S905X

Datasheet

Revision: 0.2

Release date: 3/14/2017

Amlogic, Ltd.

DISCLAIMER

Amlogic Ltd. 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 Ltd. are included as a means or 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.

REVISION HISTORY

Revision Number	Revision Date	Changes
0.1	2016/10/18	Initial version release
0.2	2016/12/13	Update VDDIO, VOH, VOL information, update power sequence information, update DDR4/LPDDR3 PINMUX, update GPU information, add TSin timing information, blkmv update to ACODEC, delete BT656_2 information, update GPIO information, update ISA_TIMER_MUX1 description
0.3	2017/03/14	Add register RESET0_MASK~RESET7_MASK, change RESET0_LEVEL ~RESET7_LEVEL address to 0xc11004480~ 0xc1100449c

CONTACT INFORMATION

Amlogic, Shanghai, Ltd.
2518 Mission College Blvd, Suite 120
Santa Clara, CA 95054
U.S.A.
www.amlogic.com

Contents

Section I About This Documentation

1. Documentation Overview

This documentation is an overall Description of Amlogic advanced quad-core OTT/IP Set Top Box(STB) application processor S905X, providing programmers instructions from the following aspects: system, video path, audio path, memory interfaces, I/O interfaces and system interfaces.

2. Acronyms and Abbreviations

Table I.2.1 Acronyms and Abbreviations

A		
AHB	AMBA High-speed Bus	A bus protocal designed for the connection and management of functional blocks in system-on-a-chip (SoC) designs
APB	Advanced Peripheral Bus	A bus protocal designed for low bandwidth control accesses
D		
DMAC	Direct Memory Access Controller	An engine connected to the DDR controller for the purposes of moving data to/from DDR memory.
DMC	DDR memory controller	An engine controls DDR
E		
eMMC	Embedded multimedia card	An architecture consisting of an embedded stortage solution with MMC interface, flash memory and controler
H		
HDMI	High-Definition Multimedian Interface	A compact audio/video interface used for transmitting uncompressed digital data
I		
I2C	Inter-Integrated Circuit	An multi-master, multi-slave, single-ended, serial computer bus used for attaching lower-speed peripheral ICs to processors and microcontrollers.
I2S	Inter IC sound	An electrical serial bus interface stand used for connecting digital audio device together
P		
PCM	Pulse Code Modulation	A method used to digitally represent sampled analog signals
PDM	Pulse Density Modulation	A mothed used to represent an analog signal with digital signal
S		
SPI	Synchronous Peripheral Interface	A synchronous serial data link standard operating in sull duplex mode, devices communicate in master/slave mode where the master device initiates the date frame

Section II General Information

S905X is an advanced application processor designed for OTT/IP Set Top Box(STB) and high-end media box applications. It integrates a powerful CPU/GPU subsystem, a secured 4K video CODEC engine and a best-in-class HDR image processing pipeline with all major peripherals to form the ultimate low power multimedia AP.

The main system CPU is a quad-core ARM Cortex-A53 CPU with L1 instruction/data cache for each core and a large unified L2 cache to improve system performance. In addition, the Cortex-A53 CPU includes the NEON SIMD co-processor to improve software media processing capability. The quad-core ARM Cortex-A53 CPU can be clocked to 1.5GHz and has a wide bus connecting to the memory sub-system.

The graphic subsystem consists of two graphic engines and a flexible video/graphic output pipeline. The five core ARM Mali-450 GPU including dual geometry processors (GP) and triple pixel processors (PP).The multi-core GPU processor handles all OpenGL ES 1.1/2.0 and OpenVG graphics 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, networking, user-interface and gaming related tasks. The video output pipeline includes advanced HDR10 and HLG HDR processing, REC709/BT2020 processing, motion adaptive edge enhancing de-interlacing, flexible programmable scalar, and many picture enhancement filters before passing the enhanced image to the video output ports.

Amlogic Video Engine (AVE-10) offloads the Cortex-A53 CPUs from all video CODEC processing. It includes dedicated hardware video decoder and encoder. AVE-10 is capable of decoding 4Kx2K resolution video at 60fps with complete Trusted Video Path (TVP) for secure applications and supports full formats including MVC, MPEG-1/2/4, VC-1/WMV, AVS, AVS+, RealVideo, MJPEG streams, H.264, H265-10, VP9-10 and also JPEG pictures with no size limitation. The independent encoder is able to encode in JPEG or H.264 up to 1080p at 60fps.

S905X integrates all standard audio/video input/output interfaces including a HDMI2.0b transmitter with 3D, HDR, CEC and HDCP 2.2 support, stereo audio DAC, a CVBS output, PCM, I2S and SPDIF digital audio input/output interfaces, and a stereo PDM digital MIC inputs.

The processor has rich advanced network and peripheral interfaces, including a 10/100M Ethernet MAC with FE PHY interface, dual USB 2.0 high-speed ports (one OTG and one HOST) and multiple SDIO/SD card controllers, UART, I2C, high-speed SPI and PWMs.

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

3. Features

3.1 CPU Architecture

- Quad core ARM Cortex-A53 CPU up to 1.5GHz (DVFS)
- ARMv8-A architecture with Neon and Crypto extensions
- 8-stage in-order full dual issue pipeline
- Unified System L2 cache
- Advanced TrustZone security system
- Application based traffic optimization using internal QoS-based switching fabrics

3.2 GPU Architecture

- Penta-core ARM Mali-450 GPU up to 750MHz+ (DFS)
- Dual Geometry Processors and triple Pixel Processors
- Concurrent multi-core processing
- 2250 Mpix/sec and 165 Mtri/Sec
- Full scene over-sampled 4X anti-aliasing engine with no additional bandwidth usage
- OpenGL ES 1.1/2.0 and OpenVG 1.1 support
- 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

3.3 Crypto Engine

- AES block cipher with 128/192/256 bits keys, standard 16 bytes block size and streaming ECB, CBC and CTR modes
- DES/TDES block cipher with ECB and CBC modes supporting 64 bits key for DES and 192 bits key for 3DES
- Built-in hardware True Random Number Generator (TRNG), CRC and HMAC SHA-1/HMAC SHA-2(SHA-224/SHA-256) engine

3.4 Video Path

- Amlogic Video Engine (AVE) with dedicated hardware decoders and encoders
- Supports multiple “secured” video decoding sessions and simultaneous decoding and encoding
- Video/Picture Decoding
  - VP9 Profile-2 up to 4Kx2K@60fps
  - H.265 HEVC MP-10@L5.1 up to 4Kx2K@60fps
  - H.264 AVC HP@L5.1 up to 4Kx2K@30fps
  - H.264 MVC up to 1080p @60fps
  - 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)
  - RealVideo 8/9/10 up to 1080P
  - WebM up to VGA
  - 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, *.rm and *.jpg file formats
- Video/Picture Encoding
  - Independent JPEG and H.264 encoder with configurable performance/bits-rate
  - JPEG image encoding
  - H.264 video encoding up to 1080P@60fps with low latency

- Built-in HDMI 2.0b transmitter including both controller and PHY with CEC and HDCP 2,2, 4Kx2K@60 max resolution output
- CVBS 480i/576i standard definition output
- Supports all standard SD/HD/FHD video output formats: 480i/p, 576i/p, 720p, 1080i/p and 4Kx2K

3.5 Audio Path

- Supports MP3, AAC, WMA, RM, FLAC, Ogg and programmable with 7.1/5.1 down-mixing
- I2S audio interface supporting 2-channel input and 8-channel output
- Built-in serial digital audio SPDIF/IEC958 output and PCM input/output
- Built-in stereo audio DAC
- Stereo digital microphone PDM input
- Supports concurrent dual audio stereo channel output with combination of analog+PCM or I2S+PCM

3.6 Memory

- 16/32-bit SDRAM memory interface running up to DDR 2400
- Supports up to 3GB DDR3, DDR3L, DDR4, LPDDR2, LPDDR3 with dual ranks
- Supports SLC/MLC/TLC NAND Flash with 60-bit ECC, compatible to Toshiba toggle mode in addition to ONFI 2.2
- SDSC/SDHC/SDXC card and SDIO interface with 1-bit and 4-bit data bus width supporting spec version 2.x/3.x/4.x DS/HS modes up to UHS-I SDR104
- eMMC and MMC card interface with 1/4/8-bit data bus width fully supporting spec version 5.0 HS400
- Supports serial 1, 2 or 4-bit NOR Flash via SPI interface
- Built-in 4k bits One-Time-Programming memory for key storage

3.7 I/O Interfaces

- Network
  - ıntegrated IEEE 802.3 10/100 M Ethernet MAC with 10/100M PHY
  - Supports Wake-On-LAN (WOL) mode
  - Optional 50MHz and 125MHz clock output to Ethernet PHY
  - WiFi/IEEE802.11 & Bluetooth supporting via SDIO/USB/UART/PCM
  - Network interface optimized for mixed WIFI and BT traffic
- Integrated I/O Controllers and Interfaces
  - Dual USB 2.0 high-speed USB I/O, one USB Host and one USB OTG
  - Multiple UART, I2C and SPI interface with slave select
  - Multiple PWMs
  - Programmable IR remote input/output controllers
  - Built-in 12bit SAR ADC with 3 input channels
  - A set of General Purpose IOs with built-in pull up and pull down

3.8 System Interface

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

3.9 Power Management

- Multiple external power domains controlled by PMIC
- Multiple internal power domains controlled by software
- Multiple sleep modes for CPU, system, DRAM, etc.
- Multiple internal PLLs for DVFS operation
- Multi-voltage I/O design for 1.8V and 3.3V
- Power management auxiliary processor in a dedicated always-on (AO) power domain that can communicate with an external PMIC

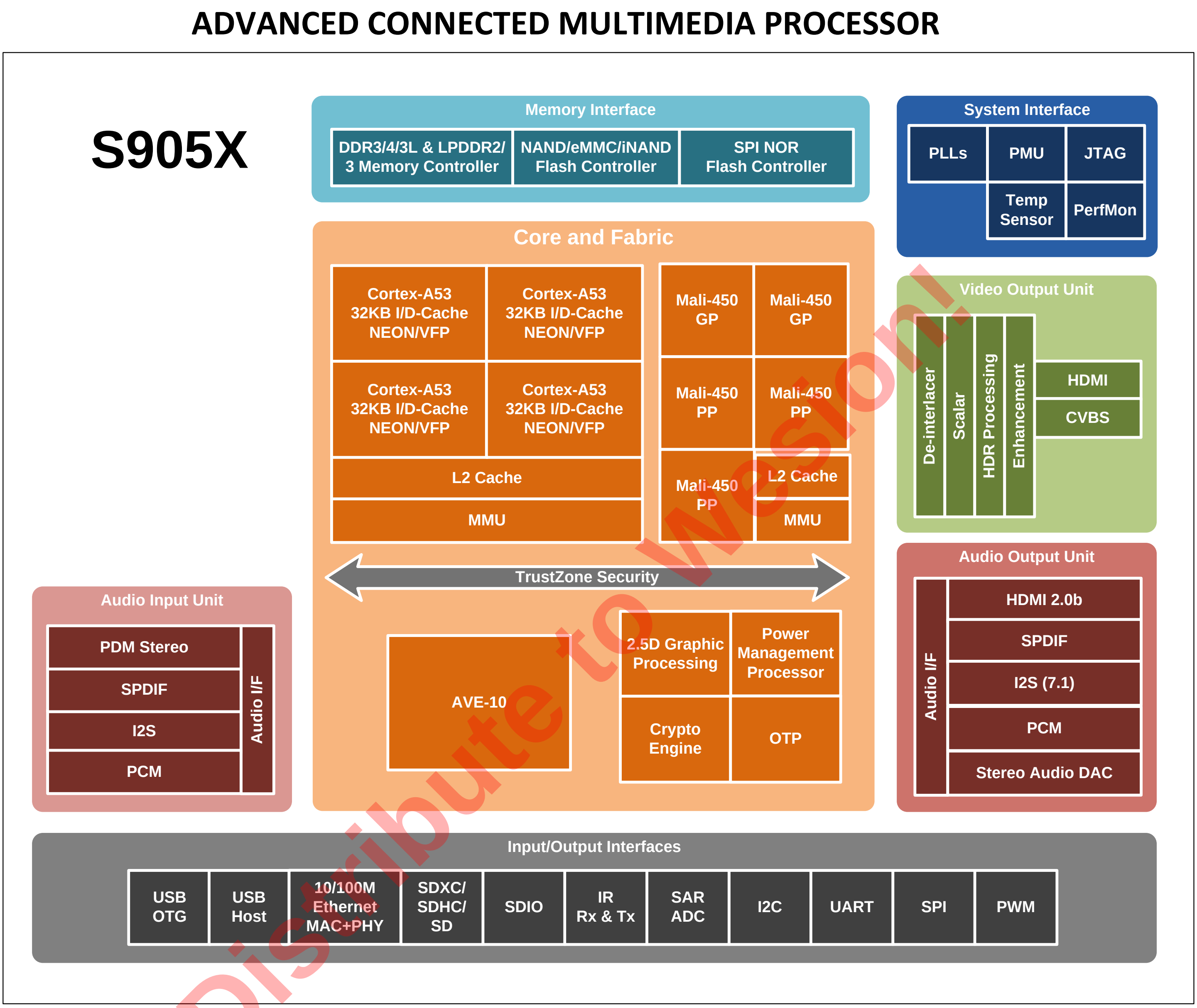
3.10 Security

- 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
- Protected memory regions and electric fence data partition
- Hardware based Trusted Video Path (TVP) and secured contents (needs SecureOS software)
- Secured I/O and secured clock

4. System Block Diagram

Fig II.4.1 shows the structure of S905X.

Fig II.4.1 Block Diagram of S905X



5. Pin-Out Diagram (Top view)

Below is the pin-out diagram of S905X.

FigII.5.1 S905X Pin-out Diagram

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
A	DVSS	GPIOX_0	GPIOX_16	.	.	GPIOX_12	.	DVSS	.	GPIOAO_4	DVSS	.	.	IOVREF_18	USBHOST_A_DP	.	.	ENET_RXN	HDMITX_2P	HDMITX_2N	A
B	GPIOX_3	GPIOX_4	GPIOX_9	GPIOX_8	GPIOX_17	GPIOX_13	GPIOX_18	SARADC_CH0	GPIOAO_1	GPIOAO_3	SARADC_CH1	RESET_N	sys_oscin	AVDD18_XTAL	USBHOST_A_DM	USBOTG_B_DM	ENET_TXN	ENET_RXP	HDMITX_1N	HDMITX_1P	B
C	GPIOX_7	GPIOX_2	GPIOX_1	GPIOX_11	GPIOX_10	GPIOX_14	GPIOX_15	GPIOCLK_1	GPIOAO_0	GPIOAO_2	GPIOAO_6	GPIOAO_9	DVSS	sys_oscout	DVSS	USBOTG_B_DP	ENET_TXP	DVSS	HDMITX_0P		C
D	GPIODV_24	GPIOX_6	GPIOX_5	DVSS	DVSS	CARD_6	CARD_5	CARD_4	CARD_1	EXTC_DPLL	.	TEST_N	GPIOAO_5	ENET_EXTRES	USBOTG_B_ID	USBA_TXRTUNE	USBB_TXRTUNE	DVSS	HDMITX_0N	DVSS	D
E	GPIODV_25	GPIODV_26	GPIOAO_7	GPIODV_28	VDDCPU	VDDCPU	VDDCPU	CARD_3	GPIOCLK_0	CARD_2	CARD_0	VDDIO_AO	VDDAO_0V9	AVDD0V9_ENET	DVSS	USBOTG_B_VBUS	DVSS	GPIOH_2	HDMITX_CKP	HDMITX_CKN	E
F	.	GPIODV_29	GPIODV_27	VDDCPU	VDDCPU	VDDCPU	DVSS	DVSS	SARADC_CH2	AVDD33_HDMITX	VDDIO_CARD	AVDD18_ENET_EFUSE	VDDEE_0V9	VDDEE_0V9	ENET_TEST_ATP	AVDD18_HDMITX	GPIOAO_8	GPIOH_1	.	.	F
G	BOOT_2	BOOT_1	BOOT_0	VDDCPU	VDDCPU	DVSS	DVSS	DVSS	DVSS	VDDIO_DV_CLK	AVDD18_PLL_ADC	AVDD18_USB	DVSS	HDMI_CEXT	AVDD0V9_HDMITX	REFP	VMID	GPIOH_0	GPIOH_4	GPIOH_3	G
H	BOOT_3	BOOT_4	BOOT_5	VDDIO_X	DVSS	VDDEE_0V9	DVSS	DVSS	DVSS	DVSS	VDDIO_HZ	AVDD33_USB	DVSS	HDMI_REXT	AVDD18_HPLL	AVDD18_Audio	.	LOLP	LOLN	AVSS_Audio	H
J	.	BOOT_6	VDDIO_BOOT	VDDEE_0V9	VDDEE_0V9	VDDEE_0V9	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	HPLL_CEXT	DVSS	DVSS	AVSS_Audio	.	LORP	.	J
K	BOOT_8	BOOT_7	DVSS	VDDEE_0V9	VDDEE_0V9	VDDEE_0V9	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	CVBS_COMP	AVDD18_CVBS	LORN	AVSS_CVBS	CVBS_IOUT	K
L	BOOT_9	BOOT_10	BOOT_11	VDDEE_0V9	VDDEE_0V9	VDDEE_0V9	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	VDDEE_0V9	DVSS	DVSS	DVSS	CARD_2	GPIOH_5	CVBS_RSET	CVBS_VREF	L
M	.	BOOT_12	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	CARD_0	CARD_3	DVSS	GPIOH_8	.	M
N	BOOT_14	BOOT_15	BOOT_13	VDDQ	VDDQ	DVSS	DVSS	DVSS	DVSS	AVDD18_PLL_ADC	DVSS	VDDEE_0V9	DVSS	DVSS	DVSS	CARD_4	CARD_1	GPIOH_7	GPIOH_9	GPIOH_6	N
P	DVSS	DVSS	VDDQ	VDDQ	VDDQ	VDDQ	VDDQ	DVSS	VDDQ	VDDQ	DVSS	VDDQ	VDDQ	DVSS	VDDEE_0V9	CARD_6	CARD_5	DVSS	GPIOZ_15	GPIOZ_14	P
R	DVSS	DQ26	PVREF	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	DVSS	PVREF	DVSS	DVSS	RAS_n	ODT1	ODT0	CS1_n	R
T	DQ30	DQ27	DQ18	DQ22	DQ20	DQ25	DQ13	DQ15	DQ0	DVSS	DQM1	DQ8	PZQ	CAS_n	A13	A15	A3	BA_2	CS0_n	BA_0	T
U	DQ31	DQ16	DVSS	.	DQ19	DQ29	.	DQ9	DVSS	.	DQ2	DQ10	.	A1	WE_n	.	A9	DVSS	A7	A5	U
V	DVSS	DQSP3	DQ17	DQ21	DVSS	DQ24	DQ4	DQ5	DQ6	DVSS	DVSS	DQ3	A8	CKE1	DVSS	CKE0	DVSS	A4	A0	A2	V
W	DQSN3	DVSS	DQM2	DQM3	DQ28	DVSS	DQ1	DVSS	DQM0	DQSP1	DQSN0	DQ7	DVSS	DQ12	CK_N	RST_N	A10	BA_1	DVSS	A11	W
Y	DQSP2	DQSN2	DQ23	.	.	DQ11	DVSS	.	.	DQSN1	DQSP0	.	.	DQ14	CK	.	.	A12	A6	A14	Y
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	

6. Pin Description

Table II.6.1 gives the pin Description of S905X.

Table II.6.1 Pin Description

Ball#	Net Name	Type	Default Pull UP/DN	Description	Power Domain	If Unused
GPIOX - Refer to Table II.7.1 for functional multiplex information.						
A2	GPIOX_0	DIO_4mA	UP	General purpose input/output bank X signal 0	VDDIO_X	-
C3	GPIOX_1	DIO_4mA	UP	General purpose input/output bank X signal 1	VDDIO_X	-
C2	GPIOX_2	DIO_4mA	UP	General purpose input/output bank X signal 2	VDDIO_X	-
B1	GPIOX_3	DIO_4mA	UP	General purpose input/output bank X signal 3	VDDIO_X	-
B2	GPIOX_4	DIO_4mA	UP	General purpose input/output bank X signal 4	VDDIO_X	-
D3	GPIOX_5	DIO_4mA	UP	General purpose input/output bank X signal 5	VDDIO_X	-
D2	GPIOX_6	DIO_2mA	DOWN	General purpose input/output bank X signal 6	VDDIO_X	-
C1	GPIOX_7	DIO_2mA	UP	General purpose input/output bank X signal 7	VDDIO_X	-
B4	GPIOX_8	DIO_3mA	UP	General purpose input/output bank X signal 8	VDDIO_X	-
B3	GPIOX_9	DIO_2mA	UP	General purpose input/output bank X signal 9	VDDIO_X	-
C5	GPIOX_10	DIO_2mA	UP	General purpose input/output bank X signal 10	VDDIO_X	-
C4	GPIOX_11	DIO_3mA	UP	General purpose input/output bank X signal 11	VDDIO_X	-
A6	GPIOX_12	DIO_2mA	UP	General purpose input/output bank X signal 12	VDDIO_X	-
B6	GPIOX_13	DIO_2mA	UP	General purpose input/output bank X signal 13	VDDIO_X	-
C6	GPIOX_14	DIO_2mA	UP	General purpose input/output bank X signal 14	VDDIO_X	-
C7	GPIOX_15	DIO_2mA	UP	General purpose input/output bank X signal 15	VDDIO_X	-
A3	GPIOX_16	DIO_2mA	UP	General purpose input/output bank X signal 16	VDDIO_X	-
B5	GPIOX_17	DIO_2mA	DOWN	General purpose input/output bank X signal 17	VDDIO_X	-
B7	GPIOX_18	DIO_2mA	Up	General purpose input/output bank X signal 18	VDDIO_X	-
H4	VDDIO_X	P	-	Power supply for GPIO bank X	-	-
GPIODV - Refer to Table II.7.2 for functional multiplex information.						
D1	GPIODV_24	DIO_4mA	Z	General purpose input/output bank DV signal 24	GPIO_DV_CLK	-
E1	GPIODV_25	DIO_4mA	Z	General purpose input/output bank DV signal 25	GPIO_DV_CLK	-
E2	GPIODV_26	DIO_4mA	Z	General purpose input/output bank DV signal 26	GPIO_DV_CLK	-
E4	GPIODV_27	DIO_4mA	Z	General purpose input/output bank DV signal 27	GPIO_DV_CLK	-

Ball#	Net Name	Type	Default Pull UP/DN	Description	Power Domain	If Unused
F2	GPIODV_28	DIO_2mA	Z	General purpose input/output bank DV signal 28	GPIO_DV_CLK	-
F3	GPIODV_29	DIO_2mA	Z	General purpose input/output bank DV signal 29	GPIO_DV_CLK	-
G10	VDDIO_DV_CLK	P	-	Power supply for GPIO bank DV and CLK	-	-
CARD - Refer to Table II.7.3 for functional multiplex information.						
E1, M16	CARD_0	DIO_4mA	UP	General purpose input/output bank CARD signal 0	VDDIO_CARD	-
D9,N17	CARD_1	DIO_4mA	UP	General purpose input/output bank CARD signal 1	VDDIO_CARD	-
E10,L17	CARD_2	DIO_4mA	UP	General purpose input/output bank CARD signal 2	VDDIO_CARD	-
E8, M17	CARD_3	DIO_4mA	UP	General purpose input/output bank CARD signal 3	VDDIO_CARD	-
D8, N16	CARD_4	DIO_4mA	UP	General purpose input/output bank CARD signal 4	VDDIO_CARD	-
D7,P17	CARD_5	DIO_4mA	UP	General purpose input/output bank CARD signal 5	VDDIO_CARD	-
D6,P16	CARD_6	DIO_2mA	UP	General purpose input/output bank CARD signal 6	VDDIO_CARD	-
F11	VDDIO_CARD	P	-	Power supply for GPIO bank CARD	-	-
BOOT - Refer to Table II.7.4 for functional multiplex information.						
G3	BOOT_0	DIO_4mA	UP	General purpose input/output bank BOOT signal 0	VDDIO_BOOT	-
G2	BOOT_1	DIO_4mA	UP	General purpose input/output bank BOOT signal 1	VDDIO_BOOT	-
G1	BOOT_2	DIO_4mA	UP	General purpose input/output bank BOOT signal 2	VDDIO_BOOT	-
H1	BOOT_3	DIO_4mA	UP	General purpose input/output bank BOOT signal 3	VDDIO_BOOT	-
H2	BOOT_4	DIO_4mA	UP	General purpose input/output bank BOOT signal 4	VDDIO_BOOT	-
H3	BOOT_5	DIO_4mA	UP	General purpose input/output bank BOOT signal 5	VDDIO_BOOT	-
J2	BOOT_6	DIO_4mA	UP	General purpose input/output bank BOOT signal 6	VDDIO_BOOT	-
K2	BOOT_7	DIO_4mA	UP	General purpose input/output bank BOOT signal 7	VDDIO_BOOT	-
K1	BOOT_8	DIO_4mA	UP	General purpose input/output bank BOOT signal 8	VDDIO_BOOT	-
L1	BOOT_9	DIO_4mA	UP	General purpose input/output bank BOOT signal 9	VDDIO_BOOT	-
L2	BOOT_10	DIO_4mA	UP	General purpose input/output bank BOOT signal 10	VDDIO_BOOT	-
L3	BOOT_11	DIO_4mA	UP	General purpose input/output bank BOOT signal 11	VDDIO_BOOT	-
M2	BOOT_12	DIO_4mA	UP	General purpose input/output bank BOOT signal 12	VDDIO_BOOT	-
N1	BOOT_13	DIO_4mA	UP	General purpose input/output bank BOOT signal 13	VDDIO_BOOT	-
N2	BOOT_14	DIO_4mA	UP	General purpose input/output bank BOOT signal 14	VDDIO_BOOT	-
N3	BOOT_15	DIO_4mA	UP	General purpose input/output bank BOOT signal 15	VDDIO_BOOT	-

Ball#	Net Name	Type	Default Pull UP/DN	Description	Power Domain	If Unused
J3	VDDIO_BOOT	P	-	Power supply for GPIO bank BOOT	-	-
GPIOH - Refer to Table II.7.5 for functional multiplex information.						
G18	GPIOH_0	DIO_OD	Z	General purpose input/output bank H signal 0	VDDIO_HZ	-
F18	GPIOH_1	DIO_OD	Z	General purpose input/output bank H signal 1	VDDIO_HZ	-
E18	GPIOH_2	DIO_OD	Z	General purpose input/output bank H signal 2	VDDIO_HZ	-
G20	GPIOH_3	DIO_OD	Z	General purpose input/output bank H signal 3	VDDIO_HZ	-
G19	GPIOH_4	DIO_3mA	UP	General purpose input/output bank H signal 4	VDDIO_HZ	
L18	GPIOH_5	DIO_2mA	DOWN	General purpose input/output bank H signal 5	VDDIO_HZ	
N20	GPIOH_6	DIO_2mA	UP	General purpose input/output bank H signal 6	VDDIO_HZ	
N18	GPIOH_7	DIO_2mA	UP	General purpose input/output bank H signal 7	VDDIO_HZ	
M19	GPIOH_8	DIO_2mA	UP	General purpose input/output bank H signal 8	VDDIO_HZ	
N19	GPIOH_9	DIO_2mA	UP	General purpose input/output bank H signal 9	VDDIO_HZ	
H11	VDDIO_HZ	P	-	Power supply for GPIO bank H and USB	-	-
GPIOZ - Refer to Table II.7.6 for functional multiplex information.						
P20	GPIOZ_14	DIO_2mA	DOWN	General purpose input/output bank H signal 14	VDDIO_HZ	-
P19	GPIOZ_15	DIO_2mA	UP	General purpose input/output bank H signal 15	VDDIO_HZ	-
H11	VDDIO_HZ	P	-	Power supply for GPIO bank Z	-	-
GPIOAO - Refer to Table II.7.7 for functional multiplex information.						
C9	GPIOAO_0	DIO_2mA	UP	General purpose input/output bank AO signal 0	VDDIO_AO	-
B9	GPIOAO_1	DIO_2mA	UP	General purpose input/output bank AO signal 1	VDDIO_AO	-
C10	GPIOAO_2	DIO_2mA	UP	General purpose input/output bank AO signal 2	VDDIO_AO	-
B10	GPIOAO_3	DIO_2mA	Z	General purpose input/output bank AO signal 3	VDDIO_AO	-
A10	GPIOAO_4	DIO_2mA	UP	General purpose input/output bank AO signal 4	VDDIO_AO	-
D13	GPIOAO_5	DIO_2mA	UP	General purpose input/output bank AO signal 5	VDDIO_AO	-
C11	GPIOAO_6	DIO_2mA	UP	General purpose input/output bank AO signal 6	VDDIO_AO	-
E3	GPIOAO_7	DIO_2mA	UP	General purpose input/output bank AO signal 7	VDDIO_AO	-
F17	GPIOAO_8	DIO_2mA	UP	General purpose input/output bank AO signal 8	VDDIO_AO	-
C12	GPIOAO_9	DIO_3mA	UP	General purpose input/output bank AO signal 9	VDDIO_AO	-
D12	TEST_N	DIO_2mA	Up	General purpose input/output bank AO signal 14. Should be pulled up during normal	VDDIO_AO	-

Ball#	Net Name	Type	Default Pull UP/DN	Description	Power Domain	If Unused
B12	RESET_N	DI	DOWN	System reset input	VDDIO_AO	-
E12	VDDIO_AO	P	-	Power supply for GPIO bank AO	-	-
GPIOCLK - Refer to Table II.7.8 for functional multiplex information.						
E9	GPIOCLK_0	DIO_3mA	UP	General purpose input/output bank CLK signal 0	VDDIO_DV_CLK	-
C8	GPIOCLK_1	DIO_3mA	UP	General purpose input/output bank CLK signal 1	VDDIO_DV_CLK	-
G10	VDDIO_DV_CLK	P	-	Power supply for GPIO bank CLK and crystal	-	-
SARADC						
B8	SARADC_CH0	AI	-	ADC channel 0 input	AVDD18_PLL_ADC	-
B11	SARADC_CH1	AI	-	ADC channel 1 input	AVDD18_PLL_ADC	-
F9	SARADC_CH2	AI	-	ADC channel 2 input	AVDD18_PLL_ADC	
G11,N10	AVDD18_PLL_ADC	AP	-	Analog power supply for SARADC and USB	-	-
CVBS						
K16	CVBS_COMP	A	-	CVBS external compensation capacitor connection	AVDD18_CVBS	NC
L19	CVBS_RSET	A	-	CVBS output strength setting resistor	AVDD18_CVBS	NC
L20	CVBS_VREF	A	-	CVBS reference voltage filter cap	AVDD18_CVBS	NC
K20	CVBS_IOUT	AO	-	CVBS output	AVDD18_CVBS	NC
K17	AVDD18_CVBS	P	-	Analog power supply for CVBS	-	To 1.8V
K19	AVSS_CVBS	P	-	Analog power ground for CVBS	-	To VSS
HDMI						
C19	HDMITX_0P	AO	-	HDMI TMDS data0 positive output	HDMITX_AVDD33	NC
D19	HDMITX_0N	AO	-	HDMI TMDS data0 negative output	HDMITX_AVDD33	NC
B20	HDMITX_1P	AO	-	HDMI TMDS data1 positive output	HDMITX_AVDD33	NC
B19	HDMITX_1N	AO	-	HDMI TMDS data1 negative output	HDMITX_AVDD33	NC
A19	HDMITX_2P	AO	-	HDMI TMDS data1 positive output	HDMITX_AVDD33	NC
A20	HDMITX_2N	AO	-	HDMI TMDS data1 negative output	HDMITX_AVDD33	NC
E19	HDMITX_CKP	AO	-	HDMI TMDS clock positive output	HDMITX_AVDD33	NC
E20	HDMITX_CKN	AO	-	HDMI TMDS clock negative output	HDMITX_AVDD33	NC
H14	HDMI_REXT	A	-	HDMI output strength setting resistor	HDMI_AVDD18	NC
G14	HDMI_CEXT	A	-	HDMI TX external filter cap	HDMI_AVDD18	NC

Ball#	Net Name	Type	Default Pull UP/DN	Description	Power Domain	If Unused
J14	HPLL_CEXT	A	-	HPLL external filter cap	AVDD18_HPLL	NC
F10	AVDD33_HDMITX	P	-	Analog power supply 3.3V for HDMI	-	To 3.3V
H15	AVDD18_HPLL	P	-	Power supply for HDMI PLL	-	To 1.8V
F16	AVDD18_HDMITX	P	-	Analog power supply 1.8V for HDMI	-	To 1.8V
G15	AVDD09_HDMITX	P	-	Power supply 0.9V for HDMI	-	To 0.9V
DRAM						
V19	A0	DO	-	DDR3/LPDDR3/DDR4 A0	VDDQ	-
U14	A1	DO	-	DDR3/LPDDR3/DDR4 A1	VDDQ	-
V20	A2	DO	-	DDR3/LPDDR3/DDR4 A2	VDDQ	-
T17	A3	DO	-	DDR3/LPDDR3/DDR4 A3	VDDQ	-
V18	A4	DO	-	DDR3/LPDDR3/DDR4 A4	VDDQ	-
U20	A5	DO	-	DDR3/LPDDR3/DDR4 A5	VDDQ	-
Y19	A6	DO	-	DDR3/LPDDR3/DDR4 A6	VDDQ	-
U19	A7	DO	-	DDR3/LPDDR3/DDR4 A7	VDDQ	-
V13	A8	DO	-	DDR3/LPDDR3/DDR4 A8	VDDQ	-
U17	A9	DO	-	DDR3/LPDDR3/DDR4 A9	VDDQ	-
W17	A10	DO	-	DDR3/DDR4 A10	VDDQ	-
W20	A11	DO	-	DDR3/DDR4 A11	VDDQ	-
Y18	A12	DO	-	DDR3/DDR4 A12	VDDQ	-
T15	A13	DO	-	DDR3/DDR4 A13	VDDQ	-
Y20	A14	DO	-	DDR3/DDR4 A 14	VDDQ	-
T16	A15	DO	-	DDR3/DDR4 A15	VDDQ	-
T20	BA_0	DO	-	DDR3/DDR4 BA 0	VDDQ	-
W18	BA_1	DO	-	DDR3/DDR4 BA1	VDDQ	-
T18	BA_2	DO	-	DDR3 BA2 DDR4 BG0	VDDQ	-
T14	CAS_N	DO	-	DDR3 CAS_N DDR4 ACT_N	VDDQ	-
Y15	CK	DO	-	DDR3/LPDDR3/DDR4 CK	VDDQ	-
W15	CK_N	DO	-	DDR3/LPDDR3/DDR4 CK_N	VDDQ	-
V16	CKE0	DO	-	DDR3/LPDDR3/DDR4 CKE0	VDDQ	-

Ball#	Net Name	Type	Default Pull UP/DN	Description	Power Domain	If Unused
V14	CKE1	DO	-	DDR3/LPDDR3/DDR4 CKE1	VDDQ	-
T19	CS0_N	DO	-	DDR3/LPDDR3/DDR4 CS0_N	VDDQ	-
R20	CS1_N	DO	-	DDR3/LPDDR3/DDR4 CS1_N	VDDQ	-
R19	ODT0	DO	-	DDR3/LPDDR3/DDR4 ODT0	VDDQ	-
R18	ODT1	DO	-	DDR3/DDR4 ODT1	VDDQ	-
R17	RAS_N	DO	-	DDR3 RAS_N DDR4 A16	VDDQ	-
U15	WE_N	DO	-	DDR3 WE_N DDR4 BG1	VDDQ	-
W16	RST_N	DO	-	DDR3/DDR4 RST_N	VDDQ	-
T9	DQ0	DIO	-	DDR3/LPDDR3/DDR4 DQ0	VDDQ	-
W7	DQ1	DIO	-	DDR3/LPDDR3/DDR4 DQ1	VDDQ	-
U11	DQ2	DIO	-	DDR3/LPDDR3/DDR4 DQ2	VDDQ	-
V12	DQ3	DIO	-	DDR3/LPDDR3/DDR4 DQ3	VDDQ	-
V7	DQ4	DIO	-	DDR3/LPDDR3/DDR4 DQ4	VDDQ	-
V8	DQ5	DIO	-	DDR3/LPDDR3/DDR4 DQ5	VDDQ	-
V9	DQ6	DIO	-	DDR3/LPDDR3/DDR4 DQ6	VDDQ	-
W12	DQ7	DIO	-	DDR3/LPDDR3/DDR4 DQ7	VDDQ	-
T12	DQ8	DIO	-	DDR3/LPDDR3/DDR4 DQ8	VDDQ	-
U8	DQ9	DIO	-	DDR3/LPDDR3/DDR4 DQ9	VDDQ	-
U12	DQ10	DIO	-	DDR3/LPDDR3/DDR4 DQ10	VDDQ	-
Y6	DQ11	DIO	-	DDR3/LPDDR3/DDR4 DQ11	VDDQ	-
W14	DQ12	DIO	-	DDR3/LPDDR3/DDR4 DQ12	VDDQ	-
T7	DQ13	DIO	-	DDR3/LPDDR3/DDR4 DQ13	VDDQ	-
Y14	DQ14	DIO	-	DDR3/LPDDR3/DDR4 DQ14	VDDQ	-
T8	DQ15	DIO	-	DDR3/LPDDR3/DDR4 DQ15	VDDQ	-
U2	DQ16	DIO	-	DDR3/LPDDR3/DDR4 DQ16	VDDQ	-
V3	DQ17	DIO	-	DDR3/LPDDR3/DDR4 DQ17	VDDQ	-
T3	DQ18	DIO	-	DDR3/LPDDR3/DDR4 DQ18	VDDQ	-
U5	DQ19	DIO	-	DDR3/LPDDR3/DDR4 DQ19	VDDQ	-
T5	DQ20	DIO	-	DDR3/LPDDR3/DDR4 DQ20	VDDQ	-

Ball#	Net Name	Type	Default Pull UP/DN	Description	Power Domain	If Unused
V4	DQ21	DIO	-	DDR3/LPDDR3/DDR4 DQ21	VDDQ	-
T4	DQ22	DIO	-	DDR3/LPDDR3/DDR4 DQ22	VDDQ	-
Y3	DQ23	DIO	-	DDR3/LPDDR3/DDR4 DQ23	VDDQ	-
V6	DQ24	DIO	-	DDR3/LPDDR3/DDR4 DQ24	VDDQ	-
T6	DQ25	DIO	-	DDR3/LPDDR3/DDR4 DQ25	VDDQ	-
R2	DQ26	DIO	-	DDR3/LPDDR3/DDR4 DQ26	VDDQ	-
T2	DQ27	DIO	-	DDR3/LPDDR3/DDR4 DQ27	VDDQ	-
W5	DQ28	DIO	-	DDR3/LPDDR3/DDR4 DQ28	VDDQ	-
U6	DQ29	DIO	-	DDR3/LPDDR3/DDR4 DQ29	VDDQ	-
T1	DQ30	DIO	-	DDR3/LPDDR3/DDR4 DQ30	VDDQ	-
U1	DQ31	DIO	-	DDR3/LPDDR3/DDR4 DQ31	VDDQ	-
W9	DQM0	DO	-	DDR3/LPDDR3/DDR4 DQM0	VDDQ	-
T11	DQM1	DO	-	DDR3/LPDDR3/DDR4 DQM1	VDDQ	-
W3	DQM2	DO	-	DDR3/LPDDR3/DDR4 DQM2	VDDQ	-
W4	DQM3	DO	-	DDR3/LPDDR3/DDR4 DQM3	VDDQ	-
Y11	DQSP0	DIO	-	DDR3/LPDDR3/DDR4 DQSP0	VDDQ	-
W11	DQSN0	DIO	-	DDR3/LPDDR3/DDR4 DQSN0	VDDQ	-
W10	DQSP1	DIO	-	DDR3/LPDDR3/DDR4 DQSP1	VDDQ	-
Y10	DQSN1	DIO	-	DDR3/LPDDR3/DDR4 DQSN1	VDDQ	-
Y1	DQSP2	DIO	-	DDR3/LPDDR3/DDR4 DQSP2	VDDQ	-
Y2	DQSN2	DIO	-	DDR3/LPDDR3/DDR4 DQSN2	VDDQ	-
V2	DQSP3	DIO	-	DDR3/LPDDR3/DDR4 DQSP3	VDDQ	-
W1	DQSN3	DIO	-	DDR3/LPDDR3/DDR4 DQSN3	VDDQ	-
R3,R14	PVREF	A	-	DRAM reference voltage	VDDQ	-
T13	PZQ	A	-	DRAM reference pin for ZQ calibration	VDDQ	-
USB						
B15	USBHOST_A_DP	AIO	-	USB host positive data signal	AVDD33_USB	NC
A15	USBHOST_A_DM	AIO	-	USB host negative data signal	AVDD33_USB	NC
D16	USBA_TXRTUNE	AIO	-	USB host output strength setting resistor	AVDD18_USB	NC

Ball#	Net Name	Type	Default Pull UP/DN	Description	Power Domain	If Unused
C16	USBOTG_B_DP	AIO	-	USB OTG positive data signal	AVDD33_USB	NC
B16	USBOTG_B_DM	AIO	-	USB OTG negative data signal	AVDD33_USB	NC
E16	USBOTG_B_VBUS	AI	-	USB OTG cable power detection (5V tolerance)	AVDD18_USB	NC
D15	USBOTG_B_ID	DI	-	USB OTG mini-receptacle identifier (Internal 10KΩ pull-up resistor to AVDD18)	AVDD18_USB	NC
D17	USBB_TXRTUNE	AIO	-	USB OTG output strength setting resistor	AVDD18_USB	NC
H12	AVDD33_USB	AP	-	Analog 3.3V power supply for USB	-	To 3.3V
G12	AVDD18_USB	AP	-	Analog 1.8V power supply for USB	-	To 1.8V
Ethernet						
D14	ENET_EXTRES	A	-	Ethernet PHY external resistor connection	AVDD18_ENET_EFUSE	NC
A18	ENET_RXN	AI	-	Ethernet PHY receive date negative input	AVDD18_ENET_EFUSE	NC
B18	ENET_RXP	AI	-	Ethernet PHY receive data positive input	AVDD18_ENET_EFUSE	NC
F15	ENET_TEST_ATP	AO	-	Ethernet PHY Test pin	AVDD18_ENET_EFUSE	NC
B17	ENET_TXN	AO	-	Ethernet PHY transmit data negative output	AVDD18_ENET_EFUSE	NC
C17	ENET_TXP	AO	-	Ethernet PHY transmit data positive output	AVDD18_ENET_EFUSE	NC
F12	AVDD18_ENET_EFU	AP	-	Analog 1.8V power supply for Ethernet and EFUSE	-	To 1.8V
E14	AVDD0V9_ENET	AP	-	Analog 0.9V power supply for Ethernet module	-	To 0.9V
Audio						
H19	LOLN	AO	-	Audio DAC line-out left channel negative signal	AVDD18_Audio	NC
H18	LOLP	AO	-	Audio DAC line-out left channel positive signal	AVDD18_Audio	NC
K18	LORN	AO	-	Audio DAC line-out right channel negative signal	AVDD18_Audio	NC
J19	LORP	AO	-	Audio DAC line-out right channel positive signal	AVDD18_Audio	NC
G16	REFP	A	-	Audio DAC positive reference voltage	AVDD18_Audio	NC
G17	VMID	A	-	Audio DAC external filter cap connection	AVDD18_Audio	NC
H16	AVDD18_Audio	AP	-	Analog 1.8V for Audio DAC	-	To 1.8V
H20,J17	AVSS_Audio	AP	-	Analog power ground for Audio DAC	-	To VSS
System Clock & PLL						
B13	sys_osc_in	AI	-	24MHz crystal oscillator input	AVDD18_XTAL	-
C14	sys_osc_out	AO	-	24MHz crystal oscillator output	AVDD18_XTAL	-
D10	EXTC_DPLL	A	-	DPLL external filter cap	AVDD18_PLL_ADC	-

Ball#	Net Name	Type	Default Pull UP/DN	Description	Power Domain	If Unused
B14	AVDD18_XTAL	P	-	Analog 1.8V power supply for XTAL	-	-
Digital Power						
E5，E6，E7，F4，F5，F6，G4，G5	VDDCPU	P	-	Power supply for CPU (Cortex A5)	-	-
N4,N5,P3,P4,P5,P6,P7,P9,P10,P12,P13	VDDQ	P	-	Power supply for DDR	-	-
H6,J4,J5,J6,K4,K5,K6,L4,L5,L6,L13,F13,F14,N12,P15	VDDEE_0V9	P	-	Power Supply for GPU and core logic	-	-
E13	VDDAO_0V9	P	-	Power supply 0.9V for AO domain	-	-
A14	IOVREF_18	P	-	1.8V Power supply for IOVREF	-	-
Digital Ground						
A8,A11,H5,H7,H8,H9,H10,H13,J7,J8,J9,J10,J11,J12,J13,J15,J16,K7,K8,K9,K10,K11,K12,R7,R8,R9,R10,R11.R12,R13,R15,R16,T10,D4,D5,K13,K14,K15,L7,L8,D18,D20,E15,L9,L10,L11,L12,L14,L15,L16,M3,M4,M5,M6,M7,M8,M9,M10,M11,M13,M14,M15,M18,U3,U9,U18,V1,V5,V10,V11,V15,V17,E17,F7,F8,G6,G7,G8,G9,G13,N6,N7,N8,N9,N11,N13,N14,N15,P1,P2,P8,P11,P14,P18,R1,R4,R5,R6,W2,W6,W8,W13,W19,Y7,	DVSS	P	-	Digital power ground	-	-

Ball#	Net Name	Type	Default Pull UP/DN	Description	Power Domain	If Unused
A1,K3,C13,C15,C18						

Note 1: USB_TXRTUNE can only be NC when both USBA/USBB are not used.

Abbreviations:

- DI = Digital input pin
- DO = Digital output pin
- DIO = Digital input/output pin
- DIO_OD = 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
- Z = Tri-State

7. Pin Multiplexing

Multiple usage pins are used to conserve pin consumption for different features. The S905X devices can be used in many different applications but each application will not utilize all the on chip 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 II.7.1 GPIOX_x Multi-Function Pin

Pin Name	Func1	Func2	Func3	Func4
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_A			
GPIOX_7	SDIO_IRQ	PWM_F		
GPIOX_8	PCM_OUT_A	UART_TX_C		SPI_MOSI
GPIOX_9	PCM_IN_A	UART_RX_C		SPI_MISO
GPIOX_10	PCM_FS_A	UART_CTS_C	I2C_SDA_D	SPI_SS0
GPIOX_11	PCM_CLK_A	UART_RTS_C	I2C_SCK_D	SPI_SCLK
GPIOX_12	UART_TX_A			
GPIOX_13	UART_RX_A			
GPIOX_14	UART_CTS_A			
GPIOX_15	UART_RTS_A			
GPIOX_16	PWM_E			
GPIOX_17	GPIO(BT_EN)			
GPIOX_18	GPIO(BT_WAKE_CPU)			

Table II.7.2 GPIODV_x Multi-Function Pin

Pin Name	Func1	Func2	Func3
GPIODV_24	UART_TX_B	DMIC_IN	I2C_SDA_A
GPIODV_25	UART_RX_B	DMIC_CLK_OUT	I2C_SCK_A
GPIODV_26	UART_CTS_B		I2C_SDA_B
GPIODV_27	UART_RTS_B		I2C_SCK_B
GPIODV_28	PWM_D	REMOTE_INPUT	I2C_SDA_C
GPIODV_29	PWM_B	PWM_VS	I2C_SCK_C

Table II.7.3 CARD_x Multi-Function Pin

Pin Name	Func1	Func2	Func3	Func4
CARD_0	SDCARD_D1			JTAG_TDI
CARD_1	SDCARD_D0			JTAG_TDO
CARD_2	SDCARD_CLK			JTAG_CLK
CARD_3	SDCARD_CMD			JTAG_TMS
CARD_4	SDCARD_D3	UART_TX_AO_A	UART_RX_AO_A	
CARD_5	SDCARD_D2	UART_RX_AO_A	UART_TX_AO_A	
CARD_6	GPIO(Card_DET / EN)			

Table II.7.4 BOOT_x Multi-Function Pin

Pin Name	Func1	Func2	Func3
BOOT_0	EMMC_NAND_D0		
BOOT_1	EMMC_NAND_D1		
BOOT_2	EMMC_NAND_D2		
BOOT_3	EMMC_NAND_D3		
BOOT_4	EMMC_NAND_D4		
BOOT_5	EMMC_NAND_D5		
BOOT_6	EMMC_NAND_D6		
BOOT_7	EMMC_NAND_D7		
BOOT_8	EMMC_CLK	NAND_CE0	
BOOT_9		NAND_CE1	
BOOT_10	EMMC_CMD	NAND_RB0	
BOOT_11		NAND_ALE	NOR_D
BOOT_12		NAND_CLE	NOR_Q
BOOT_13		NAND_WEN_CLK	NOR_C
BOOT_14		NAND_REN_WR	
BOOT_15	EMMC_DS	NAND_DQS	NOR_CS

Table II.7.5 GPIOH_x Multi-Function Pin

Pin Name	Func1	Func2	Func3	Func4
GPIOH_0	HDMI_HPD			
GPIOH_1	HDMI_SDA			
GPIOH_2	HDMI_SCL			
GPIOH_3				
GPIOH_4	SPDIF_OUT	SPDIF_IN		
GPIOH_5				
GPIOH_6		JTAG_TCK	I2S_AM_CLK	
GPIOH_7		JTAG_TMS	I2S_AO_CLK_OUT	I2S_AO_CLK_IN
GPIOH_8		JTAG_TDI	I2S_LR_CLK_OUT	I2S_LR_CLK_IN
GPIOH_9		JTAG_TDO	I2SOUT_CH01	

Table II.7.6 GPIOZ_x Multi-Function Pin

Pin Name	Func1	Func2
GPIOZ_14	SPDIF_IN	ETH_LINK_LED
GPIOZ_15	PWM_C	ETH_ACT_LED

Table II.7.7 GPIOAO_x Multi-Function Pin

Pin Name	Func1	Func2	Func3	Func4
GPIOAO_0	UART_TX_AO_A	UART_TX_AO_B		
GPIOAO_1	UART_RX_AO_A	UART_RX_AO_B		
GPIOAO_2	UART_CTS_AO_A	UART_CTS_AO_B		
GPIOAO_3	UART_RTS_AO_A	UART_RTS_AO_B		PWM_AO_A
GPIOAO_4	UART_TX_AO_B	I2C_SCK_AO		
GPIOAO_5	UART_RX_AO_B	I2C_SDA_AO		
GPIOAO_6	CLK_32K_IN	I2S_IN_01	SPDIF_OUT	PWM_AO_B
GPIOAO_7	REMOTE_INPUT	REMOTE_OUTPUT		
GPIOAO_8	AO_CEC	EE_CEC	I2SOUT_CH23	PWM_AO_A
GPIOAO_9	REMOTE_OUTPUT	SPDIF_OUT	I2SOUT_CH45	PWM_AO_B
TEST_N	TEST_N	wd_gpio	I2SOUT_CH67	
RESET_N	RESET_N			

Table II.7.8 GPIOCLK_x Multi-Function Pin

Pin Name	Func1	Func2
GPIOCLK_0	CLK24	CLK12
GPIOCLK_1	CLK25	pwm_F

8. Signal Description

Table II.8.1~Table II.8. 17 give the Description of signals.

Table II. 8.1 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
SDIO_IRQ	DIO	SDIO interrupt request signal

Table II.8.2 SDCARD 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 II.8.3 Clock Interface Signal Description

Signal Name	Type	Description
CLK_32K_IN	DI	32KHz clock input
CLK12	DO	12MHz clock output
CLK24	DO	24MHz clock output
CLK25	DO	25MHz clock output

Table II.8.4 UART Interface Signal Description

Signal Name	Type	Description
UART_TX_A	DO	UART Port A data output
UART_RX_A	DI	UART Port A data input
UART_CTS_A	DI	UART Port A Clear To Send Signal
UART_RTS_A	DO	UART Port A Ready To Send Signal
UART_TX_B	DO	UART Port B data output
UART_RX_B	DI	UART Port B data input
UART_CTS_B	DI	UART Port B Clear To Send Signal
UART_RTS_B	DO	UART Port B Ready To Send Signal
UART_TX_C	DO	UART Port C data output
UART_RX_C	DI	UART Port C data input
UART_CTS_C	DI	UART Port C Clear To Send Signal
UART_RTS_C	DO	UART Port C Ready To Send Signal
UART_TX_AO_A	DO	UART Port AO_A data output
UART_RX_AO_A	DI	UART Port AO_A data input
UART_CTS_AO_A	DI	UART Port AO_A Clear To Send Signal
UART_RTS_AO_A	DO	UART Port AO_A Ready To Send Signal
UART_TX_AO_B	DO	UART Port AO_B data output
UART_RX_AO_B	DI	UART Port AO_B data input
UART_CTS_AO_B	DI	UART Port AO_B Clear To Send Signal
UART_RTS_AO_B	DO	UART Port AO_B Ready To Send Signal

Table II.8.5 PCM Interface Signal Description

Signal Name	Type	Description
PCM_OUT_A	DO	PCM port A output data stream
PCM_IN_A	DI	PCM port A input data stream
PCM_FS_A	DO	PCM port A frame synchronization
PCM_CLK_A	DO	PCM port A master clock input

Table II.8.6 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 channel AO_A output signal
PWM_AO_B	DO	PWM channel AO_B output signal
PWM_VS	DO	PWM VSYNC output signal

Table II.8.7 I2C Interface Signal Description

Signal Name	Type	Description
I2C_SDA_A	DIO	I2C bus port A data input/output, Master or Slave
I2C_SCK_A	DIO	I2C bus port A clock input/output, Master or Slave
I2C_SDA_B	DIO	I2C bus port B data input/output, Master or Slave
I2C_SCK_B	DIO	I2C bus port B clock input/output, Master or Slave
I2C_SDA_C	DIO	I2C bus port C data input/output, Master or Slave
I2C_SCK_C	DIO	I2C bus port C clock input/output, Master or Slave
I2C_SDA_AO	DIO	Always-on I2C bus data input/output, Master or Slave
I2C_SCK_AO	DIO	Always-on I2C bus port A clock input/output, Master or Slave

Table II.8.8 I2S Interface Signal Description

Signal Name	Type	Description
I2S_OUT_01	DO	I2S Audio Data Output channel 0 and 1
I2S_OUT_23	DO	I2S Audio Data Output channel 2 and 3
I2S_OUT_45	DO	I2S Audio Data Output channel 4 and 5
I2S_OUT_67	DO	I2S Audio Data Output channel 6 and 7
I2S_LR_CLK_OUT	DO	I2S Left/Right Clock output
I2S_AO_CLK_OUT	DO	I2S data clock output
I2S_IN_CH01	DI	I2S Audio Data Input channel 0 and 1
I2S_AO_CLK_IN	DI	I2S data clock input
I2S_LR_CLK_IN	DI	I2S Left/Right clock input
I2S_AM_CLK	DO	I2S master clock output

Table II.8.9 EMMC/NAND Interface Signal Description

Signal Name	Type	Description
EMMC_NAND_D0	DIO	eMMC/NAND data bus bit 0
EMMC_NAND_D1	DIO	eMMC/NAND data bus bit 1

Signal Name	Type	Description
EMMC_NAND_D2	DIO	eMMC/NAND data bus bit 2
EMMC_NAND_D3	DIO	eMMC/NAND data bus bit 3
EMMC_NAND_D4	DIO	eMMC/NAND data bus bit 4
EMMC_NAND_D5	DIO	eMMC/NAND data bus bit 5
EMMC_NAND_D6	DIO	eMMC/NAND data bus bit 6
EMMC_NAND_D7	DIO	eMMC/NAND data bus bit 7
NAND_CE0	DO	NAND chip enable 0
NAND_CE1	DO	NAND chip enable 1
NAND_CE2	DO	NAND chip enable 2
NAND_CE3	DO	NAND chip enable 3
NAND_RB0	DI	NAND ready/busy
NAND_ALE	DO	NAND address latch enable:
NAND_CLE	DO	NAND command latch enable:
NAND_WEN_CLK	DO	NAND write enable and clock:
NAND_REN_WR	DO	NAND read enable and write/read:
NAND_DQS	DIO	NAND data strobe
EMMC_CLK	DO	eMMC clock output
EMMC_CMD	DIO	eMMC command signal
EMMC_DS	DI	eMMC data strobe

Table II.8.10 NOR Interface Signal Description

Signal Name	Type	Description
NOR_D	DO	SPI NOR Master Output
NOR_Q	DI	SPI NOR Master Input
NOR_C	DO	SPI NOR Serial Clock
NOR_CS	DO	SPI NOR chip select

Table II.8.11 SPI Interface Signal Description

Signal Name	Type	Description
SPI_MOSI	DIO	SPI master output, slave input
SPI_MISO	DIO	SPI master input, slave output
SPI_SCLK	DO	SPI serial clock
SPI_SS0	DO	SPI slave select 0

Table II.8.12 Remote Interface Signal Description

Signal Name	Type	Description
IR_REMOTE_INPUT	DI	IR remote control input
IR_REMOTE_OUTPUT	DO	IR remote control output

Table II.8.13 JTAG Interface Signal Description

Signal Name	Type	Description
JTAG_TDO	DO	JTAG data output
JTAG_TDI	DI	JTAG data input
JTAG_TMS	DI	JTAG Test mode select input
JTAG_TCK	DI	JTAG Test clock input

Table II.8.14 HDMI Interface Signal Description

Signal Name	Type	Description
HDMI_HPD	DI	HDMI hot plug in detection signal input
HDMI_SDA	DIO	HDMI I2C control interface data signal

Signal Name	Type	Description
HDMI_SCL	DO	HDMI I2C control interface clock signal
EE_CEC	DIO	HDMI EE CEC (Consumer electronics control)
AO_CEC	DIO	HDMI AO CEC (Consumer electronics control)

Table II.8.15 Ethernet Interface Signal Description

Signal Name	Type	Description
ETH_LINK_LED	DO	Ethernet link LED indicator
ETH_ACT_LED	DO	Ethernet active LED indicator

Table II.8.16 SPDIF Interface Signal Description

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

Table II.8.17 Digital MIC Interface Signal Description

Signal Name	Type	Description
DMIC_IN	DI	Digital MIC input signal
DMIC_CLK_OUT	DO	Digital MIC clock output signal

9. 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 II.9.1 Absolute Maximum Rating

Characteristic	Value	Unit
VDD_CPU Supply Voltage	1.2	V
VDD_EE Supply Voltage	1.2	V
VDDQ Supply Voltage	1.75	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

10. Recommended Operating Conditions

The table below gives the recommended operating conditions of S905X. Exposure to stresses beyond those listed in this table may result in permanent device damage, unreliability or both.

Table II.10.1 Recommended Operating Conditions

Symbol	Parameter	Min.	Typ.	Max.	Unit
VDDCPU	Voltage for Cortex A53 CPU	0.77[1]	1.0	1.19[2]	V
VDDEE_0V9	Voltage for GPU & core logic	0.77[1]	0.95	1.19[2]	V
VDDAO_0V9	Voltage for AO core logic	0.77[1]	0.95	1.19[2]	V
VDDQ	DDR3/LPDDR2/LPDDR3 IO Supply Voltage	1.15		1.6	V
AVDD0V9	0.9V AVDD for AVDD0V9_ENET, AVDD0V9_HDMITX	0.81	0.9	0.99	V
AVDD18	1.8V AVDD for AVDD18_Audio, AVDD18_CVBS, AVDD18_ENET_EFUSE, AVDD18_HDMITX, AVDD18_HPLL, AVDD18_PLL_ADC, AVDD18_USB, AVDD18_XTAL	1.71	1.80	1.89	V
AVDD33	3.3V AVDD for HDMITX, USB	3.15	3.3	3.45	V
VDDIO	LV mode	1.71	1.80	1.89	V
	HV mode	3.0	3.3	3.45	V
IOVREF_1V8	Voltage for IOVREF	1.71	1.80	1.89	V
T_J	Junction Temperature	0		125[3]	°C
T_A	Operating Temperature	0		70	°C

Note:

1) Minimal VDD_CPU/VDD_EE voltage is for sleep mode while system runs at 32.768KHz. 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 0.8V.
2) Likewise, this maximum VDD_CPU/VDD_EE voltage in actual application should not be higher than 1.1V. Voltage of VDD_CPU will affect CPU speed. Use lower voltage when CPU runs on lower speed to save power.
3) For operating temperature, good heat sink may be needed to guarantee Tj < 125°C.

11. Thermal Operating Specifications

The table below gives the thermal operation specification.

Table II.11.1 Thermal Operation Specification

Symbol	Parameter	Value.	Unit
Θjc	Package junction-to-case thermal resistance in nature convection	TBD	°C/Watt

12. DC Electrical Characteristics

12.1 Normal GPIO Specifications

Table II.12.1 gives the normal GPIO specifications.

Table II.12.1 Normal GPIO Specifications

Symbol	Parameter	Min.	Typ.	Max.	Unit
V_{iH}	High-level input voltage	IOVREF/2+0.3		VDDIO+0.3	V
V_{iL}	Low-level input voltage	-0.3		IOVREF/2-0.3	V
$R_{PU/PD}$	Built-in pull up/down resistor		60K		ohm
DIO_2mA	2mA IO driving capability	2[1]		3[2]	mA
DIO_3mA	3mA IO driving capability	3[1]		4.5[2]	mA
DIO_4mA	4mA IO driving capability	4[1]		6[2]	mA
VOH	Output high level, output IOH=2/3/4 mA per IO type	VDDIO-0.5			V
VOL	Output low level, sink IOL=2/3/4 mA per IO type			0.4	V

Note:

1) Minimal driving capability applies when VDDIO LV 1.71V, or VDDIO HV 3.0V, VOL<0.4V.
2) Maximal driving capability only applies to applications such as driving LED when VOL<0.6V.

12.2 Open Drain GPIO Specifications

Table II.12.2 shows the open drain GPIO Specifications.

Table II.12.2 Open Drain GPIO Specification

Symbol	Parameter	Min.	Typ.	Max.	Unit
V_{iH}	High-level input voltage	2.2		5.5	V
V_{iL}	Low-level input voltage	-0.3		0.8	V
$R_{PU/PD}$	No built-in pull up/down resistor on OD IO		N/A		ohm
Iol	OD IO driving low capability		6		mA

12.3 DDR3/DDR3L/DDR4/LPDDR2/LPDDR3 SDRAM Specifications

Table II.12.3~Table II.12.7 show the recommended operating conditions of DDR/SDRAM, DDR3/DDR3L/DDR3U LPDDR2 and LPDDR3 module respectively.

Table II.12.3 Recommended Operating Conditions of DDR/SDRAM Module

Symbol	Parameter	Min.	Typ.	Max.	Unit
VDDQ	IO supply voltage(DDR3)	1.46	1.50	1.57	V
VDDQ	IO supply voltage(DDR3L)	1.31	1.35	1.45	V
VDDQ	IO supply voltage(DDR3U)	1.21	1.25	1.31	V
VDDQ	IO supply voltage(DDR4)	1.14	1.20	1.26	V

Symbol	Parameter	Min.	Typ.	Max.	Unit
VDDQ	IO supply voltage(LPDDR2/LPDDR3)	1.16	1.20	1.30	V
Vref	Input reference supply voltage	0.49*VDDQ	0.5*VDDQ	0.51*VDDQ	V

Note: The minimal VDDQ voltage in sleep mode is defined by memory.

Table II.12.4 DC specifications - DDR3/DDR3L mode

Symbol	Parameter	Min.	Typ.	Max.	Unit
VIH	DC input voltage high	Vref + 0.100		VDDQ	V
VIL	DC input voltage low	VSSQ		Vref-0.100	V
VOH	DC output logic high	0.8*VDDQ			V
VOL	DC output logic low			0.2*VDDQ	V
RTT	Input termination resistance to VDDQ/2	100 54 36	120 60 40	140 66 44	ohm

Table II.12.5 DC specifications – DDR4 mode

Symbol	Parameter	Min.	Typ.	Max.	Unit
VIH	DC input voltage high	TBD		TBD	V
VIL	DC input voltage low	TBD		TBD	V
VOH	DC output logic high	TBD		TBD	V
VOL	DC output logic low			TBD	V
RTT	Input termination resistance to VDDQ/2	TBD	TBD	TBD	ohm

Table II.12.6 DC Specifications – LPDDR2 mode

Symbol	Parameter	Min.	Typ.	Max.	Unit
VIH	DC input voltage high	Vref + 0.13		VDDQ	V
VIL	DC input voltage low	VSSQ		Vref-0.13	V
VOH	DC output logic high	0.9*VDDQ			V
VOL	DC output logic low			0.1*VDDQ	V

Table II.12.7 DC Specifications – LPDDR3 mode

Symbol	Parameter	Min.	Typ.	Max.	Unit
VIH	DC input voltage high	Vref + 0.100		VDDQ	V
VIL	DC input voltage low	VSSQ		Vref-0.100	V
VOH	DC output logic high	0.9*VDDQ			V
VOL	DC output logic low			0.1*VDDQ	V
RTT	Input termination resistance to VDDQ	100 200	120 240	140 280	ohm

12.4 Recommended Oscillator Electrical Characteristics

S905X requires the 24MHz oscillator for generating the main clock source.

Table II.12.8 24MHz Oscillator Specification

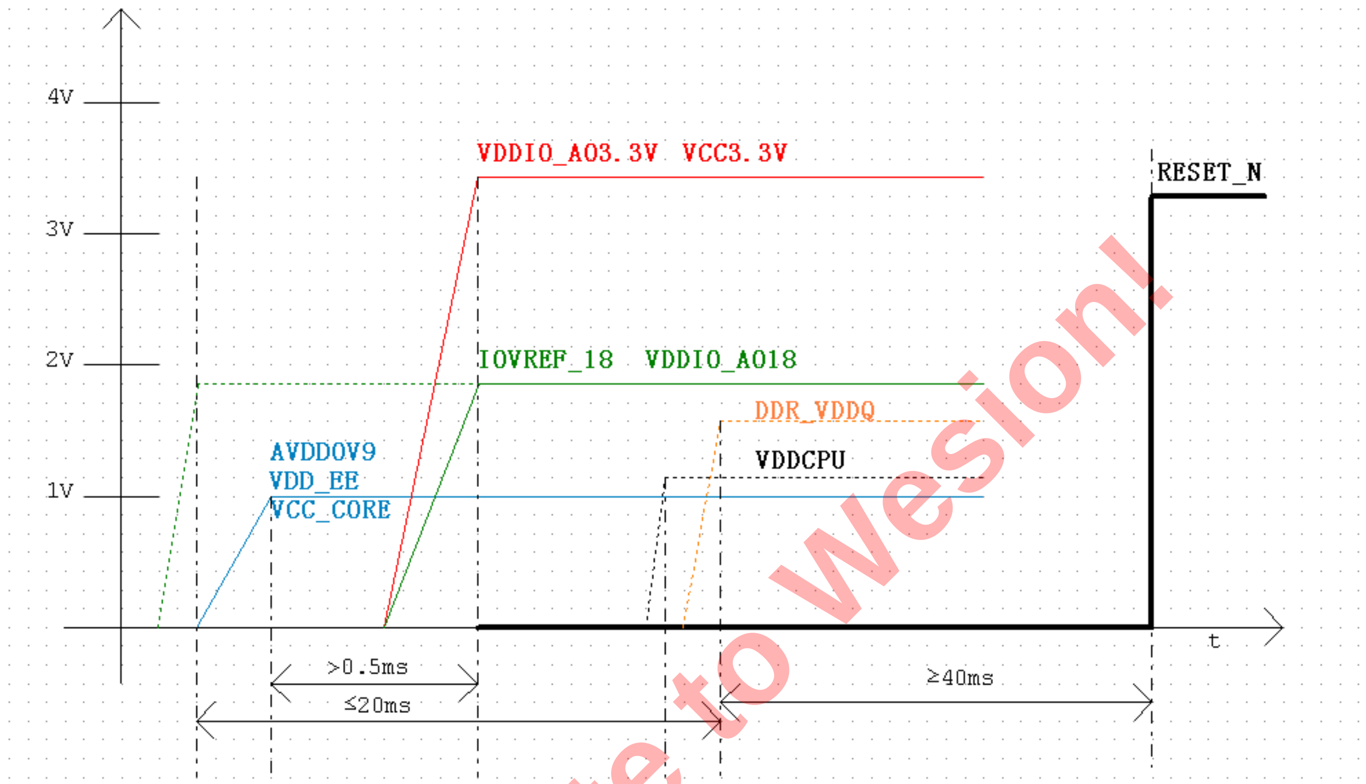
Symbol	Description	Min.	Typ.	Max.	Unit	Notes
F_o	Nominal Frequency		24		MHz	
$\Delta f/f_o$	Frequency Tolerance	-30		+30	ppm	At 25 ℃
		-50		+50	ppm	At -20~85 ℃
C_L	Load Capacitance	8	12	20	pF	
ESR	Equivalent Series Resistance			80	oHm	

Note: 10ppm Tolerance is preferred if 24MHz XTAL is also driving WIFI module.

13. Recommended Power on sequence

Example power on sequence is shown in Fig II.13.1:

VDD_EE->VDDIO_AO3.3V&VCC3.3V->VDDCPU&DDR_VDDQ



Note:

1) VDDIO_AO3.3V&VCC3.3V should ramp up > 0.5ms later than VDD_EE
2) All power sources should get stable within 20ms(except for DDR_VDDQ)
3) No sequence requirement between IOVREF_18 and VDD_EE
   No sequence requirement between VDDCPU & DDR_VDDQ and other power source
4) IOVREF_18 should ramps up earlier or at the same time with VDDIO_AO3.3V&VCC3.3V, VDDIO_AO3.3V&VCC3.3V should never be 2.5V higher than IOVER_18.
5) In some ref designs, VDDCPU & VDDEE are merged to VCC_CORE, the power on sequence should be same as VDD_EE.
6) RESET_n should keep low for at least 40 ms after power up(except DDR_VDDQ)

Please refer to reference schematics.

14. Power Consumption

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

Table II. 14.1 System Power Consumption

Symbol	Maximum Current	Note
VDDCPU	2A	
VDDEE_0V9	2A	
VDDQ	1A	Total power of controller and DDR3 memory

Table II.14.2 Module Power Consumption

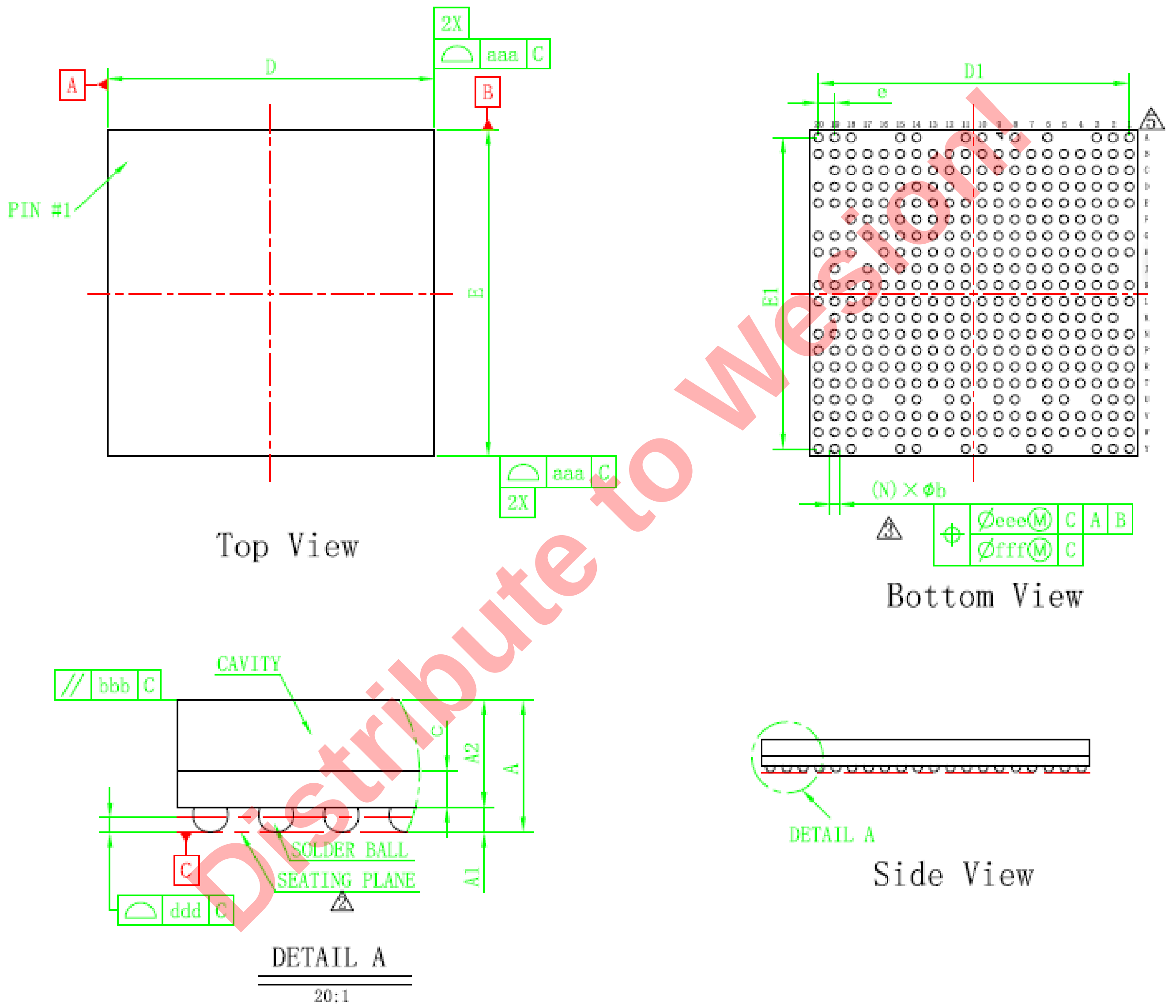
Symbol	Typical current	Maximum current	Note
HDMITX_AVDD33	TBD	TBD	At 6Gbps mode
AVDD33_USB	TBD	TBD	USB: 16mA Per channel, full/low speed mode, total 2 channel VDDIOH: normally 0mA, will be higher if there's pull low loading.
AVDD18_Audio	TBD	TBD	
AVDD18_CVBS	TBD	TBD	38mA typically when CVBS output enabled.
AVDD18_ENET_EFUSE	TBD	TBD	EFUSE: Max 100mA when programing EFUSE. Normally 0mA
AVDD18_HDMITX	TBD	TBD	At 6Gbps mode
AVDD18_HPLL	TBD	TBD	
AVDD18_PLL_ADC	TBD	TBD	ADC: 1.2mA During sampling
AVDD18_USB	TBD	TBD	USB: 19mA per channel at high speed mode , total 2 channel
AVDD18_XTAL	TBD	TBD	Typical value 0.4mA when XTAL is oscillating
AVDD0V9_ENET	TBD	TBD	
AVDD0V9_HDMITX	TBD	TBD	

15. Mechanical Dimension

The S905X processor comes in a 20x20 ball matrix LFBGA RoHS package. The mechanical dimensions are given in millimeters as below.

Note the following positions have null balls: A4, A5, A7, A9, A12, A13, A16, A17, C20, D11, F1, F19, F20, H17, J1, J18, J20, M1, M20, U4, U7, U10, U13, U16, Y4, Y5, Y8, Y9, Y12, Y13, Y16, Y17.

Fig II.15.1 Mechanical Dimension of S905X



symbol	Dimension in mm			Dimension in inch		
	MIN	NOM	MAX	MIN	NOM	MAX
A	—	—	1.410	—	—	0.056
A1	0.200	0.250	0.300	0.008	0.010	0.012
A2	1.010	1.060	1.110	0.040	0.042	0.044
c	0.320	0.360	0.400	0.013	0.014	0.016
D	12.900	13.000	13.100	0.508	0.512	0.516
E	12.900	13.000	13.100	0.508	0.512	0.516
D1	—	12.350	—	—	0.486	—
E1	—	12.350	—	—	0.486	—
e	—	0.650	—	—	0.026	—
b	0.300	0.350	0.400	0.012	0.014	0.016
aaa	0.100			0.004		
bbb	0.100			0.004		
ddd	0.100			0.004		
eee	0.150			0.006		
fff	0.080			0.003		
Ball Diam	0.350			0.014		
N	368			368		
MD/ME	20/20			20/20		

Section III System

This part describes the S905X system architecture from the following aspects:

- MEMORY MAP
- POWER DOMAIN
- CPU and GPU SUBSYSTEM
- CLOCK AND RESET UNIT
- SYSTEM BOOT
- GENERAL PURPOSE INPUT/OUTPUT (GPIO)
- INTERRUPT CONTROLLER
- DIRECT MEMORY ACCESS CONTROLLER (DMAC)
- TIMER
- Crypto

16. Memory Map

S905X's memory map is listed as following in Table III.16.1.

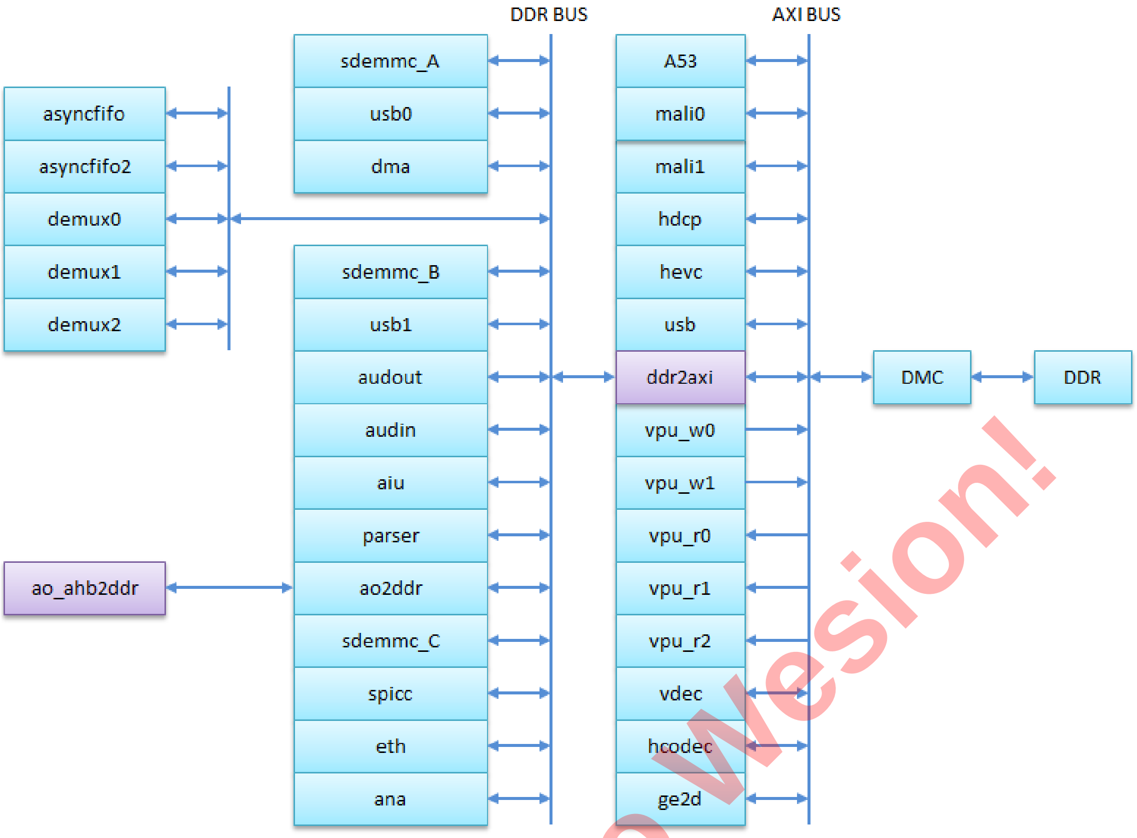
Table III.16.1 S905X Memory Map

Start	End	Region	CCI/NIC Arbitor
0x00000000	0xBFFFFFFF	DDR	**DDR**
0xC0000000	0xC01FFFFF	USB	**CAPB3**
0xC0200000	0xC03FFFFF	DAP	
0xC0400000	0xC07FFFFF	CSSYS	
0xC0800000	0xC08043FF		
0xC0804400	0xC08044FF	RESET	
0XC0804500	0xC08047FF		
0xC0804800	0xC08048FF	VDIN	
0xC0804900	0xC08053FF		
0xC0805400	0xC08054FF	AIU	
0xC0805500	0xC08057FF		
0xC0805800	0xC08058FF	STB	
0xC0805900	0xC0807BFF		
0xC0807C00	0xC0807CFF	ASSIST	
0xC0807D00	0xC08083FF		
0xC0808400	0xC08084FF	PERIPHS	
0xC0808500	0xC0808BFF		
0xC0808C00	0xC0808CFF	PERIPHS	
0xC0808D00	0xC08097FF		
0xC0809800	0xC08098FF	ISA	
0xC0809900	0xC080BFFF		
0xC080A000	0xC080A0FF	AUDIN	
0xC080A100	0xC080A3FF		
0xC080A400	0xC080A4FF	PARSER	
0xC080A500	0xC12FFFFF		
0xC1300000	0xC13FFFFF	GPV	**GPV**
0xC1400000	0xC42FFFFF		
0xC4300000	0xC4307FFF	GIC	**GIC**
0xC4308FFF	0xC7FFFFFF		
0xC8000000	0xC8013FFF	AHB SRAM	**SEC AHB**
0xC8014000	0xC80FFFFF		
0xC8100000	0xC81FFFFF	RTI	
0xC8200000	0xC881FFFF		
0xC8820000	0xC882FFFF	DOS	
0xC8832000	0xC8833FFF	ACODEC	
0xC8834000	0xC8835FFF	PERIPHS	
0xC8836000	0xC8837FFF	DDR TOP	
0xC8838000	0xC8839FFF	DMC	
0xC883A000	0xC883BFFF	HDMITX	
0xC883C000	0xC883DFFF	HIU	
0xC883E000	0xC8FFFFFFF		
0xC9000000	0xC90FFFFF	USB0	
0xC9100000	0xC91FFFFF	USB1	
0xC9200000	0xC940FFFF		
0xC9410000	0xC941FFFF	ETHERNET	
0xCC000000	0xCFFFFFFF	SPI	
0xD0000000	0xD0041FFF		**CAPB3**

Start	End	Region	CCI/NIC Arbitor
0xD0042000	0xD0043FFF	PDM	
0xD0044000	0xD00440FF	HDCP22	
0xD0044100	0xD0047FFF		
0xD0048000	0xD004FFFF	BT656	
0xD0050000	0xD005FFFF		
0xD0060000	0xD006FFFF		
0xD0070000	0xD0071FFF	SD_EMMC_A	
0xD0072000	0xD0073FFF	SD_EMMC_B	
0xD0074000	0xD0075FFF	SD_EMMC_C	
0xD0076000	0xD00BFFFF		
0xD00C0000	0xD00FFFFF	MALI APB	
0xD0100000	0xD013FFFF	VPU	**VAPB3**
0xD0150000	0xD015FFFF		
0xD0160000	0xD016FFFF	GE2D	
0xD0170000	0xD01FFFFF		
0xD0200000	0xD8FFFFFF		
0xD9000000	0xD9013FFF	AHB SRAM	**SEC AHB**
0xD9014000	0xD903FFFF		
0xD9040000	0xD907FFFF	BOOT ROM	
0xD9080000	0xDA0FFFFF		
0xDA100000	0xDA1FFFFF	RTI	
0xDA200000	0xDA81FFFF		
0xDA820000	0xDA82FFFF	DOS	
0xDA830000	0xDA831FFF	EFUSE	
0xDA832000	0xDA833FFF	ACODEC	
0xDA834000	0xDA835FFF	PERIPHS	
0xDA836000	0xDA837FFF	DDR TOP	
0xDA838000	0xDA839FFF	DMC	
0xDA83A000	0xDA83BFFF	HDMITX	
0xDA83C000	0xDA83DFFF	HIU	
0xDA83E000	0xEFFFFFFF		

Below is the DDR bus.

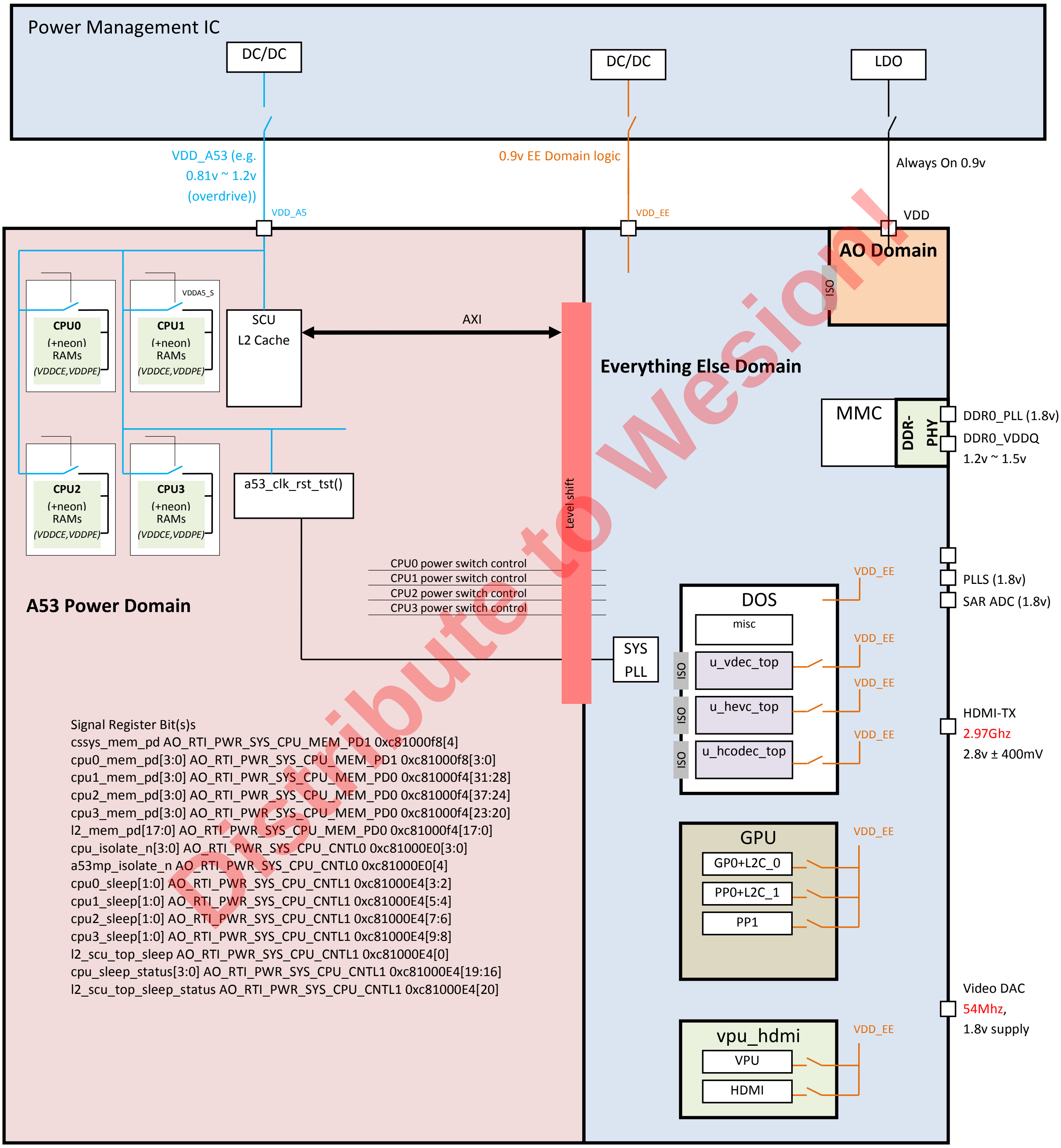
Fig III.16.1 S905X DDR Bus


DDR BUS
AXI BUS
asyncfifo
asyncfifo2
demux0
demux1
demux2
sdemmc_A
usb0
dma
sdemmc_B
usb1
audout
audin
aiu
parser
ao2ddr
sdemmc_C
spicc
eth
ana
ao_ahb2ddr
A53
mali0
mali1
hdcp
hevc
usb
ddr2axi
vpu_w0
vpu_w1
vpu_r0
vpu_r1
vpu_r2
vdec
hcodec
ge2d
DMC
DDR

17. Power Domain

Fig III.17.1 shows the power domain of S905X.

Fig III.17.1 Power Domain



17.1 Top Level Power Domains

The power supplies for the different domains must follow a specific power supply order: The A53 can't be powered without the EE domain. The EE domain can't be powered up without the AO domain. If you read the Table III.17.1 left to right then right to left, that's essentially the power up/down sequence for the entire chip.

Table III.17.1 Power on Sequence of Different Power Domains

	Always On	**EE Domain**		**A53 domain**				
	Logic	EE Logic	Mali and DOS and VPU	A53 (SCU/L2)	L2 Cache	CPU0	CPU1	CPU2/3
STATE 0 All off	Off	The EE domain must be OFF if the AO domain is off		The A53 domain must be off if the EE domain is OFF				
STATE 1 Hibernate: Only the Always on Domain is powered	On	Off	The Mali must be OFF is if the EE domain is off	The A53 domain must be off if the EE domain is OFF				
STATE 2 Always On/EE only (for example audio applications or simple video applications that don't need the A53)	On	On	On or off as needed	Off	Off	Off	Off	Off
STATE 3 Single CPU	On	On	On or off as needed	All three must be enabled			Off	Off
STATE 4 2 CPU's	On	On	On or off as needed	All three must be enabled			On	Off
STATE 5 4 CPUs	On	On	On or off as needed	All three must be enabled			On	On

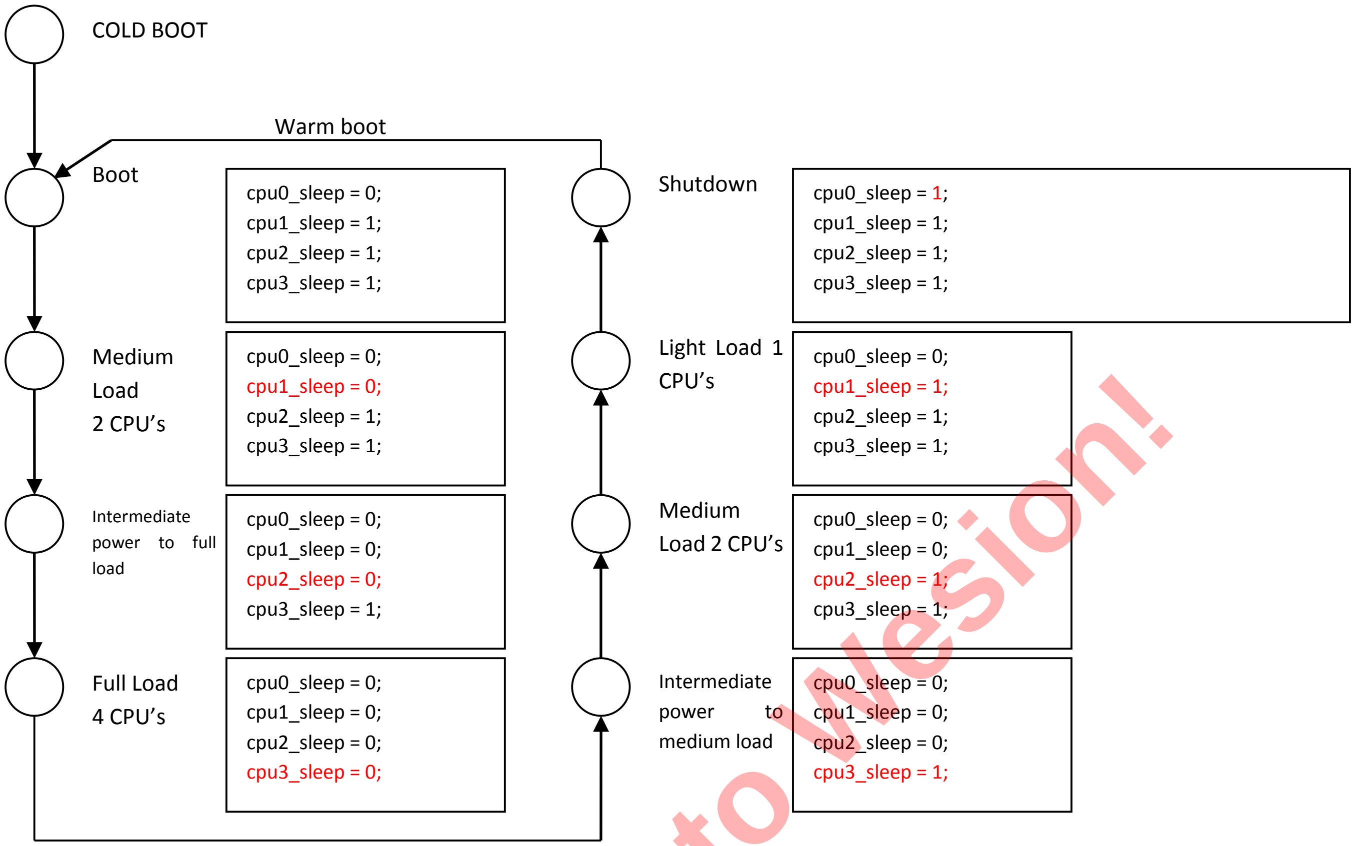
17.2 A53 Power Modes

The A53 domain is the last to power up and the first to power down. The A53 domain itself consists of a quad (4) CPU, an L2 cache controller and an SCU. The A53 CPU boots with the SCU/L2 powered and CPU0 powered. After CPU0 boots, subsequent CPU's can be enabled and disabled independently of one another using the control bits described below. The most likely scenario would be for the CPU0 to be used for low load conditions, CPU0 and CPU1 to be used for medium load conditions and CPU0,1,2 and CPU3 to all be used for heavy load conditions. It's unlikely we will run a 3 CPU configuration (although it's not precluded by the hardware). The flow-diagram below illustrates the transition to each legal state.

Table III.17.2 Power On Sequence of A53

	CPU0	**CPU1**	**CPU2**	**CPU3**
Domain Power Sleep bit (1 = power off)	Bit[2] of 0xC81000E4	Bit[4] of 0xC81000E4	Bit[6] of 0xC81000E4	Bit[8] of 0xC81000E4
Domain Power Acknowledge bit	Bit[16] of 0xC81000E4	Bit[17] of 0xC81000E4	Bit[18] of 0xC81000E4	Bit[19] of 0xC81000E4
Input Signal isolation bit (1 = isolated)	Bit[0] of 0xC81000E0	Bit[1] of 0xC81000E0	Bit[2] of 0xC81000E0	Bit[3] of 0xC81000E0
Output signal isolation bit (1 = isolated)	Bit[0] of 0xC81000E0	Bit[1] of 0xC81000E0	Bit[2] of 0xC81000E0	Bit[3] of 0xC81000E0

Fig III.17.2 Power Seuquence of A53



17.3 EE Top Level Power Modes

EE domain is powered off by DC-DC. VDD_EE also share the same VDD with VDDCE of A53 memory arrays. So if EE domains is shut off, A53 memory is also shut off. That does not matter. Before EE power domain is shut off, A53 should be shut off at first.

Table III.17.3 Power Sequence of EE Domain

	Register/Bit(s)
Ethernet Memory PD	Bits[3:2] of 0xC883C100 (HI_MEM_PD_REG0) 0x3 = power off, 0x0 = normal operation
HDMI Memory PD	Bits[15:8] of 0xC883C100 (HI_MEM_PD_REG0) 0xFF = power off, 0x0 = normal operation
VPU memory PD	Bit[31:0] of 0xC883C104/ 0xC883C108 0xFFFFFFFF = power off, 0x0 = normal operation

17.4 Mali Power Modes

The Mali block sits within the EE domain and the Mali module itself has 3 distinct power domains:

- GP + L2C_0
- PP0 + L2C_1
- PP12

We will power these up in sequence so as to reduce the surge currents. The following combinations are possible.

Table II.17.4 Power Sequence of Mali Module

Mali AXI Cores	1PP Power up/down	3PP Power up/down
GP + L2C_0	1st / last	1st / last
PP0 + L2C_1	2nd / 1st	2nd / 2nd
PP12		3rd / 1st

17.5 Power/Isolation/Memory Power Down Register Summary

Below lists the registers related to A53.

A53 Isolation 0xc81000e0

Bit(s)	R/W	Default	Description
31~30	R	-	CPU3 CTRL_MODE set by the CPU
29~28	R	-	CPU3 CTRL_MODE set by the CPU
27~26	R	-	CPU3 CTRL_MODE set by the CPU
25~24	R	-	CPU3 CTRL_MODE set by the CPU
23~22	R/W	11	CPU3 CTRL_MODE
21~20	R/W	11	CPU2 CTRL_MODE
19~18	R/W	11	CPU1 CTRL_MODE
17~16	R/W	00	CPU0 CTRL_MODE (set to tell the CPU which mode to enter)
15~14	R/W	0	Reserved
13	R/W	0	A53 Pwr top level clamp
12	R/W	0	SCURAM Clamp
11~8	R/W	0xE	NEON[3:0] Clamp
7~4	R/W	0xE	CPURAM[3:0] clamp
3~0	R/W	0xE	CPU[3:0] clamp

A53 Power 0xc81000e4

Bit(s)	R/W	Default	Description
31~20	R	0	Reserved
19	R	-	CPU3 Sleep status: 1 = powered down
18	R	-	CPU2 Sleep status: 1 = powered down
17	R	-	CPU1 Sleep status: 1 = powered down
16	R	-	CPU0 Sleep status: 1 = powered down
15~10	R/W	00	Reserved
9~8	R/W	11	CPU3 sleep: 1 = powered down
7~6	R/W	11	CPU2 sleep: 1 = powered down
5~4	R/W	11	CPU1 sleep: 1 = powered down
3~2	R/W	00	CPU0 sleep: 1 = powered down
1-0	R/W	10	Reserved

A53 RAM Power Down Control 0xc81000f4

Bit(s)	R/W	Default	Description
31-28	R/W	0xF	CPU1 RAM power down (Each Bit controls different RAMs): 1 = powered down
27-24	R/W	0xF	CPU2 RAM power down (Each Bit controls different RAMs): 1 = powered down
23-20	R/W	0xF	CPU3 RAM power down (Each Bit controls different RAMs): 1 = powered down
19-18	-	0	Reserved
17~0	R/W	0x00000	L2 RAM power down (Each Bit controls different RAMs): 1 = powered down

A53 RAM Power Down Control (cont.) 0xc81000f8

Bit(s)	R/W	Default	Description
31-4	-	0	Reserved
3~0	R/W	0x0	CPU0 RAM power down (Each Bit controls different RAMs): 1 = powered down

A53 SYS PLL 0xc1104300

PLL Enable = Bit[30]: 1 = pll enabled, 0 = pll disabled

Always On domain PWR_CNTL1 0xc810000c

Bit(s)	R/W	Default	Description
31~4	R/W	0	Reserved
3	R/W	0	DDR1PHYIO_RET_EN
2	R/W	0	DDR1PHYIO_REG_EN_N
1	R/W	0	DDR0PHYIO_RET_EN
0	R/W	0	DDR0PHYIO_REG_EN_N

Always On domain PWR_CNTL0 0xc8100010

Bit(s)	R/W	Default	Description
31~28	R/W	0	Reserved
27~22	R/W	0	reserved
21~20	R/W	0	AHB SRAM Memory power down: 00 = normal operation, 10 = power down. Other conditions reserved
19~16	R/W	0	Reserved
15~14	R/W	0	Reserved
13~12	R/W	0	32khz DS:
11~10	R/W	0	Alternate 32khz input clock select from GPIO pad
9	R/W	1	Reset to the EE domain. 0 = reset, 1 = normal operation
8	R/W	0	RTC oscillator input select: 1 = use RTC clock as clock. 0= used clk81 as the clock
7~5	R/W	0	reserved
4	R/W	0	EE domain isolation In
3	R/W	0	EE domain Isolation out
2	R/W	0	Reserved
1	R/W	0	Always on RESET isolation: 1 = isolated
0	R/W	0	Reserved

General Power gen_pwr_sleep_cntl0 0xc81000e8

Bit(s)	R/W	Default	Description
31~10	R/W	0	Reserved
9	R/W	0	VPU/HDMI Isolation
8	R/W	0	VPU/HDMI power: 1 = powered off
7~6	R/W	0	Reserved
5	R/W	1	Reserved
4	R/W	1	Reserved
3	R/W	1	Reserved
2	R/W	1	Reserved
1	R/W	1	Reserved
0	R/W	1	Reserved

General Isolation gen_pwr_iso_cntl0 0xc81000ec

Bit(s)	Description
31~10	Reserved
9	Reserved
8	Reserved
7	Reserved
6	Reserved
5	Reserved
4	Reserved
3	GPU ISO OUT
2	GPU ISO IN
1~0	reserved

General Acknowledge gen_pwr_sleep_ack0 0xc81000f0

Bit(s)	Description
31~9	Reserved
8	Reserved
6	MALI_PWRUP_ACK: PP12 acknowledge: 1 = powered off
5	MALI_PWRUP_ACK: PP0 acknowledge: 1 = powered off
4	MALI_PWRUP_ACK: GP acknowledge: 1 = powered off
3	Reserved
2	Reserved
1	Reserved
0	Reserved

Mali Power UP (domains_npwr_up) 0xd00c2000

Bit(s)	Description
31~3	Reserved
2	MALI PP12 power up
1	MALI PP0 power up
0	MALI GP power up

Mali Power Down (domains_npwr_up) 0xd00c2004

Bit(s)	Description
31~3	Reserved
2	MALI PP12 power down
1	MALI PP0 power down
0	MALI GP power down

18. System Booting

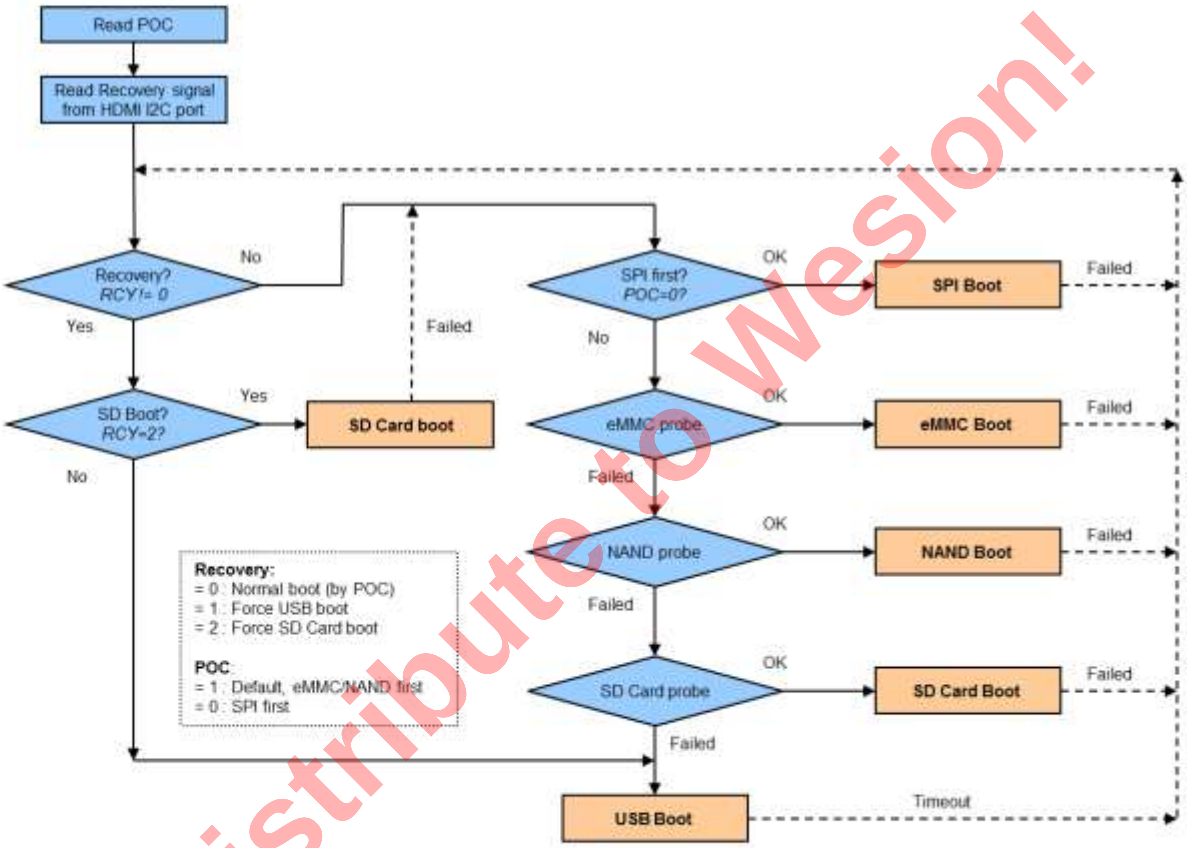
18.1 Overview

The part describes the power–on mode configuration of S905X, which include two portions: Cortex-M3 for security control and A53 for others.

18.2 Power-on Flow Chart

Fig III.18.1 illustrates S905X's power on sequence.

Fig III.18.1 Power-on Flow Chart



19. CPU

19.1 Overview

The Cortex™-A53 MP subsystem of the chip is a high-performance, low-power, ARM macrocell 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 runs32-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:

- in-order pipeline 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 Macrocell (ETMv4) interface
- Intelligent Energy Manager (IEM) support with
  - asynchronous AXI wrappers
  - two voltage domains
- Media Processing Engine (MPE) with NEON technology
- Supports FPU
- Supports Hardware Virtualization

20. GPU

The Mali-450 MP GPU is a hardware accelerator for 2D and 3D graphics system which compatible with the following graphics standards: OpenGL ES 2.0, OpenGL ES 1.1, OpenVG 1.1, EGL 1.5. The developers can follow the ARM and Khronos official reference documents for programming details.

21. Clock and Reset

21.1 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. S905X uses an external 25 24MHz crystal; there are 5 PLLs: MPLL, SYS_PLL, HDMI_PLL, GP0_PLL and DDR_PLL, these 5 PLLs can generate 18 clock sources, as shown in the following table.

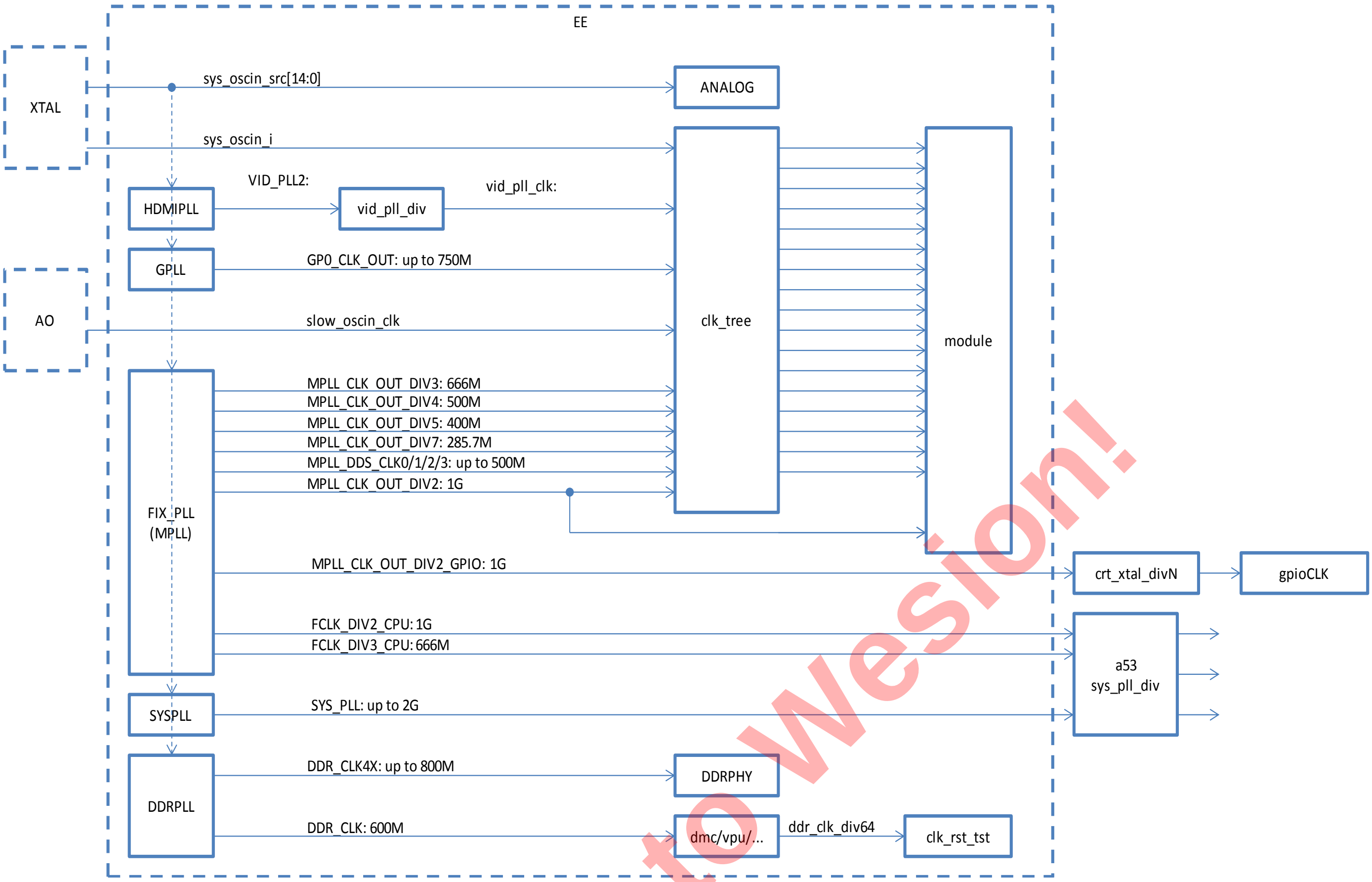
Table III.21.1 S905X PLLs

	OUTPUT CLOCK	MAX FREQ	INTERNAL DIV	SS mode	COMMENT
SYS_PLL	SYS_CLK_OUT	2G	/1 /2 /4		CPU source
GP0	GP0_CLK_OUT	3G	/1 /2 /4	yes	
MPLL	FCLK_DIV2_CPU	1G			CPU source
	FCLK_DIV3_CPU	666M			CPU source
	MPLL_CLK_OUT_DIV2_GPIO	1G			to GPIOCLK
	MPLL_CLK_OUT_DIV2	1G			
	MPLL_CLK_OUT_DIV3	666M			
	MPLL_CLK_OUT_DIV4	500M			
	MPLL_CLK_OUT_DIV5	400M			
	MPLL_CLK_OUT_DIV7	285.7M			
	MPLL_DDS_CLK0	500M	Up to /32	yes	
	MPLL_DDS_CLK1	500M	Up to /32		
	MPLL_DDS_CLK2	500M	Up to /32		
	MPLL_DDS_CLK3	500M	Up to /32	yes	
HDMI_PLL	HDMI_CLK_OUT	6G	/1/2/4/8/16	yes	
	HDMI_CLK_OUT2	6G	/1/2/4/8/16/32/64	yes	
DDR_PLL	DDR_CLK4X_OUT	3G	/1 /2 /4	yes	DDR
	DDR_CLK_OUT	3G	/1/2/4/8/16/	yes	DDR

21.2 Clock Trees

Fig III. 21.1 shows the clock connections of S905X. In this part, we will discuss A53 clock tree, AO clock tree, HDMI clock tree and EE clock tree in detail.

Fig III.21.1 Clock Connections

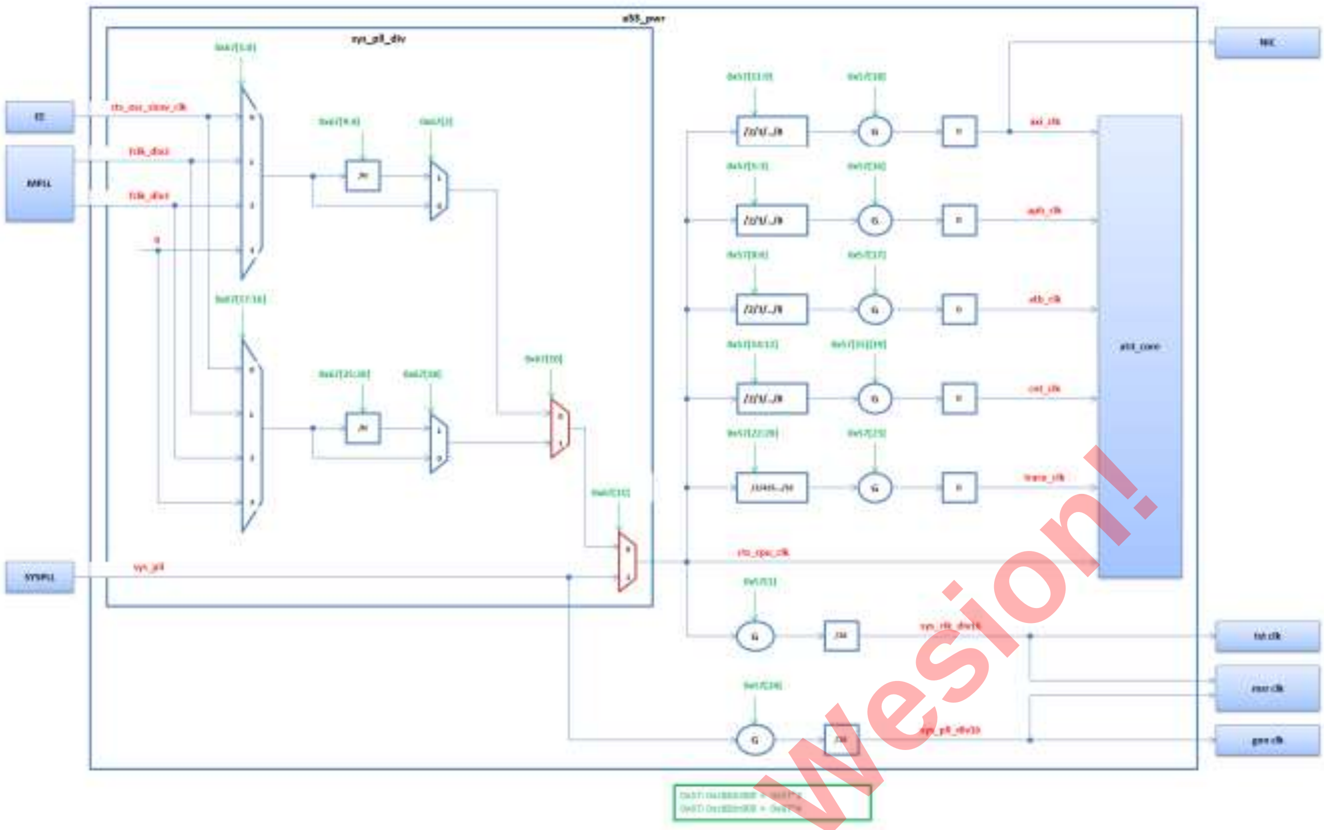


A53 Clock Tree

A53 has 4 clock source, as shown in the following figure, among which,

1) cts_osc_slow_clk is for low power and debug,
2) fclk_div2 and fclk_div3 are for frequencies lower than 1G.
3) sys_pll is for frequencies higher than 1G.

Fig III.21. 2Mutil Phase PLLS of A53



To avoid glitch when change frequencies, there are 2 specially designed dynamic muxes, labeled by red in the above figure. When frequencies changes, the dynamic muxes will first stop the first frequency, then start the second so there will be no mixing of 2 different frequencies thus generate no frequency glitch.

It is possible to do the following switch without glitch:

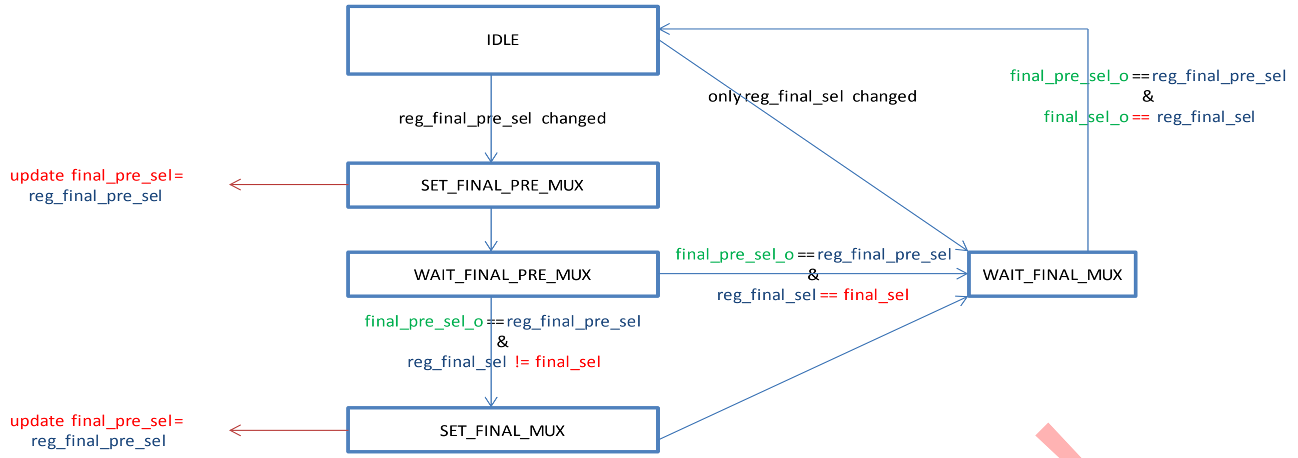
1). Between any 2 frequencies lower than 1GHz;

2). From a frequency lower than 1GHz to a frequency higher than 1GHz;

If the user want to switch between 2 frequencies both higher than 1GHz, it is strongly recommended to change to frequencies lower than 1GHz first.

The diagram of specially designed dynamic mux is shown in the following diagram:

Fig III.21.4 Diagram of Dynamic Mux

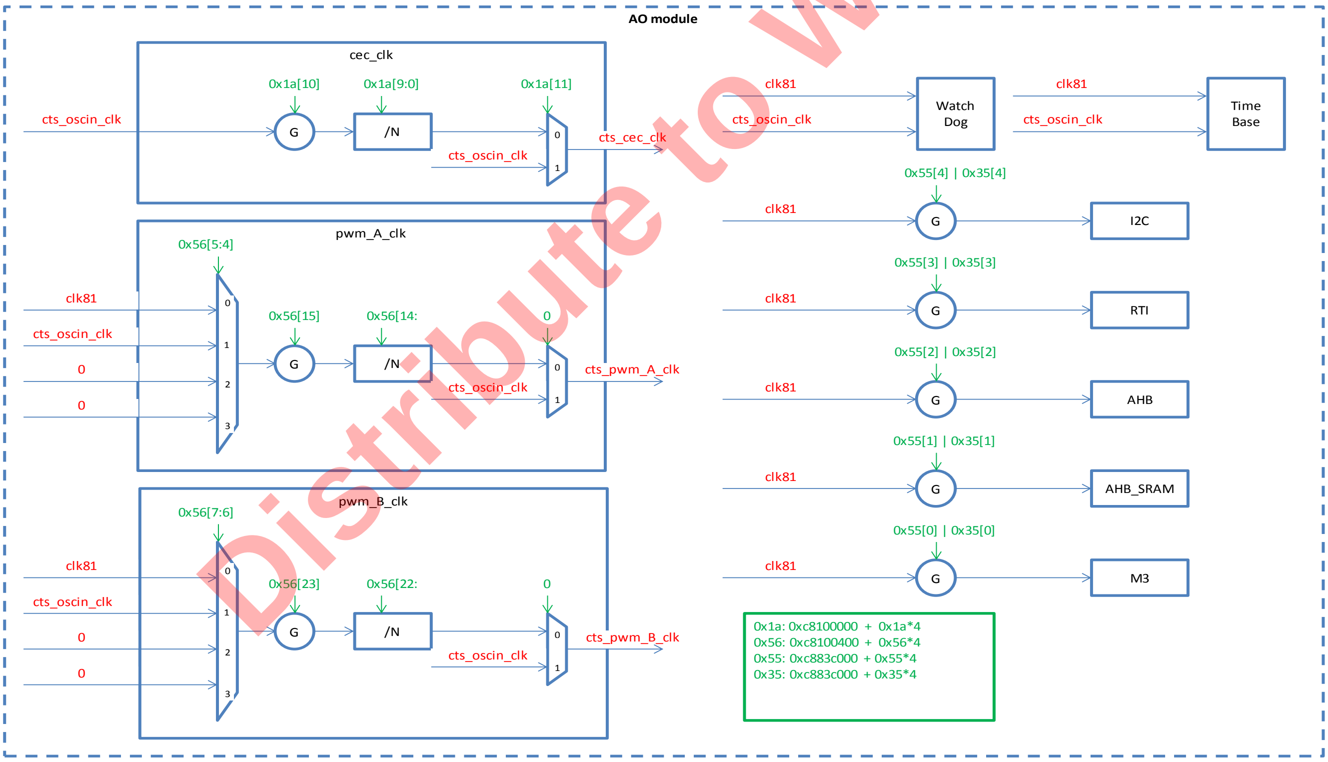


Note, when use these 2 dynamic muxes, both control bits of these muxes have to be configured at the same time, otherwise they will not function correctly.

AO Clock Tree

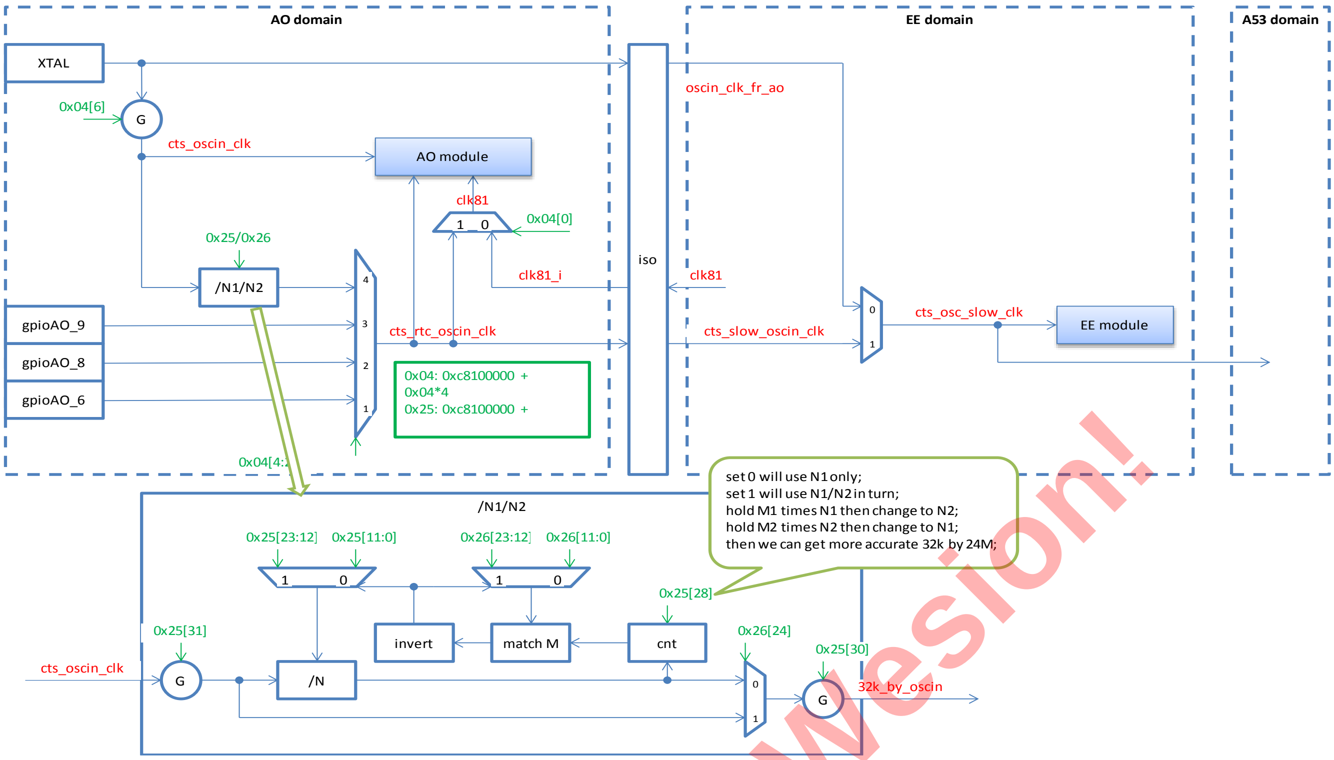
The following figure shows the clock source of AO modules:

Fig III.21.5 AO Clock Sources



For low power mode, 32KHz clock is needed, and it is generated in AO domain as shown in the following diagram:

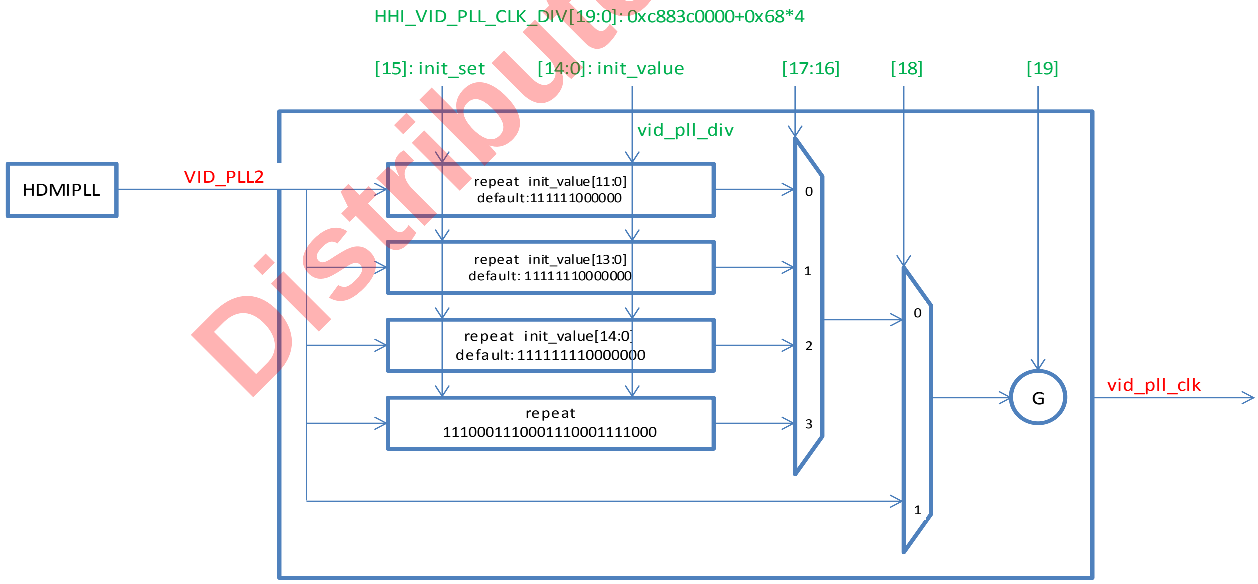
Fig III.21.6 How to generate 32KHz Clock



HDMI Clock Tree

The HDMI_PLL goes through vid_pll_div to generate new clock. The HDMI clock tree is shown in the following figure:

Fig III.21.7 HDMI Clock Tree



EE Clock Tree

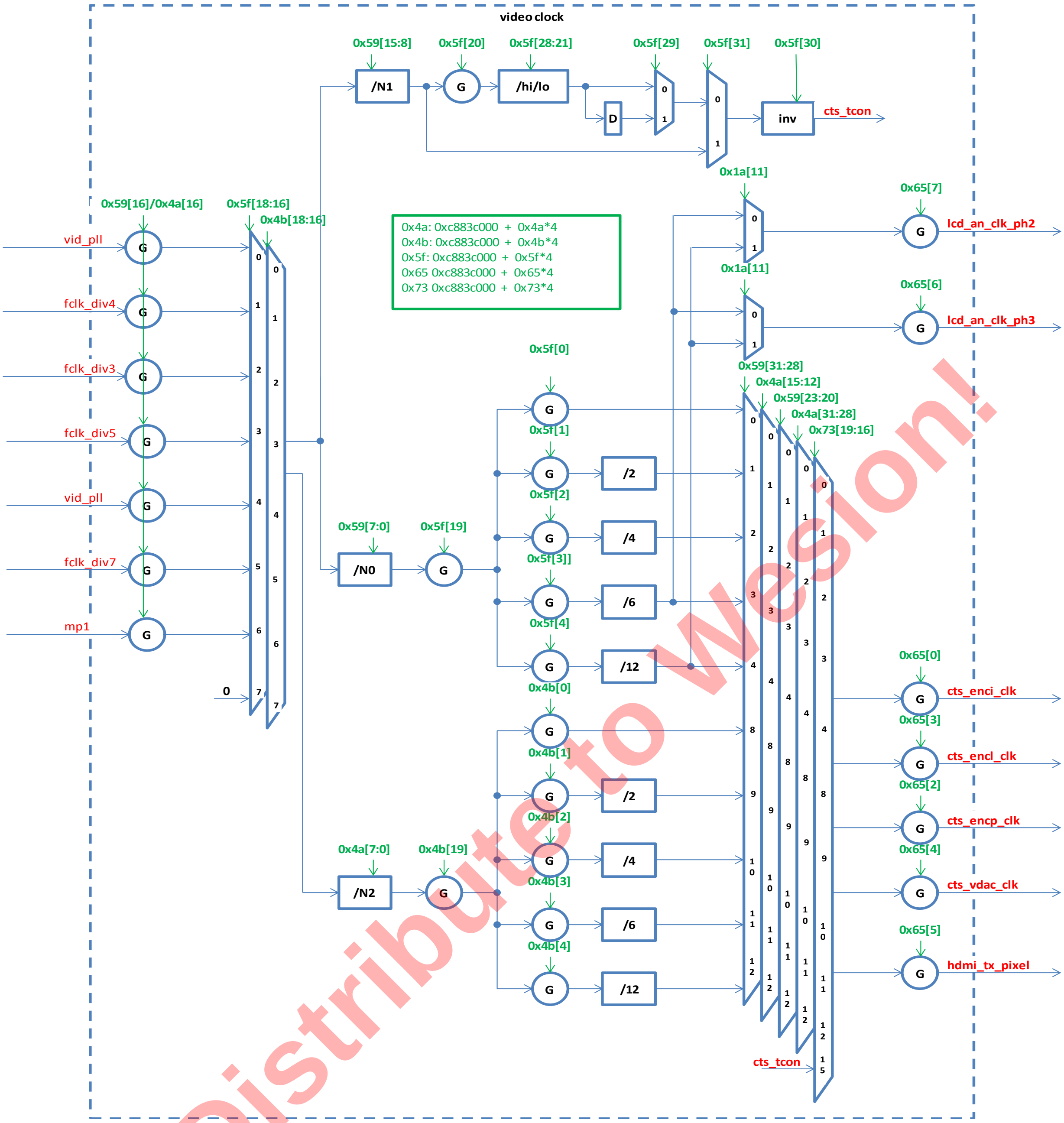
All EE clocks except video clock are listed in the following table:

Table III.21.2 EE Clock

	name	destination	gate	N	sel	src0	src1	src2	src3	src4	src5	src6	src7	src8
	cts slow oscin clk	a53			0x5d[9]	xtal	cts slow oscin							
	amclk o int	aiu		0x1516[11]		cts amclk								
if 0x1519[5:0]=0, div = 0x1516[3:2]	cts aoclk int	aiu	0x1516[0]	0x1519[5:0]		amclk o int								
if 0x1519[5:0]=0, div = 0x1516[3:2]	cts aoclkx2 int	aiu	0x1516[0]	0x1519[5:1]		amclk o int								
x64[27] = 0, will use the same as cts ar	cts mclk i958	aiu	0x64[24]	0x64[23:16]	0x64[26:25]		mp0	mp1	mp2					
if 0x1516[12] 0, will div 1/2/3/4; if 0x1516[12] 1, will div 2/4/6/8;	cts_clk_i958	aiu	0x1516[1]	0x1516[5:4]		cts_mclk_i958								
	cts pcm mclk	aiu	0x96[9]	0x96[8:0]	0x96[11:10]	mp0	mp1	mp2	fclk div5					
	cts amclk	aiu/audin	0x5e[8]	0x5e[7:0]	0x5e[10:9]		mp0	mp1	mp2					
	cts pcm sclk	aiu/audin	0x96[22]	0x96[21:16]		cts pcm mclk								
div = 0x5d[31]?0x5d[30:16]:0x5d[6:0]	mpeg pll clk	aiu	0x5d[7]	0x5d[31:0]	0x5d[14:12]	cts slow oscin clk		fclk div7	mp1	mp2	fclk div4	fclk div3	fclk div5	
	clk81 ee o	ao			0x5d[8]	cts slow oscin clk	mpeg pll clk							
	oscin clk to ao	ao				xtal								
	cts pdm	audin	0x69[16]	0x69[15:0]	0x69[18:17]	cts amclk	mp0	mp1	mp2					
	cts bt656 clk0	bt656	0xf5[7]	0xf5[6:0]	0xf5[10:9]	fclk div2	fclk div3	fclk div5	fclk div7					
if 0x7a[31] = 0, use 0x78; if 0x7a[31] = 1, use 0x7a;	cts_hcodec_clk	dos	0x78[24] 0x7a[24]	0x78[22:16] 0x7a[22:16]	0x78[27:25] 0x7a[27:25]	fclk_div4	fclk_div3	fclk_div5	fclk_div7	mp1	mp2	gp0	xtal	
if 0x7c[31] = 0, use 0x79; if 0x7c[31] = 1, use 0x7c;	cts_hevc_clk	dos	0x79[24] 0x7c[24]	0x79[22:16] 0x7c[22:16]	0x79[27:25] 0x7c[27:25]	fclk_div4	fclk_div3	fclk_div5	fclk_div7	mp1	mp2	gp0	xtal	
if 0x7a[15] = 0, use 0x78; if 0x7a[15] = 1, use 0x7a;	cts_vdec_clk	dos	0x78[8] 0x7a[8]	0x78[6:0] 0x7a[6:0]	0x78[11:9] 0x7a[11:9]	fclk_div4	fclk_div3	fclk_div5	fclk_div7	mp1	mp2	gp0	xtal	
	cts ge2d clk	ge2d/dmc	0x7d[30]			cts vapbclk								
0x8a[15] = 0 , use up 8 src; 0x8a[15] = 1 , use low 8 src;	gen_clk_out	gpio	0x8a[11]	0x8a[10:0]	0x8a[15:12]	xtal	cts slow oscin	cpu clk div16	ddr clk div64	vid pll	vid pll	mp0	mp1	
						mp2	fclk div4	fclk div3	fclk div5	cts msr clk	fclk div7	gp0	DAC CLK	
all fixed div32	tst clk out[10:0]	gpio		/32		sys pll div16	ddr clk div64	vid pll	mp0	mp1	mp2	mp3	fclk div5	gp0
if 0x6c[31] = 0, use 0x6c[11:0]; if 0x6c[31] = 1, use 0x6c[27:16]	cts_mali_clk	mali	0x6c[8] 0x6c[24]	0x6c[6:0] 0x6c[22:16]	0x6c[11:9] 0x6c[27:25]	xtal	gp0	mp2	mp1	fclk_div7	fclk_div4	fclk_div3	fclk_div5	
	cts nand core clk	nand	0x97[7]	0x97[6:0]	0x97[11:9]	xtal	fclk div2	fclk div3	fclk div5	fclk div7	mp2	mp3	gp0	
	cts msr clk	periphs	0x21d7[19]	0x21d8[7:0]	0x21d7[26:20]	see clkmsr								
	cts msr clk hs	periphs	0x21d8[28]			fclk div5								
	cts pwm A clk	periphs	0x2156[15]	0x2156[14:8]	0x2156[5:4]	xtal	vid pll	fclk div4	fclk div3					
	cts pwm B clk	periphs	0x2156[23]	0x2156[22:16]	0x2156[7:6]	xtal	vid pll	fclk div4	fclk div3					
	cts pwm C clk	periphs	0x2192[15]	0x2192[14:8]	0x2192[5:4]	xtal	vid pll	fclk div4	fclk div3					
	cts pwm D clk	periphs	0x2192[23]	0x2192[22:16]	0x2192[7:6]	xtal	vid pll	fclk div4	fclk div3					
	cts pwm E clk	periphs	0x21b2[15]	0x21b2[14:8]	0x21b2[5:4]	xtal	vid pll	fclk div4	fclk div3					
	cts pwm F clk	periphs	0x21b2[23]	0x21b2[22:16]	0x21b2[7:6]	xtal	vid pll	fclk div4	fclk div3					
	cts sar adc clk	periphs	0xf6[8]	0xf6[7:0]	0xf6[10:9]	xtal	clk81							
	sc clk	periphs	0x2110[24]	0x2112[19:12]	0x2112[31:30]	fclk div4	fclk div3	fclk div5	xtal					
	cts sd emmc clk A	sd emmc A	0x99[7]	0x99[6:0]	0x99[11:9]	xtal	fclk div2	fclk div3	fclk div5	fclk div7	mp2	mp3	gp0	
	cts sd emmc clk B	sd emmc B	0x99[23]	0x99[22:16]	0x99[27:25]	xtal	fclk div2	fclk div3	fclk div5	fclk div7	mp2	mp3	gp0	
	clk81	all module			0x5d[8]	cts slow oscin clk	mpeg pll clk							
	usb 32k alt	usb	0x88[9]	0x88[8:0]	0x88[11:10]	xtal	cts slow oscin							
if 0x7d[31] = 0, use 0x7d[11:0]; if 0x7d[31] = 1, use 0x7d[27:16];	cts_vapbclk	vapb	0x7d[8] 0x7d[24]	0x7d[6:0] 0x7d[22:16]	0x7d[11:9] 0x7d[27:25]	fclk_div4	fclk_div3	fclk_div5	fclk_div7	mp1	vid_pll	mp2	gp0	
	alt 32k clk	vpu	0x89[15]	0x89[14:0]	0x89[17:16]	xtal	cts slow oscin	fclk div3	fclk div5					
see video	cts enci clk	vpu	see video			vid pll	fclk div4	fclk div3	fclk div5	vid pll	fclk div7	mp1		
	cts encl clk	vpu				vid pll	fclk div4	fclk div3	fclk div5	vid pll	fclk div7	mp1		
	cts encp clk	vpu				vid pll	fclk div4	fclk div3	fclk div5	vid pll	fclk div7	mp1		
	cts hdmi tx pixel clk	vpu				vid pll	fclk div4	fclk div3	fclk div5	vid pll	fclk div7	mp1	tcon clk	
	cts vdac clk	vpu				vid pll	fclk div4	fclk div3	fclk div5	vid pll	fclk div7	mp1		
	lcd an clk ph2	vpu				vid pll	fclk div4	fclk div3	fclk div5	vid pll	fclk div7	mp1		
	lcd an clk ph3	vpu				vid pll	fclk div4	fclk div3	fclk div5	vid pll	fclk div7	mp1		
	cts hdcp22 skpclk	vpu	0x7c[24]	0x7c[22:16]	0x7c[26:25]	xtal	fclk div4	fclk div3	fclk div5					
	cts vdin meas clk	vpu	0x94[8]	0x94[6:0]	0x94[11:9]	xtal	fclk div4	fclk div3	fclk div5	vid pll	vid pll			
	cts vid lock clk	vpu	0xf2[7]	0xf2[6:0]	0xf2[9:8]	xtal	cts encl clk	cts enci clk	cts encp clk					
0x59-0xf6: base = 0xc883c000	cts_vpu_clk	vpu	0x6f[8] 0x6f[24]	0x6f[6:0] 0x6f[22:16]	0x6f[11:9] 0x6f[27:25]	fclk_div4	fclk_div3	fclk_div5	fclk_div7	mp1	vid_pll	mp2	gp0	
	cts hdmitx sys clk	vpu	0x73[8]	0x73[6:0]	0x73[10:9]	xtal	fclk div4	fclk div3	fclk div5					
	cts hdcp22 esmclk	vpu/dmc	0x7c[8]	0x7c[6:0]	0x7c[10:9]	fclk div7	fclk div4	fclk div3	fclk div5					
0x1516-0x21d8: base = 0xc1100000	cts_oscin_clk	watchdog/ periphs				xtal								

The video clock tree is shown as following:

Fig III.21.8 Video Clock Tree



21.3 Clock Gating

Modules and sub-modules within the chip can be disabled by shutting off the clock. The control for these clocks comes from six CBUS registers that collectively make up a 64-bit register that controls the MPEG_DOMAIN and a32-bit register that controls the OTHER_ DOMAIN. The table below indicates the Bits associated with either the MPEG_DOMAIN and OTHER_DOMAIN gated clock enables. The table is organized by function rather than by bit order because it makes it easier to determine how to turn on/off a particular function within the chip. If a bit is set high, the clock is enabled. If a bit is set low, the clock is turned off and the module is disabled.

Table III. 21.3 AO Domain Clock Gating CBUS 0x1055 control Bits

Address	Bit(s)	Module Description
0xc883c154 0xc883c0d0	31:4	Unused
	4	AO I2C
	3	Always on Registers, timers,…
	2	AHB bus in the always on domain
	1	AHB SRAM
	0	AO CPU

Table III.21.4 EE Domain Clock Gating (clk81)

Address	Bit(s)	Module Description
0xc883c140 0xc883c0c0	31	Reserved
	30	SPI Interface
	29	Reserved
	28	acodec
	27	dma
	26	emmc_c
	25	emmc_b
	24	emmc_a
	23	ASSIST_MISC
	22	bt656
	21	Reserved
	20	Reserved
	19	HIU Registers
	18	Reserved
	17	Reserved
	16	ASYNC_FIFO
	15	STREAM Interface
	14	Reserved
	13	UART0
	12	Random Number generator
	11	Smart Card
	10	SAR ADC
	9	I2C Master / I2C SLAVE
	8	SPICC
	7	PERIPHS module top level (there are separate register bits for internal blocks)
	6	PL310 (AXI Matrix) to CBUS
	5	ISA module
	4	Reserved
	3	Reserved
	2	Reserved
	1	u_dos_top()
	0	DDR Interfaces and bridges
0xc883c144 0xc883c0c4	31	ROM BOOT ROM clock
	30	EFUSE logic
	29	AHB ARB0
	28	Reserved
	27	Reserved
	26	USB General

Address	Bit(s)	Module Description
	25	U_parser_top()
	24	Reserved
	23	RESET
	22	USB 1
	21	USB 0
	20	General 2D Graphics Engine
	19	Reserved
	18	Reserved
	17	Reserved
	16	UART1
	15	AIU Top level (there is internal gating shown below)
	14	Block move core logic
	13	ADC
	12	Mixer Registers: u_ai_top.u_mixer_reg.clk
	11	Mixer: u_ai_top.u_aud_mixer.clk
	10	AIFIFO2: u_ai_top.u_aififo2.clk
	9	AMCLK measurement circuit
	8	I2S Out: This bit controls the clock to the logic between the DRAM control unit and the FIFO's that transfer data to the audio clock domain. (u_ai_top.i2s_fast.clk)
	7	IEC958: iec958_fast()
	6	AIU – ai_top_glue u_ai_top.ai_top_glue.clk u_ai_top.fifo_async_fast_i2s.clk u_ai_top.fifo_async_fast_958.clk
	5	Reserved
	4	Set top box demux module u_stb_top.clk
	3	Ethernet core logic
	2	I2S / SPDIF Input
	1	Reserved
	0	Reserved
0xc883c148 0xc883c0c8	31	Reserved
	30	gic
	29	Reserved
	28	Reserved
	27	Reserved
	26	Secure AHB to APB3 Bridge
	25	VPU Interrupt
	24	Reserved
	23	Reserved
	22	sar_adc
	21	UART3
	20	Reserved
	19	Reserved
	18	Reserved
	17	Reserved
	16	Reserved

Address	Bit(s)	Module Description
	15	UART 2
	14	Reserved
	13	Reserved
	12	DVIN
	11	MMC PCLK
	10	AIU PCLK
	9	USB0 to DDR bridge
	8	USB1 to DDR bridge
	7	bt656_2
	6	bt656
	5	pdm
	4	HDMI PCLK
	3	HDMI interrupt synchronization
	2	AHB control bus
	1	AHB data bus
	0	Reserved
0xc883c150 0xc883c0cc	31	Reserved
	30	Reserved
	29	Reserved
	28	Reserved
	27	Reserved
	26	VCLK2_OTHER
	25	VCLK2_VENCL
	24	VCLK2_VENCL MMC Clock All
	23	VCLK2_ENCL
	22	VCLK2_ENCT
	21	Random Number Generator
	20	ENC480P
	19	Reserved
	18	Reserved
	17	Reserved
	16	IEC958_GATE
	15	Reserved
	14	AOCLK_GATE
	13	Reserved
	12	Reserved
	11	Reserved
	10	DAC_CLK
	9	VCLK2_ENCP
	8	VCLK2_ENCI
	7	VCLK2_OTHER
	6	VCLK2_VENCT
	5	VCLK2_VENCT
	4	VCLK2_VENCP
	3	VCLK2_VENCP
	2	VCLK2_VENCI
	1	VCLK2_VENCI

Address	Bit(s)	Module Description
	0	Reserved

21.4 Register Description

Each register final address = 0xC883C000 + offset * 4

SCR System Clock Reference 0x0B

Bit(s)	R/W	Default	Description
31-0	R/W	0	System clock reference high: Bits 31:16

TIMEOUT_VALUE: Program timer 0x0F

Bit(s)	R/W	Default	Description
15-12	R	0	Unused
11-0	R/W	0	Program timer

Increased by 1 every 900 cycles. Triggers timer interrupt to CPU when it expires.

HHI_GP0_PLL_CNTL 0x10

Bit(s)	R/W	Default	Description
31	R	0	LOCK
30	R/W	0	ENABLE
29	R/W	0	RESET
28~18	R/W	0	Reserved
17~16	R/W	0	OD
15~14	R/W	0	Reserved
13~9	R/W	0	N
8~0	R/W	0	M

HHI_GP0_PLL_CNTL2 0x11

Bit(s)	R/W	Default	Description
[31:28]	R/W	0	LM_W
[27:22]	R/W	0	LM_S
[21]	R/W	0	DPFD_LMODE
[20:19]	R/W	0	FB_OD
[18:17]	R/W	0	FREF_SEL
[16]	R/W	0	FREQ_SHIFT_EN
[15:14]	R/W	0	FREQ_SHIFT_V
[13]	R/W	0	LOCK_BYPASSN
[12]	R/W	0	SDMNC_EN
[11]	R/W	0	SDMNC_MODE
[10]	R/W	0	SDMNC_RANGE
[9:7]	R/W	0	SDMNC_ULMS
[6:0]	R/W	0	SDMNC_POWER

HHI_GP0_PLL_CNTL3 0x12

Bit(s)	R/W	Default	Description
31~30	R/W	0	Reserved
29~26	R/W	0	FILTER_PVT2
25~22	R/W	0	FILTER_PVT1
21~11	R/W	0	FILTER_ACQ2
10~0	R/W	0	FILTER_ACQ1

HHI_GP0_PLL_CNTL4 0x13

Bit(s)	R/W	Default	Description
[31:28]	R/W	0	SSC_DEP_SEL
[27]	R/W	0	SSC_EN
[26:25]	R/W	0	SSC_MODE
[24:23]	R/W	0	SSC_OFFSET
[22:21]	R/W	0	SSC_STR_M
[20]	R/W	0	SSEN
[19:17]	R/W	0	SS_AMP
[15:4]	R/W	0	REVE
[3]	R/W	0	PVT_FIX_EN
[2]	R/W	0	DCO_SDM_EN
[1]	R/W	0	IIR_BYPASS_N
[0]	R/W	0	TDC_EN

HHI_GP0_PLL_CNTL5 0x14

Bit(s)	R/W	Default	Description
[31:27]	R/W	0	SS_CLK
[26]	R/W	0	SS_CLK_SEL
[25]	R/W	0	TDC_CAL_EN
[24]	R/W	0	TDC_CODE_NEW
[23:22]	R/W	0	TDC_DELAY_C
[21:20]	R/W	0	TDC_OFF_C
[19:18]	R/W	0	VBG_CT_VC
[17:16]	R/W	0	VBG_PTAT_VC
[15]	R/W	0	CKOUT_EN

HHI_GP0_PLL_STS 0x15

Bit(s)	R/W	Default	Description
[31]	R/W	0	DPLL_LOCK
[22:16]	R/W	0	SDMNC_MONITOR
[10:1]	R/W	0	DPLL_OUT_RSV
[0]	R/W	0	AFC_DONE

HHI_GP0_PLL_CNTL1 0x16

Bit(s)	R/W	Default	Description
[31:30]	R/W	0	ACQ_R_CTR
[29]	R/W	0	AFC_CLK_SEL
[28]	R/W	0	AFC_DSEL_BYPASS
[27:26]	R/W	0	AFC_DSEL_IN
[25:24]	R/W	0	AFC_HOLD_T
[23:22]	R/W	0	AFC_NT
[21:19]	R/W	0	DATA_SEL
[18]	R/W	0	DCO_BAND_OPT
[17]	R/W	0	DCO_M_EN
[16:15]	R/W	0	DCO_SDMCK_SEL
[14:13]	R/W	0	DCVC_IN
[12]	R/W	0	DIV_MODE
[11:0]	R/W	0	DIV_FRAC

HHI_XTAL_DIVN_CNTL 0x2f

Bit(s)	R/W	Default	Description
31~20	R/W	0	reserved
19	R/W	0	gpioCLK_3_xtal_en, if=1,gclk3=xtal
18	R/W	0	gpioCLK_3_xtal_2_en, if=1, gclk3=xtal/2
17	R/W	0	reserved
16	R/W	0	reserved
15	R/W	0	gpioCLK_2_xtal_2, if=1,gclk2=xtal/2
14	R/W	0	gpioCLK_2_xtal, if=1,gclk2=xtal

Bit(s)	R/W	Default	Description
13	R/W	0	reserved
12	R/W	0	gpioCLK_1_fclk_div2_en, if=1, clk1=fclk/2/N
11	R/W	0	gpioCLK_0_xtal_2, if=1,gclk0=xtal/2
10	R/W	0	gpioCLK_0_xtal, if=1,gclk0=xtal
9	R/W	0	reserved
8	R/W	0	Fclk_div2_gclk_en
7~0	R/W	0	Fclk_div2_tcnt, only to gpioclk1 = fclk/2/tcnt

HHI_GCLK2_MPEG0 0x30

Bit(s)	R/W	Default	Description
31-0	R/W	All 0	Bits [31:0] of the composite MPEG clock gating register

HHI_GCLK2_MPEG1 0x31

Bit(s)	R/W	Default	Description
31-0	R/W	All 0	Bits [63:32] of the composite MPEG clock gating register

HHI_GCLK2_MPEG2 0x32

Bit(s)	R/W	Default	Description
31-0	R/W	All 0	Bits [63:32] of the composite MPEG clock gating register

HHI_GCLK2_OTHER 0x34

Bit(s)	R/W	Default	Description
31-0	R/W	All 0	Bits [31:0] of the composite Other clock gating register

HHI_GCLK2_AO 0x35

Bit(s)	R/W	Default	Description
31-8	R	0	Unused
7-5	R/W	0x0	unused
4	R/W	0	i2c
3	R/W	0	Always On Register logic
2	R/W	0	Always on general clock in the always on domain
1	R/W	0	AHB SRAM
0	R/W	0	AO CPU

HHI_TIMER90K 0x3b

Bit(s)	R/W	Default	Description
31-16	R/W	0	Unused
15-0	R/W	0x384	90khz divider

HHI_MEM_PD_REG0 0x40

Bit(s)	R/W	Default	Description
31~16	R/W	0xFFFFFFFF	Reserved
15~8	R/W	0xFF	HDMI memory PD
7~4	R/W	0xF	Reserved
3~2	R/W	0x3	Ethernet memory PD
1~0	R/W	0x3	Reserved

HHI_VPU_MEM_PD_REG0 0x41

Bit(s)	R/W	Default	Description
31~30	R/W	0x3	sharp
29~28	R/W	0x3	Deinterlacer – di_post: 11 = power down. 00 = normal operation
27~26	R/W	0x3	Deinterlacer – di_pre
25~24	R/W	0x3	Reserved
23~22	R/W	0x3	Reserved
21~20	R/W	0x3	Reserved
19~18	R/W	0x3	Vdin1 memory
17~16	R/W	0x3	Vdin0 memory
15~14	R/W	0x3	Osd_scaler memory
13~12	R/W	0x3	Scaler memory
11~10	R/W	0x3	Vpp output fifo
9~8	R/W	0x3	Color management module
7~6	R/W	0x3	Vd2 memory
5~4	R/W	0x3	Vd1 memory
3~2	R/W	0x3	Osd2 memory
1~0	R/W	0x3	Osd1 memory

HHI_VPU_MEM_PD_REG1 0x42

Bit(s)	R/W	Default	Description
31~30	R/W	0x3	Reserved
29~28	R/W	0x3	Reserved
27~26	R/W	0x3	Reserved
25~24	R/W	0x3	CVBS inci interface
23~22	R/W	0x3	Panel encl top
21~20	R/W	0x3	Hdmi encp interface
19~18	R/W	0x3	Reserved
17~16	R/W	0x3	Afbc dec
15~14	R/W	0x3	Vpu arb
13~12	R/W	0x3	Reserved
11~10	R/W	0x3	Reserved
9~8	R/W	0x3	Reserved
7~6	R/W	0x3	Reserved
5~4	R/W	0x3	Reserved
3~2	R/W	0x3	Reserved
1~0	R/W	0x3	Reserved

HHI_VIID_CLK_DIV 0x4a

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

HHI_VIID_CLK_CNTL 0x4b

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

Bit(s)	R/W	Default	Description
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

HHI_GCLK_MPEG0 0x50

Bit(s)	R/W	Default	Description
31-0	R/W	0	Bits [31:0] of the composite MPEG clock gating register

HHI_GCLK_MPEG1 0x51

Bit(s)	R/W	Default	Description
31-0	R/W	0	Bits [63:32] of the composite MPEG clock gating register

HHI_GCLK_MPEG2 0x52

Bit(s)	R/W	Default	Description
31-0	R/W	0	Bits [63:32] of the composite MPEG clock gating register

HHI_GCLK_OTHER 0x54

Bit(s)	R/W	Default	Description
31-0	R/W	0	Bits [31:0] of the composite Other clock gating register

HHI_GCLK_AO 0x55

Bit(s)	R/W	Default	Description
31-8	R	0	Unused
7-5	R/W	0x7	unused
4	R/W	1	i2c
3	R/W	1	Always On Register logic
2	R/W	1	Always on general clock in the always on domain
1	R/W	1	AHB SRAM
0	R/W	1	AO CPU

HHI_SYS_CPU_CLK_CNTL1 0x57

Bit(s)	R/W	Default	Description
31-25	R/W	0	Reserved
24	R/W	0	Sys_pll_div16_en
23	R/W	0	A53_trace_clk _DIS: Set to 1 to manually disable the A53_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	0	A53_trace_clk: 1. A53 clock divided by 2 2. A53 clock divided by 3 3. A53 clock divided by 4 4. A53 clock divided by 5 5. A53 clock divided by 6 6. A53 clock divided by 7 **7.** A53 clock divided by 8
19	R/W	0	Timestamp CNTCLKEN_dis
18	R/W	0	AXI_CLK_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
16	R/W	0	APB_CLK_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 CNTCLKEN
14~12	R/W	0	Timestamp cntclk mux(not used)

11~9	R/W	0	AXI_CLK _MUX: 0 A53 clock divided by 2 1 A53 clock divided by 3 2 A53 clock divided by 4 3 A53 clock divided by 5 4 A53 clock divided by 6 5 A53 clock divided by 7 **6** A53 clock divided by 8
8~6	R/W	0	atCLK _MUX: 0 A53 clock divided by 2 1 A53 clock divided by 3 2 A53 clock divided by 4 3 A53 clock divided by 5 4 A53 clock divided by 6 5 A53 clock divided by 7 **6** A53 clock divided by 8
5~3	R/W	0	APB_CLK _MUX: 1. A53 clock divided by 2 2. A53 clock divided by 3 3. A53 clock divided by 4 4. A53 clock divided by 5 5. A53 clock divided by 6 6. A53 clock divided by 7 **7.** A53 clock divided by 8
2	R/W	0	Soft_reset
1	R/W	0	Sys_cpu_clk_div16_en
0	R/W	0	Pclk_en_dbg

HHI_SYS_CPU_RESET_CNTL 0x58

Bit(s)	R/W	Default	Description
31-11	R/W	0	Reserved
10	R/W	0	Cpu_axi_reset
9	R/W	0	nPRESETDBG
8	R/W	0	nL2RESET
7-4	R/W	0	nCORERESET[3:0]
3-0	R/W	0	Cpu_soft_reset[3:0]

HHI_VID_CLK_DIV 0x59

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
15-8	R/W	0	XD1
7-0	R/W	0	XD0

HHI_MPEG_CLK_CNTL 0x5d

Bit(s)	R/W	Default	Description
31	R/W	0	NEW_DIV_EN: If This bit is set to 1, then Bits[30:16] make up the clk81 divider. If This bit is 0, then Bits[6:0] dictate the divider value. This is a new feature that allows clk81 to be divided down to a very slow frequency.
30~16	R/W	0	NEW_DIV: New divider value if Bit[31] = 1
15	R/W	0	Production clock enable
14-12	R.W	6	MPEG_CLK_SEL (See clock document)
11-10	R/W	0	unused
9	R.W	0	RTC Oscillator Enable: Set This bit to 1 to connect the RTC 32khz oscillator output as the XTAL input for the divider above
8	R/W	0	Divider Mux: 0 = the ao cpu clock and the MPEG system clock are connected to the 27Mhz crystal. 1 = the ao cpu clock and the MPEG system clock are connected to the MPEG PLL divider

Bit(s)	R/W	Default	Description
7	R/W	1	PLL Mux: 0 = all circuits associated with the MPEG PLL are connected to 27Mhz. 1 = all circuits associated with the MPEG PLL are connected to the MPEG PLL
6-0	R/W	0	PLL Output divider. The MPEG System clock equals the video PLL clock frequency divided by (N+1). Note: N must be odd (1,3,5,….) so that the MPEG clock is divided by an even number to generate a 50% duty cycle.

HHI_AUD_CLK_CNTL 0x5e

Bit(s)	R/W	Default	Description
31-26	R/W	0	Unused
25-24	R/W	0	Audio DAC clock select (See clock document)
23	R/W	0	Audio DAC Clock enable
22-16	R/W	0	Audio DAC Clock divider
15-11	R/W	0	Unused
10-9	R/W	0	AMCLK_SRC_SEL: (see clock tree document)
8	R/W	0	PLL Clock Enable. Set This bit to 1 to enable the PLL to the rest of the logic. To avoid glitching state machines inside the chip please change the PLL using the following sequence: • Set Bit[8] = 0 to block the logic from the PLL • set the PLL using register 0x15b and wait for the PLL to settle (1mS) • Set Bit[8] = 1 to enable the PLL driving all of the logic
7-0	R/W	1	PLL Output divider. The Audio System clock equals the video PLL clock frequency divided by (N+1).

HHI_VID_CLK_CNTL 0x5f

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

HHI_AUD_CLK_CNTL2 0x64

Bit(s)	R/W	Default	Description
31-28	R/W	0	Unused
27	R/W	0	IEC958_USE_CNTL: If this bit is set to 1, then bits[26;16] are used as the IEC958 clock divider
26-25	R/W	0	IEC958_CLK_SRC_SEL. See the clock tree document
24	R/W	0	IEC958_CLK_EN
23-16	R/W	0	IEC958_CLK_DIV. This is a new feature in M6TV. In the past, the IEC958 Clock was slaved to the AMCLK. In M6TV we added the ability to separately control the clock divider for IEC958. For these bits to take effect, you must set bit[27] above.
15-0	R/W	0	Unused

HHI_VID_CLK_CNTL2 0x65

Bit(s)	R/W	Default	Description
31-16	R	0	
15-9	R/W	0	Reserved
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

Bit(s)	R/W	Default	Description
1	R/W	1	ENCT gated clock control
0	R/W	1	ENCI gated clock control

HHI_SYS_CPU_CLK_CNTL0 0x67

Bit(s)	R/W	Default	Description
31	R	0	Final_mux_sel
30	R	0	Final_dyn_mux_sel
29	R	0	Busy_cnt
28	R	0	busy
26	R/W	0	Dyn_enable
25-20	R/W	0	Mux1_divn_tcnt
18	R/W	0	Postmux1
17-16	R/W	0	Premux1
15	R/W	0	Manual_mux_mode
14	R/W	0	Manual_mode_post
13	R/W	0	Manual_mode_pre
12	R/W	0	Force_update_t
11	R/W	0	Final_mux_sel
10	R/W	0	Final_dyn_mux_sel
9-4	R/W	0	mux0_divn_tcnt
3	R/W	0	Rev
2	R/W	0	Postmux0
1-0	R/W	0	Premux0

HHI_VID_PLL_CLK_DIV 0x68

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

HHI_AUD_CLK_CNTL3 0x69

Bit(s)	R/W	Default	Description
31-29	R/W	0	Unused
18:17	R/W	0	Pdm_clk_src_sel: 0:cts_amclk 1:mp0_clk_out 2:mp1_clk_out 3:mp2_clk_out
16	R/W	0	Pdm_clk_en
15-0	R/W	0	Pdm_clk_div

HHI_MALI_CLK_CNTL 0x6c

Bit(s)	R/W	Default	Description
31-28	R/W	0	Reserved
27~25	R/W	0	CLK_SEL: See the Clock Tree document for information related to cts_mali _clk
24	R/W	0	CLK_EN: See the Clock Tree document for information related to cts_ mali _clk
23	R/W	0	Reserved
22~16	R/W	0	CLK_DIV: See the Clock Tree document for information related to cts_ mali _clk
15~12			
11~9	R/W	0	CLK_SEL: See the Clock Tree document for information related to cts_mali _clk
8	R/W	0	CLK_EN: See the Clock Tree document for information related to cts_ mali _clk
7	R/W	0	Reserved
6-0	R/W	0	CLK_DIV: See the Clock Tree document for information related to cts_ mali _clk

HHI_VPU_CLK_CNTL 0x6f

Bit(s)	R/W	Default	Description
31	R/W	0	Final mux sel
30-29	R/W	0	Reserved
28	R/W	0	TVFE MCLK EN
27~25	R/W	0	CLK_SEL: See the Clock Tree document for information related to cts_vpu_clk
24	R/W	0	CLK_EN: See the Clock Tree document for information related to cts_vpu_clk
23	R/W	0	Reserved
22~16	R/W	0	CLK_DIV: See the Clock Tree document for information related to cts_vpu_clk
15~12			
11~9	R/W	0	CLK_SEL: See the Clock Tree document for information related to cts_vpu_clk
8	R/W	0	CLK_EN: See the Clock Tree document for information related to cts_vpu_clk
7	R/W	0	Reserved
6-0	R/W	0	CLK_DIV: See the Clock Tree document for information related to cts_vpu_clk

HHI_HDMI_CLK_CNTL 0x73

Bit(s)	R/W	Default	Description
31-20	R/W	0	Reserved
19~16	R/W	0	crt_hdmi_pixel_clk_sel
15~11	R/W	0	Reserved
10~9	R/W	0	CLK_SEL: See the Clock Tree document for information related to cts_hdmi_sys_clk
8	R/W	0	CLK_EN: See the Clock Tree document for information related to cts_hdmi_sys_clk
7	R/W	0	Reserved
6-0	R/W	0	CLK_DIV: See the Clock Tree document for information related to cts_hdmi_sys_clk

HHI_VDEC_CLK_CNTL 0x78

Bit(s)	R/W	Default	Description
31-28	R/W	0	Reserved
27~25	R/W	0	CLK_SEL: See the Clock Tree document for information related to cts_hcodec_clk
24	R/W	0	CLK_EN: See the Clock Tree document for information related to cts_hcodec_clk
23	R/W	0	Reserved
22~16	R/W	0	CLK_DIV: See the Clock Tree document for information related to ctshcodec_clk
15~12	R/W	0	Reserved
11~9	R/W	0	CLK_SEL: See the Clock Tree document for information related to cts_vdec_clk
8	R/W	0	CLK_EN: See the Clock Tree document for information related to cts_vdec_clk
7	R/W	0	Reserved
6-0	R/W	0	CLK_DIV: See the Clock Tree document for information related to cts_vdec_clk

HHI_VDEC2_CLK_CNTL 0x79

Bit(s)	R/W	Default	Description
31-28	R/W	0	Reserved
27~25	R/W	0	HEVC_CLK_SEL: See the Clock Tree document for information related to cts_vdec2_clk
24	R/W	0	HEVC_CLK_EN: See the Clock Tree document for information related to cts_vdec2_clk
23	R/W	0	Reserved
22~16	R/W	0	HEVC_CLK_DIV: See the Clock Tree document for information related to cts_vdec2_clk
15~12			
11~9	R/W	0	VDEC2_CLK_SEL: See the Clock Tree document for information related to cts_vdec2_clk
8	R/W	0	VDEC2_CLK_EN: See the Clock Tree document for information related to cts_vdec2_clk
7	R/W	0	Reserved
6-0	R/W	0	VDEC2_CLK_DIV: See the Clock Tree document for information related to cts_vdec2_clk

HHI_VDEC3_CLK_CNTL 0x7a

Bit(s)	R/W	Default	Description
31~0	R/W	0	See the clock tree document. This register controls the Alternate clock for cts_vdec_clk and cts_hcodec_clk

HHI_VDEC4_CLK_CNTL 0x7b

Bit(s)	R/W	Default	Description
31-0	R/W	0	See the clock tree document. This register controls the Alternate clock for cts_vdec2_clk and cts_hevc_clk

HHI_HDCP22_CLK_CNTL 0x7c

Bit(s)	R/W	Default	Description
31-0	R/W	0	Reserved
26-25	R/W	0	Clk_sel: 0:cts_oscin_clk 1:fclk_div4 2:fclk_div3 3:fclk_div5
24	R/W	0	Clk_en
23-0	R/W	0	Clk_div

HHI_VAPBCLK_CNTL 0x7d

Bit(s)	R/W	Default	Description
31	R/W	0	Final_mux_sel
30	R/W	0	Enable
29-28	R/W	0	Reserved
27-25	R/W	0	Mux1_sel: 0:fclk_div4 1:fclk_div3 2:fclk_div5 3:fclk_div7 4:mp1_clk_out 5:vid_pll_clk 6:mp2_clk_out 7:gp0_pll_clk
24	R/W	0	Mux1_en
23	R/W	0	Reserved
22-16	R/W	0	Mux1_div
15-12	R/W	0	Reserved
11-9	R/W	0	Mux0_sel,as mux1_sel
8	R/W	0	Mux0_en
7	R/W	0	Reserved
6-0	R/W	0	Mux0_div

HHI_VPU_CLKB_CNTL 0x83

Bit(s)	R/W	Default	Description
31-9	R/W	0	Reserved
8	R/W	0	Vpu_clkb_en
7-0	R/W	0	Vpu_clkb_div

HHI_USB_CLK_CNTL0x88

Bit(s)	R/W	Default	Description
31-12	R/W	0	Reserved
11-10	R/W	0	Usb_clk_sel: 0:cts_oscin_clk 1:rtc_oscin_i
9	R/W	0	Usb_clk_en
8-0	R/W	0	Usb_clk_div

HHI_32K_CLK_CNTL 0x89

Bit(s)	R/W	Default	Description
31-18	R/W	0	Reserved
17-16	R/W	0	Hi_32k_clk_sel: 0:cts_oscin_clk 1:cts_slow_oscin_clk

Bit(s)	R/W	Default	Description
			2:fclk_div3 3:fclk_div5
15	R/W	0	Hi_32k_clk_en
14	R/W	0	Reserved
13-0	R/W	0	Hi_32k_clk_div

HHI_GEN_CLK_CNTL 0x8a

Bit(s)	R/W	Default	Description
31-17	R/W	0	Reserved
16	R/W	0	OSCIN GATE CLK ENABLE
15~12	R/W	0	CLK_SEL: 0:cts_oscin_clk 1:rtc_oscin_i 2:sys_cpu_clk_div16 3:ddr_dpll_pt_clk 4:vid_pll_clk 5:vid2_pll_clk 6:mp0_clk_out 7:mp1_clk_out 8:mp2_clk_out 9:fclk_div4 10:fclk_div3 11:fclk_div5 12:cts_msr_clk 13:fclk_div7 14:gp0_pll_clk
11	R/W	0	CLK_EN: See the Clock Tree document for information related to gen_clk_out
10~0	R/W	0	CLK_DIV: See the Clock Tree document for information related to gen_clk_out

HHI_PCM_CLK_CNTL 0x96

Clock control for cts_pcm_mclk and cts_pcm_sclk

Bit(s)	R/W	Default	Description
31-23	R/W	0	unused
22	R/W	0	CLK_EN: See the Clock Tree document for information related to cts_pcm_sclk
21-16	R/W	0	CLK_DIV: See the Clock Tree document for information related to cts_pcm_sclk
15-12	R/W	0	Unused
11-10	R/W	0	CLK_SEL: See the Clock Tree document for information related to cts_pcm_mclk
9	R/W	0	CLK_EN: See the Clock Tree document for information related to cts_pcm_mclk
8-0	R/W	48	CLK_DIV: See the Clock Tree document for information related to cts_pcm_mclk

HHI_NAND_CLK_CNTL 0x97

Clock control for cts_pcm_mclk and cts_pcm_sclk

Bit(s)	R/W	Default	Description
31-12	R/W	0	unused
11-9	R/W	0	CLK_SEL: 0:cts_oscin_clk 1:fclk_div2 2:fclk_div3 3:fclk_div5 4:fclk_div7 5:mp2_clk_out 6:mp3_clk_out 7:gp0_pll_clk
8			Reserved
7	R/W	0	CLK_EN:
6-0	R/W	48	CLK_DIV

HHI_SD_EMMC_CLK_CNTL 0x99

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:fclk_div2 2:fclk_div3 3:fclk_div5 4:fclk_div7 5:mp2_clk_out 6:mp3_clk_out 7:gp0_pll_clk
24			Reserved
23	R/W	0	Sd_emmc_B_CLK_EN:
22-16	R/W	48	Sd_emmc_B_CLK_DIV
15-12	R/W	0	Reserved
11-9	R/W	0	Sd_emmc_A_CLK_SEL: 0:cts_oscin_clk 1:fclk_div2 2:fclk_div3 3:fclk_div5 4:fclk_div7 5:mp2_clk_out 6:mp3_clk_out 7:gp0_pll_clk
8			Reserved
7	R/W	0	Sd_emmc_A_CLK_EN:
6-0	R/W	48	Sd_emmc_A_CLK_DIV

HHI_MPLL_CNTL 0xa0

Bit(s)	R/W	Default	Description
31	R	-	MPLL_LOCK
30	R/W	0	MPLL_ENABLE. 1 = enable
29	R/W	0	MPLL_RESET
28-26	R/W	0	DDS0_P_SET
25	R/W	0	DDS0_SSEN
24-20	R/W	0	DDS0_F_SET
19~18	R/W	0	Reserved
17-16	R/W	0	MPLL_OD (See Clock Document for details)
15-14	R/W	0	Unused
13-9	R/W	0	MPLL_N
8-0	R/W	0	MPLL_M

HHI_MPLL_CNTL2 0xa1

Bit(s)	R/W	Default	Description
31-28	R/W	0	LM_W
27~22	R/W	0	LM_S
21	R/W	0	DPFD_LMODE
20~19	R/W	0	VC_IN
18~17	R/W	0	SDMCK_SEL
16	R/W	0	DCO_M_EN
15	R/W	0	SDM_PR_EN
14	R/W	0	DIV_MODE
13	R/W	0	AFC_DSEL_BYPASS
12	R/W	0	AFC_DSEL_IN
11-0	R/W	0	DPLL_DIV_FRAC

HHI_MPLL_CNTL3 0xa2

Bit(s)	R/W	Default	Description
31~30	R/W	0	RESERVED
29~26	R/W	0	FILTER_PVT2
25~22	R/W	0	FILTER_PVT1
21~11	R/W	0	FILTER_ACQ2
10~0	R/W	0	FILTER_ACQ1

HHI_MPLL_CNTL4 0xa3

Bit(s)	R/W	Default	Description
31~24	R/W	0	REVE
23~16	R/W	0	TDC_BUF
15	R/W	0	TDC_CAL_EN
14	R/W	0	PVT_FIX_EN
13~12	R/W	0	DCO_IUP
11~8	R/W	0	SS_AMP
7~4	R/W	0	SS_CLK
3	R/W	0	SSEN
2	R/W	0	DCO_SDM_EN
1	R/W	0	IIR_BYPASS_EN
0	R/W	0	TDC_EN

HHI_MPLL_CNTL5 0xa4

Bit(s)	R/W	Default	Description
31~28	R/W	0	DPLL_BGP_C
27~26	R/W	0	VR_FB2
25~24	R/W	0	VR_FB1
23~22	R/W	0	DDS_VC_VDD
21	R/W	0	DDS_LDO_RUPSEL
20	R/W	0	DDS_ENLDO
19~17	R/W	0	TEST_C
16~14	R/W	0	RESERVED
13	R/W	0	IRSEL
12	R/W	0	MP_OD
11~10	R/W	0	TDC_OFF_C
9~8	R/W	0	TDO_NC_SEL
7	R/W	0	TDO_CLK_SEL
6~5	R/W	0	TDC_CAL_PG
4~2	R/W	0	TDC_CAL_OFF
1-0	R/W	0	TDC_CAL_IG

HHI_MPLL_CNTL6 0xa5

Bit(s)	R/W	Default	Description
31~28	R/W	0	CKEN[3:0] [0] for MPLL output DIV3 [1] for MPLL output DIV4 [2] for MPLL output DIV5 [3] for MPLL output DIV7
27	R/W	0	CKEN[4] MPLL output DIV2 enable
26	R/W	0	SYS_DPLL_BGP_EN
25~24	R/W	0	SYS_DPLL_EXLDO
23~20	R/W	0	VREF_CS2
19~16	R/W	0	VREF_CF2
15~12	R/W	0	VREF_CS1
11~8	R/W	0	VREF_CF1
7~4	R/W	0	VREF_CS0
3~0	R/W	0	VREF_CF0

HHI_MPLL_CNTL7 (MP0) 0xa6

Bit(s)	R/W	Default	Description
31	R/W	0	IR_BYPASS0
30	R/W	0	MODLSEL0
29~26	R/W	0	IR_BYIN0
25	R/W	0	LP_EN0
24~16	R/W	0	N_IN0
15	R/W	0	SDM_EN0
14	R/W	0	EN_DDS0
13~0	R/W	0	SDM_IN0

HHI_MPLL_CNTL8 (MP1) 0xa7

Bit(s)	R/W	Default	Description
31	R/W	0	IR_BYPASS1
30	R/W	0	MODLSEL1
29~26	R/W	0	IR_BYIN1
25	R/W	0	LP_EN1
24~16	R/W	0	N_IN1
15	R/W	0	SDM_EN1
14	R/W	0	EN_DDS1
13~0	R/W	0	SDM_IN1

HHI_MPLL_CNTL9 (MP2) 0xa8

Bit(s)	R/W	Default	Description
31	R/W	0	IR_BYPASS2
30	R/W	0	MODLSEL2
29~26	R/W	0	IR_BYIN2
25	R/W	0	LP_EN2
24~16	R/W	0	N_IN2
15	R/W	0	SDM_EN2
14	R/W	0	EN_DDS2
13~0	R/W	0	SDM_IN2

HHI_MPLL_CNTL10 (MP2) 0xa9

Bit(s)	R/W	Default	Description
31~29	R/W	0	Reserved
28	R/W	0	Mpll_clk25M_en
27-25	R/W	0	Reserved
24	R/W	0	Mpll_clk_out_div3_en
23	R/W	0	MPLL_SSCLK_SEL
22:20	R/W	0	MPLL_DDS3_P_SET
19:15	R/W	0	MPLL_DDS3_F_SET
14	R/W	0	MPLL_DDS3_EN
1	R/W	0	MPLL_CLK_OUT_DIV3_EN
0	R/W	0	MPLL_CLK_OUT_DIV2_EN

HHI_MPLL3_CNTL0 0xb8

Bit(s)	R/W	Default	Description
31-28	R/W	0	MPLL_IR_BYIN3
25-12	R/W	0	MPLL_SDM_IN3
11	R/W	0	MPLL_SDM_EN3
10~2	R/W	0	MPLL_N_IN3
1	R/W	0	MPLL_MODELSEL3
0	R/W	0	MPLL_EN_DDS3

HHI_MPLL3_CNTL1 0xb9

Bit(s)	R/W	Default	Description
31~10	R/W	0	Reserved

Bit(s)	R/W	Default	Description
9	R/W	0	MPLL_LP_EN3
8	R/W	0	MPLL_IR_BYPASS3
7-4	R/W	0	MPLL_VREF_CS3
3-0	R/W	0	MPLL_VREF_CF3

HHI_VDAC_CNTL0 0xbd

Bit(s)	R/W	Default	Description
27	R/W	0	CDAC_BIAS_C
26	R/W	0	CDAC_EXT_VREF_EN
25	R/W	0	CDAC_DRIVER_ADJ
24	R/W	0	CDAC_CLK_PHASE_SEL
23~21	R/W	0	CDAC_RL_ADJ
20~16	R/W	0	CDAC_VREF_ADJ
15~8	R/W	0	CDAC_CTRL_RESV2
7~0	R/W	0	CDAC_CTRL_RESV1

HHI_VDAC_CNTL1 0xbe

Bit(s)	R/W	Default	Description
23~16	R	0	CDAC_DIG_OUT_RESV
15~4	R	0	Reserved
3	R/W	0	Cdac_pwd
2~0	R/W	0	CDAC_GSW

HHI_SYS_PLL_CNTL1 0xbf

Bit(s)	R/W	Default	Description
[31:30]	R/W	0	ACQ_R_CTR
[29]	R/W	0	AFC_CLK_SEL
[28:27]	R/W	0	AFC_HOLD_T
[26:25]	R/W	0	AFC_NT
[24:22]	R/W	0	DATA_SEL
[21]	R/W	0	DCO_BAND_OPT
[20]	R/W	0	DCO_M_EN
[19:18]	R/W	0	DCO_SDMCK_SEL
[17:16]	R/W	0	DCVC_IN
[15:14]	R/W	0	FB_OD
[13:12]	R/W	0	FREF_SEL
[11]	R/W	0	FREQ_SHIFT_EN
[10:9]	R/W	0	FREQ_SHIFT_V
[8]	R/W	0	LOCK_BYPASSN
[7]	R/W	0	SDMNC_EN
[6]	R/W	0	SDMNC_MODE
[5]	R/W	0	SDMNC_RANGE

HHI_SYS_PLL_CNTL 0xc0

Bit(s)	R/W	Default	
31	R		LOCK
[30]	R/W	0x0	EN
[29]	R/W	0x1	RESET
[28]	R/W	0x0	SSEN
[27:25]	R/W	0	SS_AMP
[24:20]	R/W	0x0	SS_CLK
[17:16]	R/W	0x0	OD
[13:9]	R/W	0x0	N
[8:0]	R/W	0x0	M

HHI_SYS_PLL_CNTL2 0xc1

Bit(s)	R/W	Default	Description
[31:28]	R/W	0x0	LM_W
[27:22]	R/W	0x0	LM_S
[21]	R/W	0x0	DPFD_LMODE
[16:15]	R/W	0x0	AFC_DSEL_IN
[14]	R/W	0x0	DIV_MODE
[13]	R/W	0x0	AFC_DSEL_BYPASS
[11:0]	R/W	0x0	DIV_FRAC

HHI_SYS_PLL_CNTL3 0xc2

Bit(s)	R/W	Default	Description
31-30	R/W	0x0	Reserved
29~26	R/W	0x0	FILTER_PVT2
25~22	R/W	0x0	FILTER_PVT1
21~11	R/W	0x0	FILTER_ACQ2
10~0	R/W	0x0	FILTER_ACQ1

HHI_SYS_PLL_CNTL4 0xc3

Bit(s)	R/W	Default	Description
[31:20]	R/W	0	REVE
[5]	R/W	0	CKOUT_EN
[4]	R/W	0	TDC_CAL_EN
[3]	R/W	0	PVT_FIX_EN
[2]	R/W	0	DCO_SDM_EN
[1]	R/W	0	IIR_BYPASS_N
[0]	R/W	0	TDC_EN

HHI_SYS_PLL_CNTL5 0xc4

Bit(s)	R/W	Default	Description
[31:25]	R/W	0	SDMNC_POWER
[24:22]	R/W	0	SDMNC_ULMS
[21:18]	R/W	0	SSC_DEP_SEL
[17]	R/W	0	SSC_EN
[16:15]	R/W	0	SSC_MODE
[14:13]	R/W	0	SSC_OFFSET
[12:11]	R/W	0	SSC_STR_M
[10]	R/W	0	SS_CLK_SEL
[9]	R/W	0	TDC_CODE_NEW
[7:6]	R/W	0	TDC_OFF_C
[5:4]	R/W	0	TDC_DELAY_C
[3:2]	R/W	0	VBG_CT_VC
[1:0]	R/W	0	VBG_PTAT_VC

HHI_SYS_PLL_STS 0xc5

Bit(s)	R/W	Default	Description
[31]	R	-	DPLL_LOCK
[22:16]	R	-	SDMNC_MONITOR
[10:1]	R	-	DPLL_OUT_RSV
[0]	R	-	AFC_DONE

HHI_DPLL_TOP_I 0xc6

Bit(s)	R/W	Default	Description
[18]	R	0	MPLL_IR_DONE2
[17]	R	-	MPLL_IR_DONE1
[16]	R	-	MPLL_IR_DONE0
[15:12]	R	-	MPLL_IR_OUT2
[11:8]	R	-	MPLL_IR_OUT1

Bit(s)	R/W	Default	Description
[7:4]	R	-	MPLL_IR_OUT0
[2]	R	-	MPLL_TDC_CAL_DONE

HHI_DPLL_TOP2_I 0xc7

Bit(s)	R/W	Default	Description
31-24	R	0	MPLL_DPLL_OUT_RSV
12	R	-	MPLL_IR_DONE3
11-8	R		MPLL_IR_OUT3

HHI_HDMI_PLL_CNTL 0xc8

There is some internal muxing controlled by “VLOCK_CNTL_EN” bit[20] of HHI_HDMI_PLL_CNTL6.

Bit(s)	R/W	Default	Description
31	R/W	0	Lock
30	R/W	0	Enable
29	R/W	0	Reserved
28	R/W	0	RESET
27~14	R/W	0	Reserved
13~9	R/W	0	N
8~0	R/W	0	M = VLOCK_CNTL_EN ? vpu_hdmi_top.m_int_pll : (this register)

HHI_HDMI_PLL_CNTL1 0xc9

There is some internal muxing controlled by “VLOCK_CNTL_EN” bit[20] of HHI_HDMI_PLL_CNTL6.

Bit(s)	R/W	Default	Description
[31:30]	R/W	0	ACQ_R_CTR
[29]	R/W	0	AFC_CLK_SEL
[28]	R/W	0	AFC_DSEL_BYPASS
[27:26]	R/W	0	AFC_DSEL_IN
[25:24]	R/W	0	AFC_HOLD_T
[23:22]	R/W	0	AFC_NT
[21:19]	R/W	0	DATA_SEL
[18]	R/W	0	DCO_BAND_OPT
[17]	R/W	0	DCO_M_EN
[16:15]	R/W	0	DCO_SDMCK_SEL
[14:13]	R/W	0	DCVC_IN
[12]	R/W	0	DIV_MODE
[11:0]	R/W	0	DIV_FRAC

HHI_HDMI_PLL_CNTL2 0xca

There is some internal muxing controlled by “VLOCK_CNTL_EN” bit[20] of HHI_HDMI_PLL_CNTL6.

Bit(s)	R/W	Default	Description
[31:30]	R/W	0	FB_OD
[29:28]	R/W	0	FREF_SEL
[27:26]	R/W	0	FREQ_SHIFT_V
[25]	R/W	0	LOCK_BYPASSN
[24:21]	R/W	0	OD
[20:19]	R/W	0	OD2
[18]	R/W	0	SDMNC_EN
[17]	R/W	0	SDMNC_MODE
[16]	R/W	0	SDMNC_RANGE
[15]	R/W	0	FREQ_SHIFT_EN
[14]	R/W	0	SSC_EN
[13:10]	R/W	0	SSC_DEP_SEL
[9:7]	R/W	0	SDMNC_ULMS
[6:0]	R/W	0	SDMNC_POWER

HHI_HDMI_PLL_CNTL3 0xcb

Bit(s)	R/W	Default	Description
[31:30]	R/W	0	SSC_MODE
[28:18]	R/W	0	FILTER_ACQ2
[17:7]	R/W	0	FILTER_ACQ1
[6]	R/W	0	DPFD_LMODE
[5:4]	R/W	0	SSC_OFFSET
[3:2]	R/W	0	SSC_STR_M
[1]	R/W	0	SS_CLK_SEL
[0]	R/W	0	TDC_CODE_NEW

HHI_HDMI_PLL_CNTL4 0xcc

Bit(s)	R/W	Default	Description
[29:27]	R/W	0	SS_AMP
[26:22]	R/W	0	SS_CLK
[21]	R/W	0	SSEN
[20]	R/W	0	DCO_SDM_EN
[19]	R/W	0	IIR_BYPASS_N
[18]	R/W	0	TDC_EN
[17:14]	R/W	0	LM_W
[13:8]	R/W	0	LM_S
[7:4]	R/W	0	FILTER_PVT2
[3:0]	R/W	0	FILTER_PVT1

HHI_HDMI_PLL_CNTL5 0xcd

Bit(s)	R/W	Default	Description
[31:20]	R/W	0	REVE
[19:18]	R/W	0	TDC_OFF_C
[17:16]	R/W	0	OUT_GATE_CTRL
[15:14]	R/W	0	TDC_DELAY_C
[13:12]	R/W	0	VBG_CT_VC
[11:10]	R/W	0	VBG_PTAT_VC
[9]	R/W	0	TDC_CAL_EN
[8]	R/W	0	PVT_FIX_EN
[7]	R/W	0	CLK_OUT_SEL
[5:4]	R/W	0	0: Adj_en_to_pll = adj_en_from_vpu 1: Adj_en_to_pll = 0 2: Adj_en_to_pll = adj_en_from_vpu 3: Adj_en_to_pll = 1
[3]	R/W	0	0: pll_m/pll_m_frac from hiu register; 1: pll_m/pll_m_frac from vpu

HHI_HDMI_PLL_STS 0xce

Bit(s)	R/W	Default	Description
[31]	R	0	HDMI_DPLL_LOCK
[22:16]	R/W	0	HDMI_SDMNC_MONITOR
[10:1]	R/W	0	HDMI_DPLL_OUT_RSV
[0]	R/W	0	HDMI_AFC_DONE

HHI_HDMI_PHY_CNTL0 0xe8

Bit(s)	R/W	Default	Description
31~16	R/W	0	HDMI_CTL1
15~0	R/W	0	HDMI_CTL0

HHI_HDMI_PHY_CNTL1 0xe9

Bit(s)	R/W	Default	Description
31:30	R/W	0	New_prbs_mode
29:28	R/W	0	New_prbs_prbsmode
27	R/W	0	New_prbs_sel
26	R/W	0	New_prbs_en
25:24	R/W	3	Ch3_swap: 0:ch0/1:ch1/2:ch2/3:ch3
23:22	R/W	2	Ch2_swap: 0:ch0/1:ch1/2:ch2/3:ch3
21:20	R/W	1	Ch1_swap: 0:ch0/1:ch1/2:ch2/3:ch3
19:18	R/W	0	Ch0_swap: 0:ch0/1:ch1/2:ch2/3:ch3
17	R/W	0	BIT_INVERT
16	R/W	0	MSB_LSB_SWAP
15	R/W	0	Capture_add1
14	R/W	0	CAPTURE_CLK_GATE_EN
13	R/W	0	HDMI_TX_PRBS_EN: Set to 1 to enable the PRBS engine
12	R/W	0	HDMI_TX_PRBS_ERR_EN: Set to 1 to enable the error flag detector. Set to 0 to reset the error detection logic
11~8	R/W	0	HDMI_TX_SET_HIGH: Set each bit to 1 to set the HDMI pin high
7~4	R/W	0	HDMI_TX_SET_LOW: Set each bit to 0 to set the HDMI Pins low
3	R/W	0	HDMI_FIFO_WR_ENALBE
2	R/W	0	HDMI_FIFO_ENABLE
1	R/W	0	HDMI_TX_PHY_CLK_EN: Set to 1 to enable the HDMI TX PHY
0	R/W	0	HDMI_TX_PHY_SOFT_RESET: Set to 1 to reset the HDMI TX PHY

HHI_HDMI_PHY_CNTL2 0xea

Bit(s)	R/W	Default	Description
31~9	R	0	Reserved
8	R	0	Test error
7~0	R	0	HDMI_REGRD

HHI_HDMI_PHY_CNTL3 0xeb

Bit(s)	R/W	Default	Description
31~0	R/W	0	RSV

HHI_HDMI_PHY_CNTL4 0xec

Bit(s)	R/W	Default	Description
[31:24]	R/W	0	New_prbs_err_thr
[21:20]	R/W	0	Dtest_sel
[19]	R/W	0	New_prbs_clr_ber_meter
[17]	R/W	0	New_prbs_freez_ber
[16]	R/W	0	New_prbs_inverse_in
[15:14]	R/W	0	New_prbs_mode
[13:12]	R/W	0	New_prbs_prbs_mode
[11:0]	R/W	0	New_prbs_time_window

HHI_HDMI_PHY_STATUS 0xed

Bit(s)	R/W	Default	Description
[29]	R		Prbs_enable
[28]	R		Test_err
[24]	R		New_prbs_pattern_nok
[20]	R		New_prbs_lock
[19:0]	R		New_prbs_ber_meter

HHI_VID_LOCK_CLK_CNTL 0xf2

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	Clk_div

HHI_BT656_CLK_CNTL 0xf5

Bit(s)	R/W	Default	Description
31~27	R/W	0	Reserved
26-25	R/W	0	Bt656_2_clk_sel, the same as bt656_1_clk_sel
24	R/W	0	Reserved
23	R/W	0	Bt656_2_clk_en
22-16	R/W	0	Bt656_2_clk_div
15-11	R/W	0	Reserved
10-9	R/W	0	Bt656_1_clk_sel 0:fclk_div2 1:fclk_div3 2:fclk_div5 3:fclk_div7
8	R/W	0	Reserved
7	R/W	0	Bt656_1_clk_en
6-0	R/W	0	Bt656_1_clk_div

HHI_SAR_CLK_CNTL0xf6

Bit(s)	R/W	Default	Description
31~11	R/W	0	Reserved
10-9	R/W	0	Clk_sel: 0:cts_oscin_clk 1:clk81
8	R/W	0	Clk_en
7-0	R/W	0	Clk_div

RESET_REGISTER 0xc11004404

Bit(s)	R/W	Default	Description
31-28	R/W	0	MIPI
27	R/W	0	mmc
26	R/W	0	Vcbus_clk81
25	R/W	0	Ahb_data
24	R/W	0	Ahb_cntl
23	R/W	0	Cbus_capb3
22	R/W	0	Sys_cpu_capb3
21	R/W	0	Dos_capb3
20	R/W	0	Mali_capb3
19	R/W	0	Hdmitx_capb3
18	R/W	0	Nand_capb3
17	R/W	0	Capb3_decode
16	R/W	0	Gic
15	R/W	0	Reserved
14	R/W	0	Reserved
13	R/W	0	vcbus
12	R/W	0	AFIFO2
11	R/W	0	ASSIST
10	R/W	0	VENC
9	R/W	0	PMUX
8	R/W	0	Reserved
7	R/W	0	Vid_pll_div
6	R/W	0	AIU
5	R/W	0	VIU
4	R/W	0	DCU_RESET
3	R/W	0	DDR_TOP
2	R/W	0	DOS_RESET
1	R/W	0	Reserved
0	R/W	0	HIU

RESET1_REGISTER 0xc11004408

Bit(s)	R/W	Default	Description
31-29	R/W	0	Reserved
28	R/W	0	Sys_cpu_mbist
27	R/W	0	Sys_cpu_p
26	R/W	0	Sys_cpu_l2
25	R/W	0	Sys_cpu_axi
24	R/W	0	Sys_pll_div
23-20	R/W	0	Sys_cpu_core[3:0]
19-16	R/W	0	Sys_cpu[3:0]
15	R/W	0	Rom_boot
14	R/W	0	Sd_emmc_c
13	R/W	0	Sd_emmc_b
12	R/W	0	Sd_emmc_a
11	R/W	0	Ethernet
10	R/W	0	ISA
9	R/W	0	BLKMV (NDMA)
8	R/W	0	PARSER
7	R/W	0	Reserved
6	R/W	0	AHB_SRAM
5	R/W	0	BT656
4	R/W	0	AO Reset
3	R/W	0	DDR
2	R/W	0	USB_OTG
1	R/W	0	DEMUX
0	R/W	0	Cppm

RESET2_REGISTER 0xc1100440c

Bit(s)	R/W	Default	Description
15	R/W	0	Hdmi system reset
14	R/W	0	MALI
13	R/W	0	AO CPU RESET
12	R/W	0	Reserved
11	R/W	0	Reserved
10	R/W	0	parser_top
9	R/W	0	parser_ctl
8	R/W	0	Paser_fetch
7	R/W	0	Parser_reg
6	R/W	0	GE2D
5	R/W	0	Reserved
4	R/W	0	Reserved
3	R/W	0	Reserved
2	R/W	0	HDMI_TX
1	R/W	0	AUDIN
0	R/W	0	VD_RMEM

RESET3_REGISTER 0xc11004410

Bit(s)	R/W	Default	Description
15	R/W	0	Demux reset 2
14	R/W	0	Demux reset 1
13	R/W	0	Demux reset 0
12	R/W	0	Demux S2P 1
11	R/W	0	Demux S2p 0
10	R/W	0	Demux DES
9	R/W	0	Demux top
8	R/W	0	Audio DAC
7	R/W	0	Reserved
6	R/W	0	AHB BRIDGE CNTL
5	R/W	0	tvfe
4	R/W	0	AIFIFO
3	R/W	0	Sys_cpu_bvci
2	R/W	0	EFUSE
1	R/W	0	SYS CPU
0	R/W	0	Ring oscillator

RESET4_REGISTER 0xc11004414

Bit(s)	R/W	Default	Description
15	R/W	0	I2C_Master 1
14	R/W	0	I2C_Master 2
13	R/W	0	VENCL
12	R/W	0	VDI6
11	R/W	0	Reserved
10	R/W	0	RTC
9	R/W	0	VDAC
8	R/W	0	Reserved
7	R/W	0	VENCP
6	R/W	0	VENCI
5	R/W	0	RDMA
4	R/W	0	DVIN_RESET
3	R/W	0	Reserved
2	R/W	0	Reserved
1	R/W	0	Reserved
0	R/W	0	Reserved
1	R/W	0	MISC PLL
0	R/W	0	DDR PLL

RESET6_REGISTER 0xc1100441c

Bit(s)	R/W	Default	Description
15	R/W	0	Uart_slip
14	R/W	0	PERIPHS: SDHC
13	R/W	0	PERIPHS: SPI 0
12	R/W	0	PERIPHS: Async 1
11	R/W	0	PERIPHS: Async 0
10	R/W	0	PERIPHS: UART 1, 2
9	R/W	0	PERIPHS: UART 0
8	R/W	0	PERIPHS: SDIO
7	R/W	0	PERIPHS: Stream Interface
6	R/W	0	
5	R/W	0	SANA
4	R/W	0	PERIPHS: I2C Master 0
3	R/W	0	PERIPHS: SAR ADC
2	R/W	0	PERIPHS: Smart Card
1	R/W	0	PERIPHS: SPICC
0	R/W	0	PERIPHS: General

RESET6_REGISTER 0xc11004420

Bit(s)	R/W	Default	Description
15	R/W	0	Reserved
14	R/W	0	Reserved
13	R/W	0	Reserved
12	R/W	0	Reserved
11	R/W	0	Reserved
10	R/W	0	Reserved
9	R/W	0	Reserved
8	R/W	0	A9_dmc_pipel
7	R/W	0	vid_lock
6	R/W	0	Reserved
5	R/W	0	Device_mmc_arb
4	R/W	0	Reserved
3~0	R/W	0	Usb_ddr[3:0]

RESET0_MASK 0xc11004440

The Bits of this register correspond to the RESET[n] REGISTERs above. If a bit is set in this register, then when the watchdog timer fires, that particular module will NOT be reset.

RESET1_MASK 0xc11004444

The Bits of this register correspond to the RESET[n] REGISTERs above. If a bit is set in this register, then when the watchdog timer fires, that particular module will NOT be reset.

RESET2_MASK 0xc11004448

The Bits of this register correspond to the RESET[n] REGISTERs above. If a bit is set in this register, then when the watchdog timer fires, that particular module will NOT be reset.

RESET3_MASK 0xc1100444c

The Bits of this register correspond to the RESET[n] REGISTERs above. If a bit is set in this register, then when the watchdog timer fires, that particular module will NOT be reset.

RESET4_MASK 0xc11004450

The Bits of this register correspond to the RESET[n] REGISTERs above. If a bit is set in this register, then when the watchdog timer fires, that particular module will NOT be reset.

RESET5_MASK 0xc11004454

The Bits of this register correspond to the RESET[n] REGISTERs above. If a bit is set in this register, then when the watchdog timer fires, that particular module will NOT be reset.

RESET6_MASK 0xc11004458

The Bits of this register correspond to the RESET[n] REGISTERs above. If a bit is set in this register, then when the watchdog timer fires, that particular module will NOT be reset.

RESET7_MASK 0xc1100445c

The Bits of this register correspond to the RESET[n] REGISTERs above. If a bit is set in this register, then when the watchdog timer fires, that particular module will NOT be reset.

RESET0_LEVEL 0xc11004480

The bits of this register correspond to the RESET[n] REGISTERs above. The default of this register is 0xFFFFFFFF. Setting any bit to 0, forces the corresponding RESET LOW. This registers allows the software to "Hold" a reset LOW (in a reset condition).

RESET1_LEVEL 0xc11004484

The bits of this register correspond to the RESET[n] REGISTERs above. The default of this register is 0xFFFFFFFF. Setting any bit to 0, forces the corresponding RESET LOW. This registers allows the software to "Hold" a reset LOW (in a reset condition).

RESET2_LEVEL 0xc11004488

The bits of this register correspond to the RESET[n] REGISTERs above. The default of this register is 0xFFFFFFFF. Setting any bit to 0, forces the corresponding RESET LOW. This registers allows the software to "Hold" a reset LOW (in a reset condition).

RESET3_LEVEL 0xc1100448c

The bits of this register correspond to the RESET[n] REGISTERs above. The default of this register is 0xFFFFFFFF. Setting any bit to 0, forces the corresponding RESET LOW. This registers allows the software to "Hold" a reset LOW (in a reset condition).

RESET4_LEVEL 0xc11004490

The bits of this register correspond to the RESET[n] REGISTERs above. The default of this register is 0xFFFFFFFF. Setting any bit to 0, forces the corresponding RESET LOW. This registers allows the software to "Hold" a reset LOW (in a reset condition).

RESET5_LEVEL 0xc11004494

The bits of this register correspond to the RESET[n] REGISTERs above. The default of this register is 0xFFFFFFFF. Setting any bit to 0, forces the corresponding RESET LOW. This registers allows the software to "Hold" a reset LOW (in a reset condition).

RESET6_LEVEL 0xc11004498

The bits of this register correspond to the RESET[n] REGISTERs above. The default of this register is 0xFFFFFFFF. Setting any bit to 0, forces the corresponding RESET LOW. This registers allows the software to "Hold" a reset LOW (in a reset condition).

RESET7_LEVEL 0xc1100449c

The bits of this register correspond to the RESET[n] REGISTERs above. The default of this register is 0xFFFFFFFF. Setting any bit to 0, forces the corresponding RESET LOW. This registers allows the software to "Hold" a reset LOW (in a reset condition).

AO_RTI_GEN_CTNL_REG0 0xC8100040

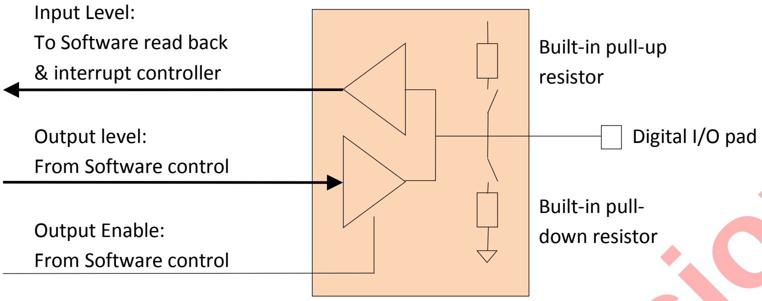
Bit(s)	R/W	Default	Description
31-24	R/W	0	Reserved
23	R/W	0	IR_BLASTER_RESET
22	R/W	0	UART2_RESET
21:20	R/W	0	Reserved
19	R/W	0	I2C_SLAVE_RESET: Set to 1 to reset
18	R/W	0	I2C_MASTER_RESET: Set to 1 to reset
17	R/W	0	UART_MODULE_RESET: Set to 1 to reset
16	R/W	0	IR_MODULE_RESET: Set to 1 to reset
15-8	R/W	0xFF	Reserved
7	R/W	1	Reserved
6	R/W	1	IR_BLASTER clock gate: 1 = clock enabled
5	R/W	1	UART2 module clock gate
4	R/W	1	Reserved
3	R/W	1	UART module clock gate
2	R/W	1	I2C slave module clock gate
1	R/W	1	I2C master module clock gate
0	R/W	1	IR remote decoder clock gate

22. GPIO

22.1 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 III.22.1. GPIO Structure



Each GPIO pin has an individual control bitthat can be used to set the output level, output enable of the I/O pad (0 = output, 1 = input). Additionally all digital I/O pads have built-in pull-up and pull-down resistors. The input from the digital I/O pad can be read back in software and is also connected to an interrupt controller. The interrupt controller allows up to 8 GPIO pins to be used as active high, active low, rising edge or falling edge interrupts.

Finally, the GPIO's are grouped into voltage domains. Some GPIOs (depending on the PCB design) can be configured to operate between 0 and 1.8v while others may operate between 0.0v and 3.3v.

22.2 GPIO Multiplex Function

The GPIO multiplex functions are shown in the sections below, where the RegNN[MM] corresponds to CBUS registers defined in Table III.22.1

Table III.22.1 Pin Mux Registers

Pin Mux Registers	Abbreviation	Offset	Default Value after power-on/Reset
PERIPHS_PIN_MUX_0	REG0	0xc88344b0	0x0000
PERIPHS_PIN_MUX_1	REG1	0xc88344b4	0x0000
PERIPHS_PIN_MUX_2	REG2	0xc88344b8	0x0000
PERIPHS_PIN_MUX_3	REG3	0xc88344bc	0x0000
PERIPHS_PIN_MUX_4	REG4	0xc88344c0	0x0000
PERIPHS_PIN_MUX_5	REG5	0xc88344c4	0x0000
PERIPHS_PIN_MUX_6	REG6	0xc88344c8	0x0000
PERIPHS_PIN_MUX_7	REG7	0xc88344cc	0x0000
PERIPHS_PIN_MUX_8	REG8	0xc88344d0	0x0000
PERIPHS_PIN_MUX_9	REG9	0xc88344d4	0x0000
AO_RTI_PIN_MUX_REG	AO_REG	0xc8100014	0x0000
AO_RTI_PIN_MUX_REG2	AO_REG2	0xc8100018	0x0000

Table III.22.2 GPIO Bank X Pin Multiplexing Table

Package Name	Func1	Func2	Func3	Func4
GPIOX_0	SDIO_D0 reg5[31]			
GPIOX_1	SDIO_D1 reg5[30]			
GPIOX_2	SDIO_D2 reg5[29]			

Package Name	Func1	Func2	Func3	Func4
GPIOX_3	SDIO_D3 reg5[28]			
GPIOX_4	SDIO_CLK reg5[27]			
GPIOX_5	SDIO_CMD reg5[26]			
GPIOX_6	PWM_A reg5[25]			
GPIOX_7	SDIO_IRQ reg5[24]	PWM_F reg5[14]		
GPIOX_8	PCM_OUT_A reg5[23]	UART_TX_C reg5[13]		SPI_MOSI reg5[3]
GPIOX_9	PCM_IN_A reg5[22]	UART_RX_C reg5[12]		SPI_MISO reg5[2]
GPIOX_10	PCM_FS_A reg5[21]	UART_CTS_C reg5[11]	I2C_SDA_D reg5[5]	SPI_SS0 reg5[1]
GPIOX_11	PCM_CLK_A reg5[20]	UART_RTS_C reg5[10]	I2C_SCK_D reg5[4]	SPI_SCLK reg5[0]
GPIOX_12	UART_TX_A (long fifo) reg5[19]			
GPIOX_13	UART_RX_A reg5[18]			
GPIOX_14	UART_CTS_A reg5[17]			
GPIOX_15	UART_RTS_A reg5[16]			
GPIOX_16	PWM_E reg5[15]			
GPIOX_17	GPIO (BT_EN)			
GPIOX_18	GPIO (BT_WAKE_CPU)			

Table III.22.3 GPIO Bank DV Pin Multiplexing Table

Package Name	Func1	Func2	Func3
GPIODV_24	UART_TX_B reg2[16]	DMIC_IN reg2[7]	I2C_SDA_A reg1[15]
GPIODV_25	UART_RX_B reg2[15]	DMIC_CLK_OUT reg2[6]	I2C_SCK_A reg1[14]
GPIODV_26	UART_CTS_B reg2[14]		I2C_SDA_B reg1[13]
GPIODV_27	UART_RTS_B reg2[13]		I2C_SCK_B reg1[12]
GPIODV_28	PWM_D reg2[12]	REMOTE_INPUT reg1[9]	I2C_SDA_C reg1[11]
GPIODV_29	PWM_B reg2[11]	PWM_VS reg2[5]	I2C_SCK_C reg1[10]

Table III.22.4 GPIO Bank H Pin Multiplexing Table

Package Name	Func1	Func2	Func3	Func4
GPIOH_0	HDMI_HPD reg6[31]			
GPIOH_1	HDMI_SDA reg6[30]			
GPIOH_2	HDMI_SCL reg6[29]			
GPIOH_3	GPIO(5V_EN)			

Package Name	Func1	Func2	Func3	Func4
GPIOH_4	SPDIF_OUT reg6[28]	SPDIF_IN reg6[27]		
GPIOH_5				
GPIOH_6		JTAG_TCK	I2S_AM_CLK reg6[26]	
GPIOH_7		JTAG_TMS	I2S_AO_CLK_OUT reg6[25]	I2S_AO_CLK_IN reg6[22]
GPIOH_8		JTAG_TDI	I2S_LR_CLK_OUT reg6[24]	I2S_LR_CLK_IN reg6[21]
GPIOH_9		JTAG_TDO	I2SOUT_CH01 reg6[23]	

Table III.22.5 GPIO Bank CLK Pin Multiplexing Table

Package Name	Func1	Func2
GPIOCLK_0	CLK24	CLK12(wifi)
GPIOCLK_1	CLK25	pwm_F

Table III.22.6 GPIO Bank AO Pin Multiplexing Table

Pin Name	Func1	Func2	Func3	Func4
GPIOAO_0	UART_TX_AO_A ao_reg[12]	UART_TX_AO_B ao_reg[26]		
GPIOAO_1	UART_RX_AO_A ao_reg[11]	UART_RX_AO_B ao_reg[25]		
GPIOAO_2	UART_CTS_AO_A ao_reg[10]	UART_CTS_AO_B ao_reg[8]		
GPIOAO_3	UART_RTS_AO_A ao_reg[9]	UART_RTS_AO_B ao_reg[7]		PWM_AO_A ao_reg[22]
GPIOAO_4	UART_TX_AO_B ao_reg[24]	I2C_SCK_AO ao_reg[6]	I2C_SLAVE_SCK_AO ao_reg[2]	
GPIOAO_5	UART_RX_AO_B ao_reg[23]	I2C_SDA_AO ao_reg[5]	I2C_SLAVE_SDA_AO ao_reg[1]	
GPIOAO_6		I2S_IN_01 default	SPDIF_OUT ao_reg[16]	PWM_AO_B ao_reg[18]
GPIOAO_7	REMOTE_INPUT ao_reg[0]	REMOTE_OUTPUT ao_reg[21]		
GPIOAO_8	AO_CEC ao_reg[15]	EE_CEC ao_reg[14]	I2SOUT_CH23 ao_reg2[0]	PWM_AO_A ao_reg[17]
GPIOAO_9	REMOTE_OUTPUT ao_reg[31]	SPDIF_OUT ao_reg[4]	I2SOUT_CH45 ao_reg2[1]	PWM_AO_B ao_reg[3]
TEST_N	TEST_N	wd_gpio ao_reg[13:20]	I2SOUT_CH67 ao_reg2[2]	
RESET_N	RESET_N			

Table III.22.7 GPIO Bank BOOT Pin Multiplexing Table

Pin Name	Func1	Func2	Func3
BOOT_0	EMMC_NAND_D0 reg7[31]		
BOOT_1	EMMC_NAND_D1 reg7[31]		
BOOT_2	EMMC_NAND_D2 reg7[31]		
BOOT_3	EMMC_NAND_D3 reg7[31]		
BOOT_4	EMMC_NAND_D4 reg7[31]		

Pin Name	Func1	Func2	Func3
BOOT_5	EMMC_NAND_D5 reg7[31]		
BOOT_6	EMMC_NAND_D6 reg7[31]		
BOOT_7	EMMC_NAND_D7 reg7[31]		
BOOT_8	EMMC_CLK reg7[30]	NAND_CE0 reg7[7]	
BOOT_9		NAND_CE1 reg7[6]	
BOOT_10	EMMC_CMD reg7[29]	NAND_RB0 reg7[5]	
BOOT_11		NAND_ALE reg7[4]	NOR_D reg7[13]
BOOT_12		NAND_CLE reg7[3]	NOR_Q reg7[12]
BOOT_13		NAND_WEN_CLK reg7[2]	NOR_C reg7[11]
BOOT_14		NAND_REN_WR reg7[1]	

Table III.22.8 GPIO Bank CARD Pin Multiplexing Table

Pin Name	Func1	Func2	Func3	Func4
CARD_0	SDCARD_D1 reg6[5]			JTAG_TDI
CARD_1	SDCARD_D0 reg6[4]			JTAG_TDO
CARD_2	SDCARD_CLK reg6[3]			JTAG_CLK
CARD_3	SDCARD_CMD reg6[2]			JTAG_TMS
CARD_4	SDCARD_D3 reg6[1]	UART_TX_AO_A reg6[9]	UART_RX_AO_A reg6[11]	
CARD_5	SDCARD_D2 reg6[0]	UART_RX_AO_A reg6[8]	UART_TX_AO_A reg6[10]	
CARD_6	GPIO(Card_DET / EN)			

Table III.22.9 GPIOZ_x Multi-Function Pin

Pin Name	Func1	Func2
GPIOZ_14	SPDIF_IN reg3[21]	ETH_LINK_LED reg4[25]
GPIOZ_15	PWM_C reg3[20]	ETH_ACT_LED reg4[24]

22.3 GPIO Interrupt

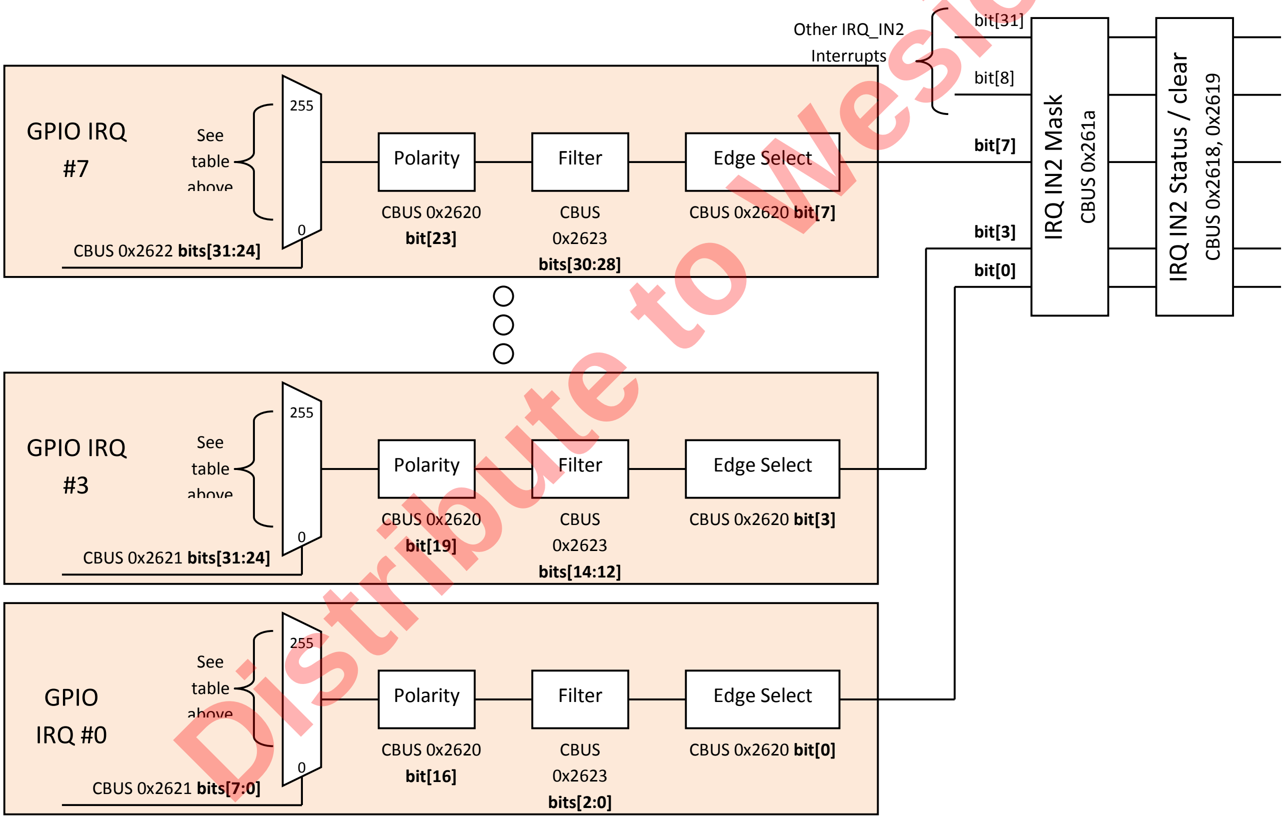
There are 8 independent filtered GPIO interrupt modules that can be programmed to use any of the GPIOs in the chip as an interrupt source (listed in the table below). For example, to select gpioD_11 as the source for GPIO IRQ #0, then CBUS 0x2609 bits[7:0] = 27 (according to the table below).

Table III.22.9 GPIO Interrupt Sources

Input Mux Location CBUS registers 0x2621 and 0x2622	Description
255~110	Undefined (no interrupt)
109:108	gpioCLK[1:0]
107:89	gpioX[18:0]
88:83	gpioDV[29:24]
58:52	Card[6:0]
51:36	Boot[15:0]
35:26	gpioH[9:0]
25:10	gpioZ[15:0]
9 :0	gpioAO[9:0]

The diagram below illustrates the path a GPIO takes to become an interrupt. The eight GPIO interrupts respond to the MASK, STATUS and STATUS/CLEAR registers just like any other interrupt in the chip. The difference for the GPIO interrupts is that they can be filtered and conditioned.

Fig III.22.2 GPIO Interrupt Path



NOTE: The input for the GPIO interrupt module (the input into the 256:1 mux) comes directly from the I/O pad of the chip. Therefore if a pad (say gpioA_13) is configured as a UART TX pin, then in theory, the UART TX pin can be a GPIO interrupt since the TX pin will drive gpioA_13 which in turn can drive the GPIO interrupt module.

22.4 Register Description

Table III.23.11 and Table III.23.12 shows the information of GPIO related registers.

Table III.22.10 GPIO Register Infromation I

Final address = 0xc8834400 + address * 4

Package Name	OEN (Read Write)	OUT (Write Only)	IN (Read Only)	IO Type
GPIOX_18~0	0x18 Bit[18:0]	0x19 Bit[18:0]	0x1a Bit[18:0]	Tri-State
GPIODV_29~24	0x0C Bit[29:24]	0x0D Bit[29:24]	0x0E Bit[29:24]	Tri-State
GPIOH_9~0	0x0F Bit[29:20]	0x10 Bit[29:20]	0x11 Bit[29:20]	Tri-State
BOOT_15~0	0x12 Bit[15:0]	0x13 Bit[15:0]	0x14 Bit[15:0]	Tri-State
CARD_6~0	0x12 Bit[26:20]	0x13 Bit[26:20]	0x14 Bit[26:20]	Tri-State
CLK_1~0	0x15 Bit[29:28]	0x16 Bit[29:28]	0x17 Bit[29:28]	Tri-State
GPIOZ_15~14	0x15 Bit[15:14]	0x16 Bit[15:14]	0x17 Bit[15:14]	Tri-State

Table III.22.11 GPIO Register Information II

Final address = 0xc8100000 + address * 4

Package Name	OEN (Read Write)	OUT (Write Only)	IN (Read Only)	IO Type
GPIOAO_9~0	0x09 Bit[9:0]	0x09 Bit[25:16]	0x0a Bit[9:0]	Tri-State

Pad pull-up/down Direction

The I/O pads contain both a pull-up and a pull-down. If a bit is set to 1 in the registers below, then the pull-up is enabled. If a bit is set to 0, then the pull-down is enabled.

NOTE: There are separate pull-up "enables" that must also be set to 1 in order for the pull-up/down direction to function. If an "enable" is set to 0, then the pull-up/down feature is disabled and the bits below are ignored (on a per pad basis).

PULL_UP_REG0 0x3a

Bit(s)	R/W	Default	Description
31~30	R/W		Unused
29~0	R/W		gpioDV[29:0] 1 = pull up. 0 = pull down

PULL_UP_REG1 0x3b

Bit(s)	R/W	Default	Description
31~30	R/W		Unused
29~20	R/W		gpioH[9:0] 1 = pull up. 0 = pull down
19~0	R/W		Reserved

PULL_UP_REG2 0x3c

Bit(s)	R/W	Default	Description
31~27	R/W		Reserved
26~20	R/W		card[6:0] 1 = pull up. 0 = pull down
19~16	R/W		Reserved
15~0	R/W		boot[15:0] 1 = pull up. 0 = pull down

PULL_UP_REG3 0x3d

Bit(s)	R/W	Default	Description
29~28	R/W		gpioCLK[1:0] 1= pull up. 0=pull down
27~16	R/W		reserved
15~0	R/W		gpioZ[15:0] 1 = pull up. 0 = pull down

PULL_UP_REG4 0x3e

Bit(s)	R/W	Default	Description
31~19	R/W		Unused
18~0	R/W		gpioX[18:0] 1 = pull up. 0 = pull down

Pad Pull-Up/Down Enables

Each I/O pad has a selectable pull-up or pull-down resistor. In order for the pull-up direction (up or down) to be operational, the appropriate bit below must be set in order to enable the pull-up/down function.

PULL_UP_EN_REG0 0x48

Bit(s)	R/W	Default	Description
31~30	R/W	0	Unused
29~0	R/W	0x0	gpioDV[29:0] pullup-enable: 1 = pullup or pull-down enabled. 0 = no-pull-up or pull-down

PULL_UP_EN_REG1 0x49

Bit(s)	R/W	Default	Description
31~30	R/W	0	Unused
29~20	R/W	0x0	gpioH[9:0] pullup-enable: 1 = pullup or pull-down enabled. 0 = no-pull-up or pull-down
19~0	R/W	0	Reserved

PULL_UP_EN_REG2 0x4a

Bit(s)	R/W	Default	Description
31~27	R/W	0	Reserved
26~20	R/W	0x1FF	card[6:0] pullup-enable: 1 = pullup or pull-down enabled. 0 = no-pull-up or pull-down
19~16	R/W	1	Reserved
15~0	R/W	0x3FFFF	boot[15:0] pullup-enable: 1 = pullup or pull-down enabled. 0 = no-pull-up or pull-down

PULL_UP_EN_REG3 0x4b

Bit(s)	R/W	Default	Description
29~28	R/W	0xF	gpioCLK[1:0] pullup-enable: 1 = pullup or pull-down enabled. 0 = no-pull-up or pull-down
27~16	R/W	0	reserved
15~0	R/W	0x3FFF	gpioZ[15:0] pullup-enable: 1 = pullup or pull-down enabled. 0 = no-pull-up or pull-down

PULL_UP_EN_REG4 0x4c

Bit(s)	R/W	Default	Description
31~19	R/W	0	Unused
18~0	R/W	0x6FFFBF	gpioX[18:0] pullup-enable: 1 = pullup or pull-down enabled. 0 = no-pull-up or pull-down

23. Interrupt Control

23.1 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 V2.0.

23.2 Interrupt Source

There are 224 interrupt sources in the chip. All of the interrupts are connected to the integrated GIC in Cortex-A53 while the AO-CPU see a sub-set of the interrupts. The control Bits of AO-CPU interrupt are listed in the following table.

Table III.23.1 EE Interrupt Source

A53 GIC Bit	Interrupt sources	Description
255	1'b0	unused
254	1'b0	unused
253	1'b0	unused
252	1'b0	unused
251	1'b0	unused
250	sd_emmc_C_irq	
249	sd_emmc_B_irq	
248	sd_emmc_A_irq	
247	m_i2c_2_irq	
246	m_i2c_1_irq	
245	mbox_irq_send5	
244	mbox_irq_send4	
243	mbox_irq_send3	
242	mbox_irq_receiv2	
241	mbox_irq_receiv1	
240	mbox_irq_receiv0	
239	1'b0	unused
238	ao_timerA_irq	
237	1'b0	unused
236	ao_watchdog_irq	
235	ao_jtag_pwd_fast_irq	
234	1'b0	unused
233	ao_gpio_irq1	
232	ao_gpio_irq0	
231	ao_cec_irq	
230	ao_ir_blaster_irq	
229	ao_uart2_irq	
228	ao_ir_dec_irq	
227	ao_i2c_m_irq	
226	ao_i2c_s_irq	
225	ao_uart_irq	
224	1'b0	unused
223	dma_irq[3]	
222	dma_irq[2]	
221	dma_irq[1]	

A53 GIC Bit	Interrupt sources	Description
220	dma_irq[0]	
219	1'b0	unused
218	vp9dec_irq	unused
217	1'b0	unused
216	1'b0	unused
215	1'b0	unused
214	1'b0	unused
213	1'b0	unused
212	1'b0	unused
201	mali_irq_ppmmu2	
200	mali_irq_pp2	
199	mali_irq_ppmmu1	
198	mali_irq_pp1	
197	mali_irq_ppmmu0	
196	mali_irq_pp0	
195	mali_irq_pmu	
194	mali_irq_pp	
193	mali_irq_gpmmu	
192	mali_irq_gp	
184	viu1_line_n_irq	
183	1'b0	
182	ge2d_irq	
181	cusad_irq	
180	asssit_mbox_irq3	
179	asssit_mbox_irq2	
178	asssit_mbox_irq1	
177	asssit_mbox_irq0	
176	det3d_int	
175	viu1_wm_int	
173	A53irq[4]	EXTERRIRQ_a
172	A53irq[3]	|CTIIRQ[3:0]
171	A53irq[2]	|VCPUMNTIRQ_a[3:0]
170	A53irq[1]	|COMMIRQ_a[3:0]
169	A53irq[0]	|PMUIRQ_a[3:0]
168	1'b0	
167	1'b0	
166	1'b0	
165	1'b0	
164	1'b0	
127	1'b0	unused
126	uart3_slip_irq	UART slip
125	uart2_irq	UART 2
124	1'b0	unused
123	1'b0	unused
122	1'b0	unused
121	rdma_done_int	RDMA

A53 GIC Bit	Interrupt sources	Description
120	i2s_cbus_ddr_irq	Audio I2S CBUS IRQ
119	1'b0	unused
118	vid1_wr_irq	
117	vdin1_vsync_int	
116	vdin1_hsync_int	
115	vdin0_vsync_int	
114	vdin0_hsync_int	
113	spi2_int	
112	spi_int	
111	vid0_wr_irq	
110	1'b0	unused
109	1'b0	unused
108	1'b0	unused
107	uart1_irq	
106	1'b0	unused
105	sar_adc_irq	SAR ADC
104	1'b0	unused
103	gpio_irq[7]	GPIO Interrupt
102	gpio_irq[6]	GPIO Interrupt
101	gpio_irq[5]	GPIO Interrupt
100	gpio_irq[4]	GPIO Interrupt
99	gpio_irq[3]	GPIO Interrupt
98	gpio_irq[2]	GPIO Interrupt
97	gpio_irq[1]	GPIO Interrupt
96	gpio_irq[0]	GPIO Interrupt
95	TimerI	TimerI
94	TimerH	TimerH
93	TimerG	TimerG
92	TimerF	TimerF
91	1'b0	unused
90	hdcp22_irq	
89	hdmi_tx_interrupt	
88	1'b0	unused
87	hdmi_cec_interrupt	
86	dmc_test_irq	
85	demux_int_2	
84	dmc_irq	
83	dmc_sec_irq	
82	ai_iec958_int	IEC958 interrupt
81	iec958_ddr_irq	IEC958 DDR interrupt
80	i2s_irq	I2S DDR Interrupt
79	crc_done	From AIU CRC done
78	deint_irq	Reserved for Deinterlacer
77	dos_mbox_slow_irq[2]	DOS Mailbox 2
76	dos_mbox_slow_irq[1]	DOS Mailbox 1
75	dos_mbox_slow_irq[0]	DOS Mailbox 0

A53 GIC Bit	Interrupt sources	Description
74	1'b0	unused
73	1'b0	unused
72	1'b0	unused
71	m_i2c_3_irq	I2C Master #3
70	1'b0	unused
69	smartcard_irq	
67	spdif_irq	
66	nand_irq	
65	viff_empty_int_cpu	
64	parser_int_cpu	
63	U2d_interrupt	USB
62	U3h_interrupt	USB
61	Timer D	Timer D
60	bus_mon1_fast_irq	
59	bus_mon0_fast_irq	
58	uart0_irq	
57	async_fifo2_flush_irq	
56	async_fifo2_fill_irq	
55	demux_int	
54	encif_irq	
53	m_i2c_0_irq	
52	bt656_irq	
51	async_fifo_flush_irq	
50	async_fifo_fill_irq	
49	bt656_2_rq	bt656_B
48	usb_iddig_irq	
47	1'b0	unused
46	eth_lip_intro_o	
45	1'b0	unused
44	1'b0	unused
43	Timer B	Timer B
42	Timer A	Timer A
41	eth_phy_irq	unused
40	eth_gmac_int	
39	audin_irq	
38	Timer C	Timer C
37	demux_int_1	
36	eth_pmt_intr_o	
35	viu1_vsync_int	VSYNC
34	viu1_hsync_int	HSYNC
33	1'b0	HIU Mailbox
32	ee_wd_irq	Watchdog Timer

23.3 Register Description

Each register final address = 0xC1100000 + address * 4

GPIO Interrupt EDGE and Polarity: 0x2620

This register controls the polarity of the GPIO interrupts and whether or not the interrupts are level or edge triggered. There are 8 GPIO interrupts. These 8 GPIO interrupts can be assigned to any one of up to 256 pins on the chip.

Bit(s)	R/W	Default	Description
31-24	R	0	unused
23			GPIO_POLARITY_PATH_7: If a bitin this field is 1, then the GPIO signal for GPIO interrupt path 7 is inverted.
22			GPIO_POLARITY_PATH_6:
21			GPIO_POLARITY_PATH_5:
20			GPIO_POLARITY_PATH_4:
19			GPIO_POLARITY_PATH_3:
18			GPIO_POLARITY_PATH_2:
17			GPIO_POLARITY_PATH_1:
16	R/W	0	GPIO_POLARITY_PATH_0:
15-8	R	0	Unused
7	R/W		GPIO_EDGE_SEL_PATH_7: If a bit is set to 1, then the GPIO interrupt for GPIO path 7 is configured to be an edge generated interrupt. If the polarity (above) is 0, then the interrupt is generated on the rising edge. If the polarity is 1, then the interrupt is generated on the falling edge of the GPIO. If a bit is this field is 0, then the GPIO is a level interrupt.
6	R/W		GPIO_EDGE_SEL_PATH_6
5	R/W		GPIO_EDGE_SEL_PATH_5
4	R/W		GPIO_EDGE_SEL_PATH_4
3	R/W		GPIO_EDGE_SEL_PATH_3
2	R/W		GPIO_EDGE_SEL_PATH_2
1	R/W		GPIO_EDGE_SEL_PATH_1
0	R/W	0	GPIO_EDGE_SEL_PATH_0

GPIO 0 ~ 3 Pin Select: 0x2621

Each GPIO interrupt can select from any number of up to 256 GPIO pins on the chip. The Bits below control the pin selection for GPIO interrupts 0 ~3.

Bit(s)	R/W	Default	Description
31-24	R/W	0	GPIO_PIN_SEL3: This value select which of up to 256 pins on the chip can be mapped to GPIO interrupt 3
23-16	R/W	0	GPIO_PIN_SEL2: This value select which of up to 256 pins on the chip can be mapped to GPIO interrupt 2
15-8	R/W	0	GPIO_PIN_SEL1: This value select which of up to 256 pins on the chip can be mapped to GPIO interrupt 1
7-0	R/W	0	GPIO_PIN_SEL0: This value select which of up to 256 pins on the chip can be mapped to GPIO interrupt 0

GPIO 4 ~ 7 Pin Select: 0x2622

Bit(s)	R/W	Default	Description
31-24	R/W	0	GPIO_PIN_SEL7: This value select which of up to 256 pins on the chip can be mapped to GPIO interrupt 7
23-16	R/W	0	GPIO_PIN_SEL6: This value select which of up to 256 pins on the chip can be mapped to GPIO interrupt 6
15-8	R/W	0	GPIO_PIN_SEL5: This value select which of up to 256 pins on the chip can be mapped to GPIO interrupt 5
7-0	R/W	0	GPIO_PIN_SEL4: This value select which of up to 256 pins on the chip can be mapped to GPIO interrupt 4

GPIO Filter Select (interrupts 0~7): 0x2623

Bit(s)	R/W	Default	Description
31	R/W	0	unused
30-28	R/W	0	FILTER_SEL7: (see FILTER_SEL0)
27	R/W	0	Unused
26-24	R/W	0	FILTER_SEL6: (see FILTER_SEL0)
23	R/W	0	Unused

Bit(s)	R/W	Default	Description
22-20	R/W	0	FILTER_SEL5: (see FILTER_SEL0)
19	R/W	0	Unused
18-16	R/W	0	FILTER_SEL4: (see FILTER_SEL0)
15	R/W	0	Unused
14-12	R/W	0	FILTER_SEL3: (see FILTER_SEL0)
11	R/W	0	Unused
10-8	R/W	0	FILTER_SEL2: (see FILTER_SEL0)
7	R/W	0	Unused
6-4	R/W	0	FILTER_SEL1: (see FILTER_SEL0)
3	R/W	0	unused
2-0	R/W	0	FILTER_SEL0: This value sets the filter selection for GPIO interrupt 0. A value of 0 = no filtering. A value of 7 corresponds to 7 x 3 x (111nS) of filtering.

24. TIMER

24.1 Overview

The SOC contains 11 general purpose timers and 2 watchdog timers.

24.2 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.

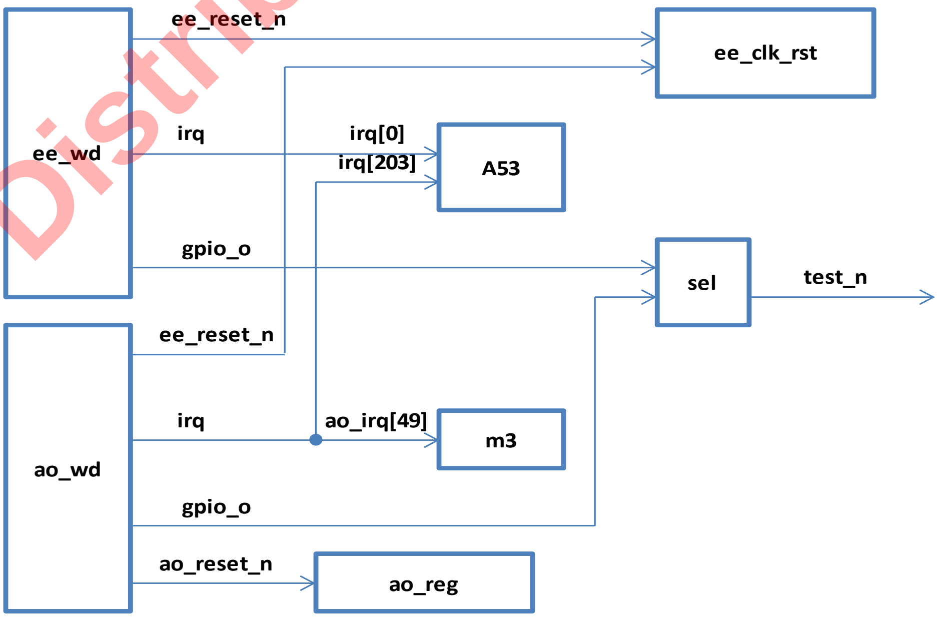
Table III. 24.1. General-Purpose Timer

Timer	Timebase Options	Counter size	Comment
Timer A	1uS, 10uS, 100uS, 1mS	16-bits	The 16-bit counter allows the timer to generate interrupts as infrequent as every 65.535 Seconds
Timer B	1uS, 10uS, 100uS, 1mS	16-bits	
Timer C	1uS, 10uS, 100uS, 1mS	16-bits	
Timer D	1uS, 10uS, 100uS, 1mS	16-bits	
Timer E	System clock, 1uS, 10uS, 100uS, 1mS	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, 1mS	16-bits	
Timer G	1uS, 10uS, 100uS, 1mS	16-bits	
Timer H	1uS, 10uS, 100uS, 1mS	16-bits	
Timer I	1uS, 10uS, 100uS, 1mS	16-bits	
Timer A-AO	System clock, 1uS, 10uS, 100uS	16-bits	Used in the Always On domain to generate interrupts for the AO-CPU
Timer E-AO	System clock	32-bits	This Always On counter doesn't generate an interrupt. Instead it simply counts up from 0 to 0xFFFFFFFF. The counter can be written at any time to reset the value to 0.

24.3 Watchdog Timer

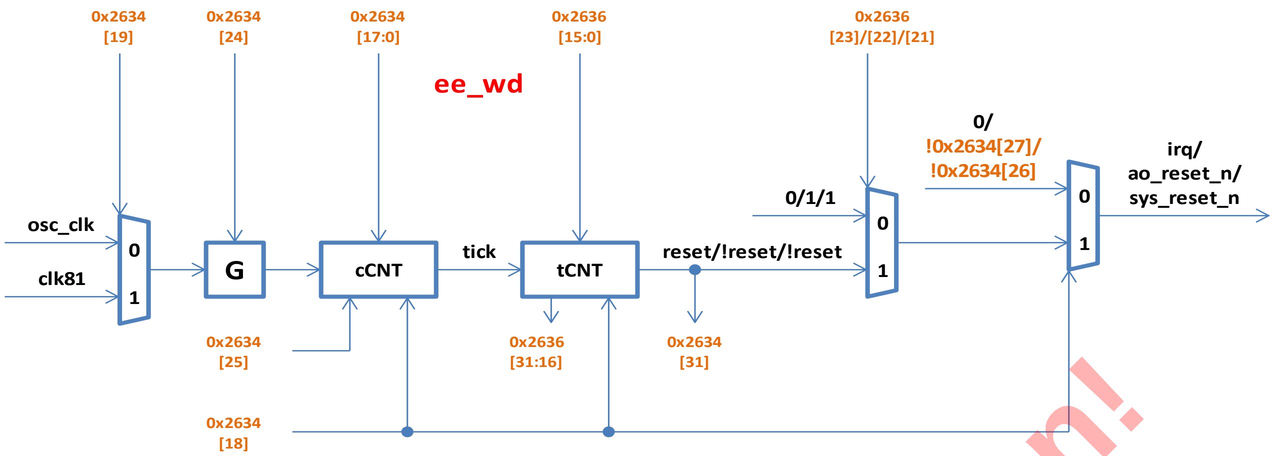
There are also two watchdog timers, one in AO and the other in EE domain, illustrated as following:

Figure 24.1 Watchdog Timer



AO domain watchdog timer and EE domain watchdog timer have the same design, as illustrated in the folloing figure:

Figure 24.2 EE domain Watchdog Timer degisn



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 (A53). 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 (the A53) 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.

24.4 Register Definitions

Each register final address = 0xC1100000 + address * 4

ISA_TIMER_MUX 0x2650

Bit(s)	R/W	Default	Description
31-20	R	0	unused
19	R/W	1	TIMERD_ EN: Set to 1 to enable Timer D
18	R/W	1	TIMERC_ EN: Set to 1 to enable Timer C
17	R/W	1	TIMERB_ EN: Set to 1 to enable Timer B

Bit(s)	R/W	Default	Description
16	R/W	1	TIMERA_EN: Set to 1 to enable Timer A
15	R/W	0	TIMERD_MODE: If This bit is set to 1, then timerD is a periodic . 0 = one-shot timer
14	R/W	0	TIMERC_MODE: If This bit is set to 1, then timerC is a periodic . 0 = one-shot timer
13	R/W	1	TIMERB_MODE: If This bit is set to 1, then timerB is a periodic . 0 = one-shot timer
12	R/W	1	TIMERA_MODE: If This bit is set to 1, then timerA is a periodic . 0 = one-shot timer
11	R	0	unused
10-8	R/W	0x1	TIMER E input clock selection: 000: System clock 001: 1uS Timebase resolution 010: 10uS Timebase resolution 011: 100uS Timebase resolution 100: 1mS timebase NOTE: The mux selection for Timer E is different from timer A, B, C and D
7-6	R/W	0x0	TIMER D input clock selection: See TIMER A below
5-4	R/W	0x0	TIMER C input clock selection: See TIMER A below
2-3	R/W	0x0	TIMER B input clock selection: See TIMER A below
1-0	R/W	0x0	TIMER A Input clock selection: These Bits select the input timebase for the counters for TimerA 00: 1uS Timebase resolution 01: 10uS Timebase resolution 10: 100uS Timebase resolution 11: 1mS Timebase resolution

ISA_TIMERA 0x2651

Timer A is a 16 bit count DOWN counter driven by the clock selected in register 0x01000530. TIMER A will count down from some value to zero, generate an interrupt and then re-load the original start count value. This timer can be used to generate a periodic interrupt (e.g. interrupt every 22 uS).

Bit(s)	R/W	Default	Description
31-16	R	-	Current Count value
15-0	R/W	0x0	Starting count value. Write this value to start TIMER A.

ISA_TIMERB 0x2652

Timer B is just like Timer A.

Bit(s)	R/W	Default	Description
31-16	R	-	Current Count value
15-0	R/W	0x0	Starting count value. Write this value to start TIMER B

ISA_TIMERC 0x2653

Timer C is just like Timer A.

Bit(s)	R/W	Default	Description
31-16	R	0	unused
15-0	R/W	0x0	Starting count value. Write this value to start TIMER C

ISA_TIMERD 0x2654

Timer D is identical to Timer A.

Bit(s)	R/W	Default	Description
31-16	R	0	unused
15-0	R/W	0x0	Starting count value. Write this value to start TIMER D

ISA_TIMERE 0x2662

Timer E is simply a32-bit counter that increments at a rate set by register 0x2650. To reset the counter to zero, simply write this register with any value. The value below is a read-only value that reflects the current count of the internal counter. This register can be used by software to simply provide a polling delay loop based on a programmable timebase.

Bit(s)	R/W	Default	Description
31-0	R	-	Current value of Timer E. Write this register with any value to clear the counter.

ISA_TIMERE_HI 0x2663

Bit(s)	R/W	Default	Description
31-0	R	-	Current value of Timer E[63:32]. Need read ISA_TIMERE first.

ISA_TIMER_MUX1 0x2664

Bit(s)	R/W	Default	Description
31-20	R	0	unused
19	R/W	1	TIMERD_ EN: Set to 1 to enable Timer I
18	R/W	1	TIMERC_ EN: Set to 1 to enable Timer H
17	R/W	1	TIMERB_ EN: Set to 1 to enable Timer G
16	R/W	1	TIMERA_EN: Set to 1 to enable Timer F
15	R/W	0	TIMERD_MODE: If This bit is set to 1, then timerD is a periodic . 0 = one-shot timer
14	R/W	0	TIMERC_MODE: If This bit is set to 1, then timerC is a periodic . 0 = one-shot timer
13	R/W	1	TIMERB_MODE: If This bit is set to 1, then timerB is a periodic. 0 = one-shot timer
12	R/W	1	TIMERA_MODE: If This bit is set to 1, then timerA is a periodic. 0 = one-shot timer
11	R	0	unused
10-8	R/W	0x1	TIMER E input clock selection: 000: System clock 001: 1uS Timebase resolution 010: 10uS Timebase resolution 011: 100uS Timebase resolution 100: 1mS timebase NOTE: The mux selection for Timer E is different from timer A, B, C and D
7-6	R/W	0x0	TIMER D input clock selection: See TIMER A below
5-4	R/W	0x0	TIMER C input clock selection: See TIMER A below
2-3	R/W	0x0	TIMER B input clock selection: See TIMER A below
1-0	R/W	0x0	TIMER A Input clock selection: These Bits select the input timebase for the counters for TimerA 00: 1uS Timebase resolution 01: 10uS Timebase resolution 10: 100uS Timebase resolution 11: 1mS Timebase resolution

ISA_TIMERF0x2665

Timer F is a 16 bit count DOWN counter driven by the clock selected in register 0x01000530. TIMER A will count down from some value to zero, generate an interrupt and then re-load the original start count value. This timer can be used to generate a periodic interrupt (e.g. interrupt every 22 uS).

Bit(s)	R/W	Default	Description
31-16	R	-	Current Count value
15-0	R/W	0x0	Starting count value. Write this value to start TIMER A.

ISA_TIMERG 0x2666

Timer G is just like Timer F.

Bit(s)	R/W	Default	Description
31-16	R	-	Current Count value
15-0	R/W	0x0	Starting count value. Write this value to start TIMER B

ISA_TIMER_H 0x2667

Timer H is just like Timer F.

Bit(s)	R/W	Default	Description
31-16	R	0	unused
15-0	R/W	0x0	Starting count value. Write this value to start TIMER C

WATCHDOG_CNTL 0x2634

Bit(s)	R/W	Default	Description
31	R	0	Watchdog_reset
30-28	R/W	0	Reserved
27	R/W	0	Ao_reset_n_now, if watchdog_en =0, output ao_reset_n = ! ao_reset_n_now
26	R/W	0	Sys_reset_n_now, if watch_dog_en = 1, output sys_reset_n = !sys_reset_n_now.
25	R/W	0	Clk_div_en: 0: no tick; 1: generate tick;

Bit(s)	R/W	Default	Description
24	R/W	0	Clk_en: 0: no clk; 1: clk work;
23	R/W	0	Interrupt_en: 0: no irq out; 1: irq = watchdog_reset;
22	R/W	0	Ao_reset_n_en: 0: output ao_reset_n = 1; 1: output ao_reset_n = ! watchdog_reset;
21	R/W	0	Sys_reset_n_en 0: output sys_reset_n = 1; 1: output sys_reset_n = ! watchdog_reset;
20	R/W	0	Reserved
19	R/W	0	Clk_sel: 0:osc_clk; 1:clk_81
18	R/W	0	Watchdog_en 0: no watchdog reset 1: gen watchdog reset
17-0	R/W	0	Clk_div_tcnt, when clk_div_en is 1, generate a tick each tcnt clock

WATCHDOG_CNTL1 0x2635

Bit(s)	R/W	Default	Description
31-18	R	0	Reserved
17	R/W	0	Gpio_pulse 0:level reset 1:pulse reset
16	R/W	0	Gpio_polarity 0: 1 is reset; 1: 0 is reset.
15-0	R/W	0	Gpio_pulse_tcnt If gpio_pulse is 1, level reset will hold tcnt clock.

WATCHDOG_TCNT 0x2636

Bit(s)	R/W	Default	Description
31-16	R	0	The cnt of tick.
15-0	R/W	5000	If watchdog_en is 1, when tick cnt reached “5000”, generate watchdog_reset.

WATCHDOG_RESET 0x2637

Bit(s)	R/W	Default	Description
31-0	W	0	When write any value(include 0), watchdog module will be reset.

AO_TIMER_REG 0xc810004c

Timer controls.

Bit(s)	R/W	Default	Description
31-5	R/W	0	Unused
4	R/W	0	TIMER_E_EN
3	R/W	0	TIMER_A_EN

Bit(s)	R/W	Default	Description
2	R/W	0	TIMER_A_MODE: 1 = periodic, 0 = one-shot
1-0	R/W	0	TIMER_CLK_MUX: 00 = TimerA clock = AO CPU clock 01 = Timer A clock = 1uS ticks 10 = Timer A clock = 10uS ticks 11 = Timer A clock = 100uS ticks

AO_TIMERA_REG 0xc8100050

Timer A starts at a non-zero value and decrements to 0. When timer A reaches a count of 0 it will re-load with the TIMER_A_TCNT value.

Bit(s)	R/W	Default	Description
31-16	R	0	TIMER A current count.
15-0	R/W	0	TIMER_A_TCNT: Timer A Terminal count

AO_TIMERE_REG 0 0xc8100054

If this register is written (with any value), then Timer E is reset to 0. Immediately after being cleared, timer E will start incrementing at a clock rate equal to the clock used for the Media CPU..

Bit(s)	R/W	Default	Description
31-0	R/W	0	TIMER E current Count

25. Crypto

25.1 Overview

The crypto engine is one encrypt/decrypt function accelerator. It supports both encryption/decryption and signature/verification. Crypto engine supports 4 different modes, i.e. A53 secure, A53 non secure, M3 secure and M3 non secure. The crypto engine has special internal DMA controller to transfer data.

It has the following features:

- AES block cipher with 128/192/256 Bits keys, standard 16 bytes block size and streaming ECB, CBC and CTR modes
- DES/TDES 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), CRC and HMAC-SHA1, HMAC-SHA224/256 engine

25.2 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 using internal storage to hold temporary states.

25.3 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): this TRNG is realized by using metastability and jitter for random bit generation based on four free running ring oscillator

Deterministic Random Number Generator (DRNG): this DRNG, which has32-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.

25.4 EFUSE

The EFUSE consists of a 4kbit One Time Programmable (OTP) memory that is broken up into 32, 128-bit blocks. Data is always read/written in 128-bit blocks using the APB bus (software) or by the Key-ladder which is integrated with EFUSE block.

Section IV Video Path

The data path of the video path module is shown in the Fig IV.1 below:

The working flow of S905X video path module is as following:

For TS input, it will go to DEMUX, which will separate the data in to video data, audio data and control data, and will write video data in to DDR, the VDEC module will read this data from DDR and decode it, and then write it back to DDR, which will be sent to VPU module and display in either HDMI format or CVBS format.

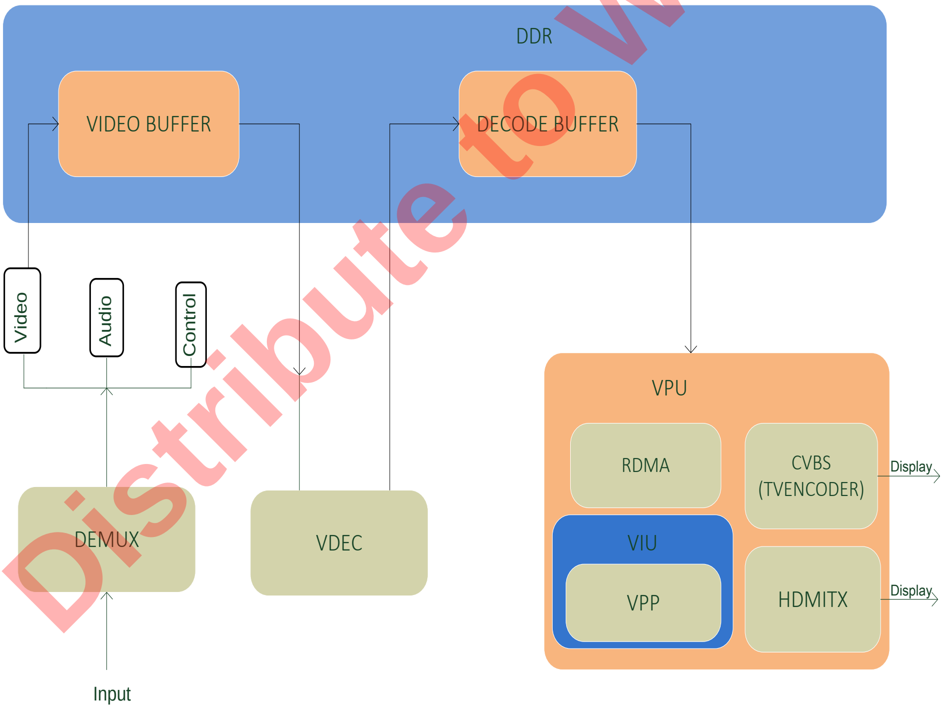
For DVP input, DVP module will write it directly to DDR, and VPU module read it and display it in either HDMI or CVBS format.

This section describe S905X video path from the following aspects:

- Video Input

- Video Output
  -RDMA
  -VPP(VIU/VPP)
  -HDMI
  -DVBS

Fig IV.1 Data Path of Video Path



26. Video Input

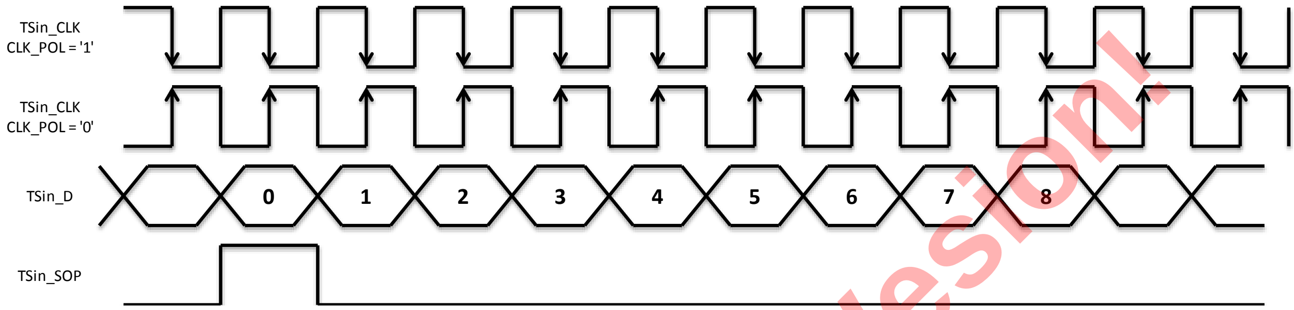
26.1 Overview

This part describes the video input module of S905X, S905X uses off-chip DEMUX to handle input video signals, and store the demuxed signal in DDR, which can be read directrly.

26.2 TSin Timing

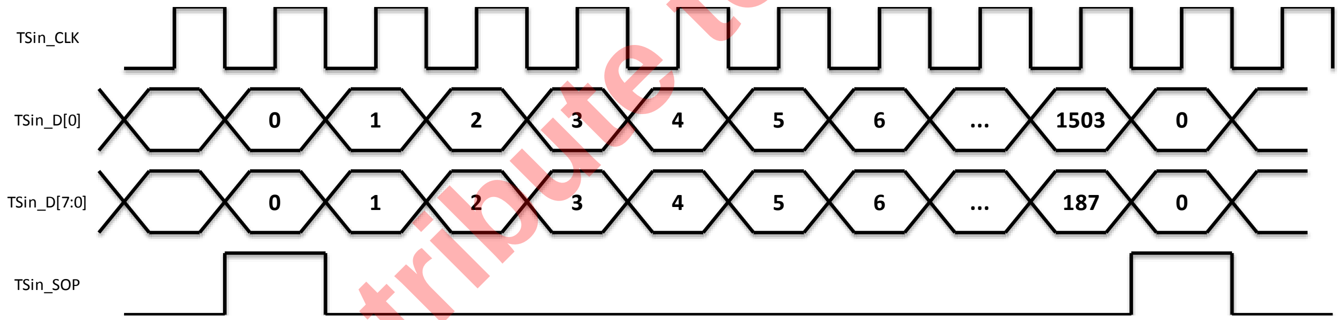
Below is TSin timing diagrams.

Fig V. 26.1 General TSin Timing Diagram



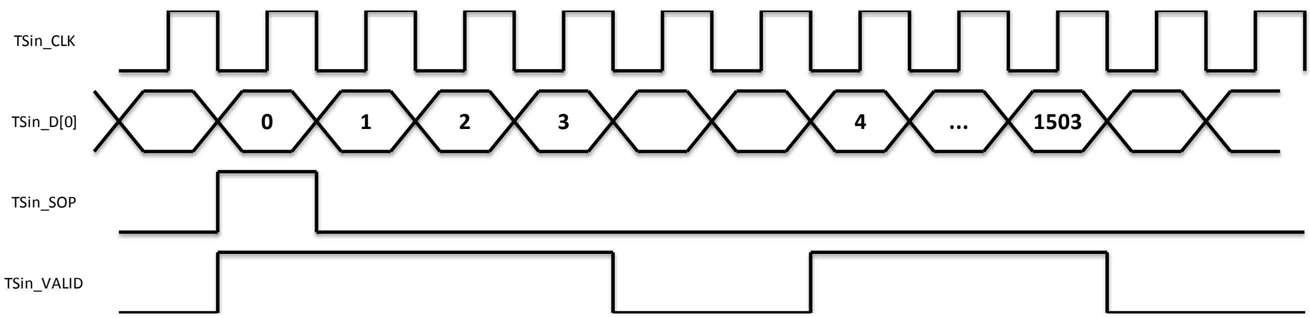
External device sends data on rising edge, TSin reads data on the falling edge, set CLK_POL = 1;

Fig V. 26.2 Different TSin Timing Modes



Sierial mode : recevie 1 bit each clock by one pad;
Parellel mode: receive 1 byte each clock by 8 pads;

Fig V. 26.3 TSin_Valid Diagram



TSin_VALID indicates valid data, if the value is '0', this data will be skipped.

26.3 Register Definition

Below are registers for Demux.

Demux Register

Demux Common Register

Final address = 0xc1105800 + offset * 4

STB_TOP_CONFIG 0xf0

Bit(s)	R/W	Default	Description
30:28	RW	0	ciplus_o_sel
27:26	RW	0	ciplus_i_sel
25	RW	0	use FAIL from TS2
24	RW	0	use FAIL from TS1
23	RW	0	use FAIL from TS0
22	RW	0	invert fec_error for S2P1
21	RW	0	invert fec_data for S2P1
20	RW	0	invert fec_sync for S2P1
19	RW	0	invert fec_valid for S2P1
18	RW	0	invert fec_clk for S2P1
17:16	RW	0	fec_s_sel for S2P1 00 - select TS0, 01 -- select TS1, 10 -- select TS2, 11 - reserved
15	RW	0	enable_des_pl_clk
14:13	RW	0	reserved
12:10	RW	0	ts_out_select, 0-TS0, 1-TS1, 2-TS2, 3,4-Reserved, 5-S2P1, 6-S2P0, 7-File
9:8	RW	0	des_i_sel 00 -- select_fec_0, 01 -- select_fec_1, 10 -- select_fec_2, 11 - reserved
7	RW	0	enable_des_pl
6	RW	0	invert fec_error for S2P0
5	RW	0	invert fec_data for S2P0
4	RW	0	invert fec_sync for S2P0
3	RW	0	invert fec_valid for S2P0
2	RW	0	invert fec_clk for S2P0
1:0	RW	0	fec_s_sel for S2P0 00 - select TS0, 01 -- select TS1, 10 -- select TS2, 11 - reserved

TS_TOP_CONFIG 0xf1

Bit(s)	R/W	Default	Description
31:28	RW	7	s2p1_clk_div
27:24	RW	7	s2p0_clk_div
23	RW	0	s2p1_disable
22	RW	0	s2p0_disable
21	RW	0	Reserved
20	RW	0	TS_OUT_error_INVERT
19	RW	0	TS_OUT_data_INVERT
18	RW	0	TS_OUT_sync_INVERT
17	RW	0	TS_OUT_valid_INVERT
16	RW	0	TS_OUT_clk_INVERT
15:8	RW	187	TS_package_length_sub_1 (default : 187)
7:0	RW	0x47	fec_sync_byte (default : 0x47)

TS_FILE_CONFIG 0xf2

Bit(s)	R/W	Default	Description

25:24	RW	3	transport_scrambling_control_odd_2 // should be 3
23:16	RW	0	file_m2ts_skip_bytes
15:8	RW	0	des_out_dly
7:6	RW	3	transport_scrambling_control_odd // should be 3
5	RW	0	ts_hiu_enable
4:0	RW	4	fec_clk_div

TS_PL_PID_INDEX 0xf3

Bit(s)	R/W	Default	Description
19:14	R	0	des_2 ts pl state -- Read Only
13:8	R	0	des ts pl state -- Read Only
3:0	RW	0	PID index to 8 PID to get key-set auto increse after TS_PL_PID_DATA read/write

TS_PL_PID_DATA 0xf4

Bit(s)	R/W	Default	Description
29	RW	0	PID #INDEX +1 match disble
28:16	RW	0	PID #INDEX+1
13	RW	0	PID #INDEX match disble
12:0	RW	0	PID #INDEX

COMM_DESC_KEY0 0xf5

Bit(s)	R/W	Default	Description
31:0	RW	0	Common descrambler key (key Bits[63:32])

COMM_DESC_KEY1 0xf6

Bit(s)	R/W	Default	Description
31:0	RW	0	Common descrambler key (key Bits[31:0])

COMM_DESC_KEY_RW 0xf7

Bit(s)	R/W	Default	Description
7	RW	0	Key endian;
6	RW	0	Write key ladder cw [127:64] to key;
5	RW	0	Write key ladder cw [63:0] to key;
4	RW	0	0: write to descramble 1; 1: write to descramble 2;
3:0	RW	0	The address of key

CIPLUS_KEY0 0xf8

Bit(s)	R/W	Default	Description
31:0	RW	0	CI+ Register defines Bits[31:0] of the key

CIPLUS_KEY1 0xf9

Bit(s)	R/W	Default	Description
31:0	RW	0	CI+ Register defines Bits[63:32] of the key

CIPLUS_KEY2 0xfa

Bit(s)	R/W	Default	Description
31:0	RW	0	CI+ Register defines Bits[95:64] of the key

CIPLUS_KEY3 0xfb

Bit(s)	R/W	Default	Description
31:0	RW	0	CI+ Register defines Bits[127:96] of the key

CIPLUS_KEY_WR 0xfc

Bit(s)	R/W	Default	Description
5	RW	0	write AES IV B value
4	RW	0	write AES IV A value
3	RW	0	write AES B key
2	RW	0	write AES A key
1	RW	0	write DES B key
0	RW	0	write DES A key

CIPLUS_CONFIG 0xfd

Bit(s)	R/W	Default	Description
15:8	RW	0	TS out delay. This controls the rate at which the CIplus module drives TS out
3	RW	0	General enable for the ciplus module
2	RW	0	AES CBC disable (default should be 0 to enable AES CBC)
1	RW	0	AES Enable
0	RW	0	DES Eanble

CIPLUS_ENDIAN 0xfe

Bit(s)	R/W	Default	Description
31:28	RW	0	AES IV endian
27:24	RW	0	AES message out endian
23:20	RW	0	AES message in endian
19:16	RW	0	AES key endian
15:11	RW	0	unused
10:8	RW	0	DES message out endian
6:4	RW	0	DES message in endian
2:0	RW	0	DES key endian

COMM_DESC_2_CTL 0xff

Bit(s)	R/W	Default	Description
15:8	RW	0	des_out_dly_2
7	RW	0	reserved
6	RW	0	enable_des_pl_clk_2
5	RW	0	enable_des_pl_2
4:2	RW	0	use_des_2 Bit[2] -- demux0, Bit[3] -- demux1, Bit[4] -- demux2
1:0	RW	0	des_i_sel_2 00 -- select_fec_0, 01 -- select_fec_1, 10 -- select_fec_2, 11 - reserved

demux core register

demux core 0 Final address = 0xc1105800 + offset * 4

demux core 1 Final address = 0xc1105940 + offset * 4

demux core 2 Final address = 0xc1105a80 + offset * 4

STB_VERSION_O 0x00

Bit(s)	R/W	Default	Description
31:0	R	0x30003	The version of stb

STB_TEST_REG_O 0x01

Bit(s)	R/W	Default	Description
31:0	RW	0xfe015aa5	Test register.

FEC_INPUT_CONTROL_O 0x02

Bit(s)	R/W	Default	Description
15	RW	0	fec_core_select 1 - select descramble output
14:12	RW	0	fec_select 0-TS0, 1-TS1, 2-TS2, 3,4-Reserved, 5-S2P1, 6-S2P0, 7-File
11	RW	0	FEC_CLK
10	RW	0	SOP
9	RW	0	D_VALID
8	RW	0	D_FAIL
7:0	RW	0	D_DATA 7:0

FEC_INPUT_DATA_O 0x03

Bit(s)	R/W	Default	Description
11	R	0	FEC_CLK
20	R	0	SOP
9	R	0	VALID
8	R	0	FAIL
7:0	R	0	FEC DATAIN

DEMUX_CONTROL_O 0x04

Bit(s)	R/W	Default	Description
31	RW	0	enable_free_clk_fec_data_valid
30	RW	0	enable_free_clk_stb_reg
29	RW	0	always_use_pes_package_length
28	RW	0	disable_pre_incomplete_section_fix
27	RW	0	pointer_field_multi_pre_en
26	RW	0	ignore_pre_incomplete_section
25	RW	0	video2_enable
24:22	RW	0	video2_type
21	RW	0	do_not_trust_pes_package_length
20 (bit4)	RW	0	Bypass use recoder path
19 (bit3)	RW	0	clear_PID_continuity_counter_valid
18 (bit2)	RW	0	Disable Splicing
17 (bit1)	RW	0	Insert PES_STRONG_SYNC in Audio PES
16 (bit0)	RW	0	Insert PES_STRONG_SYNC in Video PES
15	RW	0	do not trust section length
14	RW	0	om cmd push even zero
13	RW	0	set_buff_ready_even_not_busy
12	RW	0	SUB, OTHER PES interrupt at beginning of PES
11	RW	0	discard_av_package -- for ts_recorder use only
10	RW	0	ts_recorder_select 0:after PID filter 1:before PID filter
9	RW	0	ts_recorder_enable
8	RW	0	(table_id == 0xff) means section_end
7	RW	0	do not send uncomplete section
6	RW	0	do not discard duplicate package
5	RW	0	search SOP when trasport_error_indicator
4	RW	0	stb demux enable

Bit(s)	R/W	Default	Description
3	RW	0	do not reset state machine on SOP
2	RW	0	search SOP when error happened (when ignore_fail_n_sop, will have this case)
1	RW	0	do not use SOP input (check FEC sync byte instead)
0	RW	0	ignore fec_error bit when non sop (check error on SOP only)

FEC_SYNC_BYTE_O 0x05

Bit(s)	R/W	Default	Description
15:8	RW	187	demux package length - 1 (default : 187)
7:0	RW	0	default is 0x47

FM_WR_DATA_O 0x06

Bit(s)	R/W	Default	Description
31:16	RW	0	filter memory write data hi[31:16]
15:0	RW	0	filter memory write data low [15:0]

FM_WR_ADDR_O 0x07

Bit(s)	R/W	Default	Description
31:24	RW	0	advanced setting hi
23:16	RW	0	advanced setting low
15	R	0	filter memory write data request
7:0	R	0	filter memory write addr

MAX_FM_COMP_ADDR_O 0x08

Bit(s)	R/W	Default	Description
13:8	R	0	demux state -- read only
7:4	RW	0	maxnum section filter compare address
3:0	RW	0	maxnum PID filter compare address

TS_HEAD_0_O 0x09

Bit(s)	R/W	Default	Description
15	RW	0	transport_error_indicator
14	RW	0	payload_unit_start_indicator
13	RW	0	transport_priority
12:0	RW	0	PID

TS_HEAD_1_O 0x0a

Bit(s)	R/W	Default	Description
7:6	R	0	transport_scrambling_control
5:4	R	0	adaptation_field_control
3:0	R	0	continuity_counter

OM_CMD_STATUS_O 0x0b

Bit(s)	R/W	Default	Description
15:12	R	0	om_cmd_count (read only)
11:9	R	0	overflow_count // bit 11:9 -- om_cmd_wr_ptr (read only)
8:6	R	0	om_overwrite_count // bit 8:6 -- om_cmd_rd_ptr (read only)
5:3	R	0	type_stb_om_w_rd (read only)
2	R	0	unit_start_stb_om_w_rd (read only)

1	R	0	om_cmd_overflow (read only)
0	R	0	om_cmd_pending (read)
0	R	0	om_cmd_read_finished (write)

OM_CMD_DATA_O 0x0c

Bit(s)	R/W	Default	Description
15:9	R	0	count_stb_om_w_rd (read only)
8:0	R	0	start_stb_om_wa_rd (read only)

OM_CMD_DATA2_O 0x0d

Bit(s)	R/W	Default	Description
11:0	R	0	offset for section data

SEC_BUFF_01_START_O 0x0e

Bit(s)	R/W	Default	Description
31:16	RW	0	base address for section buffer group 0 (*0x400 to get real address)
15:0	RW	0	base address for section buffer group 1 (*0x400 to get real address)

SEC_BUFF_23_START_O 0x0f

Bit(s)	R/W	Default	Description
31:16	RW	0	base address for section buffer group 2 (*0x400 to get real address)
15:0	RW	0	base address for section buffer group 3 (*0x400 to get real address)

SEC_BUFF_SIZE_O 0x10

Bit(s)	R/W	Default	Description
3:0	RW	0	section buffer size for group 0 (bitused, for example, 10 means 1K)
7:4	RW	0	section buffer size for group 1
11:8	RW	0	section buffer size for group 2
15:12	RW	0	section buffer size for group 3

SEC_BUFF_BUSY_O 0x11

Bit(s)	R/W	Default	Description
31:0	R	0	Section buffer busy status for buff 31:0 (Read Only)

SEC_BUFF_READY_O 0x12

Bit(s)	R/W	Default	Description
31:0	RW	0	section buffer write status for buff 31:0 -- Read clear buffer status (buff READY and BUSY) – write

SEC_BUFF_NUMBER_O 0x13

Bit(s)	R/W	Default	Description
4:0	RW	0	SEC_BUFFER_INDEX RW
12:8	RW	0	SEC_BUFFER_NUMBER for the INDEX buffer Read_Only
14	RW	0	output_section_buffer_valid
15	RW	0	section_reset_busy (Read Only)

ASSIGN_PID_NUMBER_O 0x14

Bit(s)	R/W	Default	Description
9:5	RW	0	BYPASS PID number

4:0	RW	0	PCR PID number

VIDEO_STREAM_ID_O 0x15

Bit(s)	R/W	Default	Description
31:16	RW	0	for video2
15:0	RW	0	stream_id filter Bit(s) enable

AUDIO_STREAM_ID_O 0x16

Bit(s)	R/W	Default	Description
15:0	RW	0	For audio

SUB_STREAM_ID_O 0x17

Bit(s)	R/W	Default	Description
15:0	RW	0	For sub

OTHER_STREAM_ID_O 0x18

Bit(s)	R/W	Default	Description
15:0	RW	0	For other

PCR90K_CTL_O 0x19

Bit(s)	R/W	Default	Description
12	RW	0	PCR_EN
11:0	RW	0	PCR90K_DIV

PCR_DEMUX_O 0x1a

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

VIDEO_PTS_DEMUX_O 0x1b

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

VIDEO_DTS_DEMUX_O 0x1c

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

AUDIO_PTS_DEMUX_O 0x1d

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

SUB_PTS_DEMUX_O 0x1e

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

STB_PTS_DTS_STATUS_O 0x1f

Bit(s)	R/W	Default	Description
15	R	0	SUB_PTS[32]
14	R	0	AUDIO_PTS[32]
13	R	0	VIDEO_DTS[32]
12	R	0	VIDEO_PTS[32]

3	R	0	sub_pts_ready
2	R	0	audio_pts_ready
1	R	0	video_dts_ready
0	R	0	video_pts_ready

STB_DEBUG_INDEX_O 0x20

Bit(s)	R/W	Default	Description
3	RW	0	pes_ctr_byte[7:0], pes_flag_byte[7:0]
2	RW	0	pes_package_bytes_left[15:0]
1	RW	0	stream_id[7:0], pes_header_bytes_left[7:0]
0	RW	0	adaptation_field_length[7:0], adaption_field_byte_1[7:0]

STB_DEBUG_DATAOUT_O 0x21

Bit(s)	R/W	Default	Description
15:0	R	0	Debug data out[15:0]

STB_MOM_CTL_O 0x22

Bit(s)	R/W	Default	Description
31	RW	0	no_match_record_en
30:16	RW	0	reserved
15:9	RW	0	MAX OM DMA COUNT (default: 0x40)
8:0	RW	0	LAST ADDR OF OM ADDR (default: 127)

STB_INT_STATUS_O 0x23

Bit(s)	R/W	Default	Description
12	R	0	INPUT_TIME_OUT
11	R	0	PCR_ready
10	R	0	audio_splicing_point
9	R	0	video_splicing_point
8	R	0	other_PES_int
7	R	0	sub_PES_int
6	R	0	discontinuity
5	R	0	duplicated_pack_found
4	R	0	New PDTS ready
3	R	0	om_cmd_buffer ready for access
2	R	0	section buffer ready
1	R	0	transport_error_indicator
0	R	0	TS ERROR PIN

DEMUX_ENDIAN_O 0x24

Bit(s)	R/W	Default	Description
23:21	RW	0	demux om write endian control for OTHER_PES_PACKET
20:18	RW	0	demux om write endian control for SCR_ONLY_PACKET
17:15	RW	0	demux om write endian control for SUB_PACKET
14:12	RW	0	demux om write endian control for AUDIO_PACKET
11:9	RW	0	demux om write endian control for VIDEO_PACKET
8:6	RW	0	demux om write endian control for else
5:3	RW	0	demux om write endian control for bypass
2:0	RW	0	demux om write endian control for section

TS_HIU_CTL_O 0x25

Bit(s)	R/W	Default	Description
15:8	RW	0	last_burst_threshold
7	RW	0	use hi_bsf interface
6:2	RW	0	fec_clk_div
1	RW	0	ts_source_sel
0	RW	0	Hiu TS generate enable

SEC_BUFF_BASE_O 0x26

Bit(s)	R/W	Default	Description
15:0	RW	0	base address for section buffer start (*0x10000 to get real base)

DEMUX_MEM_REQ_EN_O 0x27

Bit(s)	R/W	Default	Description
11	RW	0	mask bit for OTHER_PES_AHB_DMA_EN
10	RW	0	mask bit for SUB_AHB_DMA_EN
9	RW	0	mask bit for BYPASS_AHB_DMA_EN
8	RW	0	mask bit for SECTION_AHB_DMA_EN
7	RW	0	mask bit for recoder stream
6:0	RW	0	mask bit for each type

VIDEO_PDTS_WR_PTR_O 0x28

Bit(s)	R/W	Default	Description
31:0	RW	0	vb_wr_ptr for video PDTS

AUDIO_PDTS_WR_PTR_O 0x29

Bit(s)	R/W	Default	Description
31:0	RW	0	ab_wr_ptr for video PDTS

SUB_WR_PTR_O 0x2a

Bit(s)	R/W	Default	Description
20:0	RW	0	SB_WRITE_PTR (sb_wr_ptr << 3 == byte write position)

SB_START_O 0x2b

Bit(s)	R/W	Default	Description
19:0	RW	0	SB_START (sb_start << 12 == byte address);

SB_LAST_ADDR_O 0x2c

Bit(s)	R/W	Default	Description
20:0	RW	0	SB_SIZE (sb_size << 3 == byte size, 16M maximun)

SB_PES_WR_PTR_O 0x2d

Bit(s)	R/W	Default	Description
31:0	RW	0	sb_wr_ptr for sub PES

OTHER_WR_PTR_O 0x2e

Bit(s)	R/W	Default	Description
31:21	RW	0	ob_wr_ptr for other PES

20:0	RW	0	OB_WRITE_PTR (ob_wr_ptr << 3 == byte write position)

OB_START_O 0x2f

Bit(s)	R/W	Default	Description
19:0	RW	0	OB_START (ob_start << 12 == byte address);

OB_LAST_ADDR_O 0x30

Bit(s)	R/W	Default	Description
20:0	RW	0	OB_SIZE (ob_size << 3 == byte size, 16M maximun)

OB_PES_WR_PTR_O 0x31

Bit(s)	R/W	Default	Description
31:0	RW	0	ob_wr_ptr for sub PES

STB_INT_MASK_O 0x32

Bit(s)	R/W	Default	Description
9	RW	0	splicing_point
8	RW	0	other_PES_int
7	RW	0	sub_PES_int
6	RW	0	discontinuity
5	RW	0	duplicated_pack_found
4	RW	0	New PDTS ready
3	RW	0	om_cmd_buffer ready for access
2	RW	0	section buffer ready
1	RW	0	transport_error_indicator
0	RW	0	TS ERROR PIN

VIDEO_SPLICING_CTL_O 0x33

Bit(s)	R/W	Default	Description
15	RW	0	splicing VIDEO PID change enable
14:10	RW	0	VIDEO PID FILTER ADDRESS
9	RW	0	PES splicing active (Read Only)
8	RW	0	splicing active (Read Only)
7:0	RW	0	splicing countdown (Read Only)

AUDIO_SPLICING_CTL_O 0x34

Bit(s)	R/W	Default	Description
15	RW	0	splicing AUDIO PID change enable
14:10	RW	0	AUDIO PID FILTER ADDRESS
9	RW	0	PES splicing active (Read Only)
8	RW	0	splicing active (Read Only)
7:0	RW	0	splicing countdown (Read Only)

TS_PACKAGE_BYTE_COUNT_O 0x35

Bit(s)	R/W	Default	Description
23:16	RW	0	M2TS_SKIP_BYTES
15:8	RW	0	LAST TS PACKAGE BYTE COUNT (Read Only)
7:0	RW	0	PACKAGE BYTE COUNT (Read Only)

PES_STRONG_SYNC_O 0x36

Bit(s)	R/W	Default	Description
15:0	RW	0	2 bytes strong sync add to PES

OM_DATA_RD_ADDR_O 0x37

Bit(s)	R/W	Default	Description
15	RW	0	stb_om_ren
14:11	RW	0	reserved
10:0	RW	0	OM_DATA_RD_ADDR

OM_DATA_RD_O 0x38

Bit(s)	R/W	Default	Description
15:0	RW	0	OM_DATA_RD

SECTION_AUTO_STOP_3_O 0x39

Bit(s)	R/W	Default	Description
31	RW	0	Auto stop count 31 Wr_en
30:28	RW	0	Auto stop count 31
27	RW	0	Auto stop count 30 Wr_en
26:24	RW	0	Auto stop count 30
23	RW	0	Auto stop count 29 Wr_en
22:20	RW	0	Auto stop count 29
19	RW	0	Auto stop count 28 Wr_en
18:16	RW	0	Auto stop count 28
15	RW	0	Auto stop count 27 Wr_en
14:12	RW	0	Auto stop count 27
11	RW	0	Auto stop count 26 Wr_en
10:8	RW	0	Auto stop count 26
7	RW	0	Auto stop count 25 Wr_en
6:4	RW	0	Auto stop count 25
3	RW	0	Auto stop count 24 Wr_en
2:0	RW	0	Auto stop count 24

SECTION_AUTO_STOP_2_O 0x3a

Bit(s)	R/W	Default	Description
31	RW	0	Auto stop count 23 Wr_en
30:28	RW	0	Auto stop count 23
27	RW	0	Auto stop count 22 Wr_en
26:24	RW	0	Auto stop count 22
23	RW	0	Auto stop count 21 Wr_en
22:20	RW	0	Auto stop count 21
19	RW	0	Auto stop count 20 Wr_en
18:16	RW	0	Auto stop count 20
15	RW	0	Auto stop count 19 Wr_en
14:12	RW	0	Auto stop count 19
11	RW	0	Auto stop count 18 Wr_en
10:8	RW	0	Auto stop count 18
7	RW	0	Auto stop count 17 Wr_en
6:4	RW	0	Auto stop count 17
3	RW	0	Auto stop count 16 Wr_en
2:0	RW	0	Auto stop count 16

SECTION_AUTO_STOP_1_O 0x3b

Bit(s)	R/W	Default	Description
31	RW	0	Auto stop count 15 Wr_en
30:28	RW	0	Auto stop count 15
27	RW	0	Auto stop count 14 Wr_en
26:24	RW	0	Auto stop count 14
23	RW	0	Auto stop count 13 Wr_en
22:20	RW	0	Auto stop count 13
19	RW	0	Auto stop count 12 Wr_en
18:16	RW	0	Auto stop count 12
15	RW	0	Auto stop count 11 Wr_en
14:12	RW	0	Auto stop count 11
11	RW	0	Auto stop count 10 Wr_en
10:8	RW	0	Auto stop count 10
7	RW	0	Auto stop count 9 Wr_en
6:4	RW	0	Auto stop count 9
3	RW	0	Auto stop count 8 Wr_en
2:0	RW	0	Auto stop count 8

SECTION_AUTO_STOP_0_O 0x3c

Bit(s)	R/W	Default	Description
31	RW	0	Auto stop count 7 Wr_en
30:28	RW	0	Auto stop count 7
27	RW	0	Auto stop count 6 Wr_en
26:24	RW	0	Auto stop count 6
23	RW	0	Auto stop count 5 Wr_en
22:20	RW	0	Auto stop count 5
19	RW	0	Auto stop count 4 Wr_en
18:16	RW	0	Auto stop count 4
15	RW	0	Auto stop count 3 Wr_en
14:12	RW	0	Auto stop count 3
11	RW	0	Auto stop count 2 Wr_en
10:8	RW	0	Auto stop count 2
7	RW	0	Auto stop count 1 Wr_en
6:4	RW	0	Auto stop count 1
3	RW	0	Auto stop count 0 Wr_en
2:0	RW	0	Auto stop count 0

DEMUX_CHANNEL_RESET_O 0x3d

Bit(s)	R/W	Default	Description
31:0	R	0	Bit 31:0 reset channel status - Each Bit reset each channel

DEMUX_SCRAMBLING_STATE_O 0x3e

Bit(s)	R/W	Default	Description
31:0	R	0	Scrambling state of each channel

DEMUX_CHANNEL_ACTIVITY_O 0x3f

Bit(s)	R/W	Default	Description
31:0	R	0	Channel activity of each channel

DEMUX_STAMP_CTL_O 0x40

Bit(s)	R/W	Default	Description
4	RW	0	video_stamp_use_dts
3	RW	0	audio_stamp_sync_1_en
2	RW	0	audio_stamp_insert_en
1	RW	0	video_stamp_sync_1_en
0	RW	0	video_stamp_insert_en

DEMUX_VIDEO_STAMP_SYNC_0_O 0x41

Bit(s)	R/W	Default	Description
31:0	RW	0	Video stamp sync [63:32]

DEMUX_VIDEO_STAMP_SYNC_1_O 0x42

Bit(s)	R/W	Default	Description
31:0	RW	0	Video stamp sync [31:0]

DEMUX_AUDIO_STAMP_SYNC_0_O 0x43

Bit(s)	R/W	Default	Description
31:0	RW	0	Aideo stamp sync [63:32]

DEMUX_AUDIO_STAMP_SYNC_1_O 0x44

Bit(s)	R/W	Default	Description
31:0	RW	0	Aideo stamp sync [31:0]

DEMUX_SECTION_RESET_O 0x45

Bit(s)	R/W	Default	Description
31:0	R	0	Write : Bit[4:0] secter filter number for reset Read : select according to output_section_buffer_valid: per bit per section buffer valid status or section_buffer_ignore

DEMUX_INPUT_TIMEOUT_C_O 0x46

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

DEMUX_INPUT_TIMEOUT_O 0x47

Bit(s)	R/W	Default	Description
31	RW	0	no_match_reset_timeout_disable
30:0	RW	0	input_time_out_int_cnt (0 -- means disable) Wr-setting, Rd-count

DEMUX_PACKET_COUNT_O 0x48

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

DEMUX_PACKET_COUNT_C_O 0x49

Bit(s)	R/W	Default	Description
31	RW	0	no_match_packet_count_disable
30:0	RW	0	input_packet_count

DEMUX_CHAN_RECORD_EN_O 0x4a

Bit(s)	R/W	Default	Description
31:0	RW	0xffffffff	channel_record_enable

DEMUX_CHAN_PROCESS_EN_O 0x4b

Bit(s)	R/W	Default	Description
31:0	RW	0xffffffff	channel_process_enable

DEMUX_SMALL_SEC_CTL_O 0x4c

Bit(s)	R/W	Default	Description
31:24	RW	0	small_sec_size ((n+1) * 256 Bytes)
23:16	RW	0	small_sec_rd_ptr
15:8	RW	0	small_sec_wr_ptr
7:2	RW	0	reserved
1	RW	0	small_sec_wr_ptr_wr_enable
0	RW	0	small_section_enable

27. Video Output

27.1 Overview

This section describes S905X's VPU sub-module, including RDMA sub-module, HDMI sub-module and the CVBS sub-module.

27.2 VPU

VPU is designed as the interface of the video input module and the video out module. The main function of VPU is to get data from DDR and deliver it to CVBS or HDMITX based on different format.

VPU Registers

VPU_VIU_VENC_MUX_CTRL 0x271a

Bit(s)	R/W	Default	Description
11-8	R/W	0	VIU_VDIN_SEL_DATA: Select which data to VDI6 path, must clear it first before switching the data. 4'b0000: Disable VIU to VDI6 path 4'b0001: Select ENCI data to VDI6 4'b0010: Select ENCP data to VDI6 4'b0100: Select ENCT data to VDI6 4'b1000: Select ENCL data to VDI6
7-4	R/W	0	VIU_VDIN_SEL_CLK: Select which clock to VDI6 path, must clear it first before switching the clock. 4'b0000: Disable VIU to VDI6 clock 4'b0001: Select ENCI clock to VDI6 4'b0010: Select ENCP clock to VDI6 4'b0100: Select ENCT clock to VDI6 4'b1000: Select ENCL clock to VDI6
3-2	R/W	0	VIU2_SEL_VENC: Select which one of the encI/P/T/L that Viu2 connects to. 0: ENCL 1: ENCI 2: ENCP 3: ENCT
1-0	R/W	0	VIU1_SEL_VENC: Select which one of the encI/P/T/L that Viu1 connects to. 0: ENCL 1: ENCI 2: ENCP 3: ENCT

VPU_HDMI_SETTING 0x271b

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)

Bit(s)	R/W	Default	Description
			6,7: Reserved
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

ENCI_INFO_READ 0x271c

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.

ENCP_INFO_READ 0x271d

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.

ENCT_INFO_READ 0x271e

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.

VPU_SW_RESET 0x2720

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

VPU_D2D3_MMC_CTRL 0x2721

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_depw_req_sel, 0: vdin_pre_arb, 1: vdisp_pre_arb
11-6	R/W	0x3f	d2d3_depw_brst_num
5-0	R/W	0x2e	d2d3_depw_id

VPU_CONT_MMC_CTRL 0x2722

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

VPU_CLK_GATE 0x2723

Bit(s)	R/W	Default	Description
30-18	R/W	0x0	reserved
17	R/W	0x1	reserved
16	R/W	0x1	Vpu_clkb enable
15	R/W	0x1	Gvapbclk enable
14	R/W	0x1	Vlock clock enable
13	R/W	0x1	reserved
12	R/W	0x1	Venc_dac_enable
11	R/W	0x1	Venc_i_enable
10	R/W	0x1	Venci_int_enable
9	R/W	0x1	Reserved
8	R/W	0x1	Reserved
7	R/W	0x1	Reserved
6	R/W	0x1	Mpeg_vpu_misc_enable
5	R/W	0x1	Mpeg_venc_l_top_enable
4	R/W	0x1	Mpeg_venc_l_int_enable
3	R/W	0x1	Mpeg_vencp_int_enable
2	R/W	0x1	Reserved
1	R/W	0x1	Mpeg_vi_top
0	R/W	0x1	Mpeg_venc_p_top_enable

VPU_HDMI_DATA_OVR 0x2727

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	Reserved
29-0	R/W	0	DATA_OVR: programmable pixel data value for override.

VPU_VPU_PWM_V0 0x2730

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

VPU_VPU_PWM_V1 0x2731

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

VPU_VPU_PWM_V2 0x2732

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

VPU_VPU_PWM_V3 0x2733

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

VPU_VPU_PWM_H0 0x2734

Bit(s)	R/W	Default	Description
28-16	R/W	0	reg_vpu_pwm_h_end0

Bit(s)	R/W	Default	Description
12-0	R/W	0	reg_vpu_pwm_h_start0

VPU_VPU_PWM_H1 0x2735

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

VPU_VPU_PWM_H2 0x2736

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

VPU_VPU_PWM_H3 0x2737

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

VPU_VPU_3D_SYNC1 0x2738

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
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

VPU_VPU_3D_SYNC2 0x2739

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

VPU_HDMI_FMT_CTRL 0x2743

Bit(s)	R/W	Default	Description
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_dnsmp: 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 convertion; 1 = Convert to 422; 2 = Convert to 420;

VPU_VDIN_ASYNC_HOLD_CTRL 0x2744

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

VPU_VDISP_ASYNC_HOLD_CTRL 0x2745

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

VPU_VPUARB2_ASYNC_HOLD_CTRL 0x2746

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

VPU_ARB_URG_CTRL 0x2747

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

VPU_VIU2VDIN_HDN_CTRL 0x2780

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_viu2vdin_flt_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

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

VPU_RDARB_REQEN_SLV_L1C1 0x2791

Bit(s)	R/W	Default	Description
11:0	R/W	0xfff	rdarb_dc_req_en : unsigned , default = 12'hfff rdarb_dc_req_en [0]: the slv0 req to mst port0 enable,

Bit(s)	R/W	Default	Description
			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,

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

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

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

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,

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

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

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

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

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,

VPU_RDARB_WEIGH0_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

VPU_RDWR_ARB_STATUS_L1C2 0x279c

Bit(s)	R/W	Default	Description
1:0	R/W	0	rdarb_arb_busy : unsigned , default = 0

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

Bit(s)	R/W	Default	Description
			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

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,

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

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

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

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

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,

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

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 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

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

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

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

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
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

Bit(s)	R/W	Default	Description
3:0	R/W	0	rd_rel_num : unsigned , default = 0 release the read command threshold

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

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

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
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

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

VPU Video Lock Registers

VPU_VLOCK_CTRL 0x3000

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
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

Bit(s)	R/W	Default	Description
18-16	R/W	0x0	Input Vsync source select: 0: from bt656, 1: from tv-decoder, 2:from hdmi rx, 3: from dvin, 4: from dvin, 5: refer to bit[28], from software set mpeg vsync
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

VPU_VLOCK_MISC_CTRL 0x3001

Bit(s)	R/W	Default	Description
26-24	R/W	0x0	Adj_capt_pxgrouprs, make sure the pixel number in one line of encoder is multiples of 2^pxgrouprs
23-16	R/W	0x0	Ifrm_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

VPU_VLOCK_LOOP0_ACCUM_LMT 0x3002

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	LOOP0 accumulator limit

VPU_VLOCK_LOOP0_CTRL0 0x3003

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
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

VPU_VLOCK_LOOP0_CTRL1 0x3004

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

VPU_VLOCK_LOOP1_IMISSYNC_MAX 0x3005

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

VPU_VLOCK_LOOP1_IMISSYNC_MIN 0x3006

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

VPU_VLOCK_OVERWRITE_ACCUM0 0x3007

Bit(s)	R/W	Default	Description
27-0	R/W	0x0	Overwrite value of accum0

VPU_VLOCK_OVERWRITE_ACCUM1 0x3008

Bit(s)	R/W	Default	Description
27-0	R/W	0x0	Overwrite value of accum1

VPU_VLOCK_OUTPUT0_CAPT_LMT 0x3009

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	Output0 capture limit

VPU_VLOCK_OUTPUT0_PLL_LMT 0x300a

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	Output0 pll limit

VPU_VLOCK_OUTPUT1_CAPT_LMT 0x300b

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	Output1 capture limit

VPU_VLOCK_OUTPUT1_PLL_LMT 0x300c

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	Output1 pll limit

VPU_VLOCK_LOOP1_PHSDIF_TARGET 0x300d

Bit(s)	R/W	Default	Description
27-0	R/W	0x0	Loop1 phase difference target, (input vsync - output vsync) phase distance target

VPU_VLOCK_RO_LOOP0_ACCUM 0x300e

Bit(s)	R/W	Default	Description
27-0	R	0x0	Read only, loop0 accum result

VPU_VLOCK_RO_LOOP1_ACCUM 0x300f

Bit(s)	R/W	Default	Description
27-0	R	0x0	Read only, loop1 accum result

VPU_VLOCK_OROW_OCOL_MAX 0x3010

Bit(s)	R/W	Default	Description
29-16	R/W	0x0	Ocol_max
13-0	R/W	0x0	Orow_max

VPU_VLOCK_RO_VS_I_D 0x3011

Bit(s)	R/W	Default	Description
27-0	R	0x0	Read only, input vsync counter

VPU_VLOCK_RO_VS_O_D 0x3012

Bit(s)	R/W	Default	Description
27-0	R	0x0	Read only, output vsync counter

VPU_VLOCK_RO_LINE_PIX_ADJ 0x3013

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

VPU_VLOCK_RO_OUTPUT_00_01 0x3014

Bit(s)	R/W	Default	Description
31-16	R	0x0	Read only, accum0 output 00
15-0	R	0x0	Read only, accum0 output 01

VPU_VLOCK_RO_OUTPUT_10_11 0x3015

Bit(s)	R/W	Default	Description
31-16	R	0x0	Read only, accum1 output 10
15-0	R	0x0	Read only, accum1 output 11

VPU_VLOCK_MX4096 0x3016

Bit(s)	R/W	Default	Description
20-0	R/W	0x0	Mx4096

VPU_VLOCK_STBDET_WIN0_WIN1 0x3017

Bit(s)	R/W	Default	Description
15-8	R/W	0x0	Verr_stbdet_win1
7-0	R/W	0x0	Verr_stbdet_win0

VPU_VLOCK_STBDET_CLP 0x3018

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

VPU_VLOCK_STBDET_ABS_WIN0 0x3019

Bit(s)	R/W	Default	Description
23-0	R	0x0	Read only, ro_verr_abs_win0

VPU_VLOCK_STBDET_ABS_WIN1 0x301a

Bit(s)	R/W	Default	Description
23-0	R	0x0	Read only, ro_verr_abs_win1

VPU_VLOCK_STBDET_SGN_WIN0 0x301b

Bit(s)	R/W	Default	Description
23-0	R	0x0	Read only, ro_verr_sgn_win0

VPU_VLOCK_STBDET_SGN_WIN1 0x301c

Bit(s)	R/W	Default	Description
23-0	R	0x0	Read only, ro_verr_sgn_win1

VPU_VLOCK_ADJ_EN_SYNC_CTRL 0x301d

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

VPU_VLOCK_GCLK_EN 0x301e

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

VPU_VLOCK_LOOP1_ACCUM_LMT 0x301f

Bit(s)	R/W	Default	Description
26-0	R/W	0x0	LOOP1 accumulator limit

VPU_VLOCK_RO_M_INT_FRAC 0x3020

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

VIU Top-Level Registers

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

VIU_SW_RESET 0x1A02

Bit(s)	R/W	Default	Description
2	R/W	0	afbc_dec_rst_n

VIU_MISC_CTRL0 0x1A06

Bit(s)	R/W	Default	Description
29	R/W	0	vd1 go field/line _sel: 1->post frame rst/post_go_line; 0-> enci/p/l frame rst/go_line
28	R/W	0	afbc go field/line _sel: 1->pre frame rst/pre_go_line; 0-> enci/p/l frame rst/go_line
27-22	R/W	0	afbc gate clk ctrl
20	R/W	0	afbc vd1 set: 1-> afbc to vd1(afbc) ; 0->vd1 mif(ddr) to vd1(afbc)
18	R/W	0	di_mad_en: di post to vpp enable
17	R/W	0	mif0_to_vpp_en: enable vd1(afbc) to vpp
16	R/W	0	di_mif0_en: vd1(afbc) to di post(if0) enable
8	R/W	0	vsync_int , enable field to vsync

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

VIUB_SW_RESET 0x2001

Bit(s)	R/W	Default	Description
31	R/W	0	mcvecwr_mif_rst_n
30	R/W	0	reserved
29	R/W	0	reserved
28	R/W	0	di_cont_rd_mif_rst_n
27	R/W	0	di_cont_wr_mif_rst_n
26	R/W	0	reserved
25	R/W	0	reserved
24	R/W	0	vdin1_wr_rst_n
23	R/W	0	vdin0_wr_rst_n
22	R/W	0	nrin_mux_rst_n
21	R/W	0	vdin1_rst_n
20	R/W	0	vdin0_rst_n

Bit(s)	R/W	Default	Description
19	R/W	0	di_mad_rst_n
18	R/W	0	di_mtn_rd_mif_rst_n
17	R/W	0	di_mtn_wr_mif_rst_n
16	R/W	0	di_chan2_mif_rst_n
15	R/W	0	dein_wr_mif_rst_n
14	R/W	0	di_nr_wr_mif_rst_n
13	R/W	0	di_mem_fmt_rst_n
12	R/W	0	di_mem_rst_n
11	R/W	0	di_inp_fmt_rst_n
10	R/W	0	di_inp_rst_n
9	R/W	0	di_if1_fmt_rst_n
8	R/W	0	di_if1_rst_n
7-0	R/W	0	RESERVED

VIUB_SW_RESET0 0x2002

Bit(s)	R/W	Default	Description
3	R/W	0	di_axi_arb_rst_n
2	R/W	0	mcinford_mif_rst_n
1	R/W	0	mcinfowr_mif_rst_n
0	R/W	0	mcvecrd_mif_rst_n

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)
7-6	R/W	0	Fix_disable: vdin1_wr_mif
5-4	R/W	0	Fix_disable: vdin0_wr_mif
3-2	R/W	0	Fix_disable:dein_wr_mif
1-0	R/W	0	Fix_disable: di_nr_wr_mif

VIU_MISC_CTRL1 0x3107

Bit(s)	R/W	Default	Description
15:14	R/W	0x0	mali_afbcd_gclk_ctrl : mali_afbcd clock gate control[5:4]
12	R/W	0x0	osd1_afbcd_axi_mux : 0 : use the osd mif as input; 1 : use afbcd as input
11:8	R/W	0x0	mali_afbcd_gclk_ctrl : mali_afbcd clock gate control[3:0]
7:2	R/W	0x0	vd2_afbcd_gclk_ctrl : vd2_afbcd clock gate control
1	R/W	0x0	vpp_vd2_din_sel : 0: vpp vd2 sel the mif input; 1: vpp vd2 sel the dos afbcd
0	R/W	0x0	vd2_afbcd_out_sel : 0: vd2_afbcd output to vpp; 1 : vd2_afbcd output to di inp

VD1 Path vd_rmem_if0 Registers

VD1_IF0_GEN_REG 0x1A50

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.

Bit(s)	R/W	Default	Description
			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.
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 endianess 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 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.

VD1_IF0_CANVAS0 – Picture 0 0x1A51

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).

VD1_IF0_CANVAS1 – Picture 1 0x1A52

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).

VD1_IF0_LUMA_X0 – Picture 0 0x1A53

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

VD1_IF0_LUMA_Y0 – Picture 0 0x1A54

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

VD1_IF0_CHROMA_X0 – Picture 0 0x1A55

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 tegether.
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 tegether.

VD1_IF0_ CHROMA_Y0 – Picture 0 0x1A56

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 tegether.
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 tegether.

VD1_IF0_LUMA_X1 – Picture 1 0x1A57

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

VD1_IF0_LUMA_Y1 – Picture 1 0x1A58

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

VD1_IF0_CHROMA_X1 – Picture 1 0x1A59

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 tegether.
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 tegether.

VD1_IF0_ CHROMA_Y1 – Picture 1 0x1A5A

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 tegether.
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 tegether.

VD1_IF0_REPEAT_LOOP – Pictures 0 and 1 0x1A5B

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.

VD1_IF0_LUMA0_RPT_PAT – Picture 0 LUMA repeat pattern 0x1A5C

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.

VD1_IF0_CHROMA0_RPT_PAT – Picture 0 CHROMA repeat pattern 0x1A5D

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.

VD1_IF0_LUMA1_RPT_PAT – Picture 1 LUMA repeat pattern 0x1A5E

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.

VD1_IF0_CHROMA1_RPT_PAT – Picture 1 CHROMA repeat pattern 0x1A5F

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.

VD1_IF0_LUMA_PSEL – Picture 0 and 1's LUMA 0x1A60

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.
7-4	R/W	0	Luma_psel_loop start pointer.
3-0	R/W	0	Luma_psel_loop end pointer.

VD1_IF0_CHROMA_PSEL – Picture 0 and 1's CHROMA 0x1A61

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 tegether.
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 tegether.
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 tegether.
7-4	R/W	0	Chroma_psel_loop start pointer. This value is only used when the picture is not stored tegether.
3-0	R/W	0	Chroma_psel_loop end pointer. This value is only used when the picture is not stored tegether.

VD1_IF0_DUMMY_PIXEL 0x1A62

Bit(s)	R/W	Default	Description
31-24	R/W	0	Y or R dummy pixel value
23-16	R/W	0	Cb or G dummy pixel value
15-8	R/W	0	Cr or B dummy pixel value
7-0	R/W	0	unused

VD1_RANGE_MAP_Y 0x1A6A

VD1_RANGE_MAP_CB 0x1A6B

VD1_RANGE_MAP_CR 0x1A6C

Output data range conversion function:

Y[n] = clip(Round(((Y[n]+DIN_OFFSET) * RANGE_MAP_COEF) / (1<<RANGE_MAP_SR)) +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] = clip((((Y[n]-128) * (RANGE_MAPY + 9) + 4) >> 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

VD1_IF0_GEN_REG2 0x1A6D

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).

VIU_VD1_FMT_CTRL 0x1A68

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 generete 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
17	R/W	0	veritcal formatter dont need repeat line on phase0, 1: enable, 0: disable
16	R/W	0	veritcal 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

VIU_VD1_FMT_W 0x1A69

Bit(s)	R/W	Default	Description
27-16	R/W	0	horizontal formatter width
11-0	R/W	0	vertical formatter width

VD2 Path vd_rmem_if0 Registers

VD2_IF0_GEN_REG 0x1A70

Same as VD1_IF0_GEN_REG

VD2_IF0_CANVAS0 0x1A71

Same as VD1_IF0_CANVAS0

VD2_IF0_CANVAS1 0x1A72

Same as VD1_IF0_CANVAS1

VD2_IF0_LUMA_X0 0x1A73

Same as VD1_IF0_LUMA_X0

VD2_IF0_LUMA_Y0 0x1A74

Same as VD1_IF0_LUMA_Y0

VD2_IF0_CHROMA_X0 0x1A75

Same as VD1_IF0_CHROMA_X0

VD2_IF0_CHROMA_Y0 0x1A76

Same as VD1_IF0_CHROMA_Y0

VD2_IF0_LUMA_X1 0x1A77

Same as VD1_IF0_LUMA_X1

VD2_IF0_LUMA_Y1 0x1A78

Same as VD1_IF0_LUMA_Y1

VD2_IF0_CHROMA_X1 0x1A79

Same as VD1_IF0_CHROMA_X1

VD2_IF0_CHROMA_Y1 0x1A7A

Same as VD1_IF0_CHROMA_Y1

VD2_IF0_RPT_LOOP 0x1A7B

Same as VD1_IF0_RPT_LOOP

VD2_IF0_LUMA0_RPT_PAT 0x1A7C

Same as VD1_IF0_LUMA0_RPT_PAT

VD2_IF0_CHROMA0_RPT_PAT 0x1A7D

Same as VD1_IF0_CHROMA0_RPT_PAT

VD2_IF0_LUMA1_RPT_PAT 0x1A7E

Same as VD1_IF0_LUMA1_RPT_PAT

VD2_IF0_CHROMA1_RPT_PAT 0x1A7F

Same as VD1_IF0_CHROMA1_RPT_PAT

VD2_IF0_LUMA_PSEL 0x1A80

Same as VD1_IF0_LUMA_PSEL

VD2_IF0_CHROMA_PSEL 0x1A81

Same as VD1_IF0_CHROMA_PSEL

VD2_IF0_DUMMY_PIXEL 0x1A82

Same as VD1_IF0_DUMMY_PIXEL

VD2_RANGE_MAP_Y 0x1A8A

Same as VD1_RANGE_MAP_Y

VD2_RANGE_MAP_CB 0x1A8B

Same as VD1_RANGE_MAP_CB

VD2_RANGE_MAP_CR 0x1A8C

Same as VD1_RANGE_MAP_CR

VD2_IF0_GEN_REG2 0x1A8D

Same as VD1_IF0_GEN_REG2

VIU_VD2_FMT_CTRL 0x1A88

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.
28	R/W	0	if true, horizontal formatter use repeating to generete 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
17	R/W	0	veritcal formatter dont need repeat line on phase0, 1: enable, 0: disable
16	R/W	0	veritcal 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

VIU_VD2_FMT_W 0x1A89

Bit(s)	R/W	Default	Description
27-16	R/W	0	horizontal formatter width
11-0	R/W	0	vertical formatter width

VIUB Registers (slow clock)

DI_IF1_GEN_REG3 0x20a7

Bit(s)	R/W	Default	Description
6-4	R/W	3	cntl_blk_len
2-1	R/W	1	cntl_burst_len
0	R/W	1	cntl_64bit_rev

DI_INP_GEN_REG3 0x20a8

Bit(s)	R/W	Default	Description
6-4	R/W	3	cntl_blk_len
2-1	R/W	1	cntl_burst_len
0	R/W	1	cntl_64bit_rev

DI_MEM_GEN_REG3 0x20a9

Bit(s)	R/W	Default	Description
6-4	R/W	3	cntl_blk_len
2-1	R/W	1	cntl_burst_len
0	R/W	1	cntl_64bit_rev

DI_CHAN2_GEN_REG3 0x20aa

Bit(s)	R/W	Default	Description
6-4	R/W	3	cntl_blk_len
2-1	R/W	1	cntl_burst_len
0	R/W	1	cntl_64bit_rev

VPU AFBC Registers

AFBC_ENABLE 0x1ae0

Bit(s)	R/W	Default	Description
8	R/W	0	dec_enable : unsigned , default = 0
0	R/W	0	frm_start : unsigned , default = 0

AFBC_MODE 0x1ae1

Bit(s)	R/W	Default	Description
31	R/W	0x0	soft_reset : the use as go_field
28	R/W	0x0	Blk_mem_mode : Default = 0, body space save mode when blk_mem_mode ==1
27:26	R/W	0	rev_mode : uns, default = 0 , reverse mode
25:24	R/W	3	mif_urgent : uns, default = 3 , info mif and data mif urgent
22:16	R/W	0x0	hold_line_num :
15:14	R/W	1	burst_len : uns, default = 1, 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 bit 3:2,: u component bitwidth : 00-8bit 01-9bit 10-10bit bit 5:4,: v component bitwidth : 00-8bit 01-9bit 10-10bit
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

AFBC_SIZE_IN 0x1ae2

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

AFBC_DEC_DEF_COLOR 0x1ae3

Bit(s)	R/W	Default	Description
29:20	R/W	0	def_color_y : uns, default = 0, afbc dec y default setting value
19:10	R/W	0	def_color_u : uns, default = 0, afbc dec u default setting value
9: 0	R/W	0	def_color_v : uns, default = 0, afbc dec v default setting value

AFBC_CONV_CTRL 0x1ae4

Bit(s)	R/W	Default	Description
11: 0	R/W	256	conv_lbuf_len : uns, default = 256, unit=16 pixel need to set = 2^n

AFBC_LBUF_DEPTH 0x1ae5

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;

AFBC_HEAD_BADDR 0x1ae6

Bit(s)	R/W	Default	Description
31:0	R/W	0x0	mif_info_baddr : uns, default = 0x0;

AFBC_BODY_BADDR 0x1ae7

Bit(s)	R/W	Default	Description
31:0	R/W	0x0001_0000	mif_data_baddr : uns, default = 0x0001_0000;

AFBC_OUT_XSCOPE 0x1ae8

Bit(s)	R/W	Default	Description
28:16	R/W	0	out_horz_bgn : uns, default = 0 ; // unit: 1 pixel
12:0	R/W	1919	out_horz_end : uns, default = 1919 ; // unit: 1 pixel

AFBC_OUT_YSCOPE 0x1ae9

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

AFBC_STAT 0x1aea

Bit(s)	R/W	Default	Description
0	RO	0x0	frm_end_stat : uns, frame end status

AFBC_VD_CFMT_CTRL 0x1aeb

Bit(s)	R/W	Default	Description
31	R/W	0x0	it : true, disable clock, otherwise enable clock
30	R/W	0x0	soft : rst bit
28	R/W	0x0	if : true, horizontal formatter use repeating to generete pixel, otherwise use bilinear interpolation
27:24	R/W	0x0	horizontal : formatter initial phase
23	R/W	0x0	horizontal : formatter repeat pixel 0 enable
22:21	R/W	0x0	horizontal : Y/C ratio, 00: 1:1, 01: 2:1, 10: 4:1
20	R/W	0x0	horizontal : formatter enable
19	R/W	0x0	if : true, always use phase0 while vertical formater, meaning always repeat data, no interpolation
18	R/W	0x0	if : true, disable vertical formatter chroma repeat last line
17	R/W	0x0	veritcal : formatter dont need repeat line on phase0, 1: enable, 0: disable
16	R/W	0x0	veritcal : formatter repeat line 0 enable
15:12	R/W	0x0	vertical : formatter skip line num at the beginning
11:8	R/W	0x0	vertical : formatter initial phase
7:1	R/W	0x0	vertical : formatter phase step (3.4)
0	R/W	0x0	vertical : formatter enable

AFBC_VD_CFMT_W 0x1aec

Bit(s)	R/W	Default	Description
27:16	R/W	0x0	horizontal : formatter width
11:0	R/W	0x0	vertical : formatter width

AFBC_MIF_HOR_SCOPE 0x1aed

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

AFBC_MIF_VER_SCOPE 0x1aee

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

AFBC_PIXEL_HOR_SCOPE 0x1aef

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

AFBC_PIXEL_VER_SCOPE 0x1af0

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

AFBC_VD_CFMT_H 0x1af1

Bit(s)	R/W	Default	Description
12:0	R/W	0x0	vertical : formatter height

De-Interlace vd_rmem_if1 Registers

DI_IF1_GEN_REG 0x17E8

Bit(s)	R/W	Default	Description
31	W/R	0	enable free clk
30	W/R	0	sw reset : pulse bit
29	W/R	0	reset on go field
28	W/R	0	urgent chroma
27	W/R	0	urgent luma
26	W/R	0	chroma end at last line : 0 = read last line or push dummy after last line; 1 = stop read after last line
25	W/R	0	luma end at last line : 0 = read last line or push dummy after last line; 1 = stop read after last line
24-19	W/R	4	hold line[5:0], see GEN_REG2[6]
18	W/R	1	last line mode: 0 = read last line; 1 = push fixed value
16	W/R	0	demux mode: 0 = 4:2:2 demux; 1 = RGB demuxing from a single FIFO
15-14	W/R	0	bytes per pixel : 0= 1byte per pixel; 1 = 2 bytes per pixel; 2 = 3bytes per pixel
13-12	W/R	0	burst size cr: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
11-10	W/R	0	burst size cb: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
9-8	W/R	0	burst size y: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
7	W/R	0	start frame manual : pulse bit
6	W/R	0	chroma repeat last1
5	W/R	0	Reserved
4	W/R	0	little endian: 0=Pixels are big-endian in memory; 1=Pixel are little-endian in memory
3	W/R	0	chroma hz avg: 0= output pixel by pixel per line; 1= output half line ,average between every 2 pixels
2	W/R	0	luma_hz_avg: 0= output pixel by pixel per line; 1= output half line ,average between every 2 pixels
1	W/R	0	separate_en: Set to 1 to use 3 separate FIFO's
0	W/R	0	enable

DI_IF1_GEN_REG2 0x1790

Bit(s)	R/W	Default	Description
25-14	W/R	0	shift pat cr
17-16	W/R	0	shift pat cb
9-8	W/R	0	shift pat y
6	W/R	0	hold_lines[6]
3	W/R	0	y_rev: X read direction: 0=default ,normal read; 1=reverse read
2	W/R	0	x_rev: Y read direction: 0=default ,normal read; 1=reverse read
1-0	W/R	0	color map: 0=default color map as defined by "bytes per pixel"; 1=NV12(CbCr); 2=NV21(CrCb)

DI_IF1_CANVAS0 0x17E9

Bit(s)	R/W	Default	Description
31	W/R	0	canvas addr syncen
23-16	W/R	0	canvas addr2
15-8	W/R	0	canvas addr1
7-0	W/R	0	canvas addr0

DI_IF1_LUMA_X0 0x17EA

Bit(s)	R/W	Default	Description
30-16	W/R	0	luma_x_end
14-0	W/R	0	luma_x_start

DI_IF1_LUMA_Y0 0x17EB

Bit(s)	R/W	Default	Description
28-16	W/R	0	luma_y_end
12-0	W/R	0	luma_y_start

DI_IF1_CHROMA_X0 0x17EC

Bit(s)	R/W	Default	Description
30-16	W/R	0	chroma_x_end
14-0	W/R	0	chroma_x_start

DI_IF1_CHROMA_Y0 0x17ED

Bit(s)	R/W	Default	Description
28-16	W/R	0	chroma_y_end
12-0	W/R	0	chroma_y_start

DI_IF1_RPT_LOOP 0x17EE

Bit(s)	R/W	Default	Description
15-8	W/R	0	chroma repeat loop
7-0	W/R	0	luma repeat loop

DI_IF1_LUMA0_RPT_PAT 0x17EF

Bit(s)	R/W	Default	Description
31-0	W/R	0	luma repeat pattern

DI_IF1_CHROMA0_RPT_PAT 0x17F0

Bit(s)	R/W	Default	Description
7-0	W/R	0	chroma repeat loop

DI_IF1_DUMMY_PIXEL 0x17F1

Bit(s)	R/W	Default	Description
31-0	W/R	0x808000	dummy pixel

DI_IF1_LUMA_FIFO_SIZE 0x17F2

Bit(s)	R/W	Default	Description
8-0	W/R	128	fifo size

DI_IF1_RANGE_MAP_Y 0x17FC

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_IF1_RANGE_MAP_CB 0x17FD

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_IF1_RANGE_MAP_CR 0x17FE

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_IF1_URGENT_CTRL 0x17A3

Bit(s)	R/W	Default	Description
13-16	W/R	0	urgent_ctrl_luma: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold
15-0	W/R	0	urgent_ctrl_chroma: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold

DI_IF1_FMT_CTRL 0x17F3

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 generete 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
17	R/W	0	veritcal formatter dont need repeat line on phase0, 1: enable, 0: disable
16	R/W	0	veritcal 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

DI_IF1_FMT_W 0x17F4

Bit(s)	R/W	Default	Description
27-16	R/W	0	horizontal formatter width
12-0	R/W	0	vertical formatter width

DI_INP_GEN_REG 0x17CE

Bit(s)	R/W	Default	Description
31	W/R	0	enable free clk
30	W/R	0	sw reset : pulse bit

Bit(s)	R/W	Default	Description
29	W/R	0	reset on go field
28	W/R	0	urgent chroma
27	W/R	0	urgent luma
26	W/R	0	chroma end at last line : 0 = read last line or push dummy after last line; 1 = stop read after last line
25	W/R	0	luma end at last line : 0 = read last line or push dummy after last line; 1 = stop read after last line
24-19	W/R	4	hold line[5:0], see GEN_REG2[6]
18	W/R	1	last line mode: 0 = read last line; 1 = push fixed value
16	W/R	0	demux mode: 0 = 4:2:2 demux; 1 = RGB demuxing from a single FIFO
15-14	W/R	0	bytes per pixel : 0= 1byte per pixel; 1 = 2 bytes per pixel; 2 = 3bytes per pixel
13-12	W/R	0	burst size cr: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
11-10	W/R	0	burst size cb: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
9-8	W/R	0	burst size y: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
7	W/R	0	start frame manual : pulse bit
6	W/R	0	chroma repeat last1
5	W/R	0	Reserved
4	W/R	0	little endian: 0=Pixels are big-endian in memory; 1=Pixel are little-endian in memory
3	W/R	0	chroma hz avg: 0= output pixel by pixel per line; 1= output half line ,average between every 2 pixels
2	W/R	0	luma_hz_avg: 0= output pixel by pixel per line; 1= output half line ,average between every 2 pixels
1	W/R	0	separate_en: Set to 1 to use 3 separate FIFO's
0	W/R	0	enable

DI_INP_GEN_REG2 0x1791

Bit(s)	R/W	Default	Description
25-14	W/R	0	shift pat cr
17-16	W/R	0	shift pat cb
9-8	W/R	0	shift pat y
6	W/R	0	hold_lines[6]
3	W/R	0	y_rev: X read direction: 0=default ,normal read; 1=reverse read
2	W/R	0	x_rev: Y read direction: 0=default ,normal read; 1=reverse read
1-0	W/R	0	color map: 0=default color map as defined by "bytes per pixel"; 1=NV12(CbCr); 2=NV21(CrCb)

DI_INP_CANVAS0 0x17CF

Bit(s)	R/W	Default	Description
31	W/R	0	canvas addr syncen
23-16	W/R	0	canvas addr2
15-8	W/R	0	canvas addr1
7-0	W/R	0	canvas addr0

DI_INP_LUMA_X0 0x17D0

Bit(s)	R/W	Default	Description
30-16	W/R	0	luma_x_end
14-0	W/R	0	luma_x_start

DI_INP_LUMA_Y0 0x17D1

Bit(s)	R/W	Default	Description
28-16	W/R	0	luma_y_end
12-0	W/R	0	luma_y_start

DI_INP_CHROMA_X0 0x17D2

Bit(s)	R/W	Default	Description
30-16	W/R	0	chroma_x_end
14-0	W/R	0	chroma_x_start

DI_INP_CHROMA_Y0 0x17D3

Bit(s)	R/W	Default	Description
28-16	W/R	0	chroma_y_end

Bit(s)	R/W	Default	Description
12-0	W/R	0	chroma_y_start

DI_INP_RPT_LOOP 0x17D4

Bit(s)	R/W	Default	Description
15-8	W/R	0	chroma repeat loop
7-0	W/R	0	luma repeat loop

DI_INP_LUMA0_RPT_PAT 0x17D5

Bit(s)	R/W	Default	Description
31-0	W/R	0	luma repeat pattern

DI_INP_CHROMA0_RPT_PAT 0x17D6

Bit(s)	R/W	Default	Description
7-0	W/R	0	chroma repeat loop

DI_INP_DUMMY_PIXEL 0x17D7

Bit(s)	R/W	Default	Description
31-0	W/R	0x808000	dummy pixel

DI_INP_LUMA_FIFO_SIZE 0x17D8

Bit(s)	R/W	Default	Description
8-0	W/R	128	fifo size

DI_INP_RANGE_MAP_Y 0x17BA

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_INP_RANGE_MAP_CB 0x17BB

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_INP_RANGE_MAP_CR 0x17BC

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_INP_URGENT_CTRL 0x17A4

Bit(s)	R/W	Default	Description
13-16	W/R	0	urgent_ctrl_luma: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold
15-0	W/R	0	urgent_ctrl_chroma:

Bit(s)	R/W	Default	Description
			bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold

DI_INP_FMT_CTRL 0x17D9

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 generete 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
17	R/W	0	veritcal formatter dont need repeat line on phase0, 1: enable, 0: disable
16	R/W	0	veritcal 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

DI_INP_FMT_W 0x17DA

Bit(s)	R/W	Default	Description
27-16	R/W	0	horizontal formatter width
12-0	R/W	0	vertical formatter width

DI_MEM_GEN_REG 0x17DB

Bit(s)	R/W	Default	Description
31	W/R	0	enable free clk
30	W/R	0	sw reset : pulse bit
29	W/R	0	reset on go field
28	W/R	0	urgent chroma
27	W/R	0	urgent luma
26	W/R	0	chroma end at last line : 0 = read last line or push dummy after last line; 1 = stop read after last line
25	W/R	0	luma end at last line : 0 = read last line or push dummy after last line; 1 = stop read after last line
24-19	W/R	4	hold line[5:0], see GEN_REG2[6]
18	W/R	1	last line mode: 0 = read last line; 1 = push fixed value
16	W/R	0	demux mode: 0 = 4:2:2 demux; 1 = RGB demuxing from a single FIFO
15-14	W/R	0	bytes per pixel : 0= 1byte per pixel; 1 = 2 bytes per pixel; 2 = 3bytes per pixel
13-12	W/R	0	burst size cr: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
11-10	W/R	0	burst size cb: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
9-8	W/R	0	burst size y: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
7	W/R	0	start frame manual : pulse bit
6	W/R	0	chroma repeat last1
5	W/R	0	Reserved
4	W/R	0	little endian: 0=Pixels are big-endian in memory; 1=Pixel are little-endian in memory
3	W/R	0	chroma hz avg: 0= output pixel by pixel per line; 1= output half line ,average between every 2 pixels
2	W/R	0	luma_hz_avg: 0= output pixel by pixel per line; 1= output half line ,average between every 2 pixels
1	W/R	0	separate_en: Set to 1 to use 3 separate FIFO's
0	W/R	0	enable

DI_MEM_GEN_REG2 0x1792

Bit(s)	R/W	Default	Description
25-14	W/R	0	shift pat cr

Bit(s)	R/W	Default	Description
17-16	W/R	0	shift pat cb
9-8	W/R	0	shift pat y
6	W/R	0	hold_lines[6]
3	W/R	0	y_rev: X read direction: 0=default ,normal read; 1=reverse read
2	W/R	0	x_rev: Y read direction: 0=default ,normal read; 1=reverse read
1-0	W/R	0	color map: 0=default color map as defined by "bytes per pixel"; 1=NV12(CbCr); 2=NV21(CrCb)

DI_MEM_CANVAS0 0x17DC

Bit(s)	R/W	Default	Description
31	W/R	0	canvas addr syncen
23-16	W/R	0	canvas addr2
15-8	W/R	0	canvas addr1
7-0	W/R	0	canvas addr0

DI_MEM_LUMA_X0 0x17DD

Bit(s)	R/W	Default	Description
30-16	W/R	0	luma_x_end
14-0	W/R	0	luma_x_start

DI_MEM_LUMA_Y0 0x17DE

Bit(s)	R/W	Default	Description
28-16	W/R	0	luma_y_end
12-0	W/R	0	luma_y_start

DI_MEM_CHROMA_X0 0x17DF

Bit(s)	R/W	Default	Description
30-16	W/R	0	chroma_x_end
14-0	W/R	0	chroma_x_start

DI_MEM_CHROMA_Y0 0x17E0

Bit(s)	R/W	Default	Description
28-16	W/R	0	chroma_y_end
12-0	W/R	0	chroma_y_start

DI_MEM_RPT_LOOP 0x17E1

Bit(s)	R/W	Default	Description
15-8	W/R	0	chroma repeat loop
7-0	W/R	0	luma repeat loop

DI_MEM_LUMA0_RPT_PAT 0x17E2

Bit(s)	R/W	Default	Description
31-0	W/R	0	luma repeat pattern

DI_MEM_CHROMA0_RPT_PAT 0x17E3

Bit(s)	R/W	Default	Description
7-0	W/R	0	chroma repeat loop

DI_MEM_DUMMY_PIXEL 0x17E4

Bit(s)	R/W	Default	Description
31-0	W/R	0x808000	dummy pixel

DI_MEM_LUMA_FIFO_SIZE 0x17E5

Bit(s)	R/W	Default	Description
8-0	W/R	128	fifo size

DI_MEM_RANGE_MAP_Y 0x17BD

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_MEM_RANGE_MAP_CB 0x17BE

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_MEM_RANGE_MAP_CR 0x17BF

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_MEM_URGENT_CTRL 0x17A5

Bit(s)	R/W	Default	Description
13-16	W/R	0	urgent_ctrl_luma: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold
15-0	W/R	0	urgent_ctrl_chroma: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold

DI_MEM_FMT_CTRL 0x17E6

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 generete 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
17	R/W	0	veritcal formatter dont need repeat line on phase0, 1: enable, 0: disable
16	R/W	0	veritcal 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

DI_MEM_FMT_W 0x17E7

Bit(s)	R/W	Default	Description
27-16	R/W	0	horizontal formatter width
12-0	R/W	0	vertical formatter width

DI_CHAN2_GEN_REG 0x17F5

Bit(s)	R/W	Default	Description
31	W/R	0	enable free clk
30	W/R	0	sw reset : pulse bit
29	W/R	0	reset on go field
28	W/R	0	urgent chroma
27	W/R	0	urgent luma
26	W/R	0	chroma end at last line : 0 = read last line or push dummy after last line; 1 = stop read after last line
25	W/R	0	luma end at last line : 0 = read last line or push dummy after last line; 1 = stop read after last line
24-19	W/R	4	hold line[5:0], see GEN_REG2[6]
18	W/R	1	last line mode: 0 = read last line; 1 = push fixed value
16	W/R	0	demux mode: 0 = 4:2:2 demux; 1 = RGB demuxing from a single FIFO
15-14	W/R	0	bytes per pixel : 0= 1byte per pixel; 1 = 2 bytes per pixel; 2 = 3bytes per pixel
13-12	W/R	0	burst size cr: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
11-10	W/R	0	burst size cb: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
9-8	W/R	0	burst size y: 0 = 24x64; 1 = 32x64; 2 = 48x64; 3 = 64x64
7	W/R	0	start frame manual : pulse bit
6	W/R	0	chroma repeat last1
5	W/R	0	Reserved
4	W/R	0	little endian: 0=Pixels are big-endian in memory; 1=Pixel are little-endian in memory
3	W/R	0	chroma hz avg: 0= output pixel by pixel per line; 1= output half line ,average between every 2 pixels
2	W/R	0	luma_hz_avg: 0= output pixel by pixel per line; 1= output half line ,average between every 2 pixels
1	W/R	0	separate_en: Set to 1 to use 3 separate FIFO's
0	W/R	0	enable

DI_CHAN2_GEN_REG2 0x17B7

Bit(s)	R/W	Default	Description
25-14	W/R	0	shift pat cr
17-16	W/R	0	shift pat cb
9-8	W/R	0	shift pat y
6	W/R	0	hold_lines[6]
3	W/R	0	y_rev: X read direction: 0=default ,normal read; 1=reverse read
2	W/R	0	x_rev: Y read direction: 0=default ,normal read; 1=reverse read
1-0	W/R	0	color map: 0=default color map as defined by "bytes per pixel"; 1=NV12(CbCr); 2=NV21(CrCb)

DI_CHAN2_CANVAS0 0x17F6

Bit(s)	R/W	Default	Description
31	W/R	0	canvas addr syncen
23-16	W/R	0	canvas addr2
15-8	W/R	0	canvas addr1
7-0	W/R	0	canvas addr0

DI_CHAN2_LUMA_X0 0x17F7

Bit(s)	R/W	Default	Description
30-16	W/R	0	luma_x_end
14-0	W/R	0	luma_x_start

DI_CHAN2_LUMA_Y0 0x17F8

Bit(s)	R/W	Default	Description
28-16	W/R	0	luma_y_end
12-0	W/R	0	luma_y_start

DI_CHAN2_CHROMA_X0 0x17F9

Bit(s)	R/W	Default	Description
30-16	W/R	0	chroma_x_end
14-0	W/R	0	chroma_x_start

DI_CHAN2_CHROMA_Y0 0x17FA

Bit(s)	R/W	Default	Description
28-16	W/R	0	chroma_y_end
12-0	W/R	0	chroma_y_start

DI_CHAN2_RPT_LOOP 0x17FB

Bit(s)	R/W	Default	Description
15-8	W/R	0	chroma repeat loop
7-0	W/R	0	luma repeat loop

DI_CHAN2_LUMA0_RPT_PAT 0x17B0

Bit(s)	R/W	Default	Description
31-0	W/R	0	luma repeat pattern

DI_CHAN2_CHROMA0_RPT_PAT 0x17B1

Bit(s)	R/W	Default	Description
7-0	W/R	0	chroma repeat loop

DI_CHAN2_DUMMY_PIXEL 0x17B2

Bit(s)	R/W	Default	Description
31-0	W/R	0x808000	dummy pixel

DI_CHAN2_LUMA_FIFO_SIZE 0x17B3

Bit(s)	R/W	Default	Description
8-0	W/R	128	fifo size

DI_CHAN2_RANGE_MAP_Y 0x17B4

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_CHAN2_RANGE_MAP_CB 0x17B5

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_CHAN2_RANGE_MAP_CR 0x17B6

Bit(s)	R/W	Default	Description
31-23	W/R	0	din offset
22-15	W/R	0	range map coef
13-10	W/R	0	range map din offset mult
9-1	W/R	0	dout offset
0	W/R	0	range map enable

DI_CHAN2_URGENT_CTRL 0x17A6

Bit(s)	R/W	Default	Description
13-16	W/R	0	urgent_ctrl_luma: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold
15-0	W/R	0	urgent_ctrl_chroma: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold

DI_CHAN2_FMT_CTRL 0x17B8

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 generete 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
17	R/W	0	veritcal formatter dont need repeat line on phase0, 1: enable, 0: disable
16	R/W	0	veritcal 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

DI_CHAN2_FMT_W 0x17B9

Bit(s)	R/W	Default	Description
27-16	R/W	0	horizontal formatter width
12-0	R/W	0	vertical formatter width

DI_NRWR_CTRL 0x17C2

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
16	R/W	0	Nrwr_urgent
15-8	R/W	0	Nrwr_canvas_index_chroma
7-0	R/W	0	Nrwr_canvas_index_luma

DI_NRWR_X 0x17C0

Bit(s)	R/W	Default	Description
31	R/W	0	Nrwr_little_endian
30	R/W	0	Nrwr_rev_x
29-16	R/W	0	Nrwr_start_x
13-0			Nrwr_end_x

DI_NRWR_Y 0x17C1

Bit(s)	R/W	Default	Description
31-30	R/W	1	Nrwr_words_lim
29	R/W	0	Nrwr_rev_y
28-16	R/W	0	Nrwr_start_y
15	R/W	0	Nrwr_ext_en
14	R/W	1	Nrwr bit10 mode
12-0	R/W	0	Nrwr_end_y

DI_DIWR_CTRL 0x17C8

Bit(s)	R/W	Default	Description
31	R/W	0	Pending_ddr_wrrsp_Diwr
30	R/W	0	Diwr_reg_swap
29-26	R/W	0	Diwr_burst_lim
25	R/W	0	Diwr_canvas_syncen
24	R/W	0	Diwr_no_clk_gate
23-22	R/W	0	Diwr_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	Diwr_hconv_mode
19-18	R/W	0	Diwr_vconv_mode
17	R/W	0	Diwr_swap_cbcr
16	R/W	0	Diwr_urgent
15-8	R/W	0	Diwr_canvas_index_chroma
7-0	R/W	0	Diwr_canvas_index_luma

DI_DIWR_X 0x17C6

Bit(s)	R/W	Default	Description
31	R/W	0	Diwr_little_endian
30	R/W	0	Diwr_rev_x
29-16	R/W	0	Diwr_start_x
13-0			Diwr_end_x

DI_DIWR_Y 0x17C7

Bit(s)	R/W	Default	Description
31-30	R/W	1	Diwr_words_lim
29	R/W	0	Diwr_rev_y
28-16	R/W	0	Diwr_start_y
15	R/W	0	Diwr_ext_en
14	R/W	1	Diwr bit10 mode
12-0	R/W	0	Diwr_end_y

DI_CONTWR_X 0x17A0

Bit(s)	R/W	Default	Description
30	R/W	0	rev_x
29-16	R/W	0	start_x
13-0	R/W	0	end_x

DI_CONTWR_Y 0x17A1

Bit(s)	R/W	Default	Description
29	R/W	0	rev_x
28-16	R/W	0	start_x
12-0	R/W	0	end_x

DI_CONTWR_CTRL 0x17A2

Bit(s)	R/W	Default	Description
31	R/W	0	write : clear wrrsp; read : Pending_ddr_wrrsp
11	R/W	0	canvas sync_enable
10	R/W	0	bits per pixel
8	R/W	0	urgent
7-0	R/W	0	canvas_index

DI_MTNWR_X 0x17C3

Bit(s)	R/W	Default	Description
30	R/W	0	rev_x
29-16	R/W	0	start_x
13-0	R/W	0	end_x

DI_MTNWR_Y 0x17C4

Bit(s)	R/W	Default	Description
29	R/W	0	rev_x
28-16	R/W	0	start_x
12-0	R/W	0	end_x

DI_MTNWR_CTRL 0x17C5

Bit(s)	R/W	Default	Description
31	R/W	0	write : clear wrrsp; read : Pending_ddr_wrrsp
11	R/W	0	canvas sync_enable
10	R/W	0	bits per pixel
8	R/W	0	urgent
7-0	R/W	0	canvas_index

DI_MTNRD_X 0x17CB

Bit(s)	R/W	Default	Description
29-16	R/W	0	start_x
13-0	R/W	0	end_x

DI_MTNRD_Y 0x17CC

Bit(s)	R/W	Default	Description
28-16	R/W	0	start_x
12-0	R/W	0	end_x

DI_MTNRD_CTRL 0x17CD

Bit(s)	R/W	Default	Description
23	R/W	0	canvas sync_enable
22	R/W	0	bits per pixel
18	R/W	0	x flip
17	R/W	0	y flip
16	R/W	0	urgent
15-8	R/W	0	canvas_index

DI_CONTRD_X 0x17A3

Bit(s)	R/W	Default	Description
29-16	R/W	0	start_x
13-0	R/W	0	end_x

DI_CONTRD_Y 0x17A4

Bit(s)	R/W	Default	Description
28-16	R/W	0	start_x
12-0	R/W	0	end_x

DI_CONTRD2_X 0x17A5

Bit(s)	R/W	Default	Description
29-16	R/W	0	start_x
13-0	R/W	0	end_x

DI_CONTRD2_Y 0x17A6

Bit(s)	R/W	Default	Description
28-16	R/W	0	start_x
12-0	R/W	0	end_x

DI_CONTRD_CTRL 0x17A7

Bit(s)	R/W	Default	Description
24	R/W	0	canvas sync_enable
23	R/W	0	contp bits per pixel
22	R/W	0	contp2 bits per pixel
18	R/W	0	contp x flip
17	R/W	0	contp y flip
18	R/W	0	contp2 x flip
17	R/W	0	contp2 y flip
16	R/W	0	urgent
15-8	R/W	0	contp2 canvas_index
7-0	R/W	0	contp canvas_index

MCDI_MCVECWR_X 0x2f92

Bit(s)	R/W	Default	Description
29	R/W	0	rev_x
28-16	R/W	0	start_x
12-0	R/W	0	end_x

MCDI_MCVECWR_Y 0x2f93

Bit(s)	R/W	Default	Description
29	R/W	0	rev_y
28-16	R/W	0	start_y
12-0	R/W	0	end_y

MCDI_MCVECWR_CTRL 0x2f94

Bit(s)	R/W	Default	Description
31-16	R/W	0	urgent_ctrl: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold
14	R/W	0	canvas sync enable
13	R/W	0	canvas_wr
12	R/W	0	req_en
11	R/W	0	fix disable
10	R/W	0	clear wrrsp
9	R/W	0	no clk gate
8	R/W	0	urgent
7-0	R/W	0	canvas index

MCDI_MCVECRD_X 0x2f95

Bit(s)	R/W	Default	Description
30	R/W	0	x flip
28-16	R/W	0	start_x
12-0	R/W	0	end_x

MCDI_MCVECRD_Y 0x2f96

Bit(s)	R/W	Default	Description
30	R/W	0	y flip
28-16	R/W	0	start_y
12-0	R/W	0	end_y

MCDI_MCVECRD_CTRL 0x2f97

Bit(s)	R/W	Default	Description
31-16	R/W	0	urgent_ctrl: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold
10	R/W	0	canvas sync enable
9	R/W	0	req_en
8	R/W	0	urgent
7-0	R/W	0	canvas index

MCDI_MCINFOWR_X 0x2f98

Bit(s)	R/W	Default	Description
29	R/W	0	rev_x
28-16	R/W	0	start_x
12-0	R/W	0	end_x

MCDI_MCINFOWR_Y 0x2f99

Bit(s)	R/W	Default	Description
29	R/W	0	rev_y
28-16	R/W	0	start_y
12-0	R/W	0	end_y

MCDI_MCINFOWR_CTRL 0x2f9a

Bit(s)	R/W	Default	Description
31-16	R/W	0	urgent_ctrl: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold
14	R/W	0	canvas sync enable
13	R/W	0	canvas_wr
12	R/W	0	req_en
11	R/W	0	fix disable
10	R/W	0	clear wrrsp
9	R/W	0	no clk gate
8	R/W	0	urgent
7-0	R/W	0	canvas index

MCDI_MCINFORD_X 0x2f9b

Bit(s)	R/W	Default	Description
30	R/W	0	x flip
28-16	R/W	0	start_x
12-0	R/W	0	end_x

MCDI_MCINFORD_Y 0x2f9c

Bit(s)	R/W	Default	Description
30	R/W	0	y flip
28-16	R/W	0	start_y
12-0	R/W	0	end_y

MCDI_MCINFORD_CTRL 0x2f9d

Bit(s)	R/W	Default	Description
31-16	R/W	0	urgent_ctrl: bit 15: auto urgent_en bit 14: urgent_wr bit 7-4: up_threshold bit 3-0: down_threshold
10	R/W	0	canvas sync enable
9	R/W	0	req_en
8	R/W	0	urgent
7-0	R/W	0	canvas index

MCDI_MCVECWR_CANVAS_SIZE 0x2f65

Bit(s)	R/W	Default	Description
28-16	R/W	0	hsizem1
12-0	R.W	0	vsizem1

MCDI_MCVECRD_CANVAS_SIZE 0x2f66

Bit(s)	R/W	Default	Description
28-16	R/W	0	hsizem1
12-0	R.W	0	vsizem1

MCDI_MCINFOWR_CANVAS_SIZE 0x2f67

Bit(s)	R/W	Default	Description
28-16	R/W	0	hsizem1
12-0	R.W	0	vsizem1

MCDI_MCINFORD_CANVAS_SIZE 0x2f68

Bit(s)	R/W	Default	Description
28-16	R/W	0	hsizem1
12-0	R.W	0	vsizem1

De-Interlace registers

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
27:26	W/R		mode_444c422
25	W/R		di_cont_read_en
24:23	W/R		mode_422c444
22	W/R		mtn_after_nr
21:16	W/R		pre_hold_fifo_lines
15	W/R		nr_wr_by
14	W/R		use_vdin_go_line
13	W/R		di_prevdin_en
12	W/R		di_pre_viu_link
11	W/R		di_pre_repeat
10	W/R		di_pre_drop_1st
9	W/R		di_buf2_en
8	W/R		di_chan2_en
7	W/R		prenr_hist_en
6	W/R		chan2_hist_en
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

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
21:16	W/R		post_hold_fifo_lines
13	W/R		prepost_link
12	W/R		di_post_viu_link
11	W/R		di_post_repeat
10	W/R		di_post_drop_1st
9	W/R		mif0_to_vpp_en
8	W/R		di_vpp_out_en
7	W/R		di_wr_bk_en
6	W/R		di_mux_en
5	W/R		di_blend_en
4	W/R		di_mtnp_read_en
3	W/R		di_mtn_buf_en
2	W/R		di_ei_en
1	W/R		di_buf1_en

Bit(s)	R/W	Default	Description
0	W/R		di_buf0_en

DI_POST_SIZE 0x1702

Bit(s)	R/W	Default	Description
28:16	W/R		vsize1post
12:0	W/R		hsize1post

DI_PRE_SIZE 0x1703

Bit(s)	R/W	Default	Description
28:16	W/R		vsize1pre
12:0	W/R		hsize1pre

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

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

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

DI_NR_CTRL0 0x1707

Bit(s)	R/W	Default	Description
26	W/R		nr_cue_en
25	W/R		nr2_en

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

DI_NR_CTRL2 0x1709

Bit(s)	R/W	Default	Description
29:24	W/R		mot_curtxtcore_clmt
21:16	W/R		mot_curtxtcore_ylmt
15:8	W/R		mot_curtxtcore_crate
7:0	W/R		mot_curtxtcore_yrate

DI_NR_CTRL3 0x170a

Bit(s)	R/W	Default	Description
No use	W/R		

DI_MTN_CTRL 0x170b

Bit(s)	R/W	Default	Description
No use	W/R		

DI_MTN_CTRL1 0x170c

Bit(s)	R/W	Default	Description
16	W/R		Invert pulldown field
15	W/R		Invert mcdi field
14	W/R		Swap line0 and line2 of mtn input data
13	W/R		me enable
12	W/R		me autoenable
11:8	W/R		mtn_paramtnthd
7:0	W/R		mtn_parafltthd

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
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
11:10	W/R		blend_reg1_mode
9:8	W/R		blend_reg0_mode

DI_BLEND_CTRL1 0x170e

Bit(s)	R/W	Default	Description
No use	W/R		

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

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

DI_BLEND_REG0_Y 0x1711

Bit(s)	R/W	Default	Description

DI_BLEND_REG1_X 0x1712

Bit(s)	R/W	Default	Description

DI_BLEND_REG1_Y 0x1713

Bit(s)	R/W	Default	Description

DI_BLEND_REG2_X 0x1714

Bit(s)	R/W	Default	Description

DI_BLEND_REG2_Y 0x1715

Bit(s)	R/W	Default	Description

DI_BLEND_REG3_X 0x1716

Bit(s)	R/W	Default	Description

DI_BLEND_REG3_Y 0x1717

Bit(s)	R/W	Default	Description

DI_CLKG_CTRL 0x1718

Bit(s)	R/W	Default	Description
31:24	W/R		pre_gclk_ctrl no clk gate control. if ==1, module clk is not gated (always on). [3] for pulldown,[2] for mtn_1,[1] for mtn_0,[0] for nr [1] for clk_me enable, [7] for clk_div enable
23:16	W/R		post_gclk_ctrl no clk gate control. [4] for ei_1, [3] for ei_0,[2] for ei_top, [1] for blend_1, [0] for blend_0 [5] for clk_post_div enable
1	W/R		di_gate_all clk shut down. if ==1 , all di clock shut down
0	W/R		di_no_clk_gate no clk gate control. if di_gated_all==0 and di_no_clk_gate ==1, all di clock is always working.

DI_EI_CTRL4 0x171a

Bit(s)	R/W	Default	Description
29	W/R	0	reg_ei_caldrt_amblike2_biasvertical
28:24	W/R	21	reg_ei_caldrt_addxla2list_drtmax
23	W/R	0	N/A
22:20	W/R	1	reg_ei_caldrt_addxla2list_signm0th
19	W/R	1	reg_ei_caldrt_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

DI_EI_CTRL5 0x171b

Bit(s)	R/W	Default	Description
30:28	W/R	5	reg_ei_caldrt_cnflcctchk_frcverthrd
27	W/R	0	N/A
26:24	W/R	2	reg_ei_caldrt_cnflctchk_mg
23:22	W/R	1	reg_ei_caldrt_cnflctchk_ws
21	W/R	1	reg_ei_caldrt_cnflctchk_en
20	W/R	1	reg_ei_caldrt_verfrc_final_en
19	W/R	0	reg_ei_caldrt_verfrc_retimflt_en
18:16	W/R	3	reg_ei_caldrt_verftc_eithratemth
15	W/R	0	reg_ei_caldrt_verfrc_retiming_en
14:12	W/R	2	reg_ei_caldrt_verfrc_bothratemth

Bit(s)	R/W	Default	Description
11:9	W/R	0	reg_ei_caldrt_ver_thrd
8:4	W/R	4	reg_ei_caldrt_addxla2list_drtmin
3:0	W/R	15	reg_ei_caldrt_addxla2list_drtlimit

DI_EI_CTRL6 0x171c

Bit(s)	R/W	Default	Description
31:24	W/R	80	reg_ei_caldrt_abext_sad12thhig
23:16	W/R	35	reg_ei_caldrt_abext_sad00thlow
15:8	W/R	28	reg_ei_caldrt_abext_sad12thlow
6:4	W/R	1	reg_ei_caldrt_abext_ratemth
2:0	W/R	5	reg_ei_caldrt_abext_drtthrd

DI_EI_CTRL7 0x171d

Bit(s)	R/W	Default	Description
29	W/R	1	reg_ei_caldrt_xlanopeak_codien
28:24	W/R	15	reg_ei_caldrt_xlanopeak_drtmax
23	W/R	1	reg_ei_caldrt_xlanopeak_en
22:20	W/R	3	reg_ei_caldrt_abext_monotrnd_alpha
19:18	W/R	1	reg_ei_caldrt_abext_mononum12_thrd
17:16	W/R	1	reg_ei_caldrt_abext_mononum00_thrd
15:12	W/R	6	reg_ei_caldrt_abext_sad00rate
11:8	W/R	6	reg_ei_caldrt_abext_sad12rate
7:0	W/R	80	reg_ei_caldrt_abext_sad00thhig

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_lastcurpncnfltchk_mode
22:21	W/R	0	reg_ei_retime_lastcurpncnfltchk_drtth
13:11	W/R	3	reg_ei_caldrt_amblike2_drtmg
10:8	W/R	1	reg_ei_caldrt_amblike2_valmg
7:4	W/R	10	reg_ei_caldrt_amblike2_alpha
3:0	W/R	4	reg_ei_caldrt_amblike2_drtth

DI_EI_CTRL9 0x171f

Bit(s)	R/W	Default	Description
31:28	W/R	7	reg_ei_caldrt_hcnfcheck_frcvert_xla_th3
27	W/R	1	reg_ei_caldrt_hcnfcheck_frcvert_xla_en
26:24	W/R	4	reg_ei_caldrt_conf_drtth
23:20	W/R	11	reg_ei_caldrt_conf_absdrtth
19:18	W/R	2	reg_ei_caldrt_abcheck_mode1
17:16	W/R	1	reg_ei_caldrt_abcheck_mode0
15:12	W/R	11	reg_ei_caldrt_abcheck_drtth1
11:8	W/R	11	reg_ei_caldrt_abcheck_drtth0
6:4	W/R	3	reg_ei_caldrt_abpnchk1_th
1	W/R	1	reg_ei_caldrt_abpnchk1_en
0	W/R	1	reg_ei_caldrt_abpnchk0_en

DI_EI_CTRL10 0x1793

Bit(s)	R/W	Default	Description
31:28	W/R	0	reg_ei_caldrt_hstrrgchk_drtth
27:24	W/R	8	reg_ei_caldrt_hstrrgchk_frcverthrd
23:20	W/R	4	reg_ei_caldrt_hstrrgchk_mg
19	W/R	0	reg_ei_caldrt_hstrrgchk_1sidnul
18	W/R	0	reg_ei_caldrt_hstrrgchk_excpcnf
17:16	W/R	2	reg_ei_caldrt_hstrrgchk_ws

Bit(s)	R/W	Default	Description
15	W/R	1	reg_ei_caldrt_hstrrgchk_en
14:13	W/R	2	reg_ei_caldrt_hpncheck_mode
12	W/R	0	reg_ei_caldrt_hpncheck_mute
11:9	W/R	3	reg_ei_caldrt_hcnfcheck_mg2
8:6	W/R	2	reg_ei_caldrt_hcnfcheck_mg1
5:4	W/R	2	reg_ei_caldrt_hcnfcheck_mode
3:0	W/R	9	reg_ei_caldrt_hcnfcheck_frcvert_xla_th5

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_lastmappncnfltchk_drtth
18:16	W/R	2	reg_ei_retime_lastmappncnfltchk_mode
15:14	W/R	2	reg_ei_retime_lastmapvertfrcchk_mode
13:12	W/R	3	reg_ei_retime_lastvertfrcchk_mode
11:8	W/R	0	reg_ei_retime_lastpnchk_drtth
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_hstrvertfrcchk_en
0	W/R	0	reg_ei_caldrt_hstrrgchk_mode

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_drtth
3:0	W/R	3	reg_ei_flatmsad_thrd

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

DI_EI_XWIN0 0x1798

Bit(s)	R/W	Default	Description
27:16	W/R		ei_xend0
11:0	W/R		ei_xstart0

DI_EI_XWIN1 0x1799

Bit(s)	R/W	Default	Description

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

DI_MC_REG0_Y 0x1721

Bit(s)	R/W	Default	Description

DI_MC_REG1_X 0x1722

Bit(s)	R/W	Default	Description

DI_MC_REG1_Y 0x1723

Bit(s)	R/W	Default	Description

DI_MC_REG2_X 0x1724

Bit(s)	R/W	Default	Description

DI_MC_REG2_Y 0x1725

Bit(s)	R/W	Default	Description

DI_MC_REG3_X 0x1726

Bit(s)	R/W	Default	Description

DI_MC_REG3_Y 0x1727

Bit(s)	R/W	Default	Description

DI_MC_REG4_X 0x1728

Bit(s)	R/W	Default	Description

DI_MC_REG4_Y 0x1729

Bit(s)	R/W	Default	Description

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

DI_MC_32LVL1 0x172b

Bit(s)	R/W	Default	Description
15:8	W/R		mc_reg3_32lvl
7:0	W/R		mc_reg4_32lvl

DI_MC_22LVL0 0x172c

Bit(s)	R/W	Default	Description
31:16	W/R		mc_reg0_22lvl
15:0	W/R		field_22lvl

DI_MC_22LVL1 0x172d

Bit(s)	R/W	Default	Description
31:16	W/R		mc_reg2_22lvl
15:0	W/R		mc_reg1_22lvl

DI_MC_22LVL2 0x172e

Bit(s)	R/W	Default	Description
31:16	W/R		mc_reg4_22lvl
15:0	W/R		mc_reg3_22lvl

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

DI_INTR_CTRL 0x1730

Bit(s)	R/W	Default	Description
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
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

Addr_0

Bit(s)	R/W	Default	Description
31:0	R		Field_32p , sum of difference between n-2 and n

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

Addr_2

Bit(s)	R/W	Default	Description
31:0	R		Field_22p, sum of difference between temporal and vertical difference

Addr_3

Bit(s)	R/W	Default	Description
15:0	R		Field_22max , maximum difference between temporal and verticaldifference

Addr_4

Bit(s)	R/W	Default	Description
23:0	R		Field_22num, pixel sum which difference > threshold

Addr_5

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

Addr_6

Bit(s)	R/W	Default	Description
31:0	R		Difference of 32, sum in area 0

Addr_7

Bit(s)	R/W	Default	Description
31:0	R		Difference of 32, sum in area 1

Addr_8

Bit(s)	R/W	Default	Description
31:0	R		Difference of 32, sum in area 2

Addr_9

Bit(s)	R/W	Default	Description
31:0	R		Difference of 32, sum in area 3

Addr_10

Bit(s)	R/W	Default	Description
31:0	R		Difference of 32, sum in area 4

Addr_11

Bit(s)	R/W	Default	Description
31:0	R		Difference of 22, sum in area 0

Addr_12

Bit(s)	R/W	Default	Description
31:0	R		Difference of 22, sum in area 1

Addr_13

Bit(s)	R/W	Default	Description
31:0	R		Difference of 22, sum in area 2

Addr_14

Bit(s)	R/W	Default	Description
31:0	R		Difference of 22, sum in area 3

Addr_15

Bit(s)	R/W	Default	Description
31:0	R		Difference of 22, sum in area 4

Addr_16

Bit(s)	R/W	Default	Description
31:0	R		luma, sum in area 0

Addr_17

Bit(s)	R/W	Default	Description
31:0	R		luma, sum in area 1

Addr_18

Bit(s)	R/W	Default	Description
31:0	R		luma, sum in area 2

Addr_19

Bit(s)	R/W	Default	Description
31:0	R		luma, sum in area 3

Addr_20

Bit(s)	R/W	Default	Description
31:0	R		luma, sum in area 4

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

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

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

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

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

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

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

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

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

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

DI_INFO_DATA 0x1732

Bit(s)	R/W	Default	Description

DI_PRE_HOLD 0x1733

Bit(s)	R/W	Default	Description

DI_MTN_1_CTRL1 0x1740

Bit(s)	R/W	Default	Description
31	W/R		reg_mtn_1_en
30	W/R		reg_mtn_init
29	W/R		reg_di2nr_txt_en
28	W/R		reg_di2nr_txt_mode
27:24	W/R		reg_mtn_def
23:16	W/R	32	reg_DI_cmb_adp_YCrate
15: 8	W/R	32	reg_DI_cmb_adp_2Crate
7: 0	W/R	21	reg_DI_cmb_adp_2Yrate

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

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
15: 8	W/R	10	reg_DI_m1b_core_mxcmbY
7: 0	W/R	10	reg_DI_m1b_core_mxcmbC

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

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_errod_num
15: 8	W/R	13	reg_DI_mot_core_Ykinter
7: 0	W/R	13	reg_DI_mot_core_Ckinter

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

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

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

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

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

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
4: 0	W/R	4	reg_DI_m1b_surestl_odd_th

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

NR2 Registers

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*MotY + (MotU + MotV))/4; 2: Max(MotY, MotU,MotV); 3:Max(MotY, (MotU+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.

DET3D_CB_CFG 0x1735

Bit(s)	R/W	Default	Description
7:4	R/W	0	reg_Det3D_ChessBd_NHV_ofst : U4, Noise immune offset for NON-Horizotnal or vertical combing detection.
3:0	R/W	0	reg_Det3D_ChessBd_HV_ofst : U4, Noise immune offset for Horizotnal or vertical combing detection.

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.

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.

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)

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)

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)

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

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.

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.

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.

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

NR2 REG DEFINE BEGIN////

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; ¡-X: 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.

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.
23:16	R/W	0x0	reg_NM_YHPmot_thrd_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_thrd_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.

NR2_MET_NM_CCTRL 0x1747

Bit(s)	R/W	Default	Description
31:28	R/W	0x0	reg_NM_CPLL_target : Target rate of NM_Cnoise_thrd to mean of the Chroma Noise
27:24	R/W	0x0	reg_NM_CLPmot_thrd : Chroma channel LP portion motion for condition of pixels included in Chroma Noise measurement.
23:16	R/W	0x0	reg_NM_CHPmot_thrd_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_thrd_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;

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.

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.

NR2_MET_NMFRM_TNR_YCNT 0x174a

Bit(s)	R/W	Default	Description
23:0	R.O	0x0	ro_NMFrm_TNR_Ycount : Number ofLuma channel pixels included in Frame/Field based noise level measurement.

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.

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.

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 :

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,

Bit(s)	R/W	Default	Description
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

NR2_SW_EN 0x174f

Bit(s)	R/W	Default	Description
17:8	R/W	0x0	Clk_gate_ctrl :
7	R/W	0x0	Cfr_enable :
5	R/W	0x0	Det3d_en :
4	R/W	0x0	Nr2_proc_en :
0	R/W	0x0	Nr2_sw_en :

NR2_FRM_SIZE 0x1750

Bit(s)	R/W	Default	Description
27:16	R/W	0x0	Frm_heigh : Frame/field height
11: 0	R/W	0x0	Frm_width : Frame/field width

NR2_SNR_SAD_CFG 0x1751

Bit(s)	R/W	Default	Description
12	R/W	0x0	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	0x0	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	0x0	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 2 1];
4:0	R/W	0x0	Sad_coef_num : Sad coeffient

NR2_MATNR_SNR_OS 0x1752

Bit(s)	R/W	Default	Description
7:4	R/W	0x0	reg_MATNR_SNR_COS : SNR Filter overshoot control margin for UV channel (X2 to u10 scale)
3:0	R/W	0x0	reg_MATNR_SNR_YOS : SNR Filter overshoot control margin for luma channel (X2 to u10 scale)

NR2_MATNR_SNR_NRM_CFG 0x1753

Bit(s)	R/W	Default	Description
23:16	R/W	0x0	reg_MATNR_SNR_NRM_ofst : Edge based SNR boosting normalization offset to SAD_max ;
15:8	R/W	0x0	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

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	0x0	reg_MATNR_SNR_NRM_Ygain : Edge based SNR boosting normalization Gain for Luma channel (norm 32 as 1)

NR2_MATNR_SNR_LPF_CFG 0x1755

Bit(s)	R/W	Default	Description
23:16	R/W	0x0	reg_MATNR_SNRLPF_SADmaxTH : U8, Threshold to SADmax to use TNRLPF to replace SNRLPF. i.e.if (SAD_max<reg_MATNR_SNRLPF_SADmaxTH) SNRLPF_yuv[k] = TNRLPF_yuv[k];
13:11	R/W	0x0	reg_MATNR_SNRLPF_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	0x0	reg_MATNR_SNRLPF_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[]

Bit(s)	R/W	Default	Description
7:4	R/W	0x0	reg_MATNR_SNRLPF_SADmin3TH : Offset threshold to SAD_min to Discard SAD_min3 corresponding pixel in LPF SNR filtering. (X8 to u8 scale)
3:0	R/W	0x0	reg_MATNR_SNRLPF_SADmin2TH : Offset threshold to SAD_min to Discard SAD_min2 corresponding pixel in LPF SNR filtering. (X8 to u8 scale)

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)

NR2_MATNR_SNR_EDGE2B 0x1757

Bit(s)	R/W	Default	Description
15:8	R/W	0x0	reg_MATNR_SNR_Edge2Beta_ofst : U8, Offset for Beta based on Edge.
7:0	R/W	0x0	reg_MATNR_SNR_Edge2Beta_gain : U8. Gain to SAD_min for Beta based on Edge. (norm 16 as 1)

NR2_MATNR_BETA_EGAIN 0x1758

Bit(s)	R/W	Default	Description
15:8	R/W	0x0	reg_MATNR_CBeta_Egain : U8, Gain to Edge based Beta for Chrm channel. (normalized to 32 as 1)
7:0	R/W	0x0	reg_MATNR_YBeta_Egain : U8, Gain to Edge based Beta for Luma channel. (normalized to 32 as 1)

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	0x0	reg_MATNR_beta_BRT_thrd_hi : U8, Beta adjustment based on Brightness high threshold.(u8 scale)
15:12	R/W	0x0	reg_MATNR_beta_BRT_limt_lo : U4, Beta adjustment based on Brightness low side Limit. (X16 to u8 scale)
11:8	R/W	0x0	reg_MATNR_beta_BRT_slop_lo : U4, Beta adjustment based on Brightness low side slope. Normalized to 16 as 1
7:0	R/W	0x0	reg_MATNR_beta_BRT_thrd_lo : U8, Beta adjustment based on Brightness low threshold.(u8 scale)

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;
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)

NR2_MATNR_YBETA_SCL 0x175b

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_YBeta_scale_min : U8, Final step Beta scale low limit for Luma channel;
23:16	R/W	0x0	reg_MATNR_YBeta_scale_max : U8, Final step Beta scale high limit for Luma channe;
15: 8	R/W	0x0	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 ;

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	0x0	reg_MATNR_CBeta_scale_max : U8, Final step Beta scale high limit for Chrm channel.Similar to Y
15: 8	R/W	0x0	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

NR2_SNR_MASK 0x175d

Bit(s)	R/W	Default	Description
20:0	R/W	0x0	SAD_MSK : Valid signal in the 3x7 SAD surface

NR2_SAD2NORM_LUT0 0x175e

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_SAD2Norm_LUT_3 : SAD convert normal LUT node 3
23:16	R/W	0x0	reg_MATNR_SAD2Norm_LUT_2 : SAD convert normal LUT node 2
15: 8	R/W	0x0	reg_MATNR_SAD2Norm_LUT_1 : SAD convert normal LUT node 1
7: 0	R/W	0x0	reg_MATNR_SAD2Norm_LUT_0 : SAD convert normal LUT node 0

NR2_SAD2NORM_LUT1 0x175f

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_SAD2Norm_LUT_7 : SAD convert normal LUT node 7
23:16	R/W	0x0	reg_MATNR_SAD2Norm_LUT_6 : SAD convert normal LUT node 6
15: 8	R/W	0x0	reg_MATNR_SAD2Norm_LUT_5 : SAD convert normal LUT node 5
7: 0	R/W	0x0	reg_MATNR_SAD2Norm_LUT_4 : SAD convert normal LUT node 4

NR2_SAD2NORM_LUT2 0x1760

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_SAD2Norm_LUT_11 : SAD convert normal LUT node 11
23:16	R/W	0x0	reg_MATNR_SAD2Norm_LUT_10 : SAD convert normal LUT node 10
15: 8	R/W	0x0	reg_MATNR_SAD2Norm_LUT_9 : SAD convert normal LUT node 9
7: 0	R/W	0x0	reg_MATNR_SAD2Norm_LUT_8 : SAD convert normal LUT node 8

NR2_SAD2NORM_LUT3 0x1761

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_SAD2Norm_LUT_15 : SAD convert normal LUT node 15
23:16	R/W	0x0	reg_MATNR_SAD2Norm_LUT_14 : SAD convert normal LUT node 14
15:8	R/W	0x0	reg_MATNR_SAD2Norm_LUT_13 : SAD convert normal LUT node 13
7:0	R/W	0x0	reg_MATNR_SAD2Norm_LUT_12 : SAD convert normal LUT node 12

NR2_EDGE2BETA_LUT0 0x1762

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_Edge2Beta_LUT_3 : Edge convert beta LUT node 3
23:16	R/W	0x0	reg_MATNR_Edge2Beta_LUT_2 : Edge convert beta LUT node 2
15: 8	R/W	0x0	reg_MATNR_Edge2Beta_LUT_1 : Edge convert beta LUT node 1
7: 0	R/W	0x0	reg_MATNR_Edge2Beta_LUT_0 : Edge convert beta LUT node 0

NR2_EDGE2BETA_LUT1 0x1763

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_Edge2Beta_LUT_7 : Edge convert beta LUT node 7
23:16	R/W	0x0	reg_MATNR_Edge2Beta_LUT_6 : Edge convert beta LUT node 6
15: 8	R/W	0x0	reg_MATNR_Edge2Beta_LUT_5 : Edge convert beta LUT node 5
7: 0	R/W	0x0	reg_MATNR_Edge2Beta_LUT_4 : Edge convert beta LUT node 4

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	0x0	reg_MATNR_Edge2Beta_LUT_8 : Edge convert beta LUT node 8

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

NR2_MOTION2BETA_LUT0 0x1766

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_Mot2Beta_LUT_3 : Motion convert beta LUT node 3
23:16	R/W	0x0	reg_MATNR_Mot2Beta_LUT_2 : Motion convert beta LUT node 2
15: 8	R/W	0x0	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

NR2_MOTION2BETA_LUT1 0x1767

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_Mot2Beta_LUT_7 : Motion convert beta LUT node 7
23:16	R/W	0x0	reg_MATNR_Mot2Beta_LUT_6 : Motion convert beta LUT node 6
15: 8	R/W	0x0	reg_MATNR_Mot2Beta_LUT_5 : Motion convert beta LUT node 5
7: 0	R/W	0x0	reg_MATNR_Mot2Beta_LUT_4 : Motion convert beta LUT node 4

NR2_MOTION2BETA_LUT2 0x1768

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_Mot2Beta_LUT_11 : Motion convert beta LUT node 11
23:16	R/W	0x0	reg_MATNR_Mot2Beta_LUT_10 : Motion convert beta LUT node 10
15: 8	R/W	0x0	reg_MATNR_Mot2Beta_LUT_9 : Motion convert beta LUT node 9
7: 0	R/W	0x0	reg_MATNR_Mot2Beta_LUT_8 : Motion convert beta LUT node 8

NR2_MOTION2BETA_LUT3 0x1769

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_Mot2Beta_LUT_15 : Motion convert beta LUT node 15
23:16	R/W	0x0	reg_MATNR_Mot2Beta_LUT_14 : Motion convert beta LUT node 14
15: 8	R/W	0x0	reg_MATNR_Mot2Beta_LUT_13 : Motion convert beta LUT node 13
7: 0	R/W	0x0	reg_MATNR_Mot2Beta_LUT_12 : Motion convert beta LUT node 12

NR2_MATNR_MTN_CRTL 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	0x0	reg_MATNR_mtn_txt_mode : Texture detection mode for adaptive coring of HP motion
9: 8	R/W	0x0	reg_MATNR_mtn_cor_mode : Coring selection mode based on texture detection;
6: 4	R/W	0x0	reg_MATNR_mtn_hpf_mode : video mode of current and previous frame/field for MotHPF_yuv[k] calculation:
2: 0	R/W	0x0	reg_MATNR_mtn_lpf_mode : LPF video mode of current and previous frame/field for MotLPF_yuv[k] calculation:

NR2_MATNR_MTN_CRTL2 0x176b

Bit(s)	R/W	Default	Description
18:16	R/W	0x0	reg_MATNR_iir_BS_Ymode : IIR TNR filter Band split filter mode for Luma LPF result generation (Cur and Prev);
15: 8	R/W	0x0	reg_MATNR_mtnb_alpLP_Cgain : Scale of motion_brthp_uv to motion_brtlp_uv, normalized to 32 as 1
7: 0	R/W	0x0	reg_MATNR_mtnb_alpLP_Ygain : Scale of motion_brthp_y to motion_brtlp_y, normalized to 32 as 1

NR2_MATNR_MTN_COR 0x176c

Bit(s)	R/W	Default	Description
15:12	R/W	0x0	reg_MATNR_mtn_cor_Cofst : Coring Offset for Chroma Motion.
11: 8	R/W	0x0	reg_MATNR_mtn_cor_Cgain : Gain to texture based coring for Chroma Motion. Normalized to 16 as 1
7: 4	R/W	0x0	reg_MATNR_mtn_cor_Yofst : Coring Offset for Luma Motion.
3: 0	R/W	0x0	reg_MATNR_mtn_cor_Ygain : Gain to texture based coring for Luma Motion. Normalized to 16 as 1

NR2_MATNR_MTN_GAIN 0x176d

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_mtn_hp_Cgain : Gain to MotHPF_yuv[k] Chrm channel for motion calculation, normalized to 64 as 1
23:16	R/W	0x0	reg_MATNR_mtn_hp_Ygain : Gain to MotHPF_yuv[k] Luma channel for motion calculation, normalized to 64 as 1

Bit(s)	R/W	Default	Description
15: 8	R/W	0x0	reg_MATNR_mtn_lp_Cgain : Gain to MotLPF_yuv[k] Chrm channel for motion calculation, normalized to 32 as 1
7: 0	R/W	0x0	reg_MATNR_mtn_lp_Ygain : Gain to MotLPF_yuv[k] Luma channel for motion calculation, normalized to 32 as 1

NR2_MATNR_DEGHOST 0x176e

Bit(s)	R/W	Default	Description
8	R/W	0x0	reg_MATNR_DeGhost_En : Enable signal for DeGhost function:0: disable; 1: enable
7:4	R/W	0x0	reg_MATNR_DeGhost_COS : DeGhost Overshoot margin for UV channel, (X2 to u10 scale)
3:0	R/W	0x0	reg_MATNR_DeGhost_YOS : DeGhost Overshoot margin for Luma channel, (X2 to u10 scale)

NR2_MATNR_ALPHALP_LUT0 0x176f

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_AlphaLP_LUT_3 : Matnr low-pass filter alpha LUT node 3
23:16	R/W	0x0	reg_MATNR_AlphaLP_LUT_2 : Matnr low-pass filter alpha LUT node 2
15: 8	R/W	0x0	reg_MATNR_AlphaLP_LUT_1 : Matnr low-pass filter alpha LUT node 1
7: 0	R/W	0x0	reg_MATNR_AlphaLP_LUT_0 : Matnr low-pass filter alpha LUT node 0

NR2_MATNR_ALPHALP_LUT1 0x1770

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_AlphaLP_LUT_7 : Matnr low-pass filter alpha LUT node 7
23:16	R/W	0x0	reg_MATNR_AlphaLP_LUT_6 : Matnr low-pass filter alpha LUT node 6
15: 8	R/W	0x0	reg_MATNR_AlphaLP_LUT_5 : Matnr low-pass filter alpha LUT node 5
7: 0	R/W	0x0	reg_MATNR_AlphaLP_LUT_4 : Matnr low-pass filter alpha LUT node 4

NR2_MATNR_ALPHALP_LUT2 0x1771

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_AlphaLP_LUT_11 : Matnr low-pass filter alpha LUT node 11
23:16	R/W	0x0	reg_MATNR_AlphaLP_LUT_10 : Matnr low-pass filter alpha LUT node 10
15: 8	R/W	0x0	reg_MATNR_AlphaLP_LUT_9 : Matnr low-pass filter alpha LUT node 9
7: 0	R/W	0x0	reg_MATNR_AlphaLP_LUT_8 : Matnr low-pass filter alpha LUT node 8

NR2_MATNR_ALPHALP_LUT3 0x1772

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_AlphaLP_LUT_15 : Matnr low-pass filter alpha LUT node 15
23:16	R/W	0x0	reg_MATNR_AlphaLP_LUT_14 : Matnr low-pass filter alpha LUT node 14
15: 8	R/W	0x0	reg_MATNR_AlphaLP_LUT_13 : Matnr low-pass filter alpha LUT node 13
7: 0	R/W	0x0	reg_MATNR_AlphaLP_LUT_12 : Matnr low-pass filter alpha LUT node 12

NR2_MATNR_ALPHAHP_LUT0 0x1773

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_AlphaHP_LUT_3 : Matnr high-pass filter alpha LUT node 3
23:16	R/W	0x0	reg_MATNR_AlphaHP_LUT_2 : Matnr high-pass filter alpha LUT node 2
15: 8	R/W	0x0	reg_MATNR_AlphaHP_LUT_1 : Matnr high-pass filter alpha LUT node 1
7: 0	R/W	0x0	reg_MATNR_AlphaHP_LUT_0 : Matnr high-pass filter alpha LUT node 0

NR2_MATNR_ALPHAHP_LUT1 0x1774

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_AlphaHP_LUT_7 : Matnr high-pass filter alpha LUT node 7
23:16	R/W	0x0	reg_MATNR_AlphaHP_LUT_6 : Matnr high-pass filter alpha LUT node 6
15: 8	R/W	0x0	reg_MATNR_AlphaHP_LUT_5 : Matnr high-pass filter alpha LUT node 5
7: 0	R/W	0x0	reg_MATNR_AlphaHP_LUT_4 : Matnr high-pass filter alpha LUT node 4

NR2_MATNR_ALPHAHP_LUT2 0x1775

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_AlphaHP_LUT_11 : Matnr high-pass filter alpha LUT node 11
23:16	R/W	0x0	reg_MATNR_AlphaHP_LUT_10 : Matnr high-pass filter alpha LUT node 10

Bit(s)	R/W	Default	Description
15: 8	R/W	0x0	reg_MATNR_AlphaHP_LUT_9 : Matnr high-pass filter alpha LUT node 9
7: 0	R/W	0x0	reg_MATNR_AlphaHP_LUT_8 : Matnr high-pass filter alpha LUT node 8

NR2_MATNR_ALPHAHP_LUT3 0x1776

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_MATNR_AlphaHP_LUT_15 : Matnr high-pass filter alpha LUT node 15
23:16	R/W	0x0	reg_MATNR_AlphaHP_LUT_14 : Matnr high-pass filter alpha LUT node 14
15: 8	R/W	0x0	reg_MATNR_AlphaHP_LUT_13 : Matnr high-pass filter alpha LUT node 13
7: 0	R/W	0x0	reg_MATNR_AlphaHP_LUT_12 : Matnr high-pass filter alpha LUT node 12

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	0x0	reg_MATNR_mtnb_BRT_thrd_hi : Motion adjustment based on Brightness high threshold.(u8 scale)
15:12	R/W	0x0	reg_MATNR_mtnb_BRT_limt_lo : Motion adjustment based on Brightness low side Limit. (X16 to u8 scale)
11: 8	R/W	0x0	reg_MATNR_mtnb_BRT_slop_lo : Motion adjustment based on Brightness low side slope. Normalized to 16 as 1
7: 0	R/W	0x0	reg_MATNR_mtnb_BRT_thrd_lo : Motion adjustment based on Brightness low threshold.(u8 scale)

NR2_CUE_MODE 0x1778

Bit(s)	R/W	Default	Description
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;
2:0	R/W	0x0	reg_CUE_CHRM_FLT_mode : U3, CUE improvement filter mode,

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.

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,

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_diflG and CUE_difEG
15: 8	R/W	0x0	reg_CUE_CON_difEG_thrd : U8, Theshold to the difference between current Field/Frame middle line to down line color channel(CUE_difEG).
7: 0	R/W	0x0	reg_CUE_CON_diflG_thrd : U8, Threshold to the difference between P1 field top line to current Field/Frame down line color channel (CUE_diflG).

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, Theshold to the difference between current Field/Frame middle line to up line color channel(CUE_difEC).

Bit(s)	R/W	Default	Description
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).

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),

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,

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

DET 3D REG DEFINE BEGIN //// 8 'h80~8'h8f

DET3D_RO_SPLT_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

DET3D_RO_SPLT_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

DET3D_RO_SPLT_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

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]

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),

Bit(s)	R/W	Default	Description
7:0	R.O	0x0	RO_Luma_TB_symtc : Luma statistics Top/Bottompure 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,

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,

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/Bottompure 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

DET3D_RO_MAT_CHRV_LR 0x1787

Bit(s)	R/W	Default	Description
15:0	R.O	0x0	RO_ChrV_LR_score : S2*8 CHRUstatistics 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

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/Bottompure 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

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

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/Bottompure 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

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

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/Bottompure 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

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)

Bit(s)	R/W	Default	Description
4:0	R.O	0x0	RO_Motn_LR_sum : Total score of 8x8 Motion statistics for LR like decision

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/Bottompure 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

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

NR2_CFR_PARA_CFG0 0x179c

Bit(s)	R/W	Default	Description
8	R/W	0x0	reg_CFR_CurDif_luma_mode : Current Field Top/Bot 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)

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;

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	0x0	reg_3dnr_nr3_suremot_txt_mode : ; // u3: 0: cur, 1:p2; 2: (cur+p2)/2; 3/up: min(cur,p2)

NR3_COOP_PARA 0x2ff1

Bit(s)	R/W	Default	Description
21:20	R/W	0x0	reg_3dnr_nr3_coop_mode : ; // u2 0 original pixel 1: [1 2 1]/4 lpf; 2: [1 2 2 2 1]/8; 3: 3x3 lpf
19:16	R/W	0x0	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;

NR3_CNOOP_GAIN 0x2ff2

Bit(s)	R/W	Default	Description
23:20	R/W	0x0	reg_3dnr_nr3_cnoop_ratio0 : ; // u4 cur and p2 color noop decision ratio0: (avg1<(MAX(sat0,sat2)*ratio0/8 + ofst0));
19:16	R/W	0x0	reg_3dnr_nr3_cnoop_ratio1 : ; // u4 cur and p2 color noop decision ratio1: (dif1<(MIN(sat0,sat2)*ratio1/8 + ofst1));
15:8	R/W	0x0	reg_3dnr_nr3_cnoop_ofset0 : ; // s8 cur and p2 color noop decision ofset0: (avg1<(MAX(sat0,sat2)*ratio0/8 + ofst0));

Bit(s)	R/W	Default	Description
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));

NR3_YMOT_PARA 0x2ff3

Bit(s)	R/W	Default	Description
19	R/W	0x0	reg_3dnr_nr3_ymot_only_en : ; // u1: enable signal for ignor chroma motion: (ytxt &coop)
18	R/W	0x0	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	0x0	reg_3dnr_nr3_ymot_only_txtthrd : ; // u8: threshold to luma texture to decide use ymot only
7:0	R/W	0x0	reg_3dnr_nr3_ymot_only_motthrd : ; // u8: threshold to luma motion to decide use ymot only

NR3_CMOT_PARA 0x2ff4

Bit(s)	R/W	Default	Description
19	R/W	0x0	reg_3dnr_nr3_cmot_only_en : ; // u1: enable signal for ignor luma motion: (ctxt &cnoop)
18	R/W	0x0	reg_3dnr_nr3_cmot_only_ymtmode : ; // 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)
15:8	R/W	0x0	reg_3dnr_nr3_cmot_only_txtthrd : ; // u8: threshold to chroma texture to decide use cmot only
7:0	R/W	0x0	reg_3dnr_nr3_cmot_only_motthrd : ; // u8: threshold to chroma motion to decide use cmot only

NR3_SUREMOT_YGAIN 0x2ff5

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_3dnr_nr3_suremot_dec_yrate : ; // u8: (norm 16)lpfMot>(dec_rate*txt +ofst) then force lpfMot*frc_gain+frc_ofset
23:16	R/W	0x0	reg_3dnr_nr3_suremot_dec_yofst : ; // u8: lpfMot>(dec_rate*txt +ofst) then force lpfMot*frc_gain+frc_ofset
15:8	R/W	0x0	reg_3dnr_nr3_suremot_frc_ygain : ; // u8: (norm 8)lpfMot>(dec_rate*txt +ofst) then force lpfMot*frc_gain+frc_ofset
7:0	R/W	0x0	reg_3dnr_nr3_suremot_frc_yofst : ; // u8: lpfMot>(dec_rate*txt +ofst) then force lpfMot*frc_gain+frc_ofset

NR3_SUREMOT_CGAIN 0x2ff6

Bit(s)	R/W	Default	Description
31:24	R/W	0x0	reg_3dnr_nr3_suremot_dec_crate : ; // u8: (norm 16)lpfMot>(dec_rate*txt +ofst) then force lpfMot*frc_gain+frc_ofset
23:16	R/W	0x0	reg_3dnr_nr3_suremot_dec_cofst : ; // u8: lpfMot>(dec_rate*txt +ofst) then force lpfMot*frc_gain+frc_ofset
15:8	R/W	0x0	reg_3dnr_nr3_suremot_frc_cgain : ; // u8: (norm 8)lpfMot>(dec_rate*txt +ofst) then force lpfMot*frc_gain+frc_ofset
7:0	R/W	0x0	reg_3dnr_nr3_suremot_frc_cofst : ; // u8: lpfMot>(dec_rate*txt +ofst) then force lpfMot*frc_gain+frc_ofset

VPP registers

VPP_DUMMY_DATA 0x1d00

Bit(s)	R/W	Default	Description
23-16	R/W	0	Y dummy Data
15-8	R/W	0	CB dummy data
7-0	R/W	0	Cr dummy data

VPP_LINE_IN_LENGTH 0x1d01

Bit(s)	R/W	Default	Description
11-0	R/W	1920	input line length used in VPP

VPP_PIC_IN_HEIGHT 0x1d02

Bit(s)	R/W	Default	Description
11-0	R/W	0xfff	input Picture height used in VPP

VPP_SCALE_COEF_IDX 0x1d03

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.

Bit(s)	R/W	Default	Description
15	R/W	0	index increment, if bit9 == 1 then (0: index increase 1, 1: index increase 2) else (index increase 2)

Bit(s)	R/W	Default	Description
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	R/W	0	vertical separated coef enable
12-10	R	0	Unused
9	R/W	0	if true, use 9bit resolution coef, other use 8bit resolution coef
8-7	R/W	0	00: vertical coef, 01: vertical chroma coef: 10: horizontal coef, 11: resevered
6-0	R/W	0	coefficient index

VPP_SCALE_COEF 0x1d04

Bit(s)	R/W	Default	Description
31-0	R/W	0	coefficients for vertical filter and horizontal filter

These following registers are the absolute line address pointer for output divided screen
The output divided screen is shown in the following:

```
-------------------------  <------ line zero
             .
             .
             .                region0      <---------- nonlinear region or nonscaling region
             .
--------------------------
--------------------------  <------ region1_startp
             .
             .        region1       <---------- nonlinear region
             .
             .
--------------------------
--------------------------  <------ region2_startp
             .
             .        region2       <---------- linear region
             .
             .
--------------------------
--------------------------  <------ region3_startp
             .
             .        region3       <---------- nonlinear region
             .
             .
--------------------------
--------------------------  <------ region4_startp
             .
             .        region4       <---------- nonlinear region or nonoscaling region
             .
             .
--------------------------  <------ region4_endp
```


VPP_VSC_REGION12_STARTP 0x1d05

Bit(s)	R/W	Default	Description
27-16	R/W	0	region1 startp
11-0	R/W	0	region2 startp

VPP_VSC_REGION34_STARTP 0x1d06

Bit(s)	R/W	Default	Description
27-16	R/W	1080	region3 startp
11-0	R/W	1080	region4 startp

VPP_VSC_REGION4_ENDP 0x1d07

Bit(s)	R/W	Default	Description
11-0	R/W	1079	region4 endp

VPP_VSC_START_PHASE_STEP 0x1d08

vertical start phase step, (source/dest)*(2^24)

Bit(s)	R/W	Default	Description
27-24	R/W	1	integer part
23-0	R/W	0	Fraction part

VPP_VSC_REGION0_PHASE_SLOPE 0x1d09

Bit(s)	R/W	Default	Description
24-0	R/W	0	vertical scaler region0 phase slope, Bit24 signed bit

VPP_VSC_REGION1_PHASE_SLOPE 0x1d0a

Bit(s)	R/W	Default	Description
24-0	R/W	0	vertical scaler region1 phase slope, Bit24 signed bit

VPP_VSC_REGION3_PHASE_SLOPE 0x1d0b

Bit(s)	R/W	Default	Description
24-0	R/W	0	vertical scaler region3 phase slope, Bit24 signed bit

VPP_VSC_REGION4_PHASE_SLOPE 0x1d0c

Bit(s)	R/W	Default	Description
24-0	R/W	0	vertical scaler region4 phase slope, Bit24 signed bit

VPP_VSC_PHASE_CTRL 0x1d0d

Bit(s)	R/W	Default	Description
18-17	R/W	0	double line mode, input/output line width of vscaler 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/W	0	0: progressive output, 1: interlace output
15	R/W	0	vertical scaler output line0 in advance or not for bottom fiel
14-13	R/W	1	vertical scaler initial repeat line0 number for bottom field
11-8	R/W	4	vertical scaler initial receiving number for bottom field
7	R/W	0	vertical scaler output line0 in advance or not for top field
6-5	R/W	1	vertical scaler initial repeat line0 number for top field
3-0	R/W	4	vertical scaler initial receiving number for top field

VPP_VSC_INI_PHASE 0x1d0e

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

VPP_HSC_REGION12_STARTP 0x1d10

Bit(s)	R/W	Default	Description
27-16	R/W	0	region1 startp
11-0	R/W	0	region2 startp

VPP_HSC_REGION34_STARTP 0x1d11

Bit(s)	R/W	Default	Description
27-16	R/W	1920	region3 startp
11-0	R/W	1920	region4 startp

VPP_HSC_REGION4_ENDP 0x1d12

Bit(s)	R/W	Default	Description
11-0	R/W	1919	region4 endp

VPP_HSC_START_PHASE_STEP 0x1d13

Bit(s)	R/W	Default	Description
27-24	R/W	1	integer part
23-0	R/W	0	fraction part

VPP_HSC_REGION0_PHASE_SLOPE 0x1d14

Bit(s)	R/W	Default	Description
24-0	R/W	0	horizontal scaler region0 phase slope, Bit24 signed bit

VPP_HSC_REGION1_PHASE_SLOPE 0x1d15

Bit(s)	R/W	Default	Description
24-0	R/W	0	horizontal scaler region1 phase slope, Bit24 signed bit

VPP_HSC_REGION3_PHASE_SLOPE 0x1d16

Bit(s)	R/W	Default	Description
24-0	R/W	0	horizontal scaler region3 phase slope, Bit24 signed bit

VPP_HSC_REGION4_PHASE_SLOPE 0x1d17

Bit(s)	R/W	Default	Description
24-0	R/W	0	horizontal scaler region4 phase slope, Bit24 signed bit

VPP_HSC_PHASE_CTRL 0x1d18

Bit(s)	R/W	Default	Description
22-21	R/W	1	horizontal scaler initial repeat pixel number0
19-16	R/W	4	horizontal scaler initial receiving number0
15-0	R/W	0	horizontal scaler top field initial phase0

VPP_SC_MISC 0x1d19

Bit(s)	R/W	Default	Description
20	R/W	0	Prehorizontal scaler enable
19	R/W	0	Prevertical scaler enable
18	R/W	0	vertical scaler enable
17	R/W	0	horizontal scaler enable
16	R/W	0	Scale_top_en
15	R/W	1	video1 scale out enable
12	R/W	0	if true, region0,region4 are nonlinear regions, otherwise they are not scaling regions, for horizontal scaler
10-8	R/W	4	horizontal scaler bank length
5			vertical scaler phase field mode, if true, disable the opposite parity line output, more bandwith needed if output 1080i
4	R/W	0	if true, region0,region4 are nonlinear regions, otherwise they are not scaling regions, for vertical scaler
2-0	R/W	4	vertical scaler bank length

VPP_PREBLEND_VD1_H_START_END 0x1d1a

Bit(s)	R/W	Default	Description
27-16	R/W	0	VD1 Preblend horizontal start
11-0	R/W	0x77f	VD1 Preblend horizontal end

VPP_PREBLEND_VD1_V_START_END 0x1d1b

Bit(s)	R/W	Default	Description
27-16	R/W	0	VD1 Preblend vertical start
11-0	R/W	0x437	VD1 Preblend vertical end

VPP_POSTBLEND_VD1_H_START_END 0x1d1c

Bit(s)	R/W	Default	Description
27-16	R/W	0	VD1 Postblend horizontal start
11-0	R/W	0x77f	VD1 Postblend horizontal end

VPP_POSTBLEND_VD1_V_START_END 0x1d1d

Bit(s)	R/W	Default	Description
27-16	R/W	0	VD1 Postblend vertical start
11-0	R/W	0x437	VD1 Postblend vertical end

VPP_BLEND_VD2_H_START_END 0x1d1e

Bit(s)	R/W	Default	Description
27-16	R/W	0	preblend/postblend video2 horizontal start
11-0	R/W	0	preblend/postblend video2 horizontal end

VPP_BLEND_VD2_V_START_END 0x1d1f

Bit(s)	R/W	Default	Description
27-16	R/W	0	preblend/postblend video2 vertical start
11-0	R/W	0	preblend/postblend video2 vertical end

VPP_PREBLEND_H_SIZE 0x1d20

Bit(s)	R/W	Default	Description
11-0	R/W	1920	preblend horizontal size

VPP_POSTBLEND_H_SIZE 0x1d21

Bit(s)	R/W	Default	Description
11-0	R/W	1920	preblend horizontal size

VPP_HOLD_LINES 0x1d22

Bit(s)	R/W	Default	Description
13-8	R/W	4	preblend hold lines
5-0	R/W	4	postblend hold lines

VPP_BLEND_ONECOLOR_CTRL 0x1d23

Bit(s)	R/W	Default	Description
25	R/W	0	if true, change screen to one color value for preblender
24	R/W	0	if true, change screen to one color value for postblender
23-16	R/W	0x10	one color Y
15-8	R/W	0x1d0	one color Cb
7-0	R/W	0x1d0	one color Cr

VPP_PREBLEND_CURRENT_XY 0x1d24

Bit(s)	R/W	Default	Description
27-16	R	0	VPP preblend current_x
11-0	R	0	VPP preblend current_y

VPP_POSTBLEND_CURRENT_XY 0x1d25

Bit(s)	R/W	Default	Description
27-16	R	0	VPP postblend current_x
11-0	R	0	VPP postblend current_y

VPP_MISC 0x1d26

Bit(s)	R/W	Default	Description
31	R/W	0	VD1, background OSD exchange enable for preblend
30	R/W	0	VD1, background OSD exchange enable for postblend
28	R/W	0	Enable for color_manage
27	R/W	0	vd2_use_viu2_out_en. If true, video from Viu2 is routed to Viu1's VD2.
26-18	R/W	0	vd2 alpha
17	R/W	0	osd2 enable for preblend
16	R/W	0	osd1 enable for preblend
15	R/W	0	vd2 enable for preblend
14	R/W	0	vd1 enable for preblend
13	R/W	0	osd2 enable for postblend
12	R/W	0	osd1 enable for postblend
11	R/W	0	vd2 enable for postblend
10	R/W	1	vd1 enable for postblend
9	R/W	0	if true, osd1 is alpha premultipiled
8	R/W	0	if true, osd2 is alpha premultipiled
7	R/W	1	postblend module enable
6	R/W	0	preblend module enable
5	R/W	0	if true, osd2 foreground compared with osd1 in preblend
4	R/W	0	if true, osd2 foreground compared with osd1 in postblend
2	R/W	0	if true, disable resetting async fifo every vsync, otherwise every vsync, the aync fifo will be reseted.
0	R/W	0	If true, the output result of VPP is saturated

VPP_OFIFO_SIZE 0x1d27

Bit(s)	R/W	Default	Description
31-20	R/W	0x77f	ofifo line length minus 1
18	R/W	0	if true invert input vs
18	R/W	0	if true invert input hs
17	R/W	0	force top/bottom field, enable
16	R/W	0	force top/bottom field, 0: top, 1: bottom
15	W	0	force one go_field, one pluse, write only
14	W	0	force one go_line, one pluse, write only
11-0	R/W	0x100	ofifo size (actually only bit 10:1 is valid), always even number

VPP_FIFO_STATUS 0x1d28

Bit(s)	R/W	Default	Description
26-17	R	0	current scale out fifo counter
16-12	R	0	current afifo counter
11-0	R	0x100	current ofifo counter

VPP_SMOKE_CTRL 0x1d29

Bit(s)	R/W	Default	Description
5	R/W	0	SMOKE3 postblend enable only when postblend vd2 is not enable
4	R/W	0	SMOKE3 preblend enable only when preblend vd2 is not enable
3	R/W	0	SMOKE2 postblend enable only when postblend osd2 is not enable
2	R/W	0	SMOKE2 preblend enable only when preblend osd2 is not enable
1	R/W	0	SMOKE1 postblend enable only when postblend osd1 is not enable
0	R/W	0	SMOKE1 preblend enable only when preblend osd1 is not enable

VPP_SMOKE1_VAL 0x1d2a

Bit(s)	R/W	Default	Description
31-24	R/W	0	Y
23-16	R/W	0	Cb
15-8	R/W	0	Cr
7-0	R/W	0	Alpha

VPP_SMOKE2_VAL 0x1d2b

Bit(s)	R/W	Default	Description
31-24	R/W	0	Y
23-16	R/W	0	Cb
15-8	R/W	0	Cr
7-0	R/W	0	Alpha

VPP_SMOKE3_VAL 0x1d2c

Bit(s)	R/W	Default	Description
31-24	R/W	0	Y
23-16	R/W	0	Cb
15-8	R/W	0	Cr
7-0	R/W	0	Alpha

VPP_SMOKE1_H_START_END 0x1d2d

Bit(s)	R/W	Default	Description
27-16	R/W	0	Start
11-0	R/W	0	end

VPP_SMOKE1_V_START_END 0x1d2e

Bit(s)	R/W	Default	Description
27-16	R/W	0	Start
11-0	R/W	0	end

VPP_SMOKE2_H_START_END 0x1d2f

Bit(s)	R/W	Default	Description
27-16	R/W	0	Start
11-0	R/W	0	end

VPP_SMOKE2_V_START_END 0x1d30

Bit(s)	R/W	Default	Description
27-16	R/W	0	Start
11-0	R/W	0	end

VPP_SMOKE3_H_START_END 0x1d31

Bit(s)	R/W	Default	Description
27-16	R/W	0	Start
11-0	R/W	0	end

VPP_SMOKE3_V_START_END 0x1d32

Bit(s)	R/W	Default	Description
27-16	R/W	0	Start
11-0	R/W	0	end

VPP_SCO_FIFO_CTRL 0x1d33

Bit(s)	R/W	Default	Description
27-16	R/W	0x77f	scale out fifo line length minus 1
9-0	R/W	0x200	scale out fifo size (actually only bit 9:1 is valid), always even number

VPP_HSC_PHASE_CTRL1 0x1d34

Bit(s)	R/W	Default	Description
27-24	R/W	0x0	Prehsc_mode, bit3:2, prehsc odd line interpolation mode, bit 1:0 prehsc even line interpolation mode Each 2bits, 00: (pix0 + pix1)/2, 01: pix1: 01: pix0
23	R/W	0x0	horizontal scaler double pixel mode
22-21	R/W	0x1	horizontal scaler initial repeat pixel0 number1
19-16	R/W	0x4	horizontal scaler initial receiving number1

Bit(s)	R/W	Default	Description
15-0	R/W	0x0	horizontal scaler top field initial phase1

VPP_HSC_INI_PAT_CTRL 0x1d35

Bit(s)	R/W	Default	Description
31-24	R/W	0	Prehsc pattern, each pattern 1bit, from lsb to msb
22-20	R/W	0	Prehsc pattern start
18-16	R/W	0	Prehsc pattern end
15-8	R/W	0x0	For 3d quincunx sub-sampling Hscaler pattern, each patten 1 bit, from lsb -> msb
6-4	R/W	0x0	Hscaler pattern start
2-0	R/W	0x0	Hscaler pattern end

VPP_VADJ_CTRL 0x1d40

Bit(s)	R/W	Default	Description
3	R/W	1	minus black level enable for vadj2
2	R/W	1	Video adjustment enable for vadj2
1	R/W	1	minus black level enable for vadj1
0	R/W	1	Video adjustment enable for vadj1

VPP_VADJ1_Y 0x1d41

Bit(s)	R/W	Default	Description
16-8	R/W	0	brightness, signed value
7-0	R/W	0x1d0	contrast, unsigned value, contrast from 0 <= contrast <2

VPP_VADJ1_MA_MB 0x1d42

cb' = cb*ma + cr*mb

cr' = cb*mc + cr*md

Bit(s)	R/W	Default	Description
16-8	R/W	0x100	MA, signed value, -2 < MA < 2
7-0	R/W	0	MB, signed value ,-2 < MB < 2

VPP_VADJ1_MC_MD 0x1d43

Bit(s)	R/W	Default	Description
16-8	R/W	0	MC, signed value, -2 < MC < 2
7-0	R/W	0x100	MD, signed value ,-2 < MD < 2

VPP_VADJ2_Y 0x1d44

Bit(s)	R/W	Default	Description
16-8	R/W	0	brightness, signed value
7-0	R/W	0x1d0	contrast, unsigned value, contrast from 0 <= contrast <2

VPP_VADJ2_MA_MB 0x1d45

cb' = cb*ma + cr*mb

cr' = cb*mc + cr*md

Bit(s)	R/W	Default	Description
16-8	R/W	0x100	MA, signed value, -2 < MA < 2
7-0	R/W	0	MB, signed value ,-2 < MB < 2

VPP_VADJ2_MC_MD 0x1d46

Bit(s)	R/W	Default	Description
16-8	R/W	0	MC, signed value, -2 < MC < 2
7-0	R/W	0x100	MD, signed value ,-2 < MD < 2

VPP_MATRIX_CTRL 0x1d5f

Bit(s)	R/W	Default	Description
9-8	R/W	0	Matrix coef index selection, 00: select post matrix, 01: select video1 matrix, 10: select video2 matrix
5	R/W	0	Video1 conversion matrix enable
4	R/W	0	Video2 conversion matrix enable
2	R/W	0	Output y/cb/cr saturation enable, only for post matrix(y saturate to 16-235, cb/cr saturate to 0-240)
1	R/W	0	input y/cb/cr saturation enable, only for post matrix(y saturate 16-235, cb/cr saturate to 16-240)
0	R/W	0	conversion matrix enable

VPP_MATRIX_COEF00_01 0x1d60

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient00, signed, 3.10
12-0	R/W	0	Coefficient01, signed, 3.10

VPP_MATRIX_COEF02_10 0x1d61

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient02, signed, 3.10
12-0	R/W	0	Coefficient10, signed, 3.10

VPP_MATRIX_COEF11_12 0x1d62

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient11, signed, 3.10
12-0	R/W	0	Coefficient12, signed, 3.10

VPP_MATRIX_COEF20_21 0x1d63

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient20, signed, 3.10
12-0	R/W	0	Coefficient21, signed, 3.10

VPP_MATRIX_COEF22 0x1d64

Bit(s)	R/W	Default	Description
18-16	R/W	0	convrs
12-0	R/W	0	Coefficient22, signed, 3.10

VPP_MATRIX_OFFSET0_1 0x1d65

Bit(s)	R/W	Default	Description
26-16	R/W	0	Offset0, signed value
10-0	R/W	0	Offset1, signed value

VPP_MATRIX_OFFSET2 0x1d66

Bit(s)	R/W	Default	Description
10-0	R/W	0	Offset2, signed value

VPP_MATRIX_PRE_OFFSET0_1 0x1d67

Bit(s)	R/W	Default	Description
27-16	R/W	0	pre_Offset0, signed value
11-0	R/W	0	Pre_Offset1, signed value

VPP_MATRIX_PRE_OFFSET2 0x1d68

Bit(s)	R/W	Default	Description
11-0	R/W	0	Pre_Offset2, signed value

VPP_DUMMY_DATA1 0x1d69

Bit(s)	R/W	Default	Description
23-16	R/W	0	Y
15-8	R/W	0	Cb
7-0	R/W	0	Cr

VPP_GAINOFF_CTRL0 0x1d6a

Bit(s)	R/W	Default	Description
31	R/W	0	Gain offset module enable
26-16	R/W	0	Gain0, 1.10 unsigned data
10-0	R/W	0	Gain1, 1.10 unsigned data

VPP_GAINOFF_CTRL1 0x1d6b

Bit(s)	R/W	Default	Description
26-16	R/W	0	Gain2, 1.10 unsigned data
10-0	R/W	0	Offset0, signed data

VPP_GAINOFF_CTRL2 0x1d6c

Bit(s)	R/W	Default	Description
26-16	R/W	0	Offset1, signed data
10-0	R/W	0	Offset2, signed data

VPP_GAINOFF_CTRL3 0x1d6d

Bit(s)	R/W	Default	Description
26-16	R/W	0	Pre_Offset0, signed data
10-0	R/W	0	Pre_Offset1, signed data

VPP_GAINOFF_CTRL4 0x1d6e

Bit(s)	R/W	Default	Description
10-0	R/W	0	Pre_Offset2, signed data

VPP_CHROMA_ADDR_PORT 0x1d70

Bit(s)	R/W	Default	Description
31-0	R/W	0	Color management address port

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):

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

Bit(s)	R/W	Default	Description
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)

SAT_BYYB_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 3th node, the same as below
15-8	R/W	0	The 2th node, the same as below
7-0	R/W	0	Signed, The 1th node about saturation gain offset along Y coordinate, the gain normalized to 128 as “1”.

SAT_BYYB_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”.

SAT_BYYB_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”.

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][:]

CM_ENH_SFT_MODE 0x204

Bit(s)	R/W	Default	Description
31-9	R/W	0	reserved
8-6	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;
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;

Bit(s)	R/W	Default	Description
			1: dnscale by 2 (-128,127)/2; 2: dnscale by 4 (-128,127)/4; 3: dnscale by 8 (-128,127)/8;

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

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

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)

CM_ENH_CTL 0x208

Bit(s)	R/W	Default	Description
31-6	R/W	0	reserved
5	R/W	0	Enable signal for CM2 Hue adjustments;
4	R/W	0	Enable signal for CM2 Saturation adjustments;
3	R/W	0	Enable signal for CM2 Luma adjustments;
2	R/W	0	Apply cm on LP portion enanble
1	R/W	0	CM2 enable signal
0	R/W	0	reserved

ROI_X_SCOPE 0x209

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	R/W	0	Ending col index of the Region of Interest (ROI)
15-12	R/W	0	reserved
11-0	R/W	0	Start col index of the Region of Interest (ROI)

ROI_Y_SCOPE 0x20a

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	R/W	0	Ending col index of the Region of Interest (ROI)
15-12	R/W	0	reserved
11-0	R/W	0	Start col index of the Region of Interest (ROI)

POI_XY_DIR0x20b

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	R/W	0	Row index of the pixel(position) of Interest (POI)
15-12	R/W	0	reserved
11-0	R/W	0	Col index of the pixel(position) of Interest (POI)

COI_Y_SCOPE 0x20c

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	R/W	0	Higher bound of luma value for color of interest (COI), 8bits precision
15-12	R/W	0	reserved
11-0	R/W	0	Lower bound of luma value for color of interest (COI), 8bits precision

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

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

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]

POI_RPL_MODE 0x210

Bit(s)	R/W	Default	Description
31-4	R/W	0	reserved
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]

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}

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

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

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

LUMA_ADJ_LIMT 0x215

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	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.

SAT_ADJ_LIMT 0x216

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	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.

HUE_ADJ_LIMT 0x217

Bit(s)	R/W	Default	Description
31-28	R/W	0	reserved
27-16	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

Bit(s)	R/W	Default	Description
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.

UVHS_OFST 0x218

Bit(s)	R/W	Default	Description
31-24	R/W	0	V offset after CM2, under s10 scale
23-16	R/W	0	U offset after CM2, under s10 scale
15-8	R/W	0	V offset before CM2, under s10 scale
7-0	R/W	0	U offset before CM2, under s10 scale.

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; 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 adjusments

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

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

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

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

XVYCC_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

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-offset aligned, we allocate 8 addr-space to each node parameter, for example, the parameter of 1th node uses the address space : 0x100, 0x101, 0x102, 0x103, 0x104, and 2th node uses the address space : 0x108, 0x109, 0x10a, 0x10b, 0x10c,and 2th node uses the address space : 0x110, 0x111, 0x112, 0x113, 0x114… … … …

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)

CM2_ENH_COEF1_H00 0x101

Bit(s)	R/W	Default	Description
31-24	R/W		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

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

CM2_ENH_COEF3_H00 0x103

Bit(s)	R/W	Default	Description
31-24	R/W	0	Saturation gain offsetfor 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

CM2_ENH_COEF4_H00 0x104

Bit(s)	R/W	Default	Description
31-24	R/W	0	Saturation gain offsetfor 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 offsetfor 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 offsetfor 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 offsetfor 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 0x1fc

VPP_GCLK_CTRL0 0x1d72

Bit(s)	R/W	Default	Description
31-30	R/W	0	Gate clock control for chroma Coring, for each 2bits, if bit 2*i+1 == 1, free clk, else if bit 2*i == 1 no clk, else auto gated clock
29-28	R/W	0	Gate clock control for Black extension
27-26	R/W	0	Gate clock control for DNLP
31-30	R/W	0	Gate clock control for hsvsharper
29-28	R/W	0	Gate clock control for blue stretch
27-26	R/W	0	Gate clock control for gainoff
25-24	R/W	0	Gate clock control for matrix_vd1

Bit(s)	R/W	Default	Description
23-22	R/W	0	Gate clock control for matrix_vd2
21-20	R/W	0	Gate clock control for matrix_postt
19-18	R/W	0	Gate clock control for prebld
17-16	R/W	0	Gate clock control for postbld
15-14	R/W	0	Gate clock control for hsharper
13-12	R/W	0	Gate clock control for scaler output FIFO
11-10	R/W	0	Gata clock control for vadj1
9-8	R/W	0	Gata clock control for vadj2
7-6	R/W	0	Gata clock control for output FIFO
5-4	R/W	0	Gate clock control for color management
3-2	R/W	0	Gate clock control for vpp common non auto gate clock(clk0)
1-0	R/W	0	Gate clock control for vpp register, bit0 is always 0, that means vpp_reg clk can not be turned off

VPP_GCLK_CTRL1 0x1d73

Bit(s)	R/W	Default	Description
9-8	R/W	0	Gate clock control for color management2 filter
7-6	R/W	0	Gate clock control for color management2
5-4	R/W	0	Gate clock control for chroma Coring
3-2	R/W	0	Gate clock control for Black extension
1-0	R/W	0	Gate clock control for DNLP

VPP_SC_GCLK_CTRL 0x1d74

Bit(s)	R/W	Default	Description
11-10	R/W	0	Gate clock control for pre horizontal scaler
9-8	R/W	0	Gate clock control for line buffer
7-6	R/W	0	Gate clock control for pre vertical scaler
5-4	R/W	0	Gate clock control for vertical scaler
3-2	R/W	0	Gate clock control for horizontal scaler
1-0	R/W	0	Gate clock control for the scaler common non auto gate clock(clk0)

VPP_MISC1 0X1d76

Bit(s)	R/W	Default	Description
17-9	R/W	0	VD1 alpha for preblend
8-0	R/W	0	VD1 alpha for postblend

VPP_XVYCC_GCLK_CTRL 0X1d79

Bit(s)	R/W	Default	Description
17-16	R/W	0	Gate clock control for xvycc inv-lut
15-14	R/W	0	Gate clock control for xvycc inv-lut Y
13-12	R/W	0	Gate clock control for xvycc inv-lut U
11-10	R/W	0	Gate clock control for xvycc int-lut V
9-8	R/W	0	Gate clock control for xvycc lut
7-6	R/W	0	Gate clock control for xvycc lut R
5-4	R/W	0	Gate clock control for xvycc lut G
3-2	R/W	0	Gate clock control for xvycc lut B
1-0	R/W	0	Gate clock control for xvycc register configure

VPP_BLACKEXT_CTRL 0X1d80

Bit(s)	R/W	Default	Description
31-24	R/W	0	blackext_start
23-16	R/W	0	blackext_slope1
15-8	R/W	0	blackext_midpt
7-0	R/W	0	blackext_slope2

VPP_DNLP_CTRL_00 0X1d81

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region03 output value
23-16	R/W	0	bottom of region02 output value
15-8	R/W	0	bottom of region01 output value
7-0	R/W	0	bottom of region00 output value

VPP_DNLP_CTRL_01 0X1d82

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region07 output value
23-16	R/W	0	bottom of region06 output value
15-8	R/W	0	bottom of region05 output value
7-0	R/W	0	bottom of region04 output value

VPP_DNLP_CTRL_02 0X1d83

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region11 output value
23-16	R/W	0	bottom of region10 output value
15-8	R/W	0	bottom of region09 output value
7-0	R/W	0	bottom of region08 output value

VPP_DNLP_CTRL_03 0X1d84

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region15 output value
23-16	R/W	0	bottom of region14 output value
15-8	R/W	0	bottom of region13 output value
7-0	R/W	0	bottom of region12 output value

VPP_DNLP_CTRL_04 0X1d85

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region19 output value
23-16	R/W	0	bottom of region18 output value
15-8	R/W	0	bottom of region17 output value
7-0	R/W	0	bottom of region16 output value

VPP_DNLP_CTRL_05 0X1d86

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region23 output value
23-16	R/W	0	bottom of region22 output value
15-8	R/W	0	bottom of region21 output value
7-0	R/W	0	bottom of region20 output value

VPP_DNLP_CTRL_06 0X1d87

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region27 output value
23-16	R/W	0	bottom of region26 output value
15-8	R/W	0	bottom of region25 output value
7-0	R/W	0	bottom of region24 output value

VPP_DNLP_CTRL_07 0X1d88

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region31 output value
23-16	R/W	0	bottom of region30 output value
15-8	R/W	0	bottom of region29 output value
7-0	R/W	0	bottom of region28 output value

VPP_DNLP_CTRL_08 0X1d89

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region35 output value
23-16	R/W	0	bottom of region34 output value
15-8	R/W	0	bottom of region33 output value
7-0	R/W	0	bottom of region32 output value

VPP_DNLP_CTRL_09 0X1d8a

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region39 output value
23-16	R/W	0	bottom of region38 output value
15-8	R/W	0	bottom of region37 output value
7-0	R/W	0	bottom of region36 output value

VPP_DNLP_CTRL_10 0X1d8b

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region43 output value
23-16	R/W	0	bottom of region42 output value
15-8	R/W	0	bottom of region41 output value
7-0	R/W	0	bottom of region40 output value

VPP_DNLP_CTRL_11 0X1d8c

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region47 output value
23-16	R/W	0	bottom of region46 output value
15-8	R/W	0	bottom of region45 output value
7-0	R/W	0	bottom of region44 output value

VPP_DNLP_CTRL_12 0X1d8d

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region51 output value
23-16	R/W	0	bottom of region50 output value
15-8	R/W	0	bottom of region49 output value
7-0	R/W	0	bottom of region48 output value

VPP_DNLP_CTRL_13 0X1d8e

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region55 output value
23-16	R/W	0	bottom of region54 output value
15-8	R/W	0	bottom of region53 output value
7-0	R/W	0	bottom of region52 output value

VPP_DNLP_CTRL_14 0X1d8f

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region59 output value
23-16	R/W	0	bottom of region58 output value
15-8	R/W	0	bottom of region57 output value
7-0	R/W	0	bottom of region56 output value

VPP_DNLP_CTRL_15 0X1d90

Bit(s)	R/W	Default	Description
31-24	R/W	0	bottom of region63 output value
23-16	R/W	0	bottom of region62 output value
15-8	R/W	0	bottom of region61 output value
7-0	R/W	0	bottom of region60 output value

VPP_BLUE_STRETCH_1 0X1d9c

Bit(s)	R/W	Default	Description
29	R/W	0	blue_stretch_cb_inc
28	R/W	0	blue_stretch_cr_inc
27	R/W	0	the MSB of blue_stretch_error_crp_inv[11:0]
26	R/W	0	the MSB of blue_stretch_error_crn_inv[11:0]
25	R/W	0	the MSB of blue_stretch_error_cbp_inv[11:0]
24	R/W	0	the MSB of blue_stretch_error_cbn_inv[11:0]
23-16	R/W	0	blue_stretch_gain
15-8	R/W	0	blue_stretch_gain_cb4cr
7-0	R/W	0	blue_stretch_luma_high

VPP_BLUE_STRETCH_2 0X1d9d

Bit(s)	R/W	Default	Description
31-27	R/W	0	blue_stretch_error_crp
26-16	R/W	0	the 11 LSB of blue_stretch_error_crp_inv[11:0]
15-11	R/W	0	blue_stretch_error_crn
10-0	R/W	0	the 11 LSB of blue_stretch_error_crn_inv[11:0]

VPP_BLUE_STRETCH_3 0X1d9e

Bit(s)	R/W	Default	Description
31-27	R/W	0	blue_stretch_error_cbp
26-16	R/W	0	the 11 LSB of blue_stretch_error_cbp_inv[11:0]
15-11	R/W	0	blue_stretch_error_cbn
10-0	R/W	0	the 11 LSB of blue_stretch_error_cbn_inv[11:0]

VPP_CCORING_CTRL 0X1da0

Bit(s)	R/W	Default	Description
25-16	R/W	0	Bypass chroma coring if Y less than this threshold
15-8	R/W	0	Chroma coring threshold
3-0	R/W	0	Chroma coring slope

VPP_VE_ENABLE_CTRL 0X1da1

Bit(s)	R/W	Default	Description
20	R/W	0	demo chroma coring enable
19	R/W	0	demo black enxtension enable
18	R/W	0	demo dynamic nonlinear luma processing enable
17	R/W	0	demo hsvsharp enable
16	R/W	0	demo bluestretch enable
15-14	R/W	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
4	R/W	0	chroma coring enable
3	R/W	0	black enxtension enable
2	R/W	0	dynamic nonlinear luma processing enable
1	R/W	0	hsvsharp enable
0	R/W	0	bluestretch enable

VPP_VE_DEMO_LEFT_TOP_SCREEN_WIDTH 0X1da2

Bit(s)	R/W	Default	Description
11-0	R/W	0	demo left or top screen width

VPP_VE_DEMO_CENTER_BAR 0X1da3

Bit(s)	R/W	Default	Description
31	R/W	0	Center bar enable
27-24	R/W	0	Center bar width (*2)
23-16	R/W	0	Center bar cr (*4)
15-8	R/W	0	Center bar cb (*4)
7-0	R/W	0	Center bar y (*4)

VPP_VE_H_V_SIZE 0X1da4

Bit(s)	R/W	Default	Description
27-16	R/W	0x780	Line_length for VE
11-0	R/W	0x438	Picture height for VE

VPP_PSR_H_V_SIZE 0X1da5

Bit(s)	R/W	Default	Description
27-16	R/W	0x780	Line_length for post superscalar, such as blue stretch
11-0	R/W	0x438	Picture height for post superscalar, such as blue stretch

VPP_IN_H_V_SIZE 0X1da6

Bit(s)	R/W	Default	Description
27-16	R/W	0x780	Line_length for pre superscalar, such as front cti
11-0	R/W	0x438	Picture height for pre superscalar, such as front cti

VPP_VDO_MEAS_CTRL 0x1da8

Bit(s)	R/W	Default	Description
10	R/W	0	Reset bit, high active
9	R/W	0	0: measuring rising edge, 1: measuring falling edge
8	R/W	0	if true, accumulate the counter number, otherwise not
7-0	R/W	0	vsync_span, define how many vsync span need to measure

VPP_VDO_MEAS_VS_COUNT_HI 0x1da9

Bit(s)	R/W	Default	Description
19-16	R	0	ind_meas_count_n, every number of sync_span vsyncs, this counter add 1
15-0	R/W	0	high bit portion of counter

VPP_VDO_MEAS_VS_COUNT_L0 0x1daa

Bit(s)	R/W	Default	Description
31-0	R	0	low bit portion of counter

VPP_INPUT_CTRL 0x1dab

Bit(s)	R/W	Default	Description
11-9	R/W	0x0	vd2_sel, 001: select vd1_din, 010: select vd2_din, 011: select d2d3_l_din, 100: d2d3_r_din, otherwise no selection
8-6	R/W	0x2	vd1_l_sel, 001: select vd1_din, 010: select vd2_din, 011: select d2d3_l_din, 100: d2d3_r_din, otherwise no selection

5-3	R/W	0x1	vd1_r_sel, 001: select vd1_din, 010: select vd2_din, 011: select d2d3_l_din, 100: d2d3_r_din, otherwise no selection
			
			//note: the source vd1_l_sel selected cannot be used as the source of vd1_r_sel or vd2_sel
			// vd1_r_sel is useful only vd1_interleave_mode is not 00. And the source vd1_r_sel used can not used for the vd2_sel any more.

2-0	R/W	0x0	vd1_interleave_mode, 000: no interleave, 001: pixel interleaving, 010: line interleaving, 011: 2 pixel interleaving, // 100: 2 line interleaving

VPP_FRONT_HLTI_CTRL 0x1dbb

Bit(s)	R/W	Default	Description
31-24	R/W	0x0	front_hlti_neg_gain
23-16	R/W	0x0	front_hlti_pos_gain
15-8	R/W	0x0	front_hlti_threshold
7-0	R/W	0x0	front_hlti_blend_factor

VPP_FRONT_CTI_CTRL 0x1dbc

Bit(s)	R/W	Default	Description
31	R/W	0x0	front_enable, enable the front LTI&CTI before scaler

Bit(s)	R/W	Default	Description
26-24	R/W	0x0	front_cti_step2
23-21	R/W	0x0	front_cti_step
20-16	R/W	0x0	front_cti_blend_factor
15	R/W	0x0	front_cti_median_mode
14-8	R/W	0x0	front_cti_threshold
7-0	R/W	0x0	front_cti_gain

VPP_FRONT_CTI_CTRL2 0x1dbd

Bit(s)	R/W	Default	Description
29-28	R/W	0x0	front_hlti_step
25-24	R/W	0x0	front_cti_bpf_sel
20-16	R/W	0x0	front_cti_blend_factor_gama
12-8	R/W	0x0	front_cti_blend_factor_beta
4-0	R/W	0x0	front_cti_blend_factor_alpha

VPP OSD SCALER

VPP_OSD_VSC_PHASE_STEP 0x1dc0

Bit(s)	R/W	Default	Description
27-0	R/W	0x01000000	4.24 format

VPP_OSD_VSC_INI_PHASE 0x1dc1

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	botttom vertical scaler initial phase
15-0	R/W	0x0	top vertical scaler initial phase

VPP_OSD_VSC_CTRL0 0x1dc2

Bit(s)	R/W	Default	Description
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

VPP_OSD_HSC_PHASE_STEP 0x1dc3

Bit(s)	R/W	Default	Description
27-0	R/W	0x01000000	4.24 format

VPP_OSD_HSC_INI_PHASE 0x1dc4

Bit(s)	R/W	Default	Description
31-16	R/W	0x0	horizontal scaler initial phase1
15-0	R/W	0x0	horizontal scaler initial phase0

VPP_OSD_HSC_CTRL0 0x1dc5

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

Bit(s)	R/W	Default	Description
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

VPP_OSD_HSC_INI_PAT_CTRL 0x1dc6

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

VPP_OSD_SC_DUMMY_DATA 0x1dc7

Bit(s)	R/W	Default	Description
31-24	R/W	0x0	componet 0
23-16	R/W	0x0	component 1
15-8	R/W	0x0	component 2
7-0	R/W	0x0	component 3, alpha

VPP_OSD_SC_CTRL0 0x1dc8

Bit(s)	R/W	Default	Description
14	R/W	0x0	osc_sc_din_osd1_alpha_mode, 1: (alpha >= 128) ? alpha -1: alpha, 0: (alpha >=1) ? alpha - 1: alpha.
13	R/W	0x0	osc_sc_din_osd2_alpha_mode, 1: (alpha >= 128) ? alpha -1: alpha, 0: (alpha >=1) ? alpha - 1: alpha.
12	R/W	0x0	osc_sc_dout_alpha_mode, 1: (alpha >= 128) ? alpha + 1: alpha, 0: (alpha >=1) ? alpha + 1: alpha.
11-4	R/W	0x0	default alpha for vd1 or vd2 if they are selected as the source
3	R/W	0x0	osd scaler path enable
1-0	R/W	0x0	osd_sc_sel, 00: select osd1 input, 01: select osd2 input, 10: select vd1 input, 11: select vd2 input after matrix

VPP_OSD_SCI_WH_M1 0x1dc9

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

VPP_OSD_SCO_H_START_END 0x1dca

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

VPP_OSD_SCO_V_START_END 0x1dcb

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

VPP_OSD_SCALE_COEF_IDX 0x1dcc

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

Bit(s)	R/W	Default	Description
6-0	R/W	0x0	coef index

VPP_OSD_SCALE_COEF 0x1dcd

Bit(s)	R/W	Default	Description
31-0	R/W	0x0	

VPP_INT_LINE_NUM 0x1dce

Bit(s)	R/W	Default	Description
12-0	R/W	0x0	line number use to generate interrupt when line == this number

VPP_XVYCC_MISC 0x1dcf

Bit(s)	R/W	Default	Description
31-21			Reserved
20-18	R/W	0x0	Xvycc_lut_enable_sel 1: enable synced by vsync 0: not synced by vsync
17	R/W	0x0	Gainoff_en_sel 1: enable synced by vsync 0: not synced by vsync
16	R/W	0x0	Mat_highlight_en_sel 1: enable synced by vsync 0: not synced by vsync
14	R/W	0x0	Mat_post_conv_en_sel 1: enable synced by vsync 0: not synced by vsync
13	R/W	0x0	Vadj2_minus_black_en_sel 1: enable synced by vsync 0: not synced by vsync
12	R/W	0x0	Vadj2_en_sel 1: enable synced by vsync 0: not synced by vsync
11	R/W	0x0	Mat_vd1_highlight_en_sel 1: enable synced by vsync 0: not synced by vsync
9	R/W	0x0	Mat_vd1_conv_en_sel 1: enable synced by vsync 0: not synced by vsync
8	R/W	0x0	Vadj1_minus_black_en_sel 1: enable synced by vsync 0: not synced by vsync
7	R/W	0x0	Vadj1_en_sel 1: enable synced by vsync 0: not synced by vsync
6-4	R/W	0x0	Xvycc_cmpr_invlut_enable_sel 1: enable synced by vsync 0: not synced by vsync
2	R/W	0x0	Demo_center_bar_en_sel 1: enable synced by vsync 0: not synced by vsync
1	R/W	0x0	Demo_bluestretch_enable_sel 1: enable synced by vsync 0: not synced by vsync
0	R/W	0x0	Blue_stretch_en_sel 1: enable synced by vsync 0: not synced by vsync

VPP_OFIFO_URG_CTRL 0x1dd8

Bit(s)	R/W	Default	Description
15	R/W	0	urgent_ctrl_en
14	R/W	0	urgent_wr if true for write buffer
13	R/W	0	out_inv_en
12	R/W	0	urg_ini_value
11-6	R/W	0	up_th up threshold
5-0	R/W	0	dn_th dn threshold

VPP_CLIP_MISC0 0x1dd9

Bit(s)	R/W	Default	Description
29-20	R/W	0x3ff	Final clip of vpp display r channel top
19-10	R/W	0x3ff	Final clip of vpp display g channel top
9-0	R/W	0x3ff	Final clip of vpp display b channel top

VPP_CLIP_MISC1 0x1dda

Bit(s)	R/W	Default	Description
29-20	R/W	0x0	Final clip of vpp display r channel bottom
19-10	R/W	0x0	Final clip of vpp display g channel bottom
9-0	R/W	0x0	Final clip of vpp display b channel bottom

VPP_MATRIX_COEF13_14 0x1ddb

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient13, signed, 3.10
12-0	R/W	0	Coefficient14, signed, 3.10

VPP_MATRIX_COEF23_24 0x1ddc

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient23, signed, 3.10
12-0	R/W	0	Coefficient24, signed, 3.10

VPP_MATRIX_COEF15_25 0x1ddd

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient15, signed, 3.10
12-0	R/W	0	Coefficient25, signed, 3.10

VPP_MATRIX_CLIP 0x1dde

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
2-0	R/W	0x10	Matrix clip enable

VPP_XVYCC_MISC0 0x1ddf

Bit(s)	R/W	Default	Description
29-20	R/W	0x3ff	Final clip of vpp display r channel top
19-10	R/W	0x3ff	Final clip of vpp display g channel top
9-0	R/W	0x3ff	Final clip of vpp display b channel top

VPP_XVYCC_MISC1 0x1de0

Bit(s)	R/W	Default	Description
29-20	R/W	0x0	Final clip of vpp display r channel bottom
19-10	R/W	0x0	Final clip of vpp display g channel bottom
9-0	R/W	0x0	Final clip of vpp display b channel bottom

VPP_VD1_CLIP_MISC0 0x1de1

Bit(s)	R/W	Default	Description
31-30	R/W	0x0	reserved
29-20	R/W	0x3ff	Clip top of vd1 input channel R
19-10	R/W	0x3ff	Clip top of vd1 input channel G
9-0	R/W	0x3ff	Clip top of vd1 input channel B

VPP_VD1_CLIP_MISC1 0x1de2

31-30	R/W	0x0	reserved
29-20	R/W	0x0	Clip bottom of vd1 input channel R
19-10	R/W	0x0	Clip bottom of vd1 input channel G
9-0	R/W	0x0	Clip bottom of vd1 input channel B

VPP_VD2_CLIP_MISC0 0x1de3

Bit(s)	R/W	Default	Description
31-30	R/W	0x0	reserved
29-20	R/W	0x3ff	Clip top of vd2 input channel R
19-10	R/W	0x3ff	Clip top of vd2 input channel G
9-0	R/W	0x3ff	Clip top of vd2 input channel B

VPP_VD2_CLIP_MISC1 0x1de4

31-30	R/W	0x0	reserved
29-20	R/W	0x0	Clip bottom of vd2 input channel R
19-10	R/W	0x0	Clip bottom of vd2 input channel G
9-0	R/W	0x0	Clip bottom of vd2 input channel B

Super scaler/sharpness/xvycc

XVYCC_INV_LUT_Y_ADDR_PORT 0x3158

Bit(s)	R/W	Default	Description
31:7			Reserved
6:0	R/W	0	xvycc inv lut read/write address for Y

XVYCC_INV_LUT_Y_DATA_PORT 0x3159

Bit(s)	R/W	Default	Description
31:12			Reserved
11:0	R/W	0	xvycc inv lut read/write data for Y

XVYCC_INV_LUT_U_ADDR_PORT 0x315a

Bit(s)	R/W	Default	Description
31:6			Reserved
5:0	R/W	0	xvycc inv lut read/write address for U

XVYCC_INV_LUT_U_DATA_PORT 0x315b

Bit(s)	R/W	Default	Description
31:12			Reserved
11:0	R/W	0	xvycc inv lut read/write data for U

XVYCC_INV_LUT_V_ADDR_PORT 0x315c

Bit(s)	R/W	Default	Description
31:6			Reserved
5:0	R/W	0	xvycc inv lut read/write address for V

XVYCC_INV_LUT_V_DATA_PORT 0x315d

Bit(s)	R/W	Default	Description
31:12			Reserved
11:0	R/W	0	xvycc inv lut read/write data for V

XVYCC_LUT_R_ADDR_PORT 0x315e

Bit(s)	R/W	Default	Description
31:7			Reserved
6:0	R/W	0	xvycc lut read/write address for R

XVYCC_LUT_R_DATA_PORT 0x315f

Bit(s)	R/W	Default	Description
31:10			Reserved
9:0	R/W	0	xvycc lut read/write data for R

XVYCC_LUT_G_ADDR_PORT 0x3160

Bit(s)	R/W	Default	Description
31:7			Reserved
6:0	R/W	0	xvycc lut read/write address for G

XVYCC_LUT_G_DATA_PORT 0x3161

Bit(s)	R/W	Default	Description
31:10			Reserved
9:0	R/W	0	xvycc lut read/write data for G

XVYCC_LUT_B_ADDR_PORT 0x3162

Bit(s)	R/W	Default	Description
31:7			Reserved

6:0	R/W	0	xvycc lut read/write address for B

XVYCC_LUT_B_DATA_PORT 0x3163

Bit(s)	R/W	Default	Description
31:10			Reserved
9:0	R/W	0	xvycc lut read/write data for B

XVYCC_INV_LUT_CTL 0x3164

Bit(s)	R/W	Default	Description
31:15			Reserved
14:12	R/W	0	enable for xvycc compression inverse-lut [2] for Y, [1] for U, [0] for V
11:10	R/W	0	v LUT input scale for positive portion, u2
9:8	R/W	0	v LUT input scale for negative portion, u2
7:6	R/W	0	u LUT input scale for positive portion, u2
5:4	R/W	0	u LUT input scale for negative portion, u2
3:2	R/W	0	y LUT input scale for positive portion, u2
1:0	R/W	0	y LUT input scale for negative portion, u2

XVYCC_LUT_CTL 0x3165

Bit(s)	R/W	Default	Description
31:7			Reserved
6:4	R/W	0	enable for xvycc lut [6] for Y, [5] for U, [4] for V
3:2	R/W	0	LUT input scale for positive portion, u2
1:0	R/W	0	LUT input scale for negative portion, u2

XVYCC_VADJ1_CURV_0 0x3166

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

XVYCC_VADJ1_CURV_1 0x3167

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

XVYCC_VADJ1_CURV_2 0x3168

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

XVYCC_VADJ1_CURV_3 0x3169

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

XVYCC_VADJ2_CURV_0 0x316a

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

XVYCC_VADJ2_CURV_1 0x316b

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

XVYCC_VADJ2_CURV_2 0x316c

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

XVYCC_VADJ2_CURV_3 0x316d

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

XVYCC_VD1_RGB_CTRST 0x3170

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'
15:12			Reserved
11:2	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

XVYCC_VD1_RGB_BRGHT 0x3171

Bit(s)	R/W	Default	Description
31:14			Reserved
13:2	R/W	0	Vd1_rgb_brght: brightness level in rgb domain
1	R/W	1	Vd1_rgb_brght_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

XVYCC_VD1_RGB_DLUT_0_3 0x3172

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

XVYCC_VD1_RGB_DLUT_4_7 0x3173

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

XVYCC_VD1_RGB_DLUT_8_11 0x3174

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

XVYCC_POST_RGB_CTRST 0x3175

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_blklvl: 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

XVYCC_POST_RGB_BRGHT 0x3176

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

XVYCC_POST_RGB_DLUT_0_3 0x3177

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
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

XVYCC_POST_RGB_DLUT_4_7 0x3178

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

XVYCC_POST_RGB_DLUT_8_11 0x3179

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

VIU_EOTF_CTL 0x31d0

Bit(s)	R/W	Default	Description
31:27	R/W	0	[31] for all eotf enable, [30] for matrix3x3 enable, [29:27] for eotf_ch0~3
17-6	R/W	0	For clock gate control
5-4	R/W	0	Pscale_mode for ch2
3-2	R/W	0	Pscale_mode for ch1
1-0	R/W	0	Pscale_mode for ch0

VIU_EOTF_COEF00_01 0x31d1

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef00
12-0	R/W	0	Coef01

VIU_EOTF_COEF02_10 0x31d2

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef02
12-0	R/W	0	Coef10

VIU_EOTF_COEF11_12 0x31d3

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef11
12-0	R/W	0	Coef12

VIU_EOTF_COEF20_21 0x31d4

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef20
12-0	R/W	0	Coef21

VIU_EOTF_COEF22_RS 0x31d5

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef22
12-0	R/W	0	Coef_rs

VIU_EOTF_LUT_ADDR_PORT 0x31d6

Bit(s)	R/W	Default	Description
31-0	R/W	0	The lut is addr(33*3) * 16bits. Eatch addr has 2 16bits. So biggest valid address is 33*3/2-1 =49

VIU_EOTF_LUT_DATA_PORT 0x31d7

Bit(s)	R/W	Default	Description
31-16	R/W	0	For No.(address*2+1) 16bits. 0~32 16bits is for r , 33~65 for g, 66~98 for b
15-0	R/W	0	For No(address*2) 16bits.

VD2_AFBC_ENABLE 0x3180

Bit(s)	R/W	Default	Description
8	R/W	0	dec_enable : unsigned , default = 0
0	R/W	0	frm_start : unsigned , default = 0

VD2_AFBC_MODE 0x3181

Bit(s)	R/W	Default	Description
31	R/W	0x0	soft_reset : the use as go_field
28	R/W	0x0	Blk_mem_mode : Default = 0, body space save mode when blk_mem_mode ==1
27:26	R/W	0	rev_mode : uns, default = 0 , reverse mode
25:24	R/W	3	mif_urgent : uns, default = 3 , info mif and data mif urgent
22:16	R/W	0x0	hold_line_num :
15:14	R/W	1	burst_len : uns, default = 1, 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 bit 3:2,: u component bitwidth : 00-8bit 01-9bit 10-10bit bit 5:4,: v component bitwidth : 00-8bit 01-9bit 10-10bit
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

VD2_AFBC_SIZE_IN 0x3182

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

VD2_AFBC_DEC_DEF_COLOR 0x3183

Bit(s)	R/W	Default	Description
29:20	R/W	0	def_color_y : uns, default = 0, afbc dec y default setting value
19:10	R/W	0	def_color_u : uns, default = 0, afbc dec u default setting value
9: 0	R/W	0	def_color_v : uns, default = 0, afbc dec v default setting value

VD2_AFBC_CONV_CTRL 0x3184

Bit(s)	R/W	Default	Description
11: 0	R/W	256	conv_lbuf_len : uns, default = 256, unit=16 pixel need to set = 2^n

VD2_AFBC_LBUF_DEPTH 0x3185

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;

VD2_AFBC_HEAD_BADDR 0x3186

Bit(s)	R/W	Default	Description
31:0	R/W	0x0	mif_info_baddr : uns, default = 0x0;

VD2_AFBC_BODY_BADDR 0x3187

Bit(s)	R/W	Default	Description
31:0	R/W	0x0001_0000	mif_data_baddr : uns, default = 0x0001_0000;

VD2_AFBC_OUT_XSCOPE 0x3188

Bit(s)	R/W	Default	Description
28:16	R/W	0	out_horz_bgn : uns, default = 0 ; // unit: 1 pixel
12:0	R/W	1919	out_horz_end : uns, default = 1919 ; // unit: 1 pixel

VD2_AFBC_OUT_YSCOPE 0x3189

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

VD2_AFBC_STAT 0x318A

Bit(s)	R/W	Default	Description
0	RO	0x0	frm_end_stat : uns, frame end status

VD2_AFBC_VD_CFMT_CTRL 0x318b

Bit(s)	R/W	Default	Description
31	R/W	0x0	it : true, disable clock, otherwise enable clock
30	R/W	0x0	soft : rst bit
28	R/W	0x0	if : true, horizontal formatter use repeating to generete pixel, otherwise use bilinear interpolation
27:24	R/W	0x0	horizontal : formatter initial phase
23	R/W	0x0	horizontal : formatter repeat pixel 0 enable
22:21	R/W	0x0	horizontal : Y/C ratio, 00: 1:1, 01: 2:1, 10: 4:1
20	R/W	0x0	horizontal : formatter enable
19	R/W	0x0	if : true, always use phase0 while vertical formater, meaning always repeat data, no interpolation
18	R/W	0x0	if : true, disable vertical formatter chroma repeat last line
17	R/W	0x0	veritcal : formatter dont need repeat line on phase0, 1: enable, 0: disable
16	R/W	0x0	veritcal : formatter repeat line 0 enable
15:12	R/W	0x0	vertical : formatter skip line num at the beginning
11:8	R/W	0x0	vertical : formatter initial phase
7:1	R/W	0x0	vertical : formatter phase step (3.4)

0	R/W	0x0	vertical : formatter enable

VD2_AFBC_VD_CFMT_W 0x318c

Bit(s)	R/W	Default	Description
27:16	R/W	0x0	horizontal : formatter width
11:0	R/W	0x0	vertical : formatter width

VD2_AFBC_MIF_HOR_SCOPE 0x318d

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

VD2_AFBC_MIF_VER_SCOPE 0x318e

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

VD2_AFBC_PIXEL_HOR_SCOPE 0x318f

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

VD2_AFBC_PIXEL_VER_SCOPE 0x3190

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

VD2_AFBC_VD_CFMT_H 0x3191

Bit(s)	R/W	Default	Description
12:0	R/W	0x0	vertical : formatter height

OSD1_AFBCD_ENABLE 0x31a0

Bit(s)	R/W	Default	Description
15:9	R/W	64	id_fifo_thrd : unsigned , default = 64, axi id fifo threshold
8	R/W	0	dec_enable : unsigned , default = 0
0	R/W	0	frm_start : unsigned , default = 0

OSD1_AFBCD_MODE 0x31a1

Bit(s)	R/W	Default	Description
31	R/W	0x0	soft_reset : the use as go_field
28	R/W	0	axi_reorder_mode : default=0, the axi reorder mode, note : don't seting
25:24	R/W	3	mif_urgent : uns, default = 3 , info mif and data mif urgent
22:16	R/W	0x0	hold_line_num :
15:8	R/W	0x0	rgba_exchan_ctrl :
6	R/W	1	hreg_block_split : uns, default = 1 , Enable/disable block split mode in sparse allocation
5	R/W	1	hreg_half_block : uns, default = 1 , Enable/disable half block decoding. 1=half block, 0=full block
4:0	R/W	5	hreg_pixel_packing_fmt : uns, default = 5 , Pixel format

OSD1_AFBCD_SIZE_IN 0x31a2

Bit(s)	R/W	Default	Description
31:16	R/W	1920	hreg_hsize_in : uns, default = 1920 , pic horz size in unit: pixel
15:0	R/W	1080	hreg_vsize_in : uns, default = 1080 , pic vert size in unit: pixel

OSD1_AFBCD_HDR_PTR 0x31a3

Bit(s)	R/W	Default	Description
31:0	R/W	0	hreg_hdr_ptr : uns, default = 0 ,

OSD1_AFBCD_FRAME_PTR 0x31a4

Bit(s)	R/W	Default	Description
31:0	R/W	0	hreg_frame_ptr : uns, default = 0 , The start address of the target frame buffer. For YUV format, this pointer specifies the luma buffer.

OSD1_AFBCD_CHROMA_PTR 0x31a5

Bit(s)	R/W	Default	Description
31:0	R.O	0	hreg_chroma_ptr : uns, default = 0 , Only valid in YUV format, to specify the target chroma buffer.

OSD1_AFBCD_CONV_CTRL 0x31a6

Bit(s)	R/W	Default	Description
15: 0	R/W	1024	conv_lbuf_len : uns, default = 1024, unit=16 pixel need to set = 2^n

OSD1_AFBCD_STATUS 0x31a8

Bit(s)	R/W	Default	Description
3	R/W	0	hreg_dec_resp : uns, default = 0 , Decoder error flage from the dec4x4 core
2	R/W	0	hreg_axi_bresp : uns, default = 0 , Bus error flag for AXI write error
1	R/W	0	hreg_axi_rresp : uns, default = 0 , Bus error flag for AXI read error
0	R/W	0	hreg_idle_n : uns, default = 0 , Idle output, value 0 indicates the standalone decoder is free now and can start the next frame.

OSD1_AFBCD_PIXEL_HSCOPE 0x31a9

Bit(s)	R/W	Default	Description
31:16	R/W	0	dec_pixel_bgn_h : uns, default = 0 ; // unit: pixel
15: 0	R/W	1919	dec_pixel_end_h : uns, default = 1919 ; // unit: pixel

OSD1_AFBCD_PIXEL_VSCOPE 0x31aa

Bit(s)	R/W	Default	Description
31:16	R/W	0	dec_pixel_bgn_v : uns, default = 0 ; // unit: pixel
15: 0	R/W	1079	dec_pixel_end_v : uns, default = 1079 ; // unit: pixel

SRSHARP0_SHARP_HVSIZE 0x3200

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

SRSHARP0_SHARP_HVBLANK_NUM 0x3201

Bit(s)	R/W	Default	Description
15: 8	R/W	0d20	reg_pknr_hblank_num : . unsigned , default = 20
7: 0	R/W	0d60	reg_pknr_vblank_num : . unsigned , default = 60

SRSHARP0_NR_GAUSSIAN_MODE 0x3202

Bit(s)	R/W	Default	Description
4	R/W	0d1	reg_nr_gau_ymode : : 0 3x3 filter; 1: 5x5 filter . unsigned , default = 1
0	R/W	0d1	reg_nr_gau_cmode : : 0 3x3 filter; 1: 5x5 filter . unsigned , default = 1

SRSHARP0_PK_HVCON_LPF_MODE 0x3203

Bit(s)	R/W	Default	Description
29:28	R/W	0d2	reg_pk_hconhpf_mode : : 0: no vertical filter;1:[1 2 1]'/4 filter; 2/3: [1 2 2 2 1]'/8 filter . unsigned , default = 2
25:24	R/W	0d2	reg_pk_hconbpf_mode : : 0: no vertical filter;1:[1 2 1]'/4 filter; 2/3: [1 2 2 2 1]'/8 filter . unsigned , default = 2
21:20	R/W	0d2	reg_pk_hconlbpf_mode : : 0: no vertical filter;1:[1 2 1]'/4 filter; 2/3: [1 2 2 2 1]'/8 filter . unsigned , default = 2
17:16	R/W	0d2	reg_pk_hconllbpf_mode : : 0: no vertical filter;1:[1 2 1]'/4 filter; 2/3: [1 2 2 2 1]'/8 filter . unsigned , default = 2

13:12	R/W	0d2	reg_pk_vconhpf_mode : : 0: no horizontal filter;1:[1 2 1]/4 filter; 2: [1 2 2 2 1]/8 filter; 3: [1 2 3 4 3 2 1]/16 filter. unsigned , default = 2
9: 8	R/W	0d2	reg_pk_vconbpf_mode : : 0: no horizontal filter;1:[1 2 1]/4 filter; 2: [1 2 2 2 1]/8 filter; 3: [1 2 3 4 3 2 1]/16 filter. unsigned , default = 2
5: 4	R/W	0d2	reg_pk_vconlbpf_mode : : 0: no horizontal filter;1:[1 2 1]/4 filter; 2: [1 2 2 2 1]/8 filter; 3: [1 2 3 4 3 2 1]/16 filter. unsigned , default = 2
1: 0	R/W	0d2	reg_pk_vconllbpf_mode : : 0: no horizontal filter;1:[1 2 1]/4 filter; 2: [1 2 2 2 1]/8 filter; 3: [1 2 3 4 3 2 1]/16 filter. unsigned , default = 2

SRSHARP0_PK_CON_BLEND_GAIN 0x3204

Bit(s)	R/W	Default	Description
31:28	R/W	0d4	reg_pk_hpcon_hpgain : : 8 as normalized 1 . unsigned , default = 4
27:24	R/W	0d4	reg_pk_hpcon_bpgain : : 8 as normalized 1 . unsigned , default = 4
23:20	R/W	0d0	reg_pk_hpcon_lbpgain : : 8 as normalized 1 . unsigned , default = 0
19:16	R/W	0d0	reg_pk_hpcon_llbpgain : : 8 as normalized 1 . unsigned , default = 0
15:12	R/W	0d0	reg_pk_bpcon_hpgain : : 8 as normalized 1 . unsigned , default = 0
11: 8	R/W	0d2	reg_pk_bpcon_bpgain : : 8 as normalized 1 . unsigned , default = 2
7: 4	R/W	0d6	reg_pk_bpcon_lbpgain : : 8 as normalized 1 . unsigned , default = 6
3: 0	R/W	0d0	reg_pk_bpcon_llbpgain : : 8 as normalized 1 . unsigned , default = 0

SRSHARP0_PK_CON_2CIRHPGAIN_TH_RATE 0x3205

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

SRSHARP0_PK_CON_2CIRHPGAIN_LIMIT 0x3206

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

SRSHARP0_PK_CON_2CIRBPGAIN_TH_RATE 0x3207

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

SRSHARP0_PK_CON_2CIRBPGAIN_LIMIT 0x3208

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

SRSHARP0_PK_CON_2DRTHPGAIN_TH_RATE 0x3209

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

SRSHARP0_PK_CON_2DRTHPGAIN_LIMIT 0x320a

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

SRSHARP0_PK_CON_2DRTBPGAIN_TH_RATE 0x320b

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

SRSHARP0_PK_CON_2DRTBPGAIN_LIMIT 0x320c

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

SRSHARP0_PK_CIRFB_LPF_MODE 0x320d

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

SRSHARP0_PK_DRTFB_LPF_MODE 0x320e

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

SRSHARP0_PK_CIRFB_HP_CORING 0x320f

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

SRSHARP0_PK_CIRFB_BP_CORING 0x3210

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

SRSHARP0_PK_DRTFB_HP_CORING 0x3211

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

SRSHARP0_PK_DRTFB_BP_CORING 0x3212

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

SRSHARP0_PK_CIRFB_BLEND_GAIN 0x3213

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

SRSHARP0_NR_ALPY_SSD_GAIN_OFST 0x3214

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

SRSHARP0_NR_ALP0Y_ERR2CURV_TH_RATE 0x3215

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

SRSHARP0_NR_ALP0Y_ERR2CURV_LIMIT 0x3216

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

SRSHARP0_NR_ALP0C_ERR2CURV_TH_RATE 0x3217

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

SRSHARP0_NR_ALP0C_ERR2CURV_LIMIT 0x3218

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

SRSHARP0_NR_ALP0_MIN_MAX 0x3219

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

SRSHARP0_NR_ALP1_MIERR_CORING 0x321a

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

SRSHARP0_NR_ALP1_ERR2CURV_TH_RATE 0x321b

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

SRSHARP0_NR_ALP1_ERR2CURV_LIMIT 0x321c

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

SRSHARP0_NR_ALP1_MIN_MAX 0x321d

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

SRSHARP0_PK_ALP2_MIERR_CORING 0x321e

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

SRSHARP0_PK_ALP2_ERR2CURV_TH_RATE 0x321f

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

SRSHARP0_PK_ALP2_ERR2CURV_LIMIT 0x3220

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

SRSHARP0_PK_ALP2_MIN_MAX 0x3221

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

SRSHARP0_PK_FINALGAIN_HP_BP 0x3222

Bit(s)	R/W	Default	Description
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

SRSHARP0_PK_OS_HORZ_CORE_GAIN 0x3223

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 : : midd 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 : : midd gain (to current pixel) to adaptive overshoot margin in horizontal direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20

SRSHARP0_PK_OS_VERT_CORE_GAIN 0x3224

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 : : midd 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 : : midd gain (to current pixel) to adaptive overshoot margin in vertical direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20

SRSHARP0_PK_OS_ADPT_MISC 0x3225

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 : : minimun limit adaptive overshoot margin (1x). 0~255; . unsigned , default = 20

SRSHARP0_PK_OS_STATIC 0x3226

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

SRSHARP0_PK_NR_ENABLE 0x3227

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

SRSHARP0_PK_DRT_SAD_MISC 0x3228

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
8	R/W	0d0	reg_pk_debug_edge : : show color for edge . 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

SRSHARP0_NR_TI_DNLP_BLEND 0x3229

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

SRSHARP0_TI_DIR_CORE_ALPHA 0x322a

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

SRSHARP0_CTI_DIR_ALPHA 0x322b

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

SRSHARP0_LTI_CTI_DF_GAIN 0x322c

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

SRSHARP0_LTI_CTI_DIR_AC_DBG 0x322d

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 recomended). 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

SRSHARP0_HCTI_FLT_CLP_DC 0x322e

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_flt : , 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_flt : , 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_flt : , 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_flt : , 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_flt : , 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_flt : , 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

SRSHARP0_HCTI_BST_GAIN 0x322f

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

SRSHARP0_HCTI_BST_CORE 0x3230

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

SRSHARP0_HCTI_CON_2_GAIN_0 0x3231

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

SRSHARP0_HCTI_CON_2_GAIN_1 0x3232

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

SRSHARP0_HCTI_OS_MARGIN 0x3233

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

SRSHARP0_HLTI_FLT_CLP_DC 0x3234

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_flt : , 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_flt : , 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_flt : , 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_flt : , 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_flt : , 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_flt : , 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_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

SRSHARP0_HLTI_BST_GAIN 0x3235

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

SRSHARP0_HLTI_BST_CORE 0x3236

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

SRSHARP0_HLTI_CON_2_GAIN_0 0x3237

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 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

SRSHARP0_HLTI_CON_2_GAIN_1 0x3238

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

SRSHARP0_HLTI_OS_MARGIN 0x3239

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

SRSHARP0_VLTI_FLT_CON_CLP 0x323a

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_flt : : 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_flt : : 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_flt : : 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_flt : : 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

SRSHARP0_VLTI_BST_GAIN 0x323b

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

SRSHARP0_VLTI_BST_CORE 0x323c

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

SRSHARP0_VLTI_CON_2_GAIN_0 0x323d

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

SRSHARP0_VLTI_CON_2_GAIN_1 0x323e

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

SRSHARP0_VCTI_FLT_CON_CLP 0x323f

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_flt : : 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_dxn_flt : : 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_han_flt : : 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_dan_flt : : 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

SRSHARP0_VCTI_BST_GAIN 0x3240

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

SRSHARP0_VCTI_BST_CORE 0x3241

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

SRSHARP0_VCTI_CON_2_GAIN_0 0x3242

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

SRSHARP0_VCTI_CON_2_GAIN_1 0x3243

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

SRSHARP0_SHARP_3DLIMIT 0x3244

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

SRSHARP0_DNLP_EN 0x3245

Bit(s)	R/W	Default	Description
0	R/W	0d1	reg_dnlp_en : . unsigned , default = 1

SRSHARP0_DNLP_00 0x3246

Bit(s)	R/W	Default	Description
31: 0	R/W	0x08060402	reg_dnlp_ygrid0 : : dnlp00 . unsigned , default = 0x08060402

SRSHARP0_DNLP_01 0x3247

Bit(s)	R/W	Default	Description
31: 0	R/W	0x100e0c0a	reg_dnlp_ygrid1 : : dnlp01 . unsigned , default = 0x100e0c0a

SRSHARP0_DNLP_02 0x3248

Bit(s)	R/W	Default	Description
31: 0	R/W	0x1a171412	reg_dnlp_ygrid2 : : dnlp02 . unsigned , default = 0x1a171412

SRSHARP0_DNLP_03 0x3249

Bit(s)	R/W	Default	Description
31: 0	R/W	0x2824201d	reg_dnlp_ygrid3 : : dnlp03 . unsigned , default = 0x2824201d

SRSHARP0_DNLP_04 0x324a

Bit(s)	R/W	Default	Description
31: 0	R/W	0x3834302c	reg_dnlp_ygrid4 : : dnlp04 . unsigned , default = 0x3834302c

SRSHARP0_DNLP_05 0x324b

Bit(s)	R/W	Default	Description
31: 0	R/W	0x4b45403c	reg_dnlp_ygrid5 : : dnlp05 . unsigned , default = 0x4b45403c

SRSHARP0_DNLP_06 0x324c

Bit(s)	R/W	Default	Description
31: 0	R/W	0x605b5550	reg_dnlp_ygrid6 : : dnlp06 . unsigned , default = 0x605b5550

SRSHARP0_DNLP_07 0x324d

Bit(s)	R/W	Default	Description
31: 0	R/W	0x80787068	reg_dnlp_ygrid7 : : dnlp07 . unsigned , default = 0x80787068

SRSHARP0_DNLP_08 0x324e

Bit(s)	R/W	Default	Description
31: 0	R/W	0xa0989088	reg_dnlp_ygrid8 : : dnlp08 . unsigned , default = 0xa0989088

SRSHARP0_DNLP_09 0x324f

Bit(s)	R/W	Default	Description
31: 0	R/W	0xb8b2aca6	reg_dnlp_ygrid9 : : dnlp09 . unsigned , default = 0xb8b2aca6

SRSHARP0_DNLP_10 0x3250

Bit(s)	R/W	Default	Description
31: 0	R/W	0xc8c4c0bc	reg_dnlp_ygrid10 : : dnlp10 . unsigned , default = 0xc8c4c0bc

SRSHARP0_DNLP_11 0x3251

Bit(s)	R/W	Default	Description
31: 0	R/W	0xd4d2cecb	reg_dnlp_ygrid11 : : dnlp11 . unsigned , default = 0xd4d2cecb

SRSHARP0_DNLP_12 0x3252

Bit(s)	R/W	Default	Description
31: 0	R/W	0xdad8d7d6	reg_dnlp_ygrid12 : : dnlp12 . unsigned , default = 0xdad8d7d6

SRSHARP0_DNLP_13 0x3253

Bit(s)	R/W	Default	Description
31: 0	R/W	0xe2e0dedc	reg_dnlp_ygrid13 : : dnlp13 . unsigned , default = 0xe2e0dedc

SRSHARP0_DNLP_14 0x3254

Bit(s)	R/W	Default	Description
31: 0	R/W	0xf0ece8e4	reg_dnlp_ygrid14 : : dnlp14 . unsigned , default = 0xf0ece8e4

SRSHARP0_DNLP_15 0x3255

Bit(s)	R/W	Default	Description
31: 0	R/W	0xfffcf8f4	reg_dnlp_ygrid15 : : dnlp15 . unsigned , default = 0xfffcf8f4

SRSHARP0_DEMO_CRTL 0x3256

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

SRSHARP0_SHARP_SR2_CTRL 0x3257

Bit(s)	R/W	Default	Description
31:25	R/W		reserved
24	R/W	0	sr2_dejaggy_en, 1 to enable dejaggy
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.

SRSHARP0_SHARP_SR2_YBIC_HCOEF0 0x3258

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

SRSHARP0_SHARP_SR2_YBIC_HCOEF1 0x3259

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

SRSHARP0_SHARP_SR2_CBIC_HCOEF0 0x325a

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

SRSHARP0_SHARP_SR2_CBIC_HCOEF1 0x325b

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

SHARP_SR2_YBIC_VCOEF0 0x325c

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

SRSHARP0_SHARP_SR2_YBIC_VCOEF1 0x325d

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

SRSHARP0_SHARP_SR2_CBIC_VCOEF0 0x325e

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

SRSHARP0_SHARP_SR2_CBIC_VCOEF1 0x325f

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

SRSHARP0_SHARP_SR2_MISC 0x3260

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

SRSHARP0_SHARP_DEJ2_PRC 0x3261

Bit(s)	R/W	Default	Description
31:24	R/W	5	sr2_dejaggy2_hcon_thrd, hcon threshold, only pixels with hcon equal or larger than this value can be detected as jaggy2
23:16	R/W	30	sr2_dejaggy2_svdif_thrd, abs(sum(vdif[4])) threshold to decide jaggy2.
15: 8	R/W	32	sr2_dejaggy2_svdif_rate, sum(abs(vdif[4])) <= (rate*abs(sum(vdif[4]))/16), rate to decide jaggy2
7: 0	R/W	-3	sr2_dejaggy2_vdif_thrd, vdif threshold for same trend decision, these value is the margin for not same trend, if >0, means need to be same trend, if <0, can be a little bit glitch.

SRSHARP0_SHARP_DEJ1_PRC 0x3262

Bit(s)	R/W	Default	Description
31:24	R/W	1	sr2_dejaggy1_hcon_thrd
23:16	R/W	50	sr2_dejaggy1_svdif_thrd
15: 8	R/W	64	sr2_dejaggy1_svdif_rate
5: 0	R/W	16	sr2_dejaggy1_ dif12_rate

SRSHARP0_SHARP_DEJ2_MISC 0x3263

Bit(s)	R/W	Default	Description
31:8	R/W		reserved
7	R/W	1	sr2_dejaggy2_proc_chrm, enable to filter 2 pixels step on chroma
6	R/W	1	sr2_dejaggy2_proc_luma, enable to filter 2 pixels step on luma
5:4	R/W	3	sr2_dejaggy2_extend_mode, extend mode for dejaggy2 horizontally, 0: no extend; 1: extend 1 pixel; 2: extend 2 pixels; 3: extend 3 pixels
3	R/W	0	sr2_dejaggy2_alpha_force, force enable of the alpha for dejaggy2
2: 0	R/W	0	sr2_dejaggy2_alpha_value, forced value of alpha for dejaggy2

SRSHARP0_SHARP_DEJ1_MISC 0x3264

Bit(s)	R/W	Default	Description
31:12	R/W		reserved
11:8	R/W	2	sr2_dejaggy1_svdif_ofst, sum(abs(vdif[4])) >= (rate*abs(sum(vdif[4]))/32+ofst), offset to decide jaggy1

7	R/W	1	sr2_dejaggy1_proc_chrm
6	R/W	1	sr2_dejaggy1_proc_luma
5:4	R/W	3	sr2_dejaggy1_extend_mode
3	R/W	0	sr2_dejaggy1_alpha_force
2: 0	R/W	0	sr2_dejaggy1_alpha_value

SRSHARP1_SHARP_HVSIZE 0x3280

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

SRSHARP1_SHARP_HVBLANK_NUM 0x3281

Bit(s)	R/W	Default	Description
15: 8	R/W	0d20	reg_pknr_hblank_num : . unsigned , default = 20
7: 0	R/W	0d60	reg_pknr_vblank_num : . unsigned , default = 60

SRSHARP1_NR_GAUSSIAN_MODE 0x3282

Bit(s)	R/W	Default	Description
4	R/W	0d1	reg_nr_gau_ymode : : 0 3x3 filter; 1: 5x5 filter . unsigned , default = 1
0	R/W	0d1	reg_nr_gau_cmode : : 0 3x3 filter; 1: 5x5 filter . unsigned , default = 1

SRSHARP1_PK_HVCON_LPF_MODE 0x3283

Bit(s)	R/W	Default	Description
29:28	R/W	0d2	reg_pk_hconhpf_mode : : 0: no vertical filter;1:[1 2 1]'/4 filter; 2/3: [1 2 2 2 1]'/8 filter . unsigned , default = 2
25:24	R/W	0d2	reg_pk_hconbpf_mode : : 0: no vertical filter;1:[1 2 1]'/4 filter; 2/3: [1 2 2 2 1]'/8 filter . unsigned , default = 2
21:20	R/W	0d2	reg_pk_hconlbpf_mode : : 0: no vertical filter;1:[1 2 1]'/4 filter; 2/3: [1 2 2 2 1]'/8 filter . unsigned , default = 2
17:16	R/W	0d2	reg_pk_hconllbpf_mode : : 0: no vertical filter;1:[1 2 1]'/4 filter; 2/3: [1 2 2 2 1]'/8 filter . unsigned , default = 2
13:12	R/W	0d2	reg_pk_vconhpf_mode : : 0: no horizontal filter;1:[1 2 1]/4 filter; 2: [1 2 2 2 1]/8 filter; 3: [1 2 3 4 3 2 1]/16 filter. unsigned , default = 2
9: 8	R/W	0d2	reg_pk_vconbpf_mode : : 0: no horizontal filter;1:[1 2 1]/4 filter; 2: [1 2 2 2 1]/8 filter; 3: [1 2 3 4 3 2 1]/16 filter. unsigned , default = 2
5: 4	R/W	0d2	reg_pk_vconlbpf_mode : : 0: no horizontal filter;1:[1 2 1]/4 filter; 2: [1 2 2 2 1]/8 filter; 3: [1 2 3 4 3 2 1]/16 filter. unsigned , default = 2
1: 0	R/W	0d2	reg_pk_vconllbpf_mode : : 0: no horizontal filter;1:[1 2 1]/4 filter; 2: [1 2 2 2 1]/8 filter; 3: [1 2 3 4 3 2 1]/16 filter. unsigned , default = 2

SRSHARP1_PK_CON_BLEND_GAIN 0x3284

Bit(s)	R/W	Default	Description
31:28	R/W	0d4	reg_pk_hpcon_hpgain : : 8 as normalized 1 . unsigned , default = 4
27:24	R/W	0d4	reg_pk_hpcon_bpgain : : 8 as normalized 1 . unsigned , default = 4
23:20	R/W	0d0	reg_pk_hpcon_lbpgain : : 8 as normalized 1 . unsigned , default = 0
19:16	R/W	0d0	reg_pk_hpcon_llbpgain : : 8 as normalized 1 . unsigned , default = 0
15:12	R/W	0d0	reg_pk_bpcon_hpgain : : 8 as normalized 1 . unsigned , default = 0
11: 8	R/W	0d2	reg_pk_bpcon_bpgain : : 8 as normalized 1 . unsigned , default = 2
7: 4	R/W	0d6	reg_pk_bpcon_lbpgain : : 8 as normalized 1 . unsigned , default = 6
3: 0	R/W	0d0	reg_pk_bpcon_llbpgain : : 8 as normalized 1 . unsigned , default = 0

SRSHARP1_PK_CON_2CIRHPGAIN_TH_RATE 0x3285

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

SRSHARP1_PK_CON_2CIRHPGAIN_LIMIT 0x3286

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

SRSHARP1_PK_CON_2CIRBPGAIN_TH_RATE 0x3287

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

SRSHARP1_PK_CON_2CIRBPGAIN_LIMIT 0x3288

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

SRSHARP1_PK_CON_2DRTHPGAIN_TH_RATE 0x3289

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

SRSHARP1_PK_CON_2DRTHPGAIN_LIMIT 0x328a

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

SRSHARP1_PK_CON_2DRTBPGAIN_TH_RATE 0x328b

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

SRSHARP1_PK_CON_2DRTBPGAIN_LIMIT 0x328c

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

SRSHARP1_PK_CIRFB_LPF_MODE 0x328d

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

SRSHARP1_PK_DRTFB_LPF_MODE 0x328e

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

SRSHARP1_PK_CIRFB_HP_CORING 0x328f

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

SRSHARP1_PK_CIRFB_BP_CORING 0x3290

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

SRSHARP1_PK_DRTFB_HP_CORING 0x3291

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

SRSHARP1_PK_DRTFB_BP_CORING 0x3292

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

SRSHARP1_PK_CIRFB_BLEND_GAIN 0x3293

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

SRSHARP1_NR_ALPY_SSD_GAIN_OFST 0x3294

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

SRSHARP1_NR_ALP0Y_ERR2CURV_TH_RATE 0x3295

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

SRSHARP1_NR_ALP0Y_ERR2CURV_LIMIT 0x3296

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

SRSHARP1_NR_ALP0C_ERR2CURV_TH_RATE 0x3297

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

SRSHARP1_NR_ALP0C_ERR2CURV_LIMIT 0x3298

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

SRSHARP1_NR_ALP0_MIN_MAX 0x3299

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

SRSHARP1_NR_ALP1_MIERR_CORING 0x329a

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

SRSHARP1_NR_ALP1_ERR2CURV_TH_RATE 0x329b

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

SRSHARP1_NR_ALP1_ERR2CURV_LIMIT 0x329c

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

SRSHARP1_NR_ALP1_MIN_MAX 0x329d

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

SRSHARP1_PK_ALP2_MIERR_CORING 0x329e

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

SRSHARP1_PK_ALP2_ERR2CURV_TH_RATE 0x329f

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

SRSHARP1_PK_ALP2_ERR2CURV_LIMIT 0x32a0

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

SRSHARP1_PK_ALP2_MIN_MAX 0x32a1

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

SRSHARP1_PK_FINALGAIN_HP_BP 0x32a2

Bit(s)	R/W	Default	Description
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

SRSHARP1_PK_OS_HORZ_CORE_GAIN 0x32a3

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 : : midd 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 : : midd gain (to current pixel) to adaptive overshoot margin in horizontal direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20

SRSHARP1_PK_OS_VERT_CORE_GAIN 0x32a4

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 : : midd 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 : : midd gain (to current pixel) to adaptive overshoot margin in vertical direction. normalized to 32 as '1'. 0~255;. unsigned , default = 20

SRSHARP1_PK_OS_ADPT_MISC 0x32a5

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 : : minimun limit adaptive overshoot margin (1x). 0~255; . unsigned , default = 20

SRSHARP1_PK_OS_STATIC 0x32a6

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

SRSHARP1_PK_NR_ENABLE 0x32a7

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

SRSHARP1_PK_DRT_SAD_MISC 0x32a8

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
8	R/W	0d0	reg_pk_debug_edge : : show color for edge . 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

SRSHARP1_NR_TI_DNLP_BLEND 0x32a9

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

SRSHARP1_TI_DIR_CORE_ALPHA 0x32aa

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

SRSHARP1_CTI_DIR_ALPHA 0x32ab

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

SRSHARP1_LTI_CTI_DF_GAIN 0x32ac

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

SRSHARP1_LTI_CTI_DIR_AC_DBG 0x32ad

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 recomended). 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

SRSHARP1_HCTI_FLT_CLP_DC 0x32ae

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_flt : , 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_flt : , 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_flt : , 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_flt : , 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_flt : , 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_flt : , 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

SRSHARP1_HCTI_BST_GAIN 0x32af

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

SRSHARP1_HCTI_BST_CORE 0x32b0

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

SRSHARP1_HCTI_CON_2_GAIN_0 0x32b1

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

SRSHARP1_HCTI_CON_2_GAIN_1 0x32b2

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

SRSHARP1_HCTI_OS_MARGIN 0x32b3

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

SRSHARP1_HLTI_FLT_CLP_DC 0x32b4

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_flt : , 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_flt : , 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_flt : , 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_flt : , 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_flt : , 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_flt : , 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_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

SRSHARP1_HLTI_BST_GAIN 0x32b5

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

SRSHARP1_HLTI_BST_CORE 0x32b6

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

SRSHARP1_HLTI_CON_2_GAIN_0 0x32b7

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 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

SRSHARP1_HLTI_CON_2_GAIN_1 0x32b8

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

SRSHARP1_HLTI_OS_MARGIN 0x32b9

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

SRSHARP1_VLTI_FLT_CON_CLP 0x32ba

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_flt : : 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_flt : : 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_flt : : 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_flt : : 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

SRSHARP1_VLTI_BST_GAIN 0x32bb

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

SRSHARP1_VLTI_BST_CORE 0x32bc

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

SRSHARP1_VLTI_CON_2_GAIN_0 0x32bd

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

SRSHARP1_VLTI_CON_2_GAIN_1 0x32be

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

SRSHARP1_VCTI_FLT_CON_CLP 0x32bf

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_flt : : 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_dxn_flt : : 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_han_flt : : 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_dan_flt : : 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

SRSHARP1_VCTI_BST_GAIN 0x32c0

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

SRSHARP1_VCTI_BST_CORE 0x32c1

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

SRSHARP1_VCTI_CON_2_GAIN_0 0x32c2

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

SRSHARP1_VCTI_CON_2_GAIN_1 0x32c3

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

SRSHARP1_SHARP_3DLIMIT 0x32c4

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

SRSHARP1_DNLP_EN 0x32c5

Bit(s)	R/W	Default	Description
0	R/W	0d1	reg_dnlp_en : . unsigned , default = 1

SRSHARP1_DNLP_00 0x32c6

Bit(s)	R/W	Default	Description
31: 0	R/W	0x08060402	reg_dnlp_ygrid0 : : dnlp00 . unsigned , default = 0x08060402

SRSHARP1_DNLP_01 0x32c7

Bit(s)	R/W	Default	Description
31: 0	R/W	0x100e0c0a	reg_dnlp_ygrid1 : : dnlp01 . unsigned , default = 0x100e0c0a

SRSHARP1_DNLP_02 0x32c8

Bit(s)	R/W	Default	Description
31: 0	R/W	0x1a171412	reg_dnlp_ygrid2 : : dnlp02 . unsigned , default = 0x1a171412

SRSHARP1_DNLP_03 0x32c9

Bit(s)	R/W	Default	Description
31: 0	R/W	0x2824201d	reg_dnlp_ygrid3 : : dnlp03 . unsigned , default = 0x2824201d

SRSHARP1_DNLP_04 0x32ca

Bit(s)	R/W	Default	Description
31: 0	R/W	0x3834302c	reg_dnlp_ygrid4 : : dnlp04 . unsigned , default = 0x3834302c

SRSHARP1_DNLP_05 0x32cb

Bit(s)	R/W	Default	Description
31: 0	R/W	0x4b45403c	reg_dnlp_ygrid5 : : dnlp05 . unsigned , default = 0x4b45403c

SRSHARP1_DNLP_06 0x32cc

Bit(s)	R/W	Default	Description
31: 0	R/W	0x605b5550	reg_dnlp_ygrid6 : : dnlp06 . unsigned , default = 0x605b5550

SRSHARP1_DNLP_07 0x32cd

Bit(s)	R/W	Default	Description
31: 0	R/W	0x80787068	reg_dnlp_ygrid7 : : dnlp07 . unsigned , default = 0x80787068

SRSHARP1_DNLP_08 0x32ce

Bit(s)	R/W	Default	Description
31: 0	R/W	0xa0989088	reg_dnlp_ygrid8 : : dnlp08 . unsigned , default = 0xa0989088

SRSHARP1_DNLP_09 0x32cf

Bit(s)	R/W	Default	Description
31: 0	R/W	0xb8b2aca6	reg_dnlp_ygrid9 : : dnlp09 . unsigned , default = 0xb8b2aca6

SRSHARP1_DNLP_10 0x32d0

Bit(s)	R/W	Default	Description
31: 0	R/W	0xc8c4c0bc	reg_dnlp_ygrid10 : : dnlp10 . unsigned , default = 0xc8c4c0bc

SRSHARP1_DNLP_11 0x32d1

Bit(s)	R/W	Default	Description
31: 0	R/W	0xd4d2cecb	reg_dnlp_ygrid11 : : dnlp11 . unsigned , default = 0xd4d2cecb

SRSHARP1_DNLP_12 0x32d2

Bit(s)	R/W	Default	Description
31: 0	R/W	0xdad8d7d6	reg_dnlp_ygrid12 : : dnlp12 . unsigned , default = 0xdad8d7d6

SRSHARP1_DNLP_13 0x32d3

Bit(s)	R/W	Default	Description
31: 0	R/W	0xe2e0dedc	reg_dnlp_ygrid13 : : dnlp13 . unsigned , default = 0xe2e0dedc

SRSHARP1_DNLP_14 0x32d4

Bit(s)	R/W	Default	Description
31: 0	R/W	0xf0ece8e4	reg_dnlp_ygrid14 : : dnlp14 . unsigned , default = 0xf0ece8e4

SRSHARP1_DNLP_15 0x32d5

Bit(s)	R/W	Default	Description
31: 0	R/W	0xfffcf8f4	reg_dnlp_ygrid15 : : dnlp15 . unsigned , default = 0xfffcf8f4

SRSHARP1_DEMO_CRTL 0x32d6

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

SRSHARP1_SHARP_SR2_CTRL 0x32d7

Bit(s)	R/W	Default	Description
31:25	R/W		reserved
24	R/W	0	sr2_dejaggy_en, 1 to enable dejaggy
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.

SRSHARP1_SHARP_SR2_YBIC_HCOEF0 0x32d8

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

SRSHARP1_SHARP_SR2_YBIC_HCOEF1 0x32d9

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

SRSHARP1_SHARP_SR2_CBIC_HCOEF0 0x32da

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

SRSHARP1_SHARP_SR2_CBIC_HCOEF1 0x32db

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

SHARP_SR2_YBIC_VCOEF0 0x32dc

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

SRSHARP1_SHARP_SR2_YBIC_VCOEF1 0x32dd

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

SRSHARP1_SHARP_SR2_CBIC_VCOEF0 0x32de

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

SRSHARP1_SHARP_SR2_CBIC_VCOEF1 0x32df

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

SRSHARP1_SHARP_SR2_MISC 0x32e0

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

SRSHARP1_SHARP_DEJ2_PRC 0x32e1

Bit(s)	R/W	Default	Description
31:24	R/W	5	sr2_dejaggy2_hcon_thrd, hcon threshold, only pixels with hcon equal or larger than this value can be detected as jaggy2
23:16	R/W	30	sr2_dejaggy2_svdif_thrd, abs(sum(vdif[4])) threshold to decide jaggy2.
15: 8	R/W	32	sr2_dejaggy2_svdif_rate, sum(abs(vdif[4])) <= (rate*abs(sum(vdif[4]))/16), rate to decide jaggy2
7: 0	R/W	-3	sr2_dejaggy2_vdif_thrd, vdif threshold for same trend decision, these value is the margin for not same trend, if >0, means need to be same trend, if <0, can be a little bit glitch.

SRSHARP1_SHARP_DEJ1_PRC 0x32e2

Bit(s)	R/W	Default	Description
31:24	R/W	1	sr2_dejaggy1_hcon_thrd
23:16	R/W	50	sr2_dejaggy1_svdif_thrd
15: 8	R/W	64	sr2_dejaggy1_svdif_rate
5: 0	R/W	16	sr2_dejaggy1_dif12_rate

SRSHARP1_SHARP_DEJ2_MISC 0x32e3

Bit(s)	R/W	Default	Description
31:8	R/W		reserved
7	R/W	1	sr2_dejaggy2_proc_chrm, enable to filter 2 pixels step on chroma
6	R/W	1	sr2_dejaggy2_proc_luma, enable to filter 2 pixels step on luma
5:4	R/W	3	sr2_dejaggy2_extend_mode, extend mode for dejaggy2 horizontally, 0: no extend; 1: extend 1 pixel; 2: extend 2 pixels; 3: extend 3 pixels
3	R/W	0	sr2_dejaggy2_alpha_force, force enable of the alpha for dejaggy2
2: 0	R/W	0	sr2_dejaggy2_alpha_value, forced value of alpha for dejaggy2

SRSHARP1_SHARP_DEJ1_MISC 0x32e4

Bit(s)	R/W	Default	Description
31:12	R/W		reserved
11:8	R/W	2	sr2_dejaggy1_svdif_ofst, sum(abs(vdif[4])) >= (rate*abs(sum(vdif[4]))/32+ofst), offset to decide jaggy1
7	R/W	1	sr2_dejaggy1_proc_chrm
6	R/W	1	sr2_dejaggy1_proc_luma
5:4	R/W	3	sr2_dejaggy1_extend_mode
3	R/W	0	sr2_dejaggy1_alpha_force
2: 0	R/W	0	sr2_dejaggy1_alpha_value

OSD1 registers

VIU_OSD1_CTRL_STAT 0x1A10

Bit(s)	R/W	Default	Description
31	R	0	unused
30	R/W	0	ENABLE_FREE_CLK. 1 = Use free-running clock; 0 = Use gated clock to save power.
28	R	0	OSD_DONE. 1 = The last enabled OSD block has finished outputting; 0 = Not finished.
27-24	R	0	OSD_BLK_MODE: the input pixel format of which the current OSD block is being processed.
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.: Alpha_tmp = Alpha_internal + (Alpha_internal == 0 ? 0 : 1); Alpha_out = (Alpha_tmp * GLOBAL_ALPHA) / 256;
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.
0	R/W	0	OSD_BLK_ENABLE: Each bit to enable display an OSD block, ~~four~~one blocks in total. E.g. Bit[0]: 1 = Enable displaying block 0; 0 = Disable displaying block 0.

VIU_OSD1_CTRL_STAT2 0x1A2d

Bit(s)	R/W	Default	Description
31-16	R	0	unused
15	R/W	0	osd_dpath_sel 0-osd1 mif 1-vpu mali afbcd Bit 14 , RW, replaced_alpha_en
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

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.

Bit(s)	R/W	Default	Description
31-30	R/W	0	Reserved
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. Virtual canvas LUT entry.
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. 0-2 = unused; 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;
7	R/W	0	RGB_EN. 1 = Enable conversion of input pixel's R/G/B value to output Y/Cb/Cr value. 0 = Disable.
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; 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; 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.

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.

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.

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.

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.

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 2words 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.
17-12	R/W	32	FIFO_DEPTH_VAL: Define the depth of FIFO which stores 64-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.
2-1	R/W	0	ENDIAN: define the endianess of the 64-bit data stored in memory, and how to convert. 0 = No convertion; 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.

VIU_OSD1_MATRIX_CTRL 0x1A90

Bit(s)	R/W	Default	Description
9-8	R/W	0	Matrix coef index selection, 00: select post matrix, 01: select video1 matrix, 10: select video2 matrix
5	R/W	0	Video1 conversion matrix enable
4	R/W	0	Video2 conversion matrix enable
2	R/W	0	Output y/cb/cr saturation enable, only for post matrix(y saturate to 16-235, cb/cr saturate to 0-240)
1	R/W	0	input y/cb/cr saturation enable, only for post matrix(y saturate 16-235, cb/cr saturate to 16-240)
0	R/W	0	conversion matrix enable

VIU_OSD1_MATRIX_COEF00_01 0x1A91

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient00, signed, 3.10
12-0	R/W	0	Coefficient01, signed, 3.10

VIU_OSD1_MATRIX_COEF02_10 0x1A92

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient02, signed, 3.10
12-0	R/W	0	Coefficient10, signed, 3.10

VIU_OSD1_MATRIX_COEF11_12 0x1A93

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient11, signed, 3.10
12-0	R/W	0	Coefficient12, signed, 3.10

VIU_OSD1_MATRIX_COEF20_21 0x1A94

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient20, signed, 3.10
12-0	R/W	0	Coefficient21, signed, 3.10

VIU_OSD1_MATRIX_COLMOD_COEF42 0x1A95

Bit(s)	R/W	Default	Description
18-16	R/W	0	convrs
12-0	R/W	0	Coefficient42, signed, 3.10

VIU_OSD1_MATRIX_OFFSET0_1 0x1A96

Bit(s)	R/W	Default	Description
26-16	R/W	0	Offset0, signed value
10-0	R/W	0	Offset1, signed value

VIU_OSD1_MATRIX_OFFSET2 0x1A97

Bit(s)	R/W	Default	Description
10-0	R/W	0	Offset2, signed value

VIU_OSD1_MATRIX_PRE_OFFSET0_1 0x1A98

Bit(s)	R/W	Default	Description
27-16	R/W	0	pre_Offset0, signed value
11-0	R/W	0	Pre_Offset1, signed value

VIU_OSD1_MATRIX_PRE_OFFSET2 0x1A99

Bit(s)	R/W	Default	Description
11-0	R/W	0	Pre_Offset2, signed value

VIU_OSD1_MATRIX_COEF22_30 0x1A9d

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef22
12-0	R/W	0	Coef30

VIU_OSD1_MATRIX_COEF31_32 0x1A9e

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef31
12-0	R/W	0	Coef32

VIU_OSD1_MATRIX_COEF40_41 0x1A9f

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef40
12-0	R/W	0	Coef41

VIU_OSD1_EOTF_CTL 0x1Ad4

Bit(s)	R/W	Default	Description
31:27	R/W	0	[31] for all eotf enable, [30] for matrix3x3 enable, [29:27] for eotf_ch0~3
17-6	R/W	0	For clock gate control
5-4	R/W	0	Pscale_mode for ch2
3-2	R/W	0	Pscale_mode for ch1
1-0	R/W	0	Pscale_mode for ch0

VIU_OSD1_EOTF_COEF00_01 0x1Ad5

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef00
12-0	R/W	0	Coef01

VIU_OSD1_EOTF_COEF02_10 0x1Ad6

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef02
12-0	R/W	0	Coef10

VIU_OSD1_EOTF_COEF11_12 0x1Ad7

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef11
12-0	R/W	0	Coef12

VIU_OSD1_EOTF_COEF20_21 0x1Ad8

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef20
12-0	R/W	0	Coef21

VIU_OSD1_EOTF_COEF22_RS 0x1Ad9

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef22
12-0	R/W	0	Coef_rs

VIU_OSD1_EOTF_LUT_ADDR_PORT 0x1Ada

Bit(s)	R/W	Default	Description
31-0	R/W	0	The lut is addr(33*3) * 16bits. Eatch addr has 2 16bits. So biggest valid address is 33*3/2-1 =49

VIU_OSD1_EOTF_LUT_DATA_PORT 0x1Adb

Bit(s)	R/W	Default	Description
31-16	R/W	0	For No.(address*2+1) 16bits. 0~32 16bits is for r , 33~65 for g, 66~98 for b
15-0	R/W	0	For No(address*2) 16bits.

VIU_OSD1_OETF_CTL 0x1Adc

Bit(s)	R/W	Default	Description
31-29	R/W	0	for eotf_ch0~3
21-12	R/W	0	For clock gate control
11-8	R/W	0	Oetf_scl for ch2
7-4	R/W	0	Oetf_scl for ch1
3-0	R/W	0	Oetf_scl for ch0

VIU_OSD1_OETF_LUT_ADDR_PORT 0x1Add

Bit(s)	R/W	Default	Description
31-0	R/W	0	The lut is addr(41*3) * 16bits. Eatch addr has 2 16bits. So biggest valid address is 41*3/2-1 =61

VIU_OSD1_OETF_LUT_DATA_PORT 0x1Ade

Bit(s)	R/W	Default	Description
31-16	R/W	0	For No.(address*2+1) 16bits. 0~40 16bits is for r , 41~81 for g, 82~122 for b
15-0	R/W	0	For No(address*2) 16bits.

OSD2 registers

VIU_OSD2_CTRL_STAT 0x1A30

Bit(s)	R/W	Default	Description
31	R	0	unused
30	R/W	0	ENABLE_FREE_CLK. 1 = Use free-running clock; 0 = Use gated clock to save power.
28	R	0	OSD_DONE. 1 = The last enabled OSD block has finished outputting; 0 = Not finished.
27-24	R	0	OSD_BLK_MODE: the input pixel format of which the current OSD block is being processed.
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.: Alpha_tmp = Alpha_internal + (Alpha_internal == 0 ? 0 : 1); Alpha_out = (Alpha_tmp * GLOBAL_ALPHA) / 256;
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.
0	R/W	0	OSD_BLK_ENABLE: Each bit to enable display an OSD block, ~~four~~ one blocks in total. E.g. Bit[0]: 1 = Enable displaying block 0; 0 = Disable displaying block 0.

VIU_OSD2_CTRL_STAT2 0x1A4d

Bit(s)	R/W	Default	Description
31-15	R	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~~	~~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.~~
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_OSD2_TCOLOR_AG0 0x1A37

VIU_OSD2_TCOLOR_AG1 0x1A38

VIU_OSD2_TCOLOR_AG2 0x1A39

VIU_OSD2_TCOLOR_AG3 0x1A3a

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_OSD2_BLK0_CFG_W0 0x1A3b

VIU_OSD2_BLK1_CFG_W0 0x1A3f

VIU_OSD2_BLK2_CFG_W0 0x1A43

VIU_OSD2_BLK3_CFG_W0 0x1A47

Defines display block 0/1/2/3's property, word 0.

Bit(s)	R/W	Default	Description
31-24	R/W	0	Unused.
23-16	R/W	0	TBL_ADDR. Virtual canvas LUT entry.
15	R/W	0	LITTLE_ENDIAN: define the endianess 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. 0-2 = unused; 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_OSD2_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;
7	R/W	0	RGB_EN. 1 = Enable conversion of input pixel's R/G/B value to output Y/Cb/Cr value. 0 = Disable.
6	R/W	0	TC_ALPHA_EN: refer to reg VIU_OSD2_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_OSD2_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_OSD2_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_OSD2_CTRL_STAT.CTRL_MTCH_Y; 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_OSD2_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.

VIU_OSD2_BLK0_CFG_W1 0x1A3c

VIU_OSD2_BLK1_CFG_W1 0x1A40

VIU_OSD2_BLK2_CFG_W1 0x1A44

VIU_OSD2_BLK3_CFG_W1 0x1A48

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.

VIU_OSD2_BLK0_CFG_W2 0x1A3d

VIU_OSD2_BLK1_CFG_W2 0x1A41

VIU_OSD2_BLK2_CFG_W2 0x1A45

VIU_OSD2_BLK3_CFG_W2 0x1A49

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.

VIU_OSD2_BLK0_CFG_W3 0x1A3e

VIU_OSD2_BLK1_CFG_W3 0x1A42

VIU_OSD2_BLK2_CFG_W3 0x1A46

VIU_OSD2_BLK3_CFG_W3 0x1A4a

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.

VIU_OSD2_BLK0_CFG_W4 0x1A64

VIU_OSD2_BLK1_CFG_W4 0x1A65

VIU_OSD2_BLK2_CFG_W4 0x1A66

VIU_OSD2_BLK3_CFG_W4 0x1A67

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.

VIU_OSD2_FIFO_CTRL_STAT 0x1A4b

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 2words 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.
17-12	R/W	32	FIFO_DEPTH_VAL: Define the depth of FIFO which stores 64-bit data from DDR to be FIFO_DEPTH_VAL * 8.
11-10	R/W	0	BURST_LEN_SEL: 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.
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.
2-1	R/W	0	ENDIAN: define the endianess of the 64-bit data stored in memory, and how to convert. 0 = No convertion; 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.

VIU_OSD2_MATRIX_CTRL 0x1AB0

Bit(s)	R/W	Default	Description
9-8	R/W	0	Matrix coef index selection, 00: select post matrix, 01: select video1 matrix, 10: select video2 matrix
5	R/W	0	Video1 conversion matrix enable
4	R/W	0	Video2 conversion matrix enable
2	R/W	0	Output y/cb/cr saturation enable, only for post matrix(y saturate to 16-235, cb/cr saturate to 0-240)
1	R/W	0	input y/cb/cr saturation enable, only for post matrix(y saturate 16-235, cb/cr saturate to 16-240)
0	R/W	0	conversion matrix enable

VIU_OSD2_MATRIX_COEF00_01 0x1AB1

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient00, signed, 3.10
12-0	R/W	0	Coefficient01, signed, 3.10

VIU_OSD2_MATRIX_COEF02_10 0x1AB2

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient02, signed, 3.10
12-0	R/W	0	Coefficient10, signed, 3.10

VIU_OSD2_MATRIX_COEF11_12 0x1AB3

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient11, signed, 3.10
12-0	R/W	0	Coefficient12, signed, 3.10

VIU_OSD2_MATRIX_COEF20_21 0x1AB4

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coefficient20, signed, 3.10
12-0	R/W	0	Coefficient21, signed, 3.10

VIU_OSD2_MATRIX_COEF22 0x1AB5

Bit(s)	R/W	Default	Description
18-16	R/W	0	convrs
12-0	R/W	0	Coefficient22, signed, 3.10

VIU_OSD2_MATRIX_OFFSET0_1 0x1AB6

Bit(s)	R/W	Default	Description
26-16	R/W	0	Offset0, signed value
10-0	R/W	0	Offset1, signed value

VIU_OSD2_MATRIX_OFFSET2 0x1AB7

Bit(s)	R/W	Default	Description
10-0	R/W	0	Offset2, signed value

VIU_OSD2_MATRIX_PRE_OFFSET0_1 0x1AB8

Bit(s)	R/W	Default	Description
27-16	R/W	0	pre_Offset0, signed value
11-0	R/W	0	Pre_Offset1, signed value

VIU_OSD2_MATRIX_PRE_OFFSET2 0x1AB9

Bit(s)	R/W	Default	Description
11-0	R/W	0	Pre_Offset2, signed value

VDIN0_SCALE_COEF_IDX 0x1200

VDIN0_SCALE_COEF 0x1201

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
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: digtial video input; 7: internal loop back from Viu; 8: Reserved(mipi_csi2) ;9: Reserved(isp); 10: second bt656 input; otherwise no input.

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

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

VDIN0_COM_STATUS0 0x1205

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

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
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

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

VDIN0_ASFIFO_CTRL0 0x1208

Bit(s)	R/W	Default	Description
23	R/W	0	vdi2 DE enable
22	R/W	0	vdi2 go field enable
21	R/W	0	vdi2 go line enable
20	R/W	0	vdi2 if true, negative active input vsync
19	R/W	0	vdi2 if true, negative active input hsync
18	R/W	0	vdi2 vsync soft reset fifo enable
17	R/W	0	vdi2 overflow status clear
16	R/W	0	vdi2 asfifo soft reset, level signal
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

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
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

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

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

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

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

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

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

VDIN0_MATRIX_CTRL 0x1210

Bit(s)	R/W	Default	Description
3-2	R/W	0	matrix coef idx selection, 00: select mat0, 01: select mat1, otherwise slect nothing
1	R/W	0	mat1 conversion matrix enable
0	R/W	0	Mat0 conversion matrix enable

VDIN0_MATRIX_COEF00_01 0x1211

Bit(s)	R/W	Default	Description
28-16	R/W	0	coef00
12-0	R/W	0	coef01

VDIN0_MATRIX_COEF02_10 0x1212

Bit(s)	R/W	Default	Description
28-16	R/W	0	coef02
12-0	R/W	0	Coef10

VDIN0_MATRIX_COEF11_12 0x1213

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef11
12-0	R/W	0	Coef12

VDIN0_MATRIX_COEF20_21 0x1214

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef20
12-0	R/W	0	coef21

VDIN0_MATRIX_COEF22 0x1215

Bit(s)	R/W	Default	Description
18-16	R/W	0	convrs
7-0	R/W	0	Coef22

VDIN0_MATRIX_OFFSET0_1 0x1216

Bit(s)	R/W	Default	Description
26-16	R/W	0	offset0
10-0	R/W	0	Offset1

VDIN0_MATRIX_OFFSET2 0x1217

Bit(s)	R/W	Default	Description
10-0	R/W	0	Offset2

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

VDIN0_MATRIX_PRE_OFFSET2 0x1219

Bit(s)	R/W	Default	Description
10-0	R/W	0	Pre_Offset2

VDIN0_LFIFO_CTRL 0x121a

Bit(s)	R/W	Default	Description
11-0	R/W	0	lfifo_buf_size

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

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

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 : 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
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.

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&odd pair of pixels; 3=Output all CbCr. Only applys 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 enble, 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: Reserved.
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)

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

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

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

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

VDIN0_SCIN_HEIGHTM1 0x1225

Bit(s)	R/W	Default	Description
12-0	R/W	0x437	scaler input height minus 1

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

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: form 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_DNLP_HISTXX should multiple with 2^(dnlp_low+3)
3-2	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
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

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

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

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

VDIN0_HIST_SPL_VAL 0x1234

Bit(s)	R/W	Default	Description
31-0	R	0	hist_spl_rd , counts for the total luma value

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

VDIN0_HIST_CHROMA_SUM 0x1236

Bit(s)	R/W	Default	Description
31-0	R	0	hist_chroma_sum , counts for the total chroma value

VDIN0_DNLP_HIST00 0x1237

//0-255 are splited 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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_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

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

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

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

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

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

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

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

VDIN0_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

VDIN0_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

VDIN0_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

VDIN0_LDIM_STTS_HIST_REGION_IDX 0x1257

Bit(s)	R/W	Default	Description
31	R	0	local dimming 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

VDIN0_LDIM_STTS_HIST_SET_REGION 0x1258

Bit(s)	R/W	Default	Description

28:16	R	0	if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h0: read/write hvstart0
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h1: read/write hend01
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h2: read/write vend01
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h3: read/write hend23
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h4: read/write vend23
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h5: read/write hend45
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h6: read/write vend45
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'd7: read/write hend67
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h8: read/write vend67
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h9: read/write hend89
			if VDIN_LDIM_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	

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

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
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.
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

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

VDIN0_MEAS_VS_COUNT_LO 0x125c

Bit(s)	R/W	Default	Description
31-0	R	0	low bit portion of vsync total counter

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

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

VDIN0_BLKBAR_CTRL1 0x125f

Bit(s)	R/W	Default	Description
8	R/W	0	white_enable
7-0	R/W	0	blkbar_white_level

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

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

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.

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

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

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

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

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

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

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

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

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

VDIN0_BLKBAR_STATUS1 0x126c

Bit(s)	R/W	Default	Description
28-16	R	0	blkbar_left_pos. Left black bar posiont
12-0	R	0	blkbar_right_pos. Right black bar position

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

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

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
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

VDIN1_SCALE_COEF_IDX 0x1270

VDIN1_SCALE_COEF 0x1271

VDIN1_COM_CTRL0 0x1272

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: digtial video input; 7: internal loop back from Viu; 8: Reserved(mipi_csi2) ;9: Reserved(isp); 10: second bt656 input; otherwise no input.

VDIN1_ACTIVE_MAX_PIX_CNT_STATUS 0x1273

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

VDIN1_LCNT_STATUS 0x1274

Bit(s)	R/W	Default	Description
28-16	R	0	go_line_cnt, readonly
12-0	R	0	active_line_cnt, readonly

VDIN1_COM_STATUS0 0x1275

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

VDIN1_COM_STATUS1 0x1276

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

VDIN1_LCNT_SHADOW_STATUS 0x1277

Bit(s)	R/W	Default	Description
28-16	R	0	go_line_cnt_shadow, readonly
12-0	R	0	active_line_cnt_shadow, readonly

VDIN1_ASFIFO_CTRL0 0x1278

Bit(s)	R/W	Default	Description
23	R/W	0	vdi2 DE enable
22	R/W	0	vdi2 go field enable
21	R/W	0	vdi2 go line enable
20	R/W	0	vdi2 if true, negative active input vsync
19	R/W	0	vdi2 if true, negative active input hsync
18	R/W	0	vdi2 vsync soft reset fifo enable
17	R/W	0	vdi2 overflow status clear
16	R/W		vdi2 asfifo soft reset, level signal
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

VDIN1_ASFIFO_CTRL1 0x1279

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

VDIN1_WIDTHM1I_WIDTHM1O 0x127a

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

VDIN1_SC_MISC_CTRL 0x127b

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
3	R/W	0	phase0_always_en
2-0	R/W	0	hsc_bank_length

VDIN1_HSC_PHASE_STEP 0x127c

Bit(s)	R/W	Default	Description
28-24	R/W	0	integer portion
23-0	R/W	0	fraction portion

VDIN1_HSC_INI_CTRL 0x127d

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

VDIN1_COM_STATUS2 0x127e

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

VDIN1_ASFIFO_CTRL2 0x127f

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

VDIN1_MATRIX_CTRL 0x1280

Bit(s)	R/W	Default	Description
0	R/W	0	post conversion matrix enable

VDIN1_MATRIX_COEF00_01 0x1281

Bit(s)	R/W	Default	Description
28-16	R/W	0	coef00
12-0	R/W	0	coef01

VDIN1_MATRIX_COEF02_10 0x1282

Bit(s)	R/W	Default	Description
28-16	R/W	0	coef02
12-0	R/W	0	Coef10

VDIN1_MATRIX_COEF11_12 0x1283

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef11
12-0	R/W	0	Coef12

VDIN1_MATRIX_COEF20_21 0x1284

Bit(s)	R/W	Default	Description
28-16	R/W	0	Coef20
12-0	R/W	0	coef21

VDIN1_ 0x1285

Bit(s)	R/W	Default	Description
18-16	R/W	0	convrs
7-0	R/W	0	Coef22

VDIN1_MATRIX_OFFSET0_1 0x1286

Bit(s)	R/W	Default	Description
26-16	R/W	0	offset0
10-0	R/W	0	Offset1

VDIN1_MATRIX_OFFSET2 0x1287

Bit(s)	R/W	Default	Description
10-0	R/W	0	Offset2

VDIN1_MATRIX_PRE_OFFSET0_1 0x1288

Bit(s)	R/W	Default	Description
26-16	R/W	0	Pre_offset0
10-0	R/W	0	Pre_Offset1

VDIN1_MATRIX_PRE_OFFSET2 0x1289

Bit(s)	R/W	Default	Description
10-0	R/W	0	Pre_Offset2

VDIN1_LFIFO_CTRL 0x128a

Bit(s)	R/W	Default	Description
11-0	R/W	0	lfifo_buf_size

VDIN1_COM_GCLK_CTRL 0x128b

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

VDIN1_INTF_WIDTHM1 0x128c

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

VDIN1_WR_CTRL2 0x128f

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 : 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
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.

VDIN1_WR_CTRL 0x1290

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&odd pair of pixels; 3=Output all CbCr. Only applys 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 enble, 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. 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: Reserved.
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)

VDIN1_WR_H_START_END 0x1291

Bit(s)	R/W	Default	Description
27-16	R/W	0	start
11-0	R/W	0	end

VDIN1_WR_V_START_END 0x1292

Bit(s)	R/W	Default	Description
27-16	R/W	0	start
11-0	R/W	0	end

VDIN1_VSC_PHASE_STEP 0x1293

Bit(s)	R/W	Default	Description
24-16	R/W	0	integer portion
19-0	R/W	0	fraction portion

VDIN1_VSC_INI_CTRL 0x1294

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

VDIN1_SCIN_HEIGHTM1 0x1295

Bit(s)	R/W	Default	Description
12-0	R/W	0x437	scaler input height minus 1

VDIN1_DUMMY_DATA 0x1296

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

VDIN1_HIST_CTRL 0x12a0

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
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)
3-2	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
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

VDIN1_HIST_H_START_END 0x12a1

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

VDIN1_HIST_V_START_END 0x12a2

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

VDIN1_HIST_MAX_MIN 0x12a3

Bit(s)	R/W	Default	Description
15-8	R	0	hist_max maximum value
7-0	R	0	hist_min minimum value

VDIN1_HIST_SPL_VAL 0x12a4

Bit(s)	R/W	Default	Description
31-0	R	0	hist_spl_rd , counts for the total luma value

VDIN1_HIST_SPL_PIX_CNT 0x12a5

Bit(s)	R/W	Default	Description
21-0	R	0	hist_spl_pixel_count, counts for the total calculated pixels

VDIN1_HIST_CHROMA_SUM 0x12a6

Bit(s)	R/W	Default	Description
31-0	R	0	hist_chroma_sum , counts for the total chroma value

VDIN1_DNLP_HIST00 0x12a7

//0-255 are splited 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

VDIN1_DNLP_HIST01 0x12a8

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 4th bin
15-0	R	0	counts for the 3rd bin

VDIN1_DNLP_HIST02 0x12a9

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 6th bin
15-0	R	0	counts for the 5th bin

VDIN1_DNLP_HIST03 0x12aa

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 8th bin
15-0	R	0	counts for the 7th bin

VDIN1_DNLP_HIST04 0x12ab

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 10th bin
15-0	R	0	counts for the 9th bin

VDIN1_DNLP_HIST05 0x12ac

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 12th bin
15-0	R	0	counts for the 11th bin

VDIN1_DNLP_HIST06 0x12ad

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 14th bin
15-0	R	0	counts for the 13th bin

VDIN1_DNLP_HIST07 0x12ae

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 16th bin
15-0	R	0	counts for the 15th bin

VDIN1_DNLP_HIST08 0x12af

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 18th bin
15-0	R	0	counts for the 17th bin

VDIN1_DNLP_HIST09 0x12b0

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 20th bin
15-0	R	0	counts for the 19th bin

VDIN1_DNLP_HIST10 0x12b1

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 22nd bin
15-0	R	0	counts for the 21st bin

VDIN1_DNLP_HIST11 0x12b2

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 24th bin
15-0	R	0	counts for the 23rd bin

VDIN1_DNLP_HIST12 0x12b3

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 26th bin
15-0	R	0	counts for the 25th bin

VDIN1_DNLP_HIST13 0x12b4

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 28th bin
15-0	R	0	counts for the 27th bin

VDIN1_DNLP_HIST14 0x12b5

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 30th bin
15-0	R	0	counts for the 29th bin

VDIN1_DNLP_HIST15 0x12b6

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 32nd bin
15-0	R	0	counts for the 31st bin

VDIN1_DNLP_HIST16 0x12b7

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 34th bin
15-0	R	0	counts for the 33rd bin

VDIN1_DNLP_HIST17 0x12b8

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 36th bin
15-0	R	0	counts for the 35th bin

VDIN1_DNLP_HIST18 0x12b9

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 38th bin
15-0	R	0	counts for the 37th bin

VDIN1_DNLP_HIST19 0x12ba

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 40th bin
15-0	R	0	counts for the 39th bin

VDIN1_DNLP_HIST20 0x12bb

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 42nd bin
15-0	R	0	counts for the 41st bin

VDIN1_DNLP_HIST21 0x12bc

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 44th bin
15-0	R	0	counts for the 43rd bin

VDIN1_DNLP_HIST22 0x12bd

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 46th bin
15-0	R	0	counts for the 45th bin

VDIN1_DNLP_HIST23 0x12be

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 48th bin
15-0	R	0	counts for the 47th bin

VDIN1_DNLP_HIST24 0x12bf

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 50th bin
15-0	R	0	counts for the 49th bin

VDIN1_DNLP_HIST25 0x12c0

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 52nd bin
15-0	R	0	counts for the 51st bin

VDIN1_DNLP_HIST26 0x12c1

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 54th bin
15-0	R	0	counts for the 53rd bin

VDIN1_DNLP_HIST27 0x12c2

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 56th bin
15-0	R	0	counts for the 55th bin

VDIN1_DNLP_HIST28 0x12c3

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 58th bin
15-0	R	0	counts for the 57th bin

VDIN1_DNLP_HIST29 0x12c4

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 60th bin
15-0	R	0	counts for the 59th bin

VDIN1_DNLP_HIST30 0x12c5

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 62nd bin
15-0	R	0	counts for the 61st bin

VDIN1_DNLP_HIST31 0x12c6

Bit(s)	R/W	Default	Description
31-16	R	0	counts for the 64th bin
15-0	R	0	counts for the 63rd bin

VDIN1_LDIM_STTS_HIST_REGION_IDX 0x12c7

Bit(s)	R/W	Default	Description
31	R	0	local dimming 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

VDIN1_LDIM_STTS_HIST_SET_REGION 0x12c8

Bit(s)	R/W	Default	Description

28:16	R	0	if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h0: read/write hvstart0
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h1: read/write hend01
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h2: read/write vend01
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h3: read/write hend23
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h4: read/write vend23
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h5: read/write hend45
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h6: read/write vend45
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'd7: read/write hend67
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h8: read/write vend67
			if VDIN_LDIM_STTS_HIST_REGION_IDX[19:16] == 5'h9: read/write hend89
			if VDIN_LDIM_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	

VDIN1_LDIM_STTS_HIST_READ_REGION 0x12c9

Bit(s)	R/W	Default	Description
29:20	R	0	Max_comp2
19:10	R	0	Max_comp1
9:0	R	0	Max_comp0

VDIN1_MEAS_CTRL0 0x12ca

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.
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

VDIN1_MEAS_VS_COUNT_HI 0x12cb

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

VDIN1_MEAS_VS_COUNT_LO 0x12cc

Bit(s)	R/W	Default	Description
31-0	R	0	low bit portion of vsync total counter

VDIN1_MEAS_HS_RANGE 0x12cd

//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

VDIN1_MEAS_HS_COUNT 0x12ce

//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

VDIN1_BLKBAR_CTRL1 0x12cf

Bit(s)	R/W	Default	Description
8	R/W	0	white_enable
7-0	R/W	0	blkbar_white_level

VDIN1_BLKBAR_CTRL0 0x12d0

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

VDIN1_BLKBAR_H_START_END 0x12d1

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

VDIN1_BLKBAR_V_START_END 0x12d2

Bit(s)	R/W	Default	Description
28-16	R/W	0	blkbar_vstart.
12-0	R/W	0	blkbar_vend.

VDIN1_BLKBAR_CNT_THRESHOLD 0x12d3

Bit(s)	R/W	Default	Description
19-0	R/W	0	blkbar_cnt_threshold. threshold to judge whether a block is totally black

VDIN1_BLKBAR_ROW_TH1_TH2 0x12d4

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

VDIN1_BLKBAR_IND_LEFT_START_END 0x12d5

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

VDIN1_BLKBAR_IND_RIGHT_START_END 0x12d6

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

VDIN1_BLKBAR_IND_LEFT1_CNT 0x12d7

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_left1_cnt. Black pixel counter. left part of the left region

VDIN1_BLKBAR_IND_LEFT2_CNT 0x12d8

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_left2_cnt. Black pixel counter. right part of the left region

VDIN1_BLKBAR_IND_RIGHT1_CNT 0x12d9

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_right1_cnt. Black pixel counter. left part of the right region

VDIN1_BLKBAR_IND_RIGHT2_CNT 0x12da

Bit(s)	R/W	Default	Description
19-0	R	0	blkbar_ind_right2_cnt. Black pixel counter. right part of the right region

VDIN1_BLKBAR_STATUS0 0x12db

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

VDIN1_BLKBAR_STATUS1 0x12dc

Bit(s)	R/W	Default	Description
28-16	R	0	blkbar_left_pos. Left black bar posiont
12-0	R	0	blkbar_right_pos. Right black bar position

VDIN1_WIN_H_START_END 0x12dd

Bit(s)	R/W	Default	Description
28-16	R/W	0	input window H start
12-0	R/W	0	input window H end

VDIN1_WIN_V_START_END 0x12de

Bit(s)	R/W	Default	Description
28-16	R/W	0	input window V start
12-0	R/W	0	input window V end

VDIN1_ASFIFO_CTRL3 0x12df

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
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

VDIN1_COM_GCLK_CTRL2 0x12e0

Bit(s)	R/W	Default	Description
3-2	R/W	0	Vshrk_clk2 ctrl
1-0	R/W	0	Vshrk_clk1 ctrl

VDIN1_VSHRK_CTRL 0x12e1

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

VDIN1_HIST32 0x12e2

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

VDIN1_COM_STATUS3 0x12e3

Bit(s)	R/W	Default	Description
7	R	0	Vdi9 fifo overflow
5:0	R	0	Vdi9 asfifo cnt

VI_HIST_CTRL 0x2e00

Bit(s)	R/W	Default	Description
17-16	R/W	0	Spl_sft: the splt 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
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_hist_spl_en, 1: enable the histogram statistic

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

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

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

VI_HIST_SPL_VAL 0x2e04

Bit(s)	R/W	Default	Description
31-0	R	0	hist_spl_rd , counts for the total luma value

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

VI_HIST_CHROMA_SUM 0x2e06

Bit(s)	R/W	Default	Description
31-0	R	0	hist_chroma_sum , counts for the total chroma value

VI_DNLP_HIST00 0x2e07

//0-255 are splited 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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

MCDI registers

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

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_blkhsize 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

MCDI_BLKTOTAL 0x2f02

Bit(s)	R/W	Default	Description
31-24	R/W		reserved

Bit(s)	R/W	Default	Description
23-0	R/W	0	reg_mcdi_blktotal

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

MCDI_CTRL_MODE 0x2f04

Bit(s)	R/W	Default	Description
31-28	R/W		reserved
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
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, defautl = 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 neighoring mvs in refinement, default = 1
11	R/W		reserved
10	R/W	1	reg_mcdi_referrfrqchken 0-unable; 1: enable enable mv frquency 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_lmven0-unable; 1: enable enable line mv estimation for hme, default = 1
2	R/W	1	reg_mcdi_chkedgeen0-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

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

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

Bit(s)	R/W	Default	Description
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

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

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 (>=) 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

MCDI_FLT_MODESEL 0x2f09

Bit(s)	R/W	Default	Description
31	R/W		reserved
30-28	R/W	1	reg_mcdi_flthorlineselmode 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_fltsadselmode 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_fltpd22chkselmode 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)

MCDI_CHK_EDGE_THD 0x2f0a

Bit(s)	R/W	Default	Description
23-28	R/W		reserved.
27-24	R/W	1	reg_mcdi_chkedgedifsadthd. thd (<=) for sad dif check, 0~8, default = 1
23-16	R/W		reserved.
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

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: orignal 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, defautl = 1
1	R/W		reg_mcdi_chkedgestartedge. edge mode for start edge, 0- original edge, 1: lpf edge, defautl = 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

MCDI_LMV_RT 0x2f0c

Bit(s)	R/W	Default	Description
31-15	R/W		reserved
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 lmvs at first row of input field, 0- intial value = 0; 1: inital = 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

MCDI_LMV_GAINTHD 0x2f0d

Bit(s)	R/W	Default	Description
31-24	R/W	96	reg_mcdi_lmvvxmaxgain 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_lmvnumlmt least/limit number of (total number - max0), default = 20
7-0	R/W	9	reg_mcdi_lmvfltthd flt cnt thd (<) for lmv, default = 9

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

MCDI_RPTMV_THD1 0x2f0f

Bit(s)	R/W	Default	Description
31-28	R/W		reserved
27-25	R/W	2	reg_mcdi_rptmvcyccntthd 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_rptmvminmaxcntthd 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

MCDI_RPTMV_THD2 0x2f10

Bit(s)	R/W	Default	Description
31-24	R/W		reserved

Bit(s)	R/W	Default	Description
23-16	R/W	8	reg_mcdi_rptmvhdifthd0 higher hdif thd (>=) (vertical edge) for rpt detection, default = 8
15-8	R/W	4	reg_mcdi_rptmvhdifthd1 hdif thd (>=) (slope edge) for rpt detection, default = 4
7-0	R/W	1	reg_mcdi_rptmvvdifthd vdif thd (>=) (slope edge) for rpt detection, default = 1

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

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 (>=) 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,

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

MCDI_GMV_RT 0x2f14

Bit(s)	R/W	Default	Description
31	R/W		reserved
30-24	R/W	32	reg_mcdi_gmvmtnrt0 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_gmvmtnrt1 ratio 1 for motion senario, set 127 to 128, normalized 128 as '1', default = 56
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

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_gmvfltthd flat thd (<) for gmv calc. default = 3

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 (>=) of vertical dif of next block for horizontal line detection, default = 24

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

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
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

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

MCDI_REF_BS_THD_GAIN 0x2f1a

Bit(s)	R/W	Default	Description
31-28	R/W	2	reg_mcdi_refbsudgain1. up & down block stregth gain1, normalized to 8 as '1', default = 2
27-24	R/W	4	reg_mcdi_refbsudgain0. up & down block stregth 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 stregth in refinment, default = 16
7-5	R/W		reserved
4-0	R/W	8	reg_mcdi_refbsthd0. threshold 0 for detect block stregth in refinment, default = 8

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
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

MCDI_REF_ERR_GAIN1 0x2f1c

Bit(s)	R/W	Default	Description
31-20	R/W		reserved
19-16	R/W	4	reg_mcdi_referrchkedgegain. 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) gmv gain for err calc. in ref, normalized to 8 as '1', default = 0

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 frquency, normalized to 4 as '1', default = 10
23-21	R/W		reserved
20-16	R/W	31	reg_mcdi_referrfrqmax. max gain for mv frquency check, default = 31
15	R/W		reserved
14-12	R/W	3	reg_mcdi_ref_errfrqmvdifthd2. mv dif threshold 2 (<) for mv frquency check, default = 3
11	R/W		reserved
10-8	R/W	2	reg_mcdi_ref_errfrqmvdifthd1. mv dif threshold 1 (<) for mv frquency check, default = 2
7	R/W		reserved
6-4	R/W	1	reg_mcdi_ref_errfrqmvdifthd0. mv dif threshold 0 (<) for mv frquency check, default = 1
3-0	R/W		reserved

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

MCDI_REL_DIF_THD_02 0x2f1f

Bit(s)	R/W	Default	Description
31-24	R/W		reserved.
23-16	R/W	9	reg_mcdi_reldifthd2. thd (<) for (hdif+vdif), default = 9
15-8	R/W	5	reg_mcdi_reldifthd1. thd (<) for (vdif), default = 5
7-0	R/W	48	reg_mcdi_reldifthd0. thd (>=) for (hdif-vdif), default = 48

MCDI_REL_DIF_THD_34 0x2f20

Bit(s)	R/W	Default	Description
31-16	R/W		reserved.
15-8	R/W	255	reg_mcdi_reldifthd4. thd (<) for (hdif), default = 255
7-0	R/W	48	reg_mcdi_reldifthd3. thd (>=) for (vdif-hdif), default = 48

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

MCDI_REL_BADW_GAIN_OFFST_23 0x2f22

Bit(s)	R/W	Default	Description
31-24	R/W	0	reg_mcdi_relbadwoffst3. offset for badw adj, for other block, -128~127, default = 0
23-16	R/W	48	reg_mcdi_relbadwgain3. gain for badw adj, for other block, default = 48
15-8	R/W	0	reg_mcdi_relbadwoffst2. offset for badw adj, for horizontal block, -128~127, default = 0
7-0	R/W	48	reg_mcdi_relbadwgain2. gain for badw adj, for horizontal block, default = 48

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_relbadwoffst. offset for badw thd adj, -64~63, default = 0

Bit(s)	R/W	Default	Description
15-8	R/W		reserved.
7-0	R/W	16	reg_mcdi_relbadwthdgain. gain0 for badw thd adj, normalized to 16 as '1', default = 16

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_relbadwthdmax. max for badw thd adj, default = 256
7-0	R/W	16	reg_mcdi_relbadwthdmin. min for badw thd adj, default = 16

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

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.
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

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*2/16, normalized to 16 as '1', default = 42

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*32, default = 672
15-9	R/W		reserved.
8-0	R/W	42	reg_mcdi_relsadthdmin. min for sad thd adj, 21*2, default = 42

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

MCDI_REL_DET_GAIN_01 0x2f2a

Bit(s)	R/W	Default	Description
31-14	R/W		reserved.
12-8	R/W	2	reg_mcdi_reldetchkedgegain0. gain0 (gmv locked) for chk_edge, for det. calc. normalized to 16 as '1', default = 2
7	R/W		reserved.

Bit(s)	R/W	Default	Description
6-0	R/W	24	reg_mcdi_reldetnbrdstgain0. gain0 (gmv locked) for neighoring dist, for det. calc. normalized to 16 as '1', default = 24

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

MCDI_REL_DET_GAIN_11 0x2f2c

Bit(s)	R/W	Default	Description
31-14	R/W		reserved.
12-8	R/W	0	reg_mcdi_reldetchkedgegain1. 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

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

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

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_reldetgmvfltthd. flat thd (>=) for gmv lock decision, default = 0
15	R/W		reserved.
14-12	R/W	3	reg_mcdi_reldetgmvdifthd. dif thd (>=) 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

MCDI_REL_DET_LMV_DIF_CHK 0x2f30

Bit(s)	R/W	Default	Description
31-24	R/W		reserved.

Bit(s)	R/W	Default	Description
23-16	R/W	12	reg_mcdi_reldetlmvfltthd. flat thd (>=) 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
3-1	R/W		reserved
0	R/W	0	reg_mcdi_reldetlmvdifmvmode. mv mode used for lmv dif check, 0- use refmv, 1: use qmv, default = 0

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

MCDI_REL_DET_PD22_CHK 0x2f32

Bit(s)	R/W	Default	Description
31-18	R/W		reserved.
17-8	R/W	512	reg_mcdi_reldetpd22chkoffst. offset for pd22 check happened, default = 512
7-5	R/W		reserved
4-0	R/W	12	reg_mcdi_reldetpd22chkgain. gain for pd22 check happened, normalized to 8 as '1', default = 12

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

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
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

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

MCDI_REL_DET_RPT_CHK_THD_1 0x2f36

Bit(s)	R/W	Default	Description
31-16	R/W		reserved

Bit(s)	R/W	Default	Description
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

MCDI_REL_DET_LPF_DIF_THD 0x2f37

Bit(s)	R/W	Default	Description
31-24	R/W	9	reg_mcdi_reldetlpfdifthd3. hdif thd (<) for lpf selection of horizontal block, default = 9
23-16	R/W	48	reg_mcdi_reldetlpfdifthd2. vdif-hdif thd (>=) for lpf selection of horizontal block, default = 48
15-8	R/W	9	reg_mcdi_reldetlpfdifthd1. vdif thd (<) for lpf selection of vertical block, default = 9
7-0	R/W	48	reg_mcdi_reldetlpfdifthd0. hdif-vdif thd (>=) for lpf selection of vertical block, default = 48

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
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

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

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

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
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

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

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

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

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
15-8	R/W	32	reg_mcdi_reldetmaplut5. default = 32
7-0	R/W	16	reg_mcdi_reldetmaplut4. default = 16

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

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

MCDI_REL_DET_COL_CFD_THD 0x2f42

Bit(s)	R/W	Default	Description
31-24	R/W	5	reg_mcdi_reldetcolcfdfltthd. 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_reldetcolcfdcalcmode. calc. mode for column cofidence, 0- use rel, 1: use qbadw, default = 0

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
15-8	R/W	16	reg_mcdi_reldetcolcfdavgmin0. avg luma min0 (>=) for column cofidence, valid between 16~235, default = 16
7-0	R/W	21	reg_mcdi_reldetcolcfdavgmax0. avg luma max0 (<) for column cofidence, valid between 16~235, default = 21

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 (>=) for bad sad, default = 120 (480/4)
7-6	R/W		reserved
5-0	R/W	12	reg_mcdi_reldetbadbadwthd. thd (>=) for bad badw, 0~42, default = 12

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_reldetbadrelflatthd. thd (>=) of flat for bad rel detection, default = 4
15-8	R/W	160	reg_mcdi_reldetbadrelthd1. thd (>=) for bad rel without gmv/lmv locked, default = 160
7-0	R/W	120	reg_mcdi_reldetbadrelthd0. thd (>=) for bad rel with gmv/lmv locked, default = 120

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 (>=) for (pd22chksad - qsad) for pd22 check, default = 64
15-14	R/W		reserved
13-8	R/W	2	reg_mcdi_pd22chkqmvthd. thd (>=) of abs qmv for pd22 check, default = 2
7-0	R/W	4	reg_mcdi_pd22chkfltthd. thd (>=) of flat for pd22 check, default = 4

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
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

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

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_lmvlockcntmax. max lmv lock count number, default = 6
15-12	R/W	0	reg_mcdi_lmvlockcntoffst. offset for lmv lock count, -8~7, default = 0

Bit(s)	R/W	Default	Description
11-8	R/W	8	reg_mcdi_lmvlockcntgain. 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_lmvlockcntthd. lmv count thd (>=) before be locked, 1~31, default = 4

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_lmvlockdifthd2. lmv dif thd for third part, before locked, default = 1
23	R/W		reserved
22-20	R/W	1	reg_mcdi_lmvlockdifthd1. lmv dif thd for second part, before locked, default = 1
19	R/W		reserved
18-16	R/W	1	reg_mcdi_lmvlockdifthd0. lmv dif thd for first part, before locked, default = 1
15-13	R/W		reserved
12-8	R/W	24	reg_mcdi_lmvlockabsmax. max abs (<) of lmv to be locked, default = 24
7-5	R/W		reserved
4-0	R/W	1	reg_mcdi_lmvlockabsmin. min abs (>=) of lmv to be locked, default = 1

MCDI_LMV_LOCK_ROW 0x2f4b

Bit(s)	R/W	Default	Description
31-27	R/W		reserved
26-16	R/W	2047	reg_mcdi_lmvlockendrow. end row (<) for lmv lock, default = 2047
15-11	R/W		reserved
10-0	R/W	0	reg_mcdi_lmvlockstartrow. start row (>=) for lmv lock, default = 0

MCDI_LMV_LOCK_RT_MODE 0x2f4c

Bit(s)	R/W	Default	Description
31-27	R/W		reserved
26-24	R/W	2	reg_mcdi_lmvlockextmode. extend lines for lmv lock check, check how many lines for lmv locking, default = 2
23-16	R/W	32	reg_mcdi_lmvlockfltcntrt. 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_lmvlocklmvcntrt1. 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_lmvlocklmvcntrt0. ratio when use max lmv for lock check, normalized 256 as '1', 255 is set to 256, default = 106

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

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

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*Blk_Width*Blk_Height, set by software, default = 103680

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*Blk_Width*Blk_Height, set by software, default = 103680

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

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
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

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 reliabilty count of motion paradox, normalized 256 as '1', 255 is set to 256, default = 120
7-0	R/W	218	reg_mcdi_motionparadoxmtnrt. ratio for field motion count of motion paradox, normalized 256 as '1', 255 is set to 256, default = 218

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_motionreflpfmode. 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

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

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_pd22chkcntrt. ratio of pulldown 22 check count, normalized 256 as '1', 255 is set to 256, default = 100
7-5	R/W		reserved
4-0	R/W	4	reg_mcdi_pd22chkcntthd. thd (>=) for pd22 count before locked, 1~31, default = 4

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

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

MCDI_FIELD_MV 0x2f60

Bit(s)	R/W	Default	Description
31-24	R/W		reg_mcdi_pd22chkcnt
23-16	R/W		reg_mcdi_fieldgmvcnt
15	R/W		reg_mcdi_pd22chkflg
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

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.

MCDI_FIELD_LUMA_AVG_SUM_0 0x2f62

Bit(s)	R/W	Default	Description
31-0	R/W		reg_mcdi_fld_luma_avg_sum0.

MCDI_FIELD_LUMA_AVG_SUM_1 0x2f63

Bit(s)	R/W	Default	Description
31-0	R/W		reg_mcdi_fld_luma_avg_sum1.

MCDI_YCBCR_BLEND_CRTL 0x2f64

Bit(s)	R/W	Default	Description
31-16	R/W		reserved

Bit(s)	R/W	Default	Description
15-8	R/W	0	reg_mcdi_ycbcrblendgain. ycbcr blending gain for cbcr 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

MCDI_MC_CRTL 0x2f70

Bit(s)	R/W	Default	Description
31-15	R/W		reserved
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
9	R/W	0	reg_mcdi_mc_uv_en: mc for uv if needed, else use ma of uv
8	R/W	1	reg_mcdi_mcpreflg. flag to use previous field for MC, 0-forward field, 1: previous field, 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

MCDI_MC_LPF_MSK_0 0x2f71

Bit(s)	R/W	Default	Description
31-21	R/W		reserved
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

MCDI_MC_LPF_MSK_1 0x2f72

Bit(s)	R/W	Default	Description
31-21	R/W		reserved
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

MCDI_MC_LPF_MSK_2 0x2f73

Bit(s)	R/W	Default	Description
31-21	R/W		reserved
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
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

MCDI_MC_LPF_MSK_3 0x2f74

Bit(s)	R/W	Default	Description
31-21	R/W		reserved
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

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

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
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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

MCDI_MC_COL_CFD_20 0x2f8c

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 20 read from software. initial = 0

MCDI_MC_COL_CFD_21 0x2f8d

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 21 read from software. initial = 0

MCDI_MC_COL_CFD_22 0x2f8e

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 22 read from software. initial = 0

MCDI_MC_COL_CFD_23 0x2f8f

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 23 read from software. initial = 0

MCDI_MC_COL_CFD_24 0x2f90

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 24 read from software. initial = 0

MCDI_MC_COL_CFD_25 0x2f91

Bit(s)	R/W	Default	Description
31-0	R/W		mcdi_mc_col_cfd_4. column cofidence value 25 read from software. initial = 0

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

MCDI_RO_GMV_VLD_CNT 0x2fa1

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_gmvvldcnt. valid gmv's count of pre one filed (block based). initial = 0

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

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

MCDI_RO_FLD_BAD_BADW_CNT 0x2fa4

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldbadbadwcnt. bad badw count of whole pre one field (block based). initial = 0

MCDI_RO_FLD_BAD_REL_CNT 0x2fa5

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldbadrelcnt. bad rel count of whole pre one field (block based). initial = 0

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

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

MCDI_RO_FLD_PD_22_PRE_CNT 0x2fa8

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_fldpd22precnt. prevoius pd22 check count of whole pre one field (block based). initial = 0

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

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

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

MCDI_RO_GMV_LOCK_FLG 0x2fac

Bit(s)	R/W	Default	Description
31-16	R		reserved.
15-8	R		ro_mcdi_gmvlckcnt. 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_zerogmvlckflg. zero global mv lock flag of prevoius one field. initial = 0
0	R		ro_mcdi_gmvlckflg. global mv lock flag of prevoius one field. initial = 0

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)

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

MCDI_RO_PD_22_FLG 0x2faf

Bit(s)	R/W	Default	Description
31-16	R		reserved.
15-8	R		ro_mcdi_pd22cnt. pull down 22 count till prevoius one field. initial = 0

Bit(s)	R/W	Default	Description
7-1	R		reserved.
0	R		ro_mcdi_pd22flg. pull down 22 flag of prevoius one field. initial = 0

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

MCDI_RO_COL_CFD_24 0x2fc8

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_24. column cofidence value 24. initial = 0

MCDI_RO_COL_CFD_25 0x2fc9

Bit(s)	R/W	Default	Description
31-0	R		ro_mcdi_col_cfd_25. column cofidence value 25. initial = 0

PULLDOWN

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

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

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
13-8	W		Ccnt_cmmin
5-0	W		Cmb_wv_min4

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

DIPD_COMB_CTRL4 0x2fd4

Bit(s)	R/W	Default	Description
30	W		Flm_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

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

DIPD_RO_COMB_0 0x2fd6

Bit(s)	R/W	Default	Description
31-0	R		frmdif

DIPD_RO_COMB_1 0x2fd7

Bit(s)	R/W	Default	Description
31-0	R		Frmdif0

DIPD_RO_COMB_2 0x2fd8

Bit(s)	R/W	Default	Description
31-0	R		Frmdif1

DIPD_RO_COMB_3 0x2fd9

Bit(s)	R/W	Default	Description
31-0	R		Frmdif2

DIPD_RO_COMB_4 0x2fda

Bit(s)	R/W	Default	Description
31-0	R		Frmdif3

DIPD_RO_COMB_5 0x2fdb

Bit(s)	R/W	Default	Description
31-0	R		Frmdif4

DIPD_RO_COMB_6 0x2fdc

Bit(s)	R/W	Default	Description
31-0	R		flddif

DIPD_RO_COMB_7 0x2fdd

Bit(s)	R/W	Default	Description
31-0	R		Flddif0

DIPD_RO_COMB_8 0x2fde

Bit(s)	R/W	Default	Description
31-0	R		Flddif1

DIPD_RO_COMB_9 0x2fdf

Bit(s)	R/W	Default	Description
31-0	R		Flddif2

DIPD_RO_COMB_10 0x2fe0

Bit(s)	R/W	Default	Description
31-0	R		Flddif3

DIPD_RO_COMB_11 0x2fe1

Bit(s)	R/W	Default	Description
31-0	R		Flddif4

DIPD_RO_COMB_12 0x2fe2

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt0

DIPD_RO_COMB_13 0x2fe3

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt1

DIPD_RO_COMB_14 0x2fe4

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt2

DIPD_RO_COMB_15 0x2fe5

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt3

DIPD_RO_COMB_16 0x2fe6

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt4

DIPD_RO_COMB_17 0x2fe7

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt5

DIPD_RO_COMB_18 0x2fe8

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt6

DIPD_RO_COMB_19 0x2fe9

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt7

DIPD_RO_COMB_20 0x2fea

Bit(s)	R/W	Default	Description
31-0	R		Ro_rt8

DNR

DNR_CTRL 0x2d00

Bit(s)	R/W	Default	Description
31:17	R/W		reserved
16	R/W		reg_dnr_en
15	R/W		reg_dnr_db_vdbstep , vdb step, 0: 4, 1: 8 . unsigned , default = 1
14	R/W		reg_dnr_db_vdbprten , vdb protectoin 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_chrmen , 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

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

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

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

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

DNR_HBOFFST_STAT 0x2d05

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_hbof_difthd , dif threshold (>=) between LR and LL/RR. unsigned , default = 2
23:16	R/W		reg_dnr_hbof_edgethd , edge threshold (<=) for LR . unsigned , default = 32
15: 8	R/W		reg_dnr_hbof_flatthd , flat threshold (>=) 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

DNR_VBOFFST_STAT 0x2d06

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_vbof_difthd , dif threshold (>=) between Up and Dw. unsigned , default = 1
23:16	R/W		reg_dnr_vbof_edgethd , edge threshold (<=) for Up/Dw. unsigned , default = 16
15: 8	R/W		reg_dnr_vbof_flatthd , flat threshold (>=) 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

DNR_GBS_STAT 0x2d07

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_gbs_edgethd , edge threshold (<=) for LR . unsigned , default = 32
23:16	R/W		reg_dnr_gbs_flatthd , flat threshold (>=) for LR . unsigned , default = 0
15: 8	R/W		reg_dnr_gbs_varthd , variation threshold (<=) for Lvar/Rvar. unsigned , default = 16
7: 0	R/W		reg_dnr_gbs_difthd , dif threshold (>=) between LR and LL/RR. unsigned , default = 2

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

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

DNR_LUMA 0x2d0a

Bit(s)	R/W	Default	Description
31:27	R/W		reserved
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_wcmod , 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

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

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

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

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_hgbsthd , gbs thd (>=) for hbs calc. . unsigned , default = 1

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

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
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

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

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
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

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

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

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_vgbsthd , gbs thd (>=) for vbs calc. . unsigned , default = 1

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_yvdeltaadjoffst , 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

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_cvdeltaadjoffst , 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

DNR_RO_GBS_STAT_LR 0x2d18

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_lr . unsigned , default = 0

DNR_RO_GBS_STAT_LL 0x2d19

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_ll . unsigned , default = 0

DNR_RO_GBS_STAT_RR 0x2d1a

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_rr . unsigned , default = 0

DNR_RO_GBS_STAT_DIF 0x2d1b

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_dif . unsigned , default = 0

DNR_RO_GBS_STAT_CNT 0x2d1c

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_cnt . unsigned , default = 0

DNR_RO_HBOF_STAT_CNT_0 0x2d1d

Bit(s)	R/W	Default	Description
31: 0	R		ro_hbof_stat_cnt0 . unsigned , default = 0

DNR_RO_HBOF_STAT_CNT_31 0x2d3c

Bit(s)	R/W	Default	Description
31: 0	R		ro_hbof_stat_cnt31 . unsigned , default = 0

DNR_RO_VBOF_STAT_CNT_0 0x2d3d

Bit(s)	R/W	Default	Description
31: 0	R		ro_vbof_stat_cnt0 . unsigned , default = 0

DNR_RO_VBOF_STAT_CNT_31 0x2d5c

Bit(s)	R/W	Default	Description
31: 0	R		ro_vbof_stat_cnt31 . unsigned , default = 0

DNR_CTRL 0x2d00

Bit(s)	R/W	Default	Description
31:17	R/W		reserved
16	R/W		reg_dnr_en
15	R/W		reg_dnr_db_vdbstep , vdb step, 0: 4, 1: 8 . unsigned , default = 1
14	R/W		reg_dnr_db_vdbprten , vdb protectoin 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_chrmen , 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

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

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

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

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

DNR_HBOFFST_STAT 0x2d05

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_hbof_difthd , dif threshold (>=) between LR and LL/RR. unsigned , default = 2
23:16	R/W		reg_dnr_hbof_edgethd , edge threshold (<=) for LR . unsigned , default = 32
15: 8	R/W		reg_dnr_hbof_flatthd , flat threshold (>=) 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

DNR_VBOFFST_STAT 0x2d06

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_vbof_difthd , dif threshold (>=) between Up and Dw. unsigned , default = 1
23:16	R/W		reg_dnr_vbof_edgethd , edge threshold (<=) for Up/Dw. unsigned , default = 16
15: 8	R/W		reg_dnr_vbof_flatthd , flat threshold (>=) 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

DNR_GBS_STAT 0x2d07

Bit(s)	R/W	Default	Description
31:24	R/W		reg_dnr_gbs_edgethd , edge threshold (<=) for LR . unsigned , default = 32
23:16	R/W		reg_dnr_gbs_flatthd , flat threshold (>=) for LR . unsigned , default = 0
15: 8	R/W		reg_dnr_gbs_varthd , variation threshold (<=) for Lvar/Rvar. unsigned , default = 16
7: 0	R/W		reg_dnr_gbs_difthd , dif threshold (>=) between LR and LL/RR. unsigned , default = 2

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

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

DNR_LUMA 0x2d0a

Bit(s)	R/W	Default	Description
31:27	R/W		reserved
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_wcmod , 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

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

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

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

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_hgbsthd , gbs thd (>=) for hbs calc. . unsigned , default = 1

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

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
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

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

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
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

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

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

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_vgbsthd , gbs thd (>=) for vbs calc. . unsigned , default = 1

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_yvdeltaadjoffst , 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

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_cvdeltaadjoffst , 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

DNR_RO_GBS_STAT_LR 0x2d18

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_lr . unsigned , default = 0

DNR_RO_GBS_STAT_LL 0x2d19

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_ll . unsigned , default = 0

DNR_RO_GBS_STAT_RR 0x2d1a

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_rr . unsigned , default = 0

DNR_RO_GBS_STAT_DIF 0x2d1b

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_dif . unsigned , default = 0

DNR_RO_GBS_STAT_CNT 0x2d1c

Bit(s)	R/W	Default	Description
31: 0	R		ro_gbs_stat_cnt . unsigned , default = 0

DNR_RO_HBOF_STAT_CNT_0 0x2d1d

Bit(s)	R/W	Default	Description
31: 0	R		ro_hbof_stat_cnt0 . unsigned , default = 0

DNR_RO_HBOF_STAT_CNT_31 0x2d3c

Bit(s)	R/W	Default	Description
31: 0	R		ro_hbof_stat_cnt31 . unsigned , default = 0

DNR_RO_VBOF_STAT_CNT_0 0x2d3d

Bit(s)	R/W	Default	Description
31: 0	R		ro_vbof_stat_cnt0 . unsigned , default = 0

DNR_RO_VBOF_STAT_CNT_31 0x2d5c

Bit(s)	R/W	Default	Description
31: 0	R		ro_vbof_stat_cnt31 . unsigned , default = 0

DNR_DM_CTRL 0x2d60

Bit(s)	R/W	Default	Description
31:13	R/W		reserved
12	R/W	1	reg_dnr_fedgeflg_en , 1 to enable edge flag calculation for each frame
11	R/W	1	reg_dnr_fedgeflg_cl , 1 to clear the edge flag to 0 for each frame
10	R/W	0	reg_dnr_fedgeflg_df , user defined edge flag when reg_dnr_fedgeflg_en = 0
9	R/W	0	reg_dnr_dm_en , 1 to enable de-mosquito unit
8	R/W	1	reg_dnr_dm_chrmen , 1 to enable chrome processing for de-mosquito
7: 6	R/W	3	reg_dnr_dm_level , de-mosquito level
5: 4	R/W	1	reg_dnr_dm_leveldw0 , level down when gbs is small
3: 2	R/W	1	reg_dnr_dm_leveldw1 , level down for flat blocks
1: 0	R/W	0	reg_dnr_dm_gbsthd , small/large threshold for gbs

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
15:14	R/W		reserved
13: 8	R/W	32	reg_dnr_dm_alpgain , gain for DM/NR alpha, normalized 32 as 1
7: 0	R/W	0	reg_dnr_dm_alpoffset , offset for DM/NR alpha

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
7: 0	R/W	4	reg_dnr_dm_rgndifthd

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_rgngain , normalized 16 as 1
7:6	R/W		reserved
5: 0	R/W	0	reg_dnr_dm_rgnofst

DNR_DM_DIR_MISC 0x2d64

Bit(s)	R/W	Default	Description
31:30	R/W		reserved
29	R/W	1	reg_dnr_dm_diralpen
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
7:5	R/W		reserved
4: 0	R/W	31	reg_dnr_dm_diralpmax

DNR_DM_COR_DIF 0x2d65

Bit(s)	R/W	Default	Description
31:4	R/W		reserved
3:1	R/W	3	reg_dnr_dm_cordifshft
0	R/W	1	reg_dnr_dm_cordifmod , 0: use max dir dif as cordif, 1: use max3x3-min3x3 as cordif

DNR_DM_FLT_THD 0x2d66

Bit(s)	R/W	Default	Description
31:24	R/W	4	reg_dnr_dm_fltthd00 , block flat threshold0 for block average difference when gbs is small
23:16	R/W	6	reg_dnr_dm_fltthd01 , block flat threshold1 for block average difference when gbs is small
15: 8	R/W	9	reg_dnr_dm_fltthd10 , block flat threshold0 for block average difference when gbs is larger
7: 0	R/W	12	reg_dnr_dm_fltthd11 , block flat threshold1 for block average difference when gbs is larger

DNR_DM_VAR_THD 0x2d67

Bit(s)	R/W	Default	Description
31:24	R/W	2	reg_dnr_dm_varthd00 , block variance threshold0 (>=) when gbs is small
23:16	R/W	15	reg_dnr_dm_varthd01 , block variance threshold1 (<=) when gbs is small
15: 8	R/W	3	reg_dnr_dm_varthd10 , block variance threshold0 (>=) when gbs is larger
7: 0	R/W	24	reg_dnr_dm_varthd11 , block variance threshold1 (<=) when gbs is larger

DNR_DM_EDGE_DIF_THD 0x2d68

Bit(s)	R/W	Default	Description
31:24	R/W	32	reg_dnr_dm_edgethd0 , block edge threshold (<=) when gbs is small
23:16	R/W	48	reg_dnr_dm_edgethd1 , block edge threshold (<=) when gbs is larger
15: 8	R/W	48	reg_dnr_dm_difthd0 , block dif threshold (<=) when gbs is small
7: 0	R/W	64	reg_dnr_dm_difthd1 , block dif threshold (<=) when gbs is larger

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 (>=) when gbs is small
7: 0	R/W	128	reg_dnr_dm_avgthd1 , block average threshold (<=) when gbs is larger

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_avgdifthd , block average dif threshold(<) between cur and up block for flat block
7: 0	R/W	1	reg_dnr_dm_vardifthd , block variance dif threshold(>=) between cur and up block

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 (>=) for edge block detect
15: 8	R/W	40	reg_dnr_dm_edgethd2 , block edge threshold (>=)
7: 0	R/W	80	reg_dnr_dm_difthd2 , block dif threshold (>=)

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_fltalp, pre-defined alpha for dm and nr blending when block is flat with mos
15:12	R/W		reserved
11:8	R/W	12	reg_dnr_dm_fltminbdif, pre-defined min block dif for dm filter when block is flat with mos
7	R/W		reserved
6:2	R/W	16	reg_dnr_dm_difnormgain, gain for pixel dif normalization for dm filter.
1	R/W	1	reg_dnr_dm_difnormen , enable pixel dif normalization for dm filter
0	R/W	0	reg_dnr_dm_difupden , enable block dif update using max of left,cur,right difs

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
4:0	R/W	13	reg_dnr_dm_sdiflut2

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
4:0	R/W	5	reg_dnr_dm_sdiflut5

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
4:0	R/W	0	reg_dnr_dm_sdiflut8

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
4:0	R/W	12	reg_dnr_dm_ldiflut2

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
4:0	R/W	16	reg_dnr_dm_ldiflut5

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
4:0	R/W	16	reg_dnr_dm_ldiflut8

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
4:0	R/W	3	reg_dnr_dm_dif2normlut2

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
4:0	R/W	1	reg_dnr_dm_dif2normlut5

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
15:13	R/W		reserved
12:8	R/W	1	reg_dnr_dm_dif2normlut7
7:5	R/W		reserved
4:0	R/W	1	reg_dnr_dm_dif2normlut8

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
7:0	R/W	128	reg_gms_stat_thd1

DNR_RO_DM_GMS_STST_CNT 0x2d77

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

DNR_RO_DM_GMS_STST_MS 0x2d78

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

27.3 RDMA

Register Direct Memory Access (RDMA) is designed in view of speeding up software programming. With RDMA module, instead of accessing registers one by one, SW stores the register addresses (and data) in one or more blocks of memory, software then programs the few registers of RDMA, mainly of the memory block location, and VCBus access mode. RDMA will either automatically (by programmable interrupts) or manually (by SW “go” command) fetch the memory data and transfer to VCBus. If the access is for read, RDMA puts the read-back data back to the memory block.

The input and output signals of RDMA is listed in the following table.

Table IV.28.1 Input Signal of RDMA

Signal	Description
axi_awready axi_wready axi_bvalid axi_arready axi_rdata[127:0] axi_rvalid	DDR memory interface inputs.
sys_grant sys_read_rdy sys_rddat[31:0]	VCBus master interface inputs.
cpu_spadr[15:0] cpu_wrdat[31:0] cpu_spwr cpu_spval	VCBus slave interface inputs.
intr_clk interrupt_in[7:0]	Interrupt vectors that can be programmably masked to start automatic RDMA transfer. Interrupt bit map:

Signal	Description
	Bit 7: vid0_wr_irq Bit 6: viu1_line_n_int Bit 5: det3d_int Bit 4: deint_irq Bit 3: vid1_wr_irq Bit 2: vdin1_vsync_int Bit 1: vdin0_vsync_int Bit 0: viu1_vsync_int
mem_pd	MEM power down control
clk	Operating clock from cts_vpu_clk.
reset_n	Asynchronous reset, active low .
test_mode scan_clk btc_clk btc_mode btc_di	For production test.

Table IV.29.2 Output Signal of RDMA

Signal	Description
ddr_req_busy	Status signal output to vpu for dynamic request priority control.
axi_awaddr[31:0] axi_awlen[3:0] axi_awugt axi_awvalid axi_wdata[127:0] axi_wstrb[15:0] axi_wlast axi_wvalid axi_araddr[31:0] axi_arlen[3:0] axi_arugt axi_arvalid	DDR memory interface inputs.
sys_spwr sys_spval sys_spadr[15:0] sys_wrdat[31:0]	VCBus master interface outputs.
cpu_grant cpu_read_rdy cpu_rddat[31:0]	VCBus slave interface outputs.
rdma_done_int	RDMA action done interrupt status. To connect to an interrupt service module.
btc_do	For production test.

Register Description

Table IV.28. 3 Register List of RDMA

Addr	Name	RW	Function
0xd0104400	RDMA_AHB_START_ADDR_MAN	RW	Mem block start addr for DMA manual start.
0xd0104404	RDMA_AHB_END_ADDR_MAN	RW	Mem block end addr for DMA manual start.

Addr	Name	RW	Function
0xd0104408	RDMA_AHB_START_ADDR_1	RW	Mem block start addr for DMA auto start source 1.
0xd010440c	RDMA_AHB_END_ADDR_1	RW	Mem block end addr for DMA auto start source 1.
0xd0104410	RDMA_AHB_START_ADDR_2	RW	Mem block start addr for DMA auto start source 2.
0xd0104414	RDMA_AHB_END_ADDR_2	RW	Mem block end addr for DMA auto start source 2.
0xd0104418	RDMA_AHB_START_ADDR_3	RW	Mem block start addr for DMA auto start source 3.
0xd010441c	RDMA_AHB_END_ADDR_3	RW	Mem block end addr for DMA auto start source 3.
0xd0104420	RDMA_AHB_START_ADDR_4	RW	Mem block start addr for DMA auto start source 4.
0xd0104424	RDMA_AHB_END_ADDR_4	RW	Mem block end addr for DMA auto start source 4.
0xd0104428	RDMA_AHB_START_ADDR_5	RW	Mem block start addr for DMA auto start source 5.
0xd010442c	RDMA_AHB_END_ADDR_5	RW	Mem block end addr for DMA auto start source 5.
0xd0104430	RDMA_AHB_START_ADDR_6	RW	Mem block start addr for DMA auto start source 6.
0xd0104434	RDMA_AHB_END_ADDR_6	RW	Mem block end addr for DMA auto start source 6.
0xd0104438	RDMA_AHB_START_ADDR_7	RW	Mem block start addr for DMA auto start source 7.
0xd010443c	RDMA_AHB_END_ADDR_7	RW	Mem block end addr for DMA auto start source 7.
0xd0104440	RDMA_ACCESS_AUTO	RW	DMA auto start control.
0xd0104444	RDMA_ACCESS_AUTO2	RW	DMA auto start control.
0xd0104448	RDMA_ACCESS_AUTO3	RW	DMA auto start control.
0xd010444c	RDMA_ACCESS_MAN	RW	DMA manual start control.
0xd0104450	RDMA_CTRL	RW	General control.
0xd0104454	RDMA_STATUS	R	Status.
0xd0104458	RDMA_STATUS2	R	Status.
0xd010445c	RDMA_STATUS3	R	Status.

RDMA_AHB_START_ADDR_MAN

Mem block start addr for DMA manual start.

RDMA_AHB_END_ADDR_MAN

Mem block end addr for DMA manual start.

RDMA_AHB_START_ADDR_1

Mem block start addr for DMA auto start source 1.

RDMA_AHB_END_ADDR_1

Mem block end addr for DMA auto start source 1.

RDMA_AHB_START_ADDR_2

Mem block start addr for DMA auto start source 2.

RDMA_AHB_END_ADDR_2

Mem block end addr for DMA auto start source 2.

RDMA_AHB_START_ADDR_3

Mem block start addr for DMA auto start source 3.

RDMA_AHB_END_ADDR_3

Mem block end addr for DMA auto start source 3.

RDMA_AHB_START_ADDR_4

Mem block start addr for DMA auto start source 4.

RDMA_AHB_END_ADDR_4

Mem block end addr for DMA auto start source 4.

RDMA_AHB_START_ADDR_5

Mem block start addr for DMA auto start source 5.

RDMA_AHB_END_ADDR_5

Mem block end addr for DMA auto start source 5.

RDMA_AHB_START_ADDR_6

Mem block start addr for DMA auto start source 6.

RDMA_AHB_END_ADDR_6

Mem block end addr for DMA auto start source 6.

RDMA_AHB_START_ADDR_7

Mem block start addr for DMA auto start source 7.

RDMA_AHB_END_ADDR_7

Mem block end addr for DMA auto start source 7.

RDMA_ACCESS_AUTO

Bits(s)	R/W	Default	Description
31:24	RW	0	ctrl_enable_int_3: Interrupt inputs enable mask for auto-start 3.
23:16	RW	0	ctrl_enable_int_2: Interrupt inputs enable mask for auto-start 2.
15:8	RW	0	ctrl_enable_int_1: Interrupt inputs enable mask for auto-start 1.
7	RW	0	ctrl_cbus_write_3: 0=Register read for auto-start 3; 1=Register write for auto-start 3.
6	RW	0	ctrl_cbus_write_2: 0=Register read for auto-start 2; 1=Register write for auto-start 2.
5	RW	0	ctrl_cbus_write_1: 0=Register read for auto-start 1; 1=Register write for auto-start 1.
4	R	0	Reserved.
3	RW	0	ctrl_cbus_addr_incr_3: 0=Non-incremental register access for auto-start 3; 1=Incremental register access for auto-start 3.
2	RW	0	ctrl_cbus_addr_incr_2: 0=Non-incremental register access for auto-start 2; 1=Incremental register access for auto-start 2.
1	RW	0	ctrl_cbus_addr_incr_1: 0=Non-incremental register access for auto-start 1; 1=Incremental register access for auto-start 1.
0	R	0	Reserved.

RDMA_ACCESS_AUTO2

Bits(s)	R/W	Default	Description
31:8	R	0	Reserved.
7	RW	0	ctrl_cbus_write_7: 0=Register read for auto-start 7; 1=Register write for auto-start 7.
6	RW	0	ctrl_cbus_write_6: 0=Register read for auto-start 6; 1=Register write for auto-start 6.

Bits(s)	R/W	Default	Description
5	RW	0	ctrl_cbus_write_5: 0=Register read for auto-start 5; 1=Register write for auto-start 5.
4	RW	0	ctrl_cbus_write_4: 0=Register read for auto-start 4; 1=Register write for auto-start 4.
3	RW	0	ctrl_cbus_addr_incr_7: 0=Non-incremental register access for auto-start 7; 1=Incremental register access for auto-start 7.
2	RW	0	ctrl_cbus_addr_incr_6: 0=Non-incremental register access for auto-start 6; 1=Incremental register access for auto-start 6.
1	RW	0	ctrl_cbus_addr_incr_5: 0=Non-incremental register access for auto-start 5; 1=Incremental register access for auto-start 5.
0	RW	0	ctrl_cbus_addr_incr_4: 0=Non-incremental register access for auto-start 4; 1=Incremental register access for auto-start 4.

RDMA_ACCESS_AUTO3

Bits(s)	R/W	Default	Description
31:24	RW	0	ctrl_enable_int_7: Interrupt inputs enable mask for auto-start 7.
23:16	RW	0	ctrl_enable_int_6: Interrupt inputs enable mask for auto-start 6.
15:8	RW	0	ctrl_enable_int_5: Interrupt inputs enable mask for auto-start 5.
7:0	RW	0	ctrl_enable_int_4: Interrupt inputs enable mask for auto-start 4.

RDMA_ACCESS_MAN

Bits(s)	R/W	Default	Description
31:3	R	0	Reserved.
2	RW	0	ctrl_cbus_write_man: 0=Register read for manual-start; 1=Register write for manual-start.
1	RW	0	ctrl_cbus_addr_incr_man: 0=Non-incremental register access for manual-start; 1=Incremental register access for manual-start.
0	W	0	Write 1 to this bit to manual-start DMA. This bit always read back 0.

RDMA_CTRL

Bits(s)	R/W	Default	Description
31:24	W	0	ctrl_clr_rdma_done_int: 8-bit, 1 bit for each RDMA source. [0] 1=Clr rdma_done on manual RDMA src 0. Self clear. Always read 0. [1] 1=Clr rdma_done on auto RDMA src 1. Self clear. Always read 0. [2] 1=Clr rdma_done on auto RDMA src 2. Self clear. Always read 0. [3] 1=Clr rdma_done on auto RDMA src 3. Self clear. Always read 0. [4] 1=Clr rdma_done on auto RDMA src 4. Self clear. Always read 0. [5] 1=Clr rdma_done on auto RDMA src 5. Self clear. Always read 0. [6] 1=Clr rdma_done on auto RDMA src 6. Self clear. Always read 0. [7] 1=Clr rdma_done on auto RDMA src 7. Self clear. Always read 0.
22:8	R	0	Reserved.
7	RW	0	ctrl_axi_wr_urgent: 0=DDR write request not urgent; 1=DDR write request urgent.

Bits(s)	R/W	Default	Description
6	RW	0	ctrl_axi_rd_urgent: 0=DDR read request not urgent; 1=DDR read request urgent.
5:4	RW	0	ctrl_ahb_wr_burst_size: 0=AHB write request burst size 4x128-b; 1=AHB write request burst size 8x128-b; 2=AHB write request burst size 12x128-b; 3=AHB write request burst size 16x128-b.
3:2	RW	0	ctrl_ahb_rd_burst_size: 0=AHB read request burst size 4x128-b; 1=AHB read request burst size 8x128-b; 2=AHB read request burst size 12x128-b; 3=AHB read request burst size 16x128-b.
1	RW	0	ctrl_sw_reset: 0=No reset; 1=Reset RDMA logics except its own register interface.
0	RW	0	ctrl_free_clk_enable: 0=Enable clock gating; 1=No clock gating, enable free clock.

RDMA_STATUS

Bits(s)	R/W	Default	Description
31:24	R	0	rdma_done_flag: interrupt status. 8-bit, one bit for each RDMA source. [0] Manual src 0 RDMA done intr status. [1] Auto src 1 RDMA done intr status. [2] Auto src 2 RDMA done intr status. [3] Auto src 3 RDMA done intr status. [4] Auto src 4 RDMA done intr status. [5] Auto src 5 RDMA done intr status. [6] Auto src 6 RDMA done intr status. [7] Auto src 7 RDMA done intr status.
23	R	0	Reserved.
22	R	0	err_axi_wrfifo_undflow
21	R	0	err_axi_wrfifo_ovrflow
20	R	0	err_axi_rdfifo_ovrflow
19:18	R	0	axi_w_st: AXI write data status: 0=Idle; 1=Wait Data; 2=Req; 3=Wait BRSP.
17:16	R	0	axi_aw_st: AXI write address status: 0=Idle; 1=Req; 2=Wait RDY.
15:14	R	0	axi_ar_st: AXI read address status: 0=Idle; 1=Wait FIFO; 2=Req; 3=Wait Done.
13:11	R	0	rdma_id_curr
10:8	R	0	rdma_id_pipe

7:0	R	0	req_latch: Requests that are yet to be serviced. E.g.: 00000000=No request; 00000001=Req 0 waiting; 00001100=Req 2 and 3 waiting; 10000000=Req7 waiting.

RDMA_STATUS2

Bits(s)	R/W	Default	Description
31:30	R	0	Reserved.
29:24	R	0	axi_wrfifo_cnt. FIFO count for buffering VCBus read data to sent to AXI.
23:22	R	0	Reserved.
21:16	R	0	axi_rdfifo_cnt. FIFO count for buffering data read from AXI.
15:14	R	0	Reserved.
13:8	R	32	axi_wrfifo_room. FIFO room for buffering VCBus read data to sent to AXI.
7:6	R	0	Reserved.
5:0	R	32	axi_rdfifo_room. FIFO room for buffering data read from AXI.

RDMA_STATUS3

Bits(s)	R/W	Default	Description
31:24	R	0	Reserved.
23:8	R	0	axi_b_pending
7:0	R	0	axi_aw_pending

27.4 HDMI

S905X has Built-in HDMI 2.0b transmitter including both controller and PHY with CEC and HDCP 2,2, 4Kx2K@60 max resolution output.

This section specifies HDMITX module's top-level registers only. HDMITX module instantiates two IPs – HDMI TX Controller and HDCP2.2 Controller. These two IPs internal registers are specified by the corresponding IPs' proprietary documents.

Register Description

HDMITX Top-Level and HDMI TX Contoller IP Register Access

Accessing HDMITX Top-Level and TX Controller IP registers is by indirectly accessing two memory addresses, one of the addresses is mapped to register address, the other one is mapped to read/write data. There is also an extra address for access control. See Table below.

Table IV.28.4 HDMITX Top-Level and TX Controller register access ports

Absolute Address	Address Mnemonic	Bit(s)	Default	Description
0xda83a000	HDMITX_ADDR_PORT	[31:0]	0	Map to register address.
0xda83a004	HDMITX_DATA_PORT	[31:0]	0	Map to register read/write data.
0xda83a008	HDMITX_CTRL_PORT	[31:0]	0x3ff	Register access control. See Table1.1.

Table IV.28.5 HDMITX Control Port for register access

Bit(s)	R/W	Default	Description
31:16	R	0	Reserved.
15	RW	0	err_en: 0=No limit on PREADY wait time; 1=Enable APB access PREADY watchdog timer.

14	W	0	stack_err_clr. 1=Clear stack_err. Always read back 0.
13	R	0	stack_err. 1=Stack over/underflow.
12	RW	0	stack_enable. Control whether to stack the indirect reg access sequence, in case register access is interrupted by an interrupt that also indirectly access HDMITX. 0=Old behaviour that does not support interruption; 1=Enable stack function.
11:0	RW	0x3ff	max_err: If err_en=1, maximum number of APB clock cycle wait time for PREADY assertion, before it asserts APB access error.

Table IV.28.6 HDMITX sub-module base address

Base Address	Module Name	Description
0x00	dev_offset_top	Device address offset for AmLogic top-level wrapper that instantiates the IPs.
0x10	dev_offset_dwc	Device address offset for HDMI TX Controller IP module.

To write to a Top-level register:

**((volatile uint32_t *) (HDMITX_ADDR_PORT+dev_offset_top)) = addr;*
**((volatile uint32_t *) (HDMITX_DATA_PORT+dev_offset_top)) = data;*

To write to a TX controller IP register:

**((volatile uint32_t *) (HDMITX_ADDR_PORT+dev_offset_dwc)) = addr;*
**((volatile uint32_t *) (HDMITX_DATA_PORT+dev_offset_dwc)) = data;*

To read from a Top-level register:

**((volatile uint32_t *) (HDMITX_ADDR_PORT+dev_offset_top)) = addr;*
*data = *((volatile uint32_t *) (HDMITX_DATA_PORT+dev_offset_top));*

To read from a TX controller IP register:

**((volatile uint32_t *) (HDMITX_ADDR_PORT+dev_offset_dwc)) = addr;*
*data = *((volatile uint32_t *) (HDMITX_DATA_PORT+dev_offset_dwc));*

HDCP2.2 IP Register Access

Table IV.28.7 HDCP2.2 IP register base address

Absolute Address	Address Mnemonic	Description
0xd0044000	ELP_ESM_HPI_REG_BASE	Address base to HDCP2.2 IP register access.

To write to an HDCP2.2 IP register:

**((volatile uint32_t *) (ELP_ESM_HPI_REG_BASE+addr)) = data;*

To read from an HDCP2.2 IP register:

*data = *((volatile uint32_t *) (ELP_ESM_HPI_REG_BASE+addr));*

Table IV.28.8 HDMITX Top-Level Registers

Addr	Name	RW	Function
0x000	HDMITX_TOP_SW_RESET	RW	Software reset sub-modules.

Addr	Name	RW	Function
0x001	HDMITX_TOP_CLK_CNTL	RW	Clock gating and inversion.
0x002	HDMITX_TOP_HPD_FILTER	RW	HPD input glitch filter.
0x003	HDMITX_TOP_INTR_MASKN	RW	Interrupt mask.
0x004	HDMITX_TOP_INTR_STAT	RW	Interrupt status.
0x005	HDMITX_TOP_INTR_STAT_CLR	W	Interrupt clear.
0x006	HDMITX_TOP_BIST_CNTL	RW	Build-In Self Test(BIST) control.
0x007	HDMITX_TOP_SHIFT_PTTN_012	RW	Shift pattern for BIST.
0x008	HDMITX_TOP_SHIFT_PTTN_345	RW	Shift pattern for BIST.
0x009	HDMITX_TOP_SHIFT_PTTN_67	RW	Shift pattern for BIST.
0x00A	HDMITX_TOP_TMDS_CLK_PTTN_01	RW	TMDS clock pattern for generating /10 or /40 rate clock.
0x00B	HDMITX_TOP_TMDS_CLK_PTTN_23	RW	TMDS clock pattern for generating /10 or /40 rate clock.
0x00C	HDMITX_TOP_TMDS_CLK_PTTN_CNTL	RW	TMDS clock pattern for generating /10 or /40 rate clock.
0x00D	HDMITX_TOP_REVOCMEM_STAT	RW	Revocmem status
0x00E	HDMITX_TOP_STAT0	RW	Status.
0x010	HDMITX_TOP_SKP_CNTL_STAT	RW	SKP interface control for HDCP2.2.
0x011	HDMITX_TOP_NONCE_0	W	Nonce[31:0] for HDCP2.2.
0x012	HDMITX_TOP_NONCE_1	W	Nonce[63:32] for HDCP2.2.
0x013	HDMITX_TOP_NONCE_2	W	Nonce[95:64] for HDCP2.2.
0x014	HDMITX_TOP_NONCE_3	W	Nonce[127:96] for HDCP2.2.
0x015	HDMITX_TOP_PKF_0	W	PKF[31:0] for HDCP2.2.
0x016	HDMITX_TOP_PKF_1	W	PKF[63:32] for HDCP2.2.
0x017	HDMITX_TOP_PKF_2	W	PKF[95:64] for HDCP2.2.
0x018	HDMITX_TOP_PKF_3	W	PKF[127:96] for HDCP2.2.
0x019	HDMITX_TOP_DUK_0	W	DUK [31:0] for HDCP2.2.
0x01A	HDMITX_TOP_DUK_1	W	DUK [63:32] for HDCP2.2.
0x01B	HDMITX_TOP_DUK_2	W	DUK [95:64] for HDCP2.2.
0x01C	HDMITX_TOP_DUK_3	W	DUK [127:96] for HDCP2.2.
0x01D	HDMITX_TOP_INFILTER	RW	DDC and CEC input glitch filter control.
0x01E	HDMITX_TOP_NSEC_SCRATCH	RW	Scratch register for non-secure access.
0x01F	HDMITX_TOP_SEC_SCRATCH	RW	Scratch register for secure access.

HDMITX_TOP_SW_RESET

Bit(s)	R/W	Default	Description
31:8	R	0	Reserved
7	RW	1	Reserved
6	RW	1	sw_reset_flt: to reset DDC&CEC input glitch filter. 0=Release from reset; 1=Apply reset.
5	RW	1	sw_reset_hdcp22: to reset HDCP2.2 IP. 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.
2	RW	1	sw_reset_mem: to reset KSV/REVOC mem. 0=Release from reset; 1=Apply reset.
1	RW	1	sw_reset_rnd: to reset random number interface to HDCP. 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.

HDMITX_TOP_CLK_CNTL

Bit(s)	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.
30:13	RW	0	Reserved
12	RW	0	I2S_ws_inv: 1= Invert I2S_ws.
11	RW	0	I2S_clk_inv: 1= Invert I2S_clk.
10	RW	0	spdif_clk_inv: 1= Invert spdif_clk.
9	RW	0	tmds_clk_inv: 1= Invert tmds_clk.
8	RW	0	pixel_clk_inv: 1= Invert pixel_clk.
7	RW	0	hdcp22_skpclk_en: 1= Enable skpclk to HTX_HDCP2.2 IP.
6	RW	0	hdcp22_esmclk_en: 1= Enable esmclk to HTX_HDCP2.2 IP.
5	RW	0	hdcp22_tmdsclk_en: 1= Enable tmds_clk to HDCP2.2 IP.
4	RW	0	cec_clk_en: 1= Enable cec_clk.
3	RW	0	I2S_clk_en: 1= Enable I2S_clk.
2	RW	0	spdif_clk_en: 1= Enable spdif_clk.
1	RW	0	tmds_clk_en: 1= Enable tmds_clk.
0	RW	0	pixel_clk_en: 1= Enable pixel_clk.

HDMITX_TOP_HPD_FILTER

Bit(s)	R/W	Default	Description
31:16	R	0	Reserved
15:12	RW	0	hpd_glitch_width: Filter out glitch <= hpd_glitch_width.
11:0	RW	0	hpd_valid_width: Filter out width <= hpd_valid_width * 1024.

HDMITX_TOP_INTR_MASKN

Interrupt MASKN, one bit per interrupt source.

0= Disable interrupt source;

1= Enable interrupt source.

Bit(s)	R/W	Default	Description
31:8	R	0	Reserved
7:5	RW	0	Reserved
4	RW	0	hdcp22_rndnum_err
3	RW	0	nonce_rfrsh_rise
2	RW	0	hpd_fall
1	RW	0	hpd_rise
0	RW	0	TX Controller IP interrupt.

HDMITX_TOP_INTR_STAT

Interrupt status. For Each Bit of Bit[4:0], write 1 to manually set the interrupt bit, read back the interrupt status.

Bit(s)	R/W	Default	Description
31	R	0	Shadowing TX Controller IP interrupt status flag.

Bit(s)	R/W	Default	Description
30	R	0	Shadowing HDCP2.2 IP interrupt status flag.
29:5	R	0	Reserved
4	RW	0	hdcp22_rndnum_err
3	RW	0	nonce_rfrsh_rise
2	RW	0	hpd_fall
1	RW	0	hpd_rise
0	RW	0	TX Controller IP interrupt.

HDMITX_TOP_INTR_STAT_CLR

Interrupt status clear. For Each Bit, write 1 to clear the interrupt bit.

Bit(s)	R/W	Default	Description
31:5	R	0	Reserved
4	W	0	hdcp22_rndnum_err
3	W	0	nonce_rfrsh_rise
2	W	0	hpd_fall
1	W	0	hpd_rise
0	W	0	TX Controller IP interrupt.

HDMITX_TOP_BIST_CNTL

Bit(s)	R/W	Default	Description
31:16	R	0	Reserved
15	RW	0	Reserved
14:12	RW	0	tmds_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.
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.

HDMITX_TOP_SHIFT_PTTN_012

Bit(s)	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].

HDMITX_TOP_SHIFT_PTTN_345

Bit(s)	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].

HDMITX_TOP_SHIFT_PTTN_67

Bit(s)	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].

HDMITX_TOP_TMDS_CLK_PTTN_01

Bit(s)	R/W	Default	Description
31:26	R	0	Reserved
25:16	RW	0	tmds_clk_pttn[19:10].
15:10	R	0	Reserved
9:0	RW	0	tmds_clk_pttn[9:0].

HDMITX_TOP_TMDS_CLK_PTTN_23

Bit(s)	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].

HDMITX_TOP_TMDS_CLK_PTTN_CNTL

Bit(s)	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.

HDMITX_TOP_REVOCMEM_STAT

Bit(s)	R/W	Default	Description
31:1	R	0	Reserved
0	RW	0	revocmem_wr_fail: Read back 1 to indicate Host write REVOC MEM failure, write 1 to clear the failure flag.

HDMITX_TOP_STAT0

Bit(s)	R/W	Default	Description
31:1	R	0	Reserved
0	R	0	filtered HPD status: 0= HPD low; 1= HPD high.

HDMITX_TOP_SKP_CNTL_STAT

Bit(s)	R/W	Default	Description
31	R	0	Status of nonce_vld signal.
30:4	R	0	Reserved
3	RW	0	rndnum_hdcp22: 0=Randon number generator if enabled for hdcp1.4; 1= Randon number generator if enabled for hdcp2.2.
2	RW	0	DUK_vld:Set to 1 once DUK is written.
1	RW	0	PKF_vld:Set to 1 once PKF is written.

Bit(s)	R/W	Default	Description
0	RW	0	nonce_hw_en: 1=Use HW nonce; 0=Use SW nonce from reg HDMITX_TOP_NONCE_0/1/2/3.

HDMITX_TOP_NONCE_0

Bit(s)	R/W	Default	Description
31:0	W	0	nonce[31:0]

HDMITX_TOP_NONCE_1

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

HDMITX_TOP_NONCE_2

Bit(s)	R/W	Default	Description
31:0	W	0	nonce[95:64]

HDMITX_TOP_NONCE_3

Bit(s)	R/W	Default	Description
31:0	W	0	nonce[127:96]

HDMITX_TOP_PKF_0

Bit(s)	R/W	Default	Description
31:0	W	0	PKF[31:0]

HDMITX_TOP_PKF_1

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

HDMITX_TOP_PKF_2

Bit(s)	R/W	Default	Description
31:0	W	0	PKF[95:64]

HDMITX_TOP_PKF_3

Bit(s)	R/W	Default	Description
31:0	W	0	PKF[127:96]

HDMITX_TOP_DUK_0

Bit(s)	R/W	Default	Description
31:0	W	0	DUK[31:0]

HDMITX_TOP_ DUK_1

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

HDMITX_TOP_ DUK_2

Bit(s)	R/W	Default	Description
31:0	W	0	DUK[95:64]

HDMITX_TOP_ DUK_3

Bit(s)	R/W	Default	Description
31:0	W	0	DUK[127:96]

HDMITX_TOP_INFILTER

Bit(s)	R/W	Default	Description
31:27	RW	0	Reserved
26:24	RW	0	For DDC infilter: filter internal clock divider. Refer to CEC infilter.
23:16	RW	0	For DDC infilter: sampling clock divider. Refer to CEC infilter.
15:11	RW	0	Reserved
10: 8	RW	0	For CEC infilter: filter internal clock divider. 0=No divide; 1=Divide by 2; 2=Divide by 3; … 7=Divide by 8.
7:0	RW	0	For CEC 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;

HDMITX_TOP_NSEC_SCRATCH

Bit(s)	R/W	Default	Description
31:0	RW	0	Scratch register that can be used for either secure or non-secure reg access.

HDMITX_TOP_SEC_SCRATCH

Bit(s)	R/W	Default	Description
31:0	RW	0	Scratch register that can be used for secure reg access only.

ih_fc_stat2 0x102

Bits	Name	Memory Access	Description
2	DRM	RW	Active after successful transmission of an DRM InfoFrame packet. Value After Reset: 0

ih_mute_fc_stat2 0x182

Bits	Name	Memory Access	Description
2	DRM	RW	When set to 1, mutes ih_fc_stat2[2]. Value After Reset: 1

fc_datauto3 0x10b7

Bits	Name	Memory Access	Description
6	DRM_auto	RW	Enables DRM packet insertion Value After Reset: 1

fc_rdrb12 0x10c4

Bits	Name	Memory Access	Description
7:4	Reserved.		
3:0	DRMframeinterpolation	RW	DRM frame interpolation Value After Reset: 0

fc_rdrb13 0x10c5

Bits	Name	Memory Access	Description
7:4	DRMpacketsinframe	RW	DRM packets per frame Value After Reset: 0

3:0	DRMpacketlinespacing	RW	DRM packets line spacing Value After Reset: 0

fc_mask2 0x10da

Bits	Name	Memory Access	Description
2	DRM	RW	Mask bit for FC_INT2.DRM interrupt bit Value After Reset: 1

fc_packet_tx_en 0x10e3

Bits	Name	Memory Access	Description
7	DRM_tx_en	RW	DRM transmission control 1: Transmission enabled 0: Transmission disabled Value After Reset: 0

fc_drm_hb01 0x1168

Bits	Name	Memory Access	Description
7:0	fc_drm_hb0	RW	Frame composer DRM Packet Header Register 1 Value After Reset: 0

fc_drm_hb02 0x1169

Bits	Name	Memory Access	Description
7:0	fc_drm_hb1	RW	Frame composer DRM Packet Header Register 2 Value After Reset: 0

fc_drm_pb[0:26] 0x116a+(i*0x1)

Bits	Name	Memory Access	Description
7:0	fc_drm_pb	RW	Frame composer DRM Packet Body Register Array Value After Reset: 0

27.5 CVBS

S905X supports CVBS 480i/576i standard definition output.

Register Definition

Video Encoder Registers

ENCI_VIDEO_MODE 0x1b00

Bit(s)	Field Name	R/W	Default	Description
7	ENCI_VIDEO_MODE_PAL_LN309_AVON	R/W	0	Fix the bug pal video is not on some fields of ln309
6	ENCI_VIDEO_MODE_SQPX	R/W	0	Square pixel mode
5-4	ENCI_VIDEO_MODE_FSCSEL	R/W	0	0: Fc=3.5795M, 1:Fsc=4.4336M, 2: Fsc=3.5756M, 3:Fsc=3.582M.
3	ENCI_VIDEO_MODE_PED	R/W	0	0: No pedestal, 1: Pedestal enable
2	ENCI_VIDEO_MODE_PDRST	R/W	0	0: Phase reset every field, 1: Phase not reset
1	ENCI_VIDEO_MODE_PHALT	R/W	0	0: Phase alternate disable, 1: Phase alternate enable
0	ENCI_VIDEO_MODE_LNFMT	R/W	0	0: 525 lines, 1:625 lines

ENCI_VIDEO_MODE_ADV 0x1b01

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_VIDEO_MODE_EN_CSYNC_MACV	R/W	0	Enable macrovision signal on CSYNC
14	CFG_LUMA_FOR_CVBS_HIGH	R/W	0	Luma for CVBS high bandwidth
13	CFG_CHROMA_MOD_HIGH	R/W	0	Chroma modulation high bandwidth
12	CFG_LUMA_FOR_RGB_HIGH	R/W	0	Using high bandwidth luma for RGB
11-10	ENCI_VIDEO_MODE_ADV_HY_SYLPF_SEL	R/W	0	Low pass filter selection on sync of high bandwidth Luma signal. 0=Bypass LPF; 1=Use 3-tap LPF; 2=Use 5-tap LPF.
9 -8	ENCI_VIDEO_MODE_ADV_CMPT_BURST_WIN_SEL	R/W	0	
7 -6	ENCI_VIDEO_MODE_ADV_CBW	R/W	0	Chroma Filter bandwidth, 0:Low, 1:Medium, 2:Higher
5 -4	ENCI_VIDEO_MODE_ADV_YBW	R/W	0	Luma Filter bandwidth , 0:Medium, 1:Low, 2:High
2	ENCI_VIDEO_MODE_ADV_VBICTL	R/W	1	0: Blank line end at line6 1: Blank line end at line17/22
1 -0	ENCI_VIDEO_MODE_ADV_DMXMD	R/W	2'b10	Video input demux shifting mode, adjust this value When input video stream is shifted.

ENCI_VIDEO_FSC_ADJ 0x1b02

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VIDEO_FSC_ADJ	R/W	16'd0	FSC adjust , 16 bit value that can adjust FSC frequency (Add with setting value)

ENCI_VIDEO_BRIGHT 0x1b03

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_VIDEO_BRIGHT	R/W	8'd0	Brightness

ENCI_VIDEO_CONT 0x1b04

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_VIDEO_CONT	R/W	8'd0	Contrast

ENCI_VIDEO_SAT 0x1b05

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_VIDEO_SAT	R/W	8'd0	Saturation

ENCI_VIDEO_HUE 0x1b06

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_VIDEO_HUE	R/W	8'd0	Hue

ENCI_VIDEO_SCH 0x1b07

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_VIDEO_SCH	R/W	8'd0	Sch adjust, adjust the phase of the FSC

ENCI_SYNC_MODE 0x1b08

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_SYNC_MODE	R/W	8'd7	Video input Synchronization mode Define Field sync/Vertical Sync/Horizontal Sync source, 0: FFF All from fsi (Not used in T25) 1: FVH From FSI, VSI, HSI respectively 2: FVV From FSI, VSI, VSI respectively 3: 0VV No FSI, Vsync/Hsync from VSI, HSI respectively 4: FVH_EAV From CCIR601 stream directly 5: FFF_EAV From CCIR601 stream, but use Field sync only 6: FMVH field rst when vs is near around line start. 7: MASTER Master mode, free run, send HSO/VSO out

Notes: Suggest use Master mode only

ENCI_SYNC_HSO_BEGIN 0x1b09

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCI_SYNC_HSO_BEGIN	R/W	11'd3	HSO begin position

ENCI_SYNC_HSO_END 0x1b0a

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCI_SYNC_HSO_END	R/W	11'd5	HSO end postion

ENCI_SYNC_VSO_EVNLN 0x1b0b

Bit(s)	Field Name	R/W	Default	Description
15 -8	ENCI_SYNC_VSO_EVN_STRTLN	R/W	8	VSO output start line in even field
7 -0	ENCI_SYNC_VSO_EVN_ENDLN	R/W	8	VSO output end line in even field

ENCI_SYNC_VSO_ODDLN 0x1b0e

Bit(s)	Field Name	R/W	Default	Description
15 -8	ENCI_SYNC_VSO_ODD_STRTLN	R/W	9	VSO output start line in odd field
7 -0	ENCI_SYNC_VSO_ODD_ENDLN	R/W	9	VSO output end line in odd field

ENCI_SYNC_CTRL 0x1b0f

Bit(s)	Field Name	R/W	Default	Description
8	ENCI_SYNC_DE_V	R/W	0	
7	ENCI_SYNC_VSO_INACTIVE_MODE	R/W	0	
6	ENCI_SYNC_CTRL_VSOMD	R/W	0	VSO position in odd field, 0:Half line after VSO in Even field, 1:Same as Even Field
5	ENCI_SYNC_CTRL_FSOP	R/W	0	FSO polarity
4	ENCI_SYNC_CTRL_VSOP	R/W	0	VSO polarity
3	ENCI_SYNC_CTRL_HSOP	R/W	0	HSO polarity
2	ENCI_SYNC_CTRL_FSIP	R/W	1	FSI polarity
1	ENCI_SYNC_CTRL_VSIP	R/W	0	VSI polarity
0	ENCI_SYNC_CTRL_HSIP	R/W	0	HSI polarity

ENCI_SYNC_HOFFST 0x1b10

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCI_SYNC_HOFFST	R/W	2	Horizontal offset after HSI in slave mode

ENCI_SYNC_VOFFST 0x1b11

Bit(s)	Field Name	R/W	Default	Description
8 -0	ENCI_SYNC_VOFFST	R/W	0	Horizontal offset aftet VSI in slave mode

ENCI_SYNC_ADJ 0x1b12

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_SYNC_ADJ_EN	R/W	0	Analog Synchronization and color burst value adjust enable
14 -10	ENCI_SYNC_ADJ_BU_V	R/W	0	Analog V burst adjustment, Signed value
9 -5	ENCI_SYNC_ADJ_BU_U	R/W	0	Analog U burst adjustment, Signed value
4 -0	ENCI_SYNC_ADJ_SYNC	R/W	0	Analog sync adjustment, Signed value

ENCI_RGB_SETTING 0x1b13

Bit(s)	Field Name	R/W	Default	Description
12 -3	RESERVED	R/W	0	Resevered
2 -0	ENCI_RGB_SYNC	R/W	0	Analog RGB sync disable (1:Enable Sync, 0:Disable) , bit 2 for Blue, bit 1 for Green. Bit 0 for Red

ENCI_CMPN_MATRIX_CB 0x1b14

Bit(s)	Field Name	R/W	Default	Description
15:8	ENCP_VIDEO_MATRIX_MA	R/W	0	CbCr matrix parameter Ma (signed value)
7:0	ENCP_VIDEO_MATRIX_MB	R/W	0	CbCr matrix parameter Mb (signed value)

Note: These 2 register is for HUE adjustment. The formula is : Cb' = Cb*Ma + Cr*Mb, Cr' = Cb*Mc + Cr*Md
(effective only when Enable HUE matrix by bit ENCI_VIDEO_MODE_ADV[11])

ENCI_CMPN_MATRIX_CR 0x1b15

Bit(s)	Field Name	R/W	Default	Description
15:8	ENCP_VIDEO_MATRIX_MC	R/W	0	CbCr matrix parameter Mc (signed value)
7:0	ENCP_VIDEO_MATRIX_MD	R/W	0	CbCr matrix parameter Md (signed value)

Note: See above.

ENCI_VBI_SETTING 0x1b20

Bit(s)	Field Name	R/W	Default	Description
13	ENCI_VBI_SETTING_CCCSTM	R/W	0	Close caption phase adjustment (Value is from TTXDT0)
12	ENCI_VBI_SETTING_TTX_ODD_EN	R/W	0	Teletext odd field enable
11	ENCI_VBI_SETTING_TTX_EVN_EN	R/W	0	Teletext even field enable
10 -8	ENCI_VBI_SETTING_TTX_FREQ	R/W	0	Teletext frequency selection
5	ENCI_VBI_SETTING_CGMS_ODD_EN	R/W	0	CGMS odd field enable
4	ENCI_VBI_SETTING_CGMS_EVN_EN	R/W	0	CGMS even field enable
3	ENCI_VBI_SETTING_WSS_ODD_EN	R/W	0	WSS odd field enable
2	ENCI_VBI_SETTING_WSS_EVN_EN	R/W	0	WSS Even Field enable
1	ENCI_VBI_SETTING_CC_ODD_EN	R/W	0	Close Caption Odd Field enable
0	ENCI_VBI_SETTING_CC_EVN_EN	R/W	0	Close Caption Even Field enable

ENCI_VBI_CCDT_EVN 0x1b21

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_CCDT_EVN	R/W	0	Close caption even field data

ENCI_VBI_CCDT_ODD 0x1b22

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_CCDT_ODD	R/W	0	Close caption even odd data

ENCI_VBI_CC525_LN 0x1b23

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_CC525_LN	R/W	16'h1211	Close caption line in 525 format

ENCI_VBI_CC625_LN 0x1b24

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_CC625_LN	R/W	16'h1615	Close caption line in 625 format

ENCI_VBI_WSSDT 0x1b25

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_WSSDT	R/W	16'h0	WSS data

ENCI_VBI_WSS_LN 0x1b26

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_WSS_LN	R/W	16'd22	WSS line

ENCI_VBI_CGMSDT_L 0x1b27

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_CGMSDT_L	R/W	16'h0	CGMS data low 16 Bits

ENCI_VBI_CGMSDT_H 0x1b28

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_VBI_CGMSDT_H	R/W	8'h0	CGMS data high 8 Bits

ENCI_VBI_CGMS_LN 0x1b29

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_CGMS_LN	R/W	16'h1110	CGMS line

ENCI_VBI_TTX_HTIME 0x1b2a

Bit(s)	Field Name	R/W	Default	Description
15 -10	ENCI_VBI_TTX_HTIME_TOTAL_BYTES	R/W	6'h2d	Teletext total bytes
9 -0	ENCI_VBI_TTX_HTIME_STRT_POS	R/W	11'h135	Teletext start postion

ENCI_VBI_TTX_LN 0x1b2b

Bit(s)	Field Name	R/W	Default	Description
15 -9	ENCI_VBI_TTX_STRT_LN	R/W	16'h1415	Teletext Start line
8 -0	ENCI_VBI_TTX_END_LN	R/W	16'h1415	Teletext End Line

ENCI_VBI_TTXDT0 0x1b2c

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_TTXDT0	R/W	16'h0	Teletext data0

ENCI_VBI_TTXDT1 0x1b2d

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_TTXDT1	R/W	16'h0	Teletext data1

ENCI_VBI_TTXDT2 0x1b2e

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_TTXDT2	R/W	16'h0	Teletext data2

ENCI_VBI_TTXDT3 0x1b2f

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_VBI_TTXDT3	R/W	16'h0	Teletext data3

ENCI_MACV_N0 0x1b30

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_MACV_N0	R/W	8'h00	Macrovision register N0

ENCI_MACV_N1 0x1b31

Bit(s)	Field Name	R/W	Default	Description
5 -0	ENCI_MACV_N1	R/W	6'h17	Macrovision register N1

ENCI_MACV_N2 0x1b32

Bit(s)	Field Name	R/W	Default	Description
5 -0	ENCI_MACV_N2	R/W	6'h15	Macrovision register N2

ENCI_MACV_N3 0x1b33

Bit(s)	Field Name	R/W	Default	Description
5 -0	ENCI_MACV_N3	R/W	6'h21	Macrovision register N3

ENCI_MACV_N4 0x1b34

Bit(s)	Field Name	R/W	Default	Description
5 -0	ENCI_MACV_N4	R/W	6'h15	Macrovision register N4

ENCI_MACV_N5 0x1b35

Bit(s)	Field Name	R/W	Default	Description
2 -0	ENCI_MACV_N5	R/W	3'h5	Macrovision register N5

ENCI_MACV_N6 0x1b36

Bit(s)	Field Name	R/W	Default	Description
2 -0	ENCI_MACV_N6	R/W	3'h5	Macrovision register N6

ENCI_MACV_N7 0x1b37

Bit(s)	Field Name	R/W	Default	Description
1 -0	ENCI_MACV_N7	R/W	2'h2	Macrovision register N7

ENCI_MACV_N8 0x1b38

Bit(s)	Field Name	R/W	Default	Description
5 -0	ENCI_MACV_N8	R/W	6'h1b	Macrovision register N8

ENCI_MACV_N9 0x1b39

Bit(s)	Field Name	R/W	Default	Description
5 -0	ENCI_MACV_N9	R/W	6'h1b	Macrovision register N9

ENCI_MACV_N10 0x1b3a

Bit(s)	Field Name	R/W	Default	Description
5 -0	ENCI_MACV_N10	R/W	6'h24	Macrovision register N10

ENCI_MACV_N11 0x1b3b

Bit(s)	Field Name	R/W	Default	Description
14 -0	ENCI_MACV_N11	R/W	15'h07f8	Macrovision register N11

ENCI_MACV_N12 0x1b3c

Bit(s)	Field Name	R/W	Default	Description
14 -0	ENCI_MACV_N12	R/W	15'h0	Macrovision register N12

ENCI_MACV_N13 0x1b3d

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_MACV_N13	R/W	8'h0f	Macrovision register N13

ENCI_MACV_N14 0x1b3e

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_MACV_N14	R/W	8'h0f	Macrovision register N14

ENCI_MACV_N15 0x1b3f

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCI_MACV_N15	R/W	8'h60	Macrovision register N15

ENCI_MACV_N16 0x1b40

Bit(s)	Field Name	R/W	Default	Description
0 -0	ENCI_MACV_N16	R/W	1'h1	Macrovision register N16

ENCI_MACV_N17 0x1b41

Bit(s)	Field Name	R/W	Default	Description
3 -0	ENCI_MACV_N17	R/W	4'ha	Macrovision register N17

ENCI_MACV_N18 0x1b42

Bit(s)	Field Name	R/W	Default	Description
3 -0	ENCI_MACV_N18	R/W	4'h0	Macrovision register N18

ENCI_MACV_N19 0x1b43

Bit(s)	Field Name	R/W	Default	Description
3 -0	ENCI_MACV_N19	R/W	4'h5	Macrovision register N19

ENCI_MACV_N20 0x1b44

Bit(s)	Field Name	R/W	Default	Description
2 -0	ENCI_MACV_N20	R/W	3'h4	Macrovision register N20

ENCI_MACV_N21 0x1b45

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCI_MACV_N21	R/W	10'h3ff	Macrovision register N21

ENCI_MACV_N22 0x1b46

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCI_MACV_N22	R/W	16'h0	Macrovision register N22

ENCI_DBG_PX_RST 0x1b48

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_DBG_PX_RST_EN	R/W	1	1:Reset enable (Interlaced TV encoder is disabled), 0:Normal
10 -0	ENCI_DBG_PX_RST_VAL	R/W	0	Pixel value after reset

ENCI_DBG_FLDLN_RST 0x1b49

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_DBG_LN_RST_EN	R/W	0	Line reset enable
14	ENCI_DBG_FLD_RST_EN	R/W	0	Field reset enable
11 -9	ENCI_DBG_FLD_RST_VAL	R/W	0	Field reset value
8 -0	ENCI_DBG_LN_RST_VAL	R/W	0	Line reset value

ENCI_DBG_PX_INT 0x1b4a

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_DBG_PX_INT_EN	R/W	0	Pixel Interrupt enable
10 -0	ENCI_DBG_PX_INT_VAL	R/W	0	Pixel value that trig the interrupt

ENCI_DBG_FLDLN_INT 0x1b4b

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_DBG_FLD_INT_EN	R/W	0	Field interrupt enable
11	ENCI_DBG_LN_INT_EN	R/W	0	Line interrupt enable
14 -12	ENCI_DBG_FLD_INT_VAL	R/W	0	Field value that trig the interrupt
8 -0	ENCI_DBG_LN_INT_VAL	R/W	0	Line value that trig the interrupt

ENCI_DBG_MAXPX 0x1b4c

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_DBG_MAXPX_EN	R/W	0	Debug mode, change the max pixel.
10 -0	ENCI_DBG_MAXPX_CHGVAL	R/W	0	The value of max pixel

ENCI_DBG_MAXLN 0x1b4d

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_DBG_MAXLN_EN	R/W	0	Debug mode, change the max line counter
8 -0	ENCI_DBG_MAXLN_CHGVAL	R/W	0	The value want to change

ENCI_MACV_MAX_AMP 0x1b50

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_MACV_MAX_AMP_EN	R/W	0	Macrovision max amplitude change enable
10 -0	ENCI_MACV_MAX_AMP_VAL	R/W	0	The value want to change

ENCI_MACV_PULSE_LO 0x1b51

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_MACV_PULSE_LO_EN	R/W	0	Macrovision low pulse change enable.
10 -0	ENCI_MACV_PULSE_LO_VAL	R/W	0	The value want to change

ENCI_MACV_PULSE_HI 0x1b52

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_MACV_PULSE_HI_EN	R/W	0	Macrovision high pulse change enable.
10 -0	ENCI_MACV_PULSE_HI_VAL	R/W	0	The value want to change

ENCI_MACV_BKP_MAX 0x1b53

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_MACV_BKP_MAX_EN	R/W	0	Macrovision back porch max value change enable.
9 -0	ENCI_MACV_BKP_MAX_VAL	R/W	0	The value want to change

ENCI_CFILT_CTRL 0x1b54

Bit(s)	Field Name	R/W	Default	Description
12	ENCI_CFILT_CMPT_SMOOTH_FLT	R/W	1	Component Cb/Cr smooth filter
11-10	ENCI_CFILT_CMPT_SEL	R/W	0	Filter component bandwith Cb/Cr sel 0: High, 1: Medium 2: Low
1	ENCI_CFILT_CMPT_SEL	R/W	1	Filter component CbCr bandwith sel (1:high, 0:low)
0	ENCI_CFILT_CVBS_SEL	R/W	0	Filter CVBS CbCr bandwith sel (1:high, 0:low)

ENCI_CFILT7 0x1b55

Bit(s)	Field Name	R/W	Default	Description
14-0	ENCI_CFILT7	R/W	15'h4d53	Filter7 parameters

ENCI_YC_DELAY 0x1b56

Bit(s)	Field Name	R/W	Default	Description
9-8	ENCI_COMPY_DELAY	R/W	0	Component Y delay
7-4	ENCI_Y_DELAY	R/W	4'h2	Interlace Y delay
3-0	ENCI_C_DELAY	R/W	4'h0	Interlace C delay

ENCI_VIDEO_EN 0x1b57

Bit(s)	Field Name	R/W	Default	Description
0	ENCI_VIDEO_EN	R/W	0	Interlace video enable

ENCP_VFIFO2VD_CTL 0x1b58

Bit(s)	Field Name	R/W	Default	Description
15-8	VFIFO2VD_VD_SEL	R/W	0	vfifo2vd_vd_sel
7	VFIFO2VD_DROP	R/W	0	vfifo2vd_drop
6-1	VFIFO2VD_DELAY	R/W	0	vfifo2vd_delay
0	VFIFO2VD_EN	R/W	0	vfifo2vd_en

ENCP_VFIFO2VD_PIXEL_START 0x1b59

Bit(s)	Field Name	R/W	Default	Description
12-0	VFIFO2VD_PIXEL_START	R/W	0	Pixel start

ENCP_VFIFO2VD_PIXEL_END 0x1b5a

Bit(s)	Field Name	R/W	Default	Description
15	VFIFO2VD_YC_USE_FIRST_VFIFO_REQ	R/W	0	0=Keep as previous version; 1=On the very first pixel since VFIFO is enabled, YC pipeline values take from default setting, rather than from values stored in the pipeline.
12-0	VFIFO2VD_PIXEL_END	R/W	0	Pixel end

ENCP_VFIFO2VD_LINE_TOP_START 0x1b5b

Bit(s)	Field Name	R/W	Default	Description
10-0	VFIFO2VD_LINE_TOP_START	R/W	0	Top field line end

ENCP_VFIFO2VD_LINE_TOP_END 0x1b5c

Bit(s)	Field Name	R/W	Default	Description
10-0	VFIFO2VD_LINE_TOP_END	R/W	0	Top field line end

ENCP_VFIFO2VD_LINE_BOT_START 0x1b5d

Bit(s)	Field Name	R/W	Default	Description
10-0	VFIFO2VD_LINE_BOT_START	R/W	0	Bottom field line end

ENCP_VFIFO2VD_LINE_BOT_END 0x1b5e

Bit(s)	Field Name	R/W	Default	Description
10-0	VFIFO2VD_LINE_BOT_END	R/W	0	Bottom field line end

VENC_SYNC_ROUTE 0x1b60

Bit(s)	Field Name	R/W	Default	Description
2	VENC_SYNC_ROUTE	R/W	0	Internal Vencoder hsync/vsync source (0-VIU, 1-External Venc)
1	VIU_SYNC_ROUTE	R/W	0	VIU hsync/vsync input source (0-Venc, 1-External Venc)
0	VPINS_SYNC_ROUTE	R/W	1	External Vencoder hsync/vsync source (0-Internal Venc, 1-VIU)

VENC_VIDEO_EXSRC 0x1b61

Bit(s)	Field Name	R/W	Default	Description
0	VENC_VIDEO_EXSRC	R/W	1	External Video Source enable, get video data from external pins.

Notes: This is not supported by T25, clear this when init.

VENC_DVI_SETTING 0x1b62

Bit(s)	Field Name	R/W	Default	Description
15	VENC_DVI_SEL_DVI	R/W	0	1=use programmable vs/hs/de whose only connection is to DVI/HDMI.
14	VENC_DVI_SEL_INTL_DE	R/W	0	For interlace mode only. 1=use intl_de for DE.
13	VENC_DVI_GAMMA_EN	R/W	0	0=Send to DVI/HDMI interface with data NOT from Gamma table. 1=Send to DVI/HDMI interface with data from Gamma table.
12	VENC_DVI_DDR_DESEL	R/W	0	For external DVI device only.
11	VENC_DVI_DDR_CKSEL	R/W	0	For external DVI device only.
10	VENC_DVI_DDR_CKPHI	R/W	0	For external DVI device only.
9	VENC_DVI_DDR_SEL	R/W	0	For external DVI device only.
8	VENC_DVI_INV_CLK	R/W	0	Invert DVI clock
7	VENC_DVI_SEL_INTERNAL_HDMI	R/W	0	0=Select the external DVI device which is compatible to chips predate M1

Bit(s)	Field Name	R/W	Default	Description
				1=Select the internal HDMI tx which is in chips starting from meson.
6:4	VENC_DVI_CLK_SEL	R/W	0	DVI Clock selection: 0: clk27 1: clk54 2: ½ of ENCP pixel_clk 3: clk108 4: ½ of ENCI pixel_clk 5: cts_vclk1 6-7: Reserved (Note : here the clk27/54/108 is signal Name, not the real frequency)
3	VENC_DVI_INV_VSYNC	R/W	0	1=Invert DVI/HDMI Vsync output polarity
2	VENC_DVI_INV_HSYNC	R/W	0	1=Invert DVI/HDMI Hsync output polarity
1	SYNC_SEL_OUT_SYNC	R/W	0	DVI/601 sync use output sync (not viu sync)
0	VENC_DVI_SRC_SEL	R/W	1	DVI source select (0: interlace, 1:progressive)

VENC_C656_CTRL **0x1b63**

Bit(s)	Field Name	R/W	Default	Description
3	VENC_C656_CLKP	R/W	0	Clock out Polarity
2	VENC_C656_FLDP	R/W	0	Field Polarity
1	VENC_C656_VSOP	R/W	0	Vsyn Polarity
0	VENC_C656_HSOP	R/W	0	Hsyn Polarity

Upsamping is only used for 480i/576i, 480p/576p, not for 720p/1080i/1080p mode,
For 480i/576i upsampling, video signal is upsampled from 27Mhz to 108Mhz,

For 480p/576p upsampling, video signal is upsampled from 54Mz to 108Mhz
set VENC_UPSAMPLE_CTRL0 = 0x9061
set VENC_UPSAMPLE_CTRL1 = 0xa061
set VENC_UPSAMPLE_CTRL2 = 0x1b061

VENC_UPSAMPLE_CTRL0 **0x1b64**

Bit(s)	Field Name	R/W	Default	Description
15-12	UPSAMPLE_SEL	R/W	0	Upsample0 din selection 0: Interlace High bandwidth Luma 1: CVBS 2: S-Video luma 3: S-Video Chroma 4: Interlace Pb 5: Interlace Pr 6: Interlace R 7: Interlace G 8: Interlace B 9: Progressive Y a: Progressive Pb b: Progressive Pr c: Progressive R d: Progressive G e: Progressive B f: VDAC test value (VENC_VDAC_TST_VAL)
11-9	UNUSED	R	0	Unused
8	BYPASS	R/W	0	If true, bypass
7	F1_DIV_INV	R/W	0	If true, invert div
6	F1_UPSAMPLE_EN	R/W	0	Filter1 upsample enable
5	F1_EN	R/W	0	Filter1 filtering enable

Bit(s)	Field Name	R/W	Default	Description
4	F1_CLK_RATIO	R/W	0	If true, Filter1 clk ratio is 2, input data sample every 2 clocks, otherwise 1
3	F0_DIV_INV	R/W	0	If true, invert div
2	F0_UPSAMPLE_EN	R/W	0	Filter0 upsample enable
1	F0_EN	R/W	0	Filter0 filtering enable
0	F0_CLK_RATIO	R/W	0	If true, Filter0 clk ratio is 2, input data sample every 2 clocks, otherwise 1

VENC_UPSAMPLE_CTRL1 0x1b65

Bit(s)	Field Name	R/W	Default	Description
15-12	UPSAMPLE_SEL	R/W	0	Upsample1 din selection 0: Interlace High bandwidth Luma 1: CVBS 2: S-Video luma 3: S-Video Chroma 4: Interlace Pb 5: Interlace Pr 6: Interlace R 7: Interlace G 8: Interlace B 9: Progressive Y a: Progressive Pb b: Progressive Pr c: Progressive R d: Progressive G e: Progressive B f: VDAC test value (VENC_VDAC_TST_VAL)
11-9	UNUSED	R	0	Unused
8	BYPASS	R/W	0	If true, bypass
7	F1_DIV_INV	R/W	0	If true, invert div
6	F1_UPSAMPLE_EN	R/W	0	Filter1 upsample enable
5	F1_EN	R/W	0	Filter1 filtering enable
4	F1_CLK_RATIO	R/W	0	If true, Filter1 clk ratio is 2, input data sample every 2 clocks, otherwise 1
3	F0_DIV_INV	R/W	0	If true, invert div
2	F0_UPSAMPLE_EN	R/W	0	Filter0 upsample enable
1	F0_EN	R/W	0	Filter0 filtering enable
0	F0_CLK_RATIO	R/W	0	If true, Filter0 clk ratio is 2, input data sample every 2 clocks, otherwise 1

VENC_UPSAMPLE_CTRL2 0x1b66

Bit(s)	Field Name	R/W	Default	Description
15-12	UPSAMPLE_SEL	R/W	0	Upsample2 din selection 0: Interlace High bandwidth Luma 1: CVBS 2: S-Video luma 3: S-Video Chroma 4: Interlace Pb 5: Interlace Pr 6: Interlace R 7: Interlace G 8: Interlace B 9: Progressive Y a: Progressive Pb b: Progressive Pr c: Progressive R d: Progressive G e: Progressive B f: VDAC test value (VENC_VDAC_TST_VAL)

Bit(s)	Field Name	R/W	Default	Description
11-9	UNUSED	R	0	Unused
8	BYPASS	R/W	0	If true, bypass
7	F1_DIV_INV	R/W	0	If true, invert div
6	F1_UPSAMPLE_EN	R/W	0	Filter1 upsample enable
5	F1_EN	R/W	0	Filter1 filtering enable
4	F1_CLK_RATIO	R/W	0	If true, Filter1 clk ratio is 2, input data sample every 2 clocks, otherwise 1
3	F0_DIV_INV	R/W	0	If true, invert div
2	F0_UPSAMPLE_EN	R/W	0	Filter0 upsample enable
1	F0_EN	R/W	0	Filter0 filtering enable
0	F0_CLK_RATIO	R/W	0	If true, Filter0 clk ratio is 2, input data sample every 2 clocks, otherwise 1

VENC_VIDEO_PROG_MODE 0x1b68

Bit(s)	Field Name	R/W	Default	Description
9	VENC_DAC_CLK_SEL_CLK108	R/W	1'b0	Select clk108 as DAC clock
8	VENC_BIST_FIELD_SEL	R/W	1'b0	Venc bist and venc_field output selection. 0:Interlace, 1:Progressive.
7	VENC_VIU_VSIN_SEL	R/W	1'b0	VIU Vsync input select, 0:Progressive, 1:Interlace
6	VENC_VIU_HSIN_SEL	R/W	1'b0	VIU Hsync input select, 0:Progressive, 1:Interlace
5	VENC_VPIN_VSIN_SEL	R/W	1'b0	Pin Vsync input select, 0:Progressive, 1:Interlace
4	VENC_VPIN_HSIN_SEL	R/W	1'b0	Pin Hsync input select, 0:Progressive, 1:Interlace
3	VENC_DAC1_CLK_SEL	R/W	1'b0	DAC1 clock select, 0:clk54, 1:clk27
2	VENC_DAC0_CLK_SEL	R/W	1'b0	DAC0 clock select, 0:clk54, 1:clk27

VENC_ENCI_LINE 0x1b69

Bit(s)	Field Name	R/W	Default	Description
9 -0	VENC_ENCI_LINE	R	-	Current interlace encoder line

VENC_ENCI_PIXEL 0x1b6a

Bit(s)	Field Name	R/W	Default	Description
9 -0	VENC_ENCI_LINE	R	-	Current interlace encoder line

VENC_ENCP_LINE 0x1b6b

Bit(s)	Field Name	R/W	Default	Description
9 -0	VENC_ENCP_LINE	R	-	Current interlace encoder line

VENC_ENCP_PIXEL 0x1b6c

Bit(s)	Field Name	R/W	Default	Description
9 -0	VENC_ENCP_LINE	R	-	Current interlace encoder line

VENC_STATA 0x1b6d

Bit(s)	Field Name	R/W	Default	Description
2 -0	VENC_FILED	R	-	Current Venc Field

VENC_INTCTRL 0x1b6e

Bit(s)	Field Name	R/W	Default	Description
13	ENCP_TIMER_INT_EN	R/W	0	
12	ENCP_FLDINT_EN	R/W	0	
11	ENCP_PXRST_EN	R/W	0	
10	ENCP_TIMER_INT_EN	R/W	0	
9	ENCP_LNRST_INT_EN	R /W	0	Progressive encoder filed change interrupt enable
8	ENCP_PXRST_INT_EN	R /W	0	Progressive encoder line change interrupt enable
7	ENCL_TIMER_INT_EN	R/W	0	
6	ENCL_FLDINT_EN	R/W	0	
5	ENCL_PXRST_EN	R/W	0	

Bit(s)	Field Name	R/W	Default	Description
4	ENCI_TIMER_INT_EN	R/W	0	
3	ENCI_TTXLOAD_INT_EN	R /W	0	Interlace encoder teletext data interrupt enable
2	ENCI_FDRST_INT_EN	R /W	0	Interlace encoder field reset interrupt enable
1	ENCI_LNRST_INT_EN	R /W	0	Interlace encoder filed change interrupt enable
0	ENCI_PXRST_INT_EN	R /W	0	Interlace encoder line change interrupt enable

VENC_INTFLAG 0x1b6f

Bit(s)	Field Name	R/W	Default	Description
9	ENCP_LNRST_INT_F	R /W	0	Progressive encoder filed change interrupt flag
8	ENCP_PXRST_INT_F	R /W	0	Progressive encoder line change interrupt flag
3	ENCI_TTXLOAD_INT_F	R /W	0	Interlace encoder teletext data load interrupt flag
2	ENCI_FDRST_INT_F	R /W	0	Interlace encoder field reset interrupt flag
1	ENCI_LNRST_INT_F	R /W	0	Interlace encoder filed change interrupt flag
0	ENCI_PXRST_INT_F	R /W	0	Interlace encoder line change interrupt flag

VENC_VIDEO_TST_EN 0x1b70

Bit(s)	Field Name	R/W	Default	Description
0	VENC_VIDEO_TST_EN	R/W	0	BIST enable

VENC_VIDEO_TST_MDSEL 0x1b71

Bit(s)	Field Name	R/W	Default	Description
7-0	VENC_VIDEO_TST_MDSEL	R/W	8'h01	BIST Mode 0: TST_MODE_FIXVAL---- Fix Value on Y/Cb/Cr 1: TST_MODE_COLORBAR ---- 100/75 color bar 2: TST_MODE_THINLINE ---- Thin horizonl/vertical lines on screen 3: TST_MODE_DOTGRID ---- Dot grid on screen

Notes: In none color bar mode, value is from register TST_Y/TST_CB/TST_CR.

VENC_VIDEO_TST_Y 0x1b72

Bit(s)	Field Name	R/W	Default	Description
9 -0	VENC_VIDEO_TST_Y	R/W	10'd512	BIST fix value Y

Notes: Value is 10 bists,X4 if convert from 8 Bits CCIR601 value

VENC_VIDEO_TST_CB 0x1b73

Bit(s)	Field Name	R/W	Default	Description
9 -0	VENC_VIDEO_TST_CB	R/W	10'd512	BIST fix value Cb

Notes: Value is 10 bists,X4 if convert from 8 Bits CCIR601 value

VENC_VIDEO_TST_CR 0x1b74

Bit(s)	Field Name	R/W	Default	Description
9 -0	VENC_VIDEO_TST_CR	R/W	10'd512	BIST fix value Cr

Notes: Value is 10 bists,X4 if convert from 8 Bits CCIR601 value

VENC_VIDEO_TST_CLRBAR_STRT 0x1b75

Bit(s)	Field Name	R/W	Default	Description
12-0	VENC_VIDEO_TST_CLRBAR_STRT	R/W	11'd275	Colorbar BIST start position

Notes: VENC_VIDEO_TST_CLRBAR_STRT Interlace mode : 235, Progressive mode : 274

VENC_VIDEO_TST_CLRBAR_WIDTH 0x1b76

Bit(s)	Field Name	R/W	Default	Description
12 -0	VENC_VIDEO_TST_CLRBAR_WIDTH	R/W	11'd360	Colorbar BIST Width

Notes: VENC_VIDEO_TST_CLRBAR_WIDTH Interlace mode : 360, Progressive mode: 180

VENC_VIDEO_TST_VDCNT_STSET 0x1b77

Bit(s)	Field Name	R/W	Default	Description
1-0	VENC_VIDEO_TST_VDCNT_STSET	R/W	0	BIST video data shifting setting, adjust this value if the data sequence generated by BIST is incorrect.

VENC_VDAC_DACSEL0 0x1b78

Bit(s)	Field Name	R/W	Default	Description
15-12	VENC_VDAC_DLY	R/W	0	VDAC delay (0-15 clocks)
5	VENC_VDAC_SEL_ATV_DMD	R/W	0	1: select VDAC0 source from ATV demod
4-0	VENC_VDAC_DACSEL0	R/W	4'h2	Video Dac0 selection 0: Interlace Y 1: CVBS 2: S-Video luma 3: S-Video Chroma 4: Interlace Pb 5: Interlace Pr 6: Interlace R 7: Interlace G 8: Interlace B 9: Progressive Y a: Progressive Pb b: Progressive Pr c: Progressive R d: Progressive G e: Progressive B f: VDAC test value (VENC_VDAC_TST_VAL) 0x10: upsampled data0 0x11: upsampled data1 0x12 upsampled data2

VENC_VDAC_DACSEL1 0x1b79

Bit(s)	Field Name	R/W	Default	Description
			16'h4	Same as above

VENC_VDAC_DACSEL2 0x1b7a

Bit(s)	Field Name	R/W	Default	Description
			16'h5	Same as above

VENC_VDAC_DACSEL3 0x1b7b

Bit(s)	Field Name	R/W	Default	Description
			16'h1	Same as above

VENC_VDAC_DACSEL4 0x1b7c

Bit(s)	Field Name	R/W	Default	Description
			16'h2	Same as above

VENC_VDAC_DACSEL5 0x1b7d

Bit(s)	Field Name	R/W	Default	Description
			16'h3	Same as above

VENC_VDAC_SETTING 0x1b7e

Bit(s)	Field Name	R/W	Default	Description
0	VENC_VDAC0_PWDN	R/W	0	Powers down video DAC 0
1	VENC_VDAC1_PWDN	R/W	0	Powers down video DAC 1
2	VENC_VDAC2_PWDN	R/W	0	Powers down video DAC 2

Bit(s)	Field Name	R/W	Default	Description
3	VENC_VDAC3_PWDN	R/W	0	Powers down video DAC 3
4	VENC_VDAC4_PWDN	R/W	0	Powers down video DAC 4
5	VENC_VDAC5_PWDN	R/W	0	Powers down video DAC 5
6		R/W	0	Unused
7	VENC_VDAC_ALL_PWDN	R/W	0	Powers down all video DACs
8	VENC_VDAC_SOG	R/W	0	VDAC SOG signal (Sync on green, not used)
9	VENC_VDAC_IREN	R/W	0	VDAC IREN signal (7.5 IRE setup, not used)
12	VENC_VDAC_SYNC	R/W	0	VDAC sync signal when SYBL_en is 0
13	VENC_VDAC_BLNK	R/W	0	VDAC blank signal when SYBL_EN is 0
15	VENC_VDAC_SYBL_EN	R/W	0	Enable tv encoder to control VDAC sync and blank

VENC_VDAC_TST_VAL 0x1b7f

Bit(s)	Field Name	R/W	Default	Description
9 -0	VENC_VDAC_TST_VAL	R/W	10'h200	Video DAC test value

ENCP_VIDEO_EN 0x1b80

Bit(s)	Field Name	R/W	Default	Description
0	ENCP_VIDEO_EN	R/W	0	Progressive encoder enable

ENCP_VIDEO_SYNC_MODE 0x1b81

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCP_VIDEO_SYNC_MODE	R/W	8'h7	Video input Synchronization mode (4:Slave mode, 7:Master mode)

ENCP_MACV_EN 0x1b82

Bit(s)	Field Name	R/W	Default	Description
0	ENCP_MACV_EN	R/W	0	Macrovision enable

ENCP_VIDEO_Y_SCL 0x1b83

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCP_VIDEO_Y_SCL	R/W	8'd81	Y scale

ENCP_VIDEO_PB_SCL 0x1b84

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCP_VIDEO_PB_SCL	R/W	8'd79	Cb scale

ENCP_VIDEO_PR_SCL 0x1b85

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCP_VIDEO_PR_SCL	R/W	8'd79	Cr scale

ENCP_VIDEO_SYNC_SCL 0x1b86

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCP_VIDEO_SYNC_SCL	R/W	8'd128	Analog Sync value scale

ENCP_VIDEO_MACV_SCL 0x1b87

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCP_VIDEO_MACV_SCL	R/W	8'd128	Macrovision value scale

ENCP_VIDEO_Y_OFFST 0x1b88

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_VIDEO_Y_OFFST	R/W	10'h3c0	Y offset

ENCP_VIDEO_PB_OFFST 0x1b89

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_VIDEO_PB_OFFST	R/W	10'h0	Pb offset

ENCP_VIDEO_PR_OFFST 0x1b8a

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_VIDEO_PR_OFFST	R/W	10'h0	Pr offset

ENCP_VIDEO_SYNC_OFFST 0x1b8b

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_VIDEO_SYNC_OFFST	R/W	10'h0	Sync pulse offset

ENCP_VIDEO_MACV_OFFST 0x1b8c

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_VIDEO_MACV_OFFST	R/W	10'h0	Macrovision offset

ENCP_VIDEO_MODE 0x1b8d

Bit(s)	Field Name	R/W	Df lt	Description
15	ENCP_PX_LN_CNT_SHADOW_EN	R/W	0	pixel count and line count shadow enable
14	ENCP_VIDEO_MODE_DE_V	R/W	0	1=DE signal's polarity is active high. 0=DE signal's polarity is active low.
13	ENCP_VIDEO_MODE_EN_VSO_OFLD_FIX_EN	R/W	0	1=Extend ENCP_VIDEO_MAX_PXCNT by 1 pixel. 0=Treat ENCP_VIDEO_MAX_PXCNT as is.
12	ENCP_VIDEO_MODE_EN_FIELD_ODD	R/W	0	Enable odd field (for 1080i)
11	ENCP_VIDEO_MODE_EN_VSO_OFLD	R/W	0	Enable VSO odd field
10	ENCP_VIDEO _MODE_EN_VAVON_OFLD	R/W	0	Enable Vertical Active On for odd field
9	ENCP_VIDEO _MODE_EN_EQU_OFLD	R/W	0	Enable Equalization pulse for odd field
8	ENCP_VIDEO _MODE_EN_EQU	R/W	0	Enable Equalization pulse
6	ENCP_VIDEO _MODE_EN_HSEQ_SWITCH	R/W	0	Enable Hsync and equalization pulse switch in center
5	ENCP_VIDEO _MODE_EN_VP_OFLD	R/W	0	Enable Vertical Pulse for odd field
4	ENCP_VIDEO _MODE_EN_VP2	R/W	0	Enable 2nd vertical pulse in a line (1080i)
3	ENCP_VIDEO _MODE_EN_VPE_HALF_OFLD	R/W	0	Vertical pulse on end line of odd field is half
2	ENCP_VIDEO _MODE_EN_VPB_HALF_OFLD	R/W	0	Vertical pulse on begin line of odd field is half
1	ENCP_VIDEO _MODE_EN_VPE_HALF	R/W	0	Vertical pulse on end line of even field is half
0	ENCP_VIDEO _MODE_EN_VPB_HALF	R/W	0	Vertical pulse on begin line of even field is half
Note: This register should be set as 0x140 for 720p , and 0x1ffc for 1080i, and 0x0 for 480p/576p				

ENCP_VIDEO_MODE_ADV 0x1b8e

Bit(s)	Field Name	R/W	Default	Description
15-14	ENCP_SP_TIMING_CTRL	R/W	0	
13	ENCP_CR_BAPASS_LIM	R/W	0	
12	ENCP_CB_BAPASS_LIM	R/W	0	
11	ENCP_Y_BAPASS_LIM	R/W	0	
10	ENCP_SEL_GAMMA_RGB_IN	R/W	0	
9-8		R/W	0	Unused
7	ENCP_VIDEO_MODE_ADV_EN_HUE_MATRIX	R/W	0	Enable HUE matrix
6	ENCP_VIDEO_MODE_ADV_SWAP_PBPR	R/W	0	Swap PB PR
5	ENCP_VIDEO_MODE_ADV_EN_HS_PBPR	R/W	0	Enable sync pulse on PB PR
4	ENCP_VIDEO_MODE_ADV_GAIN_HDTV	R/W	0	YPBPR gain as HDTV type
3	ENCP_VIDEO_MODE_ADV_VFIFO_EN	R/W	0	Data input from VFIFO
2 :0	ENCP_VIDEO_MODE_ADV_VFIFO_UPMODE	R/W	1	Sampling rate: 0: 1 1: ½ 2: ¼ 3: 1/8

ENCP_DBG_PX_RST 0x1b90

Bit(s)	Field Name	R/W	Default	Description
15	ENCP_DBG_PX_RST_EN	R/W	0	Debug mode pixel reset enable
12 -0	ENCP_DBG_PX_RST_VAL	R/W	0	Debug mode pixel reset value

ENCP_DBG_LN_RST 0x1b91

Bit(s)	Field Name	R/W	Default	Description
15	ENCP_DBG_LN_RST_EN	R/W	0	Debug mode line reset enable
10 -0	ENCP_DBG_LN_RST_VAL	R/W	0	Debug mode line reset value

ENCP_DBG_PX_INT 0x1b92

Bit(s)	Field Name	R/W	Default	Description
15	ENCP_DBG_PX_INT_EN	R/W	0	Pixel Interrupt enable
12 -0	ENCP_DBG_PX_INT_VAL	R/W	0	Pixel value that trig the interrupt

ENCP_DBG_LN_INT 0x1b93

Bit(s)	Field Name	R/W	Default	Description
15	ENCP_DBG_LN_INT_EN	R/W	0	Line Interrupt enable
10 -0	ENCP_DBG_LN_INT_VAL	R/W	0	Line value that trig the interrupt

ENCP_VIDEO_YFP1_HTIME 0x1b94

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_YFP1_HTIME	R/W	240	Filter switch start point

ENCP_VIDEO_YFP2_HTIME 0x1b95

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_YFP2_HTIME	R/W	1657	Filter switch end point

ENCP_VIDEO_YC_DLY 0x1b96

Bit(s)	Field Name	R/W	Default	Description
5 –4	ENCP_VIDEO_Y_DLY	R/W	16'h0	Y delay
3 –2	ENCP_VIDEO_CB_DLY	R/W	16'h0	Cb delay
1 –0	ENCP_VIDEO_CR_DLY	R/W	16'h0	Cr delay

ENCP_VIDEO_MAX_PXCNT 0x1b97

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_MAX_PXCNT	R/W	13'd1715	Max pixel counter

ENCP_VIDEO_HSPULS_BEGIN 0x1b98

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_HSPULS_BEGIN	R/W	13'd1715	Analog Horizontal Sync Begin

ENCP_VIDEO_HSPULS_END 0x1b99

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_HSPULS_END	R/W	13'd125	Analog Horizontal Sync End

ENCP_VIDEO_HSPULS_SWITCH 0x1b9a

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_HSPULS_SWITCH	R/W	13'd88	Analog Horizontal switch point (for HDTV)

ENCP_VIDEO_VSPULS_BEGIN 0x1b9b

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_VSPULS_BEGIN	R/W	13'd0	Analog Veritcal Sync begin point

ENCP_VIDEO_VSPULS_END 0x1b9c

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_VSPULS_END	R/W	13'd1589	Analog Veritcal Sync end point

ENCP_VIDEO_VSPULS_BLINE 0x1b9d

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_VIDEO_VSPULS_BLINE	R/W	11'd5	Vso begin line

ENCP_VIDEO_VSPULS_ELINE 0x1b9e

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_VIDEO_VSPULS_ELINE	R/W	11'd10	Vso end line

ENCP_VIDEO_EQPULS_BEGIN 0x1b9f

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_EQPULS_BEGIN	R/W	13'd0	Analog Equalization pulse begin point

ENCP_VIDEO_EQPULS_END 0x1ba0

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_EQPULS_END	R/W	13'd1589	Analog Equalization pulse end point

ENCP_VIDEO_EQPULS_BLINE 0x1ba1

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_VIDEO_EQPULS_BLINE	R/W	11'd5	Equalization pulse begin line

ENCP_VIDEO_EQPULS_ELINE 0x1ba2

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_VIDEO_EQPULS_ELINE	R/W	11'd10	Equalization pulse end line

ENCP_VIDEO_HAVON_END 0x1ba3

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_HAVON_END	R/W	13'd1656	Vertical active video end point

ENCP_VIDEO_HAVON_BEGIN 0x1ba4

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_HAVON_BEGIN	R/W	13'd217	Vertical active video start point

ENCP_VIDEO_VAVON_ELINE 0x1baf

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_VIDEO_VAVON_ELINE	R/W	11'd519	Vertical active video on end line

ENCP_VIDEO_VAVON_BLINE 0x1ba6

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_VIDEO_VAVON_BLINE	R/W	11'd42	Vertical active video on start line

ENCP_VIDEO_HSO_BEGIN 0x1ba7

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_HSO_BEGIN	R/W	16	Digital Hsync out start point

ENCP_VIDEO_HSO_END 0x1ba8

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_HSO_END	R/W	32	Digital Hsync out end point

ENCP_VIDEO_VSO_BEGIN 0x1ba9

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_VSO_BEGIN	R/W	16	Digital Vsync out start point

ENCP_VIDEO_VSO_END 0x1baa

Bit(s)	Field Name	R/W	Default	Description
12 -0	ENCP_VIDEO_VSO_END	R/W	32	Digital Vsync out end point

ENCP_VIDEO_VSO_BLINE 0x1bab

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_VIDEO_VSO_BLINE	R/W	37	Digital Vsync out start line

ENCP_VIDEO_VSO_ELINE 0x1bac

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_VIDEO_VSO_ELINE	R/W	39	Digital Vsync out end line

ENCP_VIDEO_SYNC_WAVE_CURVE 0x1bad

Bit(s)	Field Name	R/W	Default	Description
0	ENCP_VIDEO_SYNC_WAVE_CURVE	R/W	1'b1	Enable curve wave on analog sync edge

ENCP_VIDEO_MAX_LNCNT 0x1bae

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_VIDEO_MAX_LNCNT	R/W	10'd524	Max line counter

ENCP_VIDEO_SY_VAL 0x1bb0

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_VIDEO_SY_VAL	R/W	0	Analog Sync Pulse value 1

ENCP_VIDEO_SY2_VAL 0x1bb1

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_VIDEO_SY2_VAL	R/W	480	Analog Sync Pulse value 2

ENCP_VIDEO_BLANKY_VAL 0x1bb2

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_VIDEO_BLANKY_VAL	R/W	10'd240	Blank Y value

ENCP_VIDEO_BLANKPB_VAL 0x1bb3

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_VIDEO_BLANKPB_VAL	R/W	10'd512	Blank Pb value

ENCP_VIDEO_BLANKPR_VAL 0x1bb4

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_VIDEO_BLANKPR_VAL	R/W	10'd512	Blank Pr value

ENCP_VIDEO_HOFFST 0x1bb5

Bit(s)	Field Name	R/W	Default	Description
13 -0	ENCP_VIDEO_HOFFST	R/W	2	Horizontal offset after HSI

ENCP_VIDEO_VOFFST 0x1bb6

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_VIDEO_VOFFST	R/W	0	Vertical offset after VSI

ENCP_VIDEO_RGB_CTRL 0x1bb7

Bit(s)	Field Name	R/W	Default	Description
12-8	CFG_BLANK_DLY	R/W	4	

Bit(s)	Field Name	R/W	Default	Description
6	CFG_MACV_R_EN	R/W	0	
5	CFG_MACV_G_EN	R/W	0	
4	CFG_MACV_B_EN	R/W	0	
3	CFG_RGB_SEPERATE_BLANK	R/W	0	
2	CFG_SYNC_ON_R	R/W	0	
1	CFG_SYNC_ON_G	R/W	1	
0	CFG_SYNC_ON_B	R/W	0	

ENCP_VIDEO_FILT_CTRL 0x1bb8

Bit(s)	Field Name	R/W	Default	Description
15	ENCP_VIDEO_FILT_YF_NO_RST	R/W	0	Disable auto reset Y filter each line
14	ENCP_VIDEO_FILT_CF_NO_RST	R/W	0	Disable auto reset C filter each line
14	CFILT_CHROMA444_EN	R/W	0	If true, chroma input is 444 instead of 422
12	ENCP_VIDEO_FILT_BYPASS_TOP	R/W	0	By pass all ENCP filter
11	ENCP_VIDEO_FILT_CHROMA_SWAP	R/W	0	Swap chroma Cb/Cr
10:8	ENCP_VIDEO_FILT_CF_BYPASS	R/W	0	Chroma filter by pass control
7:4	ENCP_VIDEO_FILT_YF_CFG1	R/W	0	4 Bits Y filter parameter 1, will be effect after ENCP_VIDEO_YFP1_HTIME Bit 3 : Edge mode Bit 2:0 : By pass ctrl
3:0	ENCP_VIDEO_FILT_YF_CFG2	R/W	0	4 Bits Y filter parameter 1, will be effect after ENCP_VIDEO_YFP1_HTIME Bit 3 : Edge mode Bit 2:0 : By pass ctrl

ENCP_ VIDEO_OFLD_VPEQ_OFST 0x1bb9

Bit(s)	Field Name	R/W	Default	Description
15:12	ENCP_VIDEO_OFLD_VPULS_OFST_BEGIN	R/W	0	Odd field Vsync pulse offset begin, 4 Bits signed.
11:8	ENCP_VIDEO_OFLD_VPULS_OFST_END	R/W	1	Odd field Vsync pulse offset end, 4 Bits signed.
7:4	ENCP_VIDEO_OFLD_EQPULS_OFST_BEGIN	R/W	0	Odd field EQU pulse offset begin, 4 Bits signed.
3:0	ENCP_VIDEO_OFLD_EQPULS_OFST_END	R/W	0	Odd field EQU pulse offset end, 4 Bits signed.

ENCP_ VIDEO_OFLD_VOAV_OFST 0x1bba

Bit(s)	Field Name	R/W	Default	Description
15:12	ENCP_VIDEO_OFLD_VSO_OFST_BEGIN	R/W	0	Odd field VSO offset begin, 4 Bits signed.
11:8	ENCP_VIDEO_OFLD_VSO_OFST_END	R/W	1	Odd field VSO offset end, 4 Bits signed.
7:4	ENCP_VIDEO_OFLD_VAVON_OFST_BEGIN	R/W	1	Odd field VAVON offset begin, 4 Bits signed.
3:0	ENCP_VIDEO_OFLD_VAVON_OFST_END	R/W	1	Odd field VAVON offset end, 4 Bits signed.

ENCP_ VIDEO_MATRIX_CB 0x1bbb

Bit(s)	Field Name	R/W	Default	Description
15:8	ENCP_VIDEO_MATRIX_MA	R/W	0	CbCr matrix parameter Ma (signed value)
7:0	ENCP_VIDEO_MATRIX_MB	R/W	0	CbCr matrix parameter Mb (signed value)

Note: These 2 register is for HUE adjustment. The formula is : Cb' = Cb*Ma + Cr*Mb, Cr' = Cb*Mc + Cr*Md (effective only when Enable HUE matrix by Bit ENCP_VIDEO_MODE_ADV[7])

Note : see the descritption of register ENCI_CMPN_MATRIX_CB.

ENCP_ VIDEO_MATRIX_CR 0x1bbc

Bit(s)	Field Name	R/W	Default	Description
15:8	ENCP_VIDEO_MATRIX_MC	R/W	0	CbCr matrix parameter Mc (signed value)
7:0	ENCP_VIDEO_MATRIX_MD	R/W	0	CbCr matrix parameter Md (signed value)

ENCP_ VIDEO_RGBIN_CTRL 0x1bbd

Bit(s)	Field Name	R/W	Default	Description
1		R/W	0	USE RGB data from VIU

Bit(s)	Field Name	R/W	Default	Description
0		R/W	0	VIU RGB IN BLANK ZERO

ENCP_MACV_BLANKY_VAL 0x1bc0

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_MACV_BLANKY_VAL	R/W	0	Macrovision Blank Y value

ENCP_MACV_MAXY_VAL 0x1bc1

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_MACV_MAXY_VAL	R/W	590	Macrovision max Y value

ENCP_MACV_1ST_PSSYNC_STRT 0x1bc2

Bit(s)	Field Name	R/W	Default	Description
10 -0	ENCP_MACV_1ST_PSSYNC_STRT	R/W	238	Macrovision first pseudo sync start

ENCP_MACV_PSSYNC_STRT 0x1bc3

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCP_MACV_PSSYNC_STRT	R/W	1	Macrovision pseudo sync start

ENCP_MACV_AGC_STRT 0x1bc4

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCP_MACV_AGC_STRT	R/W	61	Macrovision AGC pulse start

ENCP_MACV_AGC_END 0x1bc5

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCP_MACV_AGC_END	R/W	141	Macrovision AGC pulse end

ENCP_MACV_WAVE_END 0x1bc6

Bit(s)	Field Name	R/W	Default	Description
7 -0	ENCP_MACV_WAVE_END	R/W	175	Macrovision wave end

ENCP_MACV_STRTLINE 0x1bc7

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_MACV_STRTLINE	R/W	11	Macrovision start line

ENCP_MACV_ENDLINE 0x1bc8

Bit(s)	Field Name	R/W	Default	Description
9 -0	ENCP_MACV_ENDLINE	R/W	19	Macrovision end line

ENCP_MACV_TS_CNT_MAX_L 0x1bb9

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCP_MACV_TS_CNT_MAX_L	R/W	16'h7dcd	Macrovistion Cyclic Variation Time step Counter, Low 16Bits

ENCP_MACV_TS_CNT_MAX_H 0x1bca

Bit(s)	Field Name	R/W	Default	Description
3 -0	ENCP_MACV_TS_CNT_MAX_H	R/W	4'h3	Macrovistion Cyclic Variation Time step Counter, High 4Bits

ENCP_MACV_TIME_DOWN 0x1bcb

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCP_MACV_TIME_DOWN	R/W	16'd3068	Macrovistion Cyclic Variation Time start going down

ENCP_MACV_TIME_LO 0x1bbc

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCP_MACV_TIME_LO	R/W	16'd3658	Macrovistion Cyclic Variation Time start maitain low

ENCP_MACV_TIME_UP 0x1bcd

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCP_MACV_TIME_UP	R/W	16'd4366	Macrovistion Cyclic Variation Time start going up

ENCP_MACV_TIME_RST 0x1bce

Bit(s)	Field Name	R/W	Default	Description
15 -0	ENCP_MACV_TIME_RST	R/W	16'd4956	Macrovistion Cyclic Variation reset

ENCP_VBI_CTRL 0x1bd0

Bit(s)	Field Name	R/W	Default	Description
1	ENCP_VBI_EN	R/W	0	Enable VBI
0	ENCP_VBI_NO_CURVE	R/W	0	VBI data no curve at edge

ENCP_VBI_SETTING 0x1bd1

Bit(s)	Field Name	R/W	Default	Description
1	ENCP_VBI_LN	R/W	0x16	VBI line number
0	ENCP_VBI_CNT	R/W	0x28	VBI data bitcount

ENCP_VBI_BEGIN 0x1bd2

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_VBI_BEGIN	R/W	0	VBI pulse begin point

ENCP_VBI_WIDTH 0x1bd3

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_VBI_WIDTH	R/W	0	VBI data bitwidth

ENCP_VBI_HVAL 0x1bd4

Bit(s)	Field Name	R/W	Default	Description
9-0	ENCP_VBI_HVAL	R/W	0	VBI pulse value when data bit is high

ENCP_VBI_DATA0 0x1bd5)

Bit(s)	Field Name	R/W	Default	Description
15-0	ENCP_VBI_DATA0	R/W	0	VBI data 0 (LSB first)

ENCP_VBI_DATA1 0x1bd6

Bit(s)	Field Name	R/W	Default	Description
15-0	ENCP_VBI_DATA1	R/W	0	VBI data 1 (LSB first)

VENC_VDAC_DAC0_GAINCTRL 0x1bf0

Bit(s)	Field Name	R/W	Default	Description
15	DAC_GAIN_BP	R/W	0	gain bypass, 1:bypass
14-8		R/W	0	Unused
7-0	DAC_GAIN	R/W	0	gain setting 8-bit

VENC_VDAC_DAC0_OFFSET 0x1bf1

Bit(s)	Field Name	R/W	Default	Description
10-0	DAC_OFFSET_VAL	R/W	0	offset setting, 11 Bits

VENC_VDAC_DAC1_GAINCTRL 0x1bf2

Bit(s)	Field Name	R/W	Default	Description
15	DAC_GAIN_BP	R/W	0	gain bypass, 1:bypass
14-8		R/W	0	Unused

Bit(s)	Field Name	R/W	Default	Description
7-0	DAC_GAIN	R/W	0	gain setting 8-bit

VENC_VDAC_DAC1_OFFSET 0x1bf3

Bit(s)	Field Name	R/W	Default	Description
10-0	DAC_OFFSET_VAL	R/W	0	offset setting, 11 Bits

VENC_VDAC_DAC2_GAINCTRL 0x1bf4

Bit(s)	Field Name	R/W	Default	Description
15	DAC_GAIN_BP	R/W	0	gain bypass, 1:bypass
14-8		R/W	0	Unused
7-0	DAC_GAIN	R/W	0	gain setting 8-bit

VENC_VDAC_DAC2_OFFSET 0x1bf5

Bit(s)	Field Name	R/W	Default	Description
10-0	DAC_OFFSET_VAL	R/W	0	offset setting, 11 Bits

VENC_VDAC_DAC3_GAINCTRL 0x1bf6

Bit(s)	Field Name	R/W	Default	Description
15	DAC_GAIN_BP	R/W	0	gain bypass, 1:bypass
14-8		R/W	0	Unused
7-0	DAC_GAIN	R/W	0	gain setting 8-bit

VENC_VDAC_DAC3_OFFSET 0x1bf7

Bit(s)	Field Name	R/W	Default	Description
10-0	DAC_OFFSET_VAL	R/W	0	offset setting, 11 Bits

VENC_VDAC_DAC4_GAINCTRL 0x1bf8

Bit(s)	Field Name	R/W	Default	Description
15	DAC_GAIN_BP	R/W	0	gain bypass, 1:bypass
14-8		R/W	0	Unused
7-0	DAC_GAIN	R/W	0	gain setting 8-bit

VENC_VDAC_DAC4_OFFSET 0x1bf9

Bit(s)	Field Name	R/W	Default	Description
10-0	DAC_OFFSET_VAL	R/W	0	offset setting, 11 Bits

VENC_VDAC_DAC5_GAINCTRL 0x1bfa

Bit(s)	Field Name	R/W	Default	Description
15	DAC_GAIN_BP	R/W	0	gain bypass, 1:bypass
14-8		R/W	0	Unused
7-0	DAC_GAIN	R/W	0	gain setting 8-bit

VENC_VDAC_DAC5_OFFSET 0x1bfb

Bit(s)	Field Name	R/W	Default	Description
10-0	DAC_OFFSET_VAL	R/W	0	offset setting, 11 Bits

VENC_VDAC_FIFO_CTRL 0x1bfc

Bit(s)	Field Name	R/W	Default	Description
14	FIFO_EN_ENCT	R/W	0	
13	FIFO_EN_ENCI	R/W	0	
12	FIFO_EN_CNCP	R/W	0	
11-6	DAC_CLOCK_2X	R/W	0	
5-0	DAC_CLOCK_4X	R/W	0	

ENCI_DVI_HSO_BEGIN 0x1c00

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_DVI_HSO_BEGIN	R/W	1713	DVI/HDMI Interlace HSO begin position.

ENCI_DVI_HSO_END 0x1c01

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_DVI_HSO_END	R/W	121	DVI/HDMI Interlace HSO end position.

ENCI_DVI_VSO_BLINE_EVN 0x1c02

Bit(s)	Field Name	R/W	Default	Description
8-0	ENCI_DVI_VSO_BLINE_EVN	R/W	261	DVI/HDMI Interlace VSO start line for even field.

ENCI_DVI_VSO_BLINE_ODD 0x1c03

Bit(s)	Field Name	R/W	Default	Description
8-0	ENCI_DVI_VSO_BLINE_ODD	R/W	261	DVI/HDMI Interlace VSO start line for odd field.

ENCI_DVI_VSO_ELINE_EVN 0x1c04

Bit(s)	Field Name	R/W	Default	Description
8-0	ENCI_DVI_VSO_ELINE_EVN	R/W	1	DVI/HDMI Interlace VSO end line for even field.

ENCI_DVI_VSO_ELINE_ODD 0x1c05

Bit(s)	Field Name	R/W	Default	Description
8-0	ENCI_DVI_VSO_ELINE_ODD	R/W	2	DVI/HDMI Interlace VSO end line for odd field.

ENCI_DVI_VSO_BEGIN_EVN 0x1c06

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_DVI_VSO_BEGIN_EVN	R/W	855	DVI/HDMI Interlace VSO begin position for even field.

ENCI_DVI_VSO_BEGIN_ODD 0x1c07

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_DVI_VSO_BEGIN_ODD	R/W	1713	DVI/HDMI Interlace VSO begin position for odd field.

ENCI_DVI_VSO_END_EVN 0x1c08

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_DVI_VSO_END_EVN	R/W	1713	DVI/HDMI Interlace VSO end position for even field.

ENCI_DVI_VSO_END_ODD 0x1c09

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_DVI_VSO_END_ODD	R/W	855	DVI/HDMI Interlace VSO end position for odd field.

ENCI_CFILT_CTRL2 0x1c0a

Bit(s)	Field Name	R/W	Default	Description
15-12	ENCI_CFILT_CVBS_CB_DLY	R/W	0	Filter CVBS Cb delay. Delay same as below.
11-8	ENCI_CFILT_CVBS_CR_DLY	R/W	0	Filter CVBS Cr delay. Delay same as below.
7-4	ENCI_CFILT_CMPT_CB_DLY	R/W	0	Filter component Cb delay. Delay same as below.
3-0	ENCI_CFILT_CMPT_CR_DLY	R/W	0	Filter component Cr delay. 0=No delay; 1=Delay by 1 cycle; ... 6=Delay by 6 cycles; 7-16=Reserved.

ENCI_DACSEL0 0x1c0b

Bit(s)	Field Name	R/W	Default	Description
15-12	VENC_I_DACSEL_0	R/W	0	dac3

Bit(s)	Field Name	R/W	Default	Description
11-8	VENC_I_DACSEL_0	R/W	3	dac2
7-4	VENC_I_DACSEL_0	R/W	2	dac1
3-0	VENC_I_DACSEL_0	R/W	1	dac0

ENCI_DACSEL1 0x1c0c

Bit(s)	Field Name	R/W	Default	Description
9-8	DVI_SYNC_SEL	R/W	0	Dvi_sync_sel
7-4	VENC_I_DACSEL_1	R/W	5	dac5
3-0	VENC_I_DACSEL_1	R/W	4	dac4

ENCP_DACSEL0 0x1c0d

Bit(s)	Field Name	R/W	Default	Description
15-12	VENC_P_DACSEL_0	R/W	0	dac3
11-8	VENC_P_DACSEL_0	R/W	3	dac2
7-4	VENC_P_DACSEL_0	R/W	2	dac1
3-0	VENC_P_DACSEL_0	R/W	1	dac0

ENCP_DACSEL1 0x1c0e

Bit(s)	Field Name	R/W	Default	Description
9-8	DVI_SYNC_SEL	R/W	0	Dvi_sync_sel
7-4	VENC_P_DACSEL_1	R/W	5	dac5
3-0	VENC_P_DACSEL_1	R/W	4	dac4

ENCP_MAX_LINE_SWITCH_POINT 0x1c0f

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_DE_V_END_ODD	R/W	13'h1fff	max_pxcnt = (line >= cfg_max_line_switch_point) ? max_pxcnt_tmp - 1: max_pxcnt_tmp;

ENCI_TST_EN 0x1c10

Bit(s)	Field Name	R/W	Default	Description
0	ENCI_TST_EN	R/W	0	Enci_tst_en

ENCI_TST_MDSEL 0x1c11

Bit(s)	Field Name	R/W	Default	Description
1-0	ENCI_TST_MDSEL	R/W	1	Video test mode selectl: 0: enci_tst_mode_fixval 1: enci_tst_mode_colorbar 2: enci_tst_mode_thinline 3: enci_tst_mode_dotgrid

ENCI_TST_Y 0x1c12

Bit(s)	Field Name	R/W	Default	Description
9-0	ENCI_TST_Y	R/W	512	Default value of Y in test mode

ENCI_TST_CB 0x1c13

Bit(s)	Field Name	R/W	Default	Description
9-0	ENCI_TST_CB	R/W	512	Default value of Pb in test mode

ENCI_TST_CR 0x1c14

Bit(s)	Field Name	R/W	Default	Description
9-0	ENCI_TST_CR	R/W	512	Default value of Pr in test mode

ENCI_TST_CLRBAR_STRT 0x1c15

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_TST_CLRBAR_STRT	R/W	275	Color bar start position

ENCI_TST_CLRBAR_WIDTH 0x1c16

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_TST_CLRBAR_WIDTH	R/W	360	Color bar width

ENCI_TST_VDCNT_STSEL 0x1c17

Bit(s)	Field Name	R/W	Default	Description
1-0	ENCI_TST_VDCNT_STSEL	R/W	0	For interlace, vdata count set when color bar start

ENCI_VFIFO2VD_CTL 0x1c18

Bit(s)	Field Name	R/W	Default	Description
15-8	VFIFO2VD_VD_SEL	R/W	'h1b	Vfifo2vd_vd_sel: 00_01_10_11 -> Y_Cb_Y_Cr
7	VFIFO2VD_DROP	R/W	0	Vfifo2vd_drop
6-1	VFIFO2VD_DELAY	R/W	0	Vfifo2vd_delay
0	VFIFO2VD_EN	R/W	0	Vfifo2vd_en

ENCI_VFIFO2VD_PIXEL_START 0x1c19

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCI_VFIFO2VD_PIXEL_START	R/W	0	Vfifo2vd_pixel_start

ENCI_VFIFO2VD_PIXEL_END 0x1c1a

Bit(s)	Field Name	R/W	Default	Description
15	VDATA_YC_USE_FIRST_VFIFO_REG	R/W	0	Vdata_yc_use_vfifo_req
14-13	RESERVED			
12-0	ENCI_VFIFO2VD_PIXEL_END	R/W	0	Vfifo2vd_pixel_end

ENCI_VFIFO2VD_LINE_TOP_START 0x1c1b

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_VFIFO2VD_LINE_TOP_START	R/W	0	enci_vfifo2vd_line_top_start

ENCI_VFIFO2VD_LINE_TOP_END 0x1c1c

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_VFIFO2VD_LINE_TOP_END	R/W	0	enci_vfifo2vd_line_top_end

ENCI_VFIFO2VD_LINE_BOT_START 0x1c1d

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_VFIFO2VD_LINE_BOT_START	R/W	0	enci_vfifo2vd_line_bot_start

ENCI_VFIFO2VD_LINE_BOT_END 0x1c1e

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCI_VFIFO2VD_LINE_BOT_END	R/W	0	enci_vfifo2vd_line_bot_end

ENCI_VFIFO2VD_CTL2 0x1c1f

Bit(s)	Field Name	R/W	Default	Description
1	CFG_VFIFO2VD_OUT_SCALER_BYPASS	R/W	1	cfg_vfifo2vd_out_scaler_bypass
0	CFG_VFIFO_DIN_FULL_RANGE	R/W	0	cfg_vfifo_din_full_range

ENCP_DVI_HSO_BEGIN 0x1c30

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_DVI_HSO_BEGIN	R/W	16	DVI/HDMI Hsync out start point

ENCP_DVI_HSO_END 0x1c31

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_DVI_HSO_END	R/W	32	DVI/HDMI Hsync out end point

ENCP_DVI_VSO_BLINE_EVN 0x1c32

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCP_DVI_VSO_BLINE_EVN	R/W	40	DVI/HDMI Vsync out start line for even field.

ENCP_DVI_VSO_BLINE_ODD 0x1c33

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCP_DVI_VSO_BLINE_ODD	R/W	40	DVI/HDMI Vsync out start line for odd field.

ENCP_DVI_VSO_ELINE_EVN 0x1c34

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCP_DVI_VSO_ELINE_EVN	R/W	42	DVI/HDMI Vsync out end line for even field.

ENCP_DVI_VSO_ELINE_ODD 0x1c35

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCP_DVI_VSO_ELINE_ODD	R/W	42	DVI/HDMI Vsync out end line for odd field.

ENCP_DVI_VSO_BEGIN_EVN 0x1c36

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_DVI_VSO_BEGIN_EVN	R/W	16	DVI/HDMI Vsync out start point for even field.

ENCP_DVI_VSO_BEGIN_ODD 0x1c37

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_DVI_VSO_BEGIN_ODD	R/W	16	DVI/HDMI Vsync out start point for odd field.

ENCP_DVI_VSO_END_EVN 0x1c38

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_DVI_VSO_END_EVN	R/W	32	DVI/HDMI Vsync out end point for even field.

ENCP_DVI_VSO_END_ODD 0x1c39

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_DVI_VSO_END_ODD	R/W	32	DVI/HDMI Vsync out end point for odd field.

ENCP_DE_H_BEGIN 0x1c3a

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_DE_H_BEGIN	R/W	217	DVI/HDMI horizontal active video start point.

ENCP_DE_H_END 0x1c3b

Bit(s)	Field Name	R/W	Default	Description
12-0	ENCP_DE_H_END	R/W	1657	DVI/HDMI horizontal active video end point.

ENCP_DE_V_BEGIN_EVEN 0x1c3c

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCP_DE_V_BEGIN_EVEN	R/W	42	DVI/HDMI vertical active video start line for even field.

ENCP_DE_V_END_EVEN 0x1c3d

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCP_DE_V_END_EVEN	R/W	519	DVI/HDMI vertical active video end line for even field.

ENCP_DE_V_BEGIN_ODD 0x1c3e

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCP_DE_V_BEGIN_ODD	R/W	42	DVI/HDMI vertical active video start line for odd field.

ENCP_DE_V_END_ODD 0x1c3f

Bit(s)	Field Name	R/W	Default	Description
10-0	ENCP_DE_V_END_ODD	R/W	519	DVI/HDMI vertical active video end line for odd field.

ENCI_SYNC_LINE_LENGTH 0x1c40

Bit(s)	Field Name	R/W	Default	Description
15-11	SYNC_PULSE_LENGTH	R/W	0	Sync_pulse_length
10-0	SYNC_PULSE_START_LINE	R/W	0	Sync_pulse_start_line

ENCI_SYNC_PIXEL_EN 0x1c41

Bit(s)	Field Name	R/W	Default	Description
15	SYNC_PULSE_EN	R/W	0	Sync_pulse_enable
14	RESERVED	R/W	0	
13	SHORT_FUSSY_SYNC	R/W	0	Short_fussy_sync
12-0	SYNC_PULSE_START_PIXEL	R/W	0	Sync_pulse_start_pixel

ENCI_SYNC_TO_LINE_EN 0x1c42

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_SYNC_ENALBE	R/W	0	Enci_sync_enable
14	ENCP_SYNC_ENALBE	R/W	0	Encp_sync_enable
13	ENCT_SYNC_ENALBE	R/W	0	Enct_sync_enable
12	ENCL_SYNC_ENALBE	R/W	0	Encl_sync_enable
11	FUSSY_SYNC_ENABLE	R/W	0	Fussy_sync_enable
10-0	SYNC_PULSE_TARGET_LINE	R/W	0	Sync_pulse_target_line

ENCI_SYNC_TO_PIXEL 0x1c43

Bit(s)	Field Name	R/W	Default	Description
12-0	SYNC_PULSE_TARGET_PIXEL	R/W	0	Sync_pulse_target_pixel

ENCP_SYNC_LINE_LENGTH 0x1c44

Bit(s)	Field Name	R/W	Default	Description
15-11	SYNC_PULSE_LENGTH	R/W	0	Sync_pulse_length
10-0	SYNC_PULSE_START_LINE	R/W	0	Sync_pulse_start_line

ENCP_SYNC_PIXEL_EN 0x1c45

Bit(s)	Field Name	R/W	Default	Description
15	SYNC_PULSE_EN	R/W	0	Sync_pulse_enable
14	RESERVED	R/W	0	
13	SHORT_FUSSY_SYNC	R/W	0	Short_fussy_sync
12-0	SYNC_PULSE_START_PIXEL	R/W	0	Sync_pulse_start_pixel

ENCP_SYNC_TO_LINE_EN 0x1c46

Bit(s)	Field Name	R/W	Default	Description
15	ENCI_SYNC_ENALBE	R/W	0	Enci_sync_enable
14	ENCP_SYNC_ENALBE	R/W	0	Encp_sync_enable
13	ENCT_SYNC_ENALBE	R/W	0	Enct_sync_enable
12	ENCL_SYNC_ENALBE	R/W	0	Encl_sync_enable
11	FUSSY_SYNC_ENABLE	R/W	0	Fussy_sync_enable
10-0	SYNC_PULSE_TARGET_LINE	R/W	0	Sync_pulse_target_line

ENCP_SYNC_TO_PIXEL 0x1c47

Bit(s)	Field Name	R/W	Default	Description
12-0	SYNC_PULSE_TARGET_PIXEL	R/W	0	Sync_pulse_target_pixel

ENCP_VFIFO2VD_CTL2 0x1c50

Bit(s)	Field Name	R/W	Default	Description
3	VFIFO2VD_ENCP_LCD_SCALER_BYPASS	R/W	1	0=Scale LCD input data to y [16*4,235*4], c [16*4,240*4]

Bit(s)	Field Name	R/W	Default	Description
				1=Do not scale LCD input data
2	VFIFO2VD_ENCP_VADJ_SCALER_BYPASS	R/W	1	0=Scale enc480p_vadj input data to y [16*4,235*4], c [16*4,240*4] 1=Do not scale data to enc480p_vadj
1	VFIFO2VD_OUT_SCALER_BYPASS	R/W	1	0=Scale vfifo2vd's output vdata to y[16,235], c[16,240] 1=Do not scale vfifo2vd's output vdata
0	VFIFO_DIN_FULL_RANGE	R/W	0	0=Data from viu fifo is y[16*4,235*4], c[16*4,240*4] 1=Data from viu fifo is full range [0,1023]

VENC_DVI_SETTING_MORE 0x1c51

Bit(s)	Field Name	R/W	Default	Description
0	VENC_DVI_SEL_RGB	R/W	0	Applicable for using on-chip hdmi tx module only. This bit controls correct Bit(s)-mapping from Venc to hdmi_tx depending on whether YCbCr or RGB mode. 0=Map data bitfrom Venc to hdmi_tx for YCbCr mode; 1=Map data bitfrom Venc to hdmi_tx for RGB mode.

VENC_VDAC_DAC0_FILT_CTRL0 0x1c58

Bit(s)	Field Name	R/W	Default	Description
0	VENC_VDAC_DAC0_FILT_EN	R/W	0	0=Bypass filter; 1=Enable filter.

VENC_VDAC_DAC0_FILT_CTRL1 0x1c59

Notes: dout = ((din + din_d2) * coef1 + (din_d1 * coef0) + 32) >> 6

Bit(s)	Field Name	R/W	Default	Description
15-8	VENC_VDAC_DAC0_FILT_COEF1	R/W	8'h00	Filter coef1
7-0	VENC_VDAC_DAC0_FILT_COEF0	R/W	8'h40	Filter coef0

VENC_VDAC_DAC1_FILT_CTRL0 0x1c5a

Bit(s)	Field Name	R/W	Default	Description
0	VENC_VDAC_DAC1_FILT_EN	R/W	0	0=Bypass filter; 1=Enable filter.

VENC_VDAC_DAC1_FILT_CTRL1 0x1c5b

Notes: dout = ((din + din_d2) * coef1 + (din_d1 * coef0) + 32) >> 6

Bit(s)	Field Name	R/W	Default	Description
15-8	VENC_VDAC_DAC1_FILT_COEF1	R/W	8'h00	Filter coef1
7-0	VENC_VDAC_DAC1_FILT_COEF0	R/W	8'h40	Filter coef0

VENC_VDAC_DAC2_FILT_CTRL0 0x1c5c

Bit(s)	Field Name	R/W	Default	Description
0	VENC_VDAC_DAC2_FILT_EN	R/W	0	0=Bypass filter; 1=Enable filter.

VENC_VDAC_DAC2_FILT_CTRL1 0x1c5d

Notes: dout = ((din + din_d2) * coef1 + (din_d1 * coef0) + 32) >> 6

Bit(s)	Field Name	R/W	Default	Description
15-8	VENC_VDAC_DAC2_FILT_COEF1	R/W	8'h00	Filter coef1
7-0	VENC_VDAC_DAC2_FILT_COEF0	R/W	8'h40	Filter coef0

VENC_VDAC_DAC3_FILT_CTRL0 0x1c5e

Bit(s)	Field Name	R/W	Default	Description
0	VENC_VDAC_DAC3_FILT_EN	R/W	0	0=Bypass filter; 1=Enable filter.

VENC_VDAC_DAC3_FILT_CTRL1 0x1c5f

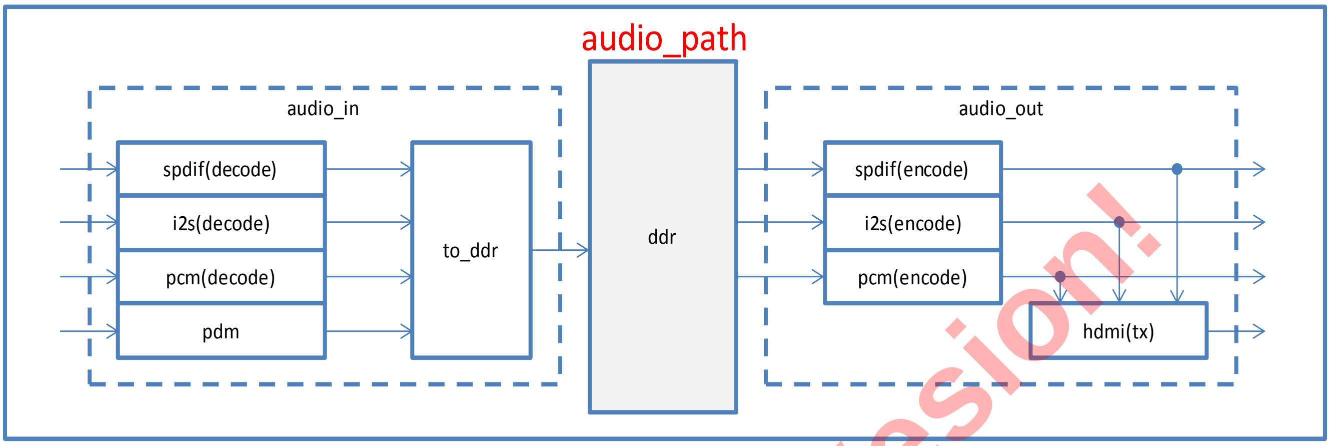
Notes: dout = ((din + din_d2) * coef1 + (din_d1 * coef0) + 32) >> 6

Bit(s)	Field Name	R/W	Default	Description
15-8	VENC_VDAC_DAC3_FILT_COEF1	R/W	8'h00	Filter coef1
7-0	VENC_VDAC_DAC3_FILT_COEF0	R/W	8'h40	Filter coef0

Section V Audio Path

Audio Path includes 2 parts: audio_in submodule and audio_out submodule, as illustrated in FigV.1. Audio_in part supports 4 formats of audio input, after decoding, these signal will be written to DDR via to_ddr module. Audio_out read signals from DDR and encode the signal to 4 formats of audio output.

Fig V.1 Diagram of Audio Path



28. Audio_Input

28.1 Overview

Audio_input supports 4 decoding formats: SPDIF, I2S, PCM and PDM, each of which has different decoding algorithm.

28.2 SPDIF

Spdif used biphase mark code(BMC) encode, it's double frequency of original data, and between two original data, there must have a invert operation.

At working status, the spdif module decodes the 1bit input it receives to 32 bit output.

The decoding procedure consists of the following 5 steps:

Resampling: lock input using a clock's(from control) posedge or negedge.

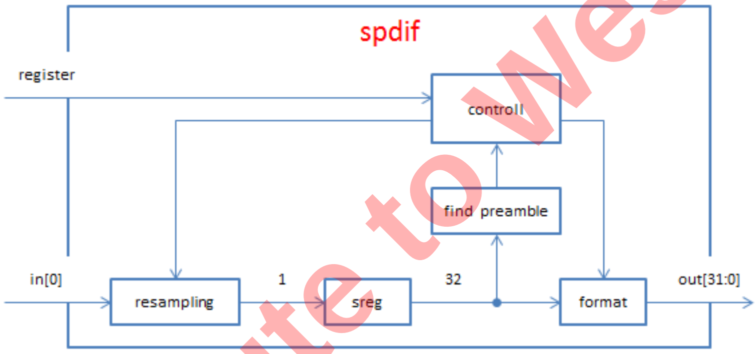
Sreg: shift store 32 input.

Find preamble: find 3 preamble and tell control.

Control: configured by register and find preamble, auto select clock.

Format: transfer 32Bit(s) data to a new format.

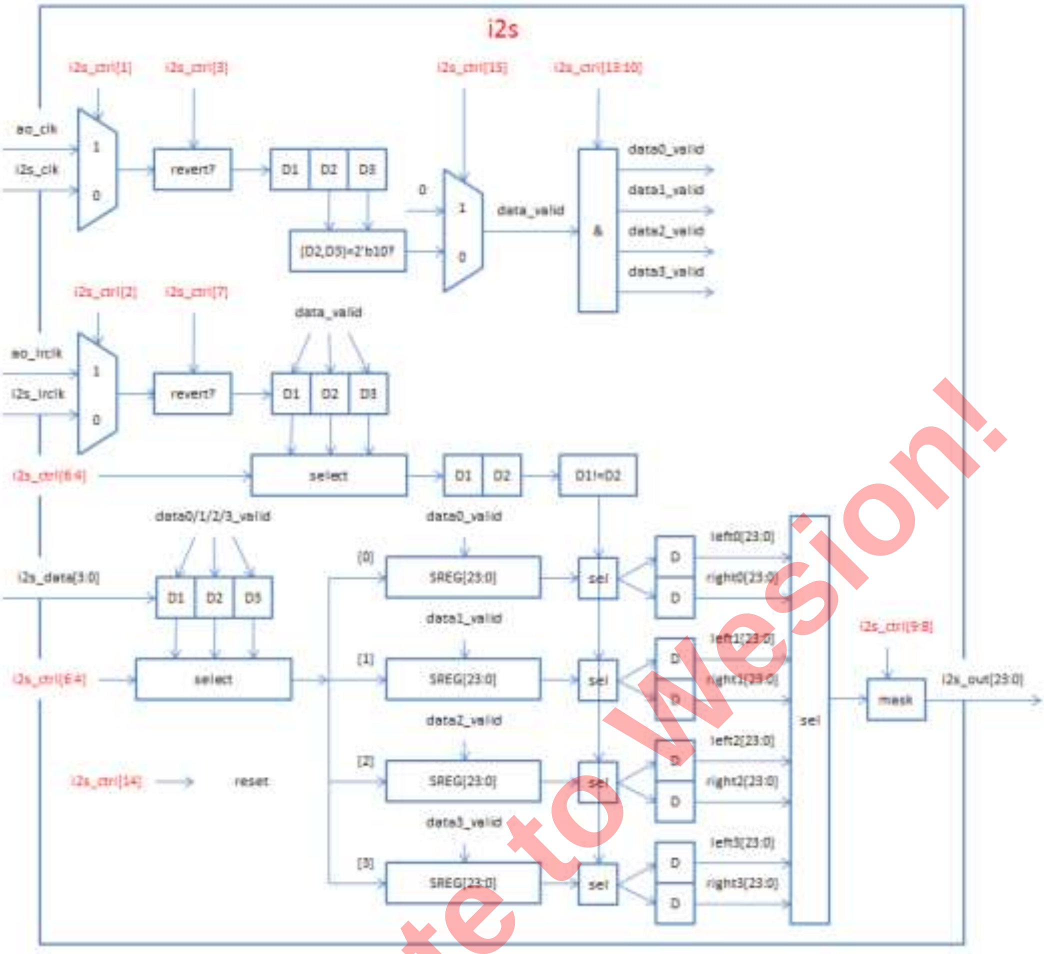
Fig V.28.1 Diagram of SPDIF



28.3 I2S(decode)

I2S(decode) supports 4 streams(8ch) input and 1 time division multiplexing output. Register I2S_ctrl[13:10] can be used to control it.

I2S will sent 24 Bits data by default, we can use I2S_ctrl[9:8] to mask some Bits, Fig V.28.2 Diagram of I2S



28.4 PCM(decode)

The hardware will detect data_valid and frame_sync first.

Next hardware will counter data_valid, if counter reached slot_msb_Bits(pcmin_ctrl1[26:21]), that means finished receive slot0, then counter clear and start receive slot1/2/…

Each slot has 1 enable bit (pcmin_ctrl1[15:0]):

if [0] is 0, slot0 will not be sent out.

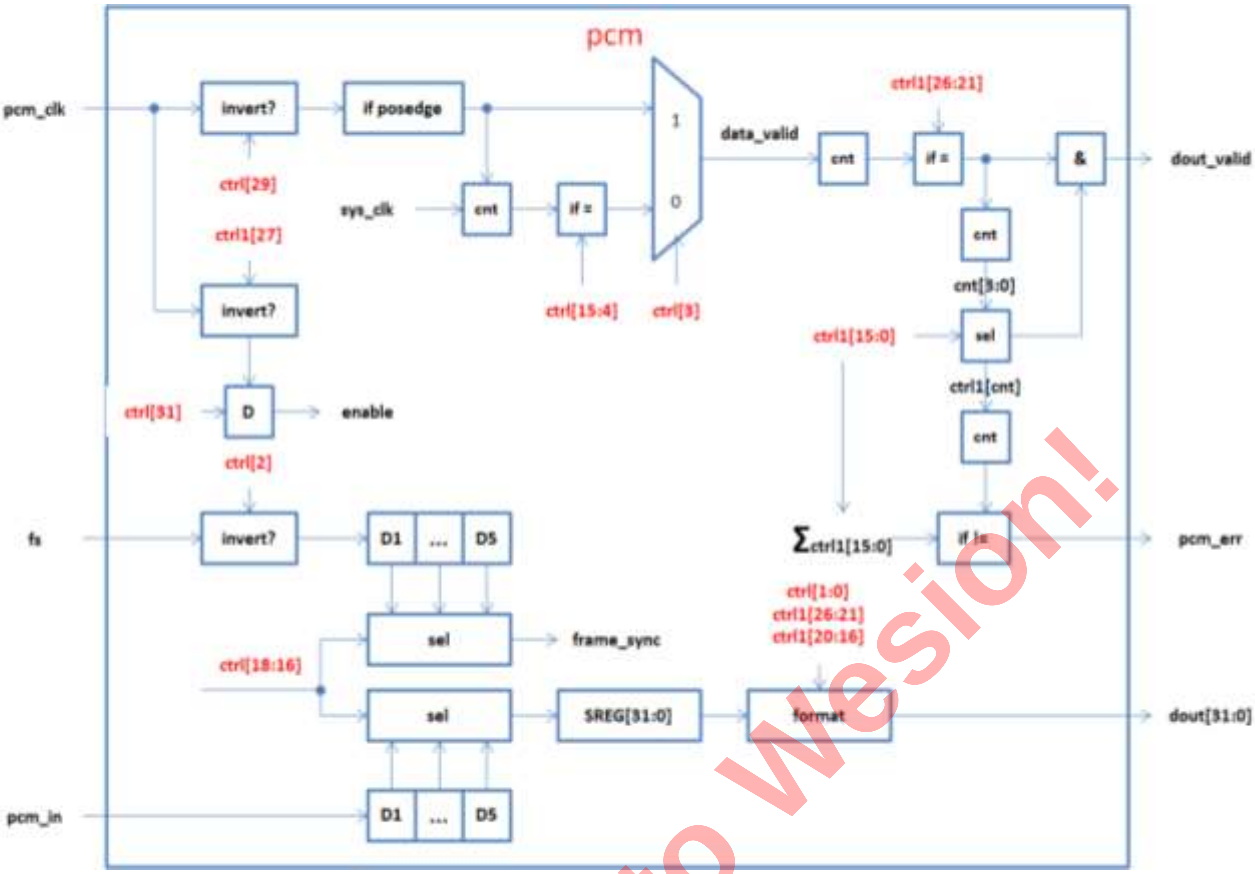
if [0] is 1, slot1 will be sent out.

if [1:0] is 2'b11, slot0 and slot1 will be sent out.

if [3:0] is 4'b1001, slot0 and slot3 will be sent out, but slot1 and slot2 will not.

Hardware has another counter to accumulate slot num. When one frame is finished(fs invert), hardware will compare it with theoretically (accumulate by pcmin_ctrl1[15:0]), if it's not the same, there will send a pcm_err to register(AUDIN_FIFO_INT[8]).

Fig V.28.3 Diagram of PCM

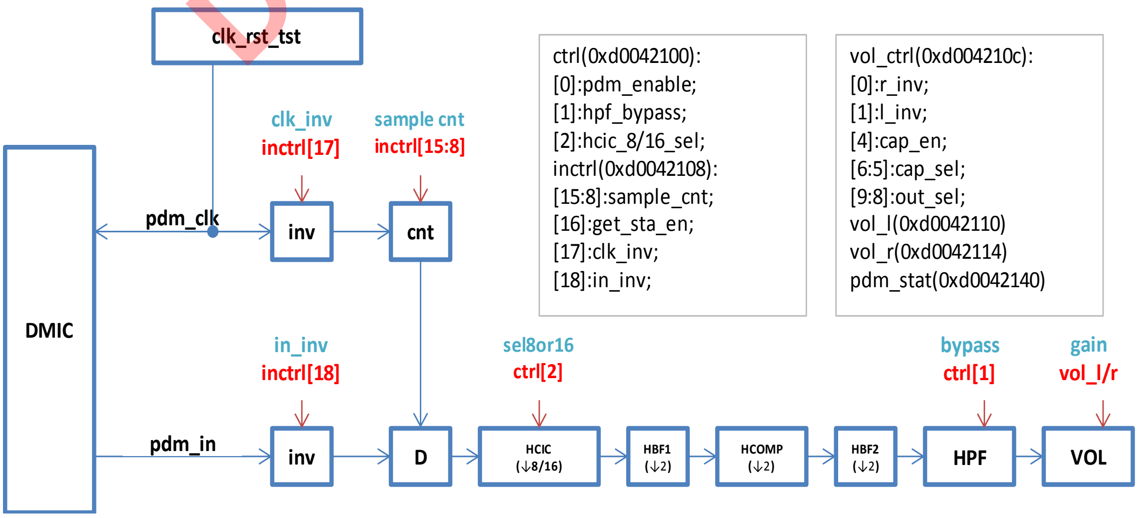


The format modification include: left_justified/msb_first/data_msb_bit (output data bit_width):

28.5 PDM

With PDM module, we can receive the pdm data and transfer it to sample data.

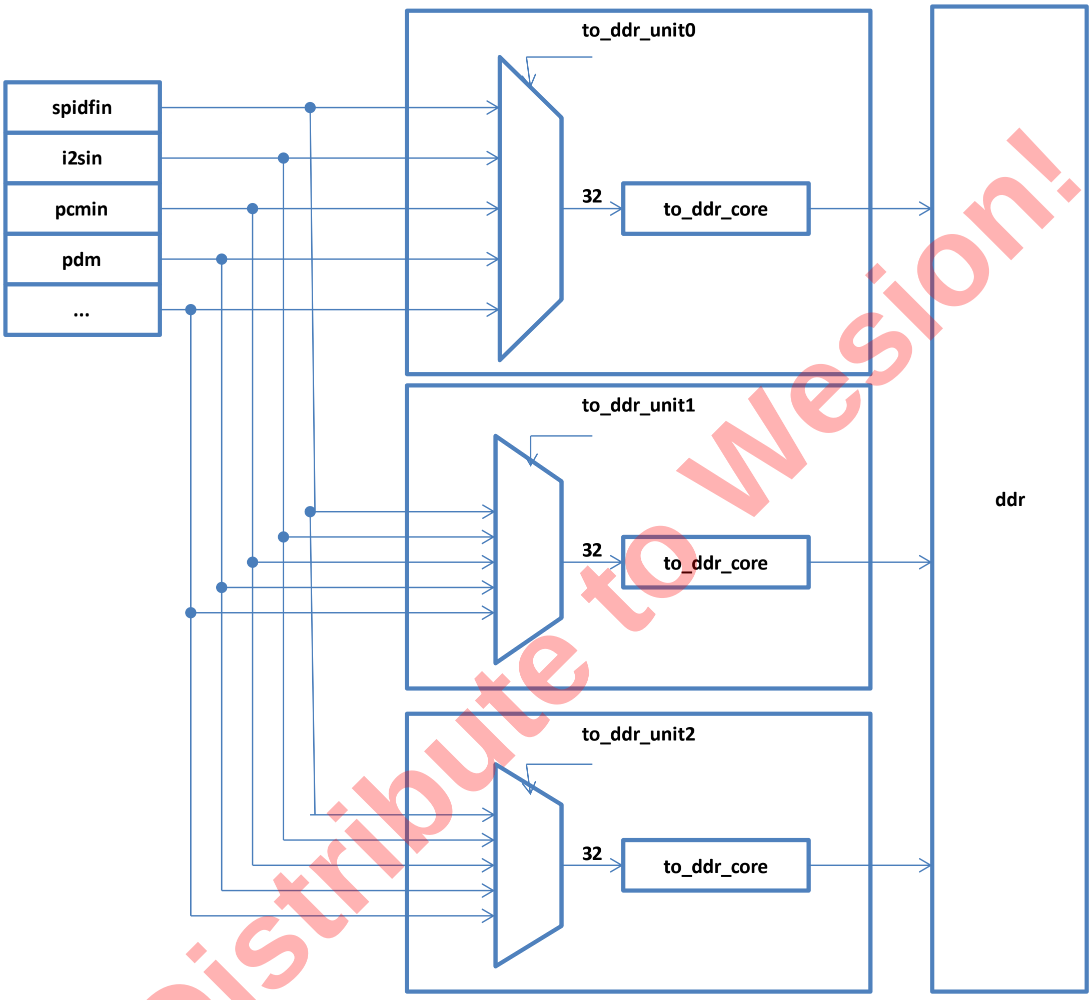
Fig V.28.4 Diagram of PDM



28.6 To_ddr

To_ddr_unit selects source audio data and modifies it, then write the modified data to the ddr memory. The modification consists of 2 parts: audio sample format modification and the ddr address obtaining. S905X can send 3 channels of audio data to ddr at the same time.

Fig V.28.5 Diagram of to_ddr



The to_ddr_unit functions in the following step:

Step1: select source audio.

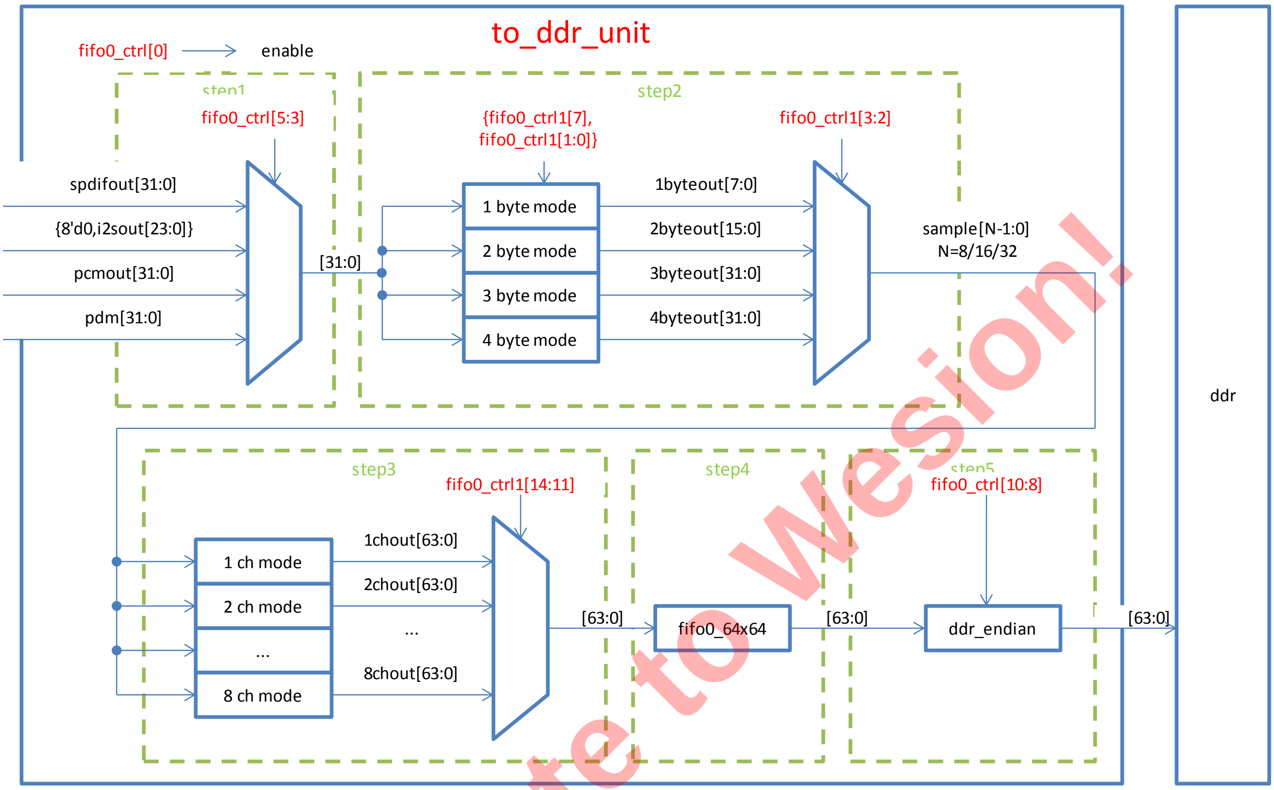
Step2: select byte number of audio sample.

Step3: select the mode of write adderss.

Step4: store audio sample to 64x64 fifo first, then read to ddr.

Step5: select endian mode before data to ddr.

Fig V.28.6 Data Path of to_ddr



28.7 Register Description

Audio In registers:

The final address of the following registers should be calculated with the following eqution:

Final Address = 0xC110A000 + offset*4

AUDIN_SPDIF_MODE 0x00

Bit(s)	R/W	Default	Description
31	RW	0x0	SPDIF_EN
30	RW	0x0	SPDIF_INT_EN
29	RW	0x0	SPDIF_BURST_PRE_INT_EN
28:25	RW	0x0	Reservered
24	RW	0x0	SPDIF_TIE_0
23	RW	0x0	SPDIF_SAMPLE_SEL
22	RW	0x0	SPDIF_REVERSE_EN
21	RW	0x1	SPDIFIN_SEL
20	RW	0x0	SPDIF_BIT_ORDER
19	RW	0x0	SPDIF_CHNL_ORDER
18	RW	0x0	SPDIF_DATA_TYPE_SEL

Bit(s)	R/W	Default	Description
17	RW	0x0	Reservered
16:14	RW	0x0	SPDIF_XTDCLK_UPD_ITVL
13:0	RW	0x2501	SPDIF_CLKNUM_54U

AUDIN_SPDIF_FS_CLK_RLTN 0x01

Bit(s)	R/W	Default	Description
29：24	RW	0x7	SPDIF_CLKNUM_192K
23:18	RW	0xe	SPDIF_CLKNUM_96K
17:12	RW	0x1d	SPDIF_CLKNUM_48K
10：6	RW	0x1f	SPDIF_CLKNUM_44K
5：0	RW	0x2b	SPDIF_CLKNUM_32K

AUDIN_SPDIF_CHNL_STS_A 0x02

Bit(s)	R/W	Default	Description
31:0	R	0x0	The channel A status information

AUDIN_SPDIF_CHNL_STS_B 0x03

Bit(s)	R/W	Default	Description
31:0	R	0x0	The channel B status information

AUDIN_SPDIF_MISC0x04

Bit(s)	R/W	Default	Description
31:9	R	0x0	reserved
8	R	0x0	Burst
7	R	0x0	Valid_in
6:4	R	0x0	Xtdclk_sel: 0: 32K 1:44K 2:46K 3:48K 4:96K 5:192K
3	R	0x0	reserved
2	R	0x0	Parity error
1	R	0x0	Validity
0	R	0x0	userdata

AUDIN_SPDIF_NPCM_PCPD 0x05

Bit(s)	R/W	Default	Description
31:16	R	0x0	The burst sub pc
15:0	R	0x0	The burst sub pd

AUDIN_I2SIN_CTRL 0x10

Bit(s)	R/W	Default	Description
15	RW	0x0	I2SIN_EN
13:10	RW	0x0	I2SIN_CHAN_EN
9:8	RW	0x0	I2SIN_SIZE
7	RW	0x0	I2SIN_LRCLK_INVT
6:4	RW	0x0	I2SIN_LRCLK_SKEW
3	RW	0x0	I2SIN_POS_SYNC
2	RW	0x0	I2SIN_LRCLK_SEL

1	RW	0x0	I2SIN_CLK_SEL I2S clk selection : 0 : from pad input. 1 : from AIU.
0	RW	0x0	I2SIN_DIR

AUDIN_SOURCE_SEL 0x11

Bit(s)	R/W	Default	Description
31:15	RW	0x0	Reserved
14:12	RW	0x0	Hdmirx_chsts_sel
11:8	RW	0x0	Hdmirx_chsts_en
7:6	RW	0x0	Reserved
5:4	RW	0x0	Spdif_src_sel
3:2	RW	0x0	Reserved
1:0	RW	0x0	I2sin_src_sel

AUDIN_DECODE_FORMAT 0x12

Bit(s)	R/W	Default	Description
31:25	RW	0x0	Reserved
24	RW	0x0	SPDIF enabled
23:22	RW	0x0	Reserved
21:20	RW	0x0	I2S_block_start_src:
19:17	RW	0x0	Reserved
16	RW	0x0	I2S enabled
15:8	RW	0x0	audio_channel_alloc
7	RW	0x0	hdmi_tx_audio_decoder input sel
6	RW	0x0	I2S_channel_config
5:4	RW	0x0	I2S_format_select
3:2	RW	0x0	I2S_bit_width
1	RW	0x0	ws polarity
0	RW	0x0	For SPDIF mode

AUDIN_DECODE_CONTROL_STATUS 0x13

Bit(s)	R/W	Default	Description
31:25	R	0x0	Reserved
24	R	0x0	channel_status stablility indicator
23:16	RW	0x0	Valid Bits for audio sample packet. [7] for valid_sp3_right, [6] for valid_sp3_left, ..., [1] for valid_sp0_right, [0] for valid_sp0_left.
15：8	RW	0x0	User Bits for audio sample packet. [7] for user_sp3_right, [6] for user_sp3_left, ..., [1] for user_sp0_right, [0] for user_sp0_left
7：4	RW	0x0	cntl_init_discard. Number of initial hdmi_tx_audio_decoder samples to discard from reset.
3	RW	0x0	cntl_invert_I2S_lrclk Invert WS before input to hdmi_tx_audio_decoder
2	RW	0x0	audio_valid_overwrite. Valid bit selection in audio packet. 0=use input data; 1=use
1	RW	0x0	audio_user_overwrite. User bit selection in audio packet. 0=use input data; 1=use
0	RW	0x0	audio_sample_valid,sample non-flat indication. 0=flat, non-valid; 1=non-flat, valid

AUDIN_DECODE_CHANNEL_STATUS_A_0 0x14

Bit(s)	R/W	Default	Description
31:0	R	0x0	Channel status information from HDMIRX, [31:0]

AUDIN_DECODE_CHANNEL_STATUS_A_1 0x15

Bit(s)	R/W	Default	Description

31:0	R	0x0	Channel status information from HDMIRX, [63:32]

AUDIN_DECODE_CHANNEL_STATUS_A_2 0x16

Bit(s)	R/W	Default	Description
31:0	R	0x0	Channel status information from HDMIRX, [95:64]

AUDIN_DECODE_CHANNEL_STATUS_A_3 0x17

Bit(s)	R/W	Default	Description
31:0	R	0x0	Channel status information from HDMIRX, [127:96]

AUDIN_DECODE_CHANNEL_STATUS_A_4 0x18

Bit(s)	R/W	Default	Description
31:0	R	0x0	Channel status information from HDMIRX, [159:128]

AUDIN_DECODE_CHANNEL_STATUS_A_5 0x19

Bit(s)	R/W	Default	Description
31:0	R	0x0	Channel status information from HDMIRX, [191:160]

AUDIN_FIFO0_START 0x20

Bit(s)	R/W	Default	Description
31:3	RW	0x1fffffff	The start address of fifo0 writes ddr.
2:0	R	0x0	reserved

AUDIN_FIFO0_END 0x21

Bit(s)	R/W	Default	Description
31:3	RW	0x1fffffff	The end address of fifo0 writes ddr.
2:0	R	0x0	reserved

AUDIN_FIFO0_PTR 0x22

Bit(s)	R/W	Default	Description
31:3	R	0x0	The current address of fifo0 writes ddr.
2:0	R	0x0	reserved

AUDIN_FIFO0_INTR 0x23

Bit(s)	R/W	Default	Description
31:3	RW	0x0	When current address of fifo0 writes ddr matched, fifo0 will generate IRQ
2:0	R	0x0	reserved

AUDIN_FIFO0_RDPTR 0x24

Bit(s)	R/W	Default	Description
31:3	RW	0x0	The read address of fifo0.
2:0	R	0x0	reserved

AUDIN_FIFO0_CTRL 0x25

Bit(s)	R/W	Default	Description
31:29	RW	0x0	reserved
28	RW	0x0	AUDIN_FIFO0_HOLD_LVL
27:26	RW	0x0	AUDIN_FIFO0_HOLD2_SEL
25:24	RW	0x0	AUDIN_FIFO0_HOLD1_SEL
23:22	RW	0x0	AUDIN_FIFO0_HOLD0_SEL

Bit(s)	R/W	Default	Description
21	RW	0x0	AUDIN_FIFO0_HOLD2_EN
20	RW	0x0	AUDIN_FIFO0_HOLD1_EN
19	RW	0x0	AUDIN_FIFO0_HOLD0_EN
15	RW	0x0	AUDIN_FIFO0_UG, urgent request enable
14:11	RW	0x0	AUDIN_FIFO0_CHAN. channel number. in M1 suppose there's only 1 channel and 2 channel
10:8	RW	0x0	AUDIN_FIFO0_ENDIAN data endian control
3	RW	0x0	AUDIN_FIFO0_DIN_SEL // 0 spdifIN // 1 I2Sin // 2 PCMIN // 3 HDMI in // 4 DEMODULATOR IN
2	RW	0x0	AUDIN_FIFO0_LOAD write 1 to load address to AUDIN_FIFO0.
1	RW	0x0	AUDIN_FIFO0_RST
0	RW	0x0	AUDIN_FIFO0_EN

AUDIN_FIFO0_CTRL1 0x26

Bit(s)	R/W	Default	Description
31:8	RW	0x0	reserved
7	RW	0x0	Din_pos[2]
6	RW	0x0	Reserved
5:4	RW	0x0	Dest_sel
3:2	RW	0x0	Din_byte_num
1:0	RW	0x0	Din_pos[1:0]

AUDIN_FIFO0_LVL0 0x27

Bit(s)	R/W	Default	Description
31:3	R	0x0	Fifo0 empty lvl0
2:0	R	0x0	reserved

AUDIN_FIFO0_LVL1 0x28

Bit(s)	R/W	Default	Description
31:3	R	0x0	Fifo0 empty lvl1
2:0	R	0x0	reserved

AUDIN_FIFO0_LVL2 0x29

Bit(s)	R/W	Default	Description
31:3	R	0x0	Fifo0 empty lvl2
2:0	R	0x0	reserved

AUDIN_FIFO0_REQID 0x30

Bit(s)	R/W	Default	Description
31:18	RW	0x0	reserved
17:12	RW	0x3	Audin req burst number
11:6	RW	0x3f	Fifo0 req burst number
5:0	RW	0x0	Fifo0 req ID

AUDIN_FIFO0_WRAP 0x31

Bit(s)	R/W	Default	Description
31:16	R	0x0	reserved

15:0	R	0x0	Fifo0 wrap counter

AUDIN_FIFO1_START 0x33

Refer to FIFO0.

AUDIN_FIFO1_END 0x34

Refer to FIFO0.

AUDIN_FIFO1_PTR 0x35

Refer to FIFO0.

AUDIN_FIFO1_INTR 0x36

Refer to FIFO0.

AUDIN_FIFO1_RDPTR 0x37

Refer to FIFO0.

AUDIN_FIFO1_CTRL 0x38

Refer to FIFO0.

AUDIN_FIFO1_CTRL1 0x39

Refer to FIFO0.

AUDIN_FIFO1_LVL0 0x40

Refer to FIFO0.

AUDIN_FIFO1_LVL1 0x41

Refer to FIFO0.

AUDIN_FIFO1_LVL2 0x42

Refer to FIFO0.

AUDIN_FIFO1_REQID 0x43

Refer to FIFO0.

AUDIN_FIFO1_WRAP 0x44

Refer to FIFO0.

AUDIN_FIFO2_START 0x45

Refer to FIFO0.

AUDIN_FIFO2_END 0x46

Refer to FIFO0.

AUDIN_FIFO2_PTR 0x47

Refer to FIFO0.

AUDIN_FIFO2_INTR 0x48

Refer to FIFO0.

AUDIN_FIFO2_RDPTR 0x49

Refer to FIFO0.

AUDIN_FIFO2_CTRL 0x4a

Refer to FIFO0.

AUDIN_FIFO2_CTRL1 0x4b

Refer to FIFO0.

AUDIN_FIFO2_LVL0 0x4c

Refer to FIFO0.

AUDIN_FIFO2_LVL1 0x4d

Refer to FIFO0.

AUDIN_FIFO2_LVL2 0x4e

Refer to FIFO0.

AUDIN_FIFO2_REQID 0x4f

Refer to FIFO0.

AUDIN_FIFO2_WRAP 0x50

Refer to FIFO0.

AUDIN_INT_CTRL 0x51

Bit(s)	R/W	Default	Description
31:16	RW	0x0	reserved
15:14	RW	0x3	reserved
13	RW	0x1	Fifo2 flush mask
12	RW	0x1	Fifo2 int mask
11	RW	0x1	Fifo2 overflow int mask
10	RW	0x1	audout address trigger interrupt mask
9	RW	0x1	PCMOUT error interrupt mask
8	RW	0x1	PCMIN error interrupt mask
7	RW	0x1	AUDOUT_FIFO level low pulse interrupt mask. once the audout fifo counter lower than the 22:16Bits. it will generate one interrupt
6	RW	0x1	AUDOUT_FIFO level low level interrupt mask. if the audout fifo coutner is lower than 22:16 Bits, it will generate interrupt
5	RW	0x1	fifo1 PIO mode flush request interrupt mask
4	RW	0x1	fifo0 PIO mode flush request interrupt mask
3	RW	0x1	fifo1 address trigger interrupt mask
2	RW	0x1	fifo1 overflow interrupt mask
1	RW	0x1	fifo0 address trigger interrupt mask
0	RW	0x1	fifo0 overflow interrupt mask

AUDIN_FIFO_INT 0x52

Bit(s)	R/W	Default	Description
31:14	R	0x0	Reserved
13	R	0x0	Fifo2 flush int st
12	R	0x0	Fifo2 int st
11	R	0x0	Fifo2 overflow int st

Bit(s)	R/W	Default	Description
10	R	0x0	audout address trigger interrupt. Write 1 to clean
9	R	0x0	PCMOUT error interrupt. Write 1 to clean
8	R	0x0	PCMIN error interrupt. Write 1 to clean
7	R	0x0	AUDOUT_FIFO level low pulse interrupt. Write 1 to clean
6	R	0x0	AUDOUT_FIFO level low level interrupt. Write 1 to clean
5	R	0x0	fifo1 PIO mode flush request interrupt. Write 1 to clean
4	R	0x0	fifo0 PIO mode flush request interrupt. Write 1 to clean
3	R	0x0	fifo1 address trigger interrupt. Write 1 to clean
2	R	0x0	fifo1 overflow interrupt. Write 1 to clean
1	R	0x0	fifo0 address trigger interrupt. Write 1 to clean
0	R	0x0	fifo0 overflow interrupt. Write 1 to clean

PCMIN_CTRL0 0x60

Bit(s)	R/W	Default	Description
31	RW	0x0	PCMIN enable
30	RW	0x0	PMCIN soft reset. write 1 to reset PCMIN module
29	RW	0x0	PCMIN sync on clock posedge
18:16	RW	0x0	PCMIN fs skew 000: FS and data no skew. 001. delay fs one cycle. 010. delay fs two cycle. 011. delay fs 3 cycle. 111. delay data one cycle. 110. delay data 2 cycle. 101. delay data 3 cycle. 100. delay data 4 cycle.
15:4	RW	0x0	system clock data sample count
3	RW	0x0	system clock data sample selection 1 = use edge detection. 0 = use the clock data sample counter
2	RW	0x0	fs invert. invert the FS signal
1	RW	0x0	1 = the coming data is msb first. 0 = lsb first.
0	RW	0x0	1 = the coming data is left justified. 0 = the coming data is right justified.

PCMIN_CTRL1 0x61

Bit(s)	R/W	Default	Description
29	RW	0x0	pcmin SRC sel. 1= pcmin is from internal audio CODEC(I2S format). 0 = external PCM interface (from pad).
28	RW	0x0	pcmin clock sel. 1= internal from cts_pcm_clk. 0 = external PCM interface
27	RW	0x0	1: use negedge clock to sample the coming data. 0. use pcmin posedge clk to sample the data
26:21	RW	0x0	max slot number in one frame
20:16	RW	0x0	max bit number in one slot
15:0	RW	0x0	valid slot . Each Bit for one slot

PCMIN1_CTRL0 0x62

Bit(s)	R/W	Default	Description
31	RW	0x0	PCMIN enable.
30	RW	0x0	PMCIN soft reset. write 1 to reset PCMIN module.
29	RW	0x0	PCMIN sync on clock posedge.

Bit(s)	R/W	Default	Description
18:16	RW	0x0	PCMIN fs skew. //000: FS and data no skew. //001. delay fs one cycle. //010. delay fs two cycle. //011. delay fs 3 cycle. //111. delay data one cycle. //110. delay data 2 cycle. //101. delay data 3 cycle. //100. delay data 4 cycle.
15:4	RW	0x0	system clock data sample count.
3	RW	0x0	system clock data sample selection. 1 = use edge detection. 0 = use the clock data sample counter.
2	RW	0x0	fs invert. invert the FS signal.
1	RW	0x0	1 = the coming data is msb first. 0 = lsb first.
0	RW	0x0	1 = the coming data is left justified. 0 = the coming data is right justified.

PCMIN1_CTRL1 0x63

Bit(s)	R/W	Default	Description
29	RW	0x0	pcmin SRC sel. 1= pcmin is from internal audio CODEC(I2S format). 0 = external PCM interface (from pad).
28	RW	0x0	pcmin clock sel. 1= internal from cts_pcm_clk. 0 = external PCM interface.
27	RW	0x0	1: use negedge clock to sample the coming data. 0. use pcmin posedge clk to sample the data.
26:21	RW	0x0	max slot number in one frame.
20:16	RW	0x0	max bit number in one slot.
15:0	RW	0x0	valid slot . Each Bit for one slot.

PCMOUT_CTRL0 0x70

Bit(s)	R/W	Default	Description
31	RW	0x0	pcmout enable bit.
29	RW	0x0	pcmout 1= master mode. 0 = slave mode.
28	RW	0x0	system clock sync at pcmout posedge clock.
27	RW	0x0	system clock sync at clock edge of pcmout clock. 0 = sync on clock counter.
26：15	RW	0x0	system clock sync at counter number if sync on clock counter.
14	RW	0x0	pcmout is msb first.
13	RW	0x0	left justified.
12	RW	0x0	output data is at h24b of the input.
11：6	RW	0x0	in PCM slave mode. when pcmo received a FS, it will sync the slot bit counter to this number.
5：0	RW	0x0	in PCM slave mode. when pcmo received a FS, it will sync the frame slot counter to this number.

PCMOUT_CTRL1 0x71

Bit(s)	R/W	Default	Description
31：30	RW	0x0	pcmo output data byte number. 00 : 8Bits. 01: 16Bits. 10: 24Bits. 11: 32Bits.
28	RW	0x0	pcmo clock slow invert. invert the pcm output logic clock. for example, use negedge to output data.
27	RW	0x0	pcmo slave parts clock invert. invert the clock which is used to sample the fs_in when pcmo in slave mode.

Bit(s)	R/W	Default	Description
26	RW	0x0	invert fs phase.
25	RW	0x0	invert the fs_o for master mode.
23：18	RW	0x0	fs_o start postion frame slot counter number
17：12	RW	0x0	fs_o start postion slot bit counter number.
11：6	RW	0x0	fs_o end postion frame slot counter number.
5：0	RW	0x0	fs_o end postion slot bit counter number.

PCMOUT_CTRL2 0x72

Bit(s)	R/W	Default	Description
31	RW	0x0	pcmo mute.
30：29	RW	0x0	pcmo_underrun mode. 00: the output data will use mute constant. 01: repeat the previous data.
27：22	RW	0x0	pcmo max slot number in one frame.
21：16	RW	0x0	pcmo max bit number in one slot.
15：0	RW	0x0	pcmo valid slot. Each Bit for one slot.

PCMOUT_CTRL3 0x73

Bit(s)	R/W	Default	Description
31:0	RW	0x0	pcmo mute constant value.

PCMOUT1_CTRL0 0x74

Bit(s)	R/W	Default	Description
31	RW	0x0	pcmout enable bit
29	RW	0x0	pcmout 1= master mode. 0 = slave mode
28	RW	0x0	system clock sync at pcmout posedge clock.
27	RW	0x0	system clock sync at clock edge of pcmout clock. 0 = sync on clock counter.
26:15	RW	0x0	system clock sync at counter number if sync on clock counter.
14	RW	0x0	pcmout is msb first.
13	RW	0x0	left justified.
12	RW	0x0	output data is at h24b of the input.
11:6	RW	0x0	in PCM slave mode. when pcmo received a FS, it will sync the slot bit counter to this number.
5:0	RW	0x0	in PCM slave mode. when pcmo received a FS, it will sync the frame slot counter to this number.

PCMOUT1_CTRL1 0x75

Bit(s)	R/W	Default	Description
31:30	RW	0x0	pcmo output data byte number. 00 : 8Bits. 01: 16Bits. 10: 24Bits. 11: 32Bits.
28	RW	0x0	pcmo clock slow invert. invert the pcm output logic clock. for example, use negedge to output data.
27	RW	0x0	pcmo slave parts clock invert. invert the clock which is used to sample the fs_in when pcmo in slave mode.
26	RW	0x0	invert fs phase
25	RW	0x0	invert the fs_o for master mode
23:18	RW	0x0	fs_o start postion frame slot counter number
17:12	RW	0x0	fs_o start postion slot bit counter number
11:6	RW	0x0	fs_o end postion frame slot counter number
5:0	RW	0x0	fs_o end postion slot bit counter number

PCMOUT1_CTRL2 0x76

Bit(s)	R/W	Default	Description

31	RW	0x0	pcmo mute
30:29	RW	0x0	pcmo_underrun mode. 00: the output data will use mute constant. 01: repeat the previous data
27:22	RW	0x0	pcmo max slot number in one frame
21:16	RW	0x0	pcmo max bit number in one slot
15:0	RW	0x0	pcmo valid slot. Each Bit for one slot.

PCMOUT1_CTRL3 0x77

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Pcmo1 mute constant value.

AUDOUT_CTRL 0x80

Bit(s)	R/W	Default	Description
31	RW	0x0	AUDOUT FIFO enable.
30	RW	0x0	AUDOUT FIFO reset. write 1 to reset
29	RW	0x0	AUDOUT fifo load DMA address. write 1 to load.
28	RW	0x0	Clr audout wrap counter. write 1 to clear.
27	RW	0x0	clr audout rdrsp counter. write 1 to clear.
24	RW	0x0	audout next bufer enable. if in pingpeng buffer mode.
23:22	RW	0x0	audout level control write pointer selection.
21:25	RW	0x0	audout DMA one time request size.
14:8	RW	0x0	audout buffer level. buffer level must > one time DMA request size.
7	RW	0x0	audout buffer level control enable.
6	RW	0x0	1 audout DMA mode. 0. audout PIO mode.(CPU use cbus to push data.)
5	RW	0x0	audout use circur Buffer in DDR2/3 SDRAM. 1 = circur buffer mdoe. 0 = pingpang buffer mode.
4	RW	0x0	audout buffer selection if in ping pang buffer mode.
3	RW	0x0	audout DMA request urgent bit.
2:0	RW	0x0	audout DMA data endian control.

AUDOUT_CTRL1 0x81

Bit(s)	R/W	Default	Description
11:6	RW	0x0	AUDOUT DMA request burst number control for pre_mmc_arb.
5:0	RW	0x0	AUDOUT DMA request ID.

AUDOUT_BUF0_STA 0x82

Bit(s)	R/W	Default	Description
31:0	RW	0x0	buf0 start address. or circur buffer start address

AUDOUT_BUF0_EDA 0x83

Bit(s)	R/W	Default	Description
31:0	RW	0x0	buf0 end address. or circur buffer end address

AUDOUT_BUF0_WPTR 0x84

Bit(s)	R/W	Default	Description
31:0	RW	0x0	buf0 write pointer.

AUDOUT_BUF1_STA 0x85

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Buf1 start address. or circur buffer start address

AUDOUT_BUF1_EDA 0x86

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Buf1 end address. or circur buffer end address

AUDOUT_BUF1_WPTR 0x87

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Buf1 write pointer.

AUDOUT_FIFO_RPTR 0x88

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Buffer DMA read address

AUDOUT_INTR_PTR 0x89

Bit(s)	R/W	Default	Description
31:0	RW	0x0	When DMA read to this address, it will generate an interrupt to CPU

AUDOUT_FIFO_STS 0x8a

Bit(s)	R/W	Default	Description
31:7	R	0x0	reserved
6:0	R	0x0	AUDOUT FIFO depth counter.

AUDIN_MUTE_VAL 0x35

Bit(s)	R/W	Default	Description
31:0	RW	0x0	cntl_mute_val: Use this value during mute, if cntl_mute_mode=2.

AUDIN_FIFO0_PIO_STS 0xb0

Bit(s)	R/W	Default	Description
31	R	0x0	Fifo0 pio request.
30:0	R	0x0	Reserved

AUDIN_FIFO0_PIO_RDL 0xb1

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Fifo0 pio write data [31:0]

AUDIN_FIFO0_PIO_RDH 8'hb2 0xb2

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Fifo0 pio write data [63:32]

AUDIN_FIFO1_PIO_STS 0xb3

Bit(s)	R/W	Default	Description
31	R	0x0	Fifo1 pio request.
30:0	R	0x0	Reserved

AUDIN_FIFO1_PIO_RDL 0xb4

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Fifo1 pio write data [31:0]

AUDIN_FIFO1_PIO_RDH 0xb5

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Fifo1 pio write data [63:32]

AUDIN_FIFO2_PIO_STS 0xb6

Bit(s)	R/W	Default	Description
31	R	0x0	Fifo2 pio request.
30:0	R	0x0	Reserved

AUDIN_FIFO2_PIO_RDL 0xb7

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Fifo2 pio write data [31:0]

AUDIN_FIFO2_PIO_RDH 0xb8

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Fifo2 pio write data [63:32]

AUDOUT_FIFO_PIO_STS 0xb9

Bit(s)	R/W	Default	Description
31	R	0x0	audout pio request.
30:7	R	0x0	Reserved
6:0	R	0x0	Audout fifo low level

AUDOUT_FIFO_PIO_WRL 0xba

Bit(s)	R/W	Default	Description
31:0	R	0x0	Audout fifo rdata [31:0]

AUDOUT_FIFO_PIO_WRH 0xbb

Bit(s)	R/W	Default	Description
31:0	R	0x0	Audout fifo rdata [63:32]

AUD_RESAMPLE_CTRL 0xbf

Bit(s)	R/W	Default	Description
31	RW	0x0	soft_reset: 1 = reset
30:29	RW	0x0	resample source select: 0: fifo0, 1:fifo1, 2:fifo2
28	RW	0x0	resample enable: 0 = disable, 1 = enable
27:26	RW	0x0	resample method select:2,3 reserved.
25:16	RW	0x0	outrdy generate count: equal mclk/lrclk*2
15:0	RW	0x0	average count initial value

AUD_RESAMPLE_CTRL1 0xc0

Bit(s)	R/W	Default	Description
31:0	RW	0x0	phase step

AUD_RESAMPLE_STATUS 0xc1

Bit(s)	R/W	Default	Description
21:0	RW	0x0	report the real frequency of input data

PDM Registers:
Final Address = 0xd0042000+offset*4

PDM_CTRL 0x40

Bit(s)	R/W	Default	Description
2	RW	0x0	CIC_DEC8OR16_SEL, 0:8, 0:16
1	RW	0x0	PDM_HPF_BYPASS
0	RW	0x0	PDM_ENABLE,1 enable

PDM_IN_CTRL 0x42

Bit(s)	R/W	Default	Description
18	RW	0x0	PDMIN_REV
17	RW	0x0	PDMCLK_REV
16	RW	0x0	GET_STA_EN
8	RW	0x0	SAMPLE_CNT

PDM_VOL_CTRL 0x43

Bit(s)	R/W	Default	Description
1	RW	0x0	PDML_INV
0	RW	0x0	PDMR_INV

PDM_VOL_GAIN_L 0x44

Bit(s)	R/W	Default	Description
25:0	RW	0x0	The left channel vol gain

PDM_VOL_GAIN_R 0x45

Bit(s)	R/W	Default	Description
25:0	RW	0x0	The right channel vol gain

PDM_STATUS 0x46

Bit(s)	R/W	Default	Description
31	R	0x0	Clk_cnt _overflow
30:24	R	0x0	Min_clk_cnt
23:16	R	0x0	Max_clk_cnt
15:8	R	0x0	Det_din_mincnt
7:0	R	0x0	Det_din_maxcnt

29. Audio Output

29.1 Overview

This part describe the audio output from the following aspect: SPDIF(encode) I2S(encode) and audio codec, also all the registers related to audio path are listed in the end of this part.

29.2 SPDIF(encode)

The SPDIF(encode) module works in the following 4 steps:

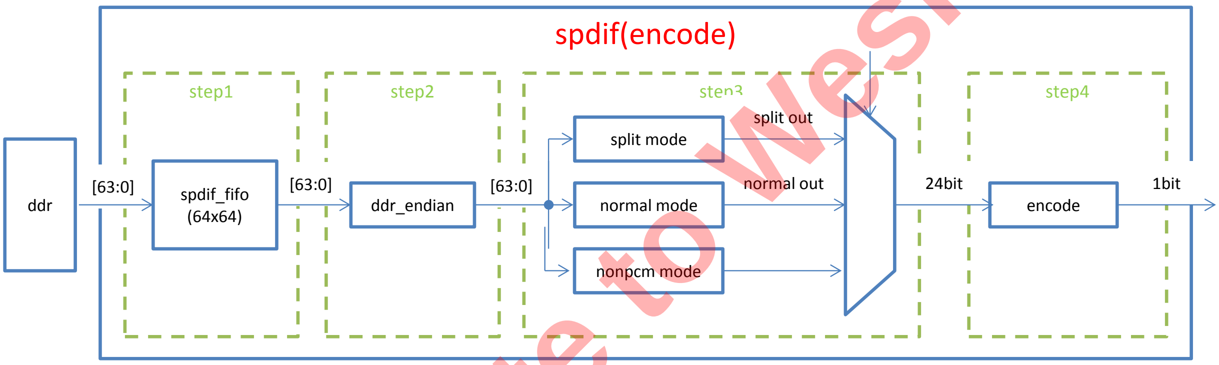
Step1: get the data from ddr.

Step2: write to fifo and make endian modification.

Step3: get the data from fifo by mode select.

Step4: encode.

Fig V.29.1 Diagram of spdif(encode)



SPDIF supports the following mode:

Table V.29.1 SPDIF Mode

	in ddr	
normal	2ch	16Bits**1
	8ch	16Bits**1
	2ch	32Bits**3
	8ch	32Bits**3
split	16Bits**1	
	24Bits**2	
	24Bits(32Bits) **3	
nonpcm	16Bits**4	

**1: In this mode, data in ddr is 16Bits.

**2: In this mode, data in ddr is 2x24Bits.

**3: In this mode, data in ddr is 32Bits.

**4: In this mode, data in ddr is 16Bits.

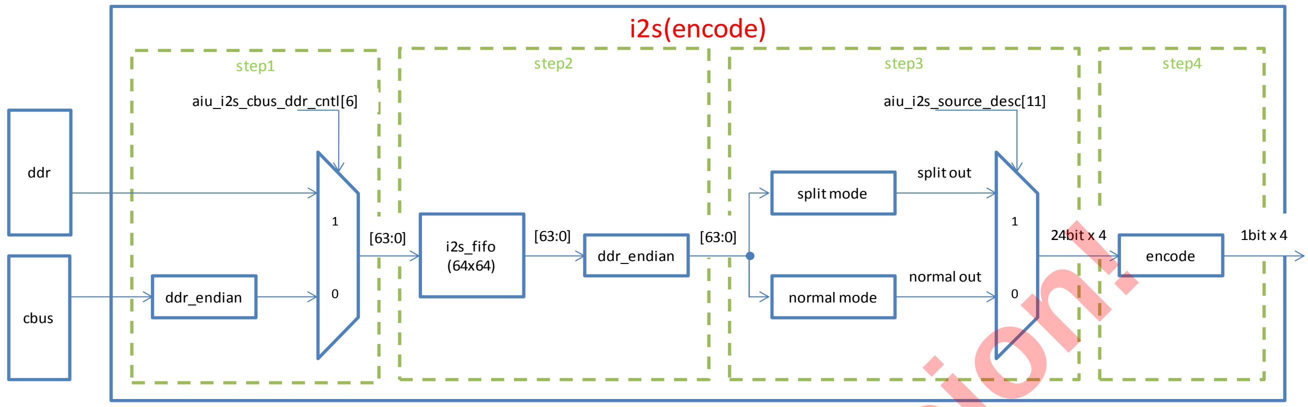
In normal mode, it support 2/8channel & 16Bits/24Bits/32Bits

29.3 I2S(encode)

The I2S(encode) module works in the following 4 steps:

Step1: get the data from ddr or cbus.
Step2: write to fifo and make endian modification.
Step3: get the data from fifo by mode select.
Step4: encode 4 streams.

Fig V.29.2 Diagram of I2S(encode)



Step3, I2S supports the following modes:

Table V.29.2 I2S Mode

	in ddr		output
normal	8ch	16Bits**1	4streams
	8ch	24Bits(32Bits)**2	4streams
split	2ch	16Bits**3	1stream
	8ch	16Bits**3	1stream
	8ch	24Bits(32Bits) **4	1stream
	8ch	24Bits(32Bits) **4	4stream

**1: In this mode, data in ddr is 16Bits,
**2: In this mode, data in ddr is 32Bits,
**3: In this mode, data in ddr is 16Bits,
**4: In this mode, data in ddr is 32Bits,
In normal mode, it support only 8channel & 4streams & 16Bits/24Bits.

The register aiu_mem_I2S_mask[15:0] can skip some read (by [15:8]) and output operation(by [7:0]).
For example:

if [15:8] = 8'b1111_1111, read ch0/ch1/ch2/ch3/ch4/ch5/ch6/ch7
if [15:8] = 8'b1110_1101, read ch0/ch2/ch3/ch5/ch6/ch7,ch1/ch4 skipped
if [7:0] = 8'b1111_1111, output ch0/ch1/ch2/ch3/ch4/ch5/ch6/ch7
if [7:0] = 8'b1110_1101, output ch0/mute/ch2/ch3/mute/ch5/ch6/ch7

In split mode, it will read data from ddr by linear sequence, and support 24Bits(1/4 streams) and 16Bits(1 stream) mode.

In split mode, there is no mask control, so aiu_mem_I2S_mask[15:0] must set 8'hffff_ffff.

I2S_encode_core can encode four I2S sample[23:0] to four 1 bitsignal at the same time.
The control includes:

hold, stop working until set to 1'b0.
underrun,
2'd0: send 0,
2'd1: send 0x800000,

2’d2: send last sample.

med_ctrl, if median work, out = (s1>(s0+s2)/2+th)?(s0+s2)/2:s1, that’s mean if there are a burst noise sample, it will be replaced.

ch_swap: change the channel, assume input = L + R,

2’d0: output = L + R,

2’d1: output = L + L,

2’d2: output = R + R,

2’d3: output = R + L.

sel: use four mux4 to change the connect of in/out

sel0:

2’d0: out0 = in0,

2’d1: out0 = in1,

2’d2: out0 = in2,

2’d3: out0 = in3,

sel1:

2’d0: out1 = in0,

2’d1: out1 = in1,

2’d2: out1 = in2,

2’d3: out1 = in3,

sel2/sel3 are the same.

mute: [0] mute channel 0, [1] mute channel1, …[7] mute channel7

msb_first: 0:lsb first, 1: msb first.

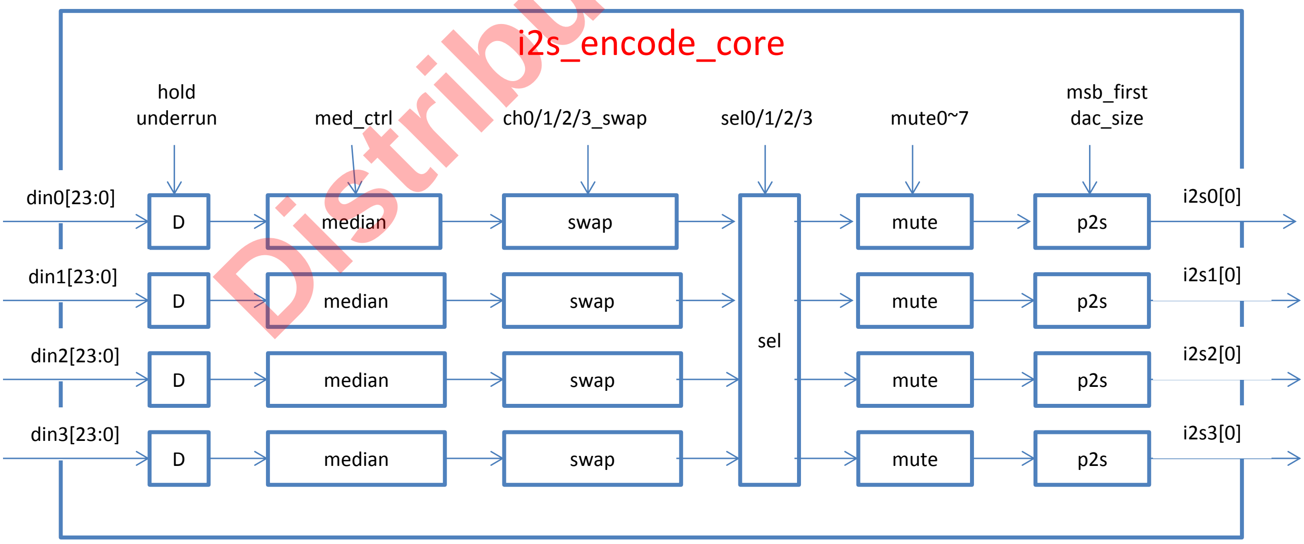
dac_size:

2’d0: output 16Bits,

2’d1: output 20Bits,

2’d2/3: output 24Bits

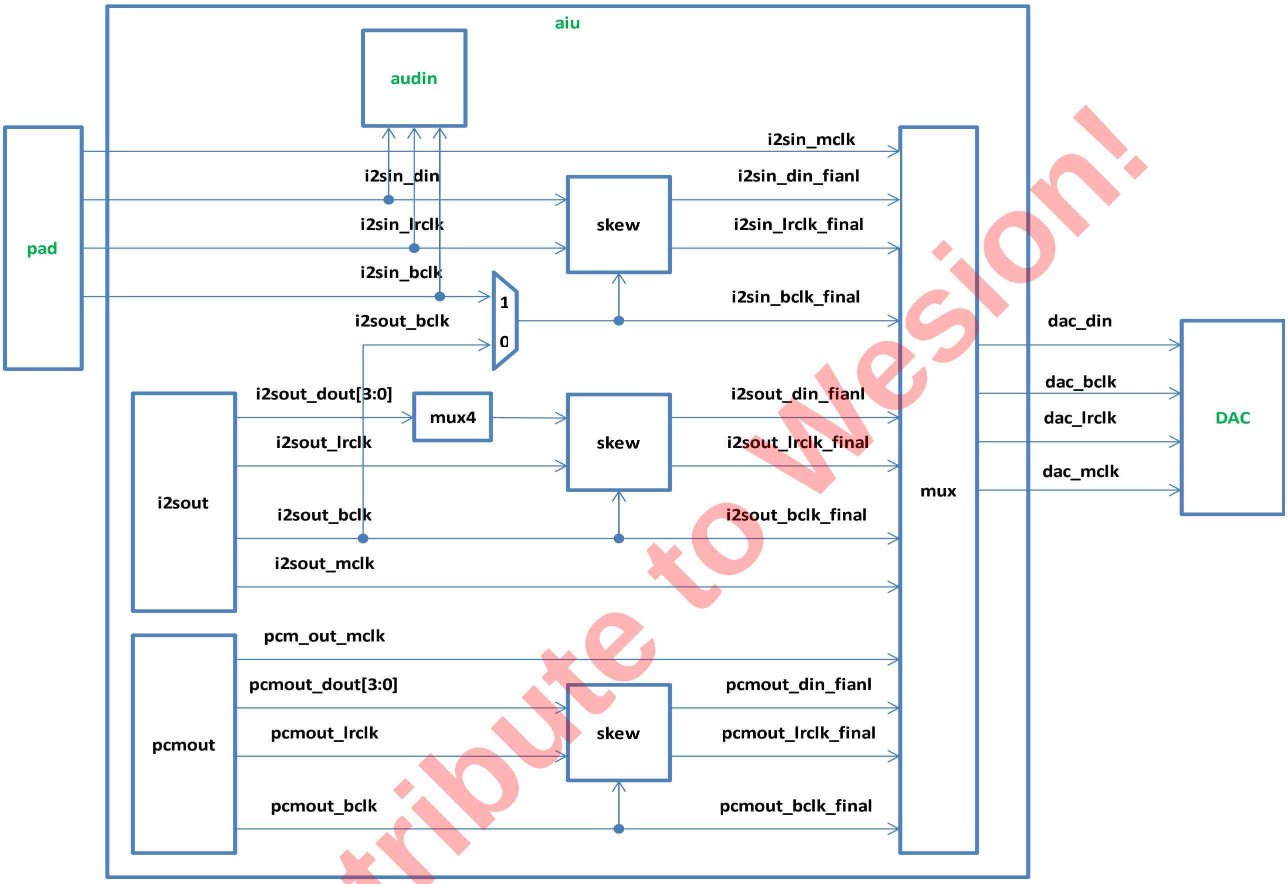
Fig V.29.3 Diagram of I2S encode core



29.4 Audio Codec

In S905X, there is an integrated acodec(DAC), as inllustrated in the Fig below:
As shown in Fig V.31.4, input signals go through MUX submodle, and generate 4 channel of outpur signals.

Fig V.29.4 Diagram of I2S Audio Codec



DAS has the following features:

- DAC Path SNR 97dB (A- weighted)
- DAC Patn THD -86dB at 48kHz Fs
- Up to -92~ 0dB line output gain
- Digital soft mute control
- 24bit D/A, 8K to 96K sampling rate

29.5 Register Description

The final addresses of the following registers should be calculated with the following equation if not stated otherwise:
Final address = 0xc1105400+offset*4

AIU_958_bpf 0x00

Bit(s)	R/W	Default	Description

15:0	RW	0x0	Bytes per frame in spdif.

AIU_958_brst 0x01

Bit(s)	R/W	Default	Description
15:0	RW	0x0	Burst info in spdif

AIU_958_length 0x02

Bit(s)	R/W	Default	Description
15:0	RW	0x0	The length of code in spdif

AIU_958_paddsize 0x03

Bit(s)	R/W	Default	Description
15:0	RW	0xc00	The padd size of spdif

AIU_958_misc 0x04

Bit(s)	R/W	Default	Description
15:14	RW	0x0	pcm_sample_ctl, 00-pcm_no_sample, 01-pcm_sample_up, 10-pcm_sample_down, 11-pcm_sample_down_drop
13	RW	0x0	if true, force each audio data to left or right according to the bit attached with the audio data.This bit should be used with Register AIU_958_force_left(0x505) together
12	RW	0x0	if true, the U(user data) is from the stream otherwise it is filled by zero while encoding iec958 frame
11	RW	0x0	if true big endian(highword,lowword) otherwise little endian(lowword,highword)for 32 bit mode
10:8	RW	0x0	shift number for 32 bit mode
7	RW	0x0	32 bit mode turn on while This bit is true and bit 1 is true
6:5	RW	0x0	Specifies output alignment for 16 bit pcm data. // 00 : dout = {8'b0, din}; // 01 : dout = {4'b0, din, 4'b0}; // 10 : dout = { din, 8'b0};
4	RW	0x0	True if data should be sent out MSB first. LSB first is the default in the spec.
3	RW	0x0	True if msb should be extended (only used with 16 bit pcm data.)
2	RW	0x0	True if msb of PCM data should be inverted.
1	RW	0x0	True if PCM data is 16 Bits wide. False if 24 bit or 32 bit mode.
0	RW	0x1	True if source data is non-PCM data. False if it is PCM data.

AIU_958_discard_num 0x06

Bit(s)	R/W	Default	Description
6:0	R	0x1	how many data discarded in the last dma after one frame data finish transfering to AIU. Should used together with register AIU_958_dcu_ff_ctrl, read only

AIU_958_dcu_ff_ctrl 0x07

Bit(s)	R/W	Default	Description
15:8	RW	0x0	A read from this register indicates the IEC958 FIFO count value
7	RW	0x0	ai_958_req_size if ture, set to 8 Bits interface, used to handle odd frame continous read
6	RW	0x0	continue seeking and dont discard the rest data in one dma after frame end
5	RW	0x0	if true, byte by byte seeking, otherwise word by word seeking
4	RW	0x0	if true, the function for sync head seeking is enabled
3:2	RW	0x0	IEC958 interrupt mode
1	RW	0x0	fifo auto disable, High means after one frame data put into the FIFO, the FIFO will automatically disabled
0	RW	0x0	fifo enable

AIU_958_chstat_l0 0x08

Bit(s)	R/W	Default	Description
31:0	R	0x0	channel status registers for Left channel

AIU_958_chstat_l1 0x09

Bit(s)	R/W	Default	Description
31:0	R	0x0	channel status registers for right channel

AIU_958_ctrl 0x0a

Bit(s)	R/W	Default	Description
9:8	RW	0x0	what to do if there is a fifo underrun 00 => insert 24'h000000 01 => insert mute constant as defined below 10 => repeat last l/r samples
7:5	RW	0x0	mute constant 000 => 24'h000000 001 => 24'h800000 010 => 24'h080000 011 => 24'h008000 100 => 24'h000001 101 => 24'h000010 110 => 24'h000100
4	RW	0x0	mute left speaker
3	RW	0x0	mute right speaker
2:1	RW	0x0	swap channels
0	RW	0x0	Set This bit to hold iec958 interface after the current subframe has been completely transmitted.

AIU_958_rpt 0x0b

Bit(s)	R/W	Default	Description
31:0	RW	0x0	A write operation to this register will cause one of the output samples to be repeated. This can be used to switch the left and the right channels.

AIU_I2S_mute_swap 0x0c

Bit(s)	R/W	Default	Description
15:8	RW	0x0	Mute 8 channel
7:6	RW	0x0	Ch6/7 swap sel.
5:4	RW	0x0	Ch4/5 swap sel.
3:2	RW	0x0	Ch2/3 swap sel.
1:0	RW	0x0	Ch0/1 swap sel.

AIU_I2S_source_desc 0x0d

Bit(s)	R/W	Default	Description
11	RW	0x0	Use_i2s_split
10	RW	0x0	Endian_32bit
9	RW	0x0	Mode_32bit
8:6	RW	0x0	Sft_bits
5	RW	0x0	Steam_16or24b
4:3	RW	0x0	Msb_position
2	RW	0x0	Msb_extend
1	RW	0x0	Msb_invert
0	RW	0x0	Num_streams:0:2ch;1:8ch;

AIU_I2S_med_ctrl 0x0e

Bit(s)	R/W	Default	Description
1	RW	0x0	0=> data is offset binary 1=> data is signed
0	RW	0x0	enable median filter

AIU_I2S_med_thresh 0x0f

Bit(s)	R/W	Default	Description
15:0	RW	0x0	Median filter threshold constant

AIU_I2S_dac_cfg 0x10

Bit(s)	R/W	Default	Description
7	RW	0x0	sign extend sample before downshift
6:4	RW	0x0	payload downshift constant
3	RW	0x0	mute constant 0 => 'h0000000 1 => 'h800000
2	RW	0x0	send msb first
1:0	RW	0x0	Size of payload, 00 => 16 bit, alrclk = aoclk/32 01 => 20 bit, alrclk = aoclk/40 10 => 24 bit, alrclk = aoclk/48 11 => 24 bit, but alrclk = aoclk/64

AIU_I2S_misc 0x12

Bit(s)	R/W	Default	Description
4	RW	0x0	if true, force each audio data to left or right according to the bit attached with the audio data.This bit should be used with Register AIU_I2S_sync(0x511) together
3	RW	0x0	Same Audio source for IEC958 and I2S stream 0, both from I2S buffer
2	RW	0x0	Set This bit to put I2S interface in hold mode
1:0	RW	0x0	How to handle underruns // 00 => send zeros // 01 => send 'h800000 // 10 => repeat last samples

AIU_I2S_out_cfg 0x13

Bit(s)	R/W	Default	Description
7:0	RW	0x0	Audio output config. 2 Bits for each dac, 7:6 for dac3, 5:4 for dac2,3:2 for dac1, 1:0 for dac0 For each 2Bits: 00: connect channel0-1 to the dac 01: connect channel2-3 to the dac 10: connect channel4-5 to the dac 11: connect channel6-7 to the dac

AIU_rst_soft 0x15

Bit(s)	R/W	Default	Description
3	RW	0x0	reset slow domain iec958
2	RW	0x0	soft reset iec958 fast domain
1	RW	0x0	reset slow domain I2S
0	RW	0x0	soft reset I2S fast domain

AIU_clk_ctrl 0x16

Bit(s)	R/W	Default	Description
15	RW	0x0	enable_ddr_arb, set low to reset
14:13	RW	0x0	parser_A_addr_sel 00-A_addr_aififo2, 01-A_addr_iec958, 10-A_addr_aififo, 11-A_addr_I2S

Bit(s)	R/W	Default	Description
12	RW	0x0	958 divisor more, if true, divided by 2, 4, 6, 8
11	RW	0x0	amclk output divisor 0 => dont divide 1 => divide by 2
10	RW	0x0	clock source selection 0 => aiclk from pin 1 => ai_pll_clk from pll
9:8	RW	0x0	alrclk skew 00 => alrclk transitions at the same time msb is sent 01 => alrclk transitions on the cycle before msb is sent 10 => alrclk transitions on the cycle after msb is sent
7	RW	0x0	invert alrclk
6	RW	0x0	invert aoclk
5:4	RW	0x0	958 divisor 00 => divide by 1 01 => divide by 2 10 => divide by 3 11 => divide by 4
3:2	RW	0x0	I2S divisor. NOTE: this value is ignored if AIU_clk_ctrl_more[5:0] != 0 00 => divide by 1 01 => divide by 2 10 => divide by 4 11 => divide by 8
1	RW	0x0	enable 958 divider
0	RW	0x0	enable I2S divider

AIU_mix_adccfg 0x17

Bit(s)	R/W	Default	Description
12	RW	0x0	selects adc input
11:10	RW	0x0	adc size 00 => 16 Bits 01 => 18 Bits 10 => 20 Bits 11 => 24 Bits
9:8	RW	0x0	adc l/r swap mode 00 => stereo 01 => send the right adc input to both l and r speakers 10 => send the left adc input to both l and r speakers 11 => sum the left and right inputs and forward to both speakers
7:5	RW	0x0	adata/lrclk skew mode
4	RW	0x0	1=>invert the adc's lrclk (This is the lrclk going _out_of the chip.
3	RW	0x0	1=>Latch the data on the positive edge of the _internal_aoclk.
2	RW	0x0	1=>adc data is in signed 2's complement mode

AIU_mix_ctrl 0x18

Bit(s)	R/W	Default	Description
12	RW	0x0	if true, toggle each mixed audio data to left or right channel
11	RW	0x0	abuf din left selection, if true, select bit 24 of the data from abuf otherwise select bit 25 of the data from abuf
10:9	RW	0x0	mix sync select, when music, mic and abuf are mixed togather, the main sync source can be selected 00: not sync source 01: music data is the main sync source 10: abuf input data is the main sync source 11: music and abuf togather as the sync source
8	RW	0x0	0=> data from abuf is offset binary 1=> data from abuf is signed
7:6	RW	0x0	the source for data from aiu to abuf 00 => mic 01 => mic saled + abuf scaled

Bit(s)	R/W	Default	Description
			10 => mic scaled + abuf scaled + music scaled 11 => music
5	RW	0x0	channel from aiu to abuf is on
4	RW	0x0	channel from abuf to aiu is on
3	RW	0x0	mic is on
2	RW	0x0	music is on
1	RW	0x0	if true the mixed data are outputed to I2S dac channel,otherwise the mixed data are outputed to IEC958 output
0	RW	0x0	if true music source for mixing is from I2S buffer, otherwise music source is from iec958 buffer

AIU_clk_ctrl_more 0x19

Bit(s)	R/W	Default	Description
15	RW	0x0	invert_audin_sclk
14	RW	0x0	enable_adc_sclk.
13:8	RW	0x0	divisor_adc_sclk.
7	RW	0x0	invert_acodec_adc_sclk
6	RW	0x0	Bit 6 hdmitx_sel_aoclkx2: 0=Select cts_clk_i958 as AIU clk to hdmi_tx_audio_master_clk; 1=Select cts_aoclkx2_int as AIU clk to hdmi_tx_audio_master_clk;
5:0	RW	0x0	More control on I2S divisor. For backward compatiblity, this value is ignored if is 0, if non-zero, it takes effect over AIU_clk_ctrl[3:2]. 0=I2S divisor will use the old value in AIU_clk_ctrl[3:2] (divide by 1/2/4/8) 1=divide by 2; 2=divide by 3; 3=divide by 4; ... and so on ... 63=divide by 64.

AIU_958_pop 0x1a

Bit(s)	R/W	Default	Description
15:0	RW	0x0	A read from this register pops 16 Bits of data off the 958 fifo. A write has no effect.

AIU_mix_gain 0x1b

Bit(s)	R/W	Default	Description
14:10	RW	0x0	mic gain
9:5	RW	0x0	abuf gain
4:0	RW	0x0	music gain

AIU_958_synword1 0x1c

Bit(s)	R/W	Default	Description
15:0	RW	0x0	sync head seeking, word1

AIU_958_synword2 0x1d

Bit(s)	R/W	Default	Description
15:0	RW	0x0	sync head seeking, word2

AIU_958_synword3 0x1e

Bit(s)	R/W	Default	Description
15:0	RW	0x0	sync head seeking, word3

AIU_958_synword1_mask 0x1f

Bit(s)	R/W	Default	Description
15:0	RW	0x0	sync head seeking, mask word1

AIU_958_synword2_mask 0x20

Bit(s)	R/W	Default	Description
15:0	RW	0x0	sync head seeking, mask word 2

AIU_958_synword3_mask 0x21

Bit(s)	R/W	Default	Description
15:0	RW	0x0	sync head seeking, mask word 3

AIU_958_ffrdout_thd 0x22

Bit(s)	R/W	Default	Description
15:0	RW	0x0	fifo read-out threshold, one condition to generate interrupt is met after fifo readout counter reach this value in a frame, please refer to register AIU_958_dcu_ff_ctrl

AIU_958_length_per_pause 0x23

Bit(s)	R/W	Default	Description
15:0	RW	0x0	For pause burst sequence adding, one pause burst sequence is consist of a serious pause burst.

AIU_958_pause_num 0x24

Bit(s)	R/W	Default	Description
15	RW	0x0	if true, one pause burst sequence will be added
14:0	RW	0x0	the number of pause burst in a pause burst sequence

AIU_958_pause_payload 0x25

Bit(s)	R/W	Default	Description
15:0	RW	0x0	When paused, use this payload .

AIU_958_auto_pause 0x26

Bit(s)	R/W	Default	Description
15	RW	0x0	if true, auto pause function enable
14	RW	0x0	pause pack option, just for debugging and adding one option
7:0	RW	0x0	auto_pause threshold

AIU_958_pause_pd_length 0x27

Bit(s)	R/W	Default	Description
15:0	RW	0x0	Pause pd length

AIU_CODEC_DAC_LRCLK_CTRL 0x28

Bit(s)	R/W	Default	Description
15:12		0x0	Reserved
11:0	RW	0x0	dac_lrclk_div

AIU_CODEC_ADC_LRCLK_CTRL 0x29

Bit(s)	R/W	Default	Description
15:14		0x0	Reserved
13	RW	0x0	inv_audin_lrclk: whether to invert lrclk before output to Audin

12	RW	0x0	inv_acodec_adc_lrclk: whether to invert lrclk before output to Audio Codec
11:0	RW	0x0	0 adc_lrclk_div

AIU_HDMI_CLK_DATA_CTRL 0x2a

Bit(s)	R/W	Default	Description
15:6		0x0	Reserved
5:4	RW	0x0	hdmi_data_sel: 00=output 0, disable hdmi data; 01=Select pcm data; 10=Select AIU I2S data; 11=Not allowed
3:2		0x0	Reserved
1:0	RW	0x0	hdmi_clk_sel: 00=Disable output hdmi clock; 01=Select pcm clock; 10=Select AIU clk; 11=Not allowed.

AIU_CODEC_CLK_DATA_CTRL 0x2b

Bit(s)	R/W	Default	Description
15:6		0x0	Reserved
5:4	RW	0x0	acodec_data_sel: 00=output 0, disable acodec_sdin; 01=Select pcm data; 10=Select AIU I2S data; 11=Not allowed.
3:2		0x0	Reserved
1:0	RW	0x0	acodec_clk_sel: 00=Disable output acodec_sclk; 01=Select pcm clock; 10=Select AIU aoclk; 11=Not allowed

AIU_958_chstat_r0 0x30

Bit(s)	R/W	Default	Description
15:0	RW	0x0	contains Bits 15:0 of the channel status word. Note that bit zero of the channel status word is sent out first.chstat_r1[15:0] contains Bits 31:16 of the channel status word

AIU_958_chstat_r1 0x31

Bit(s)	R/W	Default	Description
15:0	RW	0x0	contains Bits 15:0 of the channel status word. Note that bit zero of the channel status word is sent out first.chstat_r1[15:0] contains Bits 31:16 of the channel status word

AIU_958_valid_ctrl 0x32

Bit(s)	R/W	Default	Description
1	RW	0x0	if true, turn on Digital output Valid control
0	RW	0x0	0: output 0, 1: output 1 to the valid bit in audio digital output when bit 1 is true

AIU_AIFIFO2_CTRL 0x40

Bit(s)	R/W	Default	Description
3	RW	0x0	CRC pop aififo2 enable
2	RW	0x0	writing to This bit to 1 causes CRC module reset
1	RW	0x0	unused
1	RW	0x0	writing to This bit to 1 causes AIFIFO2 soft reset

AIU_AIFIFO2_STATUS 0x41

Bit(s)	R/W	Default	Description
4:0	R	0x0	how many Bits left in the first pop register

AIU_AIFIFO2_GBIT 0x42

Bit(s)	R/W	Default	Description
15:0	R	0x0	The gb data

AIU_AIFIFO2_CLB 0x43

Bit(s)	R/W	Default	Description
15:0	R	0x0	The gb data

AIU_CRC_CTRL 0x44

Bit(s)	R/W	Default	Description
13:8	RW	0x10	CRC polynomial equation order, between 1 to 32
3	RW	0x0	CRC pop data from FIFO enable
2	RW	0x0	CRC input register clear
1	RW	0x0	CRC core soft reset
0	RW	0x0	CRC caculation start

AIU_CRC_STATUS 0x45

Bit(s)	R/W	Default	Description
7:4	RW	0x0	CRC internal shift register bit select, just for debug purpose
3	RW	0x0	CRC internal shift register data valid, just for debug purpose
2	RW	0x0	CRC input register data valid
1	RW	0x0	CRC result, 1: CRC not correct, 0: CRC correct
0	RW	0x0	CRC state, 1: CRC busy, 0: CRC idle

AIU_CRC_SHIFT_REG 0x46

Bit(s)	R/W	Default	Description
15:0	R	0x0	CRC internal shift register, read only, for debug purpose

AIU_CRC_IREG 0x47

Bit(s)	R/W	Default	Description
15:0	RW	0x0	CRC data input register

AIU_CRC_CAL_REG1 0x48

Bit(s)	R/W	Default	Description
15:0	RW	0x0	CRC calculation register high-bit part [31:16]

AIU_CRC_CAL_REG0 0x49

Bit(s)	R/W	Default	Description
15:0	RW	0x0	CRC calculation register high-bit part [15:0]

CRC polynomial coefficient high-bit part [31:16], read/write

AIU_CRC_POLY_COEF1 0x4a

Bit(s)	R/W	Default	Description
15:0	RW	0x0	CRC calculation register high-bit part [15:0]

AIU_CRC_POLY_COEF0 0x4b

Bit(s)	R/W	Default	Description
15:0	RW	0x0	CRC polynomial coefficient low-bit part [15:0]

AIU_CRC_BIT_SIZE1 0x4c

Bit(s)	R/W	Default	Description
3:0	RW	0x0	CRC frame size, high-bit part [19:16]

AIU_CRC_BIT_SIZE0 0x4d

Bit(s)	R/W	Default	Description
15:0	RW	0x0	CRC frame size, low-bit part [15:0]

AIU_CRC_BIT_CNT1 0x4e

Bit(s)	R/W	Default	Description
3:0	R	0x0	How many Bits have been processed right now in the current frame [19:16]

AIU_CRC_BIT_CNT0 0x4f

Bit(s)	R/W	Default	Description
15:0	R	0x0	How many Bits have been processed right now in the current frame [15:0]

AIU_AMCLK_GATE_HI 0x50

Bit(s)	R/W	Default	Description
3	RW	0x0	Start msr clk
2:0	RW	0x0	The msr period [18:16]

AIU_AMCLK_GATE_LO 0x51

Bit(s)	R/W	Default	Description
15:0	RW	0x0	The msr period [15:0]

AIU_AMCLK_MSR 0x52

Bit(s)	R/W	Default	Description
15:0	R	0x0	The msr count

AIU_MEM_I2S_START_PTR 0x60

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The start address from DDR

AIU_MEM_I2S_RD_PTR 0x61

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The new read access address from DDR

AIU_MEM_I2S_END_PTR 0x62

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The end address from DDR

AIU_MEM_I2S_MASKS 0x63

Bit(s)	R/W	Default	Description
31:16	RW	0x0	IRQ block
15:8	RW	0x0	chan_mem_mask. Each Bit indicates which channels exist in memory
7:0	RW	0x0	chan_rd_mask. Each Bit indicates which channels are READ from memory

AIU_MEM_I2S_CONTROL 0x64

Bit(s)	R/W	Default	Description
11:10	RW	0x0	Select which hardware pointer to use to control the buffer level: 00 = parser 01 = audin_fifo0_wrpt 1x = audin_fifo1_wrpt

Bit(s)	R/W	Default	Description
9	RW	0x0	Use level control: 1 = use buffer level control
8	R	0x0	Read Only. This bit is 1 when there is data available for reading
7	R	0x0	Read only. 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.
6		0x0	cntl_mode_16bit:Set to 1 for 16 bit storage format in DDR
5:3	RW	0x0	endian
2	RW	0x0	cntl_empty_en Set to 1 to enable reading data from the FIFO
1	RW	0x0	cntl_fill_en Set to 1 to enable reading data from DDR memory
0	RW	0x0	cntl_init: After setting the read pointers, sizes, channel masks and read masks, set This bit to 1 and then to 0

AIU_MEM_IEC958_START_PTR 0x65

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The start address from DDR

AIU_MEM_IEC958_RD_PTR 0x66

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The read access address from DDR

AIU_MEM_IEC958_END_PTR 0x67

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The end address from DDR

AIU_MEM_IEC958_MASKS 0x68

Bit(s)	R/W	Default	Description
15:8	RW	0x0	chan_mem_mask. Each Bit indicates which channels exist in memory
7:0	RW	0x0	chan_rd_mask. Each Bit indicates which channels are READ from memory

AIU_MEM_IEC958_CONTROL 0x69

Bit(s)	R/W	Default	Description
31	RW	0x0	A_urgent
30	RW	0x0	ch_always_8
27:24	RW	0x0	rdata_rd_base_begin (used for select from different channel)
23:14		0x0	reserved
13	RW	0x0	cntl_sim_en
12	RW	0x0	cntl_use_level
11	R	0x0	Read only. This bit will be set to 1 when there is data in the FIFO to process
10	R	0x0	Read only. This bit will be high when we're fetching data from the DDR memory
9	RW	0x0	cntl_endian_jic Just in case endian. last minute byte swap of the data out of the FIFO to the rest of the IEC958 logic
8	RW	0x0	Set This bit to 1 to tell the IEC958 FIFO to read and process data linearly for raw data.
7	RW	0x0	cntl_mode_16bit:Set to 1 for 16 bit storage format in DDR. Only valid when mode_raw = 0
6	RW	0x0	cntl_rd_ddr Set This bit to read if you want AIU_MEM_IEC958_RD_PTR and AIU_MEM_IEC958_RD_PTR_HIGH to refer to the pointer into DDR memory.Otherwise, the curr_ptr,registers refer to the byte address of the data at the output of the FIFO to the rest of the IEC958 logic
5:3	RW	0x0	endian
2	RW	0x0	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	RW	0x0	cntl_fill_en Set to 1 to enable reading data from DDR memory
0	RW	0x0	cntl_init

AIU_MEM_AIFIFO2_START_PTR 0x6a

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The start address from DDR

AIU_MEM_AIFIFO2_CURR_PTR 0x6b

Bit(s)	R/W	Default	Description
31:0	R	0x0	The current address from DDR

AIU_MEM_AIFIFO2_END_PTR 0x6c

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The end address from DDR

AIU_MEM_AIFIFO2_BYTES_AVAIL 0x6d

Bit(s)	R/W	Default	Description
31:0	R	0x0	Available bytes in afifo2.

AIU_MEM_AIFIFO2_CONTROL 0x6e

Bit(s)	R/W	Default	Description
15:11		0x0	unused
10	RW	0x0	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	RW	0x0	Data Ready. This bit is set when data can be popped
8	RW	0x0	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.
7	RW	0x0	cntl_endian_jic Just in case endian. last minute byte swap of the data out of the FIFO to get bit
6		0x0	unused
5：3	RW	0x0	endian
2	RW	0x0	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	RW	0x0	cntl_fill_en Set to 1 to enable reading data from DDR memory
0	RW	0x0	cntl_init: After setting the read pointers, sizes, channel masks and read masks, set This bit to 1 and then to 0

AIU_MEM_AIFIFO2_MAN_WP 0x6f

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Manual write ptr.

AIU_MEM_AIFIFO2_MAN_RP 0x70

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Manual read ptr.

AIU_MEM_AIFIFO2_LEVEL 0x71

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Level = reg[31:0] – level_hold

AIU_MEM_AIFIFO2_BUF_CNTL 0x72

Bit(s)	R/W	Default	Description
1	RW	0x0	Set to 1 for manual write pointer mode
0	RW	0x0	Set high then low after everything has been initialized

AIU_MEM_I2S_MAN_WP 0x73

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Manual write ptr.

AIU_MEM_I2S_MAN_RP 0x74

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Manual read ptr.

AIU_MEM_I2S_LEVEL 0x75

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Level = reg[31:0] – level_hold

AIU_MEM_I2S_BUF_CNTL 0x76

Bit(s)	R/W	Default	Description
1	RW	0x0	mode 0 = parser (or audin_fifo0 or audin_fifo1), 1 for manual write pointer
0	RW	0x0	initialize Set high then low after everything has been initialized

AIU_MEM_I2S_BUF_WRAP_COUNT 0x77

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The wrap count

AIU_MEM_I2S_MEM_CTL 0x78

Bit(s)	R/W	Default	Description
29:24	RW	0x0	A_brst_num
21:16	RW	0x0	A_id
15:0	RW	0x0	level_hold

AIU_MEM_IEC958_MEM_CTL 0x79

Bit(s)	R/W	Default	Description
29:24	RW	0x0	A_brst_num
21:16	RW	0x0	A_id
15:0	RW	0x0	level_hold

AIU_MEM_IEC958_WRAP_COUNT 0x7a

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The wrap count

AIU_MEM_IEC958_IRQ_LEVEL 0x7b

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The irq level

AIU_MEM_IEC958_MAN_WP 0x7c

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Manual write ptr.

AIU_MEM_IEC958_MAN_RP 0x7d

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Manual read ptr.

AIU_MEM_IEC958_LEVEL 0x7e

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Level = reg[31:0] – level_hold

AIU_MEM_IEC958_BUF_CNTL 0x7f

Bit(s)	R/W	Default	Description
29:24	RW	0x0	A_brst_num
21:16	RW	0x0	A_id
15:0	RW	0x0	level_hold

AIU_AIFIFO_CTRL 0x80

Bit(s)	R/W	Default	Description
3	RW	0x0	CRC pop aififo enable
2	RW	0x0	writing to This bit to 1 causes CRC module reset
1	RW	0x0	enable aififo
0	RW	0x0	writing to This bit to 1 causes aififo soft reset

AIFIFO_STATUS 0x81

Same fucntion 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

Bit(s)	R/W	Default	Description
13	RW	0x0	aififo request to dcu status
12	RW	0x0	dcu select status
11:5	RW	0x0	aififo word counter number
4:0	RW	0x0	how many Bits left in the first pop register

AIFIFO_GBIT 0x82

Same function as the AICLB of AIFIFO in CDROM module return the leading zeros by reading this registers

Bit(s)	R/W	Default	Description
15:0	R	0x0	The gb data

AIFIFO_CLB 0x83

Bit(s)	R/W	Default	Description
15:0	R	0x0	The gb data

MEM_AIFIFO_START_PTR 0x84

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The start address from DDR

MEM_AIFIFO_CURR_PTR 0x85

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The current address from DDR

MEM_AIFIFO_END_PTR 0X86

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The start address from DDR

MEM_AIFIFO_BYTES_AVAIL 0x87

Bit(s)	R/W	Default	Description
31:0	RW	0x0	The available bytes.

AIU_MEM_AIFIFO_CONTROL 0x88

Bit(s)	R/W	Default	Description
15:11		0x0	unused
10	RW	0x0	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	RW	0x0	Data Ready. This bit is set when data can be popped
8	RW	0x0	fill busy This bit will be high when we're fetching data from the DDR memory
7	RW	0x0	cntl_endian_jic Just in case endian. last minute byte swap of the data out of the FIFO to get bit
6	RW	0x0	unused
5：3	RW	0x0	endian
2	RW	0x0	cntl_empty_en Set to 1 to enable reading the DDR memory FIFO and filling the pipeline to get-bit
1	RW	0x0	cntl_fill_en Set to 1 to enable reading data from DDR memory
0	RW	0x0	cntl_init

AIU_MEM_AIFIFO_MAN_WP 0x89

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Manual write ptr.

AIU_MEM_AIFIFO_MAN_RP 0x8a

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Manual read ptr.

AIU_MEM_AIFIFO_LEVEL 0x8b

Bit(s)	R/W	Default	Description
31:0	RW	0x0	Level = reg[31:0] – level_hold

AIU_MEM_AIFIFO_BUF_CNTL 0x8c

Bit(s)	R/W	Default	Description
1	RW	0x0	manual mode Set to 1 for manual write pointer mode
0	RW	0x0	Init Set high then low after everything has been initialized

AIU_MEM_AIFIFO_BUF_WRAP_COUNT 0x8d

Bit(s)	R/W	Default	Description
31:0	R	0x0	Fifo wrap count

AIU_MEM_AIFIFO2_BUF_WRAP_COUNT 0x8e

Bit(s)	R/W	Default	Description
31:0	R	0x0	Fifo2 wrap count

AIU_MEM_AIFIFO_MEM_CTL ox8f

Bit(s)	R/W	Default	Description
29:24	RW	0x0	A_brst_num
21:16	RW	0x0	A_id
15:0	RW	0x0	level_hold

AIFIFO_TIME_STAMP_CNTL 0x90

Bit(s)	R/W	Default	Description
31:16	RW	0x0	drop_bytes
15:14	RW	0x0	drop_status (Read-Only)
13:12	RW	0x0	sync_match_position (Read-Only)
11:6		0x0	reserved
5:4	RW	0x0	TIME_STAMP_NUMBER, 0-32Bits, 1-64Bits, 2-96Bits, 3-128Bits
3	RW	0x0	stamp_soft_reset
2	RW	0x0	TIME_STAMP_length_enable
1	RW	0x0	TIME_STAMP_sync64_enable
0	RW	0x0	TIME_STAMP_enable

AIFIFO_TIME_STAMP_SYNC_0 0x91

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_SYNC_CODE_0

AIFIFO_TIME_STAMP_SYNC_1 0x92

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_SYNC_CODE_1

AIFIFO_TIME_STAMP_0 0x93

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_0

AIFIFO_TIME_STAMP_1 0x94

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_1

AIFIFO_TIME_STAMP_2 0x95

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_2

AIFIFO_TIME_STAMP_3 0x96

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_3

AIFIFO_TIME_STAMP_LENGTH 0x97

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_LENGTH

AIFIFO2_TIME_STAMP_CNTL 0x98

Bit(s)	R/W	Default	Description
31：16	RW	0x0	drop_bytes
15：14	RW	0x0	drop_status (Read-Only)
13：12	RW	0x0	sync_match_position (Read-Only)
11：6		0x0	reserved
5：4	RW	0x0	TIME_STAMP_NUMBER, 0-32Bits, 1-64Bits, 2-96Bits, 3-128Bits
3	RW	0x0	stamp_soft_reset
2	RW	0x0	TIME_STAMP_length_enable

Bit(s)	R/W	Default	Description
1	RW	0x0	TIME_STAMP_sync64_enable
0	RW	0x0	TIME_STAMP_enable

AIFIFO2_TIME_STAMP_SYNC_0 0x99

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_SYNC_CODE_0

AIFIFO2_TIME_STAMP_SYNC_1 0x9a

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_SYNC_CODE_1

AIFIFO2_TIME_STAMP_0 0x9b

Bit(s)	R/W	Default	Description
31：0	RW	0x0	TIME_STAMP_0

AIFIFO2_TIME_STAMP_1 0x9c

Bit(s)	R/W	Default	Description
31：0	RW	0x0	TIME_STAMP_1

AIFIFO2_TIME_STAMP_2 0x9d

Bit(s)	R/W	Default	Description
31：0	RW	0x0	TIME_STAMP_2

AIFIFO2_TIME_STAMP_3 0x9e

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_3

AIFIFO2_TIME_STAMP_LENGTH 0x9f

Bit(s)	R/W	Default	Description
31：0	RW	0x0	TIME_STAMP_LENGTH

IEC958_TIME_STAMP_CNTL 0xa0

Bit(s)	R/W	Default	Description
31：16	RW	0x0	drop_bytes
15：14	RW	0x0	drop_status (Read-Only)
13：12	RW	0x0	sync_match_position (Read-Only)
11：6		0x0	reserved
5：4	RW	0x0	TIME_STAMP_NUMBER, 0-32Bits, 1-64Bits, 2-96Bits, 3-128Bits
3	RW	0x0	stamp_soft_reset
2	RW	0x0	TIME_STAMP_length_enable
1	RW	0x0	TIME_STAMP_sync64_enable
0	RW	0x0	TIME_STAMP_enable

IEC958_TIME_STAMP_SYNC_0 0xa1

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_SYNC_CODE_0

IEC958_TIME_STAMP_SYNC_1 0xa2

Bit(s)	R/W	Default	Description

31:0	RW	0x0	TIME_STAMP_SYNC_CODE_1

IEC958_TIME_STAMP_0 0xa3

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_0

IEC958_TIME_STAMP_1 0xa4

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_1

IEC958_TIME_STAMP_2 0xa5

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_2

IEC958_TIME_STAMP_3 0xa6

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_3

IEC958_TIME_STAMP_LENGTH 0xa7

Bit(s)	R/W	Default	Description
31:0	RW	0x0	TIME_STAMP_LENGTH

AIU_MEM_AIFIFO2_MEM_CTL 0xa8

Bit(s)	R/W	Default	Description
29:24	RW	0x0	A_brst_num
21:16	RW	0x0	A_id
15:0	RW	0x0	level_hold

AIU_I2S_CBUS_DDR_CNTL 0xa9

Bit(s)	R/W	Default	Description
31:26		0x0	unused
25	RW	0x0	A_req level
24	RW	0x0	data_req If This bit is 1, then (a_req_cnt != 'h0)
23：16	RW	0x0	a_req_cnt This value corresponds to the number of32-bit words
15:7		0x0	unused
6	RW	0x0	Set This bit to mux in the cbus_ddr_interface
5	RW	0x0	Set This bit to allow back to back A_req's to be serviced
4	RW	0x0	Set This bit to generate an IRQ on the first A_req
3：1	RW	0x0	Endian
0	RW	0x0	Set This bit enable the cbus_ddr_interface

AIU_I2S_CBUS_DDR_WDATA 0xaa

Bit(s)	R/W	Default	Description
31:0	RW	0x0	32-bit data to write to the cbus_ddr interface

AIU_I2S_CBUS_DDR_ADDR 0xab

Bit(s)	R/W	Default	Description
31:0	RW	0x0	First address associated with the first request by the I2S_fast() to read DDR data

The following register is for DAC control, and the final address is calculated with the following equation:
Final address = 0xc1100000 + offset*4

Acodec_ctrl 0x 152c

Bit(s)	R/W	Default	Description
15			enable of din, 0: all is 0; 1: all start work
14			clk_inv, 0: no change; 1: invert dac_bclk
13			lrclk_inv, 0: no change; 1: invert i2sin_lrclk/i2sout_lrclk/pcmout_lrclk;
11			i2sin_bclk_src, 0: use i2sout_bclk; 1: use i2sin_lrclk;
7~6			dac_din_lrclk_sel; 0: fix 0; 1: i2sout; 2: pcmout; 3: i2sin;
5~4			dac_bclk_mclk_sel; 0: fix 0; 1: i2sout; 2: pcmout; 3: i2sin;
3~2			din_skew; 0: no change; 1:lrclk add one delay; 2: din add one delay; 3: no change;
1~0			i2sout_src_sel; 0:dout[0];1:dout[1];2:dout[2];3:dout[3];

Section VI Memory INTERFACE

This part describes S905X's memory interfaces from the following aspects:

- DDR
- NAND
- EMMC/SDIO/SD
- SPICC
- SPIFC

30. DDR

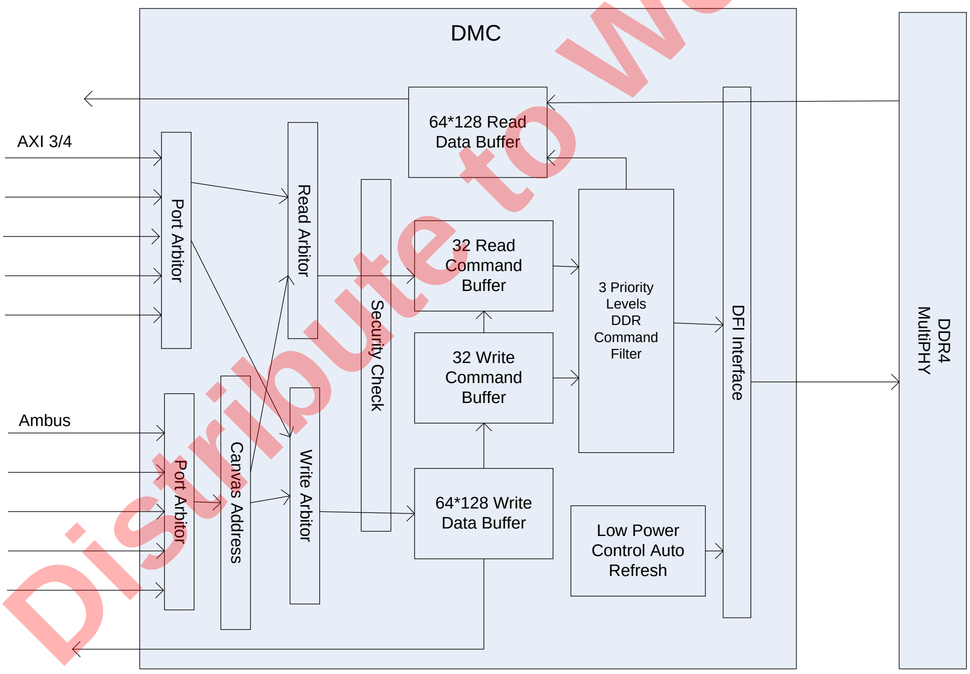
30.1 Overview

DDR memory interface consists of the 2 parts: DDR memory controller (DMC) and DDR PHY controller. The main features of this module are listed below:

- Electronic fence protected Security system.
- Optimized DDR read reordering to improve DDR efficiency.
- Support either 16b or 32b DRAM interface.
- VPU ports changed to AXI like 128Bits interface
- READ/write bus are separated.
- Device port directly to AXI arbitration. No canvas address translation latency.
- PCTL moved inside DMC to optimize DDR command generation.
- Improved DDR Frequency update 1200Mhz(DDR3/LPDDR3 2400MBPS).

Below is the diagram of DDR modul, in which AXI3 and AXI3-like are the interface (ports) to the DMC, signals will go through the Arbiter to the DDR Commend Generator to generate DDR commend signal, and through DDR Phy Controler to execute read or write to DDR.

Fig VI.30.1 Diagram of DDR Interface



30.2 Register Description

DMC Register spec.

DMC unsecure register. Base address 0xc8838000. Each register takes 4 byte address.

Each register's final address = 0xc8838000 + offset * 4.

DMC_REQ_CTRL **0x0**

Bit(s)	R/W	Default	Description
31~16	R/W	0	Not used.

Bit(s)	R/W	Default	Description
15	R/W	0	enable dmc request of port15. GE2D interface. Async interface.
14	R/W	0	enable dmc request of port 14. DOS HCODEC interface Sync interface.
13	R/W	0	enable dmc request of port 13. DOS VDEC interface Sync interface..
12	R/W	0	enable dmc request of port 12. VPU write interface 1 Sync interface.
11	R/W	0	enable dmc request of port 11. VPU write interface 0 Sync interface.
10	R/W	0	enable dmc request of port 10. VPU read interface 2. Sync interface.
9	R/W	0	enable dmc request of port 9. VPU read interface 1. Sync interface
8	R/W	0	enable dmc request of port 8. VPU read interface 0. Sync interface.
7	R/W	0	enable dmc request of port 7. DEVICE. Async interface.
6	R/W	0	enable dmc request of port 6. not used.
5	R/W	0	enable dmc request of port 5. not used.
4	R/W	0	enable dmc request of port 4. HEVC sync interface.
3	R/W	0	enable dmc request of port 3. HDCP/HDMI 32Bits. Async interface..
2	R/W	0	enable dmc request of port 2. Mali 1 Sync interface..
1	R/W	0	enable dmc request of port 1. Mali 0. Sync interface.
0	R/W	0	enable dmc request of port 0. CPU/A53 Sync interface.

DMC_SOFT_RST **0x01**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29	R/W	0	DMC test soft reset_n. 0 : reset. 1 : normal working mode.
28	R/W	0	DMC low power control moudle soft reset_n. 0 : reset. 1 : normal working mode.
27	R/W	0	DMC QOS monitor module soft reset_n. 0 : reset. 1 : normal working mode.
26	R/W	0	DMC register modle soft reset_n. 0 : reset. 1 : normal working mode.
25	R/W	0	DMC canvas transfer module soft reset_n. 0 : reset. 1 : normal working mode..
24	R/W	0	DMC command buffers and command generation modules soft reset. 0 = reset. 1:
23~17	R/W	0	Not used.
16	R/W	0	DDR channel 0 PCTL module n_clk domain soft reset_n. 0 : reset. 1 : normal working mode.
15:0	R/W	0	16 input chan inteface n_clk domain reset control. if the channel is asynchronous FIFO interface, then both side of the clocks must be turn off before reset this module. If the channel is synchronous interface, you can reset it any time.
15	R/W	0	n_clk domain port 15 soft reset_n control. 0 : reset. 1: normal working mode.
14	R/W	0	n_clk domain port 14 soft reset_n control. 0 : reset. 1: normal working mode.
13	R/W	0	n_clk domain port 13 soft reset_n control. 0 : reset. 1: normal working mode.
12	R/W	0	n_clk domain port 12 soft reset_n control. 0 : reset. 1: normal working mode.
11	R/W	0	n_clk domain port 11 soft reset_n control. 0 : reset. 1: normal working mode.
10	R/W	0	n_clk domain port 10 soft reset_n control. 0 : reset. 1: normal working mode.
9	R/W	0	n_clk domain port 9 soft reset_n control. 0 : reset. 1: normal working mode.
8	R/W	0	n_clk domain port 8 soft reset_n control. 0 : reset. 1: normal working mode.
7	R/W	0	n_clk domain port 7 soft reset_n control. 0 : reset. 1: normal working mode.
6	R/W	0	n_clk domain port 6 soft reset_n control. 0 : reset. 1: normal working mode.
5	R/W	0	n_clk domain port 5 soft reset_n control. 0 : reset. 1: normal working mode.
4	R/W	0	n_clk domain port 4 soft reset_n control. 0 : reset. 1: normal working mode.
3	R/W	0	n_clk domain port 3 soft reset_n control. 0 : reset. 1: normal working mode.
2	R/W	0	n_clk domain port 2 soft reset_n control. 0 : reset. 1: normal working mode.
1	R/W	0	n_clk domain port 1 soft reset_n control. 0 : reset. 1: normal working mode.
0	R/W	0	n_clk domain port 0 soft reset_n control. 0 : reset. 1: normal working mode.

DMC_SOFT_RST1 **0x02**

Bit(s)	R/W	Default	Description
31~16	R/W	0	Not used
15~0	R/W	0	if the input port interface is asynchronous interface, then the related bit is for the main clock domain reset control. if the interface is synchronouse interface, This bit is not used.
14~8	R/W	0	Not used
7	R/W	0	input port 7 main clock domain soft reset_n.
6~4	R/W	0	Not used
3	R/W	0	input chan 3 main clock domain soft reset_n.
2~0	R/W	0	Not used

DMC_RST_STS1 **0x04**

Bit(s)	R/W	Default	Description
31~16	R/W	0	Not used.
15~0	R	0	Read only. the DMC_SOFT_RST1 signal in n_clk domain. the purpose of this register is if the n_clk is too fast or too slow related to APB clock, we can read this register to make sure another clock domain reset is done.

DMC_VERSION **0x05**

Bit(s)	R/W	Default	Description
31~0	R	0x000a001	Read only.

DMC_RAM_PD **0x11**

Bit(s)	R/W	Default	Description
31~6	R/W	0	Not used.
5:4	R/W	0	DDR channel1 read/write data path SRAM in power down mode. 2'b11: power down. 2'b00 working mode.
3:2	R/W	0	DDR channel0 read/write data path SRAM in power down mode. 2'b11: power down. 2'b00 working mode
1:0	R/W	0	CANVAS LUT SRAM in power down mode control. 2'b11: power down. 2'b00 working mode.

DMC_CAV_LUT_DATAL **0x12**

Bit(s)	R/W	Default	Description
31~0	R/W	0	low 32 Bits of canvas data which need to be configured to canvas LUT memory

DMC_CAV_LUT_DATAH **0x13**

Bit(s)	R/W	Default	Description
31~0	R/W	0	high 32Bits of cavnas data which need to be configured to canvas memory. 64Bits CANVAS look up table bit 61:58 Endian control. bit 61: 1 : switch 2 64Bits data inside 128Bits boundary. 0 : no change. bit 60: 1 : switch 2 32Bits data inside 64Bits data boundary. 0 : no change. bit 59: 1 : switch 2 16Bits data inside 32Bits data boundary. 0 : no change. bit 58: 1 : switch 2 8Bits data inside 16Bits data bournday. 0 : no change. bit 57:56. Canvas block mode. 2 : 64x32, 1: 32x32; 0 : linear mode. bit 55: canvas Y direction wrap control. 1: wrap back in y. 0: not wrap back. bit 54: canvas X direction wrap control. 1: wrap back in X. 0: not wrap back. bit 53:41. canvas Hight. bit 40:29. canvas Width, unit: 8bytes. must in 32bytes boundary. that means last 2 Bits must be 0. bit 28:0. cavnas start address. unit. 8 bytes. must be in 32bytes boundary. that means last 2Bits must be 0.

DC_CAV_LUT_ADDR **0x14**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.

Bit(s)	R/W	Default	Description
9~8	R/W	0	write 2'b10. the canvas data will saved in canvas memory with addres 7:0
7~0	R/W	0	256 canvas look up table index address

DC_CAV_LUT_RDATAL 0x15

Bit(s)	R/W	Default	Description
31~0	R/W	0	CBUS low 32 Bits canvas read back data from LUT.

DC_CAV_LUT_RDATAH 0x16

Bit(s)	R/W	Default	Description
31~0	R/W	0	CBUS high 32 Bits canvas read back data from LUT.

DMC_2ARB_CTRL 0x20

Bit(s)	R/W	Default	Description
31~12	R/W	0	Not used
11:6	R/W	0	the final arbitration. weight for all AXI bus(ports 0~7).
5:0	R/W	0	the final arbitration. weight for all AMbus(ports 8~15).

DMC_REFR_CTRL1 0x23

Bit(s)	R/W	Default	Description
31~16	R/W	0	Not used
15	R/W	0	DMC ddr1 PVT request enable if bit3 enabled.
14	R/W	0	DMC ddr0 PVT request enable if bit3 enabled
13	R/W	0	ddr1 DDR ZQCS comand generation enable.
12	R/W	0	ddr0 DDR ZQCS command generation enable.
11	R/W	0	ddr1 refesh enable while PCTL in config state. 1: enable. 0: disable.
10	R/W	0	ddr0 refesh enable while PCTL in config state. 1: enable. 0: disable.
9:8	R/W	0	Not used
7	R/W	0	dmc to control auto_refresh enable
6:4	R/W	0	refresh number per refresh cycle
3	R/W	0	DMC controlled pvt enable 1 : PVT request generated by DMC tPVTI of refresh period. 0 : PVT generated
2	R/W	0	DMC controlled DDR ZQCS genreation enable. 1 : ZQCS request generated by DMC tZQCI refresh period. 0 : no DMC controlled ZQCS
1	R/W	0	ddr1 auto refresh dmc control select
0	R/W	0	ddr0 auto refresh dmc control select

DMC_REFR_CTRL2 0x24

Bit(s)	R/W	Default	Description
31~24	R/W	0	tZQCI
23~16	R/W	0	tPVTI
15:8	R/W	0	tREFI
7:0	R/W	0	t100ns

DMC_MON_CTRL1 0x25

Bit(s)	R/W	Default	Description
31~16	R/W	0	qos monitor 0 channel select. 16 port selection. 1 bit for one port
15~0	R/W	0	port select for the selected channel.

DMC_MON_CTRL2 0x26

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.
29~21	R/W	0	Not used.
20	R/W	0	qos_mon_trig_sel. 1 = vsync. 0 = timer.
19~4	R/W	0	Not used
3	R/W	0	qos monitor 3 enable.
2	R/W	0	qos monitor 2 enable.
1	R/W	0	qos monitor 1 enable.
0	R/W	0	qos monitor 0 enable.

DMC_MON_CTRL3 0x27

Bit(s)	R/W	Default	Description
31~0	R/W	0	qos_mon_clk_timer. How long to measure the bandwidth.

DMC_MON_CTRL4 0x18

Bit(s)	R/W	Default	Description
31~16	R/W	0	os monitor 1 channel select. 16 port selection. 1 bit for one port.
15~0	R/W	0	port select for the selected channel.

DMC_MON_CTRL5 0x19

Bit(s)	R/W	Default	Description
31~16	R/W	0	os monitor 2 channel select. 16 port selection. 1 bit for one port.
15~0	R/W	0	port select for the selected channel.

DMC_MON_CTRL6 0x1a

Bit(s)	R/W	Default	Description
31~16	R/W	0	os monitor 3 channel select. 16 port selection. 1 bit for one port.
15~0	R/W	0	port select for the selected channel.

DMC_MON_ALL_REQ_CNT 0x28

Bit(s)	R/W	Default	Description
31~0	R	0	at the test period, the whole MMC request high time.

DMC_MON_ALL_GRANT _CNT 0x29

Bit(s)	R/W	Default	Description
31~0	R	0	at the test period, the whole MMC data grant cycles.

DMC_MON_ONE _GRANT _CNT 0x2a

Bit(s)	R/W	Default	Description
31~0	read	0	at the test period, the granted data cycles for the selected port and subIDs.

DMC_MON_SEC_GRANT_CNT 0x2b

Bit(s)	R/W	Default	Description
31~0	read	0	at the test period, the granted data cycles for the selected channel and ports.

DMC_MON_THD_GRANT_CNT **0x2c**

Bit(s)	R/W	Default	Description
31~0	read	0	at the test period, the granted data cycles for the selected channel and ports.

DMC_MON_FOR_GRANT_CNT **0x2d**

Bit(s)	R/W	Default	Description
31~0	read	0	at the test period, the granted data cycles for the selected channel and ports.

DMC_CLKG_CTRL0 **0x30**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29	R/W	0	enable auto clock gating for write rsp generation.
28	R/W	0	enable auto clock gating for read rsp generation.
27	R/W	0	enable auto clock gating for ddr1 read back data buffer.
26	R/W	0	enable auto clock gating for ddr0 read back data buffer.
25	R/W	0	enable auto clock gating for ddr1 command filter.
24	R/W	0	enable auto clock gating for ddr0 command filter.
23	R/W	0	enable auto clock gating for ddr1 write reorder buffer.
22	R/W	0	enable auto clock gating for ddr0 write reorder buffer.
21	R/W	0	enable auto clock gating for ddr1 write data buffer.
20	R/W	0	enable auto clock gating for ddr0 write data buffer.
19	R/W	0	enable auto clock gating for ddr1 read reorder buffer.
18	R/W	0	enable auto clock gating for ddr0 read reorder buffer.
17	R/W	0	enable auto clock gating for read canvas.
16	R/W	0	enable auto clock gating for write canvas.
15	R/W	0	enable auto clock gating for port 15.
14	R/W	0	enable auto clock gating for port 14
13	R/W	0	enable auto clock gating for port 13.
12	R/W	0	enable auto clock gating for port 12.
11	R/W	0	enable auto clock gating for port 11.
10	R/W	0	enable auto clock gating for port 10.
9	R/W	0	enable auto clock gating for port 9.
8	R/W	0	enable auto clock gating for port 8.
7	R/W	0	enable auto clock gating for port 7.
6	R/W	0	enable auto clock gating for port 6.
5	R/W	0	enable auto clock gating for port 5.
4	R/W	0	enable auto clock gating for port 4.
3	R/W	0	enable auto clock gating for port 3.
2	R/W	0	enable auto clock gating for port 2.
1	R/W	0	enable auto clock gating for port 1.
0	R/W	0	enable auto clock gating for port 0.

DMC_CLKG_CTRL1 **0x31**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29	R/W	0	Force to disable the clock of for write rsp generation.
28	R/W	0	Force to disable the clock of read rsp generation.

Bit(s)	R/W	Default	Description
27	R/W	0	Force to disable the clock of ddr1 read back data buffer.
26	R/W	0	Force to disable the clock of ddr0 read back data buffer.
25	R/W	0	Force to disable the clock of ddr1 command filter.
24	R/W	0	Force to disable the clock of ddr0 command filter.
23	R/W	0	Force to disable the clock of ddr1 write reorder buffer.
22	R/W	0	Force to disable the clock of ddr0 write reorder buffer.
21	R/W	0	Force to disable the clock of ddr1 write data buffer.
20	R/W	0	Force to disable the clock of ddr0 write data buffer.
19	R/W	0	Force to disable the clock of ddr1 read reorder buffer.
18	R/W	0	Force to disable the clock of ddr0 read reorder buffer.
17	R/W	0	Force to disable the clock of read canvas.
16	R/W	0	Force to disable the clock of write canvas.
15	R/W	0	Force to disable the clock of port 15.
14	R/W	0	Force to disable the clock of port 14
13	R/W	0	Force to disable the clock of port 13.
12	R/W	0	Force to disable the clock of port 12.
11	R/W	0	Force to disable the clock of port 11.
10	R/W	0	Force to disable the clock of port 10.
9	R/W	0	Force to disable the clock of port 9.
8	R/W	0	Force to disable the clock of port 8.
7	R/W	0	Force to disable the clock of port 7.
6	R/W	0	Force to disable the clock of port 6.
5	R/W	0	Force to disable the clock of port 5.
4	R/W	0	Force to disable the clock of port 4.
3	R/W	0	Force to disable the clock of port 3.
2	R/W	0	Force to disable the clock of port 2.
1	R/W	0	Force to disable the clock of port 1.
0	R/W	0	Force to disable the clock of port 0.

DMC_CHAN_STS **0x32**

Bit(s)	R/W	Default	Description
31~20	R/W	0	Not used.
19	R	0	ddr0 write data buffer idle. 1 : idle 0: busy.
18	R	0	ddr0 write data buffer idle. 1 : idle 0: busy.
17	R	0	ddr1 wbuf idle. 1 : idle 0: busy.
16	R	0	ddr0 wbuf idle. 1 : idle 0: busy.
15~8	R	0	AMBUS ports idle. 1 : idle 0: busy.
7~0	R	0	AXI ports idle. 1 : idle 0: busy.

DMC_N_CLK_CTRL **0x33**

Bit(s)	R/W	Default	Description
31~9	R/W	0	Not used.
6	R/W	0	Manual control for hdcp n_clk. 1: enable clock. 0 : disable clock..
5	R/W	0	Manual control for DEVICE n_clk 1: enable clock. 0 : disable clock.
4	R/W	0	Manual control for ge2d n_clk. 1: enable clock. 0 : disable clock.
3	R/W	0	Manual control for Mali n_clk. 1: enable clock. 0 : disable clock.

Bit(s)	R/W	Default	Description
2	R/W	0	Manual control for VPU n_clk. 1: enable clock. 0 : disable clock.
1	R/W	0	Manual control for DOS n_clk. 1: enable clock. 0 : disable clock.
0	R/W	0	Manual control for CPU n_clk. 1: enable clock. 0 : disable clock.

DMC_CMD_FILTER_CTRL1 0x40

Bit(s)	R/W	Default	Description
31	R/W	0	Not used.
30	R/W	0	1: ddr3 and ddr4 use same filter 0: ddr3 and ddr4 use different filter
29~20	R/W	0x1f	nugt read buf full access waiting limit
19~10	R/W	0x1f	ugt read access waiting limit.
9~0	R/W	0x2f	nugt read access waiting limit

DMC_CMD_FILTER_CTRL2 0x41

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~20	R/W	0x0f	ugt read buf full access waiting limit
19~10	R/W	0x3f	nugt write access pending waiting limit
9~0	R/W	0x7f	ugt write access pending waiting limit.

DMC_CMD_FILTER_CTRL3 0x42

Bit(s)	R/W	Default	Description
31	R/W	0	force wbuf empty.
30~26	R/W	24	wbuf high level number
25~21	R/W	16	wbuf mid level number
20~16	R/W	8	wbuf low level number
15	RW	0	Not used.
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

DMC_CMD_FILTER_CTRL4 0x43

Bit(s)	R/W	Default	Description
31~24	R/W	8	Supper urgent pending limit. Read buffer supper urgent pending timer.
23~16	R/W	30	auto precharge timer when bank is idle.
15	R/W	0	not used.
14~10	R/W	30	tMISS latency. page miss command latency for next same page not hit command.
9~0	R/W	0	rbuf idle timer to let the wbuf output.

DMC_CMD_FILTER_CTRL5 0x44

Bit(s)	R/W	Default	Description
31~24	R/W	0x1f	Once ddr data bus switch to read, the maxmum read command number to give up the bus when there's write request pending for write buffer.
23~16	R/W	0x1f	Once ddr data bus switch to write, the maxmum write command number to give up the bus when there's read request pending too long.
15~8	R/W	0x0f	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	0x0f	Once ddr data bus switch to write, the minimum write command number to transfer back to read stage if there's still pending write.

DMC_CMD_BUFFER_CTRL **0x45**

Bit(s)	R/W	Default	Description
31~26	R/W	32	total write buffer number.
25~20	R/W	32	total read buffer number.
19~10	R/W	0x7f	ugt age waiting limit. over this age limit, this read buffer would turn to super urgent.
9~0	R/W	0x3ff	nugt age waiting limit. over this age limit, this read buffer would turn to super urgent..

DMC_PCTL_LP_CTRL **0x46**

Bit(s)	R/W	Default	Description
31~14	R/W	0	Not used.
13	R/W	0	force disable pctl cmu module clock. 1 : force disable. 0 : not forced.
12	R/W	0	force disable pctl reg module clock. 1 : force disable. 0 : not forced.
11	R/W	0	force disable pctl dcu module clock. 1 : force disable. 0 : not forced.
10	R/W	0	force disable pctl dpu module clock. 1 : force disable. 0 : not forced.
9	R/W	0	force disable pctl sel module clock. 1 : force disable. 0 : not forced.
8	R/W	0	force disable pctl upd module clock. 1 : force disable. 0 : not forced.
7~6	R/W	0	Not used.
5	R/W	0	force enable pctl cmu module clock. 1 : enable. 0 : use auto logic to control if not force to disable.
4	R/W	0	force enable pctl reg module clock. 1 : enable. 0 : use auto logic to control if not force to disable.
3	R/W	0	force enable pctl dcu module clock. 1 : enable. 0 : use auto logic to control if not force to disable.
2	R/W	0	force enable pctl dpu module clock. 1 : enable. 0 : use auto logic to control if not force to disable
1	R/W	0	force enable pctl sel module clock. 1 : enable. 0 : use auto logic to control if not force to disable.
0	R/W	0	force enable pctl upd module clock. 1 : enable. 0 : use auto logic to control if not force to disable.

DMC_RDDBUF_CTRL **0x47**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~5	R/W	8	read data BUFFER empty number restriction with no data output.
4~0	R/W	4	read data BUFFER empty number restriction with data outputing.

DMC_AM0_CHAN_CTRL **0x60**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.

DMC_AM0_QOS_INC **0x62**

Bit(s)	R/W	Default	Description
31~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AM0_QOS_INCBK **0x63**

Bit(s)	R/W	Default	Description
31~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AM0_QOS_DEC **0x64**

Bit(s)	R/W	Default	Description
31~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AM0_QOS_DECBK **0x65**

Bit(s)	R/W	Default	Description
31~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AM0_QOS_DIS 0x66

Bit(s)	R/W	Default	Description
31~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AM0_QOS_DISBK 0x67

Bit(s)	R/W	Default	Description
31~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AM0_QOS_CTRL0 0x68

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AM0_QOS_CTRL1 0x69

Bit(s)	R/W	Default	Description
31~2	R/W	0	Not used.
1	R/W	0	side bank urgent increase enable.
0	R/W	0	side bank urgent decrease enable.

DMC_AM1_CHAN_CTRL 0x6a

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.

DMC_AM1_QOS_INC 0x6c

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AM1_QOS_INCBK 0x6d

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AM1_QOS_DEC 0x6e

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AM1_QOS_DECBK 0x6f

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AM1_QOS_DIS 0x70

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AM1_QOS_DISBK 0x71

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AM1_QOS_CTRL0 0x72

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AM1_QOS_CTRL1 0x73

Bit(s)	R/W	Default	Description
31~2	R/W	0	Not used.
1	R/W	0	side bank urgent increase enable.
0	R/W	0	side bank urgent decrease enable.

DMC_AM2_CHAN_CTRL 0x74

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.

DMC_AM2_QOS_INC 0x76

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AM2_QOS_INCBK **0x77**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AM2_QOS_DEC **0x78**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AM2_QOS_DECBK **0x79**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AM2_QOS_DIS **0x7a**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AM2_QOS_DISBK **0x7b**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AM2_QOS_CTRL0 **0x7c**

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AM2_QOS_CTRL1 **0x7d**

Bit(s)	R/W	Default	Description
31~2	R/W	0	Not used.
1	R/W	0	side bank urgent increase enable.
0	R/W	0	side bank urgent decrease enable.

DMC_AM3_CHAN_CTRL **0x7e**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.

Bit(s)	R/W	Default	Description
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.

DMC_AM3_QOS_INC **0x80**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AM3_QOS_INCBK **0x81**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AM3_QOS_DEC **0x82**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AM3_QOS_DECBK **0x83**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AM3_QOS_DIS **0x84**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AM3_QOS_DISBK **0x85**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AM3_QOS_CTRL0 **0x86**

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AM3_QOS_CTRL1 **0x87**

Bit(s)	R/W	Default	Description
31~2	R/W	0	Not used.
1	R/W	0	side bank urgent increase enable.

Bit(s)	R/W	Default	Description
0	R/W	0	side bank urgent decrease enable.

DMC_AM4_CHAN_CTRL 0x88

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.

DMC_AM4_QOS_INC 0x8a

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AM4_QOS_INCBK 0x8b

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AM4_QOS_DEC 0x8c

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AM4_QOS_DECBK 0x8d

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AM4_QOS_DIS 0x8e

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AM4_QOS_DISBK 0x8f

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AM4_QOS_CTRL0 0x90

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.

Bit(s)	R/W	Default	Description
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AM4_QOS_CTRL1 0x91

Bit(s)	R/W	Default	Description
31~2	R/W	0	Not used.
1	R/W	0	side bank urgent increase enable.
0	R/W	0	side bank urgent decrease enable.

DMC_AM5_CHAN_CTRL 0x92

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.

DMC_AM5_QOS_INC 0x94

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AM5_QOS_INCBK 0x95

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AM5_QOS_DEC 0x96

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AM5_QOS_DECBK 0x97

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AM5_QOS_DIS 0x98

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AM5_QOS_DISBK 0x99

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AM5_QOS_CTRL0 **0x9a**

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AM6_CHAN_CTRL **0x9c**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.

DMC_AM6_QOS_INC **0x9e**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AM6_QOS_INCBK **0x9f**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AM6_QOS_DEC **0xa0**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AM6_QOS_DECBK **0xa1**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AM6_QOS_DIS **0xa2**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AM6_QOS_DISBK **0xa3**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.

Bit(s)	R/W	Default	Description
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AM6_QOS_CTRL0 **0xa4**

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AM7_CHAN_CTRL **0xa6**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.

DMC_AM7_HOLD _CTRL **0xa7**

Bit(s)	R/W	Default	Description
31~24	R/W	0x18	Write hold number. If the outstanding write request meet this number disable the write request.
23~16	R/W	0x10	Write hold release number. Enable the request if outstanding request is lower than this number.
15~8	R/W	0x18	Read hold number. If the outstanding write request meet this number disable the write request.
7~0	R/W	0x10	READ hold release number. Enable the request if outstanding request is lower than this number.

DMC_AM7_QOS_INC **0xa8**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AM7_QOS_INCBK **0xa9**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AM7_QOS_DEC **0xaa**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AM7_QOS_DECBK **0xab**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AM7_QOS_DIS **0xac**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AM7_QOS_DISBK **0xad**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AM7_QOS_CTRL0 **0xae**

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AM7_QOS_CTRL1 **0xaf**

DMC_AXI0 _CHAN_CTRL **0xb0**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
19	R/W	0	AXI0 default urgent bit
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.

DMC_AXI0_QOS_INC **0xb2**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AXI0_QOS_INCBK **0xb3**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AXI0_QOS_DEC **0xb4**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AXI0_QOS_DECBK **0xb5**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AXI0_QOS_DIS **0xb6**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AXI0_QOS_DISBK **0xb7**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AXI0_QOS_CTRL0 **0xb8**

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AXI0_QOS_CTRL1 **0xb9**

Bit(s)	R/W	Default	Description
31~20	R/W	0	Not used.
19~16	R	0	FIQ pin status.
15~12	R	0	IRQ pin status.
11	R/W	1	ARM FIQ controlled super urgent enable.
10	R/W	0	ARM FIQ controlled urgent enable.
9	R/W	0	ARM IRQ controlled super urgent enable.
8	R/W	1	ARM IRQ controlled urgent enable.
7	R/W	1	IRQ/FIQ enable.
6~5	R/W	0	Not used.
4	R/W	1	enable AXI0 auto urgent enable. When there's no other request, treat the AXI0 as super urgent request. other wise, use the bit 3:0 to set the urgent.
3~2	R/W	0	AXI0 urgent level if the VIU read ports are not IDLE.
1~0	R/W	2'b01	AXI0 uregent level if the VIU read ports are IDLE.

DMC_AXI1_CHAN_CTRL **0xba**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
19	R/W	0	AXI0 default urgent bit
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.

Bit(s)	R/W	Default	Description
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.

DMC_AXI1_QOS_INC 0xbc

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AXI1_QOS_INCBK 0xbd

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AXI1_QOS_DEC 0xbe

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AXI1_QOS_DECBK 0xbf

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AXI1_QOS_DIS 0xc0

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AXI1_QOS_DISBK 0xc1

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AXI1_QOS_CTRL0 0xc2

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AXI2_CHAN_CTRL 0xc4

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.

Bit(s)	R/W	Default	Description
19	R/W	0	AXI0 default urgent bit
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.
7~0	R/W	0	force this channel all request to be urgent request.

DMC_AXI2_QOS_INC **0xc6**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AXI2_QOS_INCBK **0xc7**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AXI2_QOS_DEC **0xc8**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AXI2_QOS_DECBK **0xc9**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AXI2_QOS_DIS **0xca**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AXI2_QOS_DISBK **0xcb**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AXI2_QOS_CTRL0 **0xcc**

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AXI3_CHAN_CTRL **0xce**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
19	R/W	0	AXI0 default urgent bit
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.
7~0	R/W	0	force this channel all request to be urgent request.

DMC_AXI3_HOLD _CTRL **0xcf**

Bit(s)	R/W	Default	Description
31~24	R/W	0x18	Write hold number. If the outstanding write request meet this number disable the write request.
23~16	R/W	0x10	Write hold release number. Enable the request if outstanding request is lower than this number.
15~8	R/W	0x18	Read hold number. If the outstanding write request meet this number disable the write request.
7~0	R/W	0x10	READ hold release number. Enable the request if outstanding request is lower than this number.

DMC_AXI3_QOS_INC **0xd0**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AXI3_QOS_INCBK **0xd1**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AXI3_QOS_DEC **0xd2**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AXI3_QOS_DECBK **0xd3**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AXI3_QOS_DIS **0xd4**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AXI3_QOS_DISBK **0xd5**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AXI3_QOS_CTRL0 **0xd6**

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.

Bit(s)	R/W	Default	Description
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AXI4_CHAN_CTRL 0xd8

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
19	R/W	0	AXI0 default urgent bit
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.
7~0	R/W	0	force this channel all request to be urgent request.

DMC_AXI4_QOS_INC 0xda

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AXI4_QOS_INCBK 0xdb

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AXI4_QOS_DEC 0xdc

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AXI4_QOS_DECBK 0xdd

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AXI4_QOS_DIS 0xde

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AXI4_QOS_DISBK 0xdf

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.

Bit(s)	R/W	Default	Description
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AXI4_QOS_CTRL0 **0xe0**

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.
4	R/W	0	qos idle mode enable. enable the idle counter.
3	R/W	0	qos disable enable. enable the feature to disable request.
2	R/W	0	qos decrease urgent enable. enable the decrease urgent function.
1	R/W	0	qos inccrease urgent enable. enabel the increase urgent function.
0	R/W	0	qos enable.

DMC_AXI7_CHAN_CTRL **0xf6**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29:20	R/W	0xff	write request pending cycle number to inc urgent level if not granted.
19	R/W	0	AXI0 default urgent bit
18	R/W	0	force this channel all request to be super urgent request.
17	R/W	0	force this channel all request to be urgent request.
16	R/W	0	force this channel all request to be non urgent request.
13~4	R/W	0x3c	read request pending cycle number to inc urgent level if not granted.
3~0	R/W	0xf	canvas arbiter arbiter weight.
7~0	R/W	0	force this channel all request to be urgent request.

DMC_AXI7_HOLD _CTRL **0xf7**

Bit(s)	R/W	Default	Description
31~24	R/W	0x18	Write hold number. If the outstanding write request meet this number disable the write request.
23~16	R/W	0x10	Write hold release number. Enable the request if outstanding request is lower than this number.
15~8	R/W	0x18	Read hold number. If the outstanding write request meet this number disable the write request.
7~0	R/W	0x10	READ hold release number. Enable the request if outstanding request is lower than this number.

DMC_AXI7_QOS_INC **0xf8**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	if the leacky bucket counter less this number increase the urgent.

DMC_AXI7_QOS_INCBK **0xf9**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	once the increase urgent triggered, after leacky bucket counter is larger than this number, deassert the increase urgent function.

DMC_AXI7_QOS_DEC **0xfa**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	leacky bucket counter is larger this number, decrease the urgent of this request.

DMC_AXI7_QOS_DECBK **0xfb**

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.

Bit(s)	R/W	Default	Description
9~0	R/W	0	if the decrease urgent triggered, after leacky buck counter less this number, deassert the decrease urgent function.

DMC_AXI7_QOS_DIS 0xfc

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	when leacky bucket counter larger this number, disable each request some cycles. the disable cycle number is defined in bit 19:12 of QOS_CTRL0

DMC_AXI7_QOS_DISBK 0xfd

Bit(s)	R/W	Default	Description
31~10	R/W	0	Not used.
9~0	R/W	0	after disable the request, when the leacky bucket counter less than this number, reenable this request..

DMC_AXI7_QOS_CTRL0 0xfe

Bit(s)	R/W	Default	Description
31~20	R/W	0	qos idle number. if the channel no request for that long, set leaky bucket counter to 0.
19~14	R/W	0	qos disable cycles. if the leaky bucket counter larger than QOS_DIS register number, than disable the each request how many cycles.
13~6	R/W	0	leackey bucket counter minus number..
5	R/W	0	qos mode. 1 : use bankdwidth as the QOS control. 0 : Use latency as QOS control.

DMC_AXI7_QOS_CTRL1 0xff

DMC secure register.

Base address 0x0xda838400. Each register takes 4 byte address

DMC_WTCH0_D0 0xa4

DMC_WTCH0_D1 0xa5

DMC_WTCH0_D2 0xa6

DMC_WTCH0_D3 0xa7

WTCH0 will watch upto 128Bits data access {d3, d2,d1,d0}

DMC_WTCH0_RANGE 0xa8

Bit(s)	R/W	Default	Description
31:16	R/W	0	Range end address
15:0	R/W	0	Range start address

DMC_WATCH0_CTRL 0xa9

Bit(s)	R/W	Default	Description
31:16	R/W	0	16Bits write data strb
15:0	R/W	0	16Bits input ports select

DMC_WATCH0_CTRL1 0xaa

Bit(s)	R/W	Default	Description
31:3	R/W	0	Not used
2	R/W	0	watch point 0 enable.
1:0	R/W	0	watch point0 type. 2'b00 : double bytes. only watchpoint data 15:0 and data strb 1:0 is valid. 2'b01: 4 bytes. 2'b10: 8 bytes. 2'b11, all 16bytes.

DMC_WTCH1_D0 **0xab**

DMC_WTCH1_D1 **0xac**

DMC_WTCH1_D2 **0xad**

DMC_WTCH1_D3 **0xae**

WTCH0 will watch upto 128Bits data access {d3, d2,d1,d0}

DMC_WTCH1_RANGE **0xaf**

Bit(s)	R/W	Default	Description
31:16	R/W	0	Range end address
15:0	R/W	0	Range start address

DMC_WATCH1_CTRL **0xb0**

Bit(s)	R/W	Default	Description
31:16	R/W	0	16Bits write data strb
15:0	R/W	0	16Bits input ports select

DMC_WTCH1_CTRL1 **0xb1**

Bit(s)	R/W	Default	Description
31:3	R/W	0	Not used
2	R/W	0	watch point 1 enable.
1:0	R/W	0	watch point1 type. 2'b00 : double bytes. only watchpoint data 15:0 and data strb 1:0 is valid. 2'b01: 4 bytes. 2'b10: 8 bytes. 2'b11, all 16bytes.

DMC_TRAP0 _RANGE **0xb2**

trap function: all read access with predefined PORT or SubIDs must be in the predefine range. Other wire the read access would be blocked. And an error will be generated.

Bit(s)	R/W	Default	Description
31:16	R/W	0	Range end address
15:0	R/W	0	Range start address

DMC_TRAP0 _CTRL **0xb3**

Bit(s)	R/W	Default	Description
31	R/W	0	Not used.
30	R/W	0	Trap0 port ID 2 enable.
29	R/W	0	Trap0 port ID 1 enable.
28	R/W	0	Trap0 port ID 0 enable.
27	R/W	0	Trap0 port ID 2 subID enable.
26	R/W	0	Trap0 port ID 1 subID enable.
25	R/W	0	Trap0 port ID 0 subID enable.
23~20	R/W	0	Trap0 port port ID2 ID number.
19~16	R/W	0	Trap0 port port ID1 ID number.
15~12	R/W	0	Trap0 port port ID0 ID number.
11~8	R/W	0	Trap0 port port ID2 subID number.
7~4	R/W	0	Trap0 port port ID1 subID number.
3~0	R/W	0	Trap0 port port ID0 subID number.

DMC_TRAP1 _RANGE **0xb4**

trap function: all read access with predefined PORT or SubIDs must be in the predefine range. Other wire the read access would be blocked. And an error will be generated.

Bit(s)	R/W	Default	Description
31:16	R/W	0	Range end address

Bit(s)	R/W	Default	Description
15:0	R/W	0	Range start address

DMC_TRAP1 _CTRL 0xb5

Bit(s)	R/W	Default	Description
31	R/W	0	Not used.
30	R/W	0	Trap1 port ID 2 enable.
29	R/W	0	Trap1 port ID 1 enable.
28	R/W	0	Trap1 port ID 0 enable.
27	R/W	0	Trap1 port ID 2 subID enable.
26	R/W	0	Trap1 port ID 1 subID enable.
25	R/W	0	Trap1 port ID 0 subID enable.
23~20	R/W	0	Trap1 port port ID2 ID number.
19~16	R/W	0	Trap1 port port ID1 ID number.
15~12	R/W	0	Trap1 port port ID0 ID number.
11~8	R/W	0	Trap1 port port ID2 subID number.
7~4	R/W	0	Trap1 port port ID1 subID number.
3~0	R/W	0	Trap1 port port ID0 subID number.

DDR0_ADDRMAP_4 0xd4

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~25	R/W	30	ra16 for DDR4 SDRAM
24~20	R/W	0	bg1 for DDR4 SDRAM.
19~15	R/W	29	ba2. or bg0 for DDR4.
14~10	R/W	13	ba1.
9~5	R/W	12	ba0.
4~0	R/W	0	ra15.

DDR0_ADDRMAP_3 0xd3

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~25	R/W	28	Ra 14
24~20	R/W	27	Ra 13.
19~15	R/W	26	Ra 12.
14~10	R/W	25	Ra11.
9~5	R/W	24	Ra10.
4~0	R/W	23	Ra 9.

DDR0_ADDRMAP_2 0xd2

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~25	R/W	22	Row address bit 8.
24~20	R/W	21	Row address bit 7.
19~15	R/W	20	Row address bit 16.
14~10	R/W	19	Row address bit 5.
9~5	R/W	18	Row address bit 4.
4~0	R/W	17	Row address bit 3.

DDR0_ADDRMAP_1 0xd1

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~25	R/W	16	Row address bit 2.
24~20	R/W	15	Row address bit 1.
19~15	R/W	14	Row address bit 0.
14~10	R/W	0	Column address bit 11.

Bit(s)	R/W	Default	Description
9~5	R/W	0	Column address bit 10.
4~0	R/W	11	Column address bit 9.

DDR0_ADDRMAP_0 **0xd0**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~25	R/W	10	Column address bit 8.
24~20	R/W	9	Column address bit 7.
19~15	R/W	8	Column address bit 6.
14~10	R/W	7	Column address bit 5.
9~5	R/W	6	Column address bit 4..
4~0	R/W	5	Column address bit 3..

DDR1_ADDRMAP_4 **0xd9**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~25	R/W	30	rank select..
24~20	R/W	0	Bank address 3 (not used in DDR3/LPDDR3).
19~15	R/W	29	Bank address 2
14~10	R/W	13	Bank address 1
9~5	R/W	12	Bank address 0
4~0	R/W	0	Row address Bit 15.

DDR1_ADDRMAP_3 **0xd8**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~25	R/W	28	Row address bit 14.
24~20	R/W	27	Row address bit 13.
19~15	R/W	26	Row address bit 12.
14~10	R/W	25	Row address bit 11.
9~5	R/W	24	Row address bit 10.
4~0	R/W	23	Row address bit 9..

DDR1_ADDRMAP_2 **0xd7**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~25	R/W	22	Row address bit 8.
24~20	R/W	21	Row address bit 7.
19~15	R/W	20	Row address bit 16.
14~10	R/W	19	Row address bit 5.
9~5	R/W	18	Row address bit 4.
4~0	R/W	17	Row address bit 3.

DDR1_ADDRMAP_1 **0xd6**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~25	R/W	16	Row address bit 2.
24~20	R/W	15	Row address bit 1.
19~15	R/W	14	Row address bit 0.
14~10	R/W	0	Column address bit 11.
9~5	R/W	0	Column address bit 10.
4~0	R/W	11	Column address bit 9.

DDR1_ADDRMAP_0 **0xd5**

Bit(s)	R/W	Default	Description
31~30	R/W	0	Not used.
29~25	R/W	10	Column address bit 8.
24~20	R/W	9	Column address bit 7.
19~15	R/W	8	Column address bit 6.
14~10	R/W	7	Column address bit 5.
9~5	R/W	6	Column address bit 4..
4~0	R/W	4	Column address bit 3..

DMC_DDR_CTRL **0xda**

Bit(s)	R/W	Default	Description
31~23	R/W	0	Not used.
22	R/W	0	DDR4 SDRAM enable. use DDR4 protocal.
21	R/W	0	rank1 enable bit. if 1, rank1 used the address map is as bit 5:3 defined.
20	R/W	0	BG group 1 enable for DDR4 SDRAM.
19	R/W	0	Not used.
18	R/W	0	Always 0
17	R/W	0	Not used
16	R/W	0	1 only use 16bits data in a 32bits phy data interface. 0 : normal data interface.
15~4	R/W	0	Not used.
5~3	R/W	0	DDR channel 1 size. 3'b000: DDR channel 1 : 128Mbyte. 3'b001: DDR channel 1 : 256Mbyte. 3'b010: DDR channel 1 : 512Mbyte. 3'b011: DDR channel 1 : 1GMbyte. 3'b100: DDR channel 1 : 2GMbyte.
2~0	R/W	0	DDR channel 0 size. 3'b000: DDR channel 0 : 128Mbyte. 3'b001: DDR channel 0 : 256Mbyte. 3'b010: DDR channel 0 : 512Mbyte. 3'b011: DDR channel 0 : 1GMbyte. 3'b100: DDR channel 0 : 2GMbyte.

DMC_TEST_STA **0xe0**

Test start address. For non-sha mode, the last 5 bits would be ignored. the test address at 32bytes boundary. For sha mode, address must be in 64 bytes boundary. that mean the last 6 bits must be 0.

DMC_TEST_EDA **0xe1**

Test end address. For non-sha mode, the last 5 bits would be ignored. the test address at 32bytes boundary. For sha mode, address must be in 64 bytes boundary. that mean the last 6bits must be 1.

DMC_TEST_CTRL **0xe2**

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
28	R/W	0	when enable test, enable the sha calculation function must be same as read enable but without write
27	R/W	0	enabe to compare data. when do the read enable to enable the error comparaion. suppose the read data should be same as the data in the write buffer.
26	R/W	0	reserved
25	R/W	0	address generation type. 0: continuous increase the address in the range of test start address and test end address. 1: test module would pick the random address from test start address and test end address.

Bit(s)	R/W	Default	Description
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 : finshed at end address.
23	R/W	0	Reserved
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 adddress 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].
17~16	R/W	0	sha mode for sha function enabled. 00 : not used. 01 : sha1. 2: sha2-256. 3: sha2_224. not used in GXL fixed to be Sha 2.
15~8	R/W	0	write outstanding commands limit.
7~0			read outstanding commands limit.

DMC_TEST_NUM 0xe3

How many test command for the test if the DMC_TEST_CTRL bit 24 is 0.

DMC_TEST_WD0 0xe4

Write data 0 for write command. also for read back data comparision.

DMC_TEST_WD1 0xe5

Write data 1 for write command. also for read back data comparision.

DMC_TEST_WD2 0xe6

Write data 2 for write command. also for read back data comparision.

DMC_TEST_WD3 0xe7

Write data 3 for write command. also for read back data comparision.

DMC_TEST_WD4 0xe8

Write data 4 for write command. also for read back data comparision.

DMC_TEST_WD5 0xe9

Write data 5 for write command. also for read back data comparision.

DMC_TEST_WD6 0xea

Write data 6 for write command. also for read back data comparision.

DMC_TEST_WD7 0xeb

Write data 7 for write command. also for read back data comparision.

DMC_TEST_RD0 0xec

The read back data 0. if error happens, it would capture the first error data.

DMC_TEST_RD1 0xed

The read back data 1. if error happens, it would capture the first error data.

DMC_TEST_RD2 0xee

The read back data 2. if error happens, it would capture the first error data.

DMC_TEST_RD3 **0xef**

The read back data 3. if error happens, it would capture the first error data.

DMC_TEST_RD4 **0xf0**

The read back data 4. if error happens, it would capture the first error data.

DMC_TEST_RD5 **0xf1**

The read back data 5. if error happens, it would capture the first error data.

DMC_TEST_RD6 **0xf2**

The read back data 6. if error happens, it would capture the first error data.

DMC_TEST_RD7 **0xf3**

The read back data 7. if error happens, it would capture the first error data.

DMC_TEST_ERR_ADDR **0xf4**

It capature the first error address.

DMC_TEST_ERR_CNT **0xf5**

How many data error happens in the whole test period.

DMC_TEST_STS **0xf6**

Bit(s)	R/W	Default	Description
31	R	0	test done bit. write 1 to clean.
30	R/W	0	indicate address err
29~5	R/W	0	Not used
4	R/W	0	sha done. 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.

DMC_TEST_SHA_MSG0 **0xf8**

The final sha message byte 3~0.

DMC_TEST_SHA_MSG1 **0xf9**

The final sha message byte 7~4.

DMC_TEST_SHA_MSG2 **0xfa**

The final sha message byte 11~8.

DMC_TEST_SHA_MSG3 **0xfb**

The final sha message byte 15~12.

DMC_TEST_SHA_MSG4 **0xfc**

The final sha message byte 19~16.

DMC_TEST_SHA_MSG5 **0xfd**

The final sha message byte 23~20.

DMC_TEST_SHA_MSG6 **0xfe**

The final sha message byte 27~24.

DMC_TEST_SHA_MSG7 0xff

The final sha message byte 31~28.

DMC_TEST_WRCMD_ADDR 0xdc

The current write cmd address.

DMC_TEST_RDRSP_ADDR 0xdd

The failed read response address(for error data)

DMC_TEST_RDCMD_ADDR 0xde

The current read command address.

DMC_TEST_WDG 0xdf

Bit(s)	R/W	Default	Description
31:16	R	0	write response watch dog.
15:0	R/W	0	read response watch dog.

DMC DDR Channel 1 DRAM register.
Base address 0xc8839000. Each register takes 4 byte address.

DMC_DRAM_SCFG 0x0

Bit(s)	R/W	Default	Description
31:17	R/W	0	Not Used
16:12	R/W	0	additional delay onto the assertion of ac_pdd.
11:8	R/W	0	Reserved
7	R/W	0	Enable assertion of ac_pdd to indicate to the PHY of an opportunty to switch to low power state.
6:1	R/W	0	Reserved
0	R/W	0	to enable the hardware low power interface.(not used).

DMC_DRAM_SCTL 0x1

Bit(s)	R/W	Default	Description
31:3	R/W	0	Not used
2:0	R/W	0	to control the DRAM state. 3'b000: INIT. 3'b001: CFG. 3'b010: Go 3'b011: SLEEP. 3'b100: WAKEUP.

DMC_DRAM_TMRD 0x6

Bit(s)	R/W	Default	Description
31:3	R/W	0	Not used
2:0	R/W	0	tMRD. MRS command clock cycle number in DDR clock.

DMC_DRAM_TRFC 0x7

Bit(s)	R/W	Default	Description
31:10	R/W	0	Not used
9:0	R/W	0	tRFC in DDR clock.

DMC_DRAM_TRP 0x8

Bit(s)	R/W	Default	Description
31:22	R/W	0	Not used

Bit(s)	R/W	Default	Description
21:16	R/W	0	tRP for all bank precharge.
15:6	R/W	0	Not used
5:0	R/W	0	tRP for per bank precharge.

DMC_DRAM_TRTW 0x09

Bit(s)	R/W	Default	Description
31:4	R/W	0	Not used
3:0	R/W	0	tRTW.

DMC_DRAM_TCL 0x0b

Bit(s)	R/W	Default	Description
31:5	R/W	0	Not used
4:0	R/W	0	tCL. read latency.

DMC_DRAM_TCWL 0x0c

Bit(s)	R/W	Default	Description
31:5	R/W	0	Not used
4:0	R/W	0	tCWL. write latency.

DMC_DRAM_TRAS 0x0d

Bit(s)	R/W	Default	Description
31:7	R/W	0	Not used
6:0	R/W	0	tRAS. Minimum Active to Precharge command time.

DMC_DRAM_TRC 0x0e

Bit(s)	R/W	Default	Description
31:7	R/W	0	Not used
6:0	R/W	0	tRC. active to active/Auto refresh command time must = tRAS + tRP.

DMC_DRAM_TRCD 0x0f

Bit(s)	R/W	Default	Description
31:7	R/W	0	Not used
6:0	R/W	0	tRCD active to read/write command time.

DMC_DRAM_TRRD 0x10

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:16	R/W	0	tRRD_l. tRRD between same bank group for DDR4.
15:4	R/W	0	Not used
3:0	R/W	0	tRRD_s. tRRD between different bank group for DDR4. tRRD for other DDR type. Active bank to Active

DMC_DRAM_TFAW 0x11

Bit(s)	R/W	Default	Description
31:8	R/W	0	Not used
7:0	R/W	0	tFAW. Four Bank Activate window.

DMC_DRAM_TRTP 0x12

Bit(s)	R/W	Default	Description
31:4	R/W	0	Not used
3:0	R/W	0	tRTP. read command to precharge command time.

DMC_DRAM_TWR 0x13

Bit(s)	R/W	Default	Description
31:5	R/W	0	Not used
4:0	R/W	0	tWR. write recovery time.

DMC_DRAM_TWTR 0x14

Bit(s)	R/W	Default	Description
31:4	R/W	0	Not used
3:0	R/W	0	tWTR. write command to precharge command time.

DMC_DRAM_TEXSR 0x15

Bit(s)	R/W	Default	Description
31:10	R/W	0	Not used
9:0	R/W	0	tEXSR. Exit selfrefresh to first valid command delay.

DMC_DRAM_TXP 0x16

Bit(s)	R/W	Default	Description
31:4	R/W	0	Not used
3:0	R/W	0	tXP. Exit power down to first valid command delay.

DMC_DRAM_TXPDLL 0x17

Bit(s)	R/W	Default	Description
31:6	R/W	0	Not used
5:0	R/W	0	tXPDLL. exit precharge power down to read or write command.

DMC_DRAM_TZQCS 0x18

Bit(s)	R/W	Default	Description
31:7	R/W	0	Not used
6:0	R/W	0	tZQCS. short ZQ CAL command time

DMC_DRAM_TZQCSI 0x19

tZQinit. the first time DDR3 long ZQ CAL command time.

DMC_DRAM_TCKSRE 0x1a

Bit(s)	R/W	Default	Description
31:5	R/W	0	Not used
4:0	R/W	0	tEXSR. valid clock requirement after self refresh enter.

DMC_DRAM_TCKSRX 0x1b

Bit(s)	R/W	Default	Description
31:5	R/W	0	Not used
4:0	R/W	0	tCKSRX. valid clock requirement before self refresh exit.

DMC_DRAM_TCKE 0x1c

Bit(s)	R/W	Default	Description
31:3	R/W	0	Not used
2:0	R/W	0	tCKE CKE singal minmum low/high width.

DMC_DRAM_TMOD 0x1d

Bit(s)	R/W	Default	Description
31:5	R/W	0	Not used
4:0	R/W	0	tMODE. LOAD MODE command(MRS command) to non-LOAD MODE(not MRS command) cycle time

DMC_DRAM_TDQS 0x1e

Bit(s)	R/W	Default	Description
31:4	R/W	0	Not used
3:0	R/W	0	tDQS. Additional data turnaround time for access different ranks.

DMC_DRAM_TRSTL 0x1f

Bit(s)	R/W	Default	Description
31:7	R/W	0	Not used
6:0	R/W	0	tRSTL. memory reset low time.

DMC_DRAM_TZQCL 0x20

Bit(s)	R/W	Default	Description
31:10	R/W	0	Not used
9:0	R/W	0	LPDDR3/LPDDR2 ZQ CAL (long) period

DMC_DRAM_TMRR 0x21

tMRR. not support for new PCTL.

DMC_DRAM_TCKESR 0x22

Bit(s)	R/W	Default	Description
31:4	R/W	0	Not used
3:0	R/W	0	minmum Self refresh entry to self refresh exit timing.

DMC_DRAM_TREFI 0x24

How many number of REFRESH command for one TREFI period.

DMC_DRAM_TDPD 0x25

Bit(s)	R/W	Default	Description
31:10	R/W	0	Not used
9:0	R/W	0	Minimum Deep power time for LPDDR2/LPDDR3. unit is us.

DMC_DRAM_DFITCTRLDELAY 0x26

Bit(s)	R/W	Default	Description
31:4	R/W	0	Not used
3:0	R/W	0	tctrl_delay for PHY.

DMC_DRAM_DFIODTCFG 0x27

Bit(s)	R/W	Default	Description
31:13	R/W	0	Not used
12	R/W	0	rank1 ODT default. default vulue for ODT[1] pins if theres 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 theres'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 occure in rank0.
7:5	R/W	0	Not used
4	R/W	0	rank0 ODT default. default vulue for ODT[0] pins if theres 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 theres'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 occure in rank1.

DMC_DRAM_DFIODTCFG1 0x28

Bit(s)	R/W	Default	Description
31:28	R/W	0	Not used
27:24	R/W	0	ODT length for BL8 read transfer.
23:20	R/W	0	Not used
19:16	R/W	0	ODT length for BL8 write transfer.
15:13	R/W	0	Not used
12:8	R/W	0	ODT latency for reads. suppose to be 0.
7:5	R/W	0	Not used
4:0	R/W	0	ODT latency for writes. suppose to be 0.

DMC_DRAM_DFITPHYWRDATA 0x2a

Bit(s)	R/W	Default	Description
31:5	R/W	0	Not used
4:0	R/W	0	tphy_wrdata for DFI interface PHY.

DMC_DRAM_DFITPHYWRLAT 0x2b

Bit(s)	R/W	Default	Description
31:5	R/W	0	Not used
4:0	R/W	0	t tphy_wrlat.

DMC_DRAM_DFITRDDATAEN 0x2c

Bit(s)	R/W	Default	Description
31:6	R/W	0	Not used
5:0	R/W	0	tphy_rddataen

DMC_DRAM_DFITPHYRDLAT 0x2d

Bit(s)	R/W	Default	Description
31:6	R/W	0	Not used
5:0	R/W	0	tphy_rdlat

DMC_DRAM_DFITPHYUPDTYPE0 0x2e

Bit(s)	R/W	Default	Description
31:14	R/W	0	Not used
13:0	R/W	0	maximum number of DFI clock cycles that the dfi_phyupd_req signal may remain asserted after the assertion of the dfi_phyupd_ack signals for dfi_phyupd_type = 0x0.

DMC_DRAM_DFITPHYUPDTYPE1 0x2f

Bit(s)	R/W	Default	Description
31:14	R/W	0	Not used
13:0	R/W	0	maximum number of DFI clock cycles that the dfi_phyupd_req signal may remain asserted after the assertion of the dfi_phyupd_ack signals for dfi_phyupd_type = 0x1.

DMC_DRAM_DFITPHYUPDTYPE2 0x30

Bit(s)	R/W	Default	Description
31:14	R/W	0	Not used
13:0	R/W	0	maximum number of DFI clock cycles that the dfi_phyupd_req signal may remain asserted after the assertion of the dfi_phyupd_ack signals for dfi_phyupd_type = 0x2.

DMC_DRAM_DFITPHYUPDTYPE3 0x31

Bit(s)	R/W	Default	Description
31:14	R/W	0	Not used
13:0	R/W	0	maximum number of DFI clock cycles that the dfi_phyupd_req signal may remain asserted after the assertion of the dfi_phyupd_ack signals for dfi_phyupd_type = 0x3.

DMC_DRAM_DFITCTRLUPDMIN 0x32

Bit(s)	R/W	Default	Description
31:16	R/W	0	Not used
15:0	R/W	0	specifies the minimum number of DFI clock cycles that the dfi_ctrlupd_req signal must be asserted.

DMC_DRAM_DFITCTRLUPDMAX 0x33

Bit(s)	R/W	Default	Description
31:16	R/W	0	Not used
15:0	R/W	0	specifies the maximum number of DFI clock cycles that the dfi_ctrlupd_req signal can asserted.

DMC_DRAM_DFITCTRLUPDDLY 0x34

Bit(s)	R/W	Default	Description
31:4	R/W	0	Not used
3:0	R/W	0	Delay in DFI clock cycles between time a DRAM-initiated update could be started and time DRAM-initated update actually starts.

DMC_DRAM_DFIUPDCFG 0x35

Bit(s)	R/W	Default	Description
31:3	R/W	0	Not used
2	R/W	0	enable the generation of DRAM-initiated updates during Wakeup(from low_power to Access state).
1	R/W	0	dfi_phyupd_en. enables the support for acknowledging PHY-initiated updates.
0	R/W	0	dfi_ctlupd_en. enable the generation of DRAM-initiated updates.

DMC_DRAM_DFITREFMSKI 0x36

Bit(s)	R/W	Default	Description
31:13	R/W	0	Not used
12:0	R/W	0	time period of the masked refresh interval.

DMC_DRAM_DFITCTRLUPDI 0x37

DFI DRAM-initiated updates interval, measured in terms of Refresh interval units.

DMC_DRAM_DFITRCFG0 0x38

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:16	R/W	0	dfi_rdlvl_rank_sel Determines the value to drive on the output singal dfi_wrlvl_cs_n.
15:13	R/W	0	Not used
12:4	R/W	0	dfi_rdlvl_edge Determines the value to drive on the output signal dfi_rdlvl_edge.
3:0	R/W	0	dfi_rdlvl_rank_sel Determines the value to drive on the output signal dfi_rdlvl_cs_n.

DMC_DRAM_DFITRWRLVLEN 0x39

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used
8:0	R/W	0	dfi_wrlvl_en. Determines the value to drive on the output signal dfi_wrlvl_en.

DMC_DRAM_DFITRRDLVLEN 0x3a

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used
8:0	R/W	0	dfi_rdlvl_en. Determines the value to drive on the output signal dfi_rdlvl_en.

DMC_DRAM_DFITRRDLVLGATEEN 0x3b

Bit(s)	R/W	Default	Description
31:9	R/W	0	Not used
8:0	R/W	0	dfi_rdlvl_gate_en. Determines the value to drive on the output signal dfi_rdlvl_gate_en.

DMC_DRAM_DFISTCFG0 0x3c

Bit(s)	R/W	Default	Description
31:3	R/W	0	Not used
2	R/W	0	enables the driving of the dfi_data_byte_disable signal. dfi_rdlvl_gate_en.
1	R/W	0	enables the driving of the dfi_freq_ratio signals.
0	R/W	0	dfi_init_start set the value of the dfi_init_start signal.

DMC_DRAM_DFISTCFG1 0x3d

Bit(s)	R/W	Default	Description
31:2	R/W	0	Not used
1	R/W	0	enables support of the dfi_dram-clk_disable signal with Deep Power down for LPDDR2.
0	R/W	0	enables support of the dfi_dram_clk_disable with self refresh.

DMC_DRAM_DFITDRAMCLKEN 0x3e

Bit(s)	R/W	Default	Description
31:4	R/W	0	Not used
3:0	R/W	0	t_dram_clk_enable. number of DFI clock cycles from the de-assertion of the dfi_dram_clk_disable signal on the DFI until the first valid rising edge of the clock to the dram memory device. dfi_rdlvl_gate_en.

DMC_DRAM_DFITDRAMCLKDIS 0x3f

Bit(s)	R/W	Default	Description
31:4	R/W	0	Not used
3:0	R/W	0	t_dram_clk_disable. number of DFI clock cycles from the assertionof the dfi_dram_clk_disable signal on the DFI until the clock to the DRAM memory device, maintains a low vale.

DMC_DRAM_DFILPCFG0 0x40

Bit(s)	R/W	Default	Description
31:28	R/W	0	dfi_lp_wakeup_dpd. value to drive on dfi_lp_wakeup signal when Deep power down mode is entered. 4'b0000 -16 cycels. 4'b0001 -32 cycels. 4'b0010 -64 cycels. 4'b0011 -128 cycels. 4'b0100 -256 cycels. 4'b0101 -512 cycels. 4'b0110 -1024 cycels. 4'b0111 -2048 cycels.
27:25	R/W	0	Not used
24	R/W	0	dfi_lp_en_dpd. Enables DFI low power interface handshading during Deep power down mode.
23:20	R/W	0	Not used
19:16	R/W	0	setting for tlp_resp time.
15:12	R/W	0	dfi_lp_wakeup_dr. value to drive on dfi_lp_wakeup signals when self refresh mode is entered.
11:9	R/W	0	Not used
8	R/W	0	dfi_lp_en_sr. enables DFI low power interface handshading during self refresh.
7:4	R/W	0	dfi_lp_wakeup_pd value to drive on dfi_lp_wakeup signal when Power Down mode is entered.
3:1	R/W	0	Not used
0	R/W	0	dfi_lp_en_pd. enable DFI low power interface handshading during power down entry/exit.

DMC_DRAM_MCFG 0x41

Bit(s)	R/W	Default	Description
31:24	R/W	0	clock stop idle period in n_clk cycles. for LPDDR2 and LPDDR3 . 0 to disalbe.
23:22	R/W	0	Not used
21:20	R/W	0	LPDDR2/LPDDR3 burst lenght. must ot be 2'b10. only support BL8.
17	R/W	0	PD_exit mode. 0 slow exit. 1: fast exit.
16	R/W	0	pd_type. 0 precharge power down. 1 active power down.
15:8	R/W	0	pd_idle. power down idle perild in n_clk cycles. memory puts to power down mode if the PCTL is ile for pd_idle n_clk cycles.
7	R/W	0	Always 0.

Bit(s)	R/W	Default	Description
6:4	R/W	0	DDR type. //3'b000: DDR3 mode. //3'b001: DDR4 mode. //3'b010: LPDDR3 mode. //3'b011: LPDDR2 mode.
3	R/W	0	2T mode. for DDR3 and DDR4 we can sett to 1T or 2T. for LPDDR3, we must set it to 2T for DMC generating command.
2	R/W	0	Always 0.
1	R/W	0	Not used
0	R/W	0	burst length. always 1 for burst 8. (S905X only suport burst 8.)

DMC_DRAM_MCFG1 0x42

Bit(s)	R/W	Default	Description
31:1	R/W	0	Not used
0	R/W	0	disable DMC traffic to PCTL and DFI interface.

DMC_DRAM_PPCFG 0x43

Bit(s)	R/W	Default	Description
31:1	R/W	0	Not used
0	R/W	0	ppmem enablel. 0 : ddr sdram is 32 data bits mode. 1: ddr sdram is 16 data bits mode.

DMC_DRAM_ZQCFG 0x44

Bit(s)	R/W	Default	Description
31:24	R/W	0	for LPDDR2/LPDDR3 mode. value to drive on memory address bit [19:12] for an automatic hardware generated ZQCL commands.
23:16	R/W	0	zqcl_ma. for LPDDR2/LPDDR3 mode. value to drive on memory address bits [[11:4] for an automatic hardware generated ZQCL command.
15:8	R/W	0	zqcs_op. for LPDDr2/LPDDR3 mode. value to drive on memory address bit [19:12] for automatic hardware generated ZQCS command.
16	R/W	0	pd_type. 0 precharge power down. 1 active power down.
15:8	R/W	0	pd_idle. power down idle perild in n_clk cycles. memory puts to power down mode if the PCTL is ile for pd_idle n_clk cycles.
7:0	R/W	0	zqcs_ma. for LPDDR2/LPDDR3 mode. value to drive on memory address bits [[11:4] for an automatic hardware generated ZQCS command.

DMC_DRAM_DFISTSTAT0 0x46

Bit(s)	R/W	Default	Description
31:25	R/W	0	Not used
24:16	R/W	0	DFI data byte disable. reports the value of the output signal dfi_data_byte_disable.
15:6	R/W	0	Not used
5:4	R/W	0	report the value of the output signal of dfi_freq_ratio.
3:2	R/W	0	Not used
1	R/W	0	dfi_init_start. reports the value of the value of the dfi_init_start output.
0	R/W	0	dfi_init_complete. init finish.

DMC_DRAM_STAT 0x48

Bit(s)	R/W	Default	Description
31:15	R/W	0	Not used
14	R/W	0	dmc to pctl inteface is idle. 1 = idle. 0: not idle.
13	R/W	0	pctl dpu is idle. 1 : idle. 0 : not idle.
12	R/W	0	pctl dcu is idle. 1 : idle. 0 : not idle.
11:9	R/W	0	Not used

Bit(s)	R/W	Default	Description
8	R/W	0	auto power down stats. 1 : SDRAM in power down state 0: not.
7	R/W	0	Not used.
6:4	R/W	0	Lower power state trigger state. bit 6: software drive. bit 5: hardware driven due to hardware low power interface. bit 4: hardware driven due to auto self refresh(MCFG1.sr_idle >> 0).
3	R/W	0	Not used
2:0	R/W	0	current operation state of DRAM. 3'b000 = init_mem. 3'b001 = config. 3'b010 = config_req. 3'b011 = access. 3'b100 = access_req. 3'b101 = Low_power. 3'b110 = low_power _entry_req. 3'b111 = low power exit _req.

DMC_DRAM_TCCD 0x52

Bit(s)	R/W	Default	Description
31:20	R/W	0	Not used
19:16	R/W	0	tCCD_l
15:4	R/W	0	Not used
3:0	R/W	0	tCCD_s.

MMC DDR PHY control register.

Base address 0xc8837000.

AM_DDR_PLL_CNTL0 0x0

Bit(s)	R/W	Default	Description
31	R/W	0	DDR DPLL_EN
30	R/W	0	DDR_DPLL_RESET
29:28	R/W	0	DDR_DPLL_CLK_EN
27:21	R/W	0	Not used.
20:16	R/W	0	DDR_DPLL_N
15:13	R/W	0	Not used
12:4	R/W	0	DDR_DPLL_M
3:2	R/W	0	DDR_DPLL_OD
1:0	R/W	0	DDR_DPLL_OD1

AM_DDR_PLL_CNTL1 0x1

Bit(s)	R/W	Default	Description
31~22	R/W	0	Not used
21	R/W	0	DDR_DPLL_DPFD_LMODE
20:19	R/W	0	DDR_DPLL_DCVC_IN
18	R/W	0	DDR_DPLL_DIV_MODE
17	R/W	0	DDR_DPLL_DCO_SDM_EN
16:15	R/W	0	DDR_DPLL_DCO_SDMCK_SEL
14	R/W	0	DDR_DPLL_DCO_M_EN
13	R/W	0	DDR_DPLL_DCO_BAND_OPT
12:10	R/W	0	DDR_DPLL_DATA_SEL
9:8	R/W	0	DDR_DPLL_AFC_NT
7:6	R/W	0	DDR_DPLL_AFC_HOLD_T

Bit(s)	R/W	Default	Description
5:4	R/W	0	DDR_DPLL_AFC_DSEL_IN
3	R/W	0	DDR_DPLL_AFC_DSEL_BYPASS
2	R/W	0	DDR_DPLL_AFC_CLK_SEL
1:0	R/W	0	DDR_DPLL_ACQ_R_CTR

AM_DDR_PLL_CNTL2 0x2

Bit(s)	R/W	Default	Description
31:28	R/W	0	DDR_DPLL_FILTER_PVT2
27:24	R/W	0	DDR_DPLL_FILTER_PVT1
23:22	R/W	0	Not used
21:11	R/W	0	DDR_DPLL_FILTER_ACQ2
10:0	R/W	0	DDR_DPLL_FILTER_ACQ1

AM_DDR_PLL_CNTL3 0x3

Bit(s)	R/W	Default	Description
31	R/W	0	DDR_DPLL_SDMNC_EN
30	R/W	0	DDR_DPLL_SDMNC_MODE
29	R/W	0	DDR_DPLL_SDMNC_RANGE
28:27	R/W	0	Not used
26:20	R/W	0	DDR_DPLL_SDMNC_POWER
19:17	R/W	0	DDR_DPLL_SDMNC_ULMS
16	R/W	0	DDR_DPLL_LOCK_BYPASSN
15:12	R/W	0	DDR_DPLL_LM_W
11:6	R/W	0	DDR_DPLL_LM_S
5	R/W	0	DDR_DPLL_IIR_BYPASS_N
4	R/W	0	DDR_DPLL_FREQ_SHIFT_EN
3:2	R/W	0	DDR_DPLL_FREQ_SHIFT_V
1:0	R/W	0	DDR_DPLL_FREQ_SEL

AM_DDR_PLL_CNTL4 0x4

Bit(s)	R/W	Default	Description
31	R/W	0	DDR_DPLL_SSEN
30:28	R/W	0	DDR_DPLL_SS_AMP
27	R/W	0	DDR_DPLL_SS_CLK_SEL
26	R/W	0	not used.
25:21	R/W	0	DDR_DPLL_SS_CLK
20	R/W	0	DDR_DPLL_SSC_EN
19:16	R/W	0	DDR_DPLL_SSC_DEP_SEL
15:14	R/W	0	DDR_DPLL_SSC_MODE
13:12	R/W	0	DDR_DPLL_SSC_OFFSET
11:10	R/W	0	DDR_DPLL_SSC_STR_M
9	R/W	0	not used.
8	R/W	0	DDR_DPLL_TDC_EN
7	R/W	0	DDR_DPLL_TDC_CAL_EN
6	R/W	0	DDR_DPLL_CODE_NEW
5:4	R/W	0	DDR_DPLL_TDC_DELAY_C
3:2	R/W	0	DDR_DPLL_TDC_OFF_C
1:0	R/W	0	DDR_DPLL_VBG_CT_VC

AM_DDR_PLL_CNTL5 0x5

Bit(s)	R/W	Default	Description
31:29	R/W	0	DDR_DPLL_VBG_PTAT_VC
28:27	R/W	0	Not used
26	R/W	0	DR_DPLL_PVT_FIX_EN
25:24	R/W	0	DDR_DPLL_FB_OD
23:12	R/W	0	DDR_DPLL_DIV_FRAC
11:0	R/W	0	DDR_DPLL_REVE

AM_DDR_PLL_STS 0x6

Bit(s)	R/W	Default	Description
31	R/W	0	DDR_PLL_LOCK
30:19	R/W	0	Not used.
18	R/W	0	DDR_AFC_DONE
17	R/W	0	DDR_PLL_LOCK
16:7	R/W	0	DDR_DPLL_OUT_RSV
6:0	R/W	0	DDR_SDMNC_MONITOR

DDR_CLK_CNTL 0x7

Bit(s)	R/W	Default	Description
31	R/W	0	ddr_pll_clk enable. enable the clock from DDR_PLL to clock generation. whenever change the DDR_PLL frequency, disable the clock, after the DDR_PLL locked, then enable it again..
30	R/W	0	ddr_pll_prod_test_en. enable the clock to clock/32 which to clock frequency measurement and
29	R/W	0	ddr_phy_ctl_clk enable
28	R/W	0	clock generation logic soft reset. 0 = reset.
27	R/W	0	phy_4xclk phase inverter..
26	R/W	0	pll_freq divide/2. 1: use pll div/2 clock as the n_clk. 0: use pll clock as n_clk.
25:4	R/W	0	Not used.
3	R/W	0	force to disable PUB PCLK
2	R/W	0	PUB auto ctrl n_clk clock gating enable. when the DFI_LP_REQ and DFI_LP_ACK detected , auto gated PUB n_clk.
1	R/W	0	force to disable PUB PCLK.
0	R/W	0	PUB pclk auto clock gating enable. when the IP detected PCTL enter power down mode, use this bit to

DDR0_SOFT_RESET 0x8

Bit(s)	R/W	Default	Description
31:24	R/W	0	Not used
3	R/W	0	PUB n_CLK domain soft reset. 1 : reset. 0 normal.
2	R/W	0	PUB p_CLk domain soft reset. 1: reset. 0 normal.
1	R/W	0	Not used.
0	R/W	0	Not used.

DDR0_APD_CTRL 0x9

Bit(s)	R/W	Default	Description
31:24	R/W	0	Not used
23:16	R/W	0	power down enter latency. when IP checked the dfi_lp_req && dfi_lp_ack, give PCTL and pub additional latency let them settle down, then gating the clock.
15	R/W	0	Not used
14	R/W	0	AC ctl_clk auto clock gating enable.
13	R/W	0	AC ddr_clk auto clock gating enable.
12	R/W	0	AC rdclk auto clock gating enable.
11:8	R/W	0	DX ctl_clk auto clock gating enable. 1 = enable. 0 = disable.

Bit(s)	R/W	Default	Description
7:4	R/W	0	DX ddr_clk auto clock gating enable. 1 = enable. 0 = disable.
3:0	R/W	0	DX rd_clk auto clock gating enable. 1 = enable. 0 = disable.

DDR0_PHY_CLK_CNTL0 0x0a

This final pin result is 1, means enable this clock. the final pin result is 0, means disable this clock.

If use auto, means the hardware will disable this clock if there's no traffic in DFI and PHY is in LOW power mode. PUB_CGCR regsiter is used to control if use this pins. so please check PUB data book for CGCR register define.

Bit(s)	R/W	Default	Description
31	R/W	0	Not used
30	R/W	0	PHY_TOP AC ctl_clk clock gating auto generate enable 1 = auto 0 : 0.
29	R/W	0	PHY_TOP AC ddr_clk clock gating auto generate enable 1 = auto 0 : 0.
28	R/W	0	PHY_TOP AC rdclk clock gating auto generate enable 1 = auto 0 : 0.
27:24	R/W	0	PHY_TOP DX ctl_clk clock gating auto generate enable 1 = auto 0 : 0.
23:20	R/W	0	PHY_TOP DX ddr_clk clock gating auto generate enable 1 = auto 0 : 0.
19:16	R/W	0	PHY_TOP DX rd_clk clock gating auto generate enable 1 = auto 0 : 0.
15	R/W	0	Not used
14	R/W	0	PHY_TOP AC ctl_clk clock enable pin high 1 : pin = 1. 0 : use auto ctrol
13	R/W	0	PHY_TOP AC ddr_clk clock enable pin high 1 : pin = 1. 0 : use auto ctrol
12	R/W	0	PHY_TOP AC rdclk clock enable pin high 1 : pin = 1. 0 : use auto ctrol
11:8	R/W	0	PHY_TOP DX ctl_clk clock enable pin high 1 : pin = 1. 0 : use auto ctrol
7:4	R/W	0	PHY_TOP DX ddr_clk clock enable pin high 1 : pin = 1. 0 : use auto ctrol
3:0	R/W	0	PHY_TOP DX rd_clk clock enable pin high 1 : pin = 1. 0 : use auto ctrol

DDR0_PHY_CLK_CNTL1 0x0b

This final pin result is 1, means enable this clock. the final pin result is 0. means disable this clock. If use auto, means the hardware will disable this clock if there's no traffic in DFI and PHY is in LOW power mode.PUB_CGCR regsiter is used to control if use this pins.

Bit(s)	R/W	Default	Description
31:11	R/W	0	Not used
10	R/W	0	PUB global logic auto clock gating enable. 1 = auto. 0 : pin = 0.
9	R/W	0	PUB DFI auto clock gating enable. 1 = auto. 0 : pin = 0.
8	R/W	0	PUB SCH auto clock gating enable. 1 = auto. 0 : pin = 0.
7	R/W	0	PUB global logic clock gating enable. 1 = pin = 1. 0 : use auto ctrl.
6	R/W	0	PUB schedule clock gating enable 1 = pin = 1. 0 : use auto ctrl.
5	R/W	0	PUB DFI clock gating enable 1 = pin = 1. 0 : use auto ctrl.
4	R/W	0	PUB config logic gating clock enable. 1 = pin = 1. 0 : pin = 0.
3	R/W	0	PUB initialization gating clock enable. 1 = pin = 1. 0 : pin = 0.
2	R/W	0	PUB trainning clock gating enable. 1 = pin = 1. 0 : pin = 0.
1	R/W	0	PUB bist clock gating enable. 1 = pin = 1. 0 : pin = 0.
0	R/W	0	PUB dcu clock gating enable. 1 = pin = 1. 0 : pin = 0.

DDR0_FRQ_CHG 0x0c

Bit(s)	R/W	Default	Description
31	R/W	0	enable PLL fast frequncy change. write 1 to start. after this bit cleaned, it finished.
30	R/W	0	tinit_start watch dog timeout error status. write 1 to clean. after dfi_init_start high, there's no dfi_init_complete response from PHY.
29	R/W	0	tinit_complete watch dog timeout error status. write 1 to clean. after dfi_init_start low. there's no dfi_init_complete response from PHY.
28:23	R/W	0	Not used
22	R/W	0	disable PUB n_clk when hardare change AMPLL OD. 1 : disable. 0 not.
21	R/W	0	disable PHY n_clk when hardare change AMPLL OD. 1 : dsiable. 0 not.
20	R/W	0	disable DMC(SYSTEM) n_clk when hardare change AMPLL OD. 1 : dsiable. 0 not.

Bit(s)	R/W	Default	Description
19:4	R/W	0	wait cycles for the PLL stable after change PLL OD OD1 value. suppose should be several cycles.
3:2	R/W	0	new value of the PLL OD pin . After FFC, this value copied to AM_DDR_PLL_CNTL0 3:2 .
1:0	R/W	0	new value of the PLL OD1 pin. After FFC, this value copied to AM_DDR_PLL_CNTL0 1:0.

DDR0_FRQ_PTR0 0x0d

Bit(s)	R/W	Default	Description
31:16	R/W	0	Not used
15:0	R/W	0	tinit_start(tDFI_INIT_START) for frequency change.

DDR0_FRQ_PTR1 0x0e

tinit_complete(tDFI_INIT_COMPLETE) for frequency change.

DDR0_PHY_IO_CTRL0 0x10

Bit(s)	R/W	Default	Description
31	R/W	0	dj for cs_n[0] pin
30	R/W	0	et for cs_n[0] pin
29	R/W	0	oj for cs_n[0] pin
28	R/W	0	sj for cs_n[0] pin
27	R/W	0	dj for odt[1] pin
26	R/W	0	et for odt[1] pin
25	R/W	0	oj for odt[1] pin
24	R/W	0	sj for odt[1] pin
23	R/W	0	dj for odt[0] pin
22	R/W	0	et for odt[0] pin
21	R/W	0	oj for odt[0] pin
20	R/W	0	sj for odt[0] pin
19	R/W	0	dj for cke[1] pin
18	R/W	0	et for cke[1] pin
17	R/W	0	oj for cke[1] pin
16	R/W	0	sj for cke[1] pin
15	R/W	0	dj for cke[0] pin
14	R/W	0	et for cke[0] pin
13	R/W	0	oj for cke[0] pin
12	R/W	0	sj for cke[0] pin
11	R/W	0	dj for ck_n pin
10	R/W	0	et for ck_n pin
9	R/W	0	oj for ck_n pin
8	R/W	0	sj for ck_n pin
7	R/W	0	dj for ck pin
6	R/W	0	et for ck pin
5	R/W	0	oj for ck pin
4	R/W	0	sj for ck pin
3	R/W	0	dj for mem_rst_n pin
2	R/W	0	et for mem_rst_n pin
1	R/W	0	oj for mem_rst_n pin
0	R/W	0	sj for mem_rst_n pin

DDR0_PHY_IO_CTRL1 0x11

Bit(s)	R/W	Default	Description
31:8	R/W	0	Not used
7	R/W	0	dj for act_n pin
6	R/W	0	et for act_n pin
5	R/W	0	oj for act_n pin
4	R/W	0	sj for act_n pin
3	R/W	0	dj for cs_n[1] pin
2	R/W	0	et for cs_n[1] pin
1	R/W	0	oj for cs_n[1] pin

Bit(s)	R/W	Default	Description
0	R/W	0	sj for cs_n[1] pin

DDR0_PHY_IO_CTRL2 0x12

DDR0_PHY_IO_CTRL3 0x13

DDR0_PHY_IO_CTRL4 0x14

DDR0_PHY_IO_CTRL5 0x15

DDR0_PHY_IO_CTRL6 0x16

DDR0_PHY_IO_CTRL7 0x17

DDR0_PHY_IO_CTRL8 0x18

DDR0_PHY_IO_CTRL9 0x19

DDR0_PHY_IO_ST0 0x1a

DDR0_PHY_IO_ST1 0x1b

DDR0_PHY_IO_ST2 0x1c

DDR PHY PUB register.

Base address 0xc8836000. Please refer dwc_ddr_multiphy_g2_pubm2_databook.pdf.

31. NAND

31.1 Overview

S905X supports SLC/MLC/TLC NAND Flash with 60-bit ECC.

31.2 Register Definitions

The base address of NAND registers is 0xd0074000, and the final address of each register is listed below:

Table VI.31.1 NAND Register List

Register Name	Description	Address	R/W
P_NAND_CMD	Write Command and Read Status	Base + 0x00	R/W
P_NAND_CFG	Configuration	Base + 0x04	R/W
P_NAND_DADR	Data Address	Base + 0x08	R/W
P_NAND_IADR	Information Address	Base + 0x0c	R/W
P_NAND_BUF	Read Data Buffer	Base + 0x10	R
P_NAND_INFO	Information	Base + 0x14	R
P_NAND_DC	DDR interface	Base + 0x18	R
P_NAND_ADR	DDR Address	Base + 0x1c	R
P_NAND_DL	DDR Low 32 Bits Data	Base + 0x20	R/W
P_NAND_DH	DDR High 32 Bits Data	Base + 0x24	R/W
P_NAND_CADR	Command Queus Address	Base + 0x28	R/W
P_NAND_SADR	Status Address	Base + 0x2c	R/W
P_NAND_PINS	CS2: SDRAM/NAND pin sharing	Base + 0x30	R/W
P_NAND_VER	Version number	Base + 0x38	R

P_NAND_CMD

Write : Send NAND command to controller, the command format is specified in previous section.

Bit(s)	Name	Description
21:0	Cmd	NAND command sent to NAND queue buffer
30	Cmd_go	When 1, and NAND bus is in waiting Rb mode, due to time out or longer than expected Rb waiting, the command queue will move on by disable RB waiting in current command.
31	Cmd_reset	When 1 the NAND command queue buffer is reset to zero.

Read : Read NAND controller status

Bit(s)	Name	Description
19:0	Cmd_curr	NAND command current on NAND bus, still going, not finished.
24:20	Cmd_cnt	Number of NAND commands still in NAND command queue buffer, the buffer size is 32.
25	Timer out	When 1 wait Rb command timed out.
26	Rb0	Current Rb0 status, 1: ready, 0: busy.
27	Rb1	Current Rb1 status, 1: ready, 0: busy.
28	Rb2	Current Rb2 status, 1: ready, 0: busy.
29	Rb3	Current Rb3 status, 1: ready, 0: busy.
30	Mem_rdy	When 1, DDR interface is idle and ready to accept memory movement request.

31	Ecc_rdy	When 1, ECC BCH encoder/decoder is idle and read to accept encode or decode activity.

P_NAND_CFG

Bit(s)	Name	Description
4:0	Bus_cyc	The number of system clock cycles in one NAND cycle – 1, for example, if the bus_cyc is 3, then the NAND cycle is 4 system clock cycles, the minimum setting is 3, the maximum setting is 31. program this register according NAND timing mode.
9:5	Bus_tim	The timing to lock the NAND data when read NAND data or status, please refer to "Timing Calculator" for details.
11:10	Sync	00: Async mode 01: Micron Sync mode 10: Toshiba-Samsung toggle mode
12	Cmd_start	When set to "1", if the NAND controller internal 32 command buffer has less than 16 commands, the command DMA starts reading commands from DDR and saves them to internal buffer, the DMA keeps watching the internal buffer, reads whenever there are less than 16 commands left, if an all "zero" command is met, This bit is cleared and current command DMA is done.
13	Cmd_auto	When set to '1', the command DMA will check the previous command queue end location, whether it is changed from all "zero" back to valid command, the auto check period is 1 ms.
14	Apb_mode	Special NAND mode for ROM boot or debug, when 1, DDR interface is redirected to APB register, all the read/write activities are through APB registers. When used in ROM boot and DDR is not ready.
15	Spare_only	When 1, the NAND controller read NAND with/without ECC, but only save the information bytes into DDR memory, the main data is discarded, designed for software to survey the NAND flash spare bytes and prepare NAND programming.
16	Sync_adj	Used to adjust data timing in sync or toggle mode, 0: default timing, 1: delay 1 system clock cycle.
20	Sts_irq_en	Enable STS IRQ.
21	Cmd_irq_en	Enable RB pin or RB IO IRQ.
26	Oob_on	Set to 1 oob_mode 16/0, Set to 0 no oob bytes.
27	Oob_mode	New in M8 v2, Set to 1 enable new oob mode. First page 16 bytes, all other pages 0 byte.
28	Dc_ugt	Set NAND controller DDR interface to Urgent mode.
29	Nand_wpn	When 1, the NAND wpn pin is set to low, the NAND is in write protection mode, default to 0, the NAND is not protected.
30	Core_power	When 1, internal NAND controller core clock gating is override to always on. The clock gating is disabled.
31	Top_power	When 1, internal NAND top clock gating is override to always on, the clock gating for top is disabled.

P_NAND_DADR

Set DDR data address by registers, the address is 32 Bits, since the DDR data address can also be set by NAND commands, when both happens at the same time, register setting is ignored, the NAND command setting takes effect. Software should avoid conflict address setting.

P_NAND_IADR

Set DDR information (spare bytes) address by registers, the address is 32 Bits, since the DDR information address can also be set by NAND commands, when both happens at the same time, register setting is ignored, the NAND command setting takes effect. Software should avoid conflict address setting.

P_NAND_BUF

When read NAND status, features or data, the results are buffer in this register, the register is 32 Bits, it can only hold 4 bytes, if the host not doesn't read out the results, it will be over written by the following "read".

P_NAND_INFO

One 32 Bits information per each 512 bytes in ECC mode.

Bit(s)	Name	Description
7:0	Info 0	Information (spare) byte 0, errors already corrected by BCH.
15:8	Info 1	Information (spare) byte 1, errors already corrected by BCH
21:16	Pages	Count down page number in current DMA read, starts from the total page size, count down to 1.
28:24	Errcnt	Number of errors corrected by BCH in current page, 0 means no error in current page, 0x1f means this page is uncorrectable.
29	Unc	When 1, this page is uncorrectable by BCH, this page is bad.
30	Ecc	When 1, current NAND read is with ECC on.
31	Done	When 1, the information content and data read from NAND are valid, otherwise the "read" is not done.

P_NAND_DC

Used for apb_mode, internal NAND controller still uses DDR interface, only the DDR request and DDR grant are redirected from DDR to apb registers, this enables the host to read NAND without DDR, for ROM boot and debug.

Bit(s)	Name	Description
7:0	Dc_wr_dm	DDR write data mask, Each Bit masks one byte.
8	Dc_wr	When 1, write data from NAND to DDR. When 0, read data from DDR to NAND.
9	Dc_lbrst	When 1, the 64 Bits data is the last in current DDR burst, used with Dc_req.
10	Dc_ugt	When 1, current DDR request of read/write is urgent, in NAND controller, the Dc_ugt is set to 0, none-urgent.
11	Dc_req	When 1, the NAND controller send request to DDR to read or write data, in case of apb_mode, when dc_req is "1", the host is responsible for send to or receive from NAND controller. Note: this is the only signal the host needs to check when in apb_mode.

P_NAND_ADR

32 Bits DDR address when NAND controller read or write to DDR memory, any address space within the DDR memory installed in the system is valid.

P_NAND_DL

The DDR interface uses 64 Bits width bus, this register is for low 32 Bits, [31:0].

P_NAND_DH

The DDR interface uses 64 Bits width bus, this register is for high 32 Bits, [63:32].
Read and write to this register will generate "grant" and "dc_wr_avail" or "dc_rd_avail".

Always read or write low 32 Bits first, then read or write to high 32 Bits and NAND controller hardware will generate "grant" and "dc_wr_avail" or "dc_rd_avail", combined with the data, the DDR address advances to next address.

NAND controller programs NAND flash in apb_mode:

- Check P_NAND_DC till "dc_req" is high.

- Write low 32 Bits to P_NAND_DL.
- Write high 32 Bits to P_NAND_DH.
- Go back to beginning till all the data is written.

NAND controller reads data from NAND flash in apb_mode:

- Check P_NAND_DC till "dc_req" is high.
- Read low 32 Bits from P_NAND_DL.
- Read high 32 Bits from P_NAND_DH.
- Go back to beginning till all the data is read.

Since the DDR interface in NAND controller process the data in group of 16 double words (64 Bits), the software can only check "dc_req" at the beginning of each 16 double words.

P_NAND_CADR

Set command queue memory address, 32 Bits, any memory location.
This address can only be programmed by APB bus.

P_NAND_SADR

Set status memory location, 32 Bits, any memory location.
This address can also be programmed through command queue.

P_NAND_PINS

Bit(s)	Name	Description
13:0	Pins_len	When pins are acquired by NAND, it will use Pins_len number of NAND bus cycles before releasing pins. Defult is 8.
27:14	Pins_off	When pins are released by NAND, it will wait Pins_off number of NAND bus cycles before sending next request. Default is 2.
31	Not shared	When 1 the pins is not shared, default is 0, pins are shared.

P_NAND_VER

Start from M8, NAND controller hardware version ID, 16 Bits year in hexadecimal, 8 Bits month in hexadecimal, maximum ECC in decimal.

Project	Version ID	Description
M6, M6TV older	0	N/A
M6TVlite	0x2012081e	Designed on Aug 2012, ECC 30
M8	0x2012083c	Designed on Aug 2012, ECC 60

32. eMMC/SD/SDIO

32.1 Overview

S905X has the following features of eMMC/SD/SDIO

- Supports SDSC/SDHC/SDXC card and SDIO specification version 2.x/3.x/4.x DS/HS modes up to UHS-I SDR104
- Supports eMMC and MMC card specification version 5.0 up to HS400 with data content DES crypto
- 1 bit, 4 Bits, 8 Bits date 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/SDIO CONTROLLERs to connect varied SD/MMC Card, SDIO device (e.g. Wi-Fi module), or eMMC protocol compatible memory with high throughput.

32.2 Pin Description

Table VI.32.1 Pin Description of eMMC/SD/SDIO Module

Name	Type	Description	Speed (MHz)
CLK	Output	SD eMMC clock, 0~200MHz	200
DS	eMMC optional SD Card or SDIO not used Input (used as device IRQ in SDIO) R_{DS} pull-down used in HS400 mode.	Data Strobe, HS400 mode only	200
DAT[7:0]	SD Card/SDIO 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 R_{OD} is connected when in open-drain mode.	Command Response	100
Rst_n	eMMC required	Hardware reset	Low
IRQ	SDIO required SD Card or eMMC not used.	Device interrupt can be replaced by DAT[1]	Low

32.3 eMMC/SD Mode

eMMC Mode:

Table VI.32.2. eMMC Mode

Mode Name	Data Rate	IO Voltage	Bus Width	Frequency	Max Data Transfer
Legacy MMC card	Single	3/1.8/1.2V	1, 4, 8	0-26MHz	26MB/s
High Speed SDR	Single	3/1.8/1.2V	1,4, 8	0-52MHz	52MB/s
High Speed DDR	Dual	3/1.8/1.2V	4, 8	0-52MHz	104MB/s
HS200	Single	1.8/1.2V	4, 8	0-200MHz	200MB/s
HS400 (highest)	Dual	1.8/1.2V	8	0-200MHz	400MB/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 and 1.2V
- Tuning concept for Read Operations

The HS400 mode has the following features

- DDR Data sampling method
- CLK frequency up to 200MHz, Data rate is up to 400MB/s
- Only 8-bit bus width supported
- Signaling levels of 1.8V and 1.2V
- Support up to 5 Drive Strengths
- Data strobe signal is toggled only for Data out and CRC response

SD Mode:

TableVI.32.3. 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

32.4 Clock

In S905X, clock source has the options to select from the table below.

Table VI.32.4. Clock Sources

Clock source	Name	Frequency MHz	Note
0	Cts_oscin_clk	24	Default
1	Fclk_div2	1000	
2	Fclk_div3	667	
3	Fclk_div5	400	
4	Fclk_div7	286	
5	Mp2_clk_out	Programmable	
6	Mp3_clk_out	Programmable	
7	Gp0_pll_clk		

32.5 Descriptor

Descriptor Structure

The descriptor has a size of 4x32 Bits.

TableVI.32.5 Descriptor Structure

<table>
<tr><th>byte</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>0</td><td colspan="8">length[7:0]</td></tr>
<tr><td>1</td><td colspan="4">Timeout 4 Bits</td><td>End of chain</td><td>R1b</td><td>block mode</td><td>length[8]</td></tr>
<tr><td>2</td><td>data num</td><td>resp num</td><td>resp 128</td><td>resp nocrc</td><td>Data wr</td><td>Data io</td><td>No cmd</td><td>No resp</td></tr>
<tr><td>3</td><td>owner</td><td>error</td><td colspan="6">cmd index 6 Bits</td></tr>
<tr><td>4</td><td colspan="8" rowspan="4">cmd argument 32 Bits</td></tr>
<tr><td>5</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td></tr>
<tr><td>8</td><td colspan="8">data address 32 Bits or data 0-4 bytes</td></tr>
</table>

9	[1]Big Endian, [0]SRAM
10	
11	
12	response address 32 Bits or response irq en [0]SRAM
13	
14	
15	

Descriptor Definition

TableVI.32.5 Descriptor 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]	$2^{timeout}$ 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 timeout setting is in APB3 register.
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, ths 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.

32.6 Timing Specification

FigureVI.32.1 MMC/SD/SDIO Timing Diagram

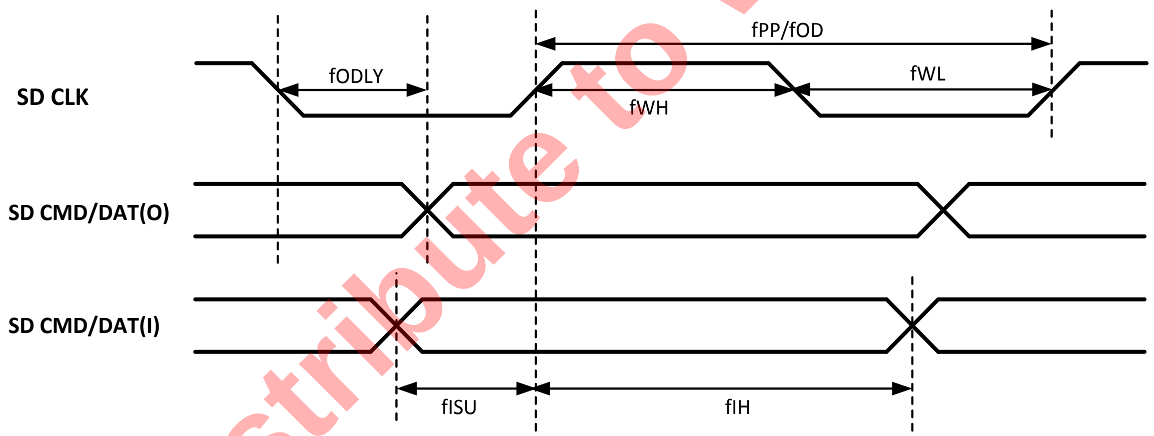


Table VI.32.6 SDHC Timing Specification

Symbol	Description	Min.	Max.	Unit	Notes
fPP	Clock Frequency Low Speed	0	400	KHz	
	Clock Frequency Full/High Speed	0	25/80	MHz	
	Clock Frequency UHS-I Speed	0	100	MHz	
fOD	Clock Frequency Identification Mode	100	400	KHz	
tWL	Clock Low Time	5		ns	
tWH	Clock High Time	5		ns	
tODLY	Output Delay	-2.5	2.5	ns	
tISU	Input Setup Time	4		ns	
tIH	Input Hold Time	4		ns	

Table VI.32.7 SDIO Timing Specification

Symbol	Description	Min.	Max.	Unit	Notes
fPP	Clock Frequency Low Speed	0	400	KHz	
	Clock Frequency Full/High Speed	0	25/50	MHz	

fOD	Clock Frequency Identification Mode	100	400	KHz	
tWL	Clock Low Time	8		ns	
tWH	Clock High Time	8		ns	
tODLY	Output Delay	-3.5	3.5	ns	
tISU	Input Setup Time	6		ns	
tIH	Input Hold Time	6		ns	

32.7 Register Definitions

Each register final address = module base address+ address * 4

Where module address addresses are 0xd0070000 for port A (Wifi/SDIO) and portC(eMMC); 0xd0072000 for port B (SD card), 0xd0074000 for port C (eMMC).

SD_EMMC_CLOCK 0x0

Bit(s)	R/W	Default	Description
31:26	R	0	unused
25	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
24	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
23:20	R/W	0	Cfg_rx_delay: RX clock delay line 0: no delay, n: delay n*200ps Maximum delay 3ns.
19:16	R/W	0	Cfg_tx_delay: TX clock delay line 0: no delay, n: delay n*200ps Maximum delay 3ns.
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.

SD_EMMC_DELAY 0x4

Bit(s)	R/W	Default	Description
31:28	R/W	0	Dly[7]: Data 7 delay line
27:24	R/W	0	Dly[6]: Data 6 delay line
23:20	R/W	0	Dly[5]: Data 5 delay line
19:16	R/W	0	Dly[4]: Data 4 delay line
15:12	R/W	0	Dly[3]: Data 3 delay line
11:8	R/W	0	Dly[2]: Data 2 delay line
7:4	R/W	0	Dly[1]: Data 1 delay line

Bit(s)	R/W	Default	Description
3:0	R/W	0	Dly[0]: Data 0 delay line Total delay = 200ps * Dly When Dly == 0, no delay. When Dly ==15, 3ns delay. NOTE: the 200ps is typical delay, actually delay may vary from chip to chip, from different temperature.

SD_EMMC_ADJUST 0x8

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
21:16	R/W	0	Adj_delay: Resample the input signals when clock index==adj_delay
15	R/W	0	DS_enable 1: Sampling the DAT based on DS in HS400 mode, simulation shows it can tracking 2.5ns range with 8000ppm. 0: disable
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_enable: 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 == 8, test CMD line.
7:4	R/W	0	Dly[9]: DS delay line
3:0	R/W	0	Dly[8]: Command delay line

SD_EMMC_CALOUT 0x10

Bit(s)	R/W	Default	Description
31:16			Unused
15:8	R		Cali_setup: Calibration reading The event happens at this index, The index starts from rising edge of core clock from 0, 1, 2, …
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: Copied from BASE+0x8 [15:8] include cali_sel, cali_enable, adj_enable, cali_rise.

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

SD_EMMC_CFG 0x44

Bit(s)	R/W	Default	Description
31:28	R/W	0	Cfg_ip_txd_adj: Data 1 interrupt,

Bit(s)	R/W	Default	Description
			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.
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	Cfg_chk_ds: Check data strobe in HS400
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, $2^{cfg\_rc\_cc}$ core clock cycles.
11:8	R/W	0	Cfg_resp_timeout: Wait response till $2c^{fg\_resp\_timeout}$ core clock cycles. Maximum 32768 core cycles.
7:4	R/W	0	Cfg_bl_len: Block length $2^{cfg\_bl\_len}$, 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)

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.

SD_EMMC_IRQ_EN 0x4c

Bit(s)	R/W	Default	Description
31:17			Unused
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.

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

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.

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.

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

Descriptor_REG4 0x60

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CMD_RSP1: Response bit 63:32

Descriptor_REG5 0x64

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CMD_RSP2: Response bit 95:64

Descriptor_REG6 0x68

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CMD_RSP3: Response bit 127:96

Descriptor_REG7 0x6c

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

Current_Next_Descriptor_REG0 0x70

Bit(s)	R/W	Default	Description
31:0	R		SD_EMMC_CURR_CFG: Current descriptor under execution.

Current_Next_Descriptor_REG1 0x74

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

Current_Next_Descriptor_REG2 0x78

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

Current_Next_Descriptor_REG3 0x7c

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

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.

Current_Next_Descriptor_REG5 0x84

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

Current_Next_Descriptor_REG6 0x88

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

Current_Next_Descriptor_REG7 0x8c

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

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.

SD_EMMC_TXD 0x94

Bit(s)	R/W	Default	Description
31:25			Unused

Bit(s)	R/W	Default	Description
24:16	R		Txd_blk: Txd BUS block counter
15			unused
14:0	R		Txd_cnt: Txd BUS cycle counter

33. SERIAL PERIPHERAL INTERFACE COMMUNICATION CONTROLLER

33.1 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.

33.2 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

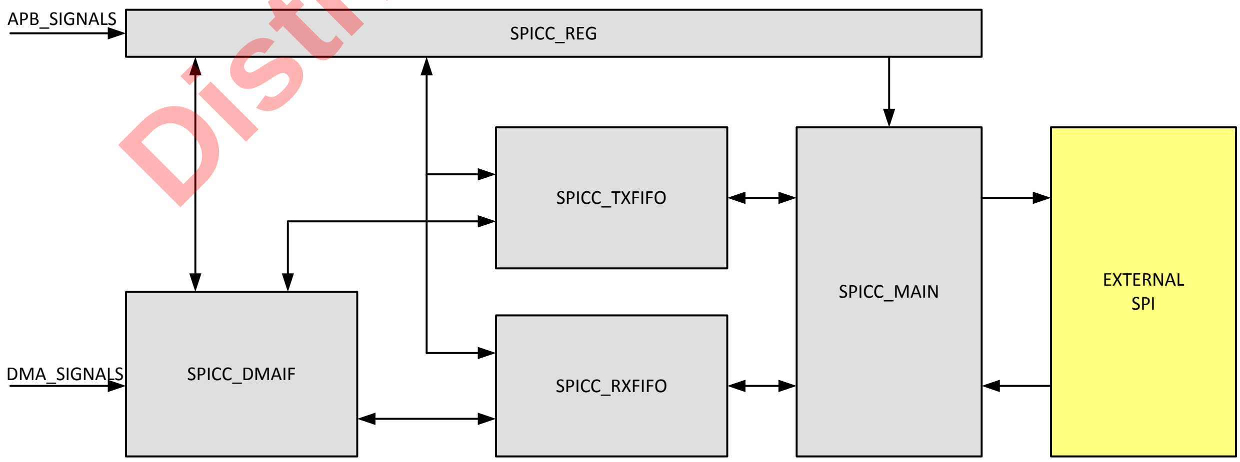
33.3 Functional Description

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.

There are 5 sub-modules in spi communication 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.

Fig VI.33.1 SPICC Block Diagram



- spicc_reg is driven by host cpu, and spicc_reg is responsible for configuring other modules.
- spicc_dmaif is responsible for dealing with DMA operatoins.
- spicc_txfifo contains a transmission FIFO.
- spicc_rxfifo contains a receiving FIFO.
- spicc_main is responsible for main control of basic spi operation.

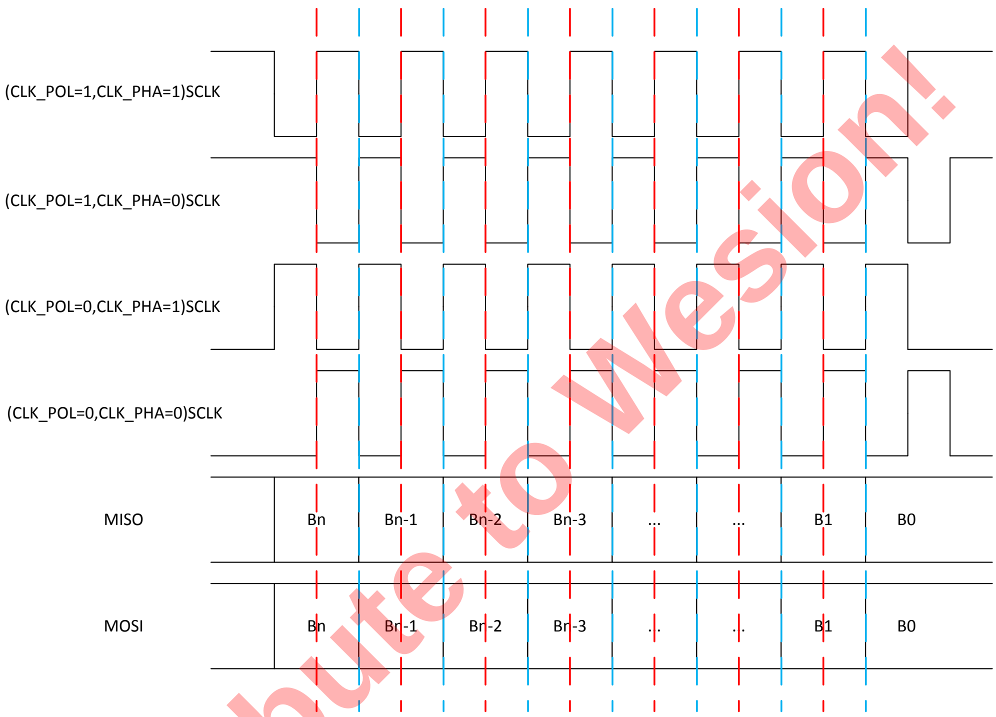
Here are SPI External Signals:

Table VI.33.1 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.

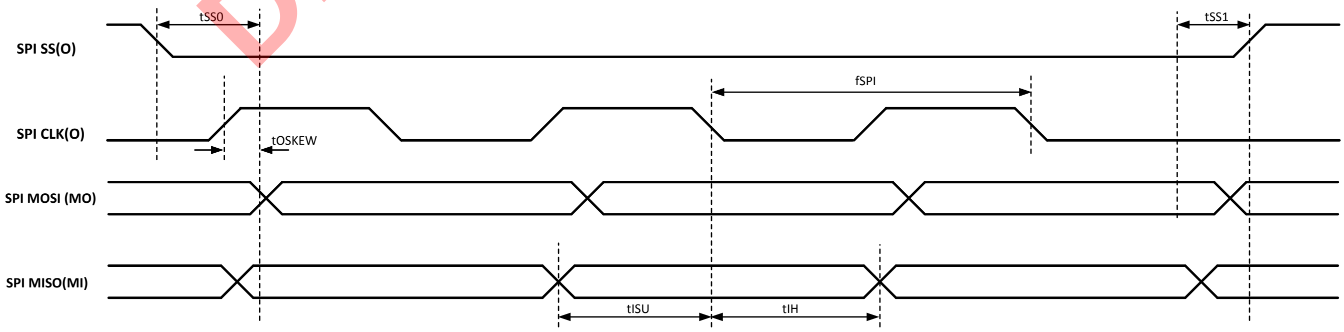
And here is SPI Generic Timing:

Fig VI.33.2 SPI Generic Timing



33.4 Timing Specification

Fig VI.33.3. SPICC Timing Diagram



Master Mode, CPOL=0, CPHA=1

Table VI.33.2. SPICC Master Timing Specification

Symbol	Description	Min.	Max.	Unit	Notes
fSPI	Clock Frequency	0	30	MHz	
tOSKEW	Output SKEW	-3.5	3.5	ns	
tISU	Input Setup Time	8		ns	
tIH	Input Hold Time	8		ns	
tSS0	SS active time before transition	8		ns	
tSS1	SS inactive time after transition	8		ns	

Slave Mode, CPOL=0, CPHA=1

Table VI.33.3. SPICC Slave Timing Specification

Symbol	Description	Min.	Max.	Unit	Notes
fSPI	Clock Frequency	0	30	MHz	
tOSKEW	Output SKEW	-3.5	3.5	ns	
tISU	Input Setup Time	8		ns	
tIH	Input Hold Time	8		ns	
tSS0	SS active time before transition	5		ns	
tSS1	SS inactive time after transition	5		ns	

33.5 Register Description

RXDATA 0xc1108d80

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

Note1: when PIO mode, programmer can get data from this register.

TXDATA 0xc1108d84

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

Note1: when PIO mode, programmer need send data to this register.

CONREG 0xc1108d88

Bit(s)	R/W	Default	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 failing 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)

Bit(s)	R/W	Default	Description
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 ==0, ss will output 0 between each spi transition. And if ssctl ==1, ss will output 1.
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.

INTREG 0xc1108d8c

Bit(s)	R/W	Default	Description
31:8			Reserved
7	RW	0	[1]tcen(transfer completed interrrupt enable)
6	RW	0	[1]roen(rxfifo overflow interrrupt enable)
5	RW	0	[1]rfen(rxfifo full interrrupt enable)
4	RW	0	[1]rhen(rxfifo half full interrrupt enable)
3	RW	0	[1]rren(rxfifo ready interrrupt enable)
2	RW	0	[1]tfen(txfifo full interrrupt enable)
1	RW	0	[1]then(txfifo half full interrrupt enable)
0	RW	0	[1]teen(txfifo empty interrrupt enable)

Note1: Interrupt Status presents in STATREG.

DMAREG 0xc1108d90

Bit(s)	R/W	Default	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<=thres, will request read)
0	RW	0	[1]DMA Enable

STATREG 0xc1108d94

Bit(s)	R/W	Default	Description
31:8			Reserved
7	RW	0	[1]tc(transfer completed, w1c, see Note1)
6	R	0	[1]ro(rxfifo overflow)
5	R	0	[1]rf(rxfifo full)
4	R	0	[1]rh(rxfifo half full)
3	R	0	[1]rr(rxfifo ready)
2	R	0	[1]tf(txfifo full)

Note1: tc is the status bit which indicates a burst transfer is completed. And a burst transfer should be started by writing xch 1'b1. This bit supports w1c(Write 1 clear).

PERIODREG 0xc1108d98

Bit(s)	R/W	Default	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.

TESTREG 0xc1108d9c

Bit(s)	R/W	Default	Description
31:23	RW	0	Reserved
Read Only (Need pay attention)			
22:21	R	0	[2]fiforst(fifo soft reset)
20:15	R	0x15	[6]dlyctl(delay control)
14	R	0	[1]swap(data swap for reading rxfifo)
13	R	0	[1]lbc(loop back control)
Write Only (Need pay attention)			
23:22	W	0	[2]fiforst(fifo soft reset)
21:16	W	0x15	[6]dlyctl(delay control)
15	W	0	[1]swap(data swap for reading rxfifo)
14	W	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.

DRADDR 0xc1108da0

Bit(s)	R/W	Default	Description
31:0	RW	0	Read Address of DMA

DWADDR 0xc1108da4

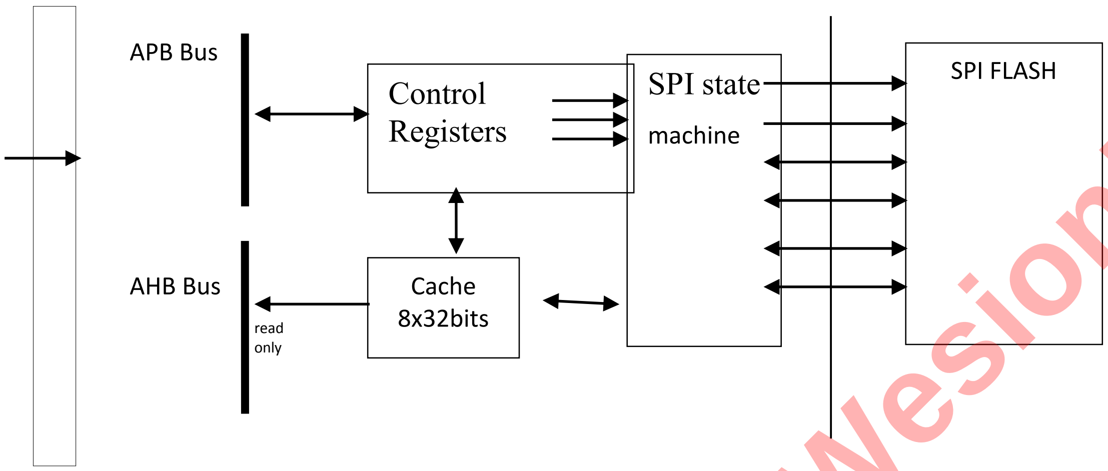
Bit(s)	R/W	Default	Description
31:0	RW	0	Write Address of DMA

34. SERIAL PERIPHERAL INTERFACE FLASH CONTROLLER

34.1 Overview

SPI Flash Controller is designed for connecting varied SPI Flash memory. Below shows the data flow of SPIFC.

Fig VI.34.1 SPIFC Data Flow Diagram

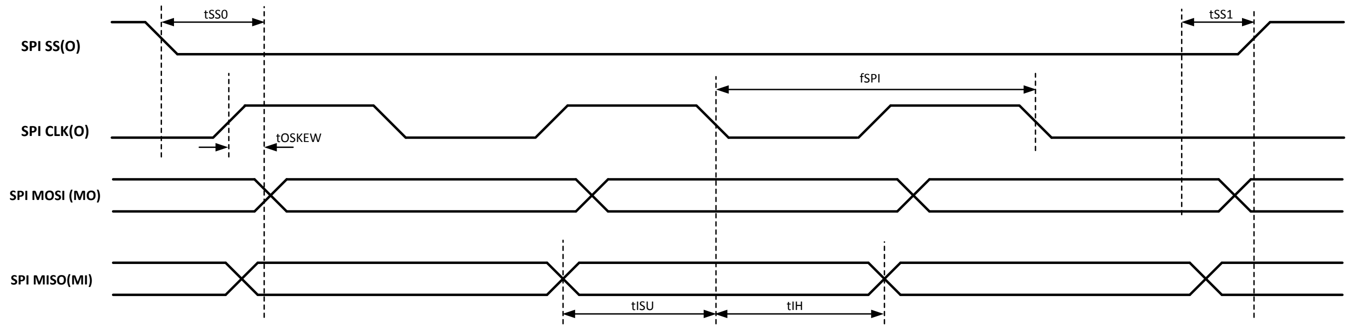


34.2 Features

- 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

34.3 Timing Specification

Fig VI.34.1 SPIFC Timing Diagram



Master Mode, CPOL=0, CPHA=1

Table VI.34.1 SPIFC Master Timing Specification

Symbol	Description	Min.	Max.	Unit	Notes
fSPI	Clock Frequency	0	20	MHz	
tWL	Clock Low Time	20		ns	
tWH	Clock High Time	20		ns	
tOSKEW	Output SKEW	-5	5	ns	
tISU	Input Setup Time	10		ns	
tIH	Input Hold Time	10		ns	
tSS0	SS active time before transition	0		ns	
tSS1	SS inactive time after transition	0		ns	
tWP0	WP active time before transition	Software Controlled		ns	
tWP1	WP inactive time after transition	Software Controlled		ns	

34.4 Register Description

SPIFC FLASH Command register 0xc1108c80

Bit(s)	R/W	Default	Description
31	R/W	0	READ command. 1 = read. When it becames 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 continous 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.

SPIFC address register 0xc1108c84

Bit(s)	R/W	Default	Description
31:0	R/W	0	The address[31:0] of the user command

SPIFC control register 0xc1108c88

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.
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.

SPIFC control register 0xc1108c8c

Bit(s)	R/W	Default	Description
31:28	R/W	5	SPI Clock cycles for SPI flash timing requirement tCSH.
27:16	R/W	0xfff	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.

SPIFC status register 0xc1108c90

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.

Bit(s)	R/W	Default	Description
31:0	R/W	0	In SPI Slave mode, the read status of the user command

SPIFC control register 2 0xc1108c904

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.

Bit(s)	R/W	Default	Description
			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 sycles 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.
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

Bit(s)	R/W	Default	Description
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 sycles 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.

SPIFC Clock register 0xc1108c98

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 frequcncy = 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 freqency 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.
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 frequcncy = 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 freqency 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.

SPIFC User register 0xc1108c9c

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

Bit(s)	R/W	Default	Description
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
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(s)-length,…
2	R/W	1	Backward Compatible: 1 = compatible to Apollo SPI This bit affect the three registers: "SPI Flash Commmand 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., bit 24 is not controlling the start address. The data output can be specified by bit 25

SPIFC User register 1 0xc1108ca0

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, …
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, …

SPIFC User register 2 0xc1108ca4

Bit(s)	R/W	Default	Description
31:28	R/W	0	USER command bit number for command state 0 = 1 bit, 1= 2 Bits, …

Bit(s)	R/W	Default	Description
27:16	R/W	0	Reserved
15:0	R/W	0	The command content of the user command

SPIFC User register 3 0xc1108ca8

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

SPIFC PIN register 0xc1108cac

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, repectively 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

SPIFC Slave register 0xc1108cb0

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

Bit(s)	R/W	Default	Description
			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 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 bit 4:0 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

SPIFC Slave register 1 0xc1108cb4

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

SPIFC Slave register 2 0xc1108cb8

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, …

SPIFC Slave register 3 0xc1108cbC

Bit(s)	R/W	Default	Description
31:24	R/W	0	In SPI slave mode, Command value for writing status, when bit 27 “SPI slave command define enable” in SPI Slave register is 1

Bit(s)	R/W	Default	Description
23:16	R/W	0	In SPI slave mode, Command value for reading status, when bit 27 “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 bit 27 “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 bit 27 “SPI slave command define enable” in SPI Slave register is 1

SPIFC controller cache 0~7 0xc1108cc0~0xc1108cdc

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.

SPIFC controller buffer 8~15 0xc1108ce0~0xc1108cfc

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.

Section VII I/O Interface

This part describes S905X's I/O interfaces from the following aspects:

- USB
- Ethernet
- SDIO
- Transport Interface and Transport Stream Demux
- IR Remote
- SAR ADC
- I2C
- UART
- PWM

35. UNIVERSAL SERIAL BUS

35.1 Overview

The chips integrates in one USB OTG (On-the-GO) controller and one USB Host controller.

The USB OTG controller is a Dual-Role-Device (DRD) controller that supports both device and host functions and complies fully with the On-The-Go Supplement to the USB 2.0 Specification, Revision 1.3a and Revision 2.0. It can also be configured as a host-only or device-only controller, fully compliant with the USB 2.0 Specification. The USB host controller supports host functions and is fully compliant with USB2.0 specification,

35.2 Features

The 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 up to 16 host channels.

The 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
- Supports up to 16 host channels.

36. ETHERNET MAC

36.1 Overview

The Ethernet MAC controller provides a complete Ethernet interface from the chip to a Reduced Gigabit Media Independent Interface (RGMII) or Reduced Media Independent Interface (RMII) compliant Ethernet PHY.

36.2 Features

Ethernet MAC has the following features:

- 10/100 MAC
- RGMII/RMII
- AHB 32 Bits internal bus
- RX FIFO 4KB, TX FIFO 2KB
- 2 MAC addresses
- Power Management

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
- 100Base-FX 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)
- On-board diagnostics
- WOL detection
- Flexible configurations for LED status indicators, supports 3 color LED’s
- Supporting military temperature range -40C to 125C
- Perfect mix of analog and digital lends itself to robustness, portability, and performance
- Multiple input clock options
- Stand-alone core

36.3 Register Description

PRG_ETHERNET_ADDR0 0xc8834108

Bit(s)	R/W	Default	Description
31	R/W	0	Set AHB to DDR interface as urgent.
30	R/W	0	RGMII mode Use RX_CLK as TX_CLK.
29-27	R/W	0	RMII & RGMII mode Select one signal from {RXDV, RXD[3:0]} to calibrate.
26	R/W	0	RMII & RGMII mode 0: test falling edge 1: test rising edge
25	R/W	0	RMII & RGMII mode Start calibration logic
24-20	R/W	0	RMII & RGMII mode 5 Bits correspondent to {RXDV, RXD[3:0]}, set to 1 will delay the data capture by 1 cycle.
19-15	R/W	0	Set bit14 to 0. RMII & RGMII mode Capture input data at clock index equal to adj_delay.
14	R/W	0	Set RXDV and RXD setup time, data is aligned with index 0. When set to 1, auto delay and skew
13	R/W	0	RMII & RGMII mode Enable data delay adjustment and calibration logic.
12	R/W	0	RMII & RGMII mode Enable TX_CLK and PHY_REF_CLK generator.
11	R/W	0	RMII mode Use inverted internal clk_rmii_i to generate 25/2.5 tx_rx_clk.
10	R/W	0	Generate 25MHz clock for PHY
9-7	R/W	0	RMII & RGMII mode, 000: invalid value. 001: mp2_clk_out is 250MHz. 010: mp2_clk_out is 500MHz. ... Mp2_clk_out is "ratio" *250MHz.
6-5	R/W	0	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	0	Unused
3	R/W	0	RMII mode CLK_RMII RGMII mode RX_CLK Use inverted signal when set to 1.
2	R/W	0	Sideband Descriptor Endianness Control Function: When set high, this signal configures the DMA to transfer descriptors in reverse endianness of the data format. When low (by default), the descriptors are transferred in the same endian format as the data. This signal is sampled during active reset (including soft-reset) only and ignored after reset is de-asserted.
1	R/W	0	Sideband Data Endianness Control Function: When set high, this signal configures the DMA to transfer data in big-endian format. When low (by default), the data is transferred in little-endian format. This signal is sampled during active reset (including soft-reset) only and ignored after reset is de-asserted.
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. 1: internal value 001: RGMII 0: internal value 100: RMII

PRG_ETHERNET_ADDR1 0xc883410c

Bit(s)	R/W	Default	Description
31-16	R	0	Unused
15	R/W	0	The result is valid
14	R/W	0	The results is rising edge test or falling edge test.
13-11	R/W	0	The signal under test.
10	R/W	0	The Calibration logic is waiting for event.
9-5	R/W	0	The RX_CLK length in 1ns.

Bit(s)	R/W	Default	Description
4-0	R/W	0	Signal switch position in 1ns.

37. SDIO

Please refer to eMMC/SD/SDIO part for SDIO information.

38. INTER-INTEGRATED CIRCUIT (I2C)

38.1 Overview

Inter-Integrated Circuit (IIC or I2C) is a multi-slave serial communication bus between ICs. S905X integrates the I2C interface and signals allowing communications with other I2C peripheral devices.

38.2 Features

The I^2C Master Module has the following features:

- Support for 7-bit and 10-bit addressable devices
- Programmable bus speed including standard speed (100kBits/s) and fast speed (400kBits/sec)
- Error transfer detection
- “Transfer complete” indication by polling or interrupt (Interrupts handled by the ISA module. See the ISA module for details).
- Internal buffer holding up to 8 bytes for transfer (in either direction)
- Flexible architecture allowing the software to dictate the format of the I^2C bit streams
- Manual setting of the I^2C bus to accommodate a software only mode

38.3 Timing Specification

There are two modes to the I2C master interface: Standard (100khz) and fast (400khz).

Fig VII.38.1 I2C Interface Timing Diagram

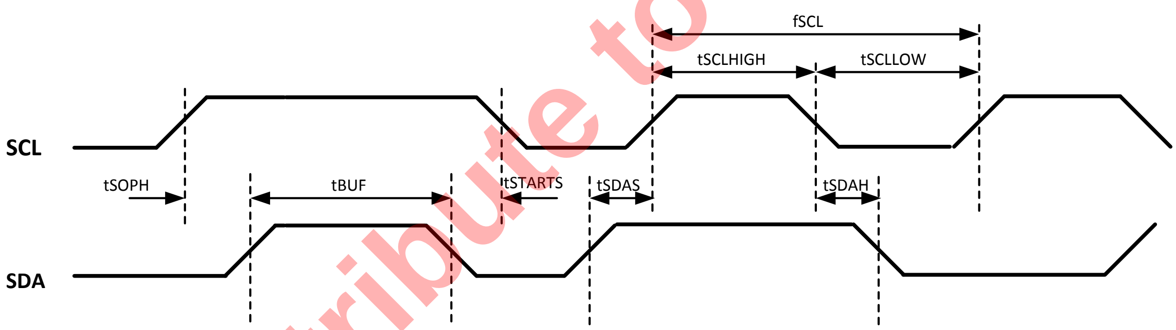


Table VII.38.1 I2C Interface Timing Specification

Symbol	Description	Min.	Max.	Unit	Notes
fSCL	SCL clock frequency		Std. 100 Fast 400	kHz	
tSDAS	Data setup before the rising edge of clock	Std. 4.0 Fast 0.6		µs	
tSDAH	Data hold after the falling edge of clock	Std. 4.0 Fast 0.6	-	µs	
tSTARTS	Clock hold time after the falling edge of SDA when a START command is issued	Std. 4.0 Fast 0.6	-	µs	
tSTOPH	Clock setup time before the rising edge of SDA when a STOP command is issued	Std. 4.0 Fast 0.6		µs	
tBUF	Delay between start and stop	Software Controlled		µs	
tSCLLOW	Clock LOW time	Std. 8.0 Fast 1.3		µs	
tSCLHIGH	Clock HIGH time	Std. 8.0 Fast 1.3		µs	

38.4 Register Description

Each register final address = 0xC1100000 + offset * 4

I2C_M_0_CONTROL_REG 0x2140

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	0	Unused
29-28	R/W	0	QTR_CLK_EXT: These two Bits extend the clock divider to 12 Bits: QTR_CLK = {[29:28],[21:12]}
27	R	0	unused
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 I^2C 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 I^2C 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 I^2C 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 I^2C transaction to get the number of bytes to read from the I^2C 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 I^2C 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(s) 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 I^2C 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.
1	R/W	0	ACK_IGNORE: Set to 1 to disable I^2C ACK detection. The I^2C bus uses an ACK signal after every byte transfer to detect problems during the transfer. Current Software implementations of the I^2C 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 I^2C 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 I^2C operation and generate an I^2C STOP command on the I^2C 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.

I2C_M_0_SLAVE ADDRESS 0x2141

Bit(s)	R/W	Default	Description
31:29	R	0	Reserved
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.
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 I^2C device, or (0xF0 | {A9,A8}) for a 10-bit I^2C device. By convention, the slave address is typically stored in by first left shifting it so that it's MSB is D7 (The I^2C bus assumes

Bit(s)	R/W	Default	Description
			the 7-bit address is left shifted one). Additionally, since the SLAVE address is always an 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.

I2C_M_0_TOKEN_LIST_REG0 0x2142

The register below describes the first 8 tokens in the token list.

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 VII.39.2 Token Definitions

Command Token	Value	Data	Description
END	0x0	N/A	Used to tell the I²C module that this is the end of the Token list. This token is not associated with the I²C 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 I²C device that this is the beginning of an I²C 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 I²C 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.

I2C_M_0_TOKEN_LIST_REG1 0x2143

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

I2C_M_0_TOKEN_WDATA_REG0 0x2144

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.

I2C_M_0_TOKEN_WDATA_REG1 0x2145

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.

I2C_M_0_TOKEN_RDATA_REG0 0x2146

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.

I2C_M_0_TOKEN_RDATA_REG1 0x2146

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.

39. UNIVERSAL ASYNCHRONOUS RECEIVER AND TRANSMITTER

39.1 Overview

There are a number of UART's 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. In some cases, the receive FIFO can be configured to be pushed directly to DDR memory without CPU intervention.

Table VII.39.1 UART List

UART	RX/TX FIFO depths	RX FIFO DMA to DDR	Comment
UART0	128 bytes	Yes	Located in the EE domain
UART1	64 bytes	Yes	Located in the EE domain
UART2	64 bytes	Yes	Located in the EE domain
UART0-AO	64 bytes	No	Located in the Always On domain
UART2-AO	64 bytes	No	Located in the Always On domain

39.2 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 CTS_{stop}. 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.

39.3 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 8 MBits/Sec. 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.

Fig VII.39.1 UART Timing Diagram

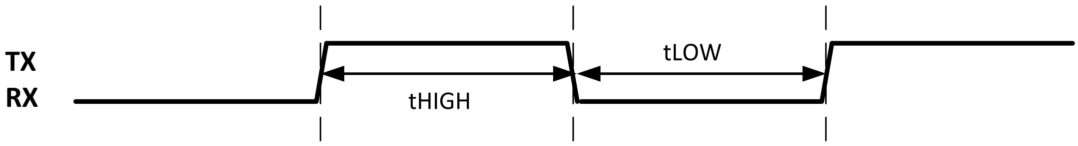


Table VII.39.2 UART Timing Specification

Symbol	Description	Min.	Max.	Unit	Notes
tHIGH	TX/RX high time	60ns	167ms		
tLOW	TX/RX low time	60ns	167ms		

39.4 Register Description

The following register definition is uniformly applied to all UART instantiations in the chip.

Table VII.39.3 UART Register List

Instantiations	Base Address
UART0	0xc1100000+ (0x2130 ~ 0x2135)*4
UART1	0xc1100000+ (0x2137 ~ 0x213c)*4
UART2	0xc1100000+ (0x21C0 ~ 0x21c5)*4
UART0-AO	0xc81004c0 ~ 0xc81004d4
UART2-AO	0xc81004e0 ~ 0xc81004f4

UARTx_WFIFO: Write data

Bit(s)	R/W	Default	Description
31-8	R	0	unused
7-0	R/W	-	Write FIFO data. The Write FIFO holds 64 bytes. The Write FIFO can be written as long as it is not full.

UARTx_RFIFO: Read Data

Bit(s)	R/W	Default	Description
31-8	R	0	unused
7-0	R/W	-	Read FIFO data. The Read FIFO holds 64 bytes. The empty flag can be used to determine if data is available

UARTx_CONTROL: UART Mode

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	Baud rate: This value sets the baud rate by dividing the MPEG system clock.

UARTx_STATUS: UART 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 0x2132
16	R	0	Parity error. To clear This bit, wriTE bit 24 of register 0x2132
15	R	0	Unused
14-8	R	0	Transmit FIFO count. Number of bytes in the transmit FIFO
7	R	0	Unused
8-0	R	0	Receive FIFO count. Number of bytes in the receive FIFO

UARTx_MISC: UART IRQ CONTROL

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.
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	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.

UARTx_REG5

Bit(s)	R/W	Default	Description
31-24	R/W	0	unused
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.

40. INFRARED REMOTE

40.1 Overview

An IR signal based Remote Control (RC) signal decoder and blaster are built in S905X to provide software a low-cost, convenient way to implement remote control function in applications. This module is located in the AO power domain.

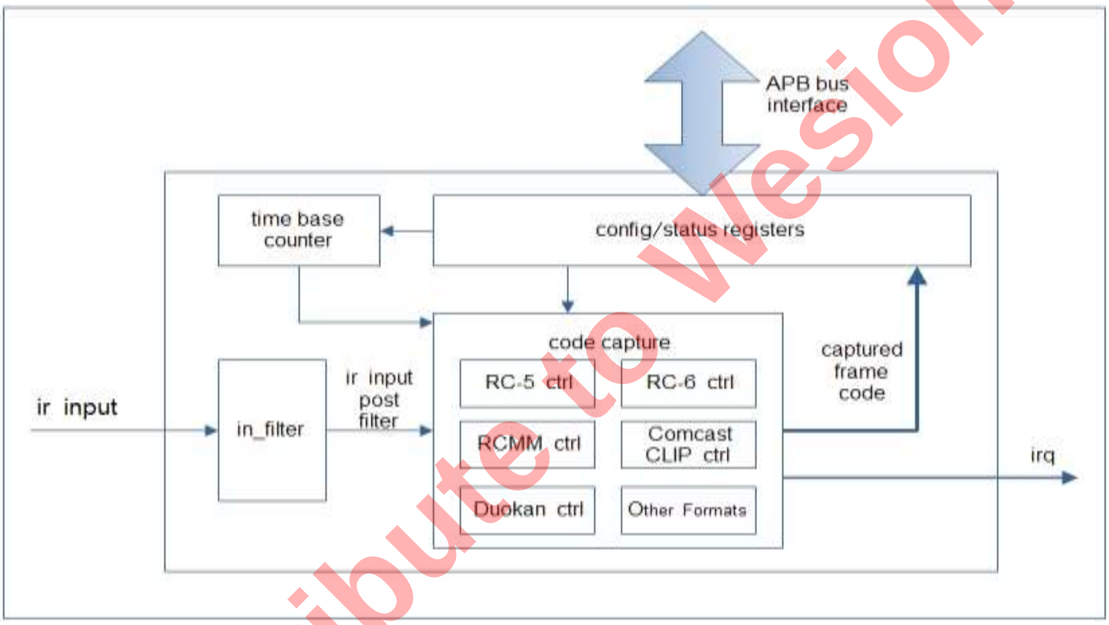
40.2 Decoder Functional Description

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.

Fig VII.40.1 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.

40.3 NEC Infrared Transmission Protocol Example

Message Bit

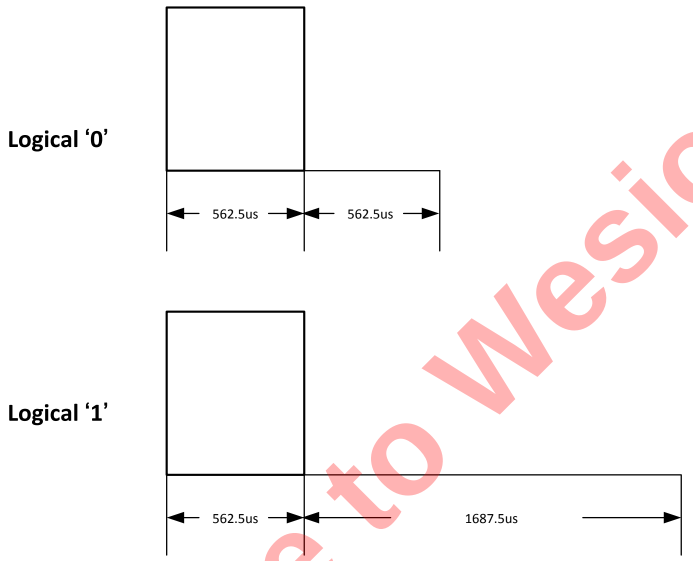
Standard NEC IR transmission protocol uses pulse distance encoding of the message Bits. The basic clock circle time is 562.5us.

The logical Bits are defined and transmitted as follow:

- Logical '0' – a 562.5us pulse burst (1 clock) followed by a 562.5us space (1 clock). The total transmit time is 1.125ms.
- Logical '1' – a 562.5us pulse burst (1 clock) followed by a 1.6875ms space (3 clocks). The total transmit time is 2.25ms.

The signals are illustrated in the picture below.

Fig VII.40.2 Message Bit Signal



Data Frame

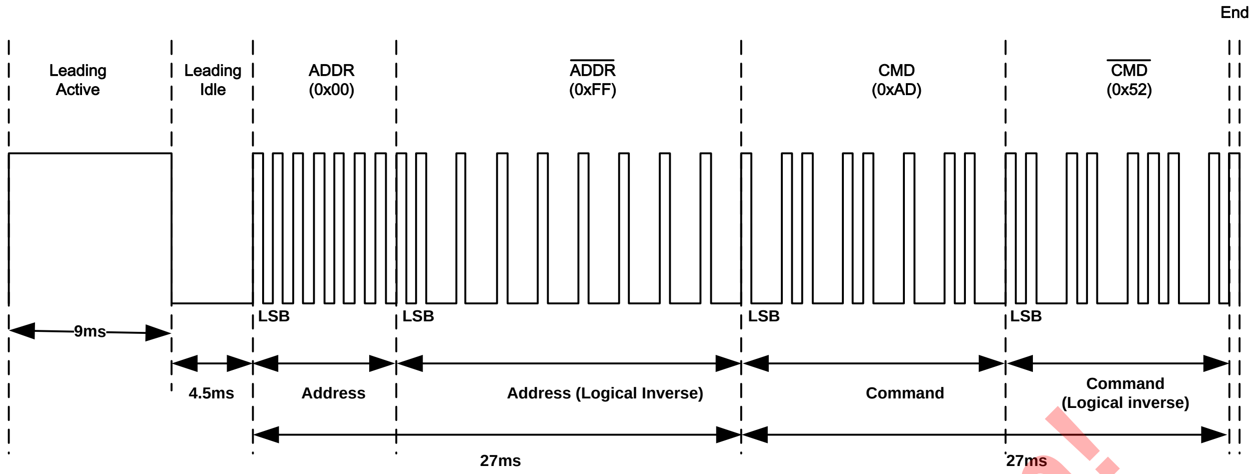
A data transmit frame is exchanged via IR to represent a message when a key is pressed on the remote control. Typically, the data frame consists of, in order:

- A Leading Active – 9ms pulse burst (16 clocks)
- A Leading Idle – 4.5ms space (8 clocks)
- An 8-bit receiver address (ADDR), LSB first.
- The complementary 8-bit receiver address (), LSB first
- An 8-bit command (CMD), LSB first
- The complementary 8-bit command (), LST first
- A final 562.5us pulse burst to indicate the end of message transmission.

The byte ADDR/ and CMD/ are sent with least significant bit (LSB) first.

The example of data frame is illustrated below with ADDR=0x00 and CMD=0xAD.

Fig VII.40.3. Data Frame



Repeat Code(s)

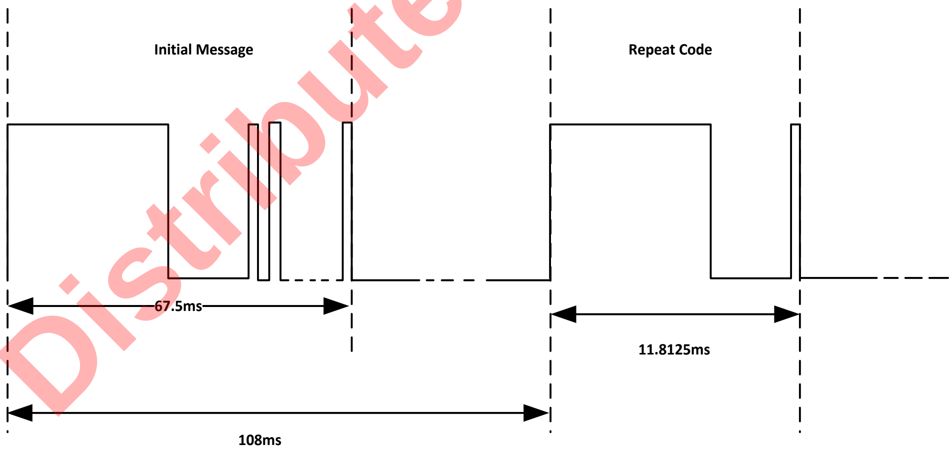
A repeat code will be issued if the key on the remote controller is kept pressed. A repeat code will continue to be sent out at 108ms intervals until the key is released.

Typically, the repeat code consists of, in order:

- A Leading Active – 9ms pulse burst (16 clocks)
- A Leading Idle – 2.25ms space (4 clocks)
- A final 562.5us pulse burst to indicate the end of message transmission.

The example of repeat code is illustrated below

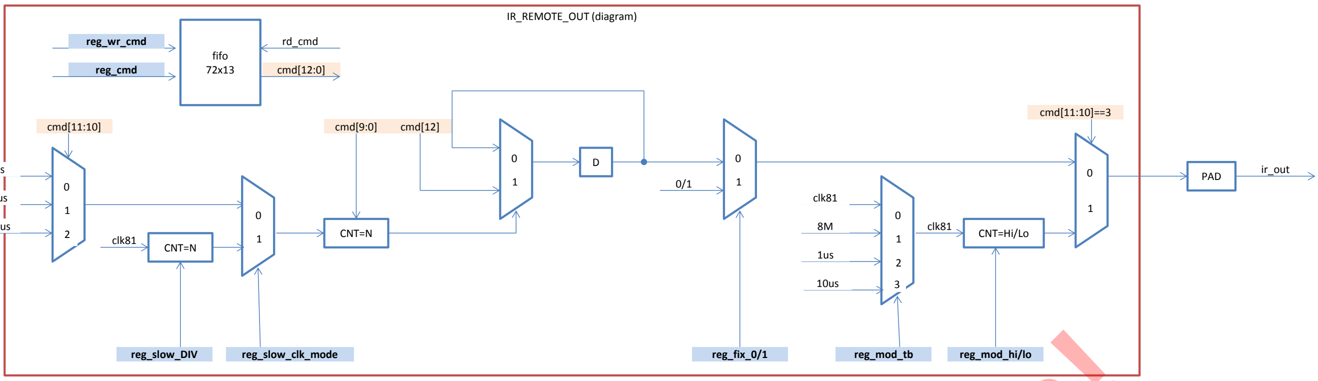
Fig VII.40.4. Repeat Code



40.4 IR RC Output

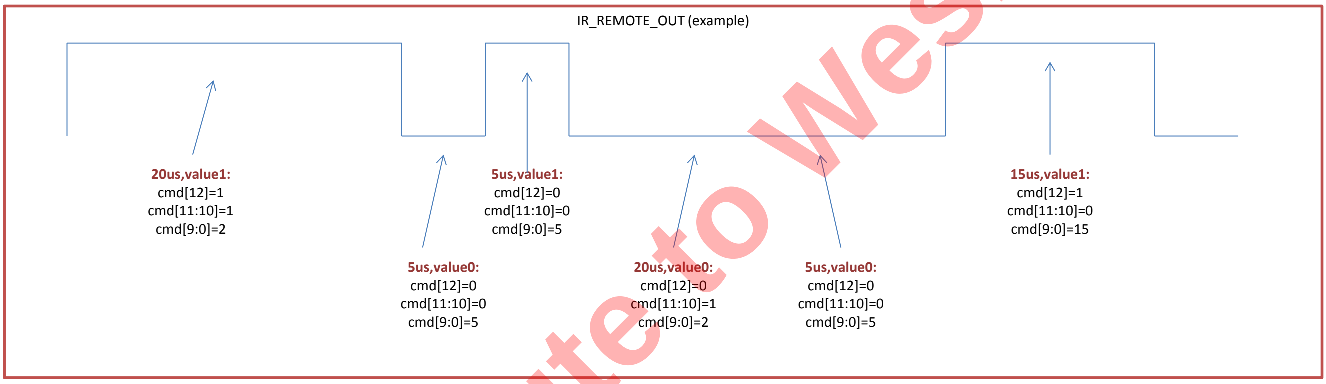
Below shows the output module of IR IC output module. Command signals will be wrote to a 72X13 FIFO, and the circuit will automatically read the 13-bit command data and generate the corresponding output data.

Fig VII.40.5. IR_Remote_output Diagram



Below gives an example of data frame.

Fig VII.40.6. IR_Remote_output Data Frame



40.5 Register Description

AO_MF_IR_DEC_LDR_ACTIVE: Leader Active control 0xc8100580

This register controls the min/max Leader Active time window. For example, for NEC format, the Leader Active time is about 9mS. To identify a Leader Active time between 8.60 mS and 9.40 mS (assuming base resolution = 20uS), user can set Max duration = 0x1d6 (‘d470) to represent 9.40 mS, and set Min duration = 0x1ae (‘d430) to represent 8.60 mS.

Bit(s)	R/W	Default	Description
31-29	R	0	Unused
28-16	R/W	0	Max duration of Leader’s active part
15-13	R	0	Unused
12-0	R/W	0	Min duration of Leader’s active part

AO_MF_IR_DEC_LDR_IDLE: Leader Idle control 0xc8100584

Bit(s)	R/W	Default	Description
31-29	R	0	Unused
28-16	R/W	0	Max duration of Leader’s idle part
15-13	R	0	Unused
12-0	R/W	0	Min duration of Leader’s idle part

AO_MF_IR_DEC_LDR_REPEAT: Repeat Leader Idle Time 0xc8100588

Bit(s)	R/W	Default	Description
31-26	R	0	Unused
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.)
15-10	R	0	Unused
9-0	R/W	0	Min duration of Repeat Code's Leader

AO_MF_IR_DEC_BIT_0: 0xc810058C

Bit(s)	R/W	Default	Description
31-26	R	0	Unused
25-16	R/W	0	Max duration of Duration Setting Regiser 0. It defines max tining 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"
15-10	R	0	Unused
9-0	R/W	0	Min duration of Duration Setting Regiser 0.

AO_MF_IR_DEC_REG0: 0xc8100590

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.

AO_MF_IR_DEC_STATUS: 0xc8100598

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 Regiser 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 Regiser 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 precedance 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

AO_MF_IR_DEC_REG1: 0xc810059C

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. 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

Bit(s)	R/W	Default	Description
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.

AO_MF_IR_DEC_REG2 0xc81005A0

Bit(s)	R/W	Default	Description
31-27	R	0	Unused
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 represet 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.
23-22	R	0	Unused
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 represet 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
13	R/W	0	Enable frame-to-frame time counter (running-counter). 0 = frame-to-frame time counter disabled 1 = frame-to-frame time counter enabled 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 **running_count_tick_mode** and **use_clock_to_counter** settings. When a frame's data are capured and stored into FrameBody/FrameBody_1 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. 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.

Bit(s)	R/W	Default	Description
12	R/W	0	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 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 **repeat_time_max** value by setting **0x38 offset** register. 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.
11	R/W	0	Enable compare frame method for repeat detection. 0 = compare frame method disabled 1 = compare frame method enabled Some IR formats transer the same data frame as repeat frame when the key is kept pressed without release. For repeat detection, compare frame method can be used. 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 repeat_time_max), the status register's **frame_status0** is set to 1 automatically as repeat detected flag. You can configure the **repeat_time_max** value by setting **0x38 offset** register.
10	R/W	0	Enable compare Bit method for repeat detection. 0 = compare Bit method disabled 1 = compare Bit method enabled Some IR formats use only one bit to represet 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.
9	R/W	0	Disable read-clear of FrameBody/FrameBody_1. 0 = read-clear enabled 1 = read-clear disabled FrameBody/FrameBody_1 registers are read-cleared in default. When these register are read, they are cleared to zero. This bit is used to disable this read-clear feature. (FrameBody/FrameBody_1 registers are used to store captured frame data).
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
7:4	R	0	Unused
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 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

AO_MF_IR_DEC_DURATN2 0xc81005A4

Bit(s)	R/W	Default	Description
31-26	R	0	Unused
25-16	R/W	0	Max duration of Duration Setting Regiser 2.

Bit(s)	R/W	Default	Description
			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 Regiser 2.

AO_MF_IR_DEC_DURATN3 0xc81005A8

Bit(s)	R/W	Default	Description
31-26	R	0	Unused
25-16	R/W	0	Max duration of Duration Setting Regiser 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 Regiser 3.

AO_MF_IR_DEC_REG3 0xc81005B8

Bit(s)	R/W	Default	Description
31-20	R	0	Unused
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.

AO_MF_IR_DEC_FRAME: Frame Body (Frame Data, LSB 32Bit) 0xc8100594

Note: New keys will be ignored until ***FrameBody*** register is read if the *hold first key* Bit is set in the decode control register. Reading this register resets an internal frame data valid flag.

Bit(s)	R/W	Default	Description
31-0	R	0	32 bit Read-Only register stores frame body (LSB 32 bit) captured from IR remote data flow, commonly includes custom/address code and data code.

AO_MF_IR_DEC_FRAME: Frame Body 1 (Frame Data, MSB 32Bit) 0xc81005AC

Note: New keys will be ignored until ***FrameBody*** register is read if the *hold first key* Bit is set in the decode control register. Reading this register resets an internal frame data valid flag.

Bit(s)	R/W	Default	Description
31-0	R	0	Stores frame body excess 32 bit range. (MSB 32 bit)

AO_MF_IR_DEC_STATUS_1 0xc81005B0

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.

AO_MF_IR_DEC_STATUS_2 0xc81005B4

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.

IR_BLASTER_CNTL0 0xc81000c0

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.

IR_BLASTER_CNTL1 0xc81000c4

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

IR_BLASTER_CNTL2 0xc81000c8

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
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

41. PULSE-WIDTH MODULATION

41.1 Overview

The chip has 4 PWM modules that can be connected to various digital I/O pins, among which 3 are in EE domain and 1 is in AO domain. 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 couter, generate 18-birt sigma, the PWM-out is the highest sigma. The PWM outputs vs counters are also illustrate below.

Fig VII.41.1 PWM Block Diagram

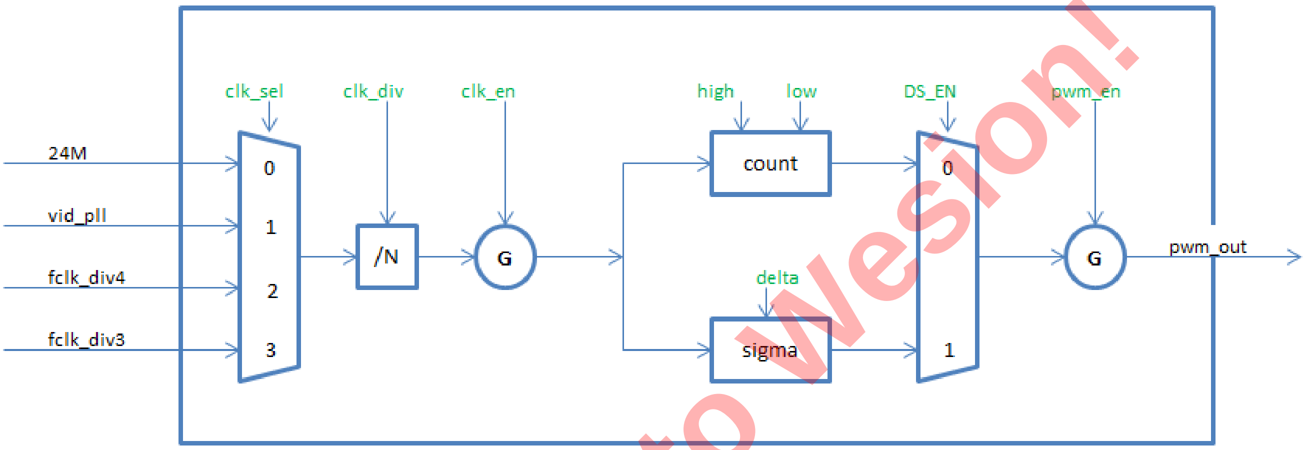


Fig VII.41.2 High/Low counter

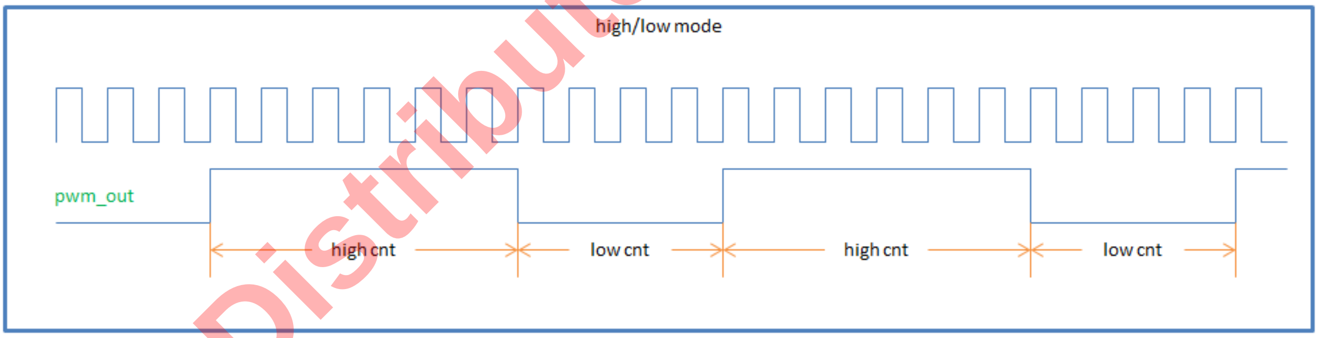
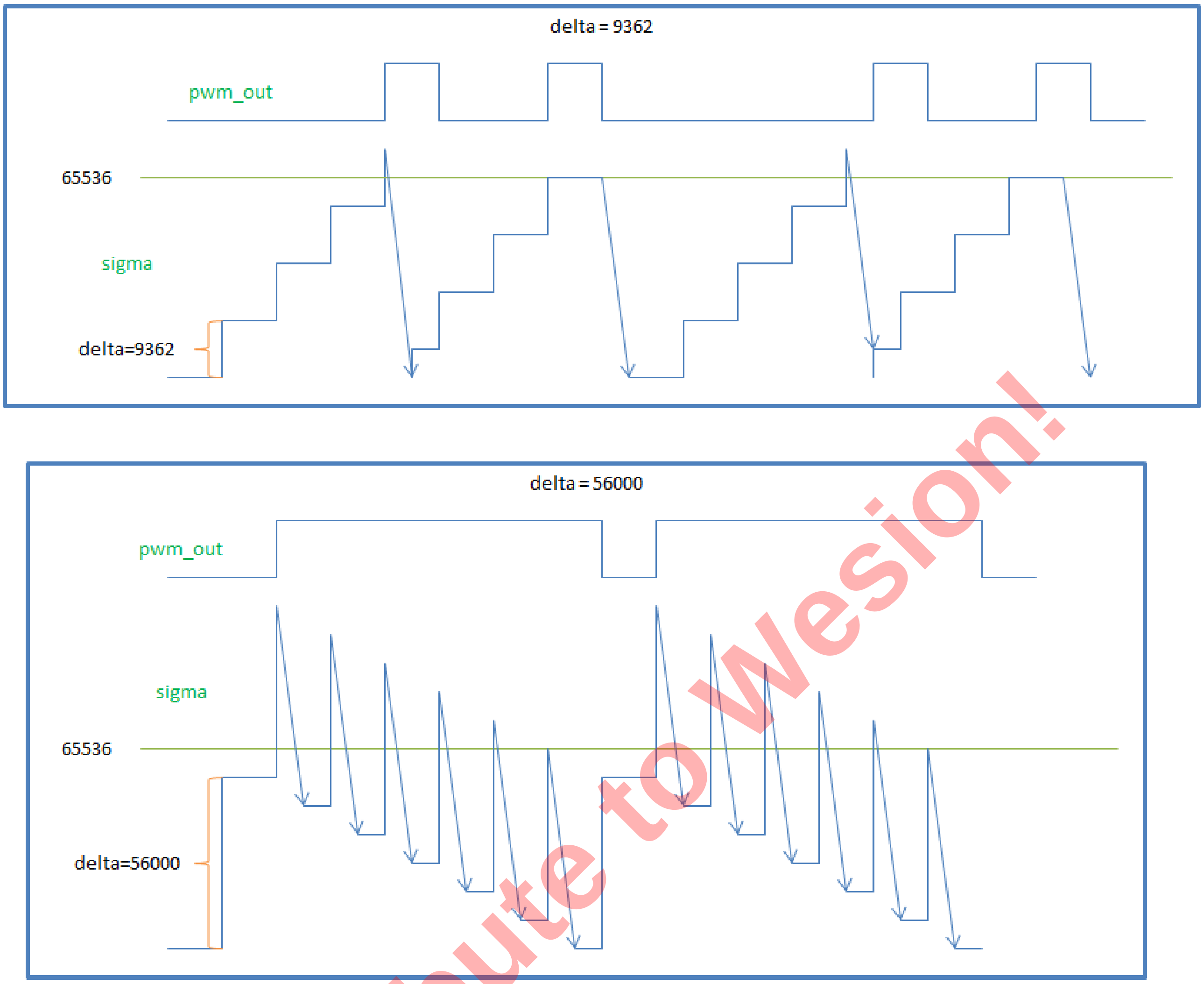
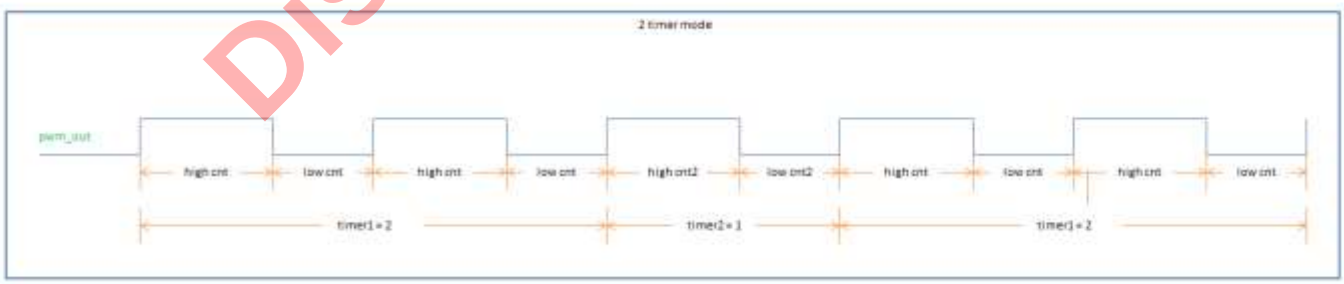


Fig VII.41.3 Delta-sigma conter, delta = 9362 & delta = 56000



PWM 2 timer mode is illustrated as following:

Fig VII.41.4 2 timer mode



41.2 Register Description

Each PWM module contains two PWM generators call A and B and controlled by 4 registers.

For PWM modules in EE domain, the each register's final address = 0xc1100000 + offset*4

PWM_PWM_A 0x2154

Bit(s)	R/W	Default	Description
31-15	R/W	0	PWM_A_HIGH: This sets the high time (in clock counts) for the PWM_A generator output
15-0	R/W	0	PWM_A_LOW: This sets the high time (in clock counts) for the PWM_A generator output

PWM_PWM_B 0x2155

Bit(s)	R/W	Default	Description
31-15	R/W	0	PWM_B_HIGH: This sets the high time (in clock counts) for the PWM_A generator output
15-0	R/W	0	PWM_B_LOW: This sets the high time (in clock counts) for the PWM_A generator output

PWM_MISC_REG_AB 0x2156

Bit(s)	R/W	Default	Description
31-26	R	0	Unused
25	R/W	0	Pwm_A 2 timer enable
24	R/W	0	Pwm_B 2 timer enable
23	R/W	0	PWM_B_CLK_EN: Set this bit to 1 to enable PWM A clock
22-16	R/W	0	PWM_B_CLK_DIV: Selects the divider (N+1) for the PWM A 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.

DS_A_B 0x2157

Bit(s)	R/W	Default	Description
31-15	R/W	0	DS_B_VAL: This value represents the delta sigma setting for channel B (PWM_B)
15-0	R/W	0	DS_A_VAL: This value represents the delta sigma setting for channel A (PWM_A)

PWM_TIME_AB 0x2158

Bit(s)	R/W	Default	Description
31-24	R/W	0	A_timer1
23:16	R/W	0	A_timer2
15:8	R/W	0	B_timer1
7:0	R/W	0	B_timer2

PWM_A2 0x2159

Bit(s)	R/W	Default	Description
31-15	R/W	0	PWM_A2_HIGH: This sets the high time (in clock counts) for the PWM_A generator output
15-0	R/W	0	PWM_A2_LOW: This sets the high time (in clock counts) for the PWM_A generator output

PWM_B2 0x215a

Bit(s)	R/W	Default	Description
31-15	R/W	0	PWM_B2_HIGH: This sets the high time (in clock counts) for the PWM_A generator output
15-0	R/W	0	PWM_B2_LOW: This sets the high time (in clock counts) for the PWM_A generator output

PWM_PWM_C 0x2194

PWM_PWM_D 0x2195

PWM_MISC_REG_CD 0x2196

PWM_DELTA_SIGMA_CD 0x2197

PWM_PWM_E 0x21b0

PWM_PWM_F 0x21b1

PWM_MISC_REG_EF 0x21b2

PWM_DELTA_SIGMA_EF 0x21b3

For PWM modules in AO domain, the each register's final address = 0xc8100400 + offset*4

AO_PWM_PWM_A 0x54

AO_PWM_PWM_B 0x55

AO_PWM_MISC_REG_AB 0x56

AO_PWM_DELTA_SIGMA_AB 0x57

42. SAR ADC

42.1 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 of the mux 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.

42.2 Register Description

Each register final address = 0xC1100000 + offset * 4

SAR_ADC_REG0: Control Register #0 0x21a0

Bit(s)	R/W	Default	Description
31	R	0	PANEL DETECT level.
30	R/W	0	DELTA_BUSY: If This bit is 1, then it indicates the delta processing engine is busy
29	R/W	0	AVG_BUSY: If This bit is 1, then it indicates the averaging engine is busy
28	R/W	0	SAMPLE_BUSY: If This bit is 1, then it indicates the sampling engine is busy
27	R/W	0	FIFO_FULL:
26	R/W	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/W	0	ADC_TEMP_SEN_SEL
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.

SAR_ADC_CHAN_LIST:Channel List 0x21a1

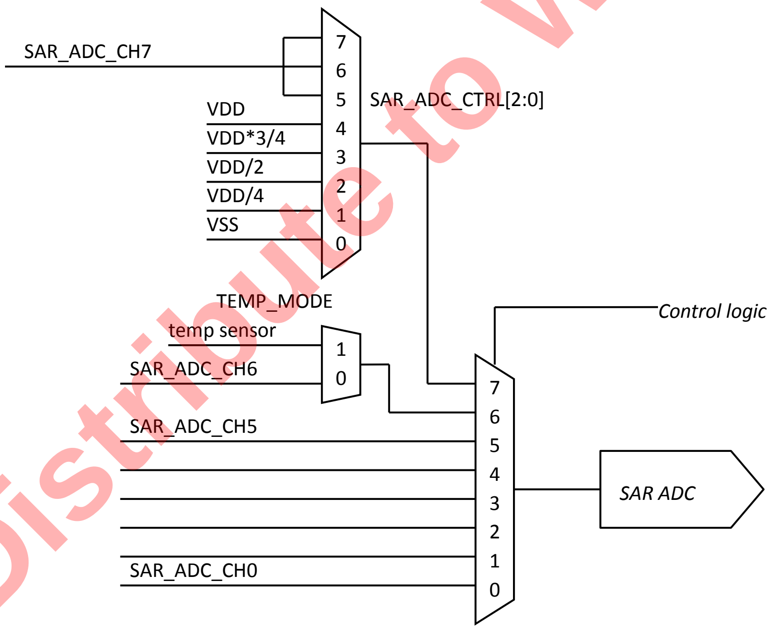
Bit(s)	R/W	Default	Description
31-27	R/W	0	unused
26-24	R/W	2	Length of the list of channels to process. If this value is 2, then only channels in Bits [8:0] below are processed.
23-21	R/W	7	8th channel
20-18	R/W	6	7th channel
17-15	R/W	5	6th channel
14-12	R/W	4	5th channel
11-9	R/W	3	4th channel
8-6	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

SAR_ADC_AVG_CNTL:Sampling/Averaging Modes 0x21a2

Each channel listed in the CHANNEL_LIST is given independent control of the number of samples to acquire and averaging mode

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.
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 2^N:
13-12	R/W	0	Channel 6: Number of samples to acquire 2^N:
11-10	R/W	0	Channel 5: Number of samples to acquire 2^N:
9-8	R/W	0	Channel 4: Number of samples to acquire 2^N:
7-6	R/W	0	Channel 3: Number of samples to acquire 2^N:
5-4	R/W	0	Channel 2: Number of samples to acquire 2^N:
3-2	R/W	0	Channel 1: Number of samples to acquire 2^N:
1-0	R/W	0	Channel 0: Number of samples to acquire 2^N: 0 = 1, 1 = 2, 2 = 4, 4 = 8.

SAR_ADC_REG3: Control Register #3 0x21a3



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

Bit(s)	R/W	Default	Description
			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
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.

SAR_ADC_DELAY:INPUT / SAMPLING DELAY 0x21a4

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.

Bit(s)	R/W	Default	Description
15-10	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
16-23	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	0	unused
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.

SAR_ADC_LAST_RD: Last Sample 0x21a5

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.

Bit(s)	R/W	Default	Description
31-24	R	0	unused
23-16	R	0	LAST_CHANNEL1
15-10	R	0	unused
9-0	R	0	LAST_CHANNEL0

SAR_ADC_FIFO_RD: Control Register #6 (FIFO RD) 0x21a6

Bit(s)	R/W	Default	Description
31-16	R	0	Unused
15	R	0	Unused
14-12	R	0	Channel ID. This value identifies the channel associated with the data in Bits [9:0] below
11-10	R	0	Unused
9-0	R	0	Sample value: 9-bit raw or averaged ADC sample written to the FIFO.

SAR_ADC_AUX_SW:Channel 2~7 ADC MUX, Switch Controls 0x21a7

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).

Bit(s)	R/W	Default	Description
31-26	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

SAR_ADC_CHAN_10_SW:Channel 0, 1 ADC MUX, Switch Controls 0x21a8

Channels 0 and 1 have independent programmable switch settings when either/both of these channels are being measured.

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
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

SAR_ADC_DETECT_IDLE_SW:DETECT / IDLE Mode switches 0x21a9

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.

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
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 YM_DRIVE_SW setting: 0: TADC_CH4N = floating 1: TADC_CH4N = GND
0	R/W	0	IDLE MODE YM_DRIVE_SW setting: 0: TADC_CH4N = floating 1: TADC_CH4N = GND

SAR_ADC_DELTA_10:Delta Mode Deltas 0x21aa

Bit(s)	R/W	Default	Description
31-28	R	0	unused
27	R/W		TEMP_SEL
26	R/W		TS_REVE[1]
25-16	R/W	0	Channel 1 delta value when delta processing for channel 1 is enabled.
15	R/W		TS_REVE[0]
14-11	R/W		TS_C[3:0]
10	R/W	0	TS_VBG_EN
9-0	R/W	0	Channel 0 delta value when delta processing for channel 0 is enabled.

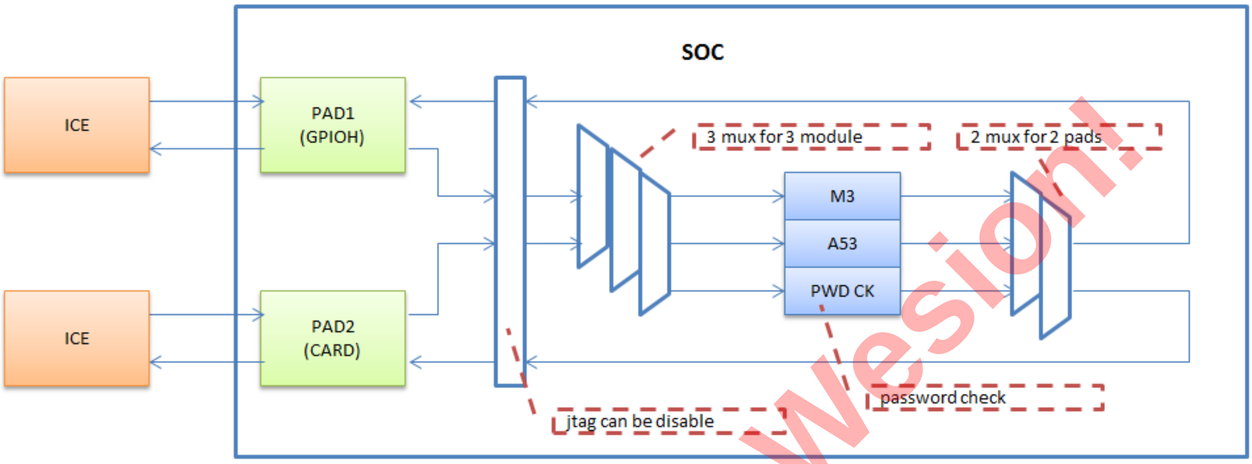
Section VIII System Interface

43. JTAG

43.1 Overview

JTAG is an interface for internal test. The structure of S905X JTAG module is shown in the following diagram:

Fig VIII.43.1 Diagram of JTAG



43.2 Register Definition

AO_SEC_REG1 0xc8100144

A53/Secure	A53/Non Secure	M3/Secure	M3/Non Secure	Reset by watchdog
No access	No access	R/W	No access	Yes

Bit(s)	R/W	Default	Description
31	R	0	Jtag_pwd_data_valid
30	R	0	Jtag_timeout
29	R	0	Jtag_en
28	R	0	Noused
27	R/W	0	Unused
26	R/W	0	jtag_pwd_clr
25	R/W	0	jtag_en
24	R/W	0	jtag_pwd_en
23-18	R/W	0	Unused
17-16	R/W	0	jtag_sel_jtag_pw_mux , / 0: disabled, 1: GPIOH, 2: CARD, 3: reserved
15	R/W	0	jtag_sel_force_tdo_low
14-13	R/W	0	jtag_sel_ee_pad_tdo , 0: disabled, 1: AO-CPU, 2: System-CPU, 3: JTAG password
12-10	R/W	0	jtag_sel_sys_cpu_tdi , 0: disabled, 1: GPIOH, 2: CARD, 3: TDO from AO
9-8	R/W	0	jtag_sel_sys_cpu_tms_tck , 0: disabled, 1: GPIOH, 2: CARD, 3: reserved
7	R/W	0	Unused
6-6	R/W	0	jtag_sel_ao_pad_tdo , 0: disabled, 1: AO-CPU, 2: System-CPU, 3: JTAG password
4-2	R/W	0	jtag_sel_ao_cpu_tdi , 0: disabled, 1: GPIOH, 2: CARD, 3: TDO from AO
1-0	R/W	0	Jtag_sel_ao_cpu_tms_tck , 0: disabled, 1: GPIOH, 2: CARD, 3: reserved

44. Temp Sensor

44.1 Overview

The temperature sensor uses TSMC 28nm CMOS mix signal process and can provide a PTAT voltage which contains the temperature information of the chip. This block together with the followed SAR ADC can give the chip temperature. Figure shows block diagram of the temperature sensor.

FigVIII.44.1 Temperature Sensor Diagram

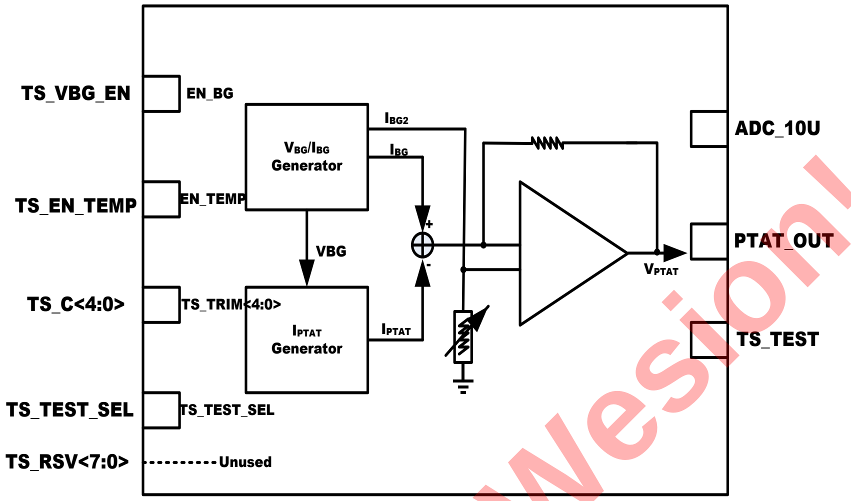
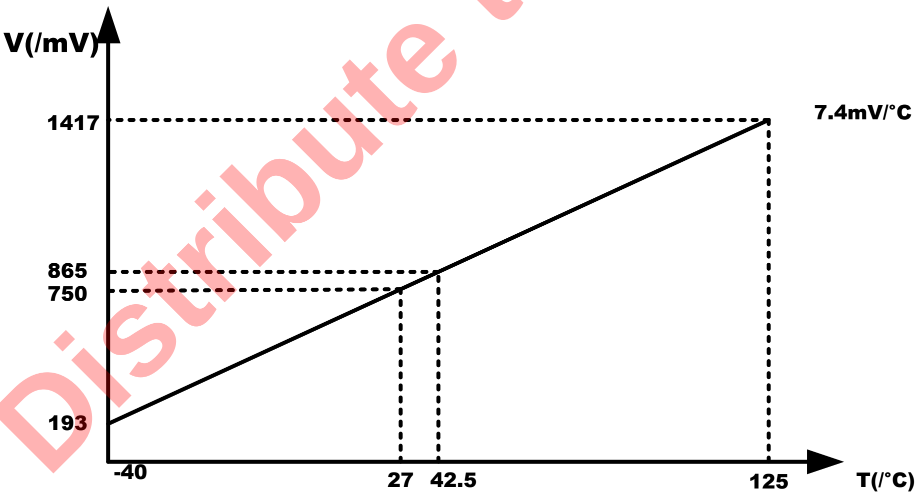


Fig VIII.44.2 Temperature sensor output

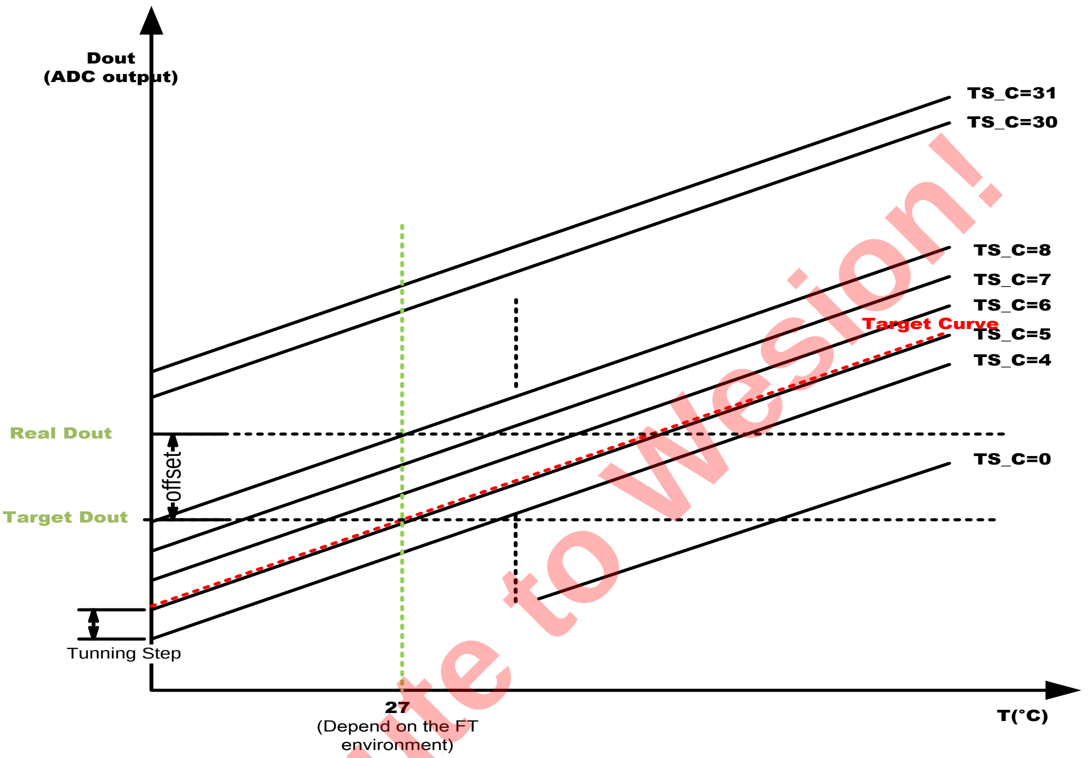


44.2 Trimming

Fig.2 denotes the ideal output voltage of this block which also gives the direction of trimming. We use 16 Bits e-fuse resistor to trim this block, 10 Bits for the recording of the result of SARADC in FT process , 5 Bits for coarse trimming(Anti-saturation of this block) and 1 bitfor trimming fag. Two fixed points (27°,750mV) and (42.5, 865mV) are provided in Fig.2, you can select either one of them as the coarse trimming target.

Fig.3 shows the real calibration curve for temperature sensor, please note the Y coordinate is different from Fig.2, but they are the input and output of the SARADC respectively. In chip application, the temperature can be calculated when we get the real time Dout if the line is determined. While in theory we can determine a line if the slope and a real arbitrary point are given, that is to say the slope(given by designer) and the Dout at a specified temperature in Fig.3.

Fig VIII.44.3 Temperature sensor calibration curve



Below is the calibration flow:

1. Get the real Dout when set the TS_C to 16;
2. According to the offset and Tunning step the ΔTS_C can be calculated(The tunning step and target Dout are given by designer);
3. Calculate the final TS_C, TS_C$_{final}$=16+ΔTS_C;
4. Read the Dout when set the TS_C to the final value;
5. Store the final the TS_C and the Dout which are get in step 4 to effuse;

44.3 Register Definition

0xc810062c

Bits	R/W	Default	Description
31-30	R	reserved	-
29-22	R/W	00000000	TS_RSV<7:0>:Unused bit, can set to 0
21	R/W	1	Enable SARADC channel 6 sampling
20	R/W	0	TS_TEST_SEL: Enable the test mode
19	R/W	1	TS_EN_TEMP:Write 1 to enable temperature sensor circuit.
18-14	R/W	10000	TS_C<4:0>.Trimming temperature sensor's output voltage.0000: 550mV 1111:850mV

13	R/W	1	TS_VBG_EN. Write 1 to enable bandgap.
12-0	R/W	reserved	-