a DATA (write) token.
23:16	R/W	0x00	3rd data byte written for a DATA (write) token.
15:8	R/W	0x00	2nd data byte written for a DATA (write) token.
7:0	R/W	0x00	1st data byte written for a DATA (write) token.

**Table 13-63 I2C\_M\_A\_WDATA\_REG1**

Bit(s)	R/W	Default	Description
31:24	R/W	0x00	8th data byte written for a DATA (write) token.
23:16	R/W	0x00	7th data byte written for a DATA (write) token.
15:8	R/W	0x00	6th data byte written for a DATA (write) token.
7:0	R/W	0x00	5th data byte written for a DATA (write) token.

**Table 13-64 I2C\_M\_A\_RDATA\_REG0**

Bit(s)	R/W	Default	Description
31:24	R/W	0x00	4th data byte read for a DATA or DATA-LAST (READ) token.
23:16	R/W	0x00	3rd data byte read for a DATA or DATA-LAST (READ) token.
15:8	R/W	0x00	2nd data byte read for a DATA or DATA-LAST (READ) token.
7:0	R/W	0x00	1st data byte read for a DATA or DATA-LAST (READ) token.

**Table 13-65 I2C\_M\_A\_RDATA\_REG1**

Bit(s)	R/W	Default	Description
31:24	R/W	0x00	8th data byte read for a DATA or DATA-LAST (READ) token.
23:16	R/W	0x00	7th data byte read for a DATA or DATA-LAST (READ) token.
15:8	R/W	0x00	6th data byte read for a DATA or DATA-LAST (READ) token.
7:0	R/W	0x00	5th data byte read for a DATA or DATA-LAST (READ) token.

**Table 13-66 I2C\_M\_A\_TIMEOUT\_TH**

Bit(s)	R/W	Default	Description
11:0	R/W	0x00	timeout irq th

**Table 13-67 I2C\_M\_A\_CTRL\_DELY1\_REG**

Bit(s)	R/W	Default	Description
31:24	R/W	0	Reserved
23:12	R/W	0x14c	repeat start command control delay value,Used to repair timing. Valid in the enhance mode.
11:0	R/W	0x14c	start command control delay value,Used to repair timing. Valid in the enhance mode.



**Table 13-68 I2C\_M\_A\_CTRL\_DELY2\_REG**

Bit(s)	R/W	Default	Description
31:12	R/W	0	Reserved
11:0	R/W	0x14c	stop command control delay value,Used to repair timing. Valid in the enhance mode.

**Table 13-69 I2C\_M\_A\_LOW\_DELY\_REG**

Bit(s)	R/W	Default	Description
31:24	R/W	0	Reserved
23:12	R/W	0x14c	low2_delay data transfer second low level delay,Used to repair timing. Valid in the enhance mode.
11:0	R/W	0x14c	low1_delay data transfer first low level delay,Used to repair timing. Valid in the enhance mode.

**Table 13-70 I2C\_M\_A\_HIGH\_DELY\_REG**

Bit(s)	R/W	Default	Description
31:24	R/W	0	Reserved
23:12	R/W	0xA6	high2_delay data transfer second high level delay,Used to repair timing. Valid in the enhance mode.
11:0	R/W	0xA6	high1_delay data transfer first high level delay,Used to repair timing. Valid in the enhance mode.

**Table 13-71 I2C\_M\_A\_FIFO\_CTRL0 0x0D**

Bit(s)	R/W	Default	Description
30	R/W	0x00	reg_fifo_mode
24	R/W	0x00	reg_fifo_timeout_stop
23	R/W	0x00	reg_fifo_timeout_clr
22	R/W	0x00	reg_fifo_timeout_en
21	R/W	0x00	reg_fifo_irq_mode :0 pluse 1 level
20	R/W	0x00	reg_fifo_err_stop
18	R/W	0x00	reg_rd_fifo_rst
17	R/W	0x00	reg_wd_fifo_rst
16	R/W	0x00	reg_tk_fifo_rst

**Table 13-72 I2C\_M\_A\_FIFO\_CTRL1 0x0E**

Bit(s)	R/W	Default	Description
31:16	R/W	0x00	reg_fifo_timeout_thd
11:8	R/W	0x00	reg_rd_fifo_thd
7:4	R/W	0x00	reg_wd_fifo_thd
2:0	R/W	0x00	reg_tk_fifo_thd

**Table 13-73 I2C\_M\_A\_FIFO\_PENDING 0x0F**

Bit(s)	R/W	Default	Description
11	R/W	0x00	tk_fifo_underflow_pending, write 1 to clear pending
10	R/W	0x00	tk_fifo_overflow_pending, write 1 to clear pending
9	R/W	0x00	wd_fifo_underflow_pending, write 1 to clear pending
8	R/W	0x00	wd_fifo_overflow_pending, write 1 to clear pending
7	R/W	0x00	rd_fifo_underflow_pending, write 1 to clear pending
6	R/W	0x00	rd_fifo_overflow_pending, write 1 to clear pending
5	R/W	0x00	fifo_timeout_pending, write 1 to clear pending
4	R/W	0x00	rd_fifo_thd_irq_pending, write 1 to clear pending
3	R/W	0x00	wd_fifo_thd_irq_pending, write 1 to clear pending
2	R/W	0x00	tk_fifo_thd_irq_pending, write 1 to clear pending
1	R/W	0x00	error_pending, write 1 to clear pending
0	R/W	0x00	i2c_st_in_idle_pending, write 1 to clear pending

**Table 13-74 I2C\_M\_A\_FIFO\_PENDING\_MASK 0x10**

Bit(s)	R/W	Default	Description
11:0	R/W	0x00	pending mask

**Table 13-75 I2C\_M\_A\_FIFO\_ST0 0x11**

Bit(s)	R/W	Default	Description
31:28	RO	0x00	wd_data_valid
26:24	RO	0x00	rd_data_valid
23:21	RO	0x00	rdata_din_valid
20:16	RO	0x00	tk_data_valid
14:12	RO	0x00	i2c_st
10:8	RO	0x00	token_count
7:4	RO	0x00	wdata_count
3:0	RO	0x00	rdata_count

## 13.5 Universal Asynchronous Receiver And Transmitter

### 13.5.1 Overview

A universal asynchronous receiver-transmitter (UART) is a computer hardware device for asynchronous serial communication in which the data format and transmission speeds are configurable. It sends data bits one by one, from the least significant to the most significant, framed by start and stop bits so that precise timing is handled by the communication channel. The UART is an Advanced Micro controller Bus Architecture (AMBA) compliant System-on-Chip (SoC) peripheral. The UART is an AMBA slave module that connects to the Advanced Peripheral Bus (APB).

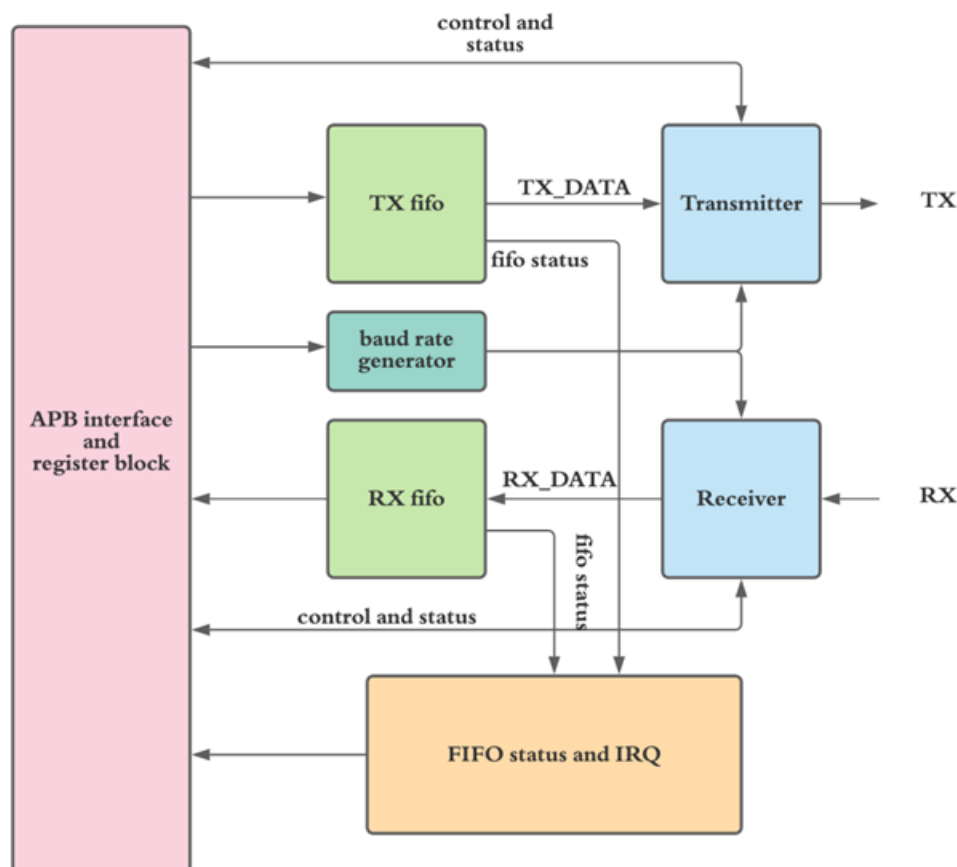
The UART performs:

- serial-to-parallel conversion on data received from a peripheral device
- parallel-to-serial conversion on data transmitted to the peripheral device.

The CPU reads and writes data and control/status information through the AMBA APB interface. The transmit and receive paths are buffered with internal FIFO memories enabling up to 64/128-bytes to be stored independently in both transmit and receive modes. The FIFO's are filled by the CPU and read by the CPU. In some cases, the receive FIFO can be configured to be pushed directly to DDR memory without CPU intervention.

There are a number of UARTs in the chip that offer 2-wire (RX/TX) and 4-wire (RX/TX, CTS/RTS) connections at the digital I/O pins. Each UART contains one transmit FIFO and a receive FIFO (see depths below). The FIFO's are filled by the CPU and read by the CPU.

Figure 13-2 UART TOP Diagram



## 13.5.2 Features

### Input filters

The CTS (clear to send) and RX (receive) input paths have input filters to deal with slow rise times. The filters are configurable to use a 125ns or 1us sampling mechanism. There is an implied 3 system clock cycle delay (15ns for a typical system clock of 200MHz) that is used to synchronize and detect the rising/falling edge of the RXD signal. The RXD signal may be passed through an optional filter to deglitch the external signal in noisy conditions. The deglitch filter has two settings which add to the “detection delay” of the RXD signal by the internal logic:

- Filter setting 1 (125ns strobe): 375ns ~ 2.6us
- Filter setting 2 (1us strobe): 3us ~ 21us

The filter is described in the register specification. If the filter is disabled, the shortest RXD low time and high time is 12 system clock cycles (60ns for a system clock of 200MHz).

### Clear to Send

CTS is a signal sent from the receiver UART back to the transmitting UART to tell the transmitting UART to stop sending data. The CTS signal must be received before the next START symbol is sent. The transmitting UART is allowed to send one more byte after the CTS signal is recognized. The CTS signal coming into the chip goes through some synchronization and detection which adds an additional 5 system clocks (typically 25ns for a 200MHz system clock). This setup time for CTS detection is called CTSstop. The CTS input also has an optional filter can be used to deglitch the incoming CTS signal. If the filter is disabled, the CTS signal must be de-asserted 5 system clock cycles before the start of the

next BYTE transfer. If the CTS filter is enabled, then additional time must be added to the 25ns requirement. There are two programmable filter settings that effectively delay CTS being seen by the internal logic:

- Filter setting 1 (125ns strobe): 375ns ~ 2.6us
- Filter setting 2 (1us strobe): 3us ~ 21us

## Interrupts

The UARTs can generate interrupts if the receive FIFO exceeds a pre-programmed threshold. An interrupt can also be generated if there is a frame or parity error.

## Clock Independent Operation

Because the system clock can be altered to accommodate dynamic frequency scaling, the UARTs have an option in which they use the 24Mhz crystal clock as the source for the UART.

## 13.5.3 Functional Description

The UART requires that a Baud Rate be established. The UART supports rates as slow as 1Hz up to rates as high as 8M bits/s. Once the baud rate has been established, bytes are transmitted as they are written to the transmit-FIFO by the CPU. A large transmit-FIFO exists to allow the CPU to pre-load a transmit package because the CPU can often write faster than the UART can transmit the data.

Data this automatically received by the UART is placed into the receive FIFO one byte at a time. The receive-FIFO decouples the UART from the CPU allowing the CPU to read the UART byte data at a rate not dictated by the UART.

## 13.5.4 Register Description

The following register description is uniformly applied to all UART instantiations in the chip.

- UART\_A\_WFIFO 0xfe078000
- UART\_A\_RFIFO 0xfe078004
- UART\_A\_CONTROL 0xfe078008
- UART\_A\_STATUS 0xfe07800c
- UART\_A\_MISC 0xfe078010
- UART\_A\_REG5 0xfe078014
- UART\_B\_WFIFO 0xfe07a000
- UART\_B\_RFIFO 0xfe07a004
- UART\_B\_CONTROL 0xfe07a008
- UART\_B\_STATUS 0xfe07a00c
- UART\_B\_MISC 0xfe07a010
- UART\_B\_REG5 0xfe07a014
- UART\_C\_WFIFO 0xfe07c000
- UART\_C\_RFIFO 0xfe07c004
- UART\_C\_CONTROL 0xfe07c008
- UART\_C\_STATUS 0xfe07c00c
- UART\_C\_MISC 0xfe07c010
- UART\_C\_REG5 0xfe07c014
- UART\_D\_WFIFO 0xfe07e000
- UART\_D\_RFIFO 0xfe07e004
- UART\_D\_CONTROL 0xfe07e008

- UART\_D\_STATUS 0xfe07e00c
- UART\_D\_MISC 0xfe07e010
- UART\_D\_REG5 0xfe07e014
- UART\_E\_WFIFO 0xfe080000
- UART\_E\_RFIFO 0xfe080004
- UART\_E\_CONTROL 0xfe080008
- UART\_E\_STATUS 0xfe08000c
- UART\_E\_MISC 0xfe080010
- UART\_E\_REG5 0xfe080014
- UART\_F\_WFIFO 0xfe082000
- UART\_F\_RFIFO 0xfe082004
- UART\_F\_CONTROL 0xfe082008
- UART\_F\_STATUS 0xfe08200c
- UART\_F\_MISC 0xfe082010
- UART\_F\_REG5 0xfe082014

**Table 13-76 UARTx\_WFIFO**

Bit(s)	R/W	Default	Description
31-8	R	0	unused
7-0	R/W	-	Write FIFO data. The Write FIFO holds 128 or 64 bytes. The Write FIFO can be written as long as it is not full.

**Table 13-77 UARTx\_RFIFO**

Bit(s)	R/W	Default	Description
31-8	R	0	unused
7-0	R/W	-	Read FIFO data. The Read FIFO holds 128 or 64 bytes. The empty flag can be used to determine if data is available

**Table 13-78 UARTx\_CONTROL**

Bit(s)	R/W	Default	Description
31	R/W	0	Invert the RTS signal
30	R/W	0	Mask Error: Set to 1 to mask errors
29	R/W	0	Invert the CTS signal
28	R/W	0	Transmit byte Interrupt: Set to 1 to enable the generation an interrupt whenever a byte is read from the transmit FIFO
27	R/W	0	Receive byte Interrupt: Set to 1 to enable the generation an interrupt whenever a byte is written to the receive FIFO
26	R/W	0	Set to 1 to invert the TX pin
25	R/W	0	Set to 1 to invert the RX pin
24	R/W	0	Clear Error
23	R/W	0	Reset the receive state machine
22	R/W	0	Reset the transmit state machine

Bit(s)	R/W	Default	Description
21-20	R/W	0	Character length: 00 = 8 bits, 01 = 7 bits, 10 = 6 bits, 11 = 5 bits
19	R/W	1	Parity Enable: Set to 1 to enable parity
18	R/W	0	Parity type: 0 = even, 1 = odd
17-16	R/W	0	Stop bit length: 00 = 1 bit, 01 = 2 bits
15	R/W	0	Two Wire mode:
14	R/W	0	Unused
13	R/W	0	Receive Enable. Set to 1 to enable the UART receive function
12	R/W	0	Transmit Enable. Set to 1 to enable the UART transmit function
11-0	R/W	0x120	Old Baud rate:

Table 13-79 UARTx\_STATUS

Bit(s)	R/W	Default	Description
31-27	R	0	Unused
26	R	0	UART_RECV_BUSY: This bit will be 1 if the uart receive state machine is busy
25	R	0	UART_XMIT_BUSY: This bit will be 1 if the uart transmit state machine is busy
24	R	0	RECV_FIFO_OVERFLOW:
23	R	0	CTS Level
22	R	0	Transmit FIFO Empty
21	R	0	Transmit FIFO Full
20	R	0	Receive FIFO empty
19	R	0	Receive FIFO full
18	R	0	This bit is set if the FIFO is written when it is full. To clear this bit, write bit 24 of register 0x2132
17	R	0	Frame error. To clear this bit, write bit 24 of register UART0_CONTROL
16	R	0	Parity error. To clear this bit, write bit 24 of register UART0_CONTROL
15	R	0	Unused
14-8	R	0	Transmit FIFO count. Number of bytes in the transmit FIFO
7	R	0	Unused
6-0	R	0	Receive FIFO count. Number of bytes in the receive FIFO

Table 13-80 UARTx\_MISC

Bit(s)	R/W	Default	Description
31	R/W	0	Added a "just in case" bit that can be set to 1 to enable clocks always. The default is 0 meaning the auto-clock gating logic is enabled.
30	R/W	0	USE old Rx Baud: There was a bug in the RX baud rate generator. The Rx baud rate generator was re-designed to compute a baud rate correctly. If you want to use the old (stupid) logic, you can set this bit to 1.

Bit(s)	R/W	Default	Description
29	R/W	0	ASYNC_FIFO_PURGE: This bit can be set to 1 after all UART bytes have been received in order to purge the data into the Async FIFO. This bit is needed because the UART receives 8-bit data, but the ASYNC FIFO can only be written with 16-bit data. In this case there might be a residual byte if the UART is not receiving an even number of bytes.
28	R/W	0	ASYNC_FIFO_EN: If this bit is set to 1, then the UART received data is automatically sent to the Async FIFO module which will in turn automatically send the data to DDR memory
27	R/W	0	CTS: Filter Timebase select: 1 = 1uS, 0 = 111nS timebase. A filter was added to the CTS input to allow for a little digital filtering. The amount of filtering is controlled by this timebase (longer = more filtering) and the value in bits FILTER_SEL below.
26-24	R/W	0	CTS: FILTER_SEL: 0 = no filter, 7 = max filtering
23-20	R/W	0	Old BAUD_RATE_EXT: These 4 bits extend the baud rate divider to 16-bits: Baud Rate = {Reg4[23:20],Reg2[11:0]}
19	R/W	0	RX: Filter Timebase select: 1 = 1uS, 0 = 111nS timebase. A filter was added to the RX input to allow for a little digital filtering. The amount of filtering is controlled by this timebase (longer = more filtering) and the value in bits FILTER_SEL below.
18-16	R/W	0	RX: FILTER_SEL: 0 = no filter, 7 = max filtering
15-8	R/W	32	XMIT_IRQ_CNT: The UART can be configured to generate an interrupt if the number of bytes in the transmit FIFO drops below this value.
7:0	R/W	15	RECV_IRQ_CNT: The UART can be configured to generate an interrupt after a certain number of bytes have been received by the UART.

Table 13-81 UARTx\_REG5

Bit(s)	R/W	Default	Description
31-28	R/W	0	unused
27	R/W	0	Xtal2_clk_sel: 0: see Xtal_clk_sel 1: xtal_div2(12M)
26	R/W	0	Xtal_clk_sel: 0: xtal_div3(8M); 1: xtal(24M);(need set xtal_tick_en = 1 first);
24	R/W	0	USE_XTAL_CLK: If this bit is set, then the clock for generating the UART Baud rate comes from the crystal pad. This allows the UART to operate independent of clk81.
23	R/W	0	USE New Baud rate. Over the years, the baud rate has been extended by concatenating bits from different registers. To take advantage of the full 23-bit baud rate generate (extended to 23 bits to accommodate very low baud rates), you must set this bit. If this bit is set, then the baud rate is configured using bits [22:0] below
22:0	R/W	15	NEW_BAUD_RATE: If bit[23] = 1 above, then the baud rate for the UART is computed using these bits.
31-28	R/W	0	unused
27	R/W	0	Xtal2_clk_sel: 0: see Xtal_clk_sel 1: xtal_div2(12M)
26	R/W	0	Xtal_clk_sel: 0: xtal_div3(8M);



Bit(s)	R/W	Default	Description
			1: xtal(24M);(need set xtal_tick_en =1 first);
24	R/W	0	USE_XTAL_CLK: If this bit is set, then the clock for generating the UART Baud rate comes from the crystal pad. This allows the UART to operate independent of clk81.
23	R/W	0	USE New Baud rate. Over the years, the baud rate has been extended by concatenating bits from different registers. To take advantage of the full 23-bit baud rate generate (extended to 23 bits to accommodate very low baud rates), you must set this bit. If this bit is set, then the baud rate is configured using bits [22:0] below
22:0	R/W	15	NEW_BAUD_RATE: If bit[23] = 1 above, then the baud rate for the UART is computed using these bits. This was added in M6 to accommodate lower baud rates.

## 13.6 Infrared Remote

### 13.6.1 Overview

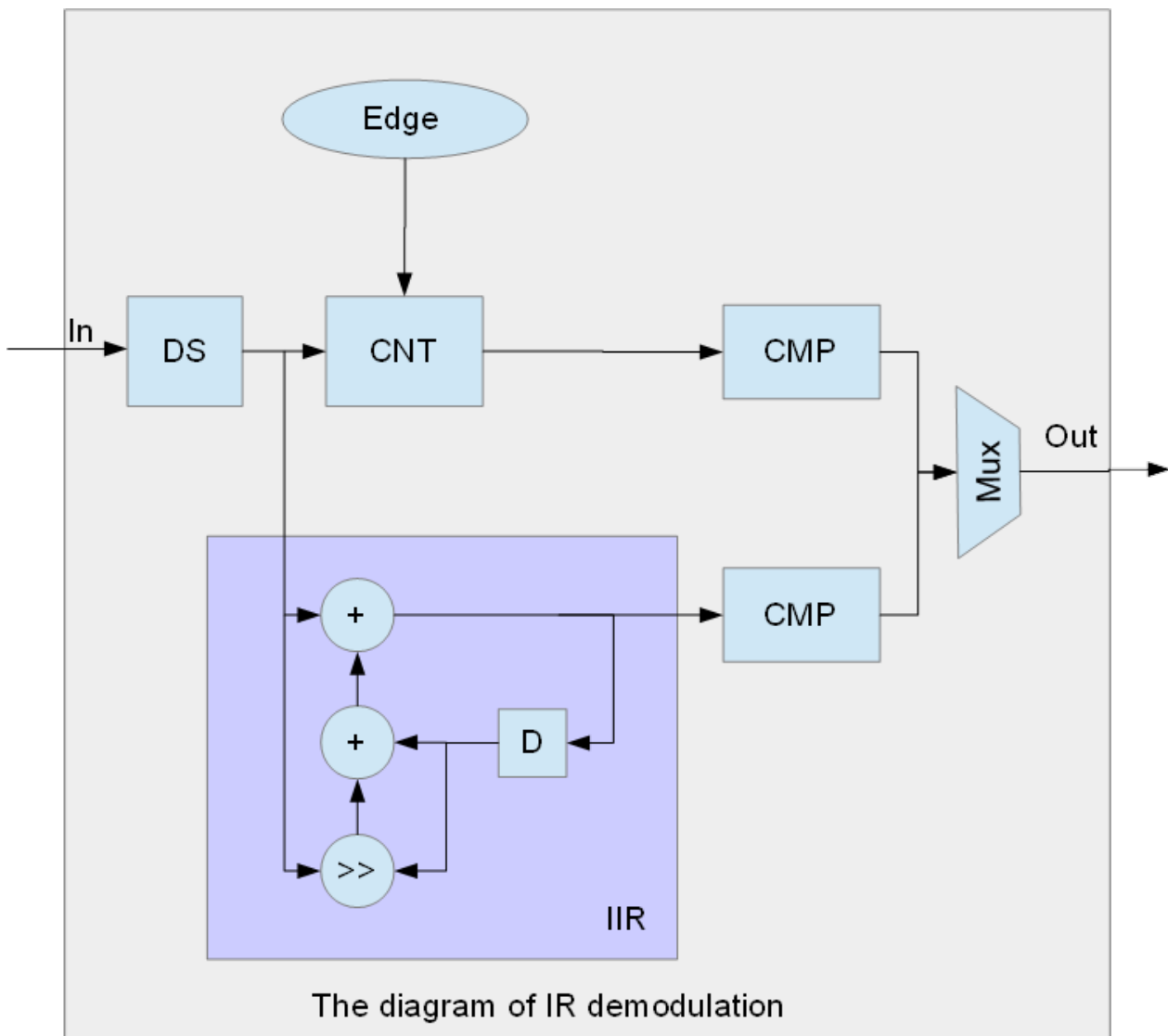
IR module includes 3 sub modules: IR demodulation, Legacy IR remote control and Multi-format IR remote control.

### 13.6.2 IR Demodulation

IR demodulation module demodulate the modulate signal to get the envelop signal.

IR demodulation diagram is shown below.

Figure 13-3 IR Demulation



It implements two methods for demodulation. The input data is an one-bit stream.

The input data are down-sampled firstly. The rate of downsample are configured by register, ranges from 0 to 255.

One method is implemented in the upper path.

The CNT is used to counte the time for '0' and '1'. It is cleared when input data changes (from 1 to 0 or from 0 to 1).

When  $(CNT > REG\_IR\_PROCT1 \ \&\& \ in == 1)$ , it outputs '1'. REG\_IR\_PROCT1 is used to canceled the glitch. Actually, the operation of downsample also have some utility to cancel the glitch.

When  $(CNT > REG\_IR\_DECT0 \ \&\& \ in == 0)$ , it outputs '0'. REG\_IR\_DECT0 is used to confirm the '0'. The value should be bigger than one carrier period.

The other method is implemented in the lower path.

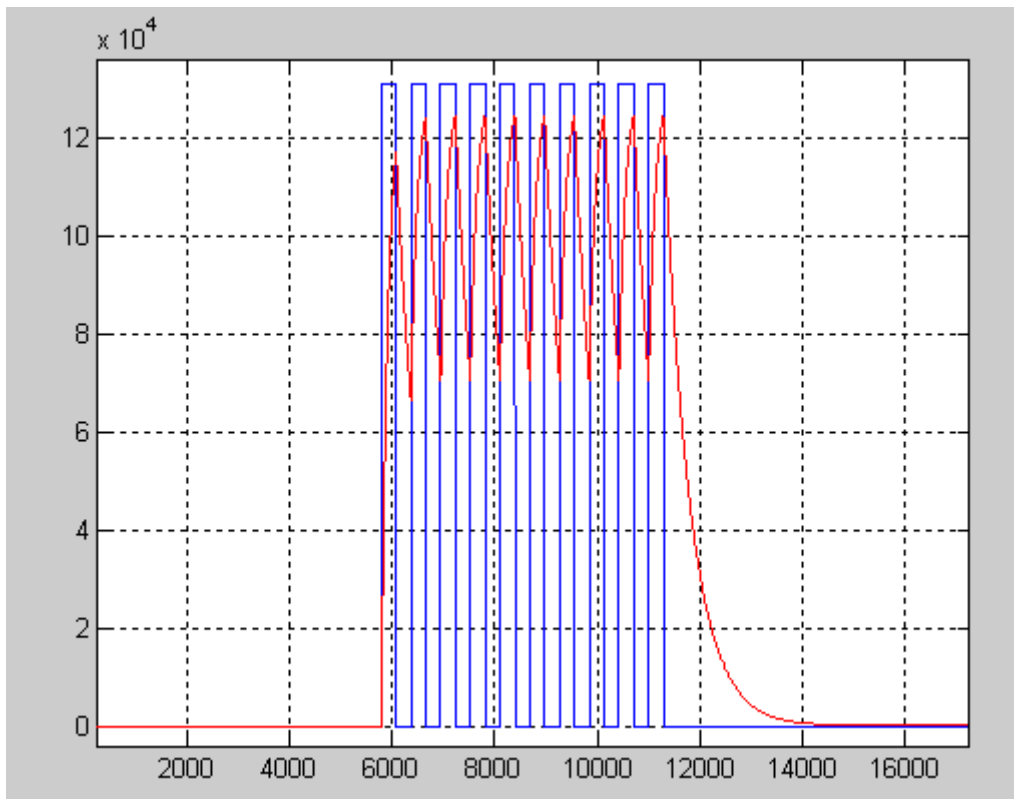
It is a one tap IIR filter. The input data control the integrating factor and the factor can be configured from registers. When 1 inputs, the accumulator can grow quickly. When 0 input, the accumulator decays slowly, and when the number of 0 is bigger enough, it decays soon.

When  $(accum > REG\_IR\_IIR\_THD1)$ , it outputs '1'.

When ( $\text{accum} < \text{REG\_IR\_IIR\_THD0}$ ), it outputs '0'.

The diagram shows the simple simulation result. The blue is the input data, the red is the filter out.

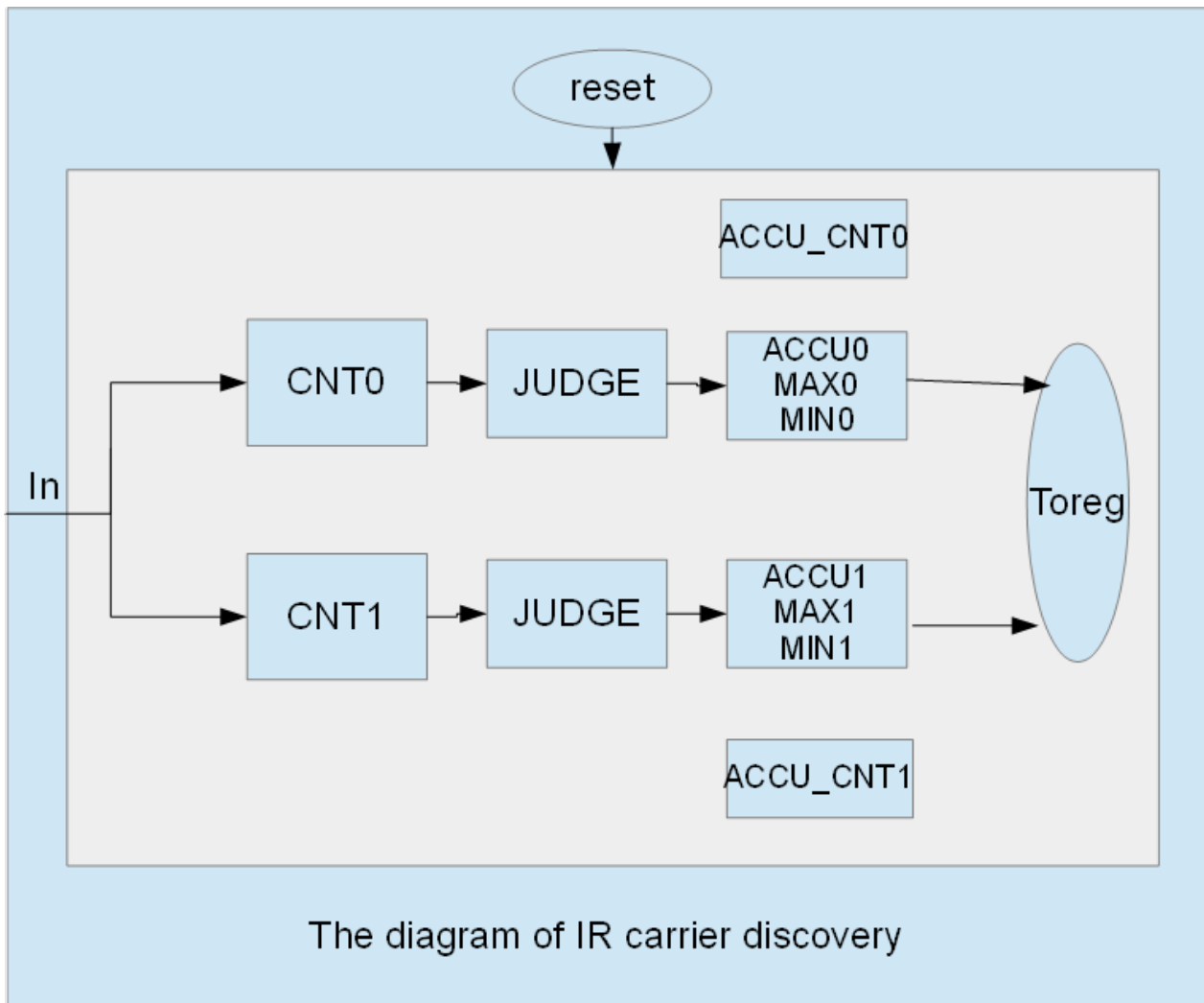
**Figure 13-4 Simulation Result of IR Demulation**



Because the input carrier signal is a rectangular wave signal, the carrier can be estimated easily by counting the wave period. For estimating precisely, the counter for the long "0" and start period and glitch should be removed.

The simple diagram is shown below:

Figure 13-5 IR Carrier Discovery



CNT is the counter for time of “0” and “1”.

JUDGE is used to remove the long “0” and glitch.

ACCU\_CNT is the counter for number of accumulating. If  $ACCU\_CNT == REG\_IR\_ST\_CNT\_THD$ , then write the value to registers.

For example:

Set  $REG\_IR\_ST\_CNT\_THD$  to 256, then accumulates 256 times.

The carrier can be calculated:

$$F_c = F_{sys} / (RO\_IR\_SUM\_CNT0 + RO\_IR\_SUM\_CNT1) * 256;$$

$$DUT = RO\_IR\_SUM\_CNT1 / (RO\_IR\_SUM\_CNT0 + RO\_IR\_SUM\_CNT1);$$

Where  $F_c$  is the carrier frequency,  $F_{sys}$  is the system clock frequency, DUT is the carrier duty ratio.

### 13.6.3 Legacy IR Control

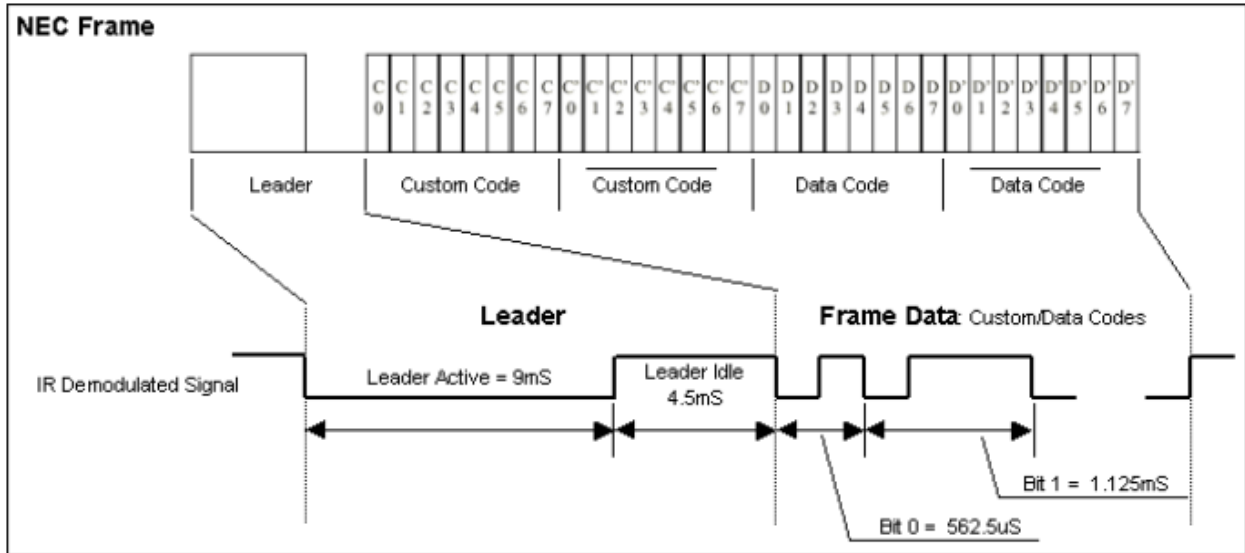
The Legacy IR Remote control module has two modes of operation:

- NEC Frame decoder mode
- General Time Measurement mode

**Note**

NEC Frame Decoder: The NEC Frame Decoder mode operates by analyzing the waveform of a TV remote.

**Figure 13-6 NEC Frame Decoder**



The waveform has a number of components, each of which must fit within a time window to be considered valid. If the entire waveform meets the specifications described by the registers below, then the TV remote codes are captured and an interrupt is generated.

**Note**

General Time Measurement: Some remotes don't follow the standard NEC format, so additional registers provide the ability to measure the time between rising and/or falling edges of the IR signal. Since the time measurement is done in hardware, the software only needs to read a "width" measurement from a register for every rising and/or falling edge event.

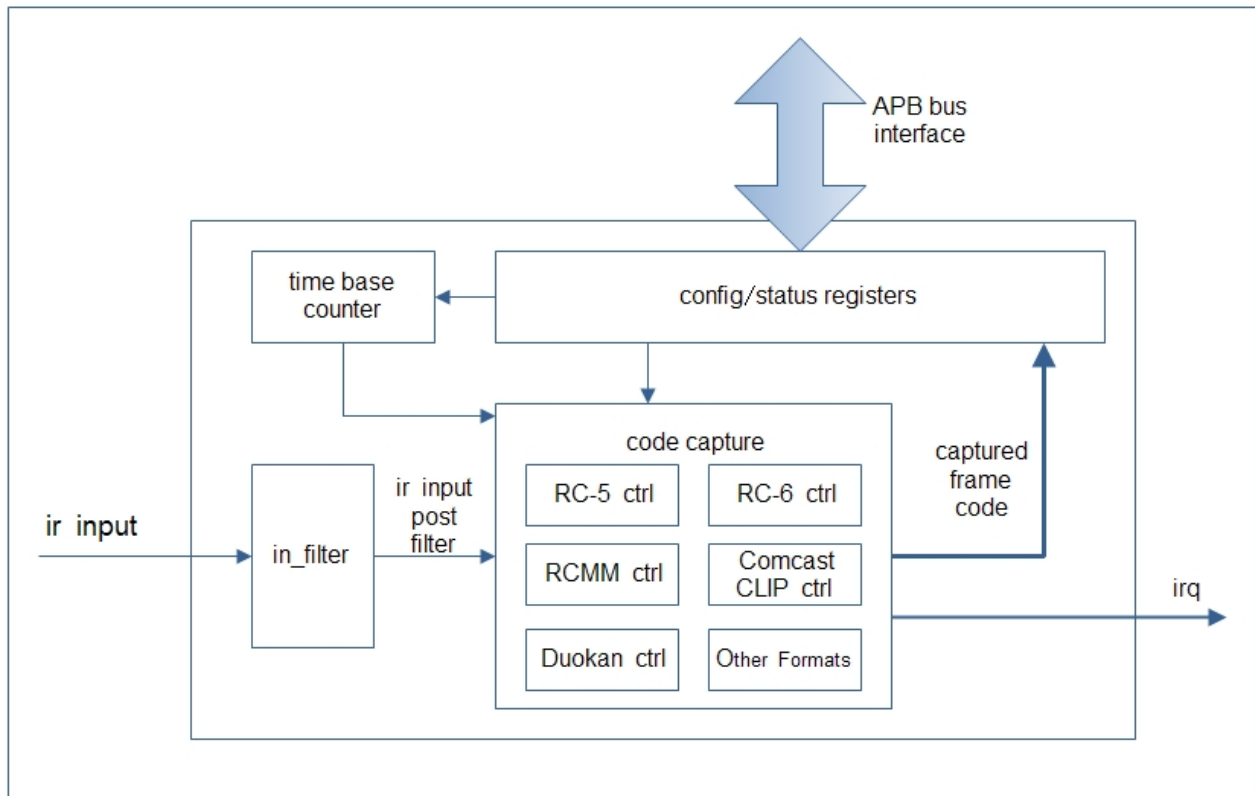
**13.6.4 Multi Format IR Control**

The decoder mainly consisted of two blocks:

- Decoder with input filter
- A set of registers including control & clock, data and tuning

The function diagram of IR decoder is illustrated in the figure below.

Figure 13-7 IR Decoder Function Block



IR Decoder decodes the IR remote control input signal. 13 operation modes are supported:

- Hardware Decode IR transmission protocol compatible frame decoder mode ( NEC MITSUBISHI Thomson Toshiba Sony SIRC RC5 RC6 RCMM Duokan Comcast Sanyo Modes)
- General programmable time measurement frame decoder mode (General Mode)

In Hardware Decode Mode, the Decoder uses signal pattern search mechanism to decode data frame. It can detect logical "0", "1", "00", "01", "10" and "11", as well as data frame start and end. Whenever Decoder detects and decodes the data frame, the data are kept in data register.

In General Mode, the Decoder uses edge detection mechanism to decode data frame. It can detect each input signal edge and record the time between two edges. The time measurement result is kept in control register.

The user should set proper operation mode corresponding to the selection of remote controller.

There is a simple time-based signal Filter between the signal input and the Decoder. The Filter is programmable and helps to improve signal integrity.

### 13.6.5 Register Description

For the following registers:

Base\_adr: 0xfe084000

Final\_adr = base\_adr + offset \*4

The following lists describe the mapping between each IR RX/TX register and its address.

IRCTRL_IR_DEC_LDR_ACTIVE	0xfe084000
IRCTRL_IR_DEC_LDR_IDLE	0xfe084004
IRCTRL_IR_DEC_LDR_REPEAT	0xfe084008

IRCTRL_IR_DEC_BIT_0	0xfe08400c
IRCTRL_IR_DEC_REG0	0xfe084010
IRCTRL_IR_DEC_FRAME	0xfe084014
IRCTRL_IR_DEC_STATUS	0xfe084018
IRCTRL_IR_DEC_REG1	0xfe08401c
IRCTRL_MF_IR_DEC_LDR_ACTIVE	0xfe084040
IRCTRL_MF_IR_DEC_LDR_IDLE	0xfe084044
IRCTRL_MF_IR_DEC_LDR_REPEAT	0xfe084048
IRCTRL_MF_IR_DEC_BIT_0	0xfe08404c
IRCTRL_MF_IR_DEC_REG0	0xfe084050
IRCTRL_MF_IR_DEC_FRAME	0xfe084054
IRCTRL_MF_IR_DEC_STATUS	0xfe084058
IRCTRL_MF_IR_DEC_REG1	0xfe08405c
IRCTRL_MF_IR_DEC_REG2	0xfe084060
IRCTRL_MF_IR_DEC_DURATN2	0xfe084064
IRCTRL_MF_IR_DEC_DURATN3	0xfe084068
IRCTRL_MF_IR_DEC_FRAME1	0xfe08406c
IRCTRL_MF_IR_DEC_STATUS1	0xfe084070
IRCTRL_MF_IR_DEC_STATUS2	0xfe084074
IRCTRL_MF_IR_DEC_REG3	0xfe084078
IRCTRL_MF_IR_DEC_FRAME_RSV0	0xfe08407c
IRCTRL_MF_IR_DEC_FRAME_RSV1	0xfe084080
IRCTRL_MF_IR_DEC_FILTE	0xfe084084
IRCTRL_MF_IR_DEC_IRQ_CTL	0xfe084088
IRCTRL_MF_IR_DEC_FIFO_CTL	0xfe08408c
IRCTRL_MF_IR_DEC_WIDTH_NEW	0xfe084090
IRCTRL_MF_IR_DEC_REPEAT_DET	0xfe084094
IRCTRL_IR_DEC_DEMOD_CNTL0	0xfe0840c0
IRCTRL_IR_DEC_DEMOD_CNTL1	0xfe0840c4
IRCTRL_IR_DEC_DEMOD_IIR_THD	0xfe0840c8
IRCTRL_IR_DEC_DEMOD_THD0	0xfe0840cc
IRCTRL_IR_DEC_DEMOD_THD1	0xfe0840d0
IRCTRL_IR_DEC_DEMOD_SUM_CNT0	0xfe0840d4
IRCTRL_IR_DEC_DEMOD_SUM_CNT1	0xfe0840d8
IRCTRL_IR_DEC_DEMOD_CNT0	0xfe0840dc
IRCTRL_IR_DEC_DEMOD_CNT1	0xfe0840e0
IRCTRL_IR_BLASTER_ADDR0	0xfe08410c
IRCTRL_IR_BLASTER_ADDR1	0xfe084110

IRCTRL\_IR\_BLAster\_ADDR2 0xfe084114

IRCTRL\_IR\_BLAster\_ADDR3 0xfe084118

**Table 13-82 IRCTRL\_IR\_BLAster\_ADDR0**

Bit(s)	R/W	Default	Description
31-27	R	0	unused
26	R	-	BUSY: If this bit is 1, then the IR Blaster module is busy.
25	R	-	This output is 1 when the FIFO is Full
24	R	-	This output is 1 when the FIFO is Empty
23-16	R	-	FIFO Level
15-14	R/W	0	Unused
13-12	R/W	0	MODULATOR_TB: This input controls the clock used to create the modulator output. The modulator is typically run between 32khz and 56khz. The modulator output will equal a divided value of the following: 00: system clock "clk" 01: mpeg_xtal3_tick 10: mpeg_1uS_tick 11: mpeg_10uS_tick
11-4	R/W	0	SLOW_CLOCK_DIV: This is a divider value used to divide down the input "clk". The divider is N+1 so a value of 0 equals divide by 1.
3	R/W	0	SLOW_CLOCK_MODE: Set this signal high to use a special mode in which the "clk" input is driven by a slow clock less than 1Mhz. This is used for low power cases where we want to run the IR Blaster between 32khz and 1Mhz
2	R/W	0	INIT_LOW: Setting this bit to 1 initializes the output to be high. Please set this bit back to 0 when done
1	R/W	0	INIT_LOW: Setting this bit to 1 initializes the output to be low. Please set this bit back to 0 when done
0	R/W	0	ENABLE: 1 = Enable. If this bit is set to 0, then the IR blaster module is reset and put into an IDLE state.

**Table 13-83 IRCTRL\_IR\_BLAster\_ADDR1**

Bit(s)	R/W	Default	Description
31-28	R/W	0	unused
27-16	R/W	0	This value is used with "modulator_tb[1:0]" above to create a low pulse. The time is computed as (mod_lo_count+1) x modulator_tb. The purpose for having a low/high count is the modulator output might not be 50% duty cycle. Hi/Lo counters allow us to modulate using a non-50% duty cycle waveform.
15-12	R/W	0	Unused
11-0	R/W	0	This value is used with "modulator_tb[1:0]" above to create a high pulse. The time is computed as (mod_hi_count+1) x modulator_tb

**Table 13-84 IRCTRL\_IR\_BLAster\_ADDR2**

Bit(s)	R/W	Default	Description
31-17	R	0	unused
16	W	0	Set this bit to 1 to write the data below to the FIFO



Bit(s)	R/W	Default	Description
15-12	R	0	Unused
11-0	R/W	0	FIFO data to be written: Bit[12] output level (or modulation enable/disable: 1 = enable) Bit[11:10] Timebase: 00 = 1uS 01 = 10uS 10 = 100uS 11 = Modulator clock Bit[9:0] Count of timebase units to delay

Table 13-85 IRCTRL\_IR\_BLAZER\_ADDR3

Bit(s)	R/W	Default	Description
31-17	R	0	unused
16	R	0	fifo thd pending
8	R/W	0	FIFO irq enable
7:0	R/W	0	FIFO irq threshold.

Table 13-86 IRCTRL\_IR\_DEC\_DEMOD\_CNTL0

Bits	R/W	Default	Description
31	R/W	0	Ir demodulator soft_reset:write 1 to reset ir_demodulator,will auto clr to 0
30	R/W	0	Reg_ir_fd_reset :It is used to reset the carrier detection module.write 1 to reset , will auto clr to 0
29	R/W	0	Reg_ir_demod_mode:It is used to set the demod mode. 0 = count mode; 1 = iir mode.
28	R/W	0	Ir demodulator clk gate bypass:write 1 will bypass clk gate,default is 0
27:16	R/W	0x40	Reg_ir_st_cnt_thd:It is used to set the statistics number for carrier detection.
15:8	R/W	8	Reg_ir_ds_rate:It is used to set the downsample rate.
7:4	R/W	7	Reg_ir_fsft_1:It is used to set the shift value for input data "1".
3:0	R/W	9	Reg_ir_fsft_0:It is used to set the shift value for input data "0".

Table 13-87 IRCTRL\_IR\_DEC\_DEMOD\_CNTL1

Bits	R/W	Default	Description
31	R/W	0	Reg_ir_demod_en:ir demod enable ,
30	R/W	0	Reg_ir_invt:ir input invert, invert input at 1.
29:16	R/W	0xa	Reg_ir_proctect1:It is used to set the protection threshold for 1. It is used to filter the glitch.
13:0	R/W	0x3e8	Reg_ir_detect0:It is used to set the detection threshold for signal "0".

**Table 13-88 IRCTRL\_IR\_DEC\_DEMOD\_IIR\_THD**

Bits	R/W	Default	Description
31:16	R/W	0xdac	Reg_ir_iir_thd1:It is used to set the detection threshold for “1” in iir mode.
15:0	R/W	0xdac	Reg_ir_iir_thd0:It is used to set the detection threshold for “0” in iir mode.

**Table 13-89 IRCTRL\_IR\_DEC\_DEMOD\_THD0**

Bits	R/W	Default	Description
29:16	R/W	0x12c	Reg_ir_thd0_low:It is used to set the low threshold for “0” when statistics.
13:0	R/W	0x1770	Reg_ir_thd0_high:It is used to set the high threshold for “0” when statistics.

**Table 13-90 IRCTRL\_IR\_DEC\_DEMOD\_THD1**

Bits	R/W	Default	Description
29:16	R/W	0x12c	Reg_ir_thd1_low:It is used to set the low threshold for “1” when statistics.
13:0	R/W	0x1770	Reg_ir_thd1_high:It is used to set the high threshold for “1” when statistics.

**Table 13-91 IRCTRL\_IR\_DEC\_DEMOD\_SUM\_CNT0**

Bits	R/W	Default	Description
25:0	R	0	Ro_ir_sum_cnt0:It is used to report the sum value for the statistics data “0”.

**Table 13-92 IRCTRL\_IR\_DEC\_DEMOD\_SUM\_CNT1**

Bits	R/W	Default	Description
25:0	R	0	Ro_ir_sum_cnt1:It is used to report the sum value for the statistics data “1”.

**Table 13-93 IRCTRL\_IR\_DEC\_DEMOD\_CNT0**

Bits	R/W	Default	Description
29:16	R	0	Ro_ir_max_cnt0:It is used to report the maximum value for the statistics data “0”.
13:0	R	0	Ro_ir_min_cnt0:It is used to report the minimum value for the statistics data “0”.

**Table 13-94 IRCTRL\_IR\_DEC\_DEMOD\_CNT1**

Bits	R/W	Default	Description
29:16	R	0	Ro_ir_max_cnt1:It is used to report the maximum value for the statistics data “1”.
13:0	R	0	Ro_ir_min_cnt1:It is used to report the minimum value for the statistics data “1”.

**Table 13-95 IRCTRL\_IR\_DEC\_LDR\_ACTIVE: Leader Active Time**

Bit(s)	R/W	Default	Description
31-29	R	0	unused
28-16	R/W	0x 1d8	Max Leader ACTIVE time: 9.44mS assuming base rate = 20uS
15-13	R	0	unused
12-0	R/W	0x1ac	Min Leader ACTIVE time: 8.56mS assuming base rate = 20uS

This register controls the min/max leader active time window. For most TV remote controls, the leader active time is about 9mS. The values in this register correspond to counts of the base rate programmed by register 0x2124

**Table 13-96 IRCTRL\_IR\_DEC\_LDR\_IDLE: Leader Idle Time**

Bit(s)	R/W	Default	Description
31-26	R	0	unused
28-16	R/W	0xf8	Max Leader IDLE time: 4.96mS assuming base rate = 20uS
15-13	R	0	unused
12-0	R/W	0xca	Min Leader IDLE time: 4.04mS assuming base rate = 20uS

This register controls the min/max leader IDLE time window. For most TV remote controls, the leader idle time is about 4.5mS. The values in this register correspond to counts of the base rate programmed by register: 0x2124

**Table 13-97 IRCTRL\_IR\_DEC\_LDR\_REPEAT: Repeat Leader Idle Time**

Bit(s)	R/W	Default	Description
31-26	R	0	Unused
25-16	R/W	0x7a	Max REPEAT Leader IDLE time: 2.44mS assuming base rate = 20uS
15-10	R	0	unused
9-0	R/W	0x66	Min REPEAT Leader IDLE time: 2.04mS assuming base rate = 20uS

This register controls the repeat leader IDLE time window. The repeat key uses the standard leader active time (9ms) but a shorter leader idle time.

**Table 13-98 IRCTRL\_IR\_DEC\_BIT\_0: BIT 0 Identification Time**

Bit(s)	R/W	Default	Description
31-26	R	0	Unused
25-16	R/W	0x42	Max BIT 0 time: 1.32mS assuming base rate = 20uS
15-10	R	0	Unused
9-0	R/W	0x2e	Min BIT 0 time: 0.92mS assuming base rate = 20uS

This register controls the min/max BIT 0 time window. For most TV remote controls, the bit 0 time is about 1.125mS. The values in this register correspond to counts of the base rate programmed by register: 0x2124

**Table 13-99 IRCTRL\_IR\_DEC\_REG0: Base Rate Generator**

Bit(s)	R/W	Default	Description
31	R	0	Just in case bit. Normally this bit is set to 0 so that the auto-clock gating is enabled. If there is a problem related to the auto-clock gating, then this bit can be set to one to disable the auto-clock gating.
30-28	R/W	0	<b>FILTER_COUNT:</b> This is a new feature to Nike. The IR remote input now has a simple filter that accommodates slow rise times by providing a little hysteresis. The logic works as follows. If the input is low, then an input signal will only be considered HIGH if it remains high for (FILTER_COUNT * 111nS). Similarly, if the input is currently high, it will only be considered LOW if the input signal remains low for (FILTER_COUNT * 111nS).
27-25	R	0	Unused
24-12	R/w	0xFA0	Max Frame Time. 80mS assuming base rate = 20uS This value is used to determine if a code is a repeat code (e.g. leader followed by no data for this amount of time). This value can also be used to catch slow remote codes (i.e. code sequences that are longer than expected).
11-0	R/W	0x13	This value dictates the base rate time for all measurements associated with the IR decoder. In the past, the base rate was divided from the system clock. In the current design, the base rate is divided from a fixed 1uS timer. This 1uS timer is constant and doesn't change (even when the system clock does) Base rate = (count + 1) * 1uS

This register controls the master rate generator for all width measurements made by the IR decoder module.

**Table 13-100 IRCTRL\_IR\_DEC\_FRAME: Frame Data**

Bit(s)	R/W	Default	Description
31-0	R	0	Frame Data. Format: {custom code, ~custom code, data, ~data}

**Note**

New keys will be ignored until this register is read if the hold first key bit is set in the decode control register. Reading this register resets an internal hold first flag.

**Table 13-101 IRCTRL\_IR\_DEC\_STATUS: Frame Status**

Bit(s)	R/W	Default	Description
31	R/W	0	Sim faster: Reserved
30	R/W	0	<b>BIT_1_MATCH_EN:</b> Set this bit to 1 to enable qualification of bit 1 times. In the previous IR decoder module, frame detection only looked at the BIT 0 time to identify a zero bit. If a zero bit time wasn't found, then it was assumed that the bit was a 1. In the updated IR decoder module, the module will look at the BIT 0 time to find zero bits, and the BIT 1 time to find 1 bits. If the width of a pulse doesn't match the zero or the one bit width time, then the frame is considered invalid.
29-20	R/W	0x89	Max BIT 1 time: 2.74mS assuming base rate = 20uS
19-10	R/W	0x57	Min BIT 1 time: 1.74mS assuming base rate = 20uS
9	R	-	IRQ Status. 1 if there is an interrupt
8	R	-	IR Decoder input. This is the level of the digital signal coming into the IR module for decoding. This is the same as reading the I/O pad level.
7	R	-	BUSY. 1 if the decoder is busy

Bit(s)	R/W	Default	Description
6-4	R	0	Decoder Status: For debug only 000: OK 001: last frame timed out 010: leader time error (invalid IR signal) 011: repeat error (repeat leader, but other IR transitions found). 100: Invalid bit
3-0	R	0	Frame Status Bit 3: Frame data valid Bit 2: data code error (data != ~data in IR bit stream) Bit 1: custom code error (custom_code != ~custom_code in IR bit stream) Bit 0: 1 = repeat key, 0 = standard key

Table 13-102 IRCTRL\_IR\_DEC\_REG1: Decode / Interrupt Control

Bit(s)	R/W	Default	Description
31	R	0	Unused
30	R/W	0	CNTL_1uS_EQ_CLK: This bit should be set to 1 if the clk81 (system clock) is less than 50 Mhz.
29	R/W	0	CNTL_111nS_EQ_CLK: This bit should be set to 1 if the clk81 (system clock) is less than 50 Mhz.
28-16	R	0	Time measurement since the last time the internal time counter was reset by the rising and/or falling edge of the IR signal. The selection of reset on rising and/or falling edge is determined by the selection of the IRQ (Bits 3-2 below)
15	R/W	1	ENABLE: If this bit is 1, then the state-machines are enabled. If this bit is zero, then the state-machines cleanup and immediately return to idle.
14	R/W	0	USE SYSTEM CLOCK: This is a new feature. 1 = use the system clock, 0 = use the 1uS timebase tick. During normal operation, the module is setup to create a 20uS tick from the 1uS internal timebase of the chip. If the chip is configured to operate using the 32khz RTC oscillator, the 1uS timebase is invalid and therefore the 20uS timebase is invalid. In order to measure time correctly, the IR remote circuit can use the system clock (which in this case is the 32khz oscillator clock) as the master timebase.
13-9	R/W	1f	Number of bits in the IR frame (N-1)
8-7	R/W	0	Decoder mode 00: NEC Frames: Decode Leader and 32 bits 01: Only accumulate bits (skip the leader) 10: Measure Mode: The internal width measuring counter is reset on the rising and/or falling edge of the IR remote signal based on the settings of IRQ Selection below. Just before being reset, the measured width is captured and stored so that it can be read in bits [28:16] of this register. 11: NEC Frames: Decode Leader and 32 bits
6	R/W	1	Hold First Key. If this bit is set true, then the frame data register (0x2125) will only be updated if hasn't already been updated. Once updated, the frame data register will not be updated again until it has been read. This bit can be used to guarantee the first TV remote code captured will not be overwritten by subsequent transmissions from a TV remote. NOTE: You must read the frame data register to clear an internal hold first flag if this bit is set.
5-4	R/W	11	Frame mask. These bits are used to qualify frames for capture. 00: Capture all frames good or bad 01: Capture only frames where data=~data. Ignore custom codes 10: Capture only frames where custom_code = ~custom_code. Ignore data codes 11: Capture only frames where (data=~data) and (custom_code = ~custom_code)
3-2	R/W	0	IRQ Selection and width measurement reset:

Bit(s)	R/W	Default	Description
			00: IR Decoder done 01: IR input rising or falling edge detected 10: IR input falling edge detected 11: IR rising edge detected
1	R/W	0	IR Polarity. Polarity of the input signal (VD[0])
0	R/W	0	Set to 1 to reset the IR decoder. This is useful because the IR remote state machine thinks in terms of milliseconds and may take tens of milliseconds to return to idle by itself.

**Table 13-103 IRCTRL\_MF\_IR\_DEC\_LDR\_ACTIVE**

Bit(s)	R/W	Default	Description
28:16	R/W	0	Max duration of Leader's active part (The value is in unit of $T_{RATE}$ )
12:0	R/W	0	Min duration of Leader's active part (The value is in unit of $T_{RATE}$ )

**Table 13-104 IRCTRL\_MF\_IR\_DEC\_LDR\_IDLE: Leader Idle control**

Bit(s)	R/W	Default	Description
28:16	R/W	0	Max duration of Leader's idle part
12:0	R/W	0	Min duration of Leader's idle part

**Table 13-105 IRCTRL\_MF\_IR\_DEC\_LDR\_REPEAT: Repeat Leader Idle Time**

Bit(s)	R/W	Default	Description
25:16	R/W	0	Max duration of Repeat Code's Leader. In NECformat, it defines for the repeat leader's idle part. In Toshiba format, it defines for the repeat leader's second idle part (In Toshiba format, the repeat leader's first idle part has the same duration time as the normal leader idle part.)
9:0	R/W	0	Min duration of Repeat Code's Leader

**Table 13-106 IRCTRL\_MF\_IR\_DEC\_BIT\_0**

Bit(s)	R/W	Default	Description
25:16	R/W	0	Max duration of Duration Setting Register 0. It defines max timing duration for: Logic"0" for NEC/Toshiba/Sony/Thomas format or Half trailer bit for RC6 format (RC6's half trailer bit typically 888.89us) or time of Duokan/RCMM/4ppm format's Logic "00"
9:0	R/W	0	Min duration of Duration Setting Register 0.

**Table 13-107 IRCTRL\_MF\_IR\_DEC\_REG0**

Bit(s)	R/W	Default	Description
31	R/W	0	Clock gating control just in case. Set 1 can force clock gating disabled.
30:28	R/W	0	Filter ctrl. Set the monitor timing for input filter, bigger value means longer monitor time. Value 0 = no filtering.
27:25	R	0	Unused
24:12	R/W	0	Max frame time. Max duration of one whole frame.
11:0	R/W	0	Base time parameter. Used to generate the timing resolution. Resolution = (base_time_paramter + 1) * (1/ Freq_sys_clk). For example, if Frequency of sys_clk is 1Mhz, and base_time_parameter=19, Then resolution = (19+1)*(1uS) = 20uS.

**Table 13-108 IRCTRL\_MF\_IR\_DEC\_FRAME**

Bit(s)	R/W	Default	Description
31:0	R/W	0	32 bit Read-Only register stores frame body (LSB 32bit) captured from IR remote data flow, commonly includes custom/address code and data code.

**Table 13-109 IRCTRL\_MF\_IR\_DEC\_STATUS**

Bit(s)	R/W	Default	Description
31	R/W	0	Frame data valid 1. (This bit is set to 1 when a captured frame is updated/stored into "FrameBody_1" register. A read of "FrameBody_1" register will clear This bit. "FrameBody_1" register is used to store the over 32bit MSBs of the formats whose length is more than 32 bit )
30	R/W	0	bit_1_match_en. Set to 1 to enable the check of whether logic"1" bit matches timing configure during the frame input process.
29:20	R/W	0	Max Duration 1. Max duration of Duration Setting Register 1. It defines max duration for: Logic"1" for NEC/Toshiba/Sony format or Whole trailer bit for RC6 format (RC6's whole trailer bit typically 1777.78us) or time of Duokan/RCMM/4ppm format's Logic "01"
19:10	R/W	0	Min Duration 1. Min duration of Duration Setting Register 1.
9	R	0	irq_status. Appear as 1 if there is an interrupt.
8	R	0	ir_i_sync. IR remote serial input after synchronization. This is the level of the digital signal coming into the IR module for decoding. This is the same as reading the I/O pad level.
7	R	0	Busy. When =1, means state machine is active.

Bit(s)	R/W	Default	Description
6:4	R	0	Decoder_status ( for debug only ). 000: OK 001: last frame timed out 010: leader time error (invalid IR signal) 011: repeat error (repeat leader, but other IR transitions found). 100: Invalid bit
3:0	R	0	Frame status. bit 3: Frame data valid (This bit is set to 1 when a captured frame is updated/stored into "FrameBody" register. A read of "FrameBody" register will clear This bit. If store and read occurs at the same time, This bit is set to 1 in common, But if "Hold first" is set to true and this valid Bit is already 1, a read clear takes precedence and This bit is clear to 0. ) bit 2: data code error (data != ~data in IR bit stream) bit 1: custom code error (custom_code != ~custom_code in IR bit stream) bit 0: 1 = received frame is repeat key, 0 = received frame is normal key

Table 13-110 IRCTRL\_MF\_IR\_DEC\_REG1

Bit(s)	R/W	Default	Description
31	R/W	0	Set to 1 to use faster timebase. .
30	R/W	0	cntl_1us_eq_clk. Just use sys_clk to relace 1uS tick.
29	R/W	0	cntl_xtal3_eq_clk. Just use sys_clk to relace 111ns tick.
28:16	R	0	Pulse Width Counter. It stores the internal counter of pulse width duration. Commonly used as time measurement when decode_mode is set to measure width mode (software decode). Time measurement starts at the last time the internal time counter was reset by the rising and/or falling edge of the IR signal. The selection of reset on rising and/or falling edge is determined by the IRQ Selection field (Bits 3-2 below)
15	R/W	0	Enable. 1 = enable the state machine of IR decoder. 0 = disable the state machine of IR decoder.
14	R/W	0	cntl_use_sys_clk. Use sys_clk for the timebase. It's useful when sys_clk at low frequency ( such as 32Khz ) and cannot create 1uS timebase tick. 1 = use the system clock as timebase. 0 = use the 1uS timebase tick as timebase.
13:8	R/W	0	bit_length minus 1. (N-1). Used to set the value of frame body's bit length (frame body commonly includes address and data code part). If a format has 24 bit frame body, this value shall be set to 23.
7	R/W	0	Record_at_error. 1= record the frame body and status forcibly, even if data/custom code error check enabled by frame_mask and relative error occurs. 0 = if data/custom code error check enabled by frame_mask and relative error occurs, not record the frame body and status forcibly
6	R/W	0	Hold_first Used to hold the first captured frame data. If This bit is set to 1, then the "FrameBody/FrameBody_1" register will only be updated if hasn't already been updated. Once updated, the "FrameBody/FrameBody_1" register will not be updated again until it has been read. This bit can be used to guarantee the first TV remote code captured will not be overwritten by subsequent transmissions from a TV remote. NOTE: Read the "FrameBody" register can clear the internal "Frame data valid" flag, and read the "FrameBody_1" register can clear the "Frame data valid 1" flag.
5:4	R/W	0	Frame_mask.



Bit(s)	R/W	Default	Description
			Some formats' body include bit-inversed data or custom/address code for error check. 00 = ignore error check from either data or custom/address code 01= check if data code matches its inverse values, ignore error check from custom/address code 10= check if custom/address code matches its inverse values, ignore error check from data code 11= check if data and custom codes match their inverse values
3:2	R/W	0	Irq_sel. IRQ Selection and width measurement reset: 00: IR Decoder done 01: IR input rising or falling edge detected 10: IR input falling edge detected 11: IR rising edge detected
1	R/W	0	IR input polarity selection. Used to adjust/invert the polarity of IR input waveform.
0	R/W	0	Decoder Reset. Set to 1 to reset the IR decoder. This is useful because the IR remote state machine thinks in terms of milliseconds and may take tens of milliseconds to return to idle by itself.

Table 13-111 IRCTRL\_MF\_IR\_DEC\_REG2

Bit(s)	R/W	Default	Description
26	R/W	0	Width_low_enable. Enable counter record of low pulse width duration. 0 = do not force enable of width low counter record 1 = force enable of width low counter record Some IR formats' decoding need to use internal width low counter record. By default, the width low counter record is enabled automatically for related formats. This bit is used for enable forcibly just in case. Besides, if "leader plus stop bit" method is enabled for repeat detection, this bit is also need to be enabled.
25	R/W	0	Width_high_enable. Enable counter record of high pulse width duration. 0 = do not force enable of width high counter record 1 = force enable of width high counter record Some IR formats' decoding need to use internal width high counter record. By default, the width high counter record is enabled automatically for related formats. This bit is used for enable forcibly just in case.
24	R/W	0	Enable "leader plus stop bit" method for repeat detection . 0 = "leader plus stop bit" method disabled 1 = "leader plus stop bit" method enabled Some IR formats use one normal frame's leader followed by a stop bit to represset repeat. There is no frame data in this kind repeat frame. To use this method, width_low_enable ( bit[26] of 0x20 offset register) shall be set to 1, and max_duration_3 and min_duration_3 in 0x28 offset register shall be set to appropriate value for stop bit's timing duration.
21:16	R/W	0	Repeat_bit_index. These bits are used for compare bit method to set the index of the bit that is used as repeat flag. The index value can be 0 to 63. Compare bit method is one of the methods for repeat detection . Some IR formats use one bit in frame to represset whether the frame is repeat.
15	R/W	0	Running_count_tick_mode. This bit is only valid when use_clock_to_counter bit is 0. 0 = use 100uS as increasing time unit of frame-to-frame counter 1 = use 10uS as increasing time unit of frame-to-frame counter
14	R/W	0	Use_clock_to_counter. If this bit is set to 1, the running_count_tick_mode bit is ignored. 0 = do not use system clock as increasing time unit of frame-to-frame counter 1 = use system clock as increasing time unit of frame-to-frame counter

Bit(s)	R/W	Default	Description
13	R/W	0	<p>Enable frame-to-frame time counter (running-counter).            0 = frame-to-frame time counter disabled            1 = frame-to-frame time counter enabled</p> <p>If enabled, the frame-to-frame counter increases every 100uS or 10uS until it reaches its max value(all bits are 1) or it is reset. When it reaches its max value, it keeps the value until it is reset. When it is reset, it becomes zero and then begin increasing again. The counter can be reset even when it has not reached its max value. The increasing time unit can be 100uS or 10uS or system clock frequency which is set by <code>running_count_tick_mode</code> and <code>use_clock_to_counter</code> settings. When a frame's data are captured and stored into <code>FrameBody/FrameBody_1</code> register, frame-to-frame counter is reset to zero. After reset to zero, the frame-to-frame counter will begin increasing again, until it reaches its max value or it is reset.</p> <p>For repeat frame detection, users can use hardware detection by enabling compare frame or compare bit method, or users can read frame-to-frame counter to let software to make the decision.</p>
12	R/W	0	<p>Enable repeat time check for repeat detection . This bit is valid only when compare frame method or compare bit method is enabled.            0 = repeat time check disabled            1 = repeat time check enabled</p> <p>When repeat frame detection is enabled by enabling compare frame or compare bit method, the frame time interval may need to be checked in order to decide whether the frames are repeat (key pressed without release) or not. You can configure the <code>repeat_time_max</code> value by setting 0x38 offset register. If frame interval is smaller than the "repeat time max", it may be considered as repeat. If frame interval is bigger than the "repeat time max", it is considered as not repeat.</p>
11	R/W	0	<p>Enable compare frame method for repeat detection .            0 = compare frame method disabled            1 = compare frame method enabled</p> <p>Some IR formats transfer the same data frame as repeat frame when the key is kept pressed without release. For repeat detection, compare frame method can be used.</p> <p>If a new frame and the old received frame are the same and the repeat time is under the limit( frame-to-frame time counter value is smaller than the <code>repeat_time_max</code> ), the status register's <code>frame_status[0]</code> is set to 1 automatically as repeat detected flag. You can configure the <code>repeat_time_max</code> value by setting 0x38 offset register.</p>
10	R/W	0	<p>Enable compare bit method for repeat detection .            0 = compare bit method disabled            1 = compare bit method enabled</p> <p>Some IR formats use only one bit to represent whether the frame is repeat. You can compare only one bit instead of compare the whole frame for repeat detection. If compare frame method is enabled, then this bit is ignored.</p>
9	R/W	0	<p>Disable read-clear of <code>FrameBody/FrameBody_1</code>.            0 = read-clear enabled            1 = read-clear disabled</p> <p><code>FrameBody/FrameBody_1</code> registers are read-cleared in default. When these registers are read, they are cleared to zero. This bit is used to disable this read-clear feature.            ( <code>FrameBody/FrameBody_1</code> registers are used to store captured frame data ).</p>

Bit(s)	R/W	Default	Description
8	R/W	0	input stream bit order. 0 = LSB first mode (first bit in input stream is considered as LSB) 1 = MSB first mode (first bit in input stream is considered as MSB) Note: Commonly the following formats shall set 1 to enable MSB first mode ( unless you insist on LSBfirst mode for your specified use): RC5, RC5 extend, RC6, RCMM, Duokan, Comcast
3:0	R/W	0	Decode_mode.(format selection) 0x0 =NEC 0x1= skip leader (just bits, without leader) 0x2=General time measurement (measure width, software decode) 0x3=MITSUBISHI or 50560 0x4=Thomson 0x5=Toshiba 0x6=Sony SIRC 0x7=RC5 0x8=Reserved 0x9=RC6 0xA=RCMM 0xB=Duokan 0xC=Reserved 0xD=Reserved 0xE=Comcast 0xF=Sanyo

Table 13-112 IRCTRL\_MF\_IR\_DEC\_DURATN2

Bit(s)	R/W	Default	Description
31:26	R	0	Unused
25:16	R/W	0	Max duration of Duration Setting Register 2. It defines max duration for: Half bit for RC5/6 format (RC5 typically 888.89us for half bit, RC6 typically 444.44us) or time of Duokan/RCMM/4ppm format's Logic "10" or time of Comcast/16ppm's base duration
15:10	R	0	Unused
9:0	R/W	0	Min duration of Duration Setting Register 2.

Table 13-113 IRCTRL\_MF\_IR\_DEC\_DURATN3

Bit(s)	R/W	Default	Description
31:26	R	0	Unused
25:16	R/W	0	Max duration of Duration Setting Register 3. It defines max duration for: Whole bit for RC5/6 format (RC5 typically 1777.78us for whole bit, RC6 typically 888.89us) or time of Duokan/RCMM/4ppm format's Logic "11" or time of Comcast/16ppm's offset duration
15:10	R	0	Unused
9:0	R/W	0	Min duration of Duration Setting Register 3.

**Table 13-114 IRCTRL\_MF\_IR\_DEC\_FRAME1: Frame Body 1 ( Frame Data, MSB 32Bit )**

Bit(s)	R/W	Default	Description
31:0	R	0	Stores frame body excess 32 bit range. (MSB 32 bit)

**Note**

New keys will be ignored until **FrameBody** register is read if the hold first key Bit is set in the de-code control register. Reading this register resets an internal frame data valid flag.

**Table 13-115 IRCTRL\_MF\_IR\_DEC\_STATUS\_1**

Bit(s)	R/W	Default	Description
31:20	R	0	Unused
19:0	R	0	Stores the last frame-to-frame counter value before the last counter reset caused by the last frame data record/update.

**Table 13-116 IRCTRL\_MF\_IR\_DEC\_STATUS\_2**

Bit(s)	R/W	Default	Description
31:20	R	0	Unused
19:0	R	0	Stores the value of the frame-to-frame counter which is running currently.

**Table 13-117 IRCTRL\_MF\_IR\_DEC\_REG3**

Bit(s)	R/W	Default	Description
19:0	R/W	0	Repeat time max. Used to set the maximum time between two repeat frames for repeat frame detection. Time unit is 100uS or 10uS according to the running_count_tick_mode. When repeat frame detection is enabled by enabling compare frame or compare bit method, the frame time interval may need to be checked in order to decide whether the frames are repeat (key pressed without release) or not. If frame interval is smaller than the “repeat time max”, it may considered as repeat. If frame interval is bigger than the “repeat time max”, it is considered as not repeat.

**Table 13-118 IRCTRL\_MF\_IR\_DEC\_FRAME\_RSV0**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Frame Reserved register 0. Stores the LSB 32bit of the previous frame’s frame data while the “Frame_Body” register stores the LSB 32bit of the current frame’s frame data.

**Table 13-119 IRCTRL\_MF\_IR\_DEC\_FRAME\_RSV1**

Bit(s)	R/W	Default	Description
31:0	R/W	0	Frame Reserved register 0. Stores the LSB 32bit of the previous frame’s frame data while the “Frame_Body” register stores the LSB 32bit of the current frame’s frame data.

Table 13-120 IRCTRL\_MF\_IR\_DEC\_FILTE

Bit(s)	R/W	Default	Description
31	R/W	0	New filter ctrl enable: 1:new filter ctrl enable 0:old filter ctrl enable
12:0	R/W	0	New Filter ctrl Set the monitor timing for input filter, bigger value means longer monitor time. Value 0 = no filtering.

Table 13-121 IRCTRL\_MF\_IR\_DEC\_IRQ\_CTL

Bit(s)	R/W	Default	Description
31	R/W	0	Interrupt enable 1:remote ir will generate irq 0:remote ir will not generate irq
30	R/W	0	Repeat idle mux 1:not check input single in NEC repeat idle state in the time of repeat_dely_time 0:check single
29	R/W	0	Release status Release pending bit ,write 1 to 0
28:16	R/W	0	Repeat_dely_time The time will not check input single after repeat leader
15:14	R/W	0	reversed
13	R/W	0	Release irq enable 1:release will generate irq 0:release will not generate irq
12:0	R/W	0	Release_det_time The time to judge release after irq

Table 13-122 IRCTRL\_MF\_IR\_DEC\_FIFO\_CTL

Bit(s)	R/W	Default	Description
31	R/W	0	Fifo mode enable,only use in measure mode 1:fifo mode enable 2:use old measure mode
30	R/W	0	Fifo trigger pending,write 1 to 0 Fifo have received X dates.X configured by IR_DEC_FIFO_CTL[20:13]
29	R/W	0	Fifo in timeout pending,write 1 to 0 Fifo have not received data in X.X configured by IR_DEC_FIFO_CTL[12:0]
28	R/W	0	Fifo full
27	R/W	0	Fifo empty
26	R/W	0	Fifo initial
25:21	R/W	0	reversed

Bit(s)	R/W	Default	Description
20:13	R/W	0	Fifo trigger irq level Default is 'h16
12:0	RIRCTR- L_MF_ W	0	Fifo in timeout Default is 'd800

**Table 13-123 IRCTRL\_MF\_IR\_DEC\_WIDTH\_NEW**

Bit(s)	R/W	Default	Description
12:0	R/W	0	Width new When use fifo mode,should read width from this register

**Table 13-124 IRCTRL\_MF\_IR\_DEC\_REPEAT\_DET**

Bit(s)	R/W	Default	Description
31	R/W	0	Repeat detect mux enable
28:16	R/W	0	Max repeat idle time
12:0	R/W	0	Min repeat idle time

## 13.7 Pulse-Width Modulation

### 13.7.1 Overview

The chip has 6 PWM modules that can be connected to various digital I/O pins. Each PWM is driven by a programmable divider driven by a 4:1 clock selector. The PWM signal is generated using two 16-bit counters. One is the High and Low counter, which is individually programmable with values between 1 and 65535. Using a combination of the divided clock (divide by N) and the HIGH and LOW counters, a wide number of PWM configurations are possible. The other is delta-sigma counter, generate 18-bit sigma, the PWM-out is the highest sigma. The PWM outputs vs counters are also illustrate below.

**Figure 13-8 PWM Block Diagram**

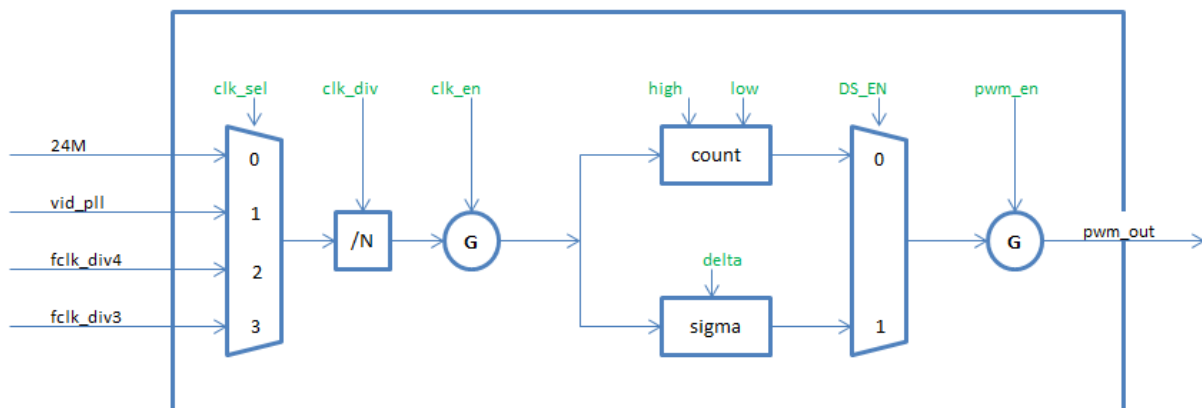


Figure 13-9 High/Low Counter

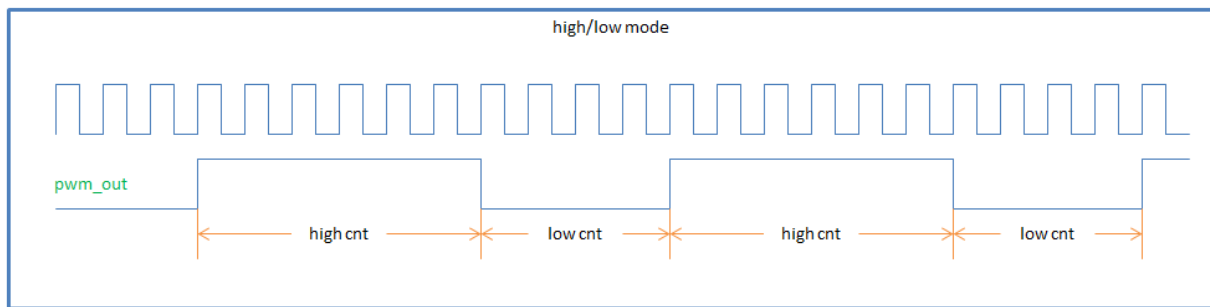
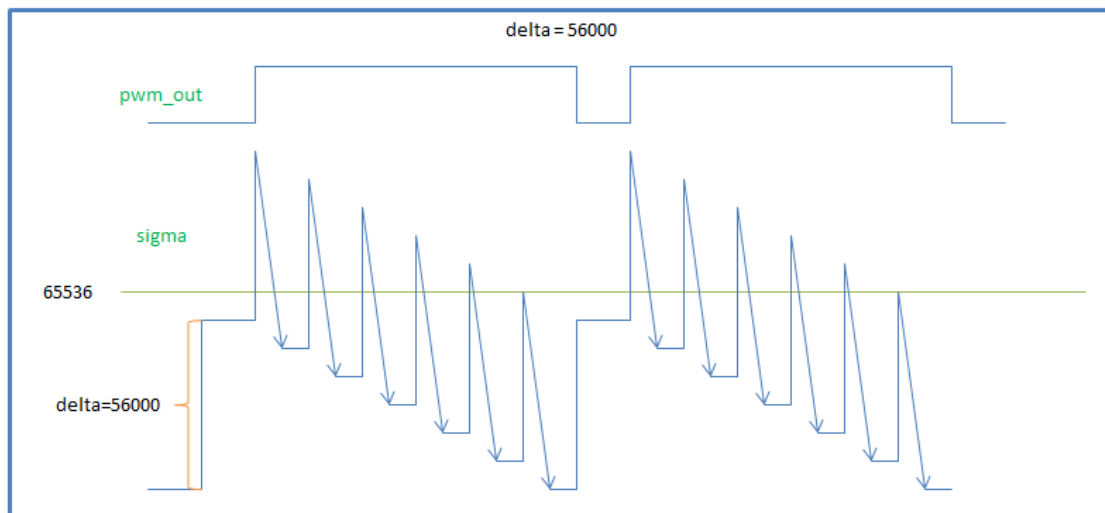
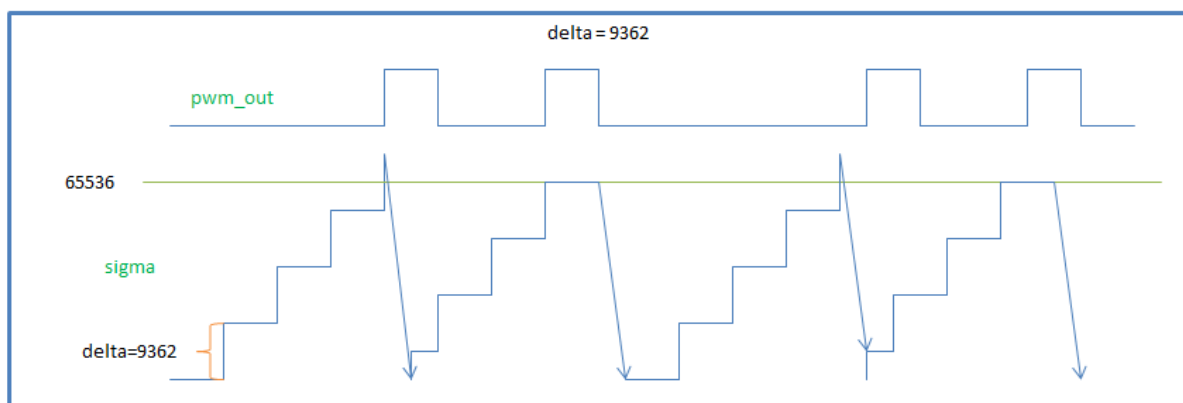
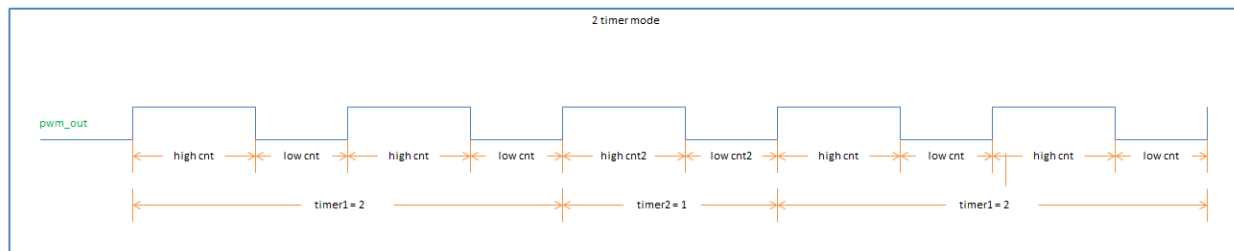


Figure 13-10 Delta-sigma Center



PWM 2 timer mode is illustrated as following:

Figure 13-11 2 Timer Mode



### 13.7.2 Register Description

Each PWM module can generate 2 PWM. PWM\_CD/EF/AO\_AB/AO\_CD/AO\_EF/AO\_GH usage are the same as PWM\_AB.

For the following registers:

Base\_adr: 0xfe058000

Final\_adr = base\_adr + offset \*4

The following lists describe the mapping between each PWM\_AB register and its address.

- PWMAB\_PWM\_A 0xfe058000
- PWMAB\_PWM\_B 0xfe058004
- PWMAB\_MISC\_REG\_AB 0xfe058008
- PWMAB\_DELTA\_SIGMA\_AB 0xfe05800c
- PWMAB\_TIME\_AB 0xfe058010
- PWMAB\_A2 0xfe058014
- PWMAB\_B2 0xfe058018
- PWMAB\_BLINK\_AB 0xfe05801c
- PWMAB\_LOCK\_AB 0xfe058020
- PWMCD\_PWM\_A 0xfe05a000
- PWMCD\_PWM\_B 0xfe05a004
- PWMCD\_MISC\_REG\_AB 0xfe05a008
- PWMCD\_DELTA\_SIGMA\_AB 0xfe05a00c
- PWMCD\_TIME\_AB 0xfe05a010
- PWMCD\_A2 0xfe05a014
- PWMCD\_B2 0xfe05a018
- PWMCD\_BLINK\_AB 0xfe05a01c
- PWMCD\_LOCK\_AB 0xfe05a020
- PWMEF\_PWM\_A 0xfe05c000
- PWMEF\_PWM\_B 0xfe05c004
- PWMEF\_MISC\_REG\_AB 0xfe05c008
- PWMEF\_DELTA\_SIGMA\_AB 0xfe05c00c
- PWMEF\_TIME\_AB 0xfe05c010
- PWMEF\_A2 0xfe05c014
- PWMEF\_B2 0xfe05c018
- PWMEF\_BLINK\_AB 0xfe05c01c



- PWMEF\_LOCK\_AB 0xfe05c020
- PWM\_AO\_AB\_PWM\_A 0xfe05e000
- PWM\_AO\_AB\_PWM\_B 0xfe05e004
- PWM\_AO\_AB\_MISC\_REG\_AB 0xfe05e008
- PWM\_AO\_AB\_DELTA\_SIGMA\_AB 0xfe05e00c
- PWM\_AO\_AB\_TIME\_AB 0xfe05e010
- PWM\_AO\_AB\_A2 0xfe05e014
- PWM\_AO\_AB\_B2 0xfe05e018
- PWM\_AO\_AB\_BLINK\_AB 0xfe05e01c
- PWM\_AO\_AB\_LOCK\_AB 0xfe05e020
- PWM\_AO\_CD\_PWM\_A 0xfe060000
- PWM\_AO\_CD\_PWM\_B 0xfe060004
- PWM\_AO\_CD\_MISC\_REG\_AB 0xfe060008
- PWM\_AO\_CD\_DELTA\_SIGMA\_AB 0xfe06000c
- PWM\_AO\_CD\_TIME\_AB 0xfe060010
- PWM\_AO\_CD\_A2 0xfe060014
- PWM\_AO\_CD\_B2 0xfe060018
- PWM\_AO\_CD\_BLINK\_AB 0xfe06001c
- PWM\_AO\_CD\_LOCK\_AB 0xfe060020
- PWM\_AO\_EF\_PWM\_A 0xfe030000
- PWM\_AO\_EF\_PWM\_B 0xfe030004
- PWM\_AO\_EF\_MISC\_REG\_AB 0xfe030008
- PWM\_AO\_EF\_DELTA\_SIGMA\_AB 0xfe03000c
- PWM\_AO\_EF\_TIME\_AB 0xfe030010
- PWM\_AO\_EF\_A2 0xfe030014
- PWM\_AO\_EF\_B2 0xfe030018
- PWM\_AO\_EF\_BLINK\_AB 0xfe03001c
- PWM\_AO\_EF\_LOCK\_AB 0xfe030020
- PWM\_AO\_GH\_PWM\_A 0xfe032000
- PWM\_AO\_GH\_PWM\_B 0xfe032004
- PWM\_AO\_GH\_MISC\_REG\_AB 0xfe032008
- PWM\_AO\_GH\_DELTA\_SIGMA\_AB 0xfe03200c
- PWM\_AO\_GH\_TIME\_AB 0xfe032010
- PWM\_AO\_GH\_A2 0xfe032014
- PWM\_AO\_GH\_B2 0xfe032018
- PWM\_AO\_GH\_BLINK\_AB 0xfe03201c
- PWM\_AO\_GH\_LOCK\_AB 0xfe032020

Table 13-125 PWMAB\_PWM\_A

Bit(s)	R/W	Default	Description
31-15	R/W	134	PWM_A_HIGH: This sets the high time (in clock counts) for the PWM_A generator output
15-0	R/W	134	PWM_A_LOW: This sets the high time (in clock counts) for the PWM_A generator output

**Table 13-126 PWMAB\_PWM\_B**

Bit(s)	R/W	Default	Description
31-15	R/W	80	PWM_B_HIGH: This sets the high time (in clock counts) for the PWM_B generator output
15-0	R/W	8	PWM_B_LOW: This sets the high time (in clock counts) for the PWM_B generator output

**Table 13-127 PWMAB\_MISC\_REG\_AB**

Bit(s)	R/W	Default	Description
31	R/W	0	pwm_B_hiz when hiz mode, pwm_o will connect to pad_oe, then pad_o will equal this bit
30	R/W	0	pwm_A_hiz when hiz mode, pwm_o will connect to pad_oe, then pad_o will equal this bit
29	R/W	0	pwm_B_constant_en
28	R/W	0	pwm_A_constant_en
27	R/W	0	pwm_B_inv_en
26	R/W	0	pwm_A_inv_en
25	R/W	0	cntl_pwm_a2_en
24	R/W	0	cntl_pwm_b2_en
23	R/W	0	PWM_B_CLK_EN: Set this bit to 1 to enable PWM B clock
22-16	R/W	0	PWM_B_CLK_DIV: Selects the divider (N+1) for the PWM B clock. See the clock tress document
15	R/W	0	PWM_A_CLK_EN: Set this bit to 1 to enable PWM A clock
14-8	R/W	0	PWM_A_CLK_DIV: Selects the divider (N+1) for the PWM A clock. See the clock tress document
7-6	R/W	0	PWM_B_CLK_SEL: Select the clock for the PWM B. See the clock tress document
5-4	R/W	0	PWM_A_CLK_SEL: Select the clock for the PWM A. See the clock tress document
3	R/W	0	DS_B_EN: This bit is only valid if PWM_B_EN is 0: if this bit is set to 1, then the PWM_B output is configured to generate a delta sigma output based on the settings in the register below. If this bit is set to 0, then the PWM_B output is set low.
2	R/W	0	DS_A_EN: This bit is only valid if PWM_A_EN is 0: if this bit is set to 1, then the PWM_A output is configured to generate a delta sigma output based on the settings in the register below. If this bit is set to 0, then the PWM_A output is set low.
1	R/W	0	PWM_B_EN: If this bit is set to 1, then the PWM_B output is configured to generate a PWM output based on the register above. If this bit is 0, then the PWM_B output is controlled by DS_B_EN above.
0	R/W	0	PWM_A_EN: If this bit is set to 1, then the PWM_A output is configured to generate a PWM output based on the register above. If this bit is 0, then the PWM_A output is controlled by DS_A_EN above.

**Table 13-128 PWMAB\_DELTA\_SIGMA\_AB**

Bit(s)	R/W	Default	Description
31-15	R/W	0x8000	DS_B_VAL: This value represents the delta sigma setting for channel B (PWM_B)
15-0	R/W	0x8000	DS_A_VAL: This value represents the delta sigma setting for channel A (PWM_A)

**Table 13-129 PWM\_TIME\_AB**

Bit(s)	R/W	Default	Description
31-24	R/W	10	A_timer1
23:16	R/W	10	A_timer2
15:8	R/W	10	B_timer1
7:0	R/W	10	B_timer2

**Table 13-130 PWMAB\_A2**

Bit(s)	R/W	Default	Description
31-15	R/W	134	PWM_A2_HIGH: This sets the high time (in clock counts) for the PWM_A2 generator output
15-0	R/W	134	PWM_A2_LOW: This sets the high time (in clock counts) for the PWM_A2 generator output

**Table 13-131 PWMAB\_B2**

Bit(s)	R/W	Default	Description
31-15	R/W	80	PWM_B2_HIGH: This sets the high time (in clock counts) for the PWM_B2 generator output
15-0	R/W	8	PWM_B2_LOW: This sets the high time (in clock counts) for the PWM_B2 generator output

**Table 13-132 PWMAB\_LOCK\_AB**

Bit(s)	R/W	Default	Description
0	R/W	0	1: will lock all PWM register

## 13.8 SAR ADC

### 13.8.1 Overview

This SAR ADC is a general purpose ADC for measuring analog signals. The module can make RAW ADC measurements or average a number of measurements to introduce filtering. The SAR ADC is a single block so an analog mux is placed in front to allow multiple different measurements to be made sequentially. Timing of the samples, and delays between muxing are all programmable as is the averaging to be applied to the SAR ADC.

### 13.8.2 Register Description

For the following registers:

Base\_adr: 0xfe026000

Final\_adr = base\_adr + offset \*4

The following lists describe the mapping between each SAR\_ADC register and its address.

SAR_ADC_REG0	0xfe026000
SAR_ADC_CHAN_LIST	0xfe026004
SAR_ADC_AVG_CNTL	0xfe026008
SAR_ADC_REG3	0xfe02600c
SAR_ADC_DELAY	0xfe026010
SAR_ADC_LAST_RD	0xfe026014
SAR_ADC_FIFO_RD	0xfe026018
SAR_ADC_AUX_SW	0xfe02601c
SAR_ADC_CHAN_10_SW	0xfe026020
SAR_ADC_DETECT_IDLE_SW	0xfe026024
SAR_ADC_DELTA_10	0xfe026028
SAR_ADC_REG11	0xfe02602c
SAR_ADC_REG12	0xfe026030
SAR_ADC_REG13	0xfe026034
SAR_ADC_CHNL01	0xfe026038
SAR_ADC_CHNL23	0xfe02603c
SAR_ADC_CHNL45	0xfe026040
SAR_ADC_CHNL67	0xfe026044

**Table 13-133 SAR\_ADC\_REG0: Control Register #0**

Bit(s)	R/W	Default	Description
31	R	0	PANEL DETECT level.
30	R	0	DELTA_BUSY: If This bit is 1, then it indicates the delta processing engine is busy
29	R	0	AVG_BUSY: If This bit is 1, then it indicates the averaging engine is busy
28	R	0	SAMPLE_BUSY: If This bit is 1, then it indicates the sampling engine is busy
27	R	0	FIFO_FULL:
26	R	0	FIFO_EMPTY:
25-21	R/W	4	FIFO_COUNT: Current count of samples in the acquisition FIFO
20-19	R/W	0	ADC_BIAS_CTRL
18-16	R/W	0	CURR_CHAN_ID: These Bits represent the current channel (0..7) that is being sampled.
15	R	0	Unused

Bit(s)	R/W	Default	Description
14	R/W	0	SAMPLING_STOP: This bit can be used to cleanly stop the sampling process in the event that continuous sampling is enabled. To stop sampling, simply set This bit and wait for all processing modules to no longer indicate that they are busy.
13-12	R/W	0	CHAN_DELTA_EN: There are two Bits corresponding to Channels 0 and 1. Channel 0 and channel 1 can be individually enabled to take advantage of the delta processing module.
11	R/W	0	Unused
10	R/W	0	DETECT_IRQ_POL: This bit sets the polarity of the detect signal. The detect signal is used during X/Y panel applications to detect if the panel is touched
9	R/W	0	DETECT_IRQ_EN: If This bit is set to 1, then an interrupt will be generated if the DETECT signal is low/high. The polarity is set in the bit above.
8-4	R/W	0	FIFO_CNT_IRQ: When the FIFO contains N samples, then generate an interrupt (if bit 3 is set below).
3	R/W	0	FIFO_IRQ_EN: Set This bit to 1 to enable an IRQ when the acquisition FIFO reaches a certain level.
2	W	0	SAMPLE_START: This bit should be written to 1 to start sampling.
1	R/W	0	CONTINUOUS_EN: If This bit is set to 1, then the channel list will be continually processed
0	R/W	0	SAMPLING_ENABLE: Setting This bit to '1' enables the touch panel controller sampling engine, averaging module, XY processing engine and the FIFO.

Table 13-134 SAR\_ADC\_CHAN\_LIST:Channel List

Bit(s)	R/W	Default	Description
31-27	R/W	0	unused
26-24	R/W	2	Length of the list of channels to process. If this value is 2, then only channels in Bits [8:0] below are processed.
23-21	R/W	7	8th channel
20-18	R/W	6	7th channel
17-15	R/W	5	6th channel
14-12	R/W	4	5th channel
11-9	R/W	3	4th channel
8-6	R/W	2	3rd channel
5-3	R/W	1	2nd channel
2-0	R/W	0	First channel in the list of channels to process

Table 13-135 SAR\_ADC\_AVG\_CNTL:Sampling/Averaging Modes

Bit(s)	R/W	Default	Description
31-30	R/W	0	Channel 7: Averaging mode: 0 = no averaging, 1 = simple averaging, 2 = median averaging.
29-28	R/W	0	Channel 6: Averaging mode: 0 = no averaging, 1 = simple averaging, 2 = median averaging.
27-26	R/W	0	Channel 5: Averaging mode: 0 = no averaging, 1 = simple averaging, 2 = median averaging.

Bit(s)	R/W	Default	Description
25-24	R/W	0	Channel 4: Averaging mode: 0 = no averaging, 1 = simple averaging, 2 = median averaging.
23-22	R/W	0	Channel 3: Averaging mode: 0 = no averaging, 1 = simple averaging, 2 = median averaging.
21-20	R/W	0	Channel 2: Averaging mode: 0 = no averaging, 1 = simple averaging, 2 = median averaging.
19-18	R/W	0	Channel 1: Averaging mode: 0 = no averaging, 1 = simple averaging, 2 = median averaging.
17-16	R/W	0	Channel 0: Averaging mode: 0 = no averaging. 1 = simple averaging of the number of samples acquired (1,2,4 or 8). 2 = median averaging. NOTE: If these Bits are set to 2, then you must set the number of samples to acquire below to 8.
15-13	R/W	0	Channel 7: Number of samples to acquire 2N:
13-12	R/W	0	Channel 6: Number of samples to acquire 2N:
11-10	R/W	0	Channel 5: Number of samples to acquire 2N:
9-8	R/W	0	Channel 4: Number of samples to acquire 2N:
7-6	R/W	0	Channel 3: Number of samples to acquire 2N:
5-4	R/W	0	Channel 2: Number of samples to acquire 2N:
3-2	R/W	0	Channel 1: Number of samples to acquire 2N:
1-0	R/W	0	Channel 0: Number of samples to acquire 2N: 0 = 1, 1 = 2, 2 = 4, 4 = 8.

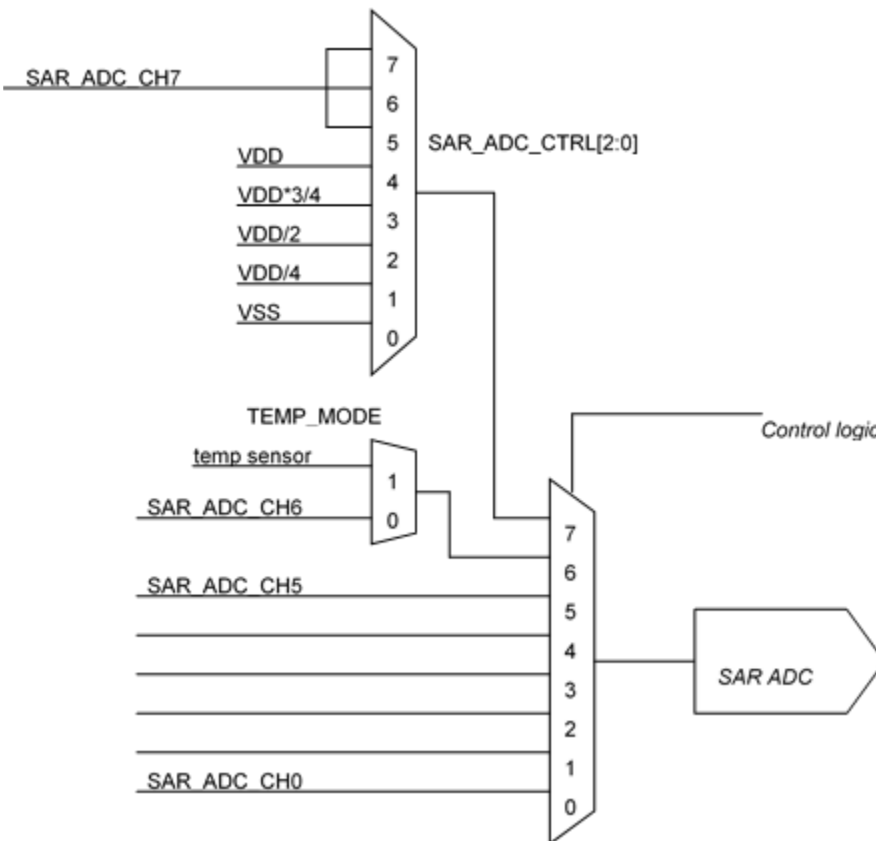
Each channel listed in the CHANNEL\_LIST is given independent control of the number of samples to acquire and averaging mode.

**Table 13-136 SAR\_ADC\_REG3: Control Register #3**

Bit(s)	R/W	Default	Description
31	R/W	0	CNTL_USE_SC_DLY: hold time delay was added to the start conversion clock. Unfortunately, it appears that the analog ADC design requires that we use the inverted clock so This bit is meaningless.
30	R/W	0	SAR_ADC_CLK_EN: 1 = enable the SAR ADC clock
29	R/W	0	reserved
28	R/W	0	reserved
27	R/W	0	SARADC_CTRL[4]: is used to control the internal ring counter. 1 = enable the continuous ring counter. 0 = disable
26	R/W	0	SARADC_CTRL[3]: used to select the internal sampling clock phase
25~23	R/W	0	SARADC_CTRL[2:0]: 000 ssa 001 vdda/4 010 vdda/2 011 vdda*3/4 100 vdda 101, 110, 111 unused
22	R/W	0	DETECT_EN: This bit controls the analog switch that connects a 50k resistor to the X+ signal. Setting This bit to 1 closes the analog switch
21	R/W	0	ADC_EN: Set This bit to 1 to enable the ADC

Bit(s)	R/W	Default	Description
20-18	R/W	2	PANEL_DETECT_COUNT: Increasing this value increases the filtering on the panel detect signal using the timebase settings in Bits [17:16] below.
17-16	R/W	0	PANEL_DETECT_FILTER_TB: 0 = count 1uS ticks, 1 = count 10uS ticks, 2 = count 100uS ticks. 3 = count 1mS ticks
15-10	R/W	20	ADC_CLK_DIV: The ADC clock is derived by dividing the 27Mhz crystal by N+1. This value divides the 27Mhz clock to generate an ADC clock. A value of 20 for example divides the 27Mhz clock by 21 to generate an equivalent 1.28Mhz clock.
9-8	R/W	1	BLOCK_DLY_SEL: 0 = count 1uS ticks, 1 = count 10uS ticks, 2 = count 100uS ticks. 3 = count 1mS ticks
7-0	R/W	10	BLOCK_DLY: After all channels in the CHANNEL_LIST have been processed, the sampling engine will delay for an amount of time before re-processing the CHANNEL_LIST again. Combined with Bits [9:8] above, this value is used to generate a delay between processing blocks of channels.

Figure 13-12 SAR\_ADC\_REG3



As the CHANNEL\_LIST is process, the input switches are set according to the requirements of the channel. After setting the switches there is a programmable delay before sampling begins. Additionally, each channel specifies the number of samples for that particular channel. The sampling rate is programmed below.

**Table 13-137 SAR\_ADC\_DELAY:INPUT / SAMPLING DELAY**

Bit(s)	R/W	Default	Description
31-29	R	0	unused
28	R/W	0	CNTL_EOC_BY_CNT: ADC dout valid controlled by counter
27	R/W	0	CNTL_USE_LATCHED_DATA: ADC dout be latched first, then be sampled
26	R	0	unused
25-24	R/W	0	INPUT_DLY_SEL: 0 = 111nS ticks, 1 = count 1uS ticks, 2 = count 10uS ticks, 3 = count 100uS ticks
23-16	R/W	3	INPUT_DLY_CNY: For channels that acquire 2,4 or 8 samples, the delay between two samples is controlled by this count (N+1) combined with the delay selection in the two bits above.
15-10	R/W	14	CNTL_EOC_DLY_CNT: the delay between SC and ADC output data ready for latch/sample
9-8	R/W	0	SAMPLE_DLY_SEL: 0 = count 1uS ticks, 1 = count 10uS ticks, 2 = count 100uS ticks. 3 = count 1mS ticks
7-0	R/W	9	SAMPLE_DLY_CNY: For channels that acquire 2,4 or 8 samples, the delay between two samples is controlled by this count (N+1) combined with the delay selection in the two bits above.

For channel 0 and channel 1, (the special X/Y channels) the last sample pushed into the FIFO for each channel is saved in a register. This allows the software to see the last sample for channel 0 and channel 1 even when the FIFO overflows. For example, if we are sampling quickly and there is a gesture on the screen, we can use the contents of the FIFO to see the direction of the gesture and use the last sample values to see where the pen finally came to rest.

**Table 13-138 SAR\_ADC\_LAST\_RD: Last Sample**

Bit(s)	R/W	Default	Description
31-28	R	0	unused
27-16	R	0	LAST_CHANNEL1
15-12	R	0	unused
11-0	R	0	LAST_CHANNEL0

**Table 13-139 SAR\_ADC\_FIFO\_RD: Control Register #6 (FIFO RD)**

Bit(s)	R/W	Default	Description
31-28	R	0	Unused
27-16	R	0	RAW data
15	R	0	Unused
14-12	R	0	Channel ID. This value identifies the channel associated with the data in Bits [11:0] below
11-0	R	0	Sample value: 12-bit raw or averaged ADC sample written to the FIFO.

Channels 2 ~ 7 can program the ADC input mux to any selection between 0 and 7. This register allows the software to associate a mux selection with a particular channel. In addition to the ADC mux, there are a number of switches that can be set in any particular state. Channels 2 ~ 7 share a common



switch setting. Channels 0 and 1 on the other hand have programmable switch settings (see other registers below).

**Table 13-140 SAR\_ADC\_AUX\_SW:Channel 2~7 ADC MUX, Switch Controls**

Bit(s)	R/W	Default	Description
31	R	0	EOC
29	R/W	0	CNTL_MANUAL_SC
28	R/W	0	CNTL_MANUAL_CLK
27	R/W	0	CNTL_MANUAL_MODE
26	R	0	unused
25-23	R/W	7	Channel 7 ADC_MUX setting when channel 7 is being measured.
22-20	R/W	7	Channel 6 ADC_MUX setting when channel 6 is being measured.
19-17	R/W	7	Channel 5 ADC_MUX setting when channel 5 is being measured.
16-14	R/W	6	Channel 4 ADC_MUX setting when channel 4 is being measured.
13-11	R/W	0	Channel 3 ADC_MUX setting when channel 3 is being measured.
10-8	R/W	1	Channel 2 ADC_MUX setting when channel 2 is being measured.
7	R	0	unused
6	R/W	0	VREF_P_MUX setting when channel 2,3..7 is being measured
5	R/W	0	VREF_N_MUX setting when channel 2,3..7 is being measured
4	R/W	0	MODE_SEL setting when channel 2,3..7 is being measured
3	R/W	1	YP_DRIVE_SW setting when channel 2,3..7 is being measured
2	R/W	1	XP_DRIVE_SW setting when channel 2,3..7 is being measured
1	R/W	0	YM_DRIVE_SW setting when channel 2,3..7 is being measured
0	R/W	0	YM_DRIVE_SW setting when channel 2,3..7 is being measured

Channels 0 and 1 have independent programmable switch settings when either/both of these channels are being measured.

**Table 13-141 SAR\_ADC\_CHAN\_10\_SW:Channel 0, 1 ADC MUX, Switch Controls**

Bit(s)	R/W	Default	Description
31-26	R	0	unused
25-23	R/W	2	Channel 1 ADC MUX setting
22	R/W	0	Channel 1 VREF_P_MUX
21	R/W	0	Channel 1 VREF_N_MUX
20	R/W	0	Channel 1 MODE_SEL
19	R/W	1	Channel 1 YP_DRIVE_SW setting: 0: TADC_CH6N = 3.3v 1: TADC_CH6N = floating
18	R/W	1	Channel 1 XP_DRIVE_SW setting: 0: TADC_CH6N = 3.3v 1: TADC_CH6N = floating

Bit(s)	R/W	Default	Description
17	R/W	0	Channel 1 YM_DRIVE_SW setting: 0: TADC_CH4N = floating 1: TADC_CH4N = GND
16	R/W	0	Channel 1 YM_DRIVE_SW setting: 0: TADC_CH4N = floating 1: TADC_CH4N = GND
15-10	R		unused
9-7	R/W	3	Channel 0 ADC MUX setting
6	R/W	0	Channel 0 VREF_P_MUX
5	R/W	0	Channel 0 VREF_N_MUX
4	R/W	0	Channel 0 MODE_SEL
3	R/W	1	Channel 0 YP_DRIVE_SW setting: 0: TADC_CH6N = 3.3v 1: TADC_CH6N = floating
2	R/W	1	Channel 0 XP_DRIVE_SW setting: 0: TADC_CH6N = 3.3v 1: TADC_CH6N = floating
1	R/W	0	Channel 0 YM_DRIVE_SW setting: 0: TADC_CH4N = floating 1: TADC_CH4N = GND
0	R/W	0	Channel 0 YM_DRIVE_SW setting: 0: TADC_CH4N = floating 1: TADC_CH4N = GND

**IDLE MODE:** When nothing is being measured, the switches should be put into a safe state. This safe state is accomplished using Bits [9:0] below.

**DETECT MODE:** When bit [26] is set, the input muxes / switches are configured according to the Bits below. Typically the software configures the switches below to correspond to the detect touch mode. That is, Y- internal MOSFET is closed so that the Y plane of the touch screen is connected to Ground. Additionally, the DETECT\_EN bit(different register) set to 1 so that the 50k resistor to VDD is connected to X+. In this configuration, the detect comparator connected to the 50k resistor will be weakly pulled up to VDD through the 50k resistor. If the user touches the screen, the X and Y planes of the touch screen will contact causing the X+ signal to be pulled to ground.

**Table 13-142 SAR\_ADC\_DETECT\_IDLE\_SW:DETECT / IDLE Mode switches**

Bit(s)	R/W	Default	Description
31-27	R	0	unused
26	R/W	0	DETECT_SW_EN: If This bit is set, then Bits [25:16] below are applied to the analog muxes/switches of the touch panel controller.
25-23	R/W	5	DETECT MODE ADC MUX setting
22	R/W	0	DETECT MODE VREF_P_MUX setting
21	R/W	0	DETECT MODE VREF_N_MUX setting
20	R/W	0	DETECT MODE MODE_SEL setting
19	R/W	1	DETECT MODE YP_DRIVE_SW setting: 0: TADC_CH6N = 3.3v 1: TADC_CH6N = floating

Bit(s)	R/W	Default	Description
18	R/W	1	DETECT MODE XP_DRIVE_SW setting: 0: TADC_CH6N = 3.3v 1: TADC_CH6N = floating
17	R/W	0	DETECT MODE YM_DRIVE_SW setting: 0: TADC_CH4N = floating 1: TADC_CH4N = GND
16	R/W	0	DETECT MODE YM_DRIVE_SW setting: 0: TADC_CH4N = floating 1: TADC_CH4N = GND
15-10	R		Unused
9-7	R/W	5	IDLE MODE ADC_MUX setting
6	R/W	0	IDLE MODE VREF_P_MUX setting
5	R/W	0	IDLE MODE VREF_N_MUX setting
4	R/W	0	IDLE MODE MODE_SEL setting
3	R/W	1	IDLE MODE YP_DRIVE_SW setting: 0: TADC_CH6N = 3.3v 1: TADC_CH6N = floating
2	R/W	1	IDLE MODE XP_DRIVE_SW setting: 0: TADC_CH6N = 3.3v 1: TADC_CH6N = floating
1	R/W	0	IDLE MODE YN_DRIVE_SW setting: 0: TADC_CH4N = floating 1: TADC_CH4N = GND
0	R/W	0	IDLE MODE XN_DRIVE_SW setting: 0: TADC_CH4N = floating 1: TADC_CH4N = GND

Table 13-143 SAR\_ADC\_DELTA\_10:Delta Mode Deltas

Bit(s)	R/W	Default	Description
31-28	R	0	unused
27-16	R/W	0	Channel 1 delta value when delta processing for channel 1 is enabled.
15-12	R/W		unused
11-0	R/W	0	Channel 0 delta value when delta processing for channel 0 is enabled.

Table 13-144 SAR\_ADC\_REG11

Bit(s)	R/W	Default	Description
31	R/W	0	fifo_disable: 1=dout don't put into fifo
30	R/W	0	period_sample_en: 1=period_sample enable, dout register into chnl reg AO_SAR_ADC_CHNL01/23/45/67
29-13	R/W	0	ts_cntl_int
12-0	R/W	0	sar_bg_cntl

**Table 13-145 SAR\_ADC\_REG12**

Bit(s)	R/W	Default	Description
31-0	R/W	0	reserved

**Table 13-146 SAR\_ADC\_REG13**

Bit(s)	R/W	Default	Description
31-8	R	0	unused
15-8	R/W	0	SARADC_RSV2
7-0	R/W	0	reserved

**Table 13-147 SAR\_ADC\_CHNL01**

Bit(s)	R/W	Default	Description
31	R/W	0	dout_chnl1_valid
30-28	R/W	0	dout_chnl1_id
27-16	R/W	0	dout_chnl1
15	R/W	0	dout_chnl0_valid
14-12	R/W	0	dout_chnl0_id
11-0	R/W	0	dout_chnl0

**Table 13-148 SAR\_ADC\_CHNL23**

Bit(s)	R/W	Default	Description
31	R/W	0	dout_chnl3_valid
30-28	R/W	0	dout_chnl3_id
27-16	R/W	0	dout_chnl3
15	R/W	0	dout_chnl2_valid
14-12	R/W	0	dout_chnl2_id
11-0	R/W	0	dout_chnl2

**Table 13-149 SAR\_ADC\_CHNL45**

Bit(s)	R/W	Default	Description
31	R/W	0	dout_chnl5_valid
30-28	R/W	0	dout_chnl5_id
27-16	R/W	0	dout_chnl5
15	R/W	0	dout_chnl4_valid
14-12	R/W	0	dout_chnl4_id
11-0	R/W	0	dout_chnl4

Table 13-150 SAR\_ADC\_CHNL67

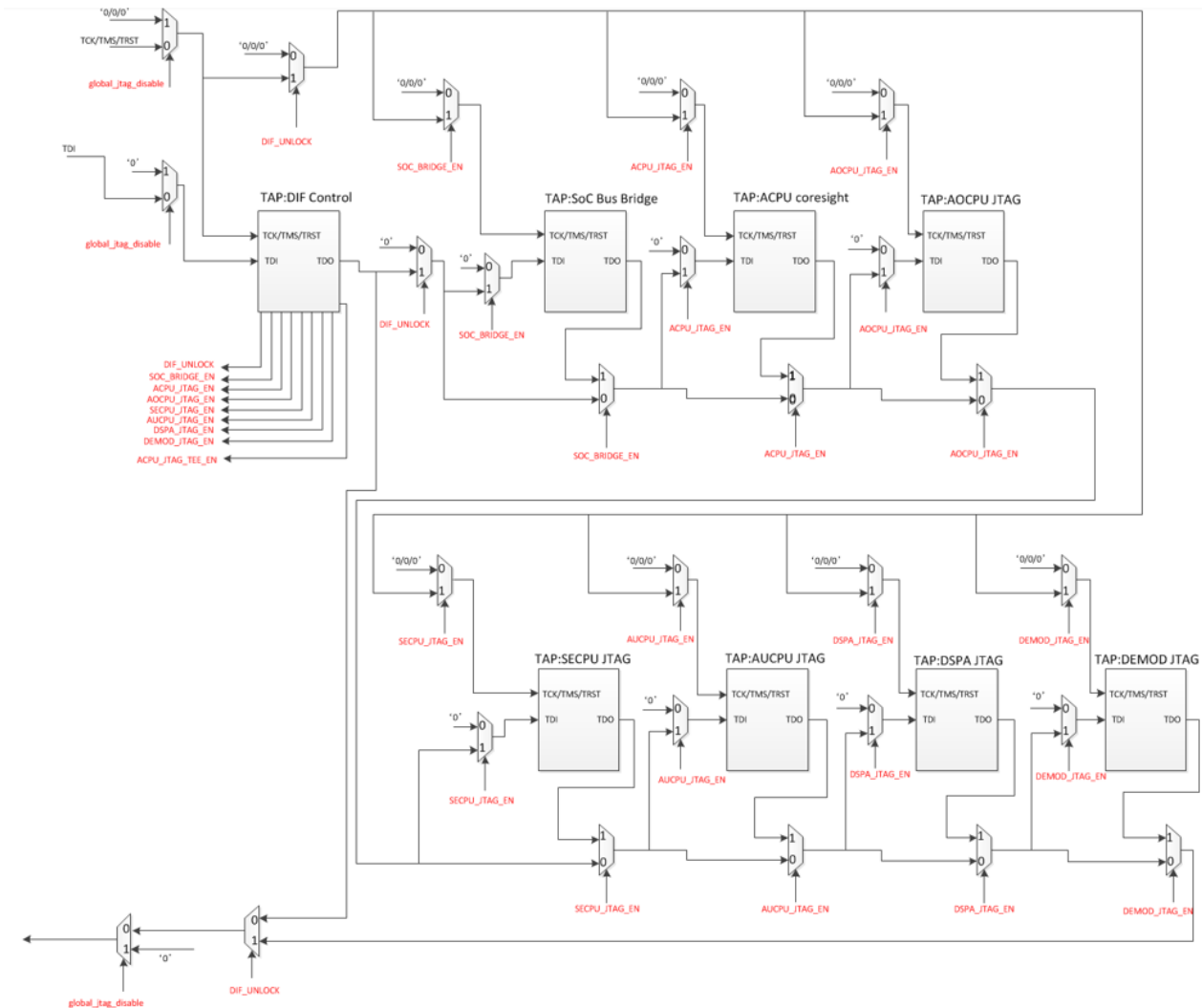
Bit(s)	R/W	Default	Description
31	R/W	0	dout_chnl7_valid
30-28	R/W	0	dout_chnl7_id
27-16	R/W	0	dout_chnl7
15	R/W	0	dout_chnl6_valid
14-12	R/W	0	dout_chnl6_id
11-0	R/W	0	dout_chnl6

# 14 System Interface

## 14.1 JTAG

JTAG is an interface for internal test. The structure of SoC JTAG module is shown in the following diagram.

Figure 14-1 JTAG Structure



## 14.2 Temp Sensor

### 14.2.1 Overview

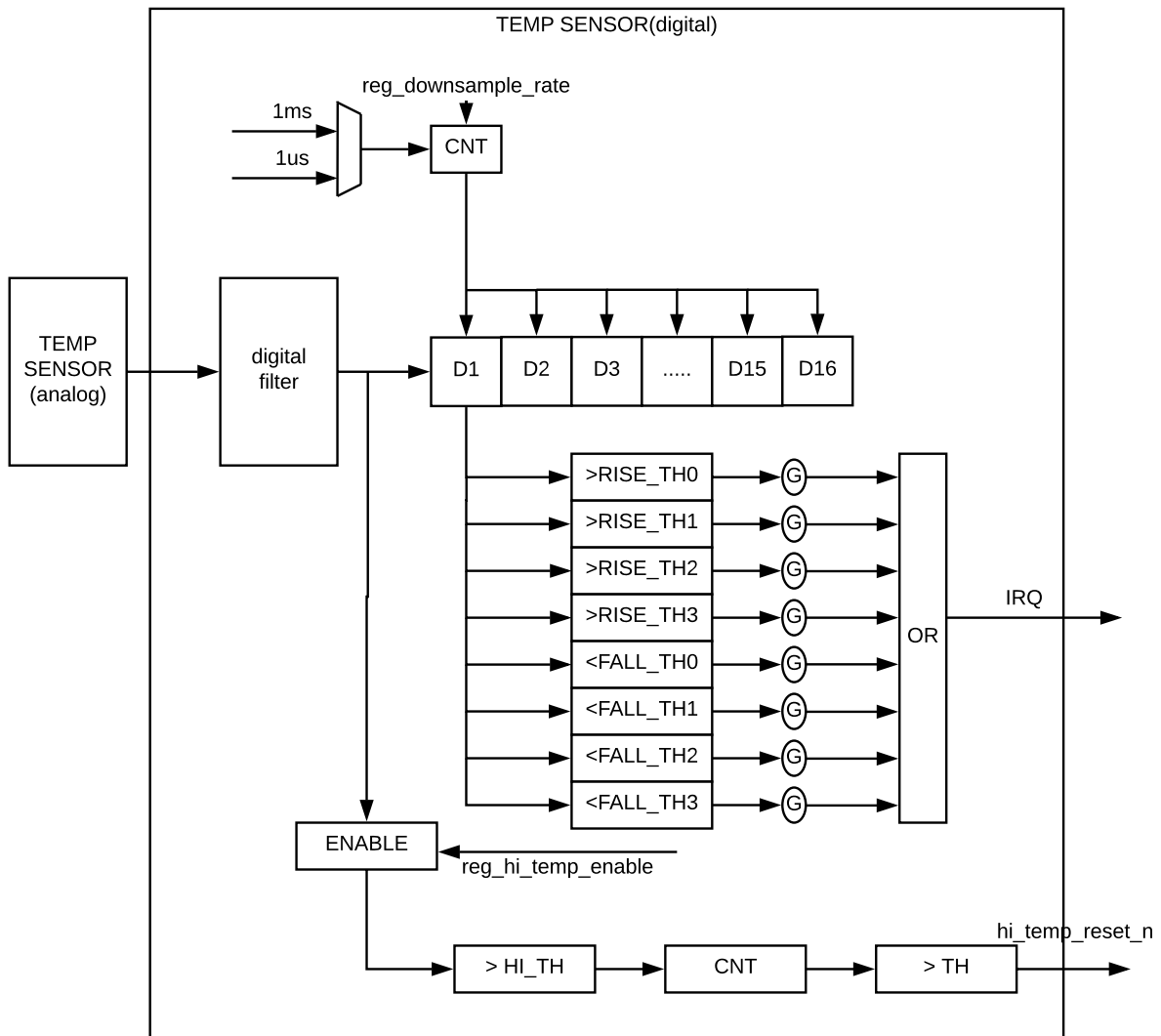
The SoC integrates 2 Temp Sensor, one is close to DDR, one is close to PLL (between CPU and GPU).

Each Temp Sensor are the same design.

- Capture temperature by programable tick;
- Store 16 temperature value;

- Can reset all chip if detect high temperature;
- Can generate IRQ by 8 threshold;

**Figure 14-2 Temp Sensor**



T02FC30

### 14.2.2 Register Description

Base Address: FE020000

Each register final address = module base address+ address \* 4

- TS\_A73\_CFG\_REG1 0xfe020004
- TS\_A73\_CFG\_REG2 0xfe020008
- TS\_A73\_CFG\_REG3 0xfe02000c
- TS\_A73\_CFG\_REG4 0xfe020010
- TS\_A73\_CFG\_REG5 0xfe020014
- TS\_A73\_CFG\_REG6 0xfe020018
- TS\_A73\_CFG\_REG7 0xfe02001c

- TS\_A73\_STAT0 0xfe020040
- TS\_A73\_STAT1 0xfe020044
- TS\_A73\_STAT2 0xfe020048
- TS\_A73\_STAT3 0xfe02004c
- TS\_A73\_STAT4 0xfe020050
- TS\_A73\_STAT5 0xfe020054
- TS\_A73\_STAT6 0xfe020058
- TS\_A73\_STAT7 0xfe02005c
- TS\_A73\_STAT8 0xfe020060
- TS\_A73\_STAT9 0xfe020064
- TS\_A53\_CFG\_REG1 0xfe022004
- TS\_A53\_CFG\_REG2 0xfe022008
- TS\_A53\_CFG\_REG3 0xfe02200c
- TS\_A53\_CFG\_REG4 0xfe022010
- TS\_A53\_CFG\_REG5 0xfe022014
- TS\_A53\_CFG\_REG6 0xfe022018
- TS\_A53\_CFG\_REG7 0xfe02201c
- TS\_A53\_STAT0 0xfe022040
- TS\_A53\_STAT1 0xfe022044
- TS\_A53\_STAT2 0xfe022048
- TS\_A53\_STAT3 0xfe02204c
- TS\_A53\_STAT4 0xfe022050
- TS\_A53\_STAT5 0xfe022054
- TS\_A53\_STAT6 0xfe022058
- TS\_A53\_STAT7 0xfe02205c
- TS\_A53\_STAT8 0xfe022060
- TS\_A53\_STAT9 0xfe022064
- TS\_GPU\_CFG\_REG1 0xfe094004
- TS\_GPU\_CFG\_REG2 0xfe094008
- TS\_GPU\_CFG\_REG3 0xfe09400c
- TS\_GPU\_CFG\_REG4 0xfe094010
- TS\_GPU\_CFG\_REG5 0xfe094014
- TS\_GPU\_CFG\_REG6 0xfe094018
- TS\_GPU\_CFG\_REG7 0xfe09401c
- TS\_GPU\_STAT0 0xfe094040
- TS\_GPU\_STAT1 0xfe094044
- TS\_GPU\_STAT2 0xfe094048
- TS\_GPU\_STAT3 0xfe09404c
- TS\_GPU\_STAT4 0xfe094050
- TS\_GPU\_STAT5 0xfe094054
- TS\_GPU\_STAT6 0xfe094058
- TS\_GPU\_STAT7 0xfe09405c
- TS\_GPU\_STAT8 0xfe094060
- TS\_GPU\_STAT9 0xfe094064



- TS\_NNA\_CFG\_REG1 0xfe096004
- TS\_NNA\_CFG\_REG2 0xfe096008
- TS\_NNA\_CFG\_REG3 0xfe09600c
- TS\_NNA\_CFG\_REG4 0xfe096010
- TS\_NNA\_CFG\_REG5 0xfe096014
- TS\_NNA\_CFG\_REG6 0xfe096018
- TS\_NNA\_CFG\_REG7 0xfe09601c
- TS\_NNA\_STAT0 0xfe096040
- TS\_NNA\_STAT1 0xfe096044
- TS\_NNA\_STAT2 0xfe096048
- TS\_NNA\_STAT3 0xfe09604c
- TS\_NNA\_STAT4 0xfe096050
- TS\_NNA\_STAT5 0xfe096054
- TS\_NNA\_STAT6 0xfe096058
- TS\_NNA\_STAT7 0xfe09605c
- TS\_NNA\_STAT8 0xfe096060
- TS\_NNA\_STAT9 0xfe096064
- TS\_VPU\_CFG\_REG1 0xfe098004
- TS\_VPU\_CFG\_REG2 0xfe098008
- TS\_VPU\_CFG\_REG3 0xfe09800c
- TS\_VPU\_CFG\_REG4 0xfe098010
- TS\_VPU\_CFG\_REG5 0xfe098014
- TS\_VPU\_CFG\_REG6 0xfe098018
- TS\_VPU\_CFG\_REG7 0xfe09801c
- TS\_VPU\_STAT0 0xfe098040
- TS\_VPU\_STAT1 0xfe098044
- TS\_VPU\_STAT2 0xfe098048
- TS\_VPU\_STAT3 0xfe09804c
- TS\_VPU\_STAT4 0xfe098050
- TS\_VPU\_STAT5 0xfe098054
- TS\_VPU\_STAT6 0xfe098058
- TS\_VPU\_STAT7 0xfe09805c
- TS\_VPU\_STAT8 0xfe098060
- TS\_VPU\_STAT9 0xfe098064
- TS\_HEVC\_CFG\_REG1 0xfe09a004
- TS\_HEVC\_CFG\_REG2 0xfe09a008
- TS\_HEVC\_CFG\_REG3 0xfe09a00c
- TS\_HEVC\_CFG\_REG4 0xfe09a010
- TS\_HEVC\_CFG\_REG5 0xfe09a014
- TS\_HEVC\_CFG\_REG6 0xfe09a018
- TS\_HEVC\_CFG\_REG7 0xfe09a01c
- TS\_HEVC\_STAT0 0xfe09a040
- TS\_HEVC\_STAT1 0xfe09a044
- TS\_HEVC\_STAT2 0xfe09a048

- TS\_HEVC\_STAT3 0xfe09a04c
- TS\_HEVC\_STAT4 0xfe09a050
- TS\_HEVC\_STAT5 0xfe09a054
- TS\_HEVC\_STAT6 0xfe09a058
- TS\_HEVC\_STAT7 0xfe09a05c
- TS\_HEVC\_STAT8 0xfe09a060
- TS\_HEVC\_STAT9 0xfe09a064

**Table 14-1 TS\_CFG\_REG1 0x001**

Bits	R/W	Default	Description
31	R/W	0	fall_th3_irq_en
30	R/W	0	fall_th2_irq_en
29	R/W	0	fall_th1_irq_en
28	R/W	0	fall_th0_irq_en
27	R/W	0	rise_th3_irq_en
26	R/W	0	rise_th2_irq_en
25	R/W	0	rise_th1_irq_en
24	R/W	0	rise_th0_irq_en
23	R/W	0	fall_th3_irq_stat_clr
22	R/W	0	fall_th2_irq_stat_clr
21	R/W	0	fall_th1_irq_stat_clr
20	R/W	0	fall_th0_irq_stat_clr
19	R/W	0	rise_th3_irq_stat_clr
18	R/W	0	rise_th2_irq_stat_clr
17	R/W	0	rise_th1_irq_stat_clr
16	R/W	0	rise_th0_irq_stat_clr
15	R/W	0	1: enable IRQ related function.
14	R/W	0	fast_mode: 0 : downsample unit = 1ms; 1: downsample unit = 1us;
13	R/W	0	clr_hi_temp_stat
12	R/W	0	ts_ana_rset_vbg reset vbg(set 0, if have error place set it plus 01000..)
11	R/W	0	ts_ana_rst_sd reset adc(set 0, if have error place set it plus 01000..)
10	R/W	0	ts_ana_en_vcm enable vcm (disable:0; enable:1)
9	R/W	0	ts_ana_en_vbg enable vbg (disable:0; enable: 1)
8:7	R/W	0	filter hcic mode 0: downsample rate = 128; 1: downsample rate = 256; 2/3: downsample rate = 512;

Bits	R/W	Default	Description
6	R/W	0	filter ts_out_ctrl; 1: add more delay for filter lock;
5	R/W	0	filter en(disable:0; enable:1)
4	R/W	0	ts_ana_en_iptat, useless.
3	R/W	0	Temp Sensor DEM enable. (disable:0; enable:1)
2:0	R/W	0	Bipolar bias current input control. recommend value : 3. ts_ana_ch_sel; 0: 8'b00000001; 1: 8'b00000011; 2: 8'b00000111; 3: 8'b00001111; 4: 8'b00011111; 5: 8'b00111111; 6: 8'b01111111; 7: 8'b11111111;

Table 14-2 TS\_CFG\_REG2 0x002

Bits	R/W	Default	Description
31	R/W	0	hi_temp_enable
30	R/W	0	reset_en, if = 0, will not reset all chip;
27:16	R/W	0	high temperature times, if continuous detect high temperature, then will reset all chip.
15:0	R/W	0	high temperature threshold, if temperature value > this th , mean detected once high temperature

Table 14-3 TS\_CFG\_REG3 0x003

Bits	R/W	Default	Description
31:16	R/W	0	
15:0	R/W	0	down_sample rate

Table 14-4 TS\_CFG\_REG4 0x004

Bits	R/W	Default	Description
23:12	R/W	0	rise_th0
11:0	R/W	0	rise_th1

Table 14-5 TS\_CFG\_REG5 0x005

Bits	R/W	Default	Description
23:12	R/W	0	rise_th2
11:0	R/W	0	rise_th3

**Table 14-6 TS\_CFG\_REG6 0x06**

Bits	R/W	Default	Description
23:12	R/W	0	fall_th0
11:0	R/W	0	fall_th1

**Table 14-7 TS\_CFG\_REG7 0x07**

Bits	R/W	Default	Description
23:12	R/W	0	fall_th2
11:0	R/W	0	fall_th3

**Table 14-8 TS\_STAT0 0x10**

Bits	R/W	Default	Description
31:18	R	0	detect_hi_temp_cnt
17	R	0	detected_hi_temp_stat
16	R	0	filter lock
15:0	R	0	filter out

**Table 14-9 TS\_STAT1 0x11**

Bits	R/W	Default	Description
31:9	R	0	
8	R	0	hi_temp_stat
7	R	0	fall_th3_irq
6	R	0	fall_th2_irq
5	R	0	fall_th1_irq
4	R	0	fall_th0_irq
3	R	0	rise_th3_irq
2	R	0	rise_th2_irq
1	R	0	rise_th1_irq
0	R	0	rise_th0_irq

**Table 14-10 TS\_STAT2 0x12**

Bits	R/W	Default	Description
31:16	R	0	temperature value D2
15:0	R	0	temperature value D1

**Table 14-11 TS\_STAT3 0x13**

Bits	R/W	Default	Description
31:16	R	0	temperature value D4
15:0	R	0	temperature value D3

**Table 14-12 TS\_STAT4 0x14**

Bits	R/W	Default	Description
31:16	R	0	temperature value D6
15:0	R	0	temperature value D5

**Table 14-13 TS\_STAT5 0x15**

Bits	R/W	Default	Description
31:16	R	0	temperature value D8
15:0	R	0	temperature value D7

**Table 14-14 TS\_STAT6 0x16**

Bits	R/W	Default	Description
31:16	R	0	temperature value D10
15:0	R	0	temperature value D9

**Table 14-15 TS\_STAT7 0x17**

Bits	R/W	Default	Description
31:16	R	0	temperature value D12
15:0	R	0	temperature value D11

**Table 14-16 TS\_STAT8 0x18**

Bits	R/W	Default	Description
31:16	R	0	temperature value D14
15:0	R	0	temperature value D13

**Table 14-17 TS\_STAT9 0x19**

Bits	R/W	Default	Description
31:16	R	0	temperature value D16
15:0	R	0	temperature value D15

# 15 Appendix

## 15.1 Net Length

The following table describes the length of all nets on the SoC.

**Table 15-1 Net Length**

Net Name	Net Length(um)
CSI_A_CLKN	5030.487
CSI_A_CLKP	4817.861
CSI_A_D0N	4711.585
CSI_A_D0P	4641.622
CSI_A_D1N	5211.345
CSI_A_D1P	5242.432
CSI_A_D2N	4639.844
CSI_A_D2P	4427.007
CSI_A_D3N	4653.205
CSI_A_D3P	4585.335
CSI_B_CLKN	4655.489
CSI_B_CLKP	4305.908
CSI_C_CLKN	4746.718
CSI_C_CLKP	4489.501
CSI_C_D0N	3502.054
CSI_C_D0P	3578.28
CSI_C_D1N	4059.798
CSI_C_D1P	3795.459
CSI_C_D2N	4346.695
CSI_C_D2P	3920.985
CSI_C_D3N	5064.898
CSI_C_D3P	4795.254
CSI_D_CLKN	5543.884
CSI_D_CLKP	5300.085
DDR0_AC_0	6399.3
DDR0_AC_1	6696.615
DDR0_AC_2	5038.469
DDR0_AC_3	6039.656
DDR0_AC_4	3918.165

Net Name	Net Length(um)
DDR0_AC_5	3929.111
DDR0_AC_6	3854.377
DDR0_AC_7	5378.488
DDR0_AC_8	7017.325
DDR0_AC_9	7211.078
DDR0_AC_10	5655.812
DDR0_AC_11	5389.325
DDR0_AC_12	6631.093
DDR0_AC_13	6215.168
DDR0_AC_14	6097.363
DDR0_AC_15	5612.287
DDR0_AC_20	3523.224
DDR0_AC_21	3500.304
DDR0_AC_22	3367.634
DDR0_AC_23	2940.266
DDR0_AC_24	5768.027
DDR0_AC_25	5929.304
DDR0_AC_26	4427.73
DDR0_AC_28	3227.357
DDR0_AC_29	4306.811
DDR0_AC_30	4369.684
DDR0_AC_31	5228.182
DDR0_AC_32	4015.173
DDR0_AC_33	5314.911
DDR0_AC_34	6854.232
DDR0_AC_35	5712.777
DDR0_AC_36	3944.826
DDR0_AC_37	1641.622
DDR0_AC_38	4189.761
DDR0_DQ0	6280.845
DDR0_DQ1	5166.292
DDR0_DQ2	5018.542
DDR0_DQ3	4418.469
DDR0_DQ4	3402.634
DDR0_DQ5	6233.136

Net Name	Net Length(um)
DDR0_DQ6	3823.584
DDR0_DQ7	5358.769
DDR0_DQ8	5946.746
DDR0_DQ9	3073.453
DDR0_DQ10	2543.422
DDR0_DQ11	4164.59
DDR0_DQ12	2715.204
DDR0_DQ13	3036.421
DDR0_DQ14	3232.893
DDR0_DQ15	4137.148
DDR0_DQ16	3360.979
DDR0_DQ17	3437.216
DDR0_DQ18	3321.299
DDR0_DQ19	3906.145
DDR0_DQ20	4272.024
DDR0_DQ21	4564.25
DDR0_DQ22	3755.071
DDR0_DQ23	3649.98
DDR0_DQ24	3815.958
DDR0_DQ25	3158.116
DDR0_DQ26	2963.749
DDR0_DQ27	4441.774
DDR0_DQ28	2131.234
DDR0_DQ29	2471.88
DDR0_DQ30	2877.104
DDR0_DQ31	3874.563
DDR0_DQM0	5024.362
DDR0_DQM1	3653.29
DDR0_DQM2	5294.813
DDR0_DQM3	3786.985
DDR0_DQSN0	4978.861
DDR0_DQSN1	4091.433
DDR0_DQSN2	3008.052
DDR0_DQSN3	3145.699
DDR0_DQSP0	5049.99



Net Name	Net Length(um)
DDR0_DQSP1	4162.587
DDR0_DQSP2	3220.211
DDR0_DQSP3	3224.322
DDR1_AC_0	4524.667
DDR1_AC_1	4270.125
DDR1_AC_2	5360.315
DDR1_AC_3	4796.72
DDR1_AC_4	3498.185
DDR1_AC_5	3385.999
DDR1_AC_6	4740.332
DDR1_AC_7	5654.689
DDR1_AC_8	5367.616
DDR1_AC_9	5087.63
DDR1_AC_10	6433.623
DDR1_AC_11	6384.632
DDR1_AC_12	4389.482
DDR1_AC_13	4968.912
DDR1_AC_14	6687.743
DDR1_AC_15	2391.553
DDR1_AC_20	5662.318
DDR1_AC_21	5536.219
DDR1_AC_22	5318.645
DDR1_AC_23	4745.42
DDR1_AC_24	3798.847
DDR1_AC_25	3783.427
DDR1_AC_26	8556.225
DDR1_AC_28	4621.285
DDR1_AC_29	2456.664
DDR1_AC_30	3850.486
DDR1_AC_31	4298.882
DDR1_AC_32	4000.001
DDR1_AC_33	4537.187
DDR1_AC_34	2765.553
DDR1_AC_35	2977.044
DDR1_AC_36	1828.854

Net Name	Net Length(um)
DDR1_AC_37	1587.874
DDR1_AC_38	4233.489
DDR1_DQ0	6467.507
DDR1_DQ1	4363.669
DDR1_DQ2	3529.762
DDR1_DQ3	4032.636
DDR1_DQ4	4709.173
DDR1_DQ5	6147.046
DDR1_DQ6	5173.892
DDR1_DQ7	4547.813
DDR1_DQ8	5402.701
DDR1_DQ9	3817.321
DDR1_DQ10	3717.225
DDR1_DQ11	4647.977
DDR1_DQ12	3280.81
DDR1_DQ13	2416.644
DDR1_DQ14	2152.137
DDR1_DQ15	4184.853
DDR1_DQ16	3347.275
DDR1_DQ17	4359.613
DDR1_DQ18	4859.07
DDR1_DQ19	5239.311
DDR1_DQ20	3643.34
DDR1_DQ21	4099.978
DDR1_DQ22	5347.057
DDR1_DQ23	4487.338
DDR1_DQ24	3715.342
DDR1_DQ25	2923.508
DDR1_DQ26	2622.058
DDR1_DQ27	3912.034
DDR1_DQ28	2721.158
DDR1_DQ29	2403.124
DDR1_DQ30	1922.521
DDR1_DQ31	4047.65
DDR1_DQM0	4793.601

Net Name	Net Length(um)
DDR1_DQM1	3320.179
DDR1_DQM2	5089.688
DDR1_DQM3	4554.696
DDR1_DQSN0	5372.125
DDR1_DQSN1	3806.477
DDR1_DQSN2	3229.741
DDR1_DQSN3	3489.591
DDR1_DQSP0	5443.279
DDR1_DQSP1	3752.942
DDR1_DQSP2	3370.435
DDR1_DQSP3	3506.484
ENET_RXN	3845.474
ENET_RXP	4081.691
ENET_TXN	4210.243
ENET_TXP	4466.105
GPIOB_0	2345.776
GPIOB_1	1849.83
GPIOB_2	1979.736
GPIOB_3	2088.165
GPIOB_4	2999.87
GPIOB_5	2703.343
GPIOB_6	3412.689
GPIOB_7	2960.572
GPIOB_8	3595.539
GPIOB_10	2083.686
GPIOB_11	2405.297
HDMIRX_ARCTXN	4966.187
HDMIRX_ARCTXP	5112.399
HDMIRX_A_CLKN	5624.866
HDMIRX_A_CLKP	5678.996
HDMIRX_A_D0N	5413.571
HDMIRX_A_D0P	5390.577
HDMIRX_A_D1N	5775.753
HDMIRX_A_D1P	5648.613
HDMIRX_A_D2N	6659.452

Net Name	Net Length(um)
HDMIRX_A_D2P	6444.356
HDMIRX_B_CLKN	5582.84
HDMIRX_B_CLKP	5245.85
HDMIRX_B_D0N	3781.092
HDMIRX_B_D0P	4138.724
HDMIRX_B_D1N	4626.73
HDMIRX_B_D1P	4496.135
HDMIRX_B_D2N	5178.643
HDMIRX_B_D2P	4997.941
HDMIRX_C_CLKN	6477.257
HDMIRX_C_CLKP	6304.805
HDMIRX_C_D0N	3187.457
HDMIRX_C_D0P	3271.665
HDMIRX_C_D1N	3767.361
HDMIRX_C_D1P	3739.942
HDMIRX_C_D2N	3423.348
HDMIRX_C_D2P	3466.655
HDMITX_ARCRXN	2223.555
HDMITX_ARCRXP	2064.952
HDMITX_CLKN	5014.033
HDMITX_CLKP	4858.154
HDMITX_D0N	4305.429
HDMITX_D0P	4133.65
HDMITX_D1N	4243.009
HDMITX_D1P	3985.343
HDMITX_D2N	3857.55
HDMITX_D2P	3444.897
PCIE_CLK_N	2837.101
PCIE_CLK_P	2590.389
PCIE_RXN	2827.673
PCIE_RXP	2767.197
PCIE_TXN	2031.891
PCIE_TXP	1928.492
USB30_RXN	4221.597
USB30_RXP	4260.951

Net Name	Net Length(um)
USB30_TXN	3499.622
USB30_TXP	3323.657
USB_A_OTG_DM	2948.228
USB_A_OTG_DP	2918.029
USB_B_OTG_DM	4237.569
USB_B_OTG_DP	4265.271
VX1_A_0N	2137.44
VX1_A_0P	2439.972
VX1_A_1N	3100.185
VX1_A_1P	3068.469
VX1_A_2N	4195.452
VX1_A_2P	4078.917
VX1_A_3N	2993.326
VX1_A_3P	2782.911
VX1_A_4N	3900.206
VX1_A_4P	4021.289
VX1_A_5N	1946.558
VX1_A_5P	1988.601
VX1_A_6N	3533.938
VX1_A_6P	3397.285
VX1_A_7N	2700.472
VX1_A_7P	2523.065
VX1_B_0N	3854.896
VX1_B_0P	4013.197
VX1_B_1N	2973.577
VX1_B_1P	2856.923
VX1_B_2N	3806.325
VX1_B_2P	3811.417
VX1_B_3N	3680.499
VX1_B_3P	3819.065
VX1_B_4N	2313.953
VX1_B_4P	2282.565
VX1_B_5N	2451.007
VX1_B_5P	2712.78
VX1_B_6N	2325.928

---

Net Name	Net Length(um)
VX1_B_6P	2450.613
VX1_B_7N	2061.203
VX1_B_7P	2082.518