OmniVision™

datasheet

PRODUCT SPECIFICATION

four channel MIPI bridge controller

For the latest data sheet, please visit www.sunnywale.com

OV683

four channel MIPI bridge controller

datasheet
PRODUCT SPECIFICATION

version 2.0
june 2016

To learn more about OmniVision Technologies, visit www.ovt.com.

applications

- stand alone 3D bridge chip for HD sensors

ordering information

- **OV00683-B33G-Z** (lead free)
  133-pin BGA

features

interfaces

- two 2-lane MIPI receiver for video input
- one of the MIPI receivers can be divided into dual 1-lane MIPI receivers
- one 4-lane MIPI receiver for video input
- one 4-lane MIPI transmitter for video output
- up to 1MHz SCCB with 13MHz~26MHz input clock
- two sets of SCCB master
- one set of SCCB master and slave
- four sensor frame rate control pin
- five general purpose IO (GPIO) pins
- UART and SPI interfaces

on-chip PLLs

- system PLL - input clock frequency ranges from 13MHz to 26MHz
- MIPI speed - 5x or 10x of system clock for raw, 4x or 8x of system clock for YUV

image signal processor (ISP)

- AEC/AGC/AWB
- two ISPs, one for each input video stream
- 2592x1944 max resolution
- max frame rate: 24fps at 5MP, 30fps at 4MP, 60fps at 1080p, 120fps at 720p
- lens shading correction (LENC)
- auto exposure and gain control
- auto white balance
- defect pixel correction
- auto contrast enhancement
- gamma correction
- YCbCr422 process

SCCB

- two sets of SCCB masters to control multiple sensors
- one SCCB master/slave to take the commands from host controller
- 7-bit SCCB slave device ID is fixed to 0x44 (0x88 for write, and 0x89 for read)
- 7-bit SCCB slave device ID is fixed to 0x42 (0x84 for write, and 0x85 for read)
- supports SCCB clock 100kHz and 400kHz and 1MHz

data format

- input: raw 8/10/12-bit, YUV422
- output: raw 8/10/12-bit, YUV422

microcontroller

- 32-bit microcontroller running at the system clock
- 64KByte program memory, 32KB ROM

power supply

- 1.8V for IO voltage (e.g., PADVDD18), 1.8V for analog voltage (e.g., M*AVDD)
- internal regulator generates 1.2V C*VDD12 from PADVDD18 for the digital core circuit
- hardware standby mode initiated by pulling PWDN high, whole system halts and input clock is gated
- software standby mode initiated by register

key specifications

- **power supply:**
  core: 1.2V
  analog: 1.8V
  I/O: 1.8V
- **power requirements:**
  active: (see **table 7-3**)
  hardware standby: 100 µW
- **temperature range:**
  operating: -30°C to 70°C junction temperature
- **output formats:** 8-bit and 10-bit Raw RGB data, YUV422 data
- **input clock frequency:** 6~27 MHz
- **maximum image transfer rate:** 120 fps
- **package dimensions:** 7 mm x 7 mm

table of contents

list of figures

list of tables

1 signal descriptions

table 1-1 lists the signal descriptions and their corresponding pin numbers for the OV683 bridge chip. The package information is shown in **section 8**.

table 1-1 signal descriptions (sheet 1 of 5)

pin number	signal name	pin type	description	voltage
A2	GPIO4	I/O	general purpose I/O 4	
A3	GPIO1	I/O	general purpose I/O 1	
A4	PWDN	input	chip power-down (active high)	
A5	XI	input	crystal clock input	
A6	XO	output	crystal clock output	
A7	MTX0_AVSS	analog I/O	MIPI TX0 analog ground	GND
A8	MTX0_DN3	analog I/O	MIPI TX0 differential negative data lane 3	
A9	MTX0_DN1	analog I/O	MIPI TX0 differential negative data lane 1	
A10	MTX0_CN	analog I/O	MIPI TX0 differential negative clock lane	
A11	MTX0_DN0	analog I/O	MIPI TX0 differential negative data lane 0	
A12	MTX0_DN2	analog I/O	MIPI TX0 differential negative data lane 2	
A13	MTX0_AVSS	analog I/O	MIPI TX0 analog ground	GND
B1	FLUSH	I/O	flush/reset chip data-path	
B3	GPIO3	I/O	general purpose I/O 3	
B5	TMODE	input	test mode enable	
B7	MTX0_AVDD	analog I/O	MIPI TX0 analog power	1.8V
B8	MTX0_DP3	analog I/O	MIPI TX0 differential positive data lane 3	
B9	MTX0_DP1	analog I/O	MIPI TX0 differential positive data lane 1	
B10	MTX0_CP	analog I/O	MIPI TX0 differential positive clock lane	
B11	MTX0_DP0	analog I/O	MIPI TX0 differential positive data lane 0	
B12	MTX0_DP2	analog I/O	MIPI TX0 differential positive data lane 2	
B13	MTX0_AVSS	analog I/O	MIPI TX0 analog ground	GND
C1	UART RX	I/O	UART receive data	
C2	UART_TX	I/O	UART transmit data	
C3	PADVSS	ground	pad ground	GND

table 1-1 signal descriptions (sheet 2 of 5)

pin number	signal name	pin type	description	voltage
C4	GPIO0	I/O	general purpose I/O 0	
C6	BOOT0	input	boot mode and test mode selection	
C9	MTX0_DVDD	analog I/O	MIPI TX0 digital power (connect to VROPK or external 1.2V)	1.2V
C10	MTX0_DVDD	analog I/O	MIPI TX0 digital power (connect to VROPK or external 1.2V)	1.2V
D1	MISO	I/O	SPI master input slave output	
D2	SSN	I/O	SPI chip select	
D3	GPIO2	I/O	general purpose I/O 2	
D5	RESETB	input	chip reset pin (active low)	
D6	BOOT1	input	boot mode and test mode selection	
D7	PADVDD18	power	pad power 1.8V from host 1.8V	1.8V
D8	CPKVDD12	power	core (power keep domain) power 1.2V (connect to VROPK)	1.2V
D9	CPDVDD12	power	core (power down domain) power 1.2V (connect to VROPD)	1.2V
D10	CVSS	ground	core ground	GND
E1	MOSI	I/O	SPI master output slave input	
E3	SPICK	I/O	SPI clock	
E4	SCCB_ID	I/O	SCCB slave ID selection	
E8	CVSS	ground	core ground	GND
E10	MTX0_AVDD	analog I/O	MIPI TX0 analog power	1.8V
E11	MTX0_DVDD	analog I/O	MIPI TX0 digital power (connect to VROPK or external 1.2V)	1.2V
F1	MSSCL	I/O	SCCB master and slave clock	
F2	MSSDA	I/O	SCCB master and slave data	
F3	CPDVDD12	power	core (power down domain) power 1.2V (connect to VROPD)	1.2V
F6	CVSS	ground	core ground	GND
F7	PADVSS	ground	pad ground	GND
F8	CVSS	ground	core ground	GND
F9	CVSS	ground	core ground	GND

table 1-1 signal descriptions (sheet 3 of 5)

pin number	signal name	pin type	description	voltage
F10	MTX0_AVDD	analog I/O	MIPI TX0 analog power	1.8V
F11	MTX0_DVDD	analog I/O	MIPI TX0 digital power (connect to VROPK or external 1.2V)	1.2V
G1	MRX1_AVSS	analog I/O	MIPI RX1 analog ground	GND
G2	MRX1_DVDD	analog I/O	MIPI RX0 digital power (connect to VROPD)	1.2V
G4	PADVDD18	power	pad power 1.8V from host 1.8V	1.8V
G6	PADVSS	ground	pad ground	GND
G7	CVSS	ground	core ground	GND
G8	CPDVDD12	power	core (power down domain) power 1.2V (connect to VROPD)	1.2V
G9	VROPD	power	regulator output power (connect to CPDVDD12, M*DVDD)	1.2V
G10	VROPD	power	regulator output power (connect to CPDVDD12, M*DVDD)	1.2V
H1	MRX1DP2	analog I/O	MIPI RX1 differential positive data lane 2	
H2	MRX1DN2	analog I/O	MIPI RX1 differential negative data lane 2	
H4	CPKVDD12	power	core (power keep domain) power 1.2V (connect to VROPK)	1.2V
H6	CVSS	ground	core ground	GND
H7	CVSS	ground	core ground	GND
H8	PADVSS	ground	pad ground	GND
H9	CPKVDD12	power	core (power keep domain) power 1.2V (connect to VROPK)	1.2V
H10	VRI	power	regulator input (host 1.8V)	1.8V
H11	VRI	power	regulator input (host 1.8V)	1.8V
H13	MTX0_AVSS	analog I/O	MIPI TX0 analog ground	GND
J1	MRX1DP0	analog I/O	MIPI RX1 differential positive data lane 0	
J2	MRX1DN0	analog I/O	MIPI RX1 differential negative data lane 0	
J3	MRX1_AVDD	analog I/O	MIPI RX1 analog power	1.8V
J4	MRX1_DVDD	analog I/O	MIPI RX0 digital power (connect to VROPD)	1.2V
J10	FSYNC1	I/O	sensor frame sync 1	
J11	FSYNC3	I/O	sensor frame sync 3	

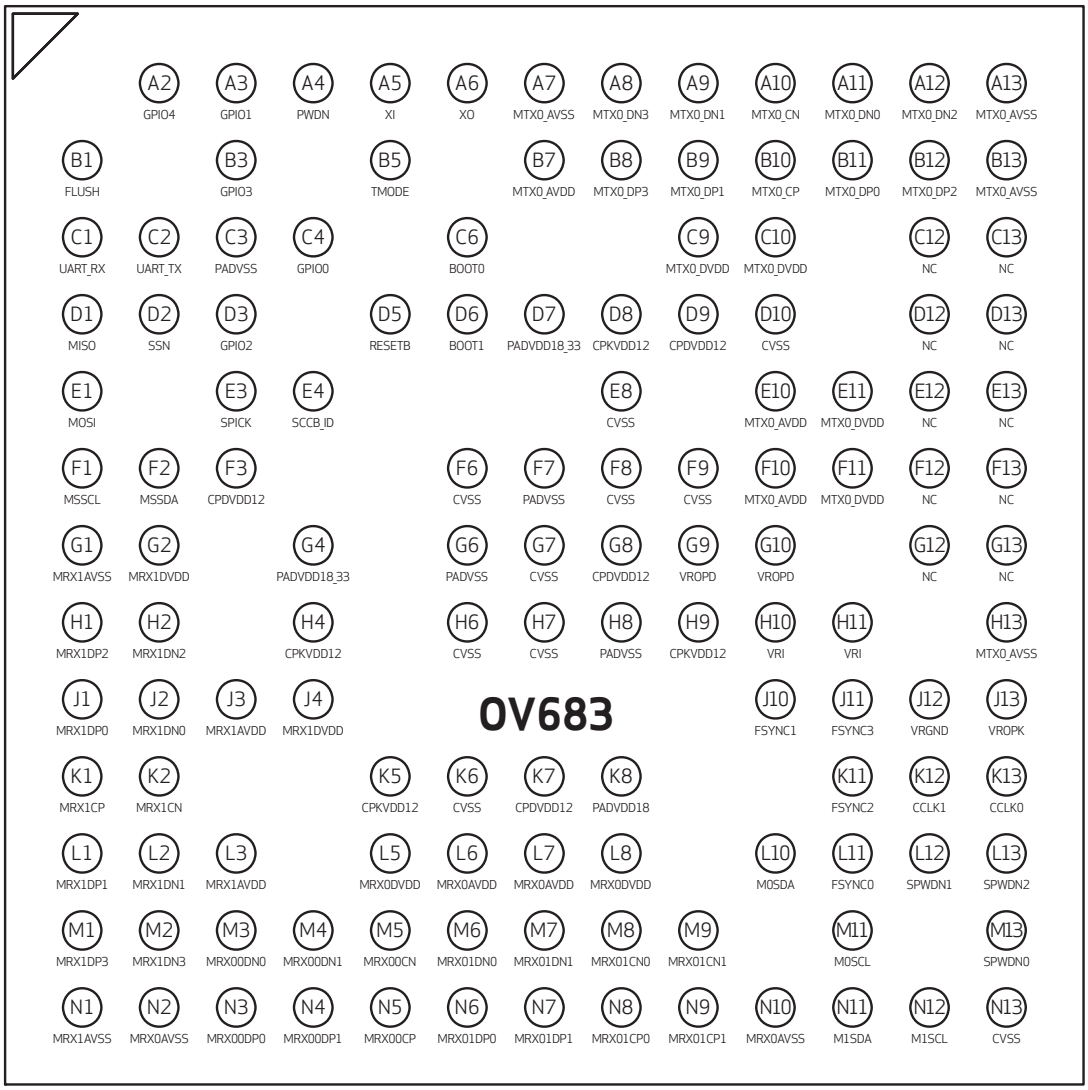
table 1-1 signal descriptions (sheet 4 of 5)

pin number	signal name	pin type	description	voltage
J12	VRGND	ground	regulator ground	GND
J13	VROPK	power	regulator output power (connect to CPKVDD12)	1.2V
K1	MRX1CP	analog I/O	MIPI RX1 differential positive clock lane	
K2	MRX1CN	analog I/O	MIPI RX1 differential negative clock lane	
K5	CPKVDD12	power	core (power keep domain) power 1.2V (connect to VROPK)	1.2V
K6	CVSS	ground	core ground	GND
K7	CPDVDD12	power	core (power down domain) power 1.2V (connect to VROPD)	1.2V
K8	PADVDD18	power	pad power 1.8V from host 1.8V	1.8V
K11	FSYNC2	I/O	sensor frame sync 2	
K12	CCLK1	I/O	sensor reference clock 1	
K13	CCLK0	I/O	sensor reference clock 0	
L1	MRX1DP1	analog I/O	MIPI RX1 differential positive data lane 1	
L2	MRX1DN1	analog I/O	MIPI RX1 differential negative data lane 1	
L3	MRX1_AVDD	analog I/O	MIPI RX1 analog power	1.8V
L5	MRX0DVDD	analog I/O	MIPI RX00 and RX01 digital power (connect to VROPD)	1.2V
L6	MRX0AVDD	analog I/O	MIPI RX00 and RX01 analog power	1.8V
L7	MRX0AVDD	analog I/O	MIPI RX00 and RX01 analog power	1.8V
L8	MRX0DVDD	analog I/O	MIPI RX00 and RX01 digital power (connect to VROPD)	1.2V
L10	M0SDA	I/O	SCCB master 0 data	
L11	FSYNC0	I/O	sensor frame sync 0	
L12	SPWDN1	I/O	sensor power down 1	
L13	SPWDN2	I/O	sensor power down 2	
M1	MRX1DP3	analog I/O	MIPI RX1 differential positive data lane 3	
M2	MRX1DN3	analog I/O	MIPI RX1 differential negative data lane 3	
M3	MRX00DN0	analog I/O	MIPI RX00 differential negative data lane 0	
M4	MRX00DN1	analog I/O	MIPI RX00 differential negative data lane 1	
M5	MRX00CN	analog I/O	MIPI RX01 differential negative clock lane	

table 1-1 signal descriptions (sheet 5 of 5)

pin number	signal name	pin type	description	voltage
M6	MRX01DN0	analog I/O	MIPI RX01 differential negative data lane 0	
M7	MRX01DN1	analog I/O	MIPI RX01 differential negative data lane 1	
M8	MRX01CN0	analog I/O	MIPI RX01 differential negative clock lane 1	
M9	MRX01CN1	analog I/O	MIPI RX01 differential negative clock lane 1	
M11	M0SCL	I/O	SCCB master 0 clock	
M13	SPWDN0	I/O	sensor power-down 0	
N1	MRX1_AVSS	analog I/O	MIPI RX1 analog ground	GND
N2	MRX0AVSS	analog I/O	MIPI RX00 and RX01 analog ground	GND
N3	MRX00DP0	analog I/O	MIPI RX00 differential positive data lane 0	
N4	MRX00DP1	analog I/O	MIPI RX00 differential positive data lane 1	
N5	MRX00CP	analog I/O	MIPI RX00 differential positive clock lane	
N6	MRX01DP0	analog I/O	MIPI RX01 differential positive data lane 0	
N7	MRX01DP1	analog I/O	MIPI RX01 differential positive data lane 1	
N8	MRX01CP0	analog I/O	MIPI RX01 differential positive clock lane 0	
N9	MRX01CP1	analog I/O	MIPI RX01 differential positive clock lane 1	
N10	MRX0AVSS	analog I/O	MIPI RX00 and RX01 analog ground	GND
N11	M1SDA	I/O	SCCB master 1 data	
N12	M1SCL	I/O	SCCB master 1 clock	
N13	CVSS	ground	core ground	GND

figure 1-1 pin diagram



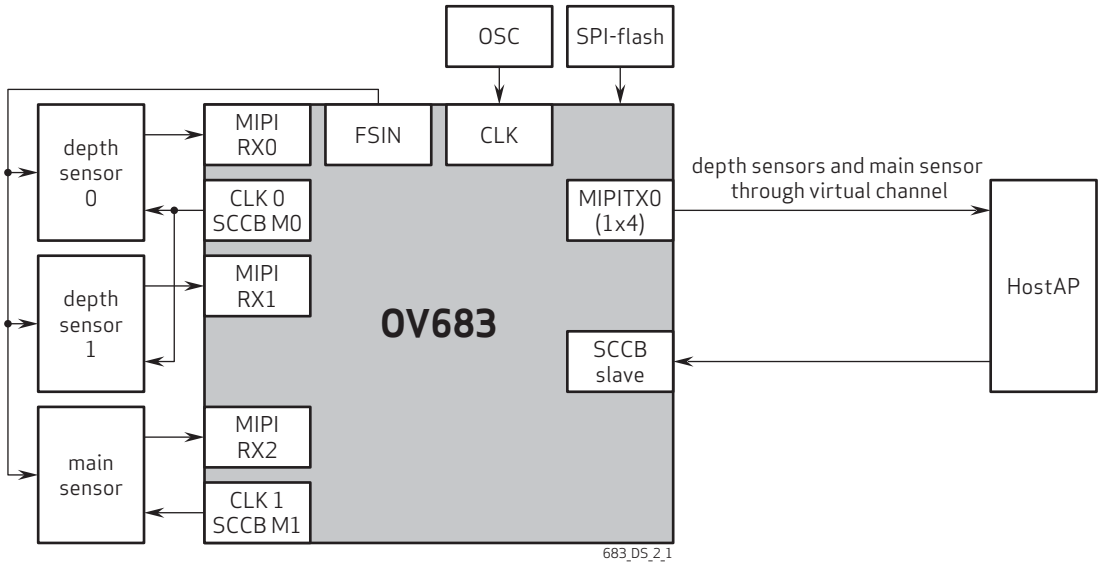
683_DS_1_1

2 system level description

2.1 overview

The OV683 is a camera synchronization bridge chip for multiple video streams, which integrate two image signal processors (ISPs), MIPI receiver and transmitter with a built-in 32-bit microcontroller (MCU). The OV683 provides the ability to process 1, 2 or 3 sensor image streams simultaneously. Lower resolution sensors (e.g., <5 Mpixel) are supported by the dual two-lane MIPI receivers. The OV683 supports 8-bit/10-bit RAW input data format and 8-bit/10-bit RAW and YUV422 output data formats. Higher resolution sensors can bypass the RAW image through the four-lane MIPI receiver. Only 8-bit/10-bit/12-bit RAW formats are supported. All required image processing functions, including exposure control, white balance, defective pixel calibration, etc., are programmable through the SCCB slave interface.

figure 2-1 system block diagram



2.2 architecture

The OV683 includes the following main functional blocks:

- MIPI receiver
- image stitching
- image signal processor (ISP)
- multiple SCCB master interfaces
- SCCB master/slave interface
- microcontroller (MCU)
- system controller
- SRAMs and BIST
- regulator and PLL
- GPIO and I/O function multiplexing

figure 2-2 reference design schematic

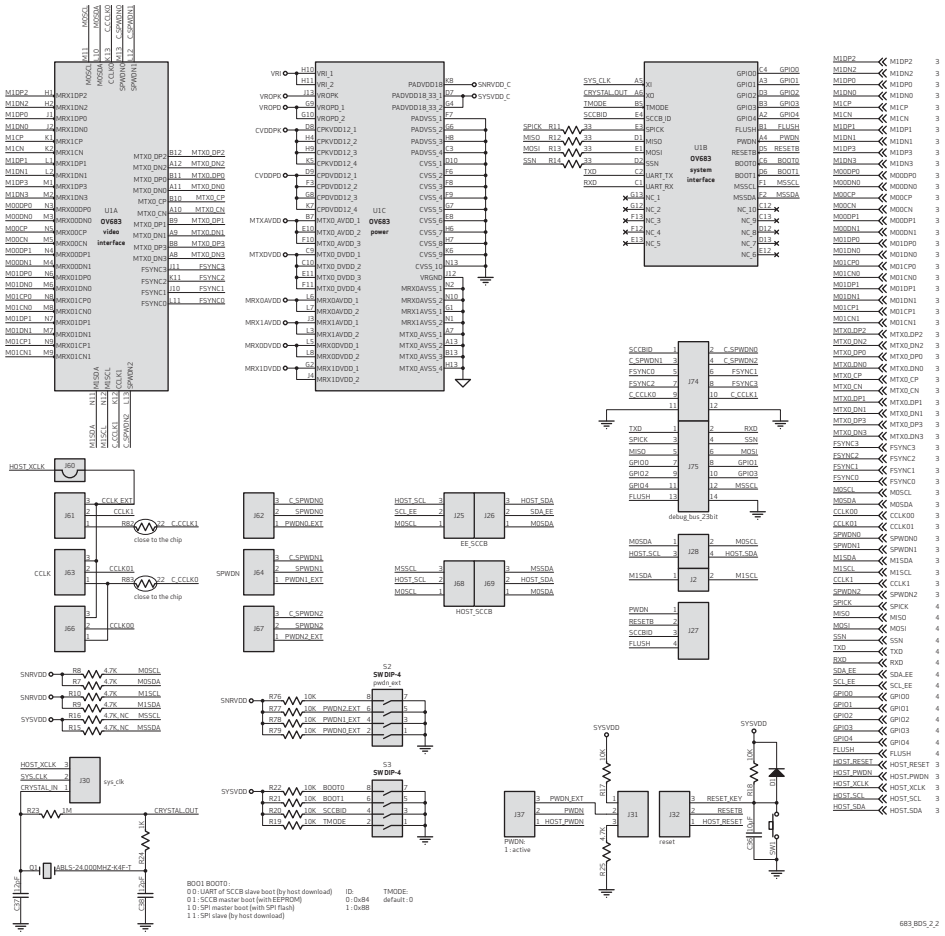


figure 2-3 MIPI schematic

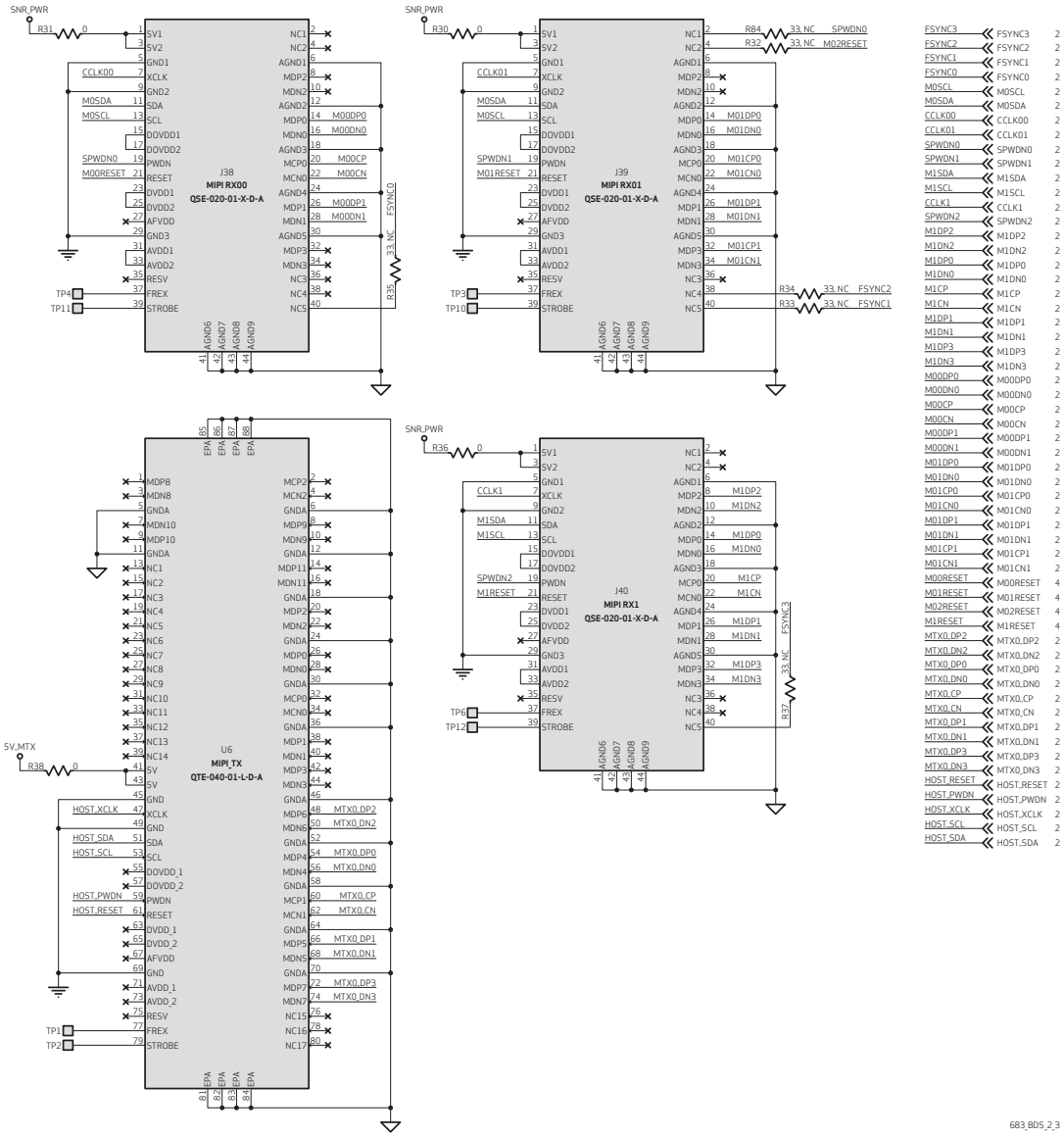


figure 2-4 SPI UART schematic

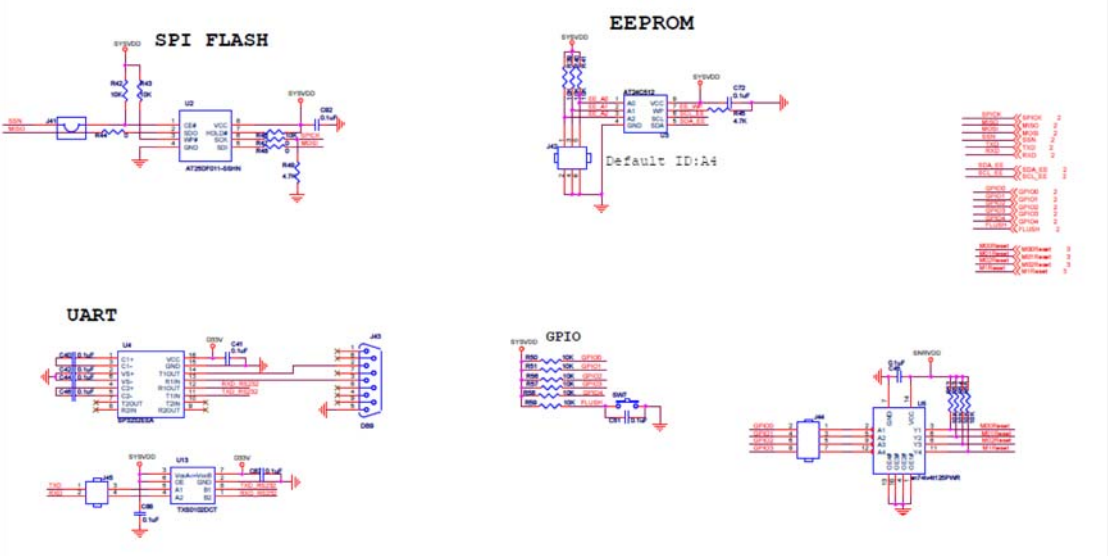


figure 2-5 power schematic

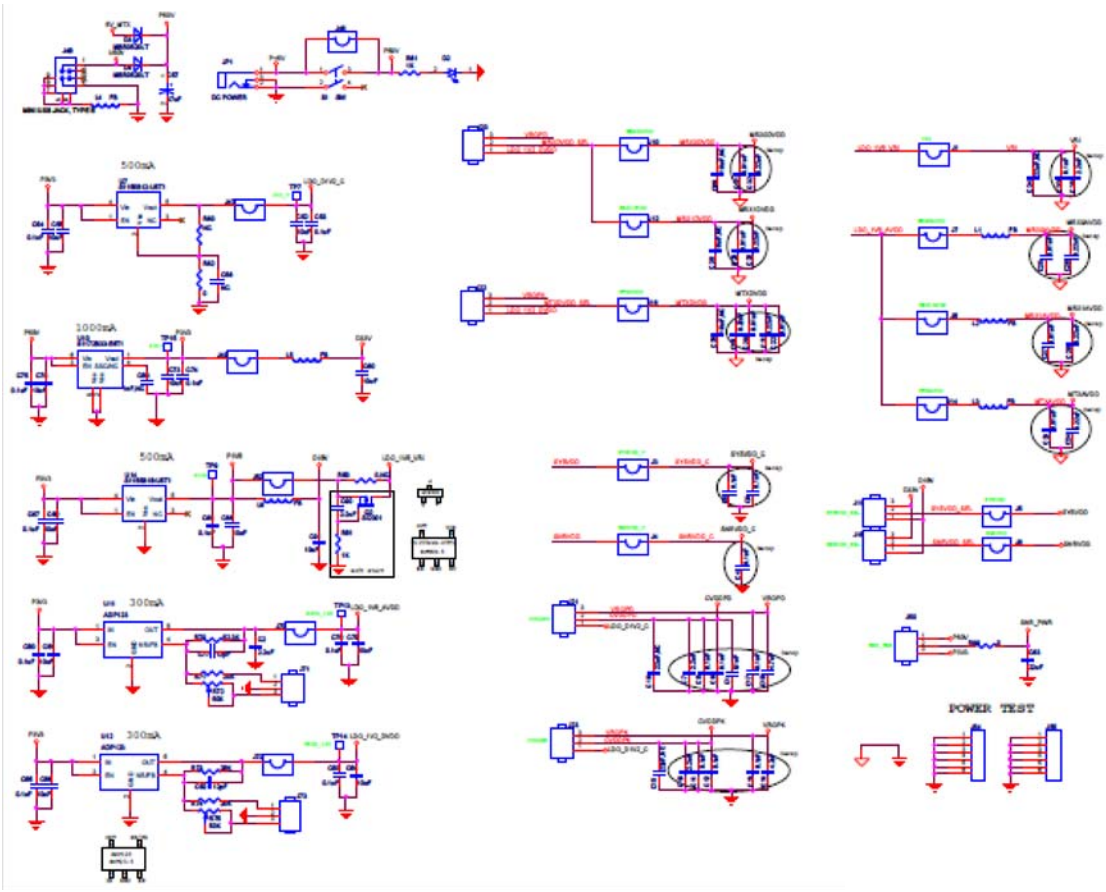
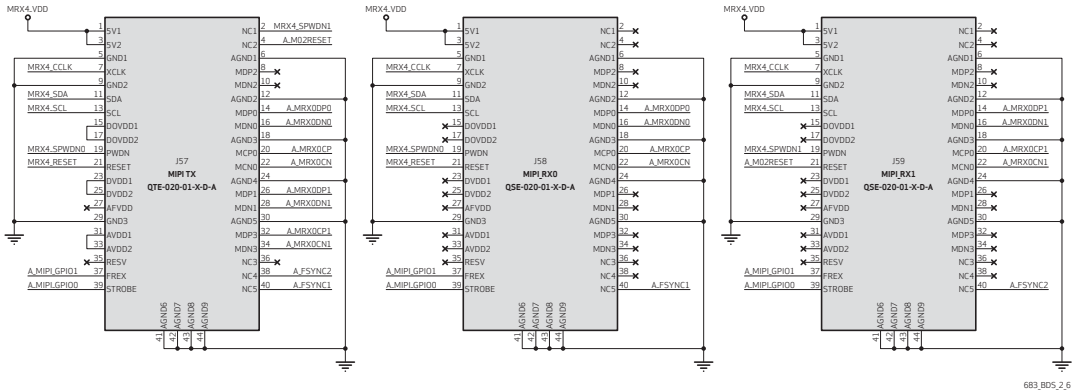


figure 2-6 MRX1lane MRX2 lane schematic



2.3 input format

The OV683 supports independent sensor inputs. The frame rate can either be independent for each sensor, or synchronized together. Here are some of the common input formats supported by the system:

table 2-1 typical 2-lane RX input formats

size	resolution	max frame rate	format
5 Mpixel	2592x1944	30 fps	RAW8/RAW10
4 Mpixel	2688x1520	30 fps	RAW8/RAW10
1080p	1920x1080	60 fps	RAW8/RAW10
720p	1280x800	120 fps	RAW8/RAW10
VGA	640x480	120 fps	RAW8/RAW10

table 2-2 typical 4-lane RX input formats

size	resolution	max frame rate	format
23 Mpixel	5592x4194	15 fps	RAW8/RAW10/RAW12
16 Mpixel	5408x3042	24 fps	RAW8/RAW10/RAW12
13 Mpixel	4224x3136	30 fps	RAW8/RAW10/RAW12
8 Mpixel	3264x2448	30 fps	RAW8/RAW10/RAW12
5 Mpixel	2592x1944	30 fps	RAW8/RAW10/RAW12

2.4 output format

table 2-3 typical side-by-side stream output formats

size	resolution	max frame rate	format
SBS 5 Mpixel	5184x1944	30 fps	RAW8/RAW10/YUV422
SBS 4 Mpixel	5376x1520	30 fps	RAW8/RAW10/YUV422
SBS 1080p	3840x1080	60 fps	RAW8/RAW10/YUV422
SBS 720p	2560x800	120 fps	RAW8/RAW10/YUV422
SBS VGA	1280x480	120 fps	RAW8/RAW10/YUV422

table 2-4 typical 3 streams by virtual channel output stream

size	resolution	max frame rate	format
SBS 1080p 8 Mpixel RAW bypass	3840x1080 3264x2448	30 fps	RAW8/RAW10/YUV422 RAW8/RAW10/RAW12
SBS 1080p 5 Mpixel RAW bypass	3840x1080 2592x1944	30 fps	RAW8/RAW10/YUV422 RAW8/RAW10/RAW12
SBS 720p 5 Mpixel RAW bypass	2560x800 2592x1944	30 fps	RAW8/RAW10/YUV422 RAW8/RAW10/RAW12

2.5 multiple stream synchronization

The OV683 supports frame rate control on each sensor stream by frame rate control (e.g., FSYNC control pin). Applications can choose to use one unified FSYNC or a dedicated control pin on each sensor. Each FSYNC control pin provides timing adjustment based on the internal clock.

Another synchronization feature is embedded line. Each sensor stream passing through the OV683 will attach an embedded data line at the end of each frame. Inside the embedded line, several timestamp fields are provided for reference. There is also a sync_valid field to indicate which current streams are synchronized with the same start timing.

2.6 PLL and clock generator

The chip has two phase locked loop (PLL) blocks, which generate all the necessary internal clocks from an external clock input. In PLL1 mode, the MIPI clock is synced with the system clock. MIPI and the system core can run at different frequencies while in PLL2 mode.

2.7 spread spectrum clocking (SSC)

Spread Spectrum Clocking (sometimes referred to as "Spectrum Spread Clocking") is a common technique where a low frequency modulation is added to the transmitter's clock to reduce the peak emissions.

All DPHY 2.0 compliant transmitters supports SSC (see **table 2-5**). SSC shall be implemented within the transmitter such that the transmitter's high-speed clock lane and all high-speed data lanes have the same modulated profile, rate and SSC deviation.

It is required for the external reference clock to be >20MHz for OV683 to support D-PHY2.0 compliant SSC function with the required SSC deviation range and modulation rate.

table 2-5 spread spectrum clocking requirements[abcd]

parameter	symbol	min	max	units	notes
modulation rate	$T_{SSC\_MOD\_RATE}$	30	33	kHz	
SSC deviation	$T_{SSC\_FREQ\_DEV}$	-5000	0	PPM	1, 2
SSC df/dt	SSCdf/dt	–	1250	PPM/µS	3, 4

a. required SSC deviation is also called "Down-Spread"

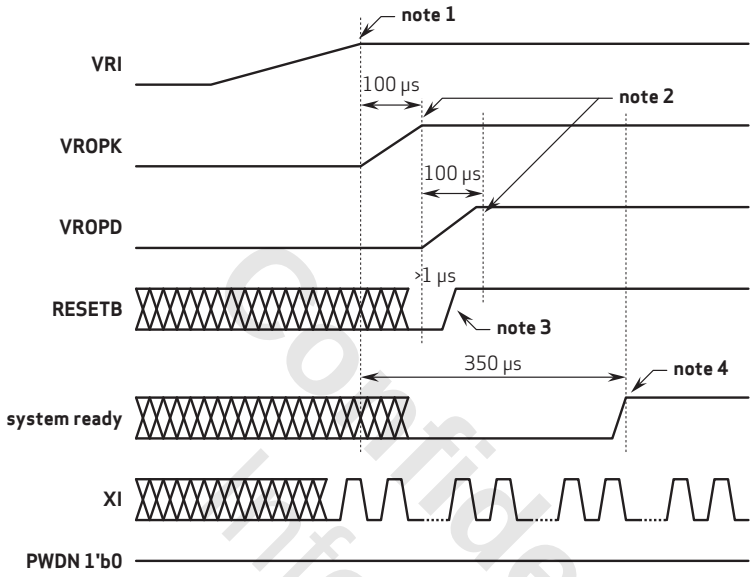
b. any implementation with an SSC deviation significantly smaller than 5000 PPM might be challenged to pass EMI testing below 1 GHz clock rate (data rate < 2 Gbps)

c. df/dt limit shall be for clock and all data lanes

d. measured over a 0.5 µs interval using an alternating 010101010… input pattern at highest data rate. Measurements shall be low pass filtered using a filter with 3 dB cutoff frequency that is 60 times the modulation rate. The filter stopband rejection shall be greater or equal to a second order low-pass of 20 dB per decade. Evaluation of the maximum df/dt is achieved by inspection of the low-pass filtered waveform.

2.8 power up/suspend sequence

figure 2-7 power on sequence



note 1 after board power on, power to VRI, PK regulator starts

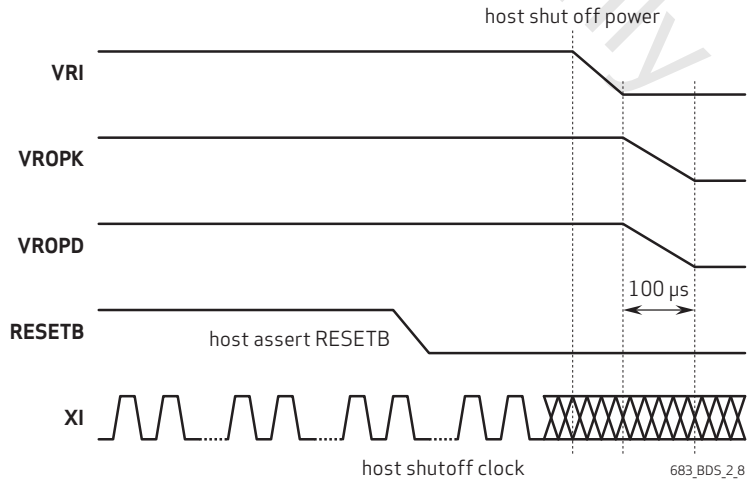
note 2 PK and PD regulator stable time: 100 µs

note 3 de-assert RESETB at least 1 µs after VROPK power stable

note 4 system is running

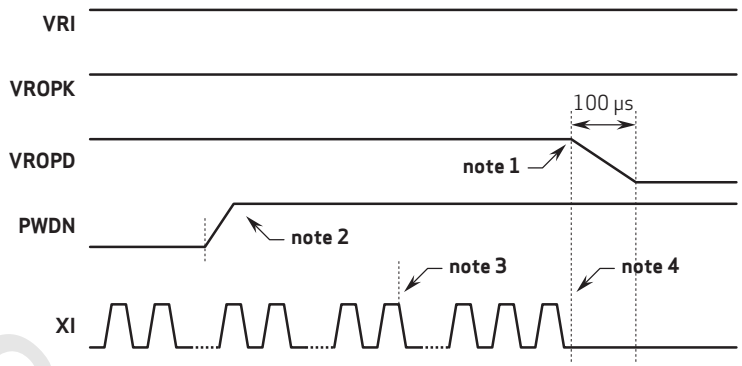
683_BDS_2_7

figure 2-8 power off sequence



683_BDS_2_8

figure 2-9 power down suspend sequence



note 1 internal regulator power down

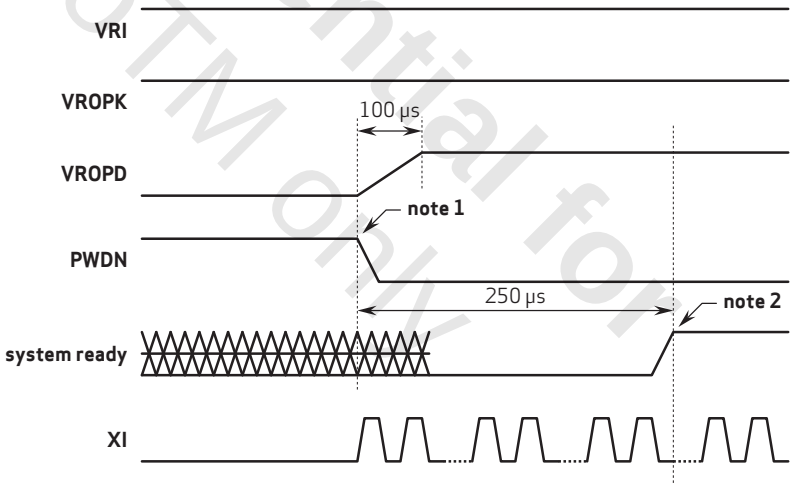
note 2 power down firmware sequence

note 3 after firmware has processed interrupt operation with power down, initiate hardware sequence

note 4 16x XI clock cycles +10 µs later, initiate power down of internal PD regulator, XI clock is gated internally

683_BDS_2_9

figure 2-10 power resume sequence



note 1 system resume request: deassert PWDN

note 2 system is running

683_BDS_2_10

2.9 test pattern

The OV683 supports solid color, 100% color bars, fade to gray color bars, and PN9 test mode. It also can support generating test patterns with sensor input timing.

3 core digital functions

3.1 MIPI receiver

The OV683 has three MIPI receivers, two 2-lane and one 4-lane receiver. One of the 2-lane receivers can be divided into two 1-lane receivers. Each interface contains one clock lane and one data lane. The maximum data rate of each data lane is 1.2Gbps/lane. This interface can support 10-bit/8-bit RAW format.

3.2 ISP

The ISP module provides lens correction, gamma, de-noise etc. These functions can be enabled by internal registers.

The main purpose of the LENC is to compensate for lens imperfection. According to the distance of each pixel to the lens center, LENC applies the gain to each pixel to compensate for lens fall off. The LENC correcting gain is also adaptive to sensor gain.

3.2.1 auto white balance (AWB)

The main function of AWB is the process of removing unrealistic color casts so that objects which appear white in person are rendered white in the video image. Thus, the AWB makes sure that the white color is always a white color in different color temperatures. It supports manual white balance and auto white balance. For auto mode, simple AWB and advanced AWB methods are supplied. Advanced AWB takes into account the color temperature of a light source.

3.2.2 color interpolation (CIP)

The CIP function is RAW to RGB interpolation. In sensor RAW format, each pixel will be either R,G or B. CIP will calculate the other two color values using the neighboring pixel of the same color. Thus, the full RGB information for each pixel can be obtained.

3.2.3 color matrix (CMX)

The main purpose of the CMX function is to cancel out crosstalk and convert color space.

3.2.4 gamma

Gamma function compensates image for the non-linear characteristics of human visual perception. The OV683 supports gamma function in YUV domain and RGB domain.

3.2.5 special digital effects (SDE)

The SDE functions include hue/saturation control, brightness, contrast, etc. SDE also supports negative, black/white, sepia, greenish. blueish, reddish, solarize and other image effects.

3.2.6 defect pixel cancellation (DPC)

The main purpose of DPC function is to remove white/black pixel defects. DPC algorithms only do white/black pixel cancellation in the horizontal and vertical directions. This function uses filters to determine which pixel is white/black and does this based on the channel.

3.2.7 scaling

The main purpose of the scaling function is to zoom out of the image. According to the new width and new height of the new image, the module uses the values of several pixels to generate values of one pixel. The values of some pixels are divided and used in two or more adjacent pixels.

3.2.8 crop

The main purpose of the crop function is to cut off the selectable size from the original image.

3.2.9 side-by-side function

The OV683 provides a side-by-side output stream. Two input streams from different sensors will be combined together and output as one stream. After the combination, one image will in the left and the other will in the right. A programmable dummy area can be inserted in the middle of the two images. Please note that when this function is applied, the two sensors must be in frame-sync to ensure the synchronization of frame.

3.2.10 MIPI transmitter interface

The OV683 provides a 4-lane MIPI transmitter, which contains one clock lane and up to four data lanes with a maximum data rate of 1.5Gbps/lane. The MIPI transmitter can run at various MIPI clock frequencies to match with the host MIPI receiver. It supports single, dual, three and four video channel data transfer simultaneously with virtual channel control implemented.

3.2.11 SCCB interface

There are two sets of SCCB master and one set of master/slave. The master SCCB interfaces are dedicated to configuring digital sensors and obtaining parameters from each sensor on-the-fly so as to dynamically adjust its performance parameters, while the slave SCCB allows users to read and write to its internal registers to control the chip or to use for debug purposes. Maximum supported speed is 1MHz.

3.2.12 embedded processor

The OV683 provides an embedded 32-bit microcontroller (MCU) architecture with high performance, great flexibility and scalability. There is an internal 64KB programmable memory for MCU.

3.2.13 general purpose input and output (GPIO)

The OV683 has five general purpose input and output (GPIO0/GPIO1/GPIO2/GPIO3/GPIO4) pins. All GPIO pins can be set to output with logic low, high, or high impedance (input mode) by writing to the appropriate registers via the SCCB. By default, all GPIO pins are set up as inputs.

4 image processor digital functions

4.1 image signal processor

The OV683 has the following ISP functions:

- ISP_top
- ISP color bar
- lens correction (LENC)
- auto white balance (AWB)
- defective pixel cancellation (DPC)
- RAW gamma
- de-noise (DNS)
- RGB DNS
- color interpolation (CIP)
- color matrix (CMX)
- RGB gamma
- UV average
- special digital effect (SDE)
- windowing
- YAVG

4.2 ISP_top

table 4-1 ISP_top registers (sheet 1 of 2)

offset	register name	default value	R/W	description
0x00	ISP EN LOW	0xFF	RW	Bit[6]: Stretch enable (active high) Bit[5]: RAW gamma enable (active high) Bit[4]: BC enable (active high) Bit[3]: WC enable (active high) Bit[2]: AWB gain enable (active high) Bit[1]: AWB enable (active high) Bit[0]: LENC enable (active high)
0x01	ISP EN MID	0xFF	RW	Bit[7]: Pad enable (active high) Bit[6]: rgb_dns enable (active high) Bit[5]: rgb2yuv enable (active high) Bit[4]: Gamma enable (active high) Bit[3]: CMX enable (active high) Bit[2]: CTRL enable (active high) Bit[1]: CIP enable (active high) Bit[0]: DNS enable (active high)

table 4-1 ISP_top registers (sheet 2 of 2)

offset	register name	default value	R/W	description
0x02	ISP EN HIGH	0xFF	RW	Bit[7]: EDR enable (active high) Bit[6]: UVDNS enable (active high) Bit[5]: YUV422to444 enable (active high) Bit[4]: YAVG enable (active high) Bit[3]: AFC enable (active high) Bit[2]: Raw WINC enable (active high) Bit[1]: YUV WINC enable (active high) Bit[0]: SDE enable (active high)
0x03	ISP FORMAT1	0x02	RW	Bit[6]: Add padding for input raw data Bit[5]: Top output follow YUV output Bit[4]: Top output bypass raw input Bit[3]: YUV output bypass YUV Input Bit[2]: RAW12 input (active high) Bit[1]: RAW10 input (active high) Bit[0]: RAW 8 input (active high)
0x04	ISP FORMAT2	0x00	RW	Bit[4]: RAW10 output Bit[3]: RAW8 output Bit[2]: afc_sel Bit[1:0]: avg_sel
0x05	ISP CTRL LOW	0x60	RW	Bit[6]: Size auto 0: Use top register 0x14~0x16 as ISP input size 1: Use pre_isp output size as ISP input size Bit[5:4]: px_order Bit[3]: LENC flip Bit[2]: LENC mirror Bit[1]: Flip Bit[0]: Mirror
0x06	ISP CTRL PB	0x02	RW	Bit[7]: ISP YUV bypass SOF select Bit[6]: ISP YUV SOF select Bit[5]: ISP top SOF select Bit[4]: rgb_dns line memory enable Bit[3]: CIP and DNS line memory enable Bit[2]: DPC line memory enable Bit[1:0]: shift_i 00: Processing 10 bit raw 01: Processing 11 bit raw 10: Processing 12 bit raw
0x12	X Input Size Low	0x80	RW	Bit[7:0]: x_input_size[7:0]
0x13	X Input Size High	0x02	RW	Bit[3:0]: x_input_size[11:8]
0x14	Y Input Size Low	0xE0	RW	Bit[7:0]: y_input_size[7:0]
0x15	Y Input Size High	0x01	RW	Bit[2:0]: y_input_size[10:8]

4.3 ISP color bar

table 4-2 ISP color bar registers

offset	register name	default value	R/W	description
0x00	ISP COLOR BAR	0x00	RW	Bit[7]: Test mode enable 0: Disable 1: Enable Bit[6]: Image rolling mode enable when test mode is enabled 0: Disable 1: Enable Bit[5]: Test data when test mode is enabled 0: Test data 1: Test data add image data Bit[4]: Black/white or color square 0: Color 1: Black/white Bit[3:2]: Color bar style when test mode is enabled 00: Standard color bar 01: Color bar become darker from top to bottom 10: Color bar become darker from right to left 11: Color bar become darker from bottom to top Bit[1:0]: Test mode select when test mode is enabled 00: Color bar, different options for color bar 01: Random data, different seed for random data 10: Square pattern, black white or color square pattern 11: Color chart / all black

4.4 LENC

table 4-3 LENC registers (sheet 1 of 4)

offset	register name	default value	R/W	description
0x00	LENC_G00	0x10	RW	(Row 0, Column 0) G Channel Parameter for Gain Computation
0x01	LENC_G01	0x10	RW	(Row 0, Column 1) G Channel Parameter for Gain Computation
0x02	LENC_G02	0x10	RW	(Row 0, Column 2) G Channel Parameter for Gain Computation
0x03	LENC_G03	0x10	RW	(Row 0, Column 3) G Channel Parameter for Gain Computation
0x04	LENC_G04	0x10	RW	(Row 0, Column 4) G Channel Parameter for Gain Computation
0x05	LENC_G05	0x10	RW	(Row 0, Column 5) G Channel Parameter for Gain Computation
0x06	LENC_G10	0x10	RW	(Row 1, Column 0) G Channel Parameter for Gain Computation
0x07	LENC_G11	0x08	RW	(Row 1, Column 1) G Channel Parameter for Gain Computation
0x08	LENC_G12	0x08	RW	(Row 1, Column 2) G Channel Parameter for Gain Computation
0x09	LENC_G13	0x08	RW	(Row 1, Column 3) G Channel Parameter for Gain Computation
0x0A	LENC_G14	0x08	RW	(Row 1, Column 4) G Channel Parameter for Gain Computation
0x0B	LENC_G15	0x10	RW	(Row 1, Column 5) G Channel Parameter for Gain Computation
0x0C	LENC_G20	0x10	RW	(Row 2, Column 0) G Channel Parameter for Gain Computation
0x0D	LENC_G21	0x08	RW	(Row 2, Column 1) G Channel Parameter for Gain Computation
0x0E	LENC_G22	0x00	RW	(Row 2, Column 2) G Channel Parameter for Gain Computation
0x0F	LENC_G23	0x00	RW	(Row 2, Column 3) G Channel Parameter for Gain Computation
0x10	LENC_G24	0x08	RW	(Row 2, Column 4) G Channel Parameter for Gain Computation

table 4-3 LENC registers (sheet 2 of 4)

offset	register name	default value	R/W	description
0x11	LENC_G25	0x10	RW	(Row 2, Column 5) G Channel Parameter for Gain Computation
0x12	LENC_G30	0x10	RW	(Row 3, Column 0) G Channel Parameter for Gain Computation
0x13	LENC_G31	0x08	RW	(Row 3, Column 1) G Channel Parameter for Gain Computation
0x14	LENC_G32	0x00	RW	(Row 3, Column 2) G Channel Parameter for Gain Computation
0x15	LENC_G33	0x00	RW	(Row 3, Column 3) G Channel Parameter for Gain Computation
0x16	LENC_G34	0x08	RW	(Row 3, Column 4) G Channel Parameter for Gain Computation
0x17	LENC_G35	0x10	RW	(Row 3, Column 5) G Channel Parameter for Gain Computation
0x18	LENC_G40	0x10	RW	(Row 4, Column 0) G Channel Parameter for Gain Computation
0x19	LENC_G41	0x08	RW	(Row 4, Column 1) G Channel Parameter for Gain Computation
0x1A	LENC_G42	0x08	RW	(Row 4, Column 2) G Channel Parameter for Gain Computation
0x1B	LENC_G43	0x08	RW	(Row 4, Column 3) G Channel Parameter for Gain Computation
0x1C	LENC_G44	0x08	RW	(Row 4, Column 4) G Channel Parameter for Gain Computation
0x1D	LENC_G45	0x10	RW	(Row 4, Column 5) G Channel Parameter for Gain Computation
0x1E	LENC_G50	0x10	RW	(Row 5, Column 0) G Channel Parameter for Gain Computation
0x1F	LENC_G51	0x10	RW	(Row 5, Column 1) G Channel Parameter for Gain Computation
0x20	LENC_G52	0x10	RW	(Row 5, Column 2) G Channel Parameter for Gain Computation
0x21	LENC_G53	0x10	RW	(Row 5, Column 3) G Channel Parameter for Gain Computation
0x22	LENC_G54	0x10	RW	(Row 5, Column 4) G Channel Parameter for Gain Computation
0x23	LENC_G55	0x10	RW	(Row 5, Column 5) G Channel Parameter for Gain Computation

table 4-3 LENC registers (sheet 3 of 4)

offset	register name	default value	R/W	description
0x24	LENC_BR00	0xAA	RW	(Row 0, Column 0) B, R Channel Parameter for Gain Computation
0x25	LENC_BR01	0xAA	RW	(Row 0, Column 1) B, R Channel Parameter for Gain Computation
0x26	LENC_BR02	0xAA	RW	(Row 0, Column 2) B, R Channel Parameter for Gain Computation
0x27	LENC_BR03	0xAA	RW	(Row 0, Column 3) B, R Channel Parameter for Gain Computation
0x28	LENC_BR04	0xAA	RW	(Row 0, Column 4) B, R Channel Parameter for Gain Computation
0x29	LENC_BR10	0xAA	RW	(Row 1, Column 0) B, R Channel Parameter for Gain Computation
0x2A	LENC_BR11	0x99	RW	(Row 1, Column 1) B, R Channel Parameter for Gain Computation
0x2B	LENC_BR12	0x99	RW	(Row 1, Column 2) B, R Channel Parameter for Gain Computation
0x2C	LENC_BR13	0x99	RW	(Row 1, Column 3) B, R Channel Parameter for Gain Computation
0x2D	LENC_BR14	0xAA	RW	(Row 1, Column 4) B, R Channel Parameter for Gain Computation
0x2E	LENC_BR20	0xAA	RW	(Row 2, Column 0) B, R Channel Parameter for Gain Computation
0x2F	LENC_BR21	0x99	RW	(Row 2, Column 1) B, R Channel Parameter for Gain Computation
0x30	LENC_BR22	0x88	RW	(Row 2, Column 2) B, R Channel Parameter for Gain Computation
0x31	LENC_BR23	0x99	RW	(Row 2, Column 3) B, R Channel Parameter for Gain Computation
0x32	LENC_BR24	0xAA	RW	(Row 2, Column 4) B, R Channel Parameter for Gain Computation
0x33	LENC_BR30	0xAA	RW	(Row 3, Column 0) B, R Channel Parameter for Gain Computation
0x34	LENC_BR31	0x99	RW	(Row 3, Column 1) B, R Channel Parameter for Gain Computation
0x35	LENC_BR32	0x99	RW	(Row 3, Column 2) B, R Channel Parameter for Gain Computation
0x36	LENC_BR33	0x99	RW	(Row 3, Column 3) B, R Channel Parameter for Gain Computation

table 4-3 LENC registers (sheet 4 of 4)

offset	register name	default value	R/W	description
0x37	LENC_BR34	0xAA	RW	(Row 3, Column 4) B, R Channel Parameter for Gain Computation
0x38	LENC_BR40	0xAA	RW	(Row 4, Column 0) B, R Channel Parameter for Gain Computation
0x39	LENC_BR41	0xAA	RW	(Row 4, Column 1) B, R Channel Parameter for Gain Computation
0x3A	LENC_BR42	0xAA	RW	(Row 4, Column 2) B, R Channel Parameter for Gain Computation
0x3B	LENC_BR43	0xAA	RW	(Row 4, Column 3) B, R Channel Parameter for Gain Computation
0x3C	LENC_BR44	0xAA	RW	(Row 4, Column 4) B, R Channel Parameter for Gain Computation
0x3D	LENC_BR_OFFSET	0x88	RW	Offset for Each B, R Channel Parameter
0x3E	LENC_MAXGAIN	0x40	RW	Maximum Gain in Auto q Computation
0x3F	LENC_MINGAIN	0x20	RW	Minimum Gain in Auto q Computation
0x40	LENC_MINQ	0x18	RW	Minimum q in Auto q Computation
0x41	LENC_CTRL41	0x0D	RW	Bit[3]: Addblc Enable adding back BLC Bit[2]: blc_en Enable BLC Bit[0]: autoq_en Enable auto q computation
0x42	LENC_BR_HSCALE	0x00	RW	Horizontal Scaling Factor for B, R Channel
0x43	LENC_BR_HSCALE	0x99	RW	Horizontal Scaling Factor for B, R Channel
0x44	LENC_BR_VSCALE	0x00	RW	Vertical Scaling Factor for B, R Channel
0x45	LENC_BR_VSCALE	0xCC	RW	Vertical Scaling Factor for B, R Channel
0x46	LENC_G_HSCALE	0x00	RW	Horizontal Scaling Factor for G Channel
0x47	LENC_G_HSCALE	0xCC	RW	Horizontal Scaling Factor for G Channel
0x48	LENC_G_VSCALE	0x00	RW	Vertical Scaling Factor for B, R Channel
0x49	LENC_G_VSCALE	0x88	RW	Vertical Scaling Factor for B, R Channel

4.5 auto white balance (AWB)

table 4-4 AWB registers

offset	register name	default value	R/W	description
0x00	AWB_CTRL0	0xFF	RW	awb_b_block
0x01	AWB_CTRL1	0x58	RW	Bit[7:6]: step_local Bit[5:4]: step_fast Bit[3]: slop_8x Bit[2]: slop_4x Bit[1]: one_zone Bit[0]: avg_all
0x02	AWB_CTRL2	0x11	RW	Bit[7:4]: max_local_cnt Bit[3:0]: max_fast_cnt

4.6 defective pixel cancellation (DPC)

table 4-5 DPC register

offset	register name	default value	R/W	description
0x00	dpc_00	0x18	RW	Bit[7]: tail_en Enable removing the tail pattern Bit[6]: sat_en Enable the saturate Bit[5]: cluster_en Enable removing the cluster pattern Bit[4]: sc_en Enable same channel detection Bit[3]: dc_en Enable different channel detection Bit[2]: smooth_en Enable using average G values when doing recovery Bit[1]: bwsnr_en Enable bw sensor Bit[0]: man_mode_en Enable manual mode

4.7 de-noise (DNS)

table 4-6 DNS registers

offset	register name	default value	R/W	description
0x00	RAW_DNS_CTRL_00	0x11	RW	Bit[5:2]: raw_dns_y_slop Bit[1]: raw_dns manual mode 0: Control parameters auto mode 1: Control parameters manual mode
0x01	RAW_DNS_CTRL_01	0x04	RW	Bit[7:0]: Y noise list0
0x02	RAW_DNS_CTRL_02	0x08	RW	Bit[7:0]: Y noise list1
0x03	RAW_DNS_CTRL_03	0x10	RW	Bit[7:0]: Y noise list2
0x04	RAW_DNS_CTRL_04	0x18	RW	Bit[7:0]: Y noise list3
0x05	RAW_DNS_CTRL_05	0x20	RW	Bit[7:0]: Y noise list4
0x06	RAW_DNS_CTRL_06	0x30	RW	Bit[7:0]: Y noise list5
0x07	RAW_DNS_CTRL_07	0x40	RW	Bit[7:0]: Y noise list6
0x08	RAW_DNS_CTRL_08	0x40	RW	Bit[7:0]: Y noise list7
0x09	RAW_DNS_CTRL_09	0xFF	RW	Bit[7:0]: Max edgethre
0x0A	RAW_DNS_CTRL_10	0x08	RW	Bit[7:0]: RAW DNS noise parameter in manual mode
0x0B	RAW_DNS_CTRL_11	0x18	RW	Bit[7:0]: RAW DNS edge threshold parameter in manual mode

4.8 RGB DNS

table 4-7 RGB DNS registers

offset	register name	default value	R/W	description
0x00	RGB_DNS_CTRL00	0x0D	RW	Bit[5:1]: Shadow extra noise Bit[0]: Smooth y enable
0x01	RGB_DNS_CTRL01	0x08	RW	Bit[3:0]: RGB DNS edgethre
0x02	RGB_DNS_CTRL02	0x08	RW	Bit[7:4]: Y noise list1 Bit[3:0]: Y noise list0
0x03	RGB_DNS_CTRL03	0x1A	RW	Bit[7:4]: Y noise list3 Bit[3:0]: Y noise list2
0x04	RGB_DNS_CTRL04	0x24	RW	Bit[7:4]: Y noise list5 Bit[3:0]: Y noise list4
0x05	RGB_DNS_CTRL05	0x24	RW	Bit[7:4]: Y noise list7 Bit[3:0]: Y noise list6
0x06	RGB_DNS_CTRL06	0x02	RW	Bit[5:0]: UV noise list0
0x07	RGB_DNS_CTRL07	0x04	RW	Bit[5:0]: UV noise list1
0x08	RGB_DNS_CTRL08	0x06	RW	Bit[5:0]: UV noise list2
0x09	RGB_DNS_CTRL09	0x08	RW	Bit[5:0]: UV noise list3
0x0A	RGB_DNS_CTRL0A	0x0A	RW	Bit[5:0]: UV noise list4
0x0B	RGB_DNS_CTRL0B	0x0A	RW	Bit[5:0]: UV noise list5
0x0C	RGB_DNS_CTRL0C	0x0A	RW	Bit[5:0]: UV noise list6
0x0D	RGB_DNS_CTRL0D	0x0A	RW	Bit[5:0]: UV noise list7
0x0E	RGB_DNS_CTRL0E	0x02	RW	Bit[3:1]: Y noise manual value Bit[0]: RGB DNS manual mode
0x0F	RGB_DNS_CTRL0F	0x04	RW	Bit[5:0]: UV noise manual value
0x10	RGB_DNS_CTRL10	–	R	Y Noise Value Read Only Register
0x11	RGB_DNS_CTRL11	–	R	UV Noise Value Read Only Register

4.9 color interpolation (CIP)

table 4-8 CIP register

offset	register name	default value	R/W	description
0x01	ISP EN MID	1'b1	RW	Bit[7]: CIP enable (active high)

4.10 color matrix (CMX)

table 4-9 CMX registers (sheet 1 of 3)

offset	register name	default value	R/W	description
0x00	CMX_CTRL_00	0x00	RW	Bit[3:0]: cmx_d00[11:8]
0x01	CMX_CTRL_01	0x00	RW	Bit[7:0]: cmx_d00[7:0]
0x02	CMX_CTRL_02	0x00	RW	Bit[3:0]: cmx_d01[11:8]
0x03	CMX_CTRL_03	0x00	RW	Bit[7:0]: cmx_d01[7:0]
0x04	CMX_CTRL_04	0x00	RW	Bit[3:0]: cmx_d10[11:8]
0x05	CMX_CTRL_05	0x00	RW	Bit[7:0]: cmx_d10[7:0]
0x06	CMX_CTRL_06	0x00	RW	Bit[3:0]: cmx_d11[11:8]
0x07	CMX_CTRL_07	0x00	RW	Bit[7:0]: cmx_d11[7:0]
0x08	CMX_CTRL_08	0x00	RW	Bit[3:0]: cmx_d20[11:8]
0x09	CMX_CTRL_09	0x00	RW	Bit[7:0]: cmx_d20[7:0]
0x0A	CMX_CTRL_0A	0x00	RW	Bit[3:0]: cmx_d21[11:8]
0x0B	CMX_CTRL_0B	0x00	RW	Bit[7:0]: cmx_d21[7:0]
0x0C	CMX_CTRL_0C	0x00	RW	Bit[3:0]: cmx_a00[11:8]
0x0D	CMX_CTRL_0D	0x00	RW	Bit[7:0]: cmx_a00[7:0]
0x0E	CMX_CTRL_0E	0x00	RW	Bit[3:0]: cmx_a01[11:8]
0x0F	CMX_CTRL_0F	0x00	RW	Bit[7:0]: cmx_a01[7:0]
0x10	CMX_CTRL_10	0x00	RW	Bit[3:0]: cmx_a10[11:8]
0x11	CMX_CTRL_11	0x00	RW	Bit[7:0]: cmx_a10[7:0]
0x12	CMX_CTRL_12	0x00	RW	Bit[3:0]: cmx_a11[11:8]
0x13	CMX_CTRL_13	0x00	RW	Bit[7:0]: cmx_a11[7:0]

table 4-9 CMX registers (sheet 2 of 3)

offset	register name	default value	R/W	description
0x14	CMX_CTRL_14	0x00	RW	Bit[3:0]: cmx_a20[11:8]
0x15	CMX_CTRL_15	0x00	RW	Bit[7:0]: cmx_a20[7:0]
0x16	CMX_CTRL_16	0x00	RW	Bit[3:0]: cmx_a21[11:8]
0x17	CMX_CTRL_17	0x00	RW	Bit[7:0]: cmx_a21[7:0]
0x18	CMX_CTRL_18	0x00	RW	Bit[3:0]: cmx_c00[11:8]
0x19	CMX_CTRL_19	0x00	RW	Bit[7:0]: cmx_c00[7:0]
0x1A	CMX_CTRL_1A	0x00	RW	Bit[3:0]: cmx_c01[11:8]
0x1B	CMX_CTRL_1B	0x00	RW	Bit[7:0]: cmx_c01[7:0]
0x1C	CMX_CTRL_1C	0x00	RW	Bit[3:0]: cmx_c10[11:8]
0x1D	CMX_CTRL_1D	0x00	RW	Bit[7:0]: cmx_c10[7:0]
0x1E	CMX_CTRL_1E	0x00	RW	Bit[3:0]: cmx_c11[11:8]
0x1F	CMX_CTRL_1F	0x00	RW	Bit[7:0]: cmx_c11[7:0]
0x20	CMX_CTRL_20	0x00	RW	Bit[3:0]: cmx_c20[11:8]
0x21	CMX_CTRL_21	0x00	RW	Bit[7:0]: cmx_c20[7:0]
0x22	CMX_CTRL_22	0x00	RW	Bit[3:0]: cmx_c21[11:8]
0x23	CMX_CTRL_23	0x00	RW	Bit[7:0]: cmx_c21[7:0]
0x24	CMX_CTRL_24	0x01	RW	Bit[4]: r_cmx_br_gain_man Bit[3:0]: r_cmx_awb_bgain[11:8]
0x25	CMX_CTRL_25	0x00	RW	Bit[7:0]: r_cmx_awb_bgain[7:0]
0x26	CMX_CTRL_26	0x01	RW	Bit[3:0]: r_cmx_awb_rgain[11:8]
0x27	CMX_CTRL_27	0x00	RW	Bit[7:0]: r_cmx_awb_rgain[7:0]
0x28	CMX_CTRL_28	0x00	RW	Bit[2:0]: r_cmx_ct_list0[9:8]
0x29	CMX_CTRL_29	0x80	RW	Bit[7:0]: r_cmx_ct_list0[7:0]
0x2A	CMX_CTRL_2A	0x00	RW	Bit[2:0]: r_cmx_ct_list1[9:8]
0x2B	CMX_CTRL_2B	0x80	RW	Bit[7:0]: r_cmx_ct_list1[7:0]
0x2C	CMX_CTRL_2C	0x00	RW	Bit[2:0]: r_cmx_ct_list2[9:8]
0x2D	CMX_CTRL_2D	0x80	RW	Bit[7:0]: r_cmx_ct_list2[7:0]

table 4-9 CMX registers (sheet 3 of 3)

offset	register name	default value	R/W	description
0x2F	CMX_CTRL_2F	0x00	RW	Bit[5]: Sign of cmx_a00 Bit[4]: Sign of cmx_a01 Bit[3]: Sign of cmx_a10 Bit[2]: Sign of cmx_a11 Bit[1]: Sign of cmx_a20 Bit[0]: Sign of cmx_a21
0x30	CMX_CTRL_30	0x00	RW	Bit[5]: Sign of cmx_c00 Bit[4]: Sign of cmx_c01 Bit[3]: Sign of cmx_c10 Bit[2]: Sign of cmx_c11 Bit[1]: Sign of cmx_c20 Bit[0]: Sign of cmx_c21
0x31	CMX_CTRL_31	0x01	RW	Bit[1]: r_cmx_auto_ct_en Bit[0]: r_cmx_auto_factor_en
0x32	CMX_CTRL_32	0x40	RW	Bit[6:0]: r_cmx_max_factor
0x33	CMX_CTRL_33	0x40	RW	Bit[6:0]: r_cmx_min_factor
0x34	CMX_CTRL_34	0x07	RW	Bit[4:0]: r_cmx_min_gain real_cmx_min_gain= (r_cmx_min_gain+1)×16
0x35	CMX_CTRL_35	0x0F	RW	Bit[4:0]: r_cmx_max_gain real_cmx_max_gain= (r_cmx_max_gain+1)×16

4.11 RGB gamma

table 4-10 RGB gamma registers (sheet 1 of 2)

offset	register name	default value	R/W	description
0x00	GAMMA_00	0x04	RW	Bit[7:0]: List00 for Y
0x01	GAMMA_01	0x08	RW	Bit[7:0]: List01 for Y
0x02	GAMMA_02	0x10	RW	Bit[7:0]: List02 for Y
0x03	GAMMA_03	0x20	RW	Bit[7:0]: List03 for Y
0x04	GAMMA_04	0x28	RW	Bit[7:0]: List04 for Y
0x05	GAMMA_05	0x30	RW	Bit[7:0]: List05 for Y
0x06	GAMMA_06	0x38	RW	Bit[7:0]: List06 for Y

table 4-10 RGB gamma registers (sheet 2 of 2)

offset	register name	default value	R/W	description
0x07	GAMMA_07	0x40	RW	Bit[7:0]: List07 for Y
0x08	GAMMA_08	0x48	RW	Bit[7:0]: List08 for Y
0x09	GAMMA_09	0x50	RW	Bit[7:0]: List09 for Y
0x0A	GAMMA_0A	0x60	RW	Bit[7:0]: List0a for Y
0x0B	GAMMA_0B	0x70	RW	Bit[7:0]: List0b for Y
0x0C	GAMMA_0C	0x90	RW	Bit[7:0]: List0c for Y
0x0D	GAMMA_0D	0xB0	RW	Bit[7:0]: List0d for Y
0x0E	GAMMA_0E	0xD0	RW	Bit[7:0]: List0e for Y
0x0F	GAMMA_0F	0xFF	RW	Bit[7:0]: gamma_max_shd_h_gain
0x10	GAMMA_10	0x01	RW	Bit[7:1]: Debug mode Bit[0]: gamma_man_en
0x40	GAMMA_40	0x0E	RW	Bit[7:0]: RISC read or write list00 for Y
0x41	GAMMA_41	0x1A	RW	Bit[7:0]: RISC read or write list01 for Y
0x42	GAMMA_42	0x31	RW	Bit[7:0]: RISC read or write list02 for Y
0x43	GAMMA_43	0x5A	RW	Bit[7:0]: RISC read or write list03 for Y
0x44	GAMMA_44	0x69	RW	Bit[7:0]: RISC read or write list04 for Y
0x45	GAMMA_45	0x75	RW	Bit[7:0]: RISC read or write list05 for Y
0x46	GAMMA_46	0x7E	RW	Bit[7:0]: RISC read or write list06 for Y
0x47	GAMMA_47	0x88	RW	Bit[7:0]: RISC read or write list07 for Y
0x48	GAMMA_48	0x8F	RW	Bit[7:0]: RISC read or write list08 for Y
0x49	GAMMA_49	0x96	RW	Bit[7:0]: RISC read or write list09 for Y
0x4A	GAMMA_4A	0xA3	RW	Bit[7:0]: RISC read or write list0a for Y
0x4B	GAMMA_4B	0xAF	RW	Bit[7:0]: RISC read or write list0b for Y
0x4C	GAMMA_4C	0xC4	RW	Bit[7:0]: RISC read or write list0c for Y
0x4D	GAMMA_4D	0xD7	RW	Bit[7:0]: RISC read or write list0d for Y
0x4E	GAMMA_4E	0xE8	RW	Bit[7:0]: RISC read or write list0e for Y

4.12 special digital effect (SDE)

table 4-11 SDE registers

offset	register name	default value	R/W	description
0x00	SDE_CTRL0	0x00	RW	Bit[7]: Fixy enable Bit[6]: Negative enable Bit[5]: Gray enable Bit[4]: fix_v enable Bit[3]: fix_u enable Bit[2]: Contrast enable Bit[1]: Saturation enable Bit[0]: Hue enable
0x01	SDE_CTRL1	0x80	RW	Bit[7:0]: hue_cos Cb' = cos(A)×(Cb-128)+sin(A)×(Cr-128)+128
0x02	SDE_CTRL2	0x00	RW	Bit[7:0]: hue_sin Cr' = cos(A)×(Cr-128)-sin(A)×(Cb-128)+128
0x03	SDE_CTRL3	0x40	RW	Bit[7:0]: sat_u or ureg Saturation coefficient for U
0x04	SDE_CTRL4	0x40	RW	Bit[7:0]: sat_v or vreg Saturation coefficient for V
0x05	SDE_CTRL5	0x00	RW	Bit[7:0]: Yoffset or fixy Offset = YOffset×(-1)^sign2
0x06	SDE_CTRL6	0x20	RW	Bit[7:0]: Ygain Y2 = (Y1+Bright-Offset)×YGain+Offset
0x07	SDE_CTRL7	0x00	RW	Bit[7:0]: Ybright Britht = YBright×(-1)^sign3
0x08	SDE_CTRL8	0x01	RW	Bit[7:6]: Debug mode Bit[5:0]: Signset
0x09	SDE_CTRL9	0x00	RW	Bit[7:0]: uvadj_th1 UVadjust threshold parameter
0x0A	SDE_CTRLA	0xFF	RW	Bit[7:0]: uvadj_th2 UVadjust threshold parameter
0x0B	SDE_CTRLB	0x00	RW	Bit[7:1]: Debug mode Bit[0]: uvadj_man_en UVadjust manual enable

4.13 16-zone luminance average (YAVG)

table 4-12 YAVG registers (sheet 1 of 2)

offset	register name	default value	R/W	description
0x00	AVG_XSTART	0x00	RW	Bit[4:0]: xstart[12:8] Start address in horizontal high byte
0x01	AVG_XSTART	0x00	RW	Bit[7:0]: xstart[7:0] Start address in horizontal low byte
0x02	AVG_YSTART	0x00	RW	Bit[3:0]: ystart[11:8] Start address in vertical high byte
0x03	AVG_YSTART	0x00	RW	Bit[7:0]: ystart[7:0] Start address in horizontal low byte
0x04	AVG_XSIZE	0x10	RW	Bit[4:0]: x_win[12:8] Select whole zone width high byte
0x05	AVG_XSIZE	0xA0	RW	Bit[7:0]: x_win[7:0] Select whole zone width low byte
0x06	AVG_YSIZE	0x0C	RW	Bit[3:0]: y_win[11:8] Select whole zone height high byte
0x07	AVG_YSIZE	0x78	RW	Bit[7:0]: y_win[7:0] Select whole zone height low byte
0x08	AVG_R8	0x11	RW	Bit[7:4]: wt1 Weight of zone01 Bit[3:0]: wt0 Weight of zone00
0x09	AVG_R9	0x11	RW	Bit[7:4]: wt3 Weight of zone03 Bit[3:0]: wt2 Weight of zone02
0x0A	AVG_RA	0x11	RW	Bit[7:4]: wt5 Weight of zone11 Bit[3:0]: wt4 Weight of zone10
0x0B	AVG_RB	0x11	RW	Bit[7:4]: wt7 Weight of zone13 Bit[3:0]: wt6 Weight of zone12
0x0C	AVG_RC	0x11	RW	Bit[7:4]: wt9 Weight of zone 21 Bit[3:0]: wt8 Weight of zone20

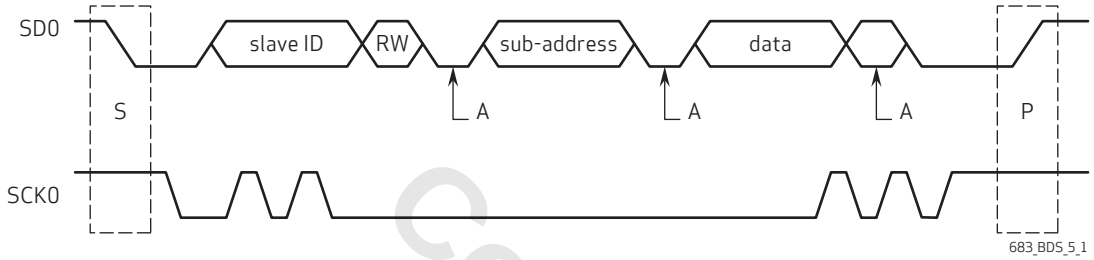
table 4-12 YAVG registers (sheet 2 of 2)

offset	register name	default value	R/W	description
0x0D	AVG_RD	0x11	RW	Bit[7:4]: wt11 Weight of zone23 Bit[3:0]: wt10 Weight of zone22
0x0E	AVG_RE	0x11	RW	Bit[7:4]: wt13 Weight of zone31 Bit[3:0]: wt12 Weight of zone30
0x0F	AVG_RF	0x11	RW	Bit[7:4]: wt15 Weight of zone33 Bit[3:0]: wt14 Weight of zone32

5 interface timing

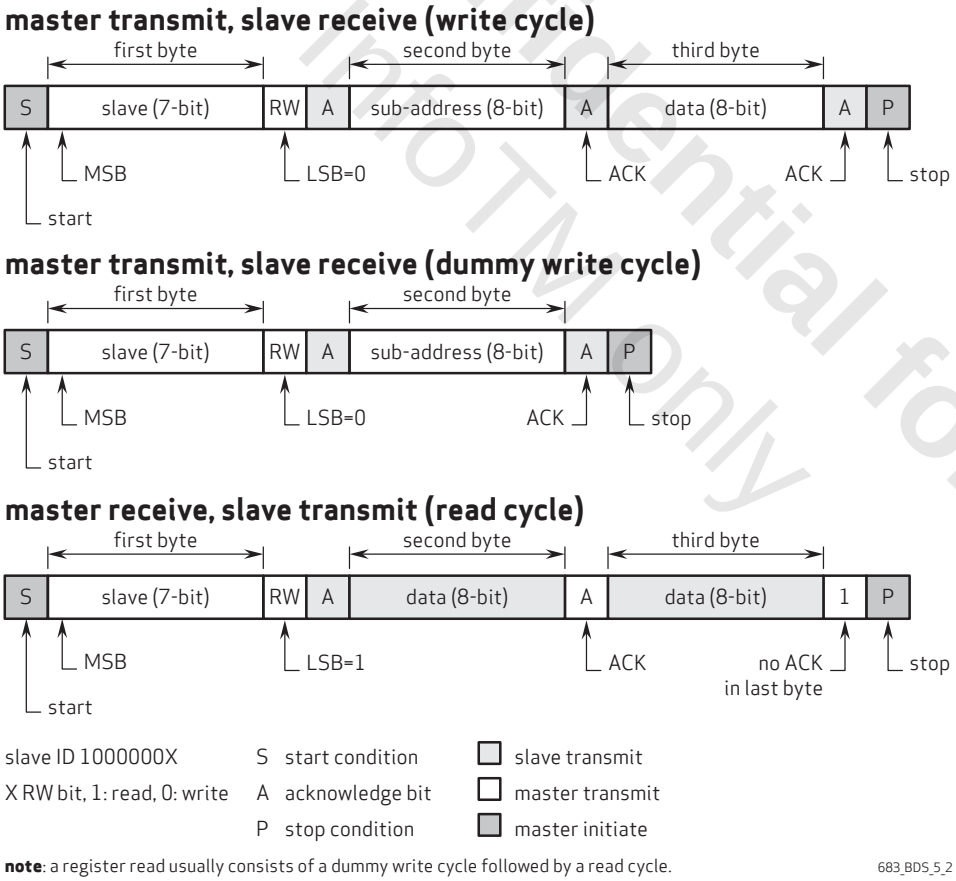
5.1 SCCB timing specification

figure 5-1 data transfer on the SCCB



note applies to both SD0/SCK0 and SD1/SCK1

figure 5-2 SCCB protocol format



note: a register read usually consists of a dummy write cycle followed by a read cycle.

683_BDS_5_2

figure 5-3 SCCB timing diagram

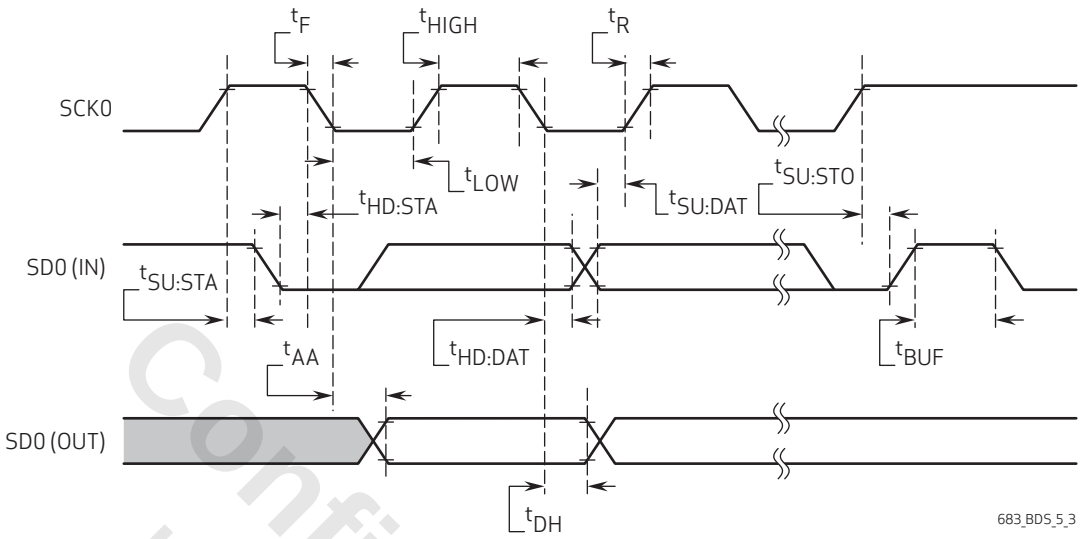


table 5-1 SCCB interface timing specifications

symbol	parameter	min	typ	max	unit
f_{SCK0}	clock frequency			1	MHz
t_{LOW}	clock low period	1.3			µs
t_{HIGH}	clock high period	600			ns
t_{AA}	SCK0 low to data out valid	100		900	ns
t_{BUF}	bus free time before new START	1.3			µs
$t_{HD:STA}$	START condition hold time	600			ns
$t_{SU:STA}$	START condition setup time	600			ns
$t_{HD:DAT}$	data in hold time	0			µs
$t_{SU:DAT}$	data in setup time	100			ns
$t_{SU:STO}$	STOP condition setup time	600			ns
t_R,t_F	SCCB rise/fall times			300	ns
t_{DH}	data out hold time	50			ns

6 register tables

6.1 system control

- base address: 0x80160200

table 6-1 system control registers (sheet 1 of 32)

address offset	register name	reset value	R/W	description
0x000	SYS_CTRL0	0x0402_1000	RW	Bit[31]: r_dis_switch When this bit is set, switch between PD regulator and PK regulator is disabled Bit[30]: r_pd_pwdn When this bit is set, PD regulator will be powered down Bit[29]: r_iso Software isolates signals from power down domain to power keep domain by setting this bit to 1'b1 Bit[28]: r_dis_osc Software disable oscillator by setting this bit to 1'b1 and xtai_i can be gated immediately Bit[27]: pwdnseq_en When software is ready to power down, setting this bit to 1'b1 can trigger hardware starting internal power down sequence Bit[26]: pwdn_func_en Enable power down PD domain power if this bit is set Bit[25]: osc_gate_en When powered down, external oscillator clock can be gated if this bit is set Bit[24]: pwdn_intr_en Enable power down interrupt, when this bit is set, hardware must generate an interrupt to software when hardware receives a power down sequence request, then waits for software to set "pwdnseq_en" to trigger a hardware power down sequence. It this bit is not set, hardware will perform power down sequence directly when it receives power down request.

table 6-1 system control registers (sheet 2 of 32)

address offset	register name	reset value	R/W	description
				Bit[23]: Reserved Bit[22]: mrx1_snr1_lane1 When this bit is set, data lane 0 of digital MIPI RX1 is fed with data lane 1 of sensor 1 (by default, data lane 0 of digital MIPI RX1 is fed by data lane 0 sensor 1, data lane 1 of sensor 1 will go to RX1's data lane 1). Bit[21]: mrx0_snr1_lane0 When this bit is set, data lane 0 of digital MIPI RX0 is fed by data lane 0 of sensor 1 (by default, digital MIPI RX0 is fed by sensor 0). Bit[20]: msnr_pwdn (active high) Main sensor power down Bit[19]: dp1snr_pwdn (active high) Depth sensor 1 power down Bit[18]: dp0snr_pwdn (active high) Depth sensor 0 power down Bit[17]: spi_new_slave_en Enable new SPI slave function If SPI I/O is used as a slave, this bit will determine which slave is working. 0: Old slave will be active 1: New stand-alone slave will be active Bit[16]: spi_master_en Enable SPI master function (default as a slave) Bit[15]: Reserved Bit[14]: fpga_pserom This bit will be used during FPGA ROM debug. If this bit is set, CPU will fetch run code from FPAG PSEUDO ROM. Bit[13]: fw_mem_sel Selects RISC program memory 0: ROM 1: PRAM Bit[12]: byte_order 0: Little endian (i.e., data[31:0] {byte3, byte2, byte1, byte0} 1: Big endian (i.e., data[31:0] {byte0, byte1, byte2, byte3}

table 6-1 system control registers (sheet 3 of 32)

address offset	register name	reset value	R/W	description
				Bit[11]: sccb_master_en Enable SCCB master function for SCCB master and slave (default as a slave) Bit[10]: sccb_master_en Enable SCCB master function for SCCB M1, default as a master Bit[9]: sccb_master_en Enable SCCB master function for SCCB M0, default as a master Bit[8:4]: Hardware clear threshold counter When counter reaches this value, hardware will clear "sw_cpu_rst" and "sw_bus_rst" signals Bit[3]: Reset bus Software set, hardware will delay some time then clear Bit[2]: Reset CPU Software set, hardware will delay some time then clear Bit[1]: System running/stop function enable If this bit is set, new system feature: system running/stop function is enabled. Bit[0]: System running/stop 0: System stop 1: System running if this bit is 1'b0, all function modules (except some modules, such as all SCCB MSs, SPI slave, AHB and CPU, SC) are reset and clocks are gating. So all configuration registers of sub-moduels must be belong to AHB bus clock domain to make sure SCCB slave/SPI slave/CPU can access them to perform read and write operations. When software configures all function blocks, it then sets this bit to 1 to take system live. If sys_ctrl0[1] is not set, this function is disabled.

table 6-1 system control registers (sheet 4 of 32)

address offset	register name	reset value	R/W	description
0x004	SYS_CTRL1	0x0000_0239	RW	Bit[31:28]: Reserved Bit[26]: cclk1_div_rst Sensor reference root clock 1 divider reset Bit[25]: cclk0_div_rst Sensor reference root clock 0 divider reset Bit[24]: syss_div_rst System (slow) root clock divider reset Bit[23]: sysf_div_rst System (fast) root clock divider reset Bit[22]: mtx1_div_rst Main sensor MIPI TX root clock divider reset Bit[21]: mtx0_div_rst Depth sensor MIPI TX root clock divider reset Bit[20]: msnr_div_rst Main sensor path root clock divider reset Bit[19]: dp1_div_rst Datapath 1 root clock divider reset Bit[18]: dp0_div_rst Datapath 0 root clock divider reset Bit[17]: mipi_looptest_src Select MIPI TX0/MIPI TX1 parallel output data as MIPI RX parallel input data Bit[16]: mipi_looptest_en Enable mipi loop test Bit[15]: Main sensor go through MIPI TX1 If this bit is set, main sensor stream will go to MIPI TX1 Bit[14]: msnr_isp_bypass Bypass main sensor path ISP Bit[13]: Reserved Bit[12]: dp1_isp_bypass Bypass datapath 1 ISP Bit[11]: Reserved Bit[10]: dp0_isp_bypass Bypass datapath 0 ISP

table 6-1 system control registers (sheet 5 of 32)

address offset	register name	reset value	R/W	description
				Bit[9:8]: msnr_isp_src_pixels Main sensor path ISP source pixel numbers 00: 1 pixels 01: 2 pixels 1x: 4 pixel Bit[7]: dp1_isp_src_two_pixel If this bit is set, dp1 ISP input data is two pixel valid; otherwise, only one pixel valid for ISP (one pixel is supported to 12-bit) Bit[6]: dp0_isp_src_two_pixel If this bit is set, dp0 ISP input data is two pixel valid; otherwise, only one pixel valid for ISP (one pixel is supported to 12-bit) Bit[5:4]: msnr_isp_src_sel Main sensor ISP source image selection 00: Colorbar image 01: Datapath 0 (depth sensor 0) image 10: Datapath 1 (depth sensor 1) image 11: Main sensor image Bit[3:2]: dp1_isp_src_sel Datapath 1 ISP source image selection 00: Colorbar image 01: Datapath 0 (depth sensor 0) image 10: Datapath 1 (depth sensor 1) image 11: Main sensor image Bit[1:0]: dp0_isp_src_sel Datapath 0 ISP source image selection 00: Colorbar image 01: Datapath 0 (depth sensor 0) image 10: Datapath 1 (depth sensor 1) image 11: Main sensor image

table 6-1 system control registers (sheet 6 of 32)

address offset	register name	reset value	R/W	description
0x008	SYS_CTRL2	0x0800_1401	RW	Bit[31]: Reserved Bit[30]: raw12 input enable Bit[29]: dpcm_10to6en 10-6 compress enable Bit[28]: dpcm_bypass1 DPCM encoder bypass mode 1, if inBit[1:0]>2, bypass_data = inBit[9:2]+1; else, bypass_data = {inBit[9:2], 2'b00} Bit[27]: dpcm_bypass0 DPCM encoder bypass mode 0, all bypass Bit[26]: addr8_mode 8-bit sub-address mode when host access sensor with SCCB bypass function Bit[25]: sccb_bypass_en1 Bypass to master 1 is enable Bit[24]: sccb_bypass_en0 Bypass to master 0 is enable Bit[23:20]: bypass_out_delay Output delay for SCCB I/O signal in SCCB bypass logic Bit[19:16]: bypass_in_delay Input delay for SCCB I/O signal in SCCB bypass logic Bit[15:14]: Reserved Bit[13:12]: TSEL Timing select Bit[11]: RTSEL Test mode for TURBO Bit[10]: TURBO Cycle time speed up Bit[9:7]: WTSEL Memory write cycle timing selection (default: 3'b000) Bit[6:5]: RTSEL Memory read cycle timing selection (default: 2'b00) Bit[4:2]: WTSEL Memory write cycle timing selection (default: 3'b000) Bit[1:0]: RTSEL Memory read cycle timing selection, default: 2'b01

table 6-1 system control registers (sheet 7 of 32)

address offset	register name	reset value	R/W	description
0x00C	SYS_CTRL3	0x0000_0000	RW	Bit[31:27]: Reserved Bit[26:24]: io_function These three bits indicate digital IO's function 000: Reserved 001: norm_funct, normal function 010: debug_23bit, debug bus output from 23 digital IOs 011: debug_7bit, debug bus output from 7 GPIOs (GPIO[4:0], UART_TX, UART_RX) 100: debug_gpio, all digital I/O serve as GPIO 101: Reserved 11x: Reserved If all digital I/Os serves as "normal function", all IO's have default direction and input enable flag. IO normal function: I/O Dir IN Enable SCCB_ID INPUT Enable SPWDN0 OUTPUT Disable SPWDN1 OUTPUT Disable SPWDN2 OUTPUT Disable FSYNC0 OUTPUT Disable FSYNC1 OUTPUT Disable FSYNC2 OUTPUT Disable FSYNC3 OUTPUT Disable CCLK0 OUTPUT Disable CCLK1 OUTPUT Disable UART_TX INPUT Disable UART_RX INPUT Disable SPICK(M) OUTPUT Disable SSN(M) OUTPUT Disable MOSI(M) OUTPUT Disable MISO(M) INPUT Enable SPICK(S) INPUT Enable SSN(S) INPUT Enable MOSI(S) INPUT Enable MISO(S) OUTPUT Disable GPIO0 INPUT Disable GPIO1 INPUT Disable GPIO2 INPUT Disable GPIO3 INPUT Disable GPIO4 INPUT Disable FLUSH INPUT Enable M0SCL OPEN DRAIN Enable

table 6-1 system control registers (sheet 8 of 32)

address offset	register name	reset value	R/W	description
				I/O Dir IN Enable M0SDA OPEN DRAIN Enable M1SCL OPEN DRAIN Enable M1SDA OPEN DRAIN Enable MSSCL OPEN DRAIN Enable MSSDA OPEN DRAIN Enable Bit[23]: Reserved Bit[22:21]: debug_sel_7bit 7-bit debug bus select 00: debug_bus_23[6:0] 01: debug_bus_23[13:7] 10: debug_bus_23[20:14] 11: debug_bus_23[22:21] Bit[20:16]: debug_sel_23bit 23-bit debug bus select 0x00: dp0_mipi_rx_debug 0x01: dp1_mipi_rx_debug 0x02: msnr_mipi_rx_debug 0x03: dp0_isp_debug 0x04: dp1_isp_debug 0x05: dp0_eof_debug 0x06: dp1_eof_debug 0x07: mipi_tx0_debug 0x08: mipi_tx1_debug 0x09: vfifo0_debug 0x0A: vfifo1_debug 0x0B: vfifo2_debug 0x0C: vcctrl0_debug 0x0D: vcctrl1_debug 0x0E: sccb_m0_debug 0x0F: sccb_m1_debug 0x10: sccb_ms_debug 0x11: spi_slv_debug 0x12: pram_debug[22:0] 0x13: {5'h00, sccb_bp_debug _i[7:0], pram_debug _i[32:23]} 0x14: con_switch_debug 0x15: mipi_rxphy_debug[22:0] 0x16: mipi_rxphy_debug[35:23] 0x17: dp0_embline_ debug[22:0] 0x18: dp1_embline_ debug[22:0] 0x19: msnr_embline_ debug[22:0] 0x1A: spi_ms_debug[22:0] 0x1B: msnr_eof_debug[22:0] 0x1C: sys_sus_debug[22:0] 0x1D: sys_rst_debug[22:0]

table 6-1 system control registers (sheet 9 of 32)

address offset	register name	reset value	R/W	description
				Bit[15]: sw_frm_sync3 Software controlled frame sync, this bit is set will assert FSYNC3 pin of chip until software clear this bit. Bit[14]: sw_frm_sync2 Software controlled frame sync, this bit is set will assert FSYNC2 pin of chip until software clear this bit. Bit[13]: sw_frm_sync1 Software controlled frame sync, this bit is set will assert FSYNC1 pin of chip until software clear this bit. Bit[12]: sw_frm_sync0 Software controlled frame sync, this bit is set will assert FSYNC0 pin of chip until software clear this bit. Bit[11]: frm_sync3_trig Trigger frame synchronizer 3 asserts frm_sync immediately, pulse width is controlled by internal register Bit[10]: frm_sync2_trig Trigger frame synchronizer 2 asserts frm_sync immediately, pulse width is controlled by internal register Bit[9]: frm_sync1_trig Trigger frame synchronizer 1 asserts frm_sync immediately, pulse width is controlled by internal register Bit[8]: frm_sync0_trig Trigger frame synchronizer 0 asserts frm_sync immediately, pulse width is controlled by internal register Bit[7]: frm_sync3_load Load a value into frame synchronizer 3 counter Bit[6]: frm_sync2_load Load a value into frame synchronizer 2 counter Bit[5]: frm_sync1_load Load a value into frame synchronizer 1 counter

table 6-1 system control registers (sheet 10 of 32)

address offset	register name	reset value	R/W	description
				Bit[4]: frm_sync0_load Load a value into frame synchronizer 0 counter Bit[3]: frm_sync3_start Start frame synchronizer 3 counter Bit[2]: frm_sync2_start Start frame synchronizer 2 counter Bit[1]: frm_sync1_start Start frame synchronizer 1 counter Bit[0]: frm_sync0_start Start frame synchronizer 0 counter
0x010	SYS_WATCH_DOG	0x0000_0000	RW	Bit[31]: Watch dog enable Bit[30]: Watch dog clear Bit[29:0]: Watch dog threshold
0x014	SYS_ROOT_SEL	0x0088_8888	RW	Bit[31:30]: Reserved Bit[29]: sys_chg_clk Toggle this bit to change system fast and slow root clock Bit[28]: mtx1_chg_clk Toggle this bit to change main sensor path MIPI TX root clock Bit[27]: mtx0_chg_clk Toggle this bit to change depth sensor path (dp0,dp1) MIPI TX root clock Bit[26]: msnr_chg_clk Toggle this bit to change main sensor path root clock Bit[25]: dp1_chg_clk Toggle this bit to change datapath 1 root clock Bit[24]: dp0_chg_clk Toggle this bit to change datapath 0 root clock Bit[23]: sys_root_clk_sel[3] System root clock (fast/slow) 0: PLL clock (MPLL/SPLL) 1: Crystal clock (XCLK) Bit[22]: sys_root_clk_sel[2] System root clock (fast/slow) 0: MIPI PLL clock (MPLL) 1: System PLL clock (SPLL)

table 6-1 system control registers (sheet 11 of 32)

address offset	register name	reset value	R/W	description
				Bit[21]: sys_root_clk_sel[1] System root clock (fast/slow) 0: spll_isp 1: spll_sys Bit[20]: sys_root_clk_sel[0] System root clock (fast/slow) 0: mpll_tx 1: mpll_sys Bit[19]: mtx1_root_clk_sel[3] Main sensor MIPI TX domain root clock 0: PLL clock (MPLL/SPLL) 1: Crystal clock (XCLK) Bit[18]: mtx1_root_clk_sel[2] Main sensor MIPI TX domain root clock 0: MIPI PLL clock (MPLL) 1: System PLL clock (SPLL) Bit[17]: mtx1_root_clk_sel[1] Main sensor MIPI TX domain root clock 0: spll_isp 1: spll_sys Bit[16]: mtx1_root_clk_sel[0] Main sensor MIPI TX domain root clock 0: mpll_tx 1: mpll_sys Bit[15]: mtx0_root_clk_sel[3] Depth sensors MIPI TX domain root clock 0: PLL clock (MPLL/SPLL) 1: Crystal clock (XCLK) Bit[14]: mtx0_root_clk_sel[2] Depth sensors MIPI TX domain root clock 0: MIPI PLL clock (MPLL) 1: System PLL clock (SPLL) Bit[13]: mtx0_root_clk_sel[1] Depth sensors MIPI TX domain root clock 0: spll_isp 1: spll_sys Bit[12]: mtx0_root_clk_sel[0] Depth sensors MIPI TX domain root clock Bit[11]: msnr_root_clk_sel[3] Main sensor path root clock select 0: PLL clock (MPLL/SPLL) 1: Crystal clock (XCLK)

table 6-1 system control registers (sheet 12 of 32)

address offset	register name	reset value	R/W	description
				Bit[10]: msnr_root_clk_sel[2] Main sensor path root clock select 0: MIPI PLL clock (MPLL) 1: System PLL clock (SPLL) Bit[9]: msnr_root_clk_sel[1] Main sensor path root clock select 0: spll_isp 1: spll_sys Bit[8]: msnr_root_clk_sel[0] Main sensor path root clock select 0: mpll_tx 1: mpll_sys Bit[7]: dp1_root_clk_sel[3] Datapath 1 root clock select 0: PLL clock (MPLL/SPLL]) 1: Crystal clock (XCLK]) Bit[6]: dp1_root_clk_sel[2] Datapath 1 root clock select 0: MIPI PLL clock (MPLL]) 1: System PLL clock (SPLL]) Bit[5]: dp1_root_clk_sel[1] Datapath 1 root clock select 0: spll_isp 1: spll_sys Bit[4]: dp1_root_clk_sel[0] Datapath 1 root clock select 0: mpll_tx 1: mpll_sys Bit[3]: dp0_root_clk_sel[3] Datapath 0 root clock select 0: PLL clock (MPLL/SPLL) 1: Crystal clock (XCLK) Bit[2]: dp0_root_clk_sel[2] Datapath 0 root clock select 0: MIPI PLL clock (MPLL) 1: System PLL clock (SPLL) Bit[1]: dp0_root_clk_sel[1] Datapath 0 root clock select 0: spll_isp 1: spll_sys Bit[0]: dp0_root_clk_sel[0] Datapath 0 root clock select 0: mpll_tx 1: mpll_sys

table 6-1 system control registers (sheet 13 of 32)

address offset	register name	reset value	R/W	description
0x018	SYS_CLK_EN0	0x0000_0000	RW	Bit[31]: Sensor 1 reference clock enable Bit[30]: Sensor 0 reference clock enable Bit[29]: Main sensor path frame control clock enable Bit[28]: Main sensor path embedded line clock enable Bit[27]: Main sensor path format and vfifo_in clock enable Bit[26]: Main sensor path DPCM clock enable Bit[25]: Reserved Bit[24]: Main sensor path colorbar clock enable Bit[23]: Reserved Bit[22]: Main sensor path MIPI RX clock enable Bit[21]: Main sensor path frame sync clock enable Bit[20]: Main sensor path root clock enable Bit[19]: Datapath 1 frame control clock enable Bit[18]: Datapath 1 embedded line clock enable Bit[17]: Datapath 1 format and vfifo_in clock enable Bit[16]: Datapath 1 ISP clock enable Bit[15]: Reserved Bit[14]: Datapath 1 colorbar clock enable Bit[13]: Reserved Bit[12]: Datapath 1 MIPI RX clock enable Bit[11]: Datapath 1 frame sync clock enable Bit[10]: Datapath 1 root clock enable Bit[9]: Datapath 0 frame control clock enable Bit[8]: Datapath 0 embedded line clock enable Bit[7]: Datapath 0 format and vfifo_in clock enable Bit[6]: Datapath 0 ISP clock enable Bit[5]: Reserved Bit[4]: Datapath 0 colorbar clock enable Bit[3]: Reserved Bit[2]: Datapath 0 MIPI RX clock enable Bit[1]: Datapath 0 frame sync clock enable Bit[0]: Datapath 0 root clock enable

table 6-1 system control registers (sheet 14 of 32)

address offset	register name	reset value	R/W	description
0x01C	SYS_CLK_EN1	0x01FF_FE00	RW	Bit[31:25]: Reserved Bit[24]: SCCB master and slave clock (slow) enable Bit[23]: SCCB master1 clock (slow) enable Bit[22]: SCCB master0 clock (slow) enable Bit[21]: SPI slave clock (slow) enable Bit[20]: System slow root clock (slow) enable Bit[19]: Context switch clock (fast) enable Bit[18]: BIST clock (fast) enable Bit[17]: Reserved Bit[16]: GPIO clock (fast) enable Bit[15]: SPI master and slave clock (fast) enable Bit[14]: UART clock (fast) enable Bit[13]: PROM clock (fast) enable Bit[12]: PRAM clock (fast) enable Bit[11]: RISC clock (fast) enable Bit[10]: AMBA clock (fast) enable Bit[9]: System fast root clock (fast) enable Bit[8]: VCCTRL1 clock enable Bit[7]: VFIFO2 clock enable Bit[6]: MSNR MIPI TX digital clock enable Bit[5]: MIPI TX 1 root clock enable Bit[4]: VCCTRL0 clock enable Bit[3]: VFIFO1 clock enable Bit[2]: VFIFO0 clock enable Bit[1]: MIPI TX digital 0 clock enable Bit[0]: MIPI TX 0 root clock enable
0x020	SYS_DIV_NUM0	0x0101_0101	RW	Bit[31:24]: mtx0_div_num Depth sensor MIPI TX domain divider number Bit[23:16]: msnr_div_num Main sensor path divider number Bit[15: 8]: dp1_div_num Datapath 1 divider number Bit[7:0]: dp0_div_num Datapath 0 divider number

table 6-1 system control registers (sheet 15 of 32)

address offset	register name	reset value	R/W	description
0x024	SYS_DIV_NUM1	0x0101_0101	RW	Bit[31:24]: cclk0_div_num Sensor reference clock 0 divider number Bit[23:16]: syss_div_num System (slow) divider number Bit[15:8]: sysf_div_num System (fast) divider number Bit[7:0]: mtx1_div_num Main sensor MIPI TX domain divider number
0x028	SYS_DIV_NUM2	0x0000_0004	RW	Bit[31:8]: Reserved Bit[7:0]: cclk1_div_num Sensor reference clock 1 divider number
0x02C	NOT USED	–	–	Not Used
0x030	SYS_SW_RST0	0x3FFF_FFFF	RW	Bit[31:30]: Reserved Bit[29]: Main sensor path frame control software reset Bit[28]: Main sensor path embedded line software reset Bit[27]: Main sensor path format and vfifo_in software reset Bit[26]: Main sensor path DPCM software reset Bit[25]: Main sensor path mux and async software reset Bit[24]: Main sensor path colorbar software reset Bit[23]: Main sensor path RX PHY parallel interface reset Bit[22]: Main sensor path MIPI RX software reset Bit[21]: Main sensor path frame sync software reset Bit[20]: Main sensor path root software reset Bit[19]: Datapath 1 frame control software reset Bit[18]: Datapath 1 embedded line software reset Bit[17]: Datapath 1 format and vfifo_in software reset Bit[16]: Datapath 1 ISP software reset Bit[15]: Datapath 1 mux and async software reset Bit[14]: Datapath 1 colorbar software reset

table 6-1 system control registers (sheet 16 of 32)

address offset	register name	reset value	R/W	description
				Bit[13]: Datapath 1 RX PHY parallel interface reset Bit[12]: Datapath 1 MIPI RX software reset Bit[11]: Datapath 1 frame sync software reset Bit[10]: Datapath 1 root software reset Bit[9]: Datapath 0 frame control software reset Bit[8]: Datapath 0 embedded line software reset Bit[7]: Datapath 0 format and vfifo_in software reset Bit[6]: Datapath 0 ISP software reset Bit[5]: Datapath 0 mux and async software reset Bit[4]: Datapath 0 colorbar software reset Bit[3]: Datapath 0 RX PHY parallel interface reset Bit[2]: Datapath 0 MIPI RX software reset Bit[1]: Datapath 0 frame sync software reset Bit[0]: Datapath 0 root software reset

table 6-1 system control registers (sheet 17 of 32)

address offset	register name	reset value	R/W	description
0x034	SYS_SW_RST1	0x0000_01FF	RW	Bit[31:25]: Reserved Bit[24]: SCCB master and slave software reset Bit[23]: SCCB master 1 software reset Bit[22]: SCCB master 0 software reset Bit[21]: SPI slave software reset Bit[20]: System slow root software reset Bit[19]: Context switch software reset Bit[18]: BIST software reset Bit[17]: Reserved Bit[16]: GPIO software reset Bit[15]: SPI master and slave software reset Bit[14]: UART software reset Bit[13]: PROM software reset Bit[12]: PRAM software reset Bit[11]: RISC software reset Bit[10]: AMBA software reset Bit[9]: System fast root software reset Bit[8]: VCCTRL1 software reset Bit[7]: VFIFO2 software reset Bit[6]: MIPI TX digital 1 software reset Bit[5]: MIPI TX 1 root software reset Bit[4]: VCCTRL0 software reset Bit[3]: VFIFO1 software reset Bit[2]: VFIFO0 software reset Bit[1]: MIPI TX digital 0 software reset Bit[0]: MIPI TX 0 root software reset
0x038	SYS_SW_RST2	0x0000_0000	RW	Bit[31]: pk_reg_rst_inresume If this bit is set, power keep domain register will be reset when system resume Bit[30:27]: Reserved Bit[26]: pk_reg_sw_rst Power keep domain register reset Bit[25]: pd_reg_sw_rst Power down domain register reset Bit[24]: chip_glb_logic_rst Reset whole chip except CPU/bus Bit[23:21]: Reserved Bit[20]: mtx1_glb_reg_rst Reset MIPI TX1 domain Control and status register (CSR) includes: MIPI_TX1, VFIFO2, VCCTRL1 in MIPI TX0 domain

table 6-1 system control registers (sheet 18 of 32)

address offset	register name	reset value	R/W	description
				Bit[19]: mtx0_glb_reg_rst Reset MIPI TX0 domain CSR includes: MIPI_TX0, VFIFO0, VFIFO1, VCCTRL0 in MIPI TX0 domain Note: If main sensor stream is transmitted through MIPI TX0, software must set bit[12] to reset VFIFO2, VCCTRL1 if software sets bit[11] to 1. Bit[18]: msnr_glb_reg_rst Reset main sensor path CSR includes: mipi_rx, embline, ISP, format, vfifo_fi in main sensor path Bit[17]: dp1_glb_reg_rst Reset datapath 1 CSR includes: mipi_rx, embline, ISP, format, vfifo_fi in datapath 1 Bit[16]: dp0_glb_reg_rst Reset datapath 0 CSR includes: mipi_rx, embline, ISP, format, vfifo_fi in datapath 0 Bit[15:13]: Reserved Bit[12]: mtx1_glb_logic_rst Reset MIPI TX1 domain logic including: mipi_tx1, VFIFO2, VCCTRL1 in MIPI TX0 domain Bit[11]: mtx0_glb_logic_rst Reset MIPI TX0 domain logic, including: mipi_tx0, VFIFO0, VFIFO1, VCCTRL0 in MIPI TX0 domain Note: If main sensor stream is transmitted through MIPI TX0, software must set bit[12] to reset VFIFO2, VCCTRL1 if software sets bit[11] to 1'b1. Bit[10]: msnr_glb_logic_rst Reset main sensor path logic, including: mipi_rx, embline, ISP, format, vfifo_fi in main sensor path Bit[9]: dp1_glb_logic_rst Reset datapath 1 logic, including: mipi_rx, embline, ISP, format, vfifo_fi in datapath 1

table 6-1 system control registers (sheet 19 of 32)

address offset	register name	reset value	R/W	description
				Bit[8]: dp0_glb_logic_rst Reset datapath 0 logic, including: mipi_rx, embline, isp, format, vfifo_fi in datapath 0 Bit[7:0]: Reserved Note: This group of registers only can be reset by external reset signal
0x03C	NOT USED	–	–	Not Used
0x040	SYS_IO_OEN	0xFFFF_FFFF	RW	Bit[31:29]: Reserved Bit[28]: MSSDA (SCCB bus data) pad direction select 0: Out 1: In Bit[27]: MSSCL (SCCB bus clock) pad direction select 0: Out 1: In Bit[26]: M1SDA (SCCB bus data) pad direction select 0: Out 1: In Bit[25]: M1SCL (SCCB bus clock) pad direction select 0: Out 1: In Bit[24]: M0SDA (SCCB bus data) pad direction select 0: Out 1: In Bit[23]: M0SCL (SCCB bus clock) pad direction select 0: Out 1: In Bit[22]: Flush pad direction select 0: Out 1: In Bit[21]: Reserved Bit[20]: GPIO4 pad direction select 0: Out 1: In Bit[19]: GPIO3 pad direction select 0: Out 1: In Bit[18]: GPIO2 pad direction select 0: Out 1: In

table 6-1 system control registers (sheet 20 of 32)

address offset	register name	reset value	R/W	description
				Bit[17]: GPIO1 pad direction select 0: Out 1: In Bit[16]: GPIO0 pad direction select 0: Out 1: In Bit[15]: MOSI (SPI master out slave in) pad direction select 0: Out 1: In Bit[14]: MISO (SPI master in slave out) pad direction select 0: Out 1: In Bit[13]: SSN (SPI chip select) pad direction select 0: Out 1: In Bit[12]: SPICK (SPI bus clock) pad direction select 0: Out 1: In Bit[11]: UART_RX (UART RX data) pad direction select 0: Out 1: In Bit[10]: UART_TX (UART TX data) pad direction select 0: Out 1: In Bit[9]: CCLK1 (sensor reference clock) pad direction select 0: Out 1: In Bit[8]: CCLK0 (sensor reference clock) pad direction select 0: Out 1: In Bit[7]: FSYNC3 (sensor frame sync) pad direction select 0: Out 1: In Bit[6]: FSYNC2 (sensor frame sync) pad direction select 0: Out 1: In Bit[5]: FSYNC1 (sensor frame sync) pad direction select 0: Out 1: In

table 6-1 system control registers (sheet 21 of 32)

address offset	register name	reset value	R/W	description
				Bit[4]: FSYNC0 (sensor frame sync) pad direction select 0: Out 1: In Bit[3]: SPWDN2 (sensor power down) pad direction select 0: Out 1: In Bit[2]: SPWDN1 (sensor power down) pad direction select 0: Out 1: In Bit[1]: SPWDN0 (sensor power down) pad direction select 0: Out 1: In Bit[0]: SCCB_ID(SCCB slave ID change) pad direction select 0: Out 1: In
0x044	SYS_IO_IE	0x1800_0001	RW	Bit[31:29]: Reserved Bit[28]: MSSDA (SCCB bus data) pad input enable Bit[27]: MSSCL (SCCB bus clock) pad input enable Bit[26]: M1SDA (SCCB bus data) pad input enable Bit[25]: M1SCL (SCCB bus clock) pad input enable Bit[24]: M0SDA (SCCB bus data) pad input enable Bit[23]: M0SCL(SCCB bus clock) pad input enable Bit[22]: Flush pad input enable Bit[21]: Reserved Bit[20]: GPIO4 pad input enable Bit[19]: GPIO3 pad input enable Bit[18]: GPIO2 pad input enable Bit[17]: GPIO1 pad input enable Bit[16]: GPIO0 pad input enable Bit[15]: MOSI (SPI master out slave in) pad input enable Bit[14]: MISO (SPI master in slave out) pad input enable Bit[13]: SSN (SPI chip select) pad input enable Bit[12]: SPICK (SPI bus clock) pad input enable

table 6-1 system control registers (sheet 22 of 32)

address offset	register name	reset value	R/W	description
				Bit[11]: UART_RX (UART RX data) pad input enable Bit[10]: UART_TX (UART TX data) pad input enable Bit[9]: CCLK1 (sensor reference clock) pad input enable Bit[8]: CCLK0 (sensor reference clock) pad input enable Bit[7]: FSYNC3 (sensor frame sync) pad input enable Bit[6]: FSYNC2 (sensor frame sync) pad input enable Bit[5]: FSYNC1 (sensor frame sync) pad input enable Bit[4]: FSYNC0 (sensor frame sync) pad input enable Bit[3]: SPWDN2 (sensor power down) pad input enable Bit[2]: SPWDN1 (sensor power down) pad input enable Bit[1]: SPWDN0 (sensor power down) pad input enable Bit[0]: SCCB_ID (SCCB slave ID change) pad input enable
0x048	SYS_IO_PE	0x0000_0000	RW	Bit[31:29]: Reserved Bit[28]: MSSDA (SCCB bus data) pad pull enable Bit[27]: MSSCL (SCCB bus clock) pad pull enable Bit[26]: M1SDA (SCCB bus data) pad pull enable Bit[25]: M1SCL (SCCB bus clock) pad pull enable Bit[24]: M0SDA (SCCB bus data) pad pull enable Bit[23]: M0SCL (SCCB bus clock) pad pull enable Bit[22]: Flush pad pull enable Bit[21]: Reserved Bit[20]: GPIO4 pad pull enable Bit[19]: GPIO3 pad pull enable Bit[18]: GPIO2 pad pull enable Bit[17]: GPIO1 pad pull enable Bit[16]: GPIO0 pad pull enable Bit[15]: MOSI (SPI master out slave in) pad pull enable Bit[14]: MISO (SPI master in slave out) pad pull enable

table 6-1 system control registers (sheet 23 of 32)

address offset	register name	reset value	R/W	description
				Bit[13]: SSN (SPI chip select) pad pull enable Bit[12]: SPICK (SPI bus clock) pad pull enable Bit[11]: UART_RX (UART RX data) pad pull enable Bit[10]: UART_TX (UART TX data) pad pull enable Bit[9]: CCLK1 (sensor reference clock) pad pull enable Bit[8]: CCLK0 (sensor reference clock) pad pull enable Bit[7]: FSYNC3 (sensor frame sync) pad pull enable Bit[6]: FSYNC2 (sensor frame sync) pad pull enable Bit[5]: FSYNC1 (sensor frame sync) pad pull enable Bit[4]: FSYNC0 (sensor frame sync) pad pull enable Bit[3]: SPWDN2 (sensor power down) pad pull enable Bit[2]: SPWDN1 (sensor power down) pad pull enable Bit[1]: SPWDN0 (sensor power down) pad pull enable Bit[0]: SCCB_ID (SCCB slave ID change) pad pull enable
0x04C	SYS_IO_DS	0x0000_0000	RW	Bit[31:29]: Reserved Bit[28]: MSSDA (SCCB bus data) pad driven strength Bit[27]: MSSCL (SCCB bus clock) pad driven strength Bit[26]: M1SDA (SCCB bus data) pad driven strength Bit[25]: M1SCL (SCCB bus clock) pad driven strength Bit[24]: M0SDA (SCCB bus data) pad driven strength Bit[23]: M0SCL (SCCB bus clock) pad driven strength Bit[22]: Flush pad driven strength Bit[21]: Reserved Bit[20]: GPIO4 pad driven strength Bit[19]: GPIO3 pad driven strength Bit[18]: GPIO2 pad driven strength Bit[17]: GPIO1 pad driven strength Bit[16]: GPIO0 pad driven strength

table 6-1 system control registers (sheet 24 of 32)

address offset	register name	reset value	R/W	description
				Bit[15]: MOSI (SPI master out slave in) pad driven strength Bit[14]: MISO (SPI master in slave out) pad driven strength Bit[13]: SSN (SPI chip select) pad driven strength Bit[12]: SPICK (SPI bus clock) pad driven strength Bit[11]: UART_RX (UART RX data) pad driven strength Bit[10]: UART_TX (UART TX data) pad driven strength Bit[9]: CCLK1 (sensor reference clock) pad driven strength Bit[8]: CCLK0 (sensor reference clock) pad driven strength Bit[7]: FSYNC3 (sensor frame sync) pad driven strength Bit[6]: FSYNC2 (sensor frame sync) pad driven strength Bit[5]: FSYNC1 (sensor frame sync) pad driven strength Bit[4]: FSYNC0 (sensor frame sync) pad driven strength Bit[3]: SPWDN2 (sensor power down) pad driven strength Bit[2]: SPWDN1 (sensor power down) pad driven strength Bit[1]: SPWDN0 (sensor power down) pad driven strength Bit[0]: SCCB_ID (SCCB slave ID change) pad driven strength
0x050	SYS_GPIO_EN	0x0000_0000	RW	Bit[31:29]: Reserved Bit[28]: MSSDA(SCCB bus data) pad as GPIO Bit[27]: MSSCL(SCCB bus clock) pad as GPIO Bit[26]: M1SDA(SCCB bus data) pad as GPIO Bit[25]: M1SCL(SCCB bus clock) pad as GPIO Bit[24]: M0SDA(SCCB bus data) pad as GPIO Bit[23]: M0SCL(SCCB bus clock) pad as GPIO Bit[22]: Flush pad as GPIO Bit[21]: Reserved Bit[20]: GPIO4 pad as GPIO Bit[19]: GPIO3 pad as GPIO

table 6-1 system control registers (sheet 25 of 32)

address offset	register name	reset value	R/W	description
				Bit[18]: GPIO2 pad as GPIO Bit[17]: GPIO1 pad as GPIO Bit[16]: GPIO0 pad as GPIO Bit[15]: MOSI (SPI master out slave in) pad as GPIO Bit[14]: MISO (SPI master in slave out) pad as GPIO Bit[13]: SSN (SPI chip select) pad as GPIO Bit[12]: SPICK (SPI bus clock) pad as GPIO Bit[11]: UART_RX (UART RX data) pad as GPIO Bit[10]: UART_TX (UART TX data) pad as GPIO Bit[9]: CCLK1 (sensor reference clock) pad as GPIO Bit[8]: CCLK0 (sensor reference clock) pad as GPIO Bit[7]: FSYNC3 (sensor frame sync) pad as GPIO Bit[6]: FSYNC2 (sensor frame sync) pad as G PIO Bit[5]: FSYNC1 (sensor frame sync) pad as GPIO Bit[4]: FSYNC0 (sensor frame sync) pad as GPIO Bit[3]: SPWDN2 (sensor power down) pad as GPIO Bit[2]: SPWDN1 (sensor power down) pad as GPIO Bit[1]: SPWDN0 (sensor power down) pad as GPIO Bit[0]: SCCB_ID (SCCB slave ID change) pad as GPIO
0x054	SYS_GPIO_DO	0x0000_0000	RW	Bit[31:29]: Reserved Bit[28]: MSSDA (SCCB bus data) pad GPIO data output Bit[27]: MSSCL (SCCB bus clock) pad GPIO data output Bit[26]: M1SDA (SCCB bus data) pad GPIO data output Bit[25]: M1SCL (SCCB bus clock) pad GPIO data output Bit[24]: M0SDA (SCCB bus data) pad GPIO data output Bit[23]: M0SCL (SCCB bus clock) pad GPIO data output Bit[22]: Flush pad data output

table 6-1 system control registers (sheet 26 of 32)

address offset	register name	reset value	R/W	description
				Bit[21]: Reserved Bit[20]: GPIO4 pad data output Bit[19]: GPIO3 pad data output Bit[18]: GPIO2 pad data output Bit[17]: GPIO1 pad data output Bit[16]: GPIO0 pad data output Bit[15]: MOSI (SPI master out slave in) pad GPIO data output Bit[14]: MISO (SPI master in slave out) pad GPIO data output Bit[13]: SSN (SPI chip select) pad GPIO data output Bit[12]: SPICK (SPI bus clock) pad GPIO data output Bit[11]: UART_RX (UART RX data) pad GPIO data output Bit[10]: UART_TX (UART TX data) pad GPIO data output Bit[9]: CCLK1 (sensor reference clock) pad GPIO data output Bit[8]: CCLK0 (sensor reference clock) pad GPIO data output Bit[7]: FSYNC3(sensor frame sync) pad GPIO data output Bit[6]: FSYNC2 (sensor frame sync) pad GPIO data output Bit[5]: FSYNC1 (sensor frame sync) pad GPIO data output Bit[4]: FSYNC0 (sensor frame sync) pad GPIO data output Bit[3]: SPWDN2 (sensor power down) pad GPIO data output Bit[2]: SPWDN1 (sensor power down) pad GPIO data output Bit[1]: SPWDN0 (sensor power down) pad GPIO data output Bit[0]: SCCB_ID (SCCB slave ID change) pad GPIO data output
0x058	SYS_GPIO_ PINTREN	0x0000_0000	RW	Bit[31:29]: Reserved Bit[28]: MSSDA (SCCB bus data) pad GPIO posedge interrupt enable Bit[27]: MSSCL (SCCB bus clock) pad GPIO posedge interrupt enable Bit[26]: M1SDA (SCCB bus data) pad GPIO posedge interrupt enable Bit[25]: M1SCL (SCCB bus clock) pad GPIO posedge interrupt enable Bit[24]: M0SDA (SCCB bus data) pad GPIO posedge interrupt enable

table 6-1 system control registers (sheet 27 of 32)

address offset	register name	reset value	R/W	description
				Bit[23]: M0SCL (SCCB bus clock) pad GPIO posedge interrupt enable Bit[22]: Flush pad posedge interrupt enable Bit[21]: Reserved Bit[20]: GPIO4 pad posedge interrupt enable Bit[19]: GPIO3 pad posedge interrupt enable Bit[18]: GPIO2 pad posedge interrupt enable Bit[17]: GPIO1 pad posedge interrupt enable Bit[16]: GPIO0 pad posedge interrupt enable Bit[15]: MOSI (SPI master out slave in) pad GPIO posedge interrupt enable Bit[14]: MISO (SPI master in slave out) pad GPIO posedge interrupt enable Bit[13]: SSN (SPI chip select) pad GPIO posedge interrupt enable Bit[12]: SPICK (SPI bus clock) pad GPIO posedge interrupt enable Bit[11]: UART_RX (UART RX data) pad GPIO posedge interrupt enable Bit[10]: UART_TX (UART TX data) pad GPIO posedge interrupt enable Bit[9]: CCLK1 (sensor reference clock) pad GPIO posedge interrupt enable Bit[8]: CCLK0(sensor reference clock) pad GPIO posedge interrupt enable Bit[7]: FSYNC3 (sensor frame sync) pad GPIO posedge interrupt enable Bit[6]: FSYNC2 (sensor frame sync) pad GPIO posedge interrupt enable Bit[5]: FSYNC1 (sensor frame sync) pad GPIO posedge interrupt enable Bit[4]: FSYNC0 (sensor frame sync) pad GPIO posedge interrupt enable Bit[3]: SPWDN2 (sensor power down) pad GPIO posedge interrupt enable

table 6-1 system control registers (sheet 28 of 32)

address offset	register name	reset value	R/W	description
				Bit[2]: SPWDN1 (sensor power down) pad GPIO posedge interrupt enable Bit[1]: SPWDN0 (sensor power down) pad GPIO posedge interrupt enable Bit[0]: SCCB_ID (SCCB slave ID change) pad GPIO posedge interrupt enable
0x05C	SYS_GPIO_NINTR_EN	0x0000_0000	RW	Bit[31:29]: Reserved Bit[28]: MSSDA (SCCB bus data) pad GPIO negedge interrupt enable Bit[27]: MSSCL (SCCB bus clock) pad GPIO negedge interrupt enable Bit[26]: M1SDA (SCCB bus data) pad GPIO negedge interrupt enable Bit[25]: M1SCL (SCCB bus clock) pad GPIO negedge interrupt enable Bit[24]: M0SDA (SCCB bus data) pad GPIO negedge interrupt enable Bit[23]: M0SCL (SCCB bus clock) pad GPIO negedge interrupt enable Bit[22]: Flush pad negedge interrupt enable Bit[21]: Reserved Bit[20]: GPIO4 pad negedge interrupt enable Bit[19]: GPIO3 pad negedge interrupt enable Bit[18]: GPIO2 pad negedge interrupt enable Bit[17]: GPIO1 pad negedge interrupt enable Bit[16]: GPIO0 pad negedge interrupt enable Bit[15]: MOSI (SPI master out slave in) pad GPIO negedge interrupt enable Bit[14]: MISO (SPI master in slave out) pad GPIO negedge interrupt enable Bit[13]: SSN (SPI chip select) pad GPIO negedge interrupt enable Bit[12]: SPICK (SPI bus clock) pad GPIO negedge interrupt enable Bit[11]: UART_RX (UART RX data) pad GPIO negedge interrupt enable Bit[10]: UART_TX (UART TX data) pad GPIO negedge interrupt enable

table 6-1 system control registers (sheet 29 of 32)

address offset	register name	reset value	R/W	description
				Bit[9]: CCLK1 (sensor reference clock) pad GPIO negedge interrupt enable Bit[8]: CCLK0 (sensor reference clock) pad GPIO negedge interrupt enable Bit[7]: FSYNC3 (sensor frame sync) pad GPIO negedge interrupt enable Bit[6]: FSYNC2 (sensor frame sync) pad GPIO negedge interrupt enable Bit[5]: FSYNC1 (sensor frame sync) pad GPIO negedge interrupt enable Bit[4]: FSYNC0 (sensor frame sync) pad GPIO negedge interrupt enable Bit[3]: SPWDN2 (sensor power down) pad GPIO negedge interrupt enable Bit[2]: SPWDN1 (sensor power down) pad GPIO negedge interrupt enable Bit[1]: SPWDN0 (sensor power down) pad GPIO negedge interrupt enable Bit[0]: SCCB_ID (SCCB slave ID change) pad GPIO negedge interrupt enable
0x060	SYS_GPIO_POS_CLR	0x0000_0000	RW	Bit[31:29]: Reserved Bit[28]: MSSDA (SCCB bus data) pad GPIO posedge interrupt clear Bit[27]: MSSCL (SCCB bus clock) pad GPIO posedge interrupt clear Bit[26]: M1SDA (SCCB bus data) pad GPIO posedge interrupt clear Bit[25]: M1SCL (SCCB bus clock) pad GPIO posedge interrupt clear Bit[24]: M0SDA (SCCB bus data) pad GPIO posedge interrupt clear Bit[23]: M0SCL (SCCB bus clock) pad GPIO posedge interrupt clear Bit[22]: Flush pad posedge interrupt clear Bit[21]: Reserved Bit[20]: GPIO4 pad posedge interrupt clear

table 6-1 system control registers (sheet 30 of 32)

address offset	register name	reset value	R/W	description
				Bit[19]: GPIO3 pad posedge interrupt clear Bit[18]: GPIO2 pad posedge interrupt clear Bit[17]: GPIO1 pad posedge interrupt clear Bit[16]: GPIO0 pad posedge interrupt clear Bit[15]: MOSI (SPI master out slave in) pad GPIO posedge interrupt clear Bit[14]: MISO (SPI master in slave out) pad GPIO posedge interrupt clear Bit[13]: SSN (SPI chip select) pad GPIO posedge interrupt clear Bit[12]: SPICK (SPI bus clock) pad GPIO posedge interrupt clear Bit[11]: UART_RX (UART RX data) pad GPIO posedge interrupt clear Bit[10]: UART_TX (UART TX data) pad GPIO posedge interrupt clear Bit[9]: CCLK1 (sensor reference clock) pad GPIO posedge interrupt clear Bit[8]: CCLK0 (sensor reference clock) pad GPIO posedge interrupt clear Bit[7]: FSYNC3 (sensor frame sync) pad GPIO posedge interrupt clear Bit[6]: FSYNC2 (sensor frame sync) pad GPIO posedge interrupt clear Bit[5]: FSYNC1 (sensor frame sync) pad GPIO posedge interrupt clear Bit[4]: FSYNC0 (sensor frame sync) pad GPIO posedge interrupt clear Bit[3]: SPWDN2 (sensor power down) pad GPIO posedge interrupt clear Bit[2]: SPWDN1 (sensor power down) pad GPIO posedge interrupt clear Bit[1]: SPWDN0 (sensor power down) pad GPIO posedge interrupt clear

table 6-1 system control registers (sheet 31 of 32)

address offset	register name	reset value	R/W	description
				Bit[0]: SCCB_ID (SCCB slave ID change) pad GPIO posedge interrupt clear Note: Self-clearing register
0x064	SYS_GPIO_NEG_CLR	0x0000_0000	RW	Bit[31:29]: Reserved Bit[28]: MSSDA (SCCB bus data) pad GPIO negedge interrupt clear Bit[27]: MSSCL (SCCB bus clock) pad GPIO negedge interrupt clear Bit[26]: M1SDA(SCCB bus data) pad GPIO negedge interrupt clear Bit[25]: M1SCL (SCCB bus clock) pad GPIO negedge interrupt clear Bit[24]: M0SDA (SCCB bus data) pad GPIO negedge interrupt clear Bit[23]: M0SCL (SCCB bus clock) pad GPIO negedge interrupt clear Bit[22]: Flush pad negedge interrupt clear Bit[21]: Reserved Bit[20]: GPIO4 pad negedge interrupt clear Bit[19]: GPIO3 pad negedge interrupt clear Bit[18]: GPIO2 pad negedge interrupt clear Bit[17]: GPIO1 pad negedge interrupt clear Bit[16]: GPIO0 pad negedge interrupt clear Bit[15]: MOSI (SPI master out slave in) pad GPIO negedge interrupt clear Bit[14]: MISO (SPI master in slave out) pad GPIO negedge interrupt clear Bit[13]: SSN (SPI chip select) pad GPIO negedge interrupt clear Bit[12]: SPICK (SPI bus clock) pad GPIO negedge interrupt clear Bit[11]: UART_RX (UART RX data) pad GPIO negedge interrupt clear Bit[10]: UART_TX (UART TX data) pad GPIO negedge interrupt clear Bit[9]: CCLK1 (sensor reference clock) pad GPIO negedge interrupt clear

table 6-1 system control registers (sheet 32 of 32)

address offset	register name	reset value	R/W	description
				Bit[8]: CCLK0 (sensor reference clock) pad GPIO negedge interrupt clear Bit[7]: FSYNC3 (sensor frame sync) pad GPIO negedge interrupt clear Bit[6]: FSYNC2 (sensor frame sync) pad GPIO negedge interrupt clear Bit[5]: FSYNC1 (sensor frame sync) pad GPIO negedge interrupt clear Bit[4]: FSYNC0 (sensor frame sync) pad GPIO negedge interrupt clear Bit[3]: SPWDN2 (sensor power down) pad GPIO negedge interrupt clear Bit[2]: SPWDN1 (sensor power down) pad GPIO negedge interrupt clear Bit[1]: SPWDN0 (sensor power down) pad GPIO negedge interrupt clear Bit[0]: SCCB_ID (SCCB slave ID change) pad GPIO negedge interrupt clear Note: Self-clearing register
0x068	SYS_AEC_CTRL0	0x0200_0200	RW	Bit[31:30]: Reserved Bit[29]: dp1_f50 50Hz/60Hz 0: 60Hz 1: 50Hz Bit[28:16]: dp1_aec_vts Vertical total size for AEC Bit[15:14]: Reserved Bit[13]: dp0_f50 50Hz/60Hz 0: 60Hz 1: 50Hz Bit[12:0]: dp0_aec_vts Vertical total size for AEC
0x06C~ 0x07C	NOT USED	–	–	Not Used

6.2 SCCB

- base address: 0x80161000 (SCCB0)
- base address: 0x80161100 (SCCB1)
- base address: 0x80161200 (SCCB_MS)

table 6-2 SCCB registers (sheet 1 of 6)

address offset	register name	reset value	R/W	description
0x00	SCCB CTRL 0	0x41	RW	Bit[7:6]: SCCB slave's internal address width 00: 8-bit 01: 16-bit 10: 24-bit 11: 32-bit Bit[5:4]: SCCB slave's internal data width 00: 8-bit 01: 16-bit 10: 24-bit 11: 32-bit Bit[3]: SCCB slave ID width 0: 7-bit 1: 10-bit Bit[2]: SCCB slave stretch SCL low enable 0: Disable 1: Enable Bit[1]: SCCB slave stretch SCL low mode 0: Hardware 1: Software Bit[0]: Software control bit for stretch SCL low (hardware clear, software set) 0: Hold SCL low 1: Release SCL high
0x01	SCCB CTRL 1	0x03	RW	Bit[7]: Extra busy detection enable 0: Disable 1: Enable Bit[6]: Hardware handle bus conflict enable 0: Disable 1: Enable Bit[5:3]: Reserved Bit[2]: START condition arbitration enable 0: Disable 1: Enable Bit[1]: STOP condition arbitration enable 0: Disable 1: Enable Bit[0]: DATA and ACK bit arbitration enable 0: Disable 1: Enable

table 6-2 SCCB registers (sheet 2 of 6)

address offset	register name	reset value	R/W	description
0x02	SCCB CTRL 2	0x00	RW	Bit[7]: SCCB master re-transmit (pulse) Bit[6]: AHB access size 0: 8-bit 1: 32-bit Bit[5]: SCCB slave AHB base address mode 0: Use internal register to store base address 1: Use 0xFFFD and 0xFFFE to store base address Bit[4]: SCCB slave no send ACK bit 0: Disable 1: Enable Bit[3]: RCV buffer clear (pulse) Bit[2]: TRN buffer's read pointer clear (pulse) Bit[1]: TRN buffer clear (pulse) Bit[0]: SCCB master abort (pulse)
0x03	SCCB CTRL 3	0xA0	RW	Bit[7]: SCCB enable 0: Disable 1: Enable Bit[6]: SCCB mode 0: Slave 1: Master Bit[5:3]: RCV buffer depth 000: 0 byte 001: 4 bytes 010: 8 bytes 011: 12 bytes 100: 16 bytes 101: 20 bytes 110: 24 bytes 111: 28 bytes Bit[2]: SCCB slave cmd decode mode 0: Hardware 1: Software Bit[1:0]: RCV buffer read byte 00: 3 bytes 01: 1 bytes 10: 2 bytes 11: 3 bytes

table 6-2 SCCB registers (sheet 3 of 6)

address offset	register name	reset value	R/W	description
0x04	SCCB STAT0	0x24	R	Bit[7:6]: Reserved Bit[5]: TRN buffer empty Bit[4]: RCV buffer full Bit[3]: TRN buffer full Bit[2]: RCV buffer empty Bit[1]: TRN buffer almost full Bit[0]: SCCB transmission direction 0: From master to slave 1: From slave to master
0x05	SCCB STAT1	–	R	Bit[7]: SCCB slave ID match Bit[6:0]: Reserved
0x06	SCCB STAT2	–	R	Bit[7:2]: Reserved Bit[1]: SCCB master busy Bit[0]: Arbitration lost (software clear)
0x07	SCCB STAT3	–	R	Bit[7]: SCCB bus busy Bit[6]: ACK stat 0: ACK received 1: NACK received Bit[5:0]: Reserved
0x08	SCCB ID0	0x42 (MS) 0x46 (M0) 0x4A (M1)	RW	Bit[7:0]: SCCB slave ID0[7:0]
0x09	SCCB ID0	0x00	RW	Bit[7:2]: Reserved Bit[1:0]: SCCB slave ID0[9:8]
0x0A	SCCB ID1	0x44 (MS) 0x48 (M0) 0x4C (M1)	RW	Bit[7:0]: SCCB slave ID1[7:0]
0x0B	SCCB ID1	0x00	RW	Bit[7:2]: Reserved Bit[1:0]: SCCB slave ID1[9:8]
0x0C	SCCB BRTL	0xFF	RW	Bit[7:0]: Base value for SCCB master's baud-rate counter[7:0]
0x0D	SCCB BRTH	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: Base value for SCCB master's baud-rate counter[11:8]
0x0E	SCCB DCA	0x00	RW	Bit[7:0]: SCCB master's SCL duty-cycle adjustment SCL's duty-cycle decreases when this value increases
0x0F	RSVD	–	–	Reserved
0x10	SCCB RCVLV	0x00	RW	Bit[7:5]: Reserved Bit[4:0]: Threshold for RCV buffer's level interrupt

table 6-2 SCCB registers (sheet 4 of 6)

address offset	register name	reset value	R/W	description
0x11	RSVD	–	–	Reserved
0x12	SCCB TRNLV	0x00	RW	Bit[7:6]: Reserved Bit[5:0]: Threshold for TRN buffer's level interrupt
0x13	RSVD	–	–	Reserved
0x14	SCCB WTC	0xFF	RW	Bit[7:0]: Threshold for wait-out interrupt This value must not be set to 0
0x15~ 0x17	RSVD	–	–	Reserved
0x18	SCCB INTERRUPT ENABLE	0x00	RW	Bit[7]: TRN buffer level interrupt enable Bit[6]: RCV buffer level interrupt enable Bit[5]: Slave's NACK received interrupt enable Bit[4]: Slave's read request (when received ID+R and match) interrupt enable Bit[3]: Master's ACK received interrupt enable Bit[2]: Master's wait-out interrupt enable Bit[1]: Master's arbitration interrupt enable Bit[0]: Reserved
0x19	RSVD	–	–	Reserved
0x1A	SCCB INTERRUPT STATUS	0x00	RW	Bit[7]: TRN buffer level interrupt flag Bit[6]: RCV buffer level interrupt flag Bit[5]: Slave's NACK received interrupt flag Bit[4]: Slave's read request interrupt flag Bit[3]: Master's ACK received interrupt flag Bit[2]: Master's wait-out interrupt flag Bit[1]: Master's arbitration interrupt flag Bit[0]: Master's NACK received interrupt flag All flag bit is set by hardware and clear by software
0x1B	RSVD	–	–	Reserved
0x1C	ACK ID LOW	–	R	Bit[7:0]: Latest ACK slave ID[7:0]
0x1D	ACK ID HIGH	–	R	Bit[7:0]: Latest ACK slave ID[15:8]
0x1E~ 0x1F	RSVD	–	–	Reserved
0x20	RCV USEDW	–	R	Bit[7:0]: Number of used unit in RCV buffer
0x21	RSVD	–	–	Reserved
0x22	TRN USEDW	–	R	Bit[7:0]: Number of used unit in TRN buffer

table 6-2 SCCB registers (sheet 5 of 6)

address offset	register name	reset value	R/W	description
0x23	TRN UNUSE	0x10	R	Bit[7:0]: Number of un-used unit in TRN buffer
0x24	LAST ID LOW	–	R	Bit[7:0]: Last slave's ID transmitted by SCCB master[7:0]
0x25	LAST ID HIGH	–	R	Bit[7:0]: Last slave's ID transmitted by SCCB master[15:8]
0x26~ 0x27	RSVD	–	–	Reserved
0x28	LAST ADDR0	–	R	Bit[7:0]: Last byte of address transmitted by SCCB master
0x29	LAST ADDR1	–	R	Bit[7:0]: Address byte transmitted by SCCB master before Last addr0
0x2A	LAST ADDR2	–	R	Bit[7:0]: Address byte transmitted by SCCB master before Last addr1
0x2B	LAST ADDR3	–	R	Bit[7:0]: Address byte transmitted by SCCB master before Last addr2
0x2C	LAST DATA	–	R	Bit[7:0]: Last byte of data transmitted by SCCB master
0x2D~ 0x2F	RSVD	–	–	Reserved
0x30	CMD STAT	–	R	Bit[7:4]: SCCB master transmission stage 0000: Idle 1000: ID transmission 0100: ADDR transmission 0010: DATA transmission 0001: Read from slave Bit[3:0]: Burst offset[11:8]
0x31	CMD STAT	–	R	Bit[7:0]: Burst offset[7:0]
0x32~ 0x33	RSVD	–	–	Reserved
0x34	SDA HOLD TIME	0x01	RW	Bit[7:0]: Hold time of SCCB master SDA's data bit
0x35	SDA PAD CTRL	0x02	RW	Bit[7]: SDA pad spike filter enable 0: Disable 1: Enable Bit[6]: Reserved Bit[5:0]: SDA pad input sample delay cycle
0x36	SCL STR LEN	0x04	RW	Bit[7:0]: Length of SCCB slave stretch SCL low

table 6-2 SCCB registers (sheet 6 of 6)

address offset	register name	reset value	R/W	description
0x37	SCL PAD CTRL	0x02	RW	Bit[7]: SCL pad spike filter enable 0: Disable 1: Enable Bit[6]: Reserved Bit[5:0]: SCL pad input sample delay cycle
0x38	BASE ADDR0	0x00	RW	Bit[7:0]: AHB base address[7:0]
0x39	BASE ADDR1	0x80	RW	Bit[7:0]: AHB base address[15:8]
0x3A	BASE ADDR2	0x14	RW	Bit[7:0]: AHB base address[23:16]
0x3B	BASE ADDR3	0x80	RW	Bit[7:0]: AHB base address[31:24]
0x3C	EXTRA PAD CTRL	0x03	RW	Bit[7:0]: SCL pad output delay cycle

6.3 ISP

6.3.1 address mapping of ISP

table 6-3 is the ISP0 register table. Register address of the ISP1 register table is 0x1000 larger than that of the ISP0, respectively.

table 6-3 ISP address mapping (sheet 1 of 2)

module name	function	address range
ISP0_TOP	ISP0 top registers	0x0016_2000 ~ 0x0016_207F
PRE_ISP_TOP	PRE ISP top registers	0x0016_2080 ~ 0x0016_20FF
YAVG	ISP0 YAVG registers	0x0016_2100 ~ 0x0016_217F
LENC	ISP0 LENC registers	0x0016_2180 ~ 0x0016_21FF
AWB	ISP0 AWB registers	0x0016_2200 ~ 0x0016_227F
AWB RO	ISP0 AWB RO registers	0x0016_2280 ~ 0x0016_22FF
EDR	ISP0 EDR registers	0x0016_2300 ~ 0x0016_237F
STRETCH	ISP0 stretch registers	0x0016_2380 ~ 0x0016_23FF
DPC	ISP0 DPC registers	0x0016_2400 ~ 0x0016_247F
RAW GAMMA	ISP0 raw gamma registers	0x0016_2480 ~ 0x0016_24FF
DNS	ISP0 DNS registers	0x0016_2500 ~ 0x0016_257F
CIP	ISP0 CIP registers	0x0016_2580 ~ 0x0016_25FF
CMX	ISP0 CMX registers	0x0016_2600 ~ 0x0016_267F

table 6-3 ISP address mapping (sheet 2 of 2)

module name	function	address range
RGB GAMMA	ISP0 RGB gamma registers	0x0016_2680 ~ 0x0016_26FF
RGB DNS	ISP0 RGB DNS registers	0x0016_2700 ~ 0x0016_277F
SDE	ISP0 SDE registers	0x0016_2780 ~ 0x0016_27FF
AEC	ISP0 AEC registers	0x0016_2800 ~ 0x0016_287F
UVSUM	ISP0 UVSUM registers	0x0016_2880 ~ 0x0016_28FF
UV DNS	ISP0 UV DNS registers	0x0016_2900 ~ 0x0016_297F
WINC	ISP0 WINC registers	0x0016_2980 ~ 0x0016_29FF
AEC PK	ISP0 AEC PK registers	0x0016_2A00 ~ 0x0016_2A7F
AWB PK	ISP0 AWB PK registers	0x0016_2A80 ~ 0x0016_2AFF

6.3.2 ISP registers

- base address: 0x80162000 (ISP0)
- base address: 0x80163000 (ISP1)

table 6-4 ISP registers (sheet 1 of 6)

address offset	register name	reset value	R/W	description
0x00	ISP EN LOW	0xFF	RW	Bit[7]: AEC enable (active high) Bit[6]: Stretch enable (active high) Bit[5]: Raw gamma enable (active high) Bit[4]: BC enable (active high) Bit[3]: WC enable (active high) Bit[2]: AWB gain enable (active high) Bit[1]: AWB enable (active high) Bit[0]: LENC enable (active high)
0x01	ISP EN MID	0xFF	RW	Bit[7]: Pad enable (active high) Bit[6]: RGB DNS enable (active high) Bit[5]: RGB2YUV enable (active high) Bit[4]: RGB gamma enable (active high) Bit[3]: CMX enable (active high) Bit[2]: Control enable (active high) Bit[1]: Cip enable (active high) Bit[0]: DNS enable (active high)

table 6-4 ISP registers (sheet 2 of 6)

address offset	register name	reset value	R/W	description
0x02	ISP EN HIGH	0xFF	RW	Bit[7]: EDR enable (active high) Bit[6]: UV DNS enable (active high) Bit[5]: YUV422to444 enable (active high) Bit[4]: YAVG enable (active high) Bit[3]: Reserved Bit[2]: Dgain enable (active high) Bit[1]: UV adj enable (active high) Bit[0]: SDE enable (active high)
0x03	DATA PATH	0x02	RW	Bit[7]: Reserved Bit[6]: Pudding Add bits to make RAW8 or RAW10 to RAW12 Bit[5]: YUV out Bit[4]: RAW bypass, do WINC Bit[3]: YUV bypass, do WINC Bit[2]: Raw 12 in Bit[1]: Raw 10 in Bit[0]: Raw 8 in
0x04	DP CFG	0x00	RW	Bit[7]: Reserved Bit[6]: dmy_man_en To use register 0x07, dmy_line_num_man. Bit[5]: All bypass Use raw_in = raw_8_in | raw_10_in | raw_12_in to select input RAW or YUV, do not do WINC Bit[4]: RAW 10 out Bit[3]: RAW 8 out Bit[2]: Reserved Bit[1:0]: avg_sel 00: Source from isp_in (raw) 01: Source from AWB out (raw) 10: Source from OVPB27 out (Y) 11: Source from SDE out (Y)
0x05	ISP CTRL LOW	0x60	RW	Bit[7]: Reserved Bit[6]: Size auto 0: Use top register 0x14~0x16 as ISP input size 1: Use pre_isp output size as ISP input size Bit[5:4]: px_order Bit[3]: LENC flip Bit[2]: LENC mirror Bit[1]: Flip Bit[0]: Mirror

table 6-4 ISP registers (sheet 3 of 6)

address offset	register name	reset value	R/W	description
0x06	OVPB27	0x02	RW	Bit[7]: Vblank clock gate enable When vertical blanking, turn off all clocks to save power Bit[6]: EOF select Bit[5]: SOF select Bit[4]: rgb_dns_cen Bit[3]: CIP DNS cen Bit[2]: DPC cen Bit[1:0]: Shift
0x07	DMY_LINE_NUM	0x00	RW	Bit[7:0]: dmy_line_num[7:0]
0x08	R_X_ADDR_START_LOW	0x00	RW	Bit[7:0]: x_addr_start[7:0]
0x09	R_X_ADDR_START_HIGH	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: x_addr_start[11:8]
0x0A	R_Y_ADDR_START_LOW	0x00	RW	Bit[7:0]: y_addr_start[7:0]
0x0B	R_Y_ADDR_START_HIGH	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: y_addr_start[11:8]
0x0C	R_X_ADDR_END_LOW	0x7F	RW	Bit[7:0]: x_addr_end[7:0]
0x0D	R_X_ADDR_END_HIGH	0x02	RW	Bit[7:4]: Reserved Bit[3:0]: x_addr_end[11:8]
0x0E	R_Y_ADDR_END_LOW	0xDF	RW	Bit[7:0]: y_addr_end[7:0]
0x0F	R_Y_ADDR_END_HIGH	0x01	RW	Bit[7:4]: Reserved Bit[3:0]: y_addr_end[11:8] Registers 0x08~0x0F is for SMIA sensor input width = x_addr_end-x_addr_start +1 height = y_addr_end-y_addr_start +1
0x10	R_X_INC	0x11	RW	Bit[7:4]: X odd increasing Bit[3:0]: X even increasing
0x11	R_Y_INC	0x11	RW	Bit[7:4]: Y odd increasing Bit[3:0]: Y even increasing
0x12	R_X_INPUT_SIZE_LOW	0x80	RW	Bit[7:0]: x_input_size[7:0]
0x13	R_X_INPUT_SIZE_HIGH	0x02	RW	Bit[7:4]: Reserved Bit[3:0]: x_input_size[11:8]
0x14	R_Y_INPUT_SIZE_LOW	0xE0	RW	Bit[7:0]: y_input_size[7:0]

table 6-4 ISP registers (sheet 4 of 6)

address offset	register name	reset value	R/W	description
0x15	R_Y_INPUT_SIZE_HIGH	0x01	RW	Bit[7:4]: Reserved Bit[3:0]: y_input_size[11:8] Normal sensor input width = x_input_size height = y_input_size
0x16	R_BINING	0x00	RW	Bit[7:6]: Reserved Bit[5]: real_gain_man 0: Use real_gain from AEC 1: ISP all submodules use {r18[2:0],r17} as real gain Bit[4:2]: win_y_offset_adjust Bit[1]: y_bining Bit[0]: x_bining
0x17	REAL_GAIN_LOW	0x20	RW	Bit[7:0]: Real gain[7:0]
0x18	REAL_GAIN_HIGH	0x00	RW	Bit[7:6]: lenc_x_skip_man Bit[5:4]: lenc_y_skip_man Bit[3]: lenc_skip_man_en Bit[2:0]: Real gain[10:8]
0x19	SENSOR_BIAS	0x00	RW	Bit[7:0]: Sensor bias
0x1A	YUV SWAP	0x00	RW	Bit[7:6]: sensor_bias_h Bit[5]: clk_444_double Bit[4]: Vfirst Bit[3]: Drop Bit[2]: clk_double Bit[1:0]: yuv_in format
0x1B	AWB BIAS	0x03	RW	Bit[7]: awb_bias_seperate Bit[6]: awb_edge_sel Bit[5]: awb_yuv2cbcr_en Bit[4]: awb_ct_sel 0: LENC data 1: pre_data Bit[3]: Reserved Bit[2]: gain_latch Bit[1]: Clkdown Bit[0]: awb_auto
0x1C	SDE_Y_AVG	0x00	RW	Bit[7:0]: sde_y_avg
0x1D	SDE_OFFSET_MAN_EN	0x00	RW	Bit[7:1]: Reserved Bit[0]: sde_offset_man_en
0x1E	R_X_OFFSET_LOW	0x00	RW	Bit[7:0]: x_offset[7:0]
0x1F	R_X_OFFSET_HIGH	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: x_offset[11:8]
0x20	R_Y_OFFSET_LOW	0x00	RW	Bit[7:0]: y_offset[7:0]

table 6-4 ISP registers (sheet 5 of 6)

address offset	register name	reset value	R/W	description
0x21	R_Y_OFFSET_HIGH	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: y_offset[11:8]
0x22	R_HSIZE_OUT_LOW	0x80	RW	Bit[7:0]: hsize_out[7:0]
0x23	R_HSIZE_OUT_HIGH	0x02	RW	Bit[7:4]: Reserved Bit[3:0]: hsize_out[11:8]
0x24	R_VSIZE_OUT_LOW	0xE0	RW	Bit[7:0]: vsize_out[7:0]
0x25	R_VSIZE_OUT_HIGH	0x01	RW	Bit[7:4]: Reserved Bit[3:0]: vsize_out[11:8] Registers 0x1E~0x25 is for window output size
0x26	DEBUG SEL	0x80	RW	Bit[7:0]: Debug
0x27	DIGI GAIN_L	0x00	RW	Bit[7:0]: Digi gain[7:0]
0x28	DIGI GAIN_H	0x02	RW	Bit[7:0]: Digi gain[15:8]
0x29~ 0x2F	RSVD	–	–	Reserved
0x30	DUMMY LINE	0x03	R	Bit[7:0]: Dummy line Margin between last HREF and EOF
0x31	YAVG	–	R	YAVG
0x32	RSVD	–	–	Reserved
0x33	INTER_EN	0xFF	RW	Bit[7]: ISP EOF interrupt enable Bit[6]: ISP SOF interrupt enable Bit[5]: UV sum interrupt enable Bit[4]: EDR interrupt enable Bit[3]: agc_int enable Bit[2]: awb_int enable Bit[1]: AEC interrupt enable Bit[0]: YAVG interrupt enable
0x34	RSVD	–	–	Reserved
0x35	INTERRUPT_CLR	0x00	RW	Bit[7]: ISP EOF interrupt clear Bit[6]: ISP SOF interrupt clear Bit[5]: UV sum interrupt clear Bit[4]: EDR interrupt clear Bit[3]: agc_int clear Bit[2]: awb_int clear Bit[1]: AEC interrupt clear Bit[0]: YAVG interrupt clear
0x36	RSVD	–	–	Reserved

table 6-4 ISP registers (sheet 6 of 6)

address offset	register name	reset value	R/W	description
0x37	INTERRUPT_STATUS	–	R	Bit[7]: ISP EOF interrupt status Bit[6]: ISP SOF interrupt status Bit[5]: UV sum interrupt status Bit[4]: EDR interrupt status Bit[3]: agc_int status Bit[2]: awb_int_status Bit[1]: AEC interrupt status Bit[0]: YAVG interrupt status
0x38	RSVD	–	–	Reserved
0x39	GAMMA_BIAS	0x00	RW	Bit[7:0]: Raw gamma bias
0x3A	U_OFFSET	0x00	RW	Bit[7:0]: u_offset of uv_adj
0x3B	V_OFFSET	0x00	RW	Bit[7:0]: v_offset of uv_adj
0x3C~ 0x49	RSVD	–	–	Reserved
0x4A	AWB_BIAS_8_I	0x10	RW	Bit[7:0]: awb_bias_8_i
0x4B	AWB_BIAS_L_I	0x40	RW	Bit[7:0]: awb_bias_l_i
0x4C	AWB_BIAS_H_I	0x00	RW	Bit[7:2]: Reserved Bit[1:0]: awb_bias_h_i awb_bias_10b = {awb_bias_h_i,awb_bias_l_i} is for awb gain If awb_bias_seperate (register 0x1B[7]) is 1, awb_bias_8_i (register 0x4A) is for AWB 20 and AWB CT, statistics module, while awb_bias_10b is for awb gain. If awb_bias_seperate (registers 0x1B[7]) is 0, awb_bias_10b[9:2] is for AWB 20 and AWB CT, statistics module, while awb_bias_10b is for AWB gain.

6.4 pre_ISP20

- base address: 0x80162080 (ISP0_pre_ISP20)
- base address: 0x80163080 (ISP1_pre_ISP20)

table 6-5 pre_ISP20 registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x00	R_CTRL00	0x00	RW	Bit[7]: Test mode enable 0: Disable 1: Enable Bit[6]: Image rolling mode enable when test mode enable 0: Disable 1: Enable Bit[5]: When test mode enable 0: Test data 1: Test data add image data Bit[4]: Black/white or color square 0: Color 1: Black/white Bit[3:2]: Colorbar style when test mode enable 00: Standard colorbar 01: Colorbar becomes darker from top to bottom 10: Colorbar becomes darker from right to left 11: Colorbar becomes darker from bottom to top Bit[1:0]: Test mode select when test mode enable 00: Colorbar, different options for colorbar 01: Random data, different seed for random data 10: Square pattern, black white or color square pattern 11: Color chart / all black
0x01	R_CTRL01	0xC1	RW	Bit[7]: Reserved Bit[6]: Redundant pixel cut enable 0: Disable 1: Enable Bit[5]: Output date low bits set to be 0 Bit[4]: Random seed same every frame enable 0: Disable 1: Enable Bit[3:0]: Random seed
0x02	R_CTRL02	0x00	RW	Bit[7:1]: Reserved Bit[0]: Line number interrupt[8]
0x03	R_CTRL03	0x00	RW	Bit[7:0]: Line number interrupt[7:0]

table 6-5 pre_ISP20 registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x04	R_CTRL04	0x02	RW	Bit[7:2]: Reserved Bit[1:0]: Scale manual horizontal output size[9:8]
0x05	R_CTRL05	0x80	RW	Bit[7:0]: Scale manual horizontal output size[7:0]
0x06	R_CTRL06	0x01	RW	Bit[7:1]: Reserved Bit[0]: Scale manual vertical output size[8]
0x07	R_CTRL07	0xE0	RW	Bit[7:0]: Scale manual vertical output size[7:0]
0x08	R_CTRL08	0x00	RW	Bit[7:2]: Reserved Bit[1:0]: Horizontal manual offset[9:8]
0x09	R_CTRL09	0x00	RW	Bit[7:0]: Horizontal manual offset[7:0]
0x0A	R_CTRL0A	0x00	RW	Bit[7:1]: Reserved Bit[0]: Vertical manual offset[8]
0x0B	R_CTRL0B	0x00	RW	Bit[7:0]: Vertical manual offset[7:0]
0x0C	R_CTRL0C	–	R	Bit[7:2]: Reserved Bit[1:0]: Input image pixel number[9:8]
0x0D	R_CTRL0D	–	R	Bit[7:0]: Input image pixel number[7:0]
0x0E	R_CTRL0E	–	R	Bit[7:1]: Reserved Bit[0]: Input image line number[8]
0x0F	R_CTRL0F	–	R	Bit[7:0]: Input image line number[7:0]
0x10	R_CTRL10	0x30	RW	Bit[7:6]: Reserved Bit[5]: Take first pixel in same position with no mirror image enable 0: Disable 1: Enable Bit[4]: Take first pixel in same position with no flip image enable 0: Disable 1: Enable Bit[3]: Mirror option from window 0: First pixel is Gb or R with window output 1: First pixel is B or Gr with window output Bit[2]: Flip option from window 0: First line is GR with window output 1: First line is BG with window output Bit[1]: Offset manual enable 0: Disable 1: Enable Bit[0]: Scale size manual enable 0: Disable 1: Enable

table 6-5 pre_ISP20 registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x11	R_CTRL11	0x00	RW	Bit[7]: Manual clock/valid ratio enable 0: Disable 1: Enable Bit[6:4]: Dummy line number[2:0] Dummy line number low bits Bit[3]: Reduce HREF low length by half Bit[2:0]: Manual clock/valid ratio
0x12	R_CTRL12	–	R	Bit[7:5]: Reserved Bit[4:0]: HREF blank length for dummy line[12:8]
0x13	R_CTRL13	–	R	Bit[7:0]: HREF blank length for dummy line[7:0]
0x14	R_CTRL14	–	R	Bit[7:5]: Reserved Bit[4:0]: HREF length for dummy line[12:8]
0x15	R_CTRL15	–	R	Bit[7:0]: HREF length for dummy line[7:0]
0x16	R_CTRL16	–	R	Bit[7:5]: Reserved Bit[4]: Dummy error indicating signal Bit[3]: Reserved Bit[2:0]: Dummy line clock ratio output
0x17	R_CTRL17	–	R	Bit[7]: Reserved Bit[6:4]: Horizontal odd pixel increase distance output Bit[3:0]: Horizontal odd pixel increase distance output
0x18	R_CTRL18	–	R	Bit[7:6]: Reserved Bit[1:0]: Horizontal sensor offset[9:8]
0x19	R_CTRL19	–	R	Bit[7:0]: Horizontal sensor offset[7:0]
0x1A	R_CTRL1A	–	R	Bit[7:6]: Reserved Bit[1:0]: Vertical sensor offset[9:8]
0x1B	R_CTRL1B	–	R	Bit[7:0]: Vertical sensor offset[7:0]
0x1C	R_CTRL1C	–	R	Bit[7:6]: Reserved Bit[1:0]: Horizontal window offset[9:8]
0x1D	R_CTRL1D	–	R	Bit[7:0]: Horizontal window offset[7:0]
0x1E	R_CTRL1E	–	R	Bit[7:6]: Reserved Bit[1:0]: Vertical window offset[9:8]
0x1F	R_CTRL1F	–	R	Bit[7:0]: Vertical window offset[7:0]
0x20	R_CTRL20	–	R	Bit[7:6]: Reserved Bit[1:0]: Horizontal window output size[9:8]
0x21	R_CTRL21	–	R	Bit[7:0]: Horizontal window output size[7:0]

table 6-5 pre_ISP20 registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x22	R_CTRL22	–	R	Bit[7:6]: Reserved Bit[1:0]: Vertical window output size[9:8]
0x23	R_CTRL23	–	R	Bit[7:0]: Vertical window output size[7:0]
0x24	R_CTRL24	–	R	Bit[7:6]: Reserved Bit[5:4]: Horizontal skip Bit[3:2]: Reserved Bit[1:0]: Vertical skip

6.5 YAVG

- base address 0x80162100 (ISP0_YAVG)
- base address 0x80163100 (ISP1_YAVG)

table 6-6 YAVG registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	XSTART H	0x00	RW	Bit[7:5]: Reserved Bit[4:0]: xstart[12:8] Start address in horizontal
0x01	XSTART L	0x00	RW	Bit[7:0]: xstart[7:0]
0x02	YSTART H	0x00	RW	Bit[7:4]:: Reserved Bit[3:0]: ystart[11:8] Start address in vertical
0x03	YSTART L	0x00	RW	Bit[7:0]: ystart[7:0]
0x04	X_WINH	0x03	RW	Bit[7:5]: Reserved Bit[4:0]: x_win[12:8] Select whole zone width
0x05	X_WINL	0x00	RW	Bit[7:0]: x_win[7:0]
0x06	Y_WINH	0x02	RW	Bit[7:4]: Reserved Bit[3:0]: y_win[11:8] Select whole zone height
0x07	Y_WINL	0x40	RW	Bit[7:0]: y_win[7:0]
0x08	WEIGHT 01	0x11	RW	Bit[7:4]: wt1 Weight of zone01 Bit[3:0]: wt0 Weight of zone00

table 6-6 YAVG registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x09	WEIGHT 23	0x11	RW	Bit[7:4]: wt3 Weight of zone03 Bit[3:0]: wt2 Weight of zone02
0x0A	WEIGHT 45	0x11	RW	Bit[7:4]: wt5 Weight of zone11 Bit[3:0]: wt4 Weight of zone10
0x0B	WEIGHT 67	0x11	RW	Bit[7:4]: wt7 Weight of zone13 Bit[3:0]: wt6 Weight of zone12
0x0C	WEIGHT 89	0x11	RW	Bit[7:4]: wt9 Weight of zone 21 Bit[3:0]: wt8 Weight of zone20
0x0D	WEIGHT 1011	0x11	RW	Bit[7:4]: wt11 Weight of zone23 Bit[3:0]: wt10 Weight of zone22
0x0E	WEIGHT 1213	0x11	RW	Bit[7:4]: wt13 Weight of zone31 Bit[3:0]: wt12 Weight of zone30
0x0F	WEIGHT 1415	0x11	RW	Bit[7:4]: wt15 Weight of zone33 Bit[3:0]: wt14 Weight of zone32
0x10	R_AVG_CTRL10	0x02	RW	Bit[7:6]: Reserved Bit[5:4]: stat_mode Bit[3:2]: Reserved Bit[1]: avg_opt 0: Sum=(4*B+9*G*2+10*R)/8 1: Sum=B+G*2+R Bit[7:1]: Reserved Bit[0]: avg_man
0x11	WT_SUM	–	R	Bit[7:0]: wt_sum
0x12	AVG_SCCB_DONE	–	R	Bit[7:1]: reserved Bit[0]: avg_sccb_done
0x13	AVG	–	R	Bit[7:0]: avg

6.6 LENC40

- base address: 0x80162180 (ISP0_LENC40)
- base address: 0x80163180 (ISP0_LENC40)

table 6-7 LENC40 registers (sheet 1 of 5)

address	register name	default value	R/W	description
0x00	LENC_G00	0x10	RW	(Row 0, Column 0) G Channel Parameter for Gain Computation
0x01	LENC_G01	0x10	RW	(Row 0, Column 1) G Channel Parameter for Gain Computation
0x02	LENC_G02	0x10	RW	(Row 0, Column 2) G Channel Parameter for Gain Computation
0x03	LENC_G03	0x10	RW	(Row 0, Column 3) G Channel Parameter for Gain Computation
0x04	LENC_G04	0x10	RW	(Row 0, Column 4) G Channel Parameter for Gain Computation
0x05	LENC_G05	0x10	RW	(Row 0, Column 5) G Channel Parameter for Gain Computation
0x06	LENC_G10	0x10	RW	(Row 1, Column 0) G Channel Parameter for Gain Computation
0x07	LENC_G11	0x08	RW	(Row 1, Column 1) G Channel Parameter for Gain Computation
0x08	LENC_G12	0x08	RW	(Row 1, Column 2) G Channel Parameter for Gain Computation
0x09	LENC_G13	0x08	RW	(Row 1, Column 3) G Channel Parameter for Gain Computation
0x0A	LENC_G14	0x08	RW	(Row 1, Column 4) G Channel Parameter for Gain Computation
0x0B	LENC_G15	0x10	RW	(Row 1, Column 5) G Channel Parameter for Gain Computation
0x0C	LENC_G20	0x10	RW	(Row 2, Column 0) G Channel Parameter for Gain Computation
0x0D	LENC_G21	0x08	RW	(Row 2, Column 1) G Channel Parameter for Gain Computation
0x0E	LENC_G22	0x00	RW	(Row 2, Column 2) G Channel Parameter for Gain Computation
0x0F	LENC_G23	0x00	RW	(Row 2, Column 3) G Channel Parameter for Gain Computation

table 6-7 LENC40 registers (sheet 2 of 5)

address	register name	default value	R/W	description
0x10	LENC_G24	0x08	RW	(Row 2, Column 4) G Channel Parameter for Gain Computation
0x11	LENC_G25	0x10	RW	(Row 2, Column 5) G Channel Parameter for Gain Computation
0x12	LENC_G30	0x10	RW	(Row 3, Column 0) G Channel Parameter for Gain Computation
0x13	LENC_G31	0x08	RW	(Row 3, Column 1) G Channel Parameter for Gain Computation
0x14	LENC_G32	0x00	RW	(Row 3, Column 2) G Channel Parameter for Gain Computation
0x15	LENC_G33	0x00	RW	(Row 3, Column 3) G Channel Parameter for Gain Computation
0x16	LENC_G34	0x08	RW	(Row 3, Column 4) G Channel Parameter for Gain Computation
0x17	LENC_G35	0x10	RW	(Row 3, Column 5) G Channel Parameter for Gain Computation
0x18	LENC_G40	0x10	RW	(Row 4, Column 0) G Channel Parameter for Gain Computation
0x19	LENC_G41	0x08	RW	(Row 4, Column 1) G Channel Parameter for Gain Computation
0x1A	LENC_G42	0x08	RW	(Row 4, Column 2) G Channel Parameter for Gain Computation
0x1B	LENC_G43	0x08	RW	(Row 4, Column 3) G Channel Parameter for Gain Computation
0x1C	LENC_G44	0x08	RW	(Row 4, Column 4) G Channel Parameter for Gain Computation
0x1D	LENC_G45	0x10	RW	(Row 4, Column 5) G Channel Parameter for Gain Computation
0x1E	LENC_G50	0x10	RW	(Row 5, Column 0) G Channel Parameter for Gain Computation
0x1F	LENC_G51	0x10	RW	(Row 5, Column 1) G Channel Parameter for Gain Computation
0x20	LENC_G52	0x10	RW	(Row 5, Column 2) G Channel Parameter for Gain Computation
0x21	LENC_G53	0x10	RW	(Row 5, Column 3) G Channel Parameter for Gain Computation
0x22	LENC_G54	0x10	RW	(Row 5, Column 4) G Channel Parameter for Gain Computation

table 6-7 LENC40 registers (sheet 3 of 5)

address	register name	default value	R/W	description
0x23	LENC_G55	0x10	RW	(Row 5, Column 5) G Channel Parameter for Gain Computation
0x24	LENC_BR00	0xAA	RW	(Row 0, Column 0) B, R Channel Parameter for Gain Computation
0x25	LENC_BR01	0xAA	RW	(Row 0, Column 1) B, R Channel Parameter for Gain Computation
0x26	LENC_BR02	0xAA	RW	(Row 0, Column 2) B, R Channel Parameter for Gain Computation
0x27	LENC_BR03	0xAA	RW	(Row 0, Column 3) B, R Channel Parameter for Gain Computation
0x28	LENC_BR04	0xAA	RW	(Row 0, Column 4) B, R Channel Parameter for Gain Computation
0x29	LENC_BR10	0xAA	RW	(Row 1, Column 0) B, R Channel Parameter for Gain Computation
0x2A	LENC_BR11	0x99	RW	(Row 1, Column 1) B, R Channel Parameter for Gain Computation
0x2B	LENC_BR12	0x99	RW	(Row 1, Column 2) B, R Channel Parameter for Gain Computation
0x2C	LENC_BR13	0x99	RW	(Row 1, Column 3) B, R Channel Parameter for Gain Computation
0x2D	LENC_BR14	0xAA	RW	(Row 1, Column 4) B, R Channel Parameter for Gain Computation
0x2E	LENC_BR20	0xAA	RW	(Row 2, Column 0) B, R Channel Parameter for Gain Computation
0x2F	LENC_BR21	0x99	RW	(Row 2, Column 1) B, R Channel Parameter for Gain Computation
0x30	LENC_BR22	0x88	RW	(Row 2, Column 2) B, R Channel Parameter for Gain Computation
0x31	LENC_BR23	0x99	RW	(Row 2, Column 3) B, R Channel Parameter for Gain Computation
0x32	LENC_BR24	0xAA	RW	(Row 2, Column 4) B, R Channel Parameter for Gain Computation
0x33	LENC_BR30	0xAA	RW	(Row 3, Column 0) B, R Channel Parameter for Gain Computation
0x34	LENC_BR31	0x99	RW	(Row 3, Column 1) B, R Channel Parameter for Gain Computation
0x35	LENC_BR32	0x99	RW	(Row 3, Column 2) B, R Channel Parameter for Gain Computation

table 6-7 LENC40 registers (sheet 4 of 5)

address	register name	default value	R/W	description
0x36	LENC_BR33	0x99	RW	(Row 3, Column 3) B, R Channel Parameter for Gain Computation
0x37	LENC_BR34	0xAA	RW	(Row 3, Column 4) B, R Channel Parameter for Gain Computation
0x38	LENC_BR40	0xAA	RW	(Row 4, Column 0) B, R Channel Parameter for Gain Computation
0x39	LENC_BR41	0xAA	RW	(Row 4, Column 1) B, R Channel Parameter for Gain Computation
0x3A	LENC_BR42	0xAA	RW	(Row 4, Column 2) B, R Channel Parameter for Gain Computation
0x3B	LENC_BR43	0xAA	RW	(Row 4, Column 3) B, R Channel Parameter for Gain Computation
0x3C	LENC_BR44	0xAA	RW	(Row 4, Column 4) B, R Channel Parameter for Gain Computation
0x3D	LENC_BR_OFFSET	0x88	RW	Offset for Each B, R Channel Parameter
0x3E	LENC_MAXGAIN	0x40	RW	Maximum Gain in Auto q Computation
0x3F	LENC_MINGAIN	0x20	RW	Minimum Gain in Auto q Computation
0x40	LENC_MINQ	0x18	RW	Minimum q in Auto q Computation
0x41	LENC_CTRL41	0x0D	RW	Bit[7:4]: Reserved Bit[3]: addblc Enable adding back BLC Bit[2]: blc_en Enable BLC Bit[1]: Reserved Bit[0]: autoq_en Enable auto q computation
0x42	LENC_BR_HSCALE	0x00	RW	Horizontal Scaling Factor for B, R Channel
0x43	LENC_BR_HSCALE	0x99	RW	Horizontal Scaling Factor for B, R Channel
0x44	LENC_BR_VSCALE	0x00	RW	Vertical Scaling Factor for B, R Channel
0x45	LENC_BR_VSCALE	0xCC	RW	Vertical Scaling Factor for B, R Channel
0x46	LENC_G_HSCALE	0x00	RW	Horizontal Scaling Factor for G Channel
0x47	LENC_G_HSCALE	0xCC	RW	Horizontal Scaling Factor for G Channel
0x48	LENC_G_VSCALE	0x00	RW	Vertical Scaling Factor for B, R Channel
0x49	LENC_G_VSCALE	0x88	RW	Vertical Scaling Factor for B, R Channel
0x50	LENC_X_OFFSET	–	R	Starting Pixel Coordinate in X Direction

table 6-7 LENC40 registers (sheet 5 of 5)

address	register name	default value	R/W	description
0x51	LENC_X_OFFSET	–	R	Starting Pixel Coordinate in X Direction
0x52	LENC_Y_OFFSET	–	R	Starting Pixel Coordinate in Y Direction
0x53	LENC_Y_OFFSET	–	R	Starting Pixel Coordinate in Y Direction
0x54	LENC_RO54	–	R	Bit[7:6]: Reserved Bit[5]: Flip Flip mode in sensor Bit[4]: Mirror Mirror mode in sensor Bit[3:2]: y_skip Skip factor in Y direction Bit[1:0]: x_skip Skip factor in X direction
0x55	LENC_RO55	–	R	Bit[7:4]: Reserved Bit[3]: overflow_gh Indicates whether horizontal G position is overflow or not Bit[2]: overflow_brh Indicates whether horizontal B/R position is overflow or not Bit[1]: overflow_gv Indicates whether vertical G position is overflow or not Bit[0]: overflow_brv Indicates whether vertical B/R position is overflow or not
0x56	LENC_M_NQ	–	R	Bit[7:0]: Calculated q value

6.7 AWB20

- base address: 0x80162200 (set AWB auto = 1 (AWB auto mode), register 0x201B[0] = 1, ISP0_AWB20)
- base address: 0x80163200 (set AWB auto = 1 (AWB auto mode), register 0x301B[0] = 1, ISP1_AWB20)

table 6-8 AWB20 registers (sheet 1 of 5)

address offset	register name	reset value	R/W	description
0x00	R_AWB_CTRL0	0xFF	RW	Bit[7:0]: awb_b_block
0x01	R_AWB_CTRL1	0x58	RW	Bit[7:6]: step_local Bit[5:4]: step_fast Bit[3]: slop_8x Bit[2]: slop_4x Bit[1]: one_zone Bit[0]: avg_all
0x02	R_AWB_CTRL2	0x11	RW	Bit[7:4]: max_local_cnt Bit[3:0]: max_fast_cnt
0x03	R_AWB_CTRL3	0x90	RW	Bit[7]: awb_simple_en 0: awb_advance 1: awb_simple Bit[6]: yuv_en 1: Simple YUV enable Bit[5]: awb_preset Bit[4]: awb_simf Bit[3:2]: awb_win Bit[0]: Reserved
0x04	R_AWB_CTRL4	0x25	RW	Bit[7:6]: count_area_sel Bit[5]: g_en Bit[4:2]: count_limit_ctrl Bit[1:0]: cnt_th
0x05	R_AWB_CTRL5	0x24	RW	Bit[7:4]: stable_range_us Unstable stable range Bit[3:0]: stable_range_s Stable range
0x06	R_AWB_CTRL6	0x10	RW	Bit[7:0]: awb_s
0x07	R_AWB_CTRL7	0x10	RW	Bit[7:0]: awb_ec
0x08	R_AWB_CTRL8	0x10	RW	Bit[7:0]: awb_fc
0x09	R_AWB_CTRL9	0x40	RW	Bit[7:0]: awb_x0
0x0A	R_AWB_CTRL10	0x40	RW	Bit[7:0]: awb_y0
0x0B	R_AWB_CTRL11	0x00	RW	Bit[7:0]: awb_kx
0x0C	R_AWB_CTRL12	0x00	RW	Bit[7:0]: awb_ky
0x0D	R_AWB_CTRL13	0x00	RW	Bit[7:0]: day_limit

table 6-8 AWB20 registers (sheet 2 of 5)

address offset	register name	reset value	R/W	description
0x0E	R_AWB_CTRL14	0x00	RW	Bit[7:0]: a_limit
0x0F	R_AWB_CTRL15	0x20	RW	Bit[7:0]: day_split
0x10	R_AWB_CTRL16	0x20	RW	Bit[7:0]: a_split
0x11	R_AWB_CTRL17	0xFF	RW	Bit[7:0]: awb_top_limit
0x12	R_AWB_CTRL18	0x00	RW	Bit[7:0]: awb_bot_limit
0x13	R_AWB_CTRL19	0xF0	RW	Bit[7:0]: red_limit
0x14	R_AWB_CTRL20	0xF0	RW	Bit[7:0]: green_limit
0x15	R_AWB_CTRL21	0xF0	RW	Bit[7:0]: blue_limit
0x16	R_AWB_CTRL22	0x03	RW	Bit[7] awb_gain_m Bit[6] Reserved Bit[5] awb_freeze Bit[4]: Reserved Bit[3:2]: awb_sim_sel 00: awb_simple from after awb_gain 01: awb_simple from after raw_gma 10: awb_simple from after HDR 11: awb_simple from after awb_gain Bit[1]: fast_enable Bit[0]: awb_bias_stat
0x17	R_AWB_CTRL23	0x02	RW	Bit[7:0]: local_limit
0x18	R_AWB_CTRL24	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: awb_r_gain_m[11:8]
0x19	R_AWB_CTRL25	0x00	RW	Bit[7:0]: awb_r_gain_m[7:0]
0x1A	R_AWB_CTRL26	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: awb_g_gain_m[11:8]
0x1B	R_AWB_CTRL27	0x00	RW	Bit[7:0]: awb_g_gain_m[7:0]
0x1C	R_AWB_CTRL28	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: awb_b_gain_m[11:8]
0x1D	R_AWB_CTRL29	0x00	RW	Bit[7:0]: awb_b_gain_m[7:0]

table 6-8 AWB20 registers (sheet 3 of 5)

address offset	register name	reset value	R/W	description
0x1E	R_AWB_CTRL30	0x00	RW	Bit[7:4]: Reserved Bit[3]: local_limit_sel Bit[2]: simple_stable_sel Bit[1:0]: RO register read select 00: Read in awb_done interrupt 01: Read in vsync_i interrupt, 1x: Anytime but value is not constant
0x1F	R_AWB_CTRL31	–	R	Bit[7:4]: Reserved Bit[3:0]: current_r_setting_i[11:8]
0x20	R_AWB_CTRL32	–	R	Bit[7:0]: current_r_setting_i[7:0]
0x21	R_AWB_CTRL33	–	R	Bit[7:4]: Reserved Bit[3:0]: current_g_setting_i[11:8]
0x22	R_AWB_CTRL34	–	R	Bit[7:0]: current_g_setting_i[7:0]
0x23	R_AWB_CTRL35	–	R	Bit[7:4]: Reserved Bit[3:0]: current_b_setting_i[11:8]
0x24	R_AWB_CTRL36	–	R	Bit[7:0]: current_b_setting_i[7:0]
0x25	R_AWB_CTRL37	–	R	Bit[7:0]: ar_i[9:2]
0x26	R_AWB_CTRL38	–	R	Bit[7:0]: ag_i[9:2]
0x27	R_AWB_CTRL39	–	R	Bit[7:0]: ab_i[9:2]
0x28	RSVD	–	–	Reserved
0x29	R_AWB_CTRL41	–	R	Bit[7:0]: r_sum_i[31:24]
0x2A	R_AWB_CTRL42	–	R	Bit[7:0]: r_sum_i[23:16]
0x2B	R_AWB_CTRL43	–	R	Bit[7:0]: r_sum_i[15:8]
0x2C	R_AWB_CTRL44	–	R	Bit[7:0]: r_sum_i[7:0]
0x2D~ 0x2E	RSVD	–	–	Reserved
0x2F	R_AWB_CTRL47	–	R	Bit[7:0]: g_sum_i[31:24]
0x30	R_AWB_CTRL48	–	R	Bit[7:0]: g_sum_i[23:16]
0x31	R_AWB_CTRL49	–	R	Bit[7:0]: g_sum_i[15:8]
0x32	R_AWB_CTRL50	–	R	Bit[7:0]: g_sum_i[7:0]
0x33~ 0x34	RSVD	–	–	Reserved
0x35	R_AWB_CTRL53	–	R	Bit[7:0]: b_sum_i[31:24]
0x36	R_AWB_CTRL54	–	R	Bit[7:0]: b_sum_i[23:16]

table 6-8 AWB20 registers (sheet 4 of 5)

address offset	register name	reset value	R/W	description
0x37	R_AWB_CTRL55	–	R	Bit[7:0]: b_sum_i[15:8]
0x38	R_AWB_CTRL56	–	R	Bit[7:0]: b_sum_i[7:0]
0x39~ 0x3A	RSVD	–	–	Reserved
0x3B	R_AWB_CTRL59	–	R	Bit[7]: Reserved Bit[6:0]: a_cnt_i[22:16]
0x3C	R_AWB_CTRL60	–	R	Bit[7:0]: a_cnt_i[15:8]
0x3D	R_AWB_CTRL61	–	R	Bit[7:0]: a_cnt_i[7:0]
0x3E	RSVD	–	–	Reserved
0x3F	R_AWB_CTRL63	–	R	Bit[7]: Reserved Bit[6:0]: cwf_cnt_i[22:16]
0x40	R_AWB_CTRL64	–	R	Bit[7:0]: cwf_cnt_i[15:8]
0x41	R_AWB_CTRL65	–	R	Bit[7:0]: cwf_cnt_i[7:0]
0x42	RSVD	–	–	Reserved
0x43	R_AWB_CTRL67	–	R	Bit[7]: Reserved Bit[6:0]: day_cnt_i[22:16]
0x44	R_AWB_CTRL68	–	R	Bit[7:0]: day_cnt_i[15:8]
0x45	R_AWB_CTRL69	–	R	Bit[7:0]: day_cnt_i[7:0]
0x46	RSVD	–	–	Reserved
0x47	R_AWB_CTRL71	–	R	Bit[7]: Reserved Bit[6:0]: all_cnt_i[22:16]
0x48	R_AWB_CTRL72	–	R	Bit[7:0]: all_cnt_i[15:8]
0x49	R_AWB_CTRL73	–	R	Bit[7:0]: all_cnt_i[7:0]
0x4A	R_AWB_CTRL74	–	R	Bit[7:4]: Reserved Bit[3:0]: r_center_i[11:8]
0x4B	R_AWB_CTRL75	–	R	Bit[7:0]: r_center_i[7:0]
0x4C	R_AWB_CTRL76	–	R	Bit[7:4]: Reserved Bit[3:0]: g_center_i[11:8]
0x4D	R_AWB_CTRL77	–	R	Bit[7:0]: g_center_i[7:0]
0x4E	R_AWB_CTRL78	–	R	Bit[7:4]: Reserved Bit[3:0]: b_center_i[11:8]
0x4F	R_AWB_CTRL79	–	R	Bit[7:0]: b_center_i[7:0]

table 6-8 AWB20 registers (sheet 5 of 5)

address offset	register name	reset value	R/W	description
0x50	R_AWB_CTRL80	–	R	Bit[7:6]: Reserved Bit[5]: red_large_i Bit[4]: g_large_i Bit[3]: b_large_i Bit[2:1]: cur_type_i Bit[0]: Reserved

6.8 AWB_CT

- base address: 0x80162200 (set AWB auto = 0 (AWB auto mode disable), register 0x201B[0] = 0, ISP0_AWBCT)
- base address: 0x80163200 (set AWB auto = 0 (AWB auto mode disable), register 0x301B[0] = 0, ISP1_AWBCT)

table 6-9 AWB_CT registers (sheet 1 of 3)

address offset	register name	reset value	R/W	description
0x00	R_AWB_CTRL0	0xFF	RW	Bit[7:0]: awb_b_block l
0x01	R_AWB_CTRL1	0x08	RW	Bit[7:4]: Reserved Bit[3]: slop_8x l Bit[2]: slop_4x l Bit[1]: reserved Bit[0]: avg_all
0x02	RSVD	–	–	Reserved
0x03	R_AWB_CTRL3	0x90	RW	Bit[7]: slop_8x s Bit[6]: slop_4x s Bit[5:4]: Reserved Bit[3:2]: awb_win Bit[1:0]: Reserved
0x04~ 0x05	RSVD	–	–	Reserved
0x06	R_AWB_CTRL6	0x10	RW	Bit[7:0]: awb_s
0x07	R_AWB_CTRL7	0x10	RW	Bit[7:0]: awb_ec
0x08	R_AWB_CTRL8	0x10	RW	Bit[7:0]: awb_fc
0x09	R_AWB_CTRL9	0x40	RW	Bit[7:0]: awb_x0
0x0A	R_AWB_CTRL10	0x40	RW	Bit[7:0]: awb_y0
0x0B	R_AWB_CTRL11	0x00	RW	Bit[7:0]: awb_kx
0x0C	R_AWB_CTRL12	0x00	RW	Bit[7:0]: awb_ky

table 6-9 AWB_CT registers (sheet 2 of 3)

address offset	register name	reset value	R/W	description
0x0D	R_AWB_CTRL13	0x00	RW	Bit[7:0]: day_limit
0x0E	R_AWB_CTRL14	0x00	RW	Bit[7:0]: a_limit
0x0F	R_AWB_CTRL15	0x20	RW	Bit[7:0]: day_split
0x10	R_AWB_CTRL16	0x20	RW	Bit[7:0]: a_split
0x11	R_AWB_CTRL17	0xFF	RW	Bit[7:0]: awb_top_limit
0x12	R_AWB_CTRL18	0x00	RW	Bit[7:0]: awb_bot_limit
0x13	R_AWB_CTRL19	0xF0	RW	Bit[7:0]: red_limit
0x14	R_AWB_CTRL20	0xF0	RW	Bit[7:0]: green_limit
0x15	R_AWB_CTRL21	0xF0	RW	Bit[7:0]: blue_limit
0x16	R_AWB_CTRL22	0x03	RW	Bit[7:6]: Reserved Bit[5]: awb_freeze Bit[4:2]: Reserved Bit[1]: fast_enable Bit[0]: awb_bias_stat
0x17~ 0x1D	RSVD	–	–	Reserved
0x1E	R_AWB_CTRL30	0x00	RW	Bit[7:0]: awb_b_block s
0x1F	R_AWB_CTRL31	0x00	RW	Bit[7:0]: awb_s s
0x20	R_AWB_CTRL32	0x00	RW	Bit[7:0]: awb_ec s
0x21	R_AWB_CTRL33	0x00	RW	Bit[7:0]: awb_fc s
0x22	R_AWB_CTRL34	0x00	RW	Bit[7:0]: awb_x0 s
0x23	R_AWB_CTRL35	0x00	RW	Bit[7:0]: awb_y0
0x24	R_AWB_CTRL36	0x00	RW	Bit[7:0]: awb_kx
0x25	R_AWB_CTRL37	0x00	RW	Bit[7:0]: awb_ky
0x26	R_AWB_CTRL38	0x00	RW	Bit[7:0]: day_limit
0x27	R_AWB_CTRL39	0x00	RW	Bit[7:0]: a_limit
0x28	R_AWB_CTRL3A	0x00	RW	Bit[7:0]: day_split
0x29	R_AWB_CTRL3B	0x00	RW	Bit[7:0]: a_split
0x2A	R_AWB_CTRL3C	0x00	RW	Bit[7:0]: awb_top_limit
0x2B	R_AWB_CTRL3D	0x00	RW	Bit[7:0]: awb_bot_limit
0x2C	R_AWB_CTRL3E	0x00	RW	Bit[7:0]: red_limit
0x2D	R_AWB_CTRL3F	0x00	RW	Bit[7:0]: green_limit

table 6-9 AWB_CT registers (sheet 3 of 3)

address offset	register name	reset value	R/W	description
0x2E	R_AWB_CTRL40	0x00	RW	Bit[7:0]: blue_limit
0x2F	R_AWB_CTRL41	0x00	RW	Bit[7]: bsum_l_fix Bit[6]: gsum_l_fix Bit[5]: rsum_l_fix Bit[4]: bsum_s_fix Bit[3]: gsum_s_fix Bit[2]: rsum_s_fix Bit[1]: allcnt_l_fix Bit[0]: allcnt_s_fix
0x30	R_AWB_CTRL42	0x00	RW	Bit[7:6]: Reserved Bit[5]: fix_whole Bit[4]: fix_eof Bit[3:0]: fix_value

6.9 AWB_CT read only

- base address: 0x80162280 (ISP0_AWBCT Read Only)
- base address: 0x80163280 (ISP1_AWBCT Read Only)

table 6-10 AWB_CT read only registers (sheet 1 of 5)

address offset	register name	reset value	R/W	description
0x00	B SUM L	–	R	Bit[7:0]: b_sum_l_i[31:24]
0x01	B SUM L	–	R	Bit[7:0]: b_sum_l_i[23:16]
0x02	B SUM L	–	R	Bit[7:0]: b_sum_l_i[15:8]
0x03	B SUM L	–	R	Bit[7:0]: b_sum_l_i[7:0]
0x04	G SUM L	–	R	Bit[7:0]: g_sum_l_i[31:24]
0x05	G SUM L	–	R	Bit[7:0]: g_sum_l_i[23:16]
0x06	G SUM L	–	R	Bit[7:0]: g_sum_l_i[15:8]
0x07	G SUM L	–	R	Bit[7:0]: g_sum_l_i[7:0]
0x08	R SUM L	–	R	Bit[7:0]: r_sum_l_i[31:24]
0x09	R SUM L	–	R	Bit[7:0]: r_sum_l_i[23:16]
0x0A	R SUM L	–	R	Bit[7:0]: r_sum_l_i[15:8]
0x0B	R SUM L	–	R	Bit[7:0]: r_sum_l_i[7:0]
0x0C	B SUM S	–	R	Bit[7:0]: b_sum_s_i[31:24]

table 6-10 AWB_CT read only registers (sheet 2 of 5)

address offset	register name	reset value	R/W	description
0x0D	B SUM S	–	R	Bit[7:0]: b_sum_s_i[23:16]
0x0E	B SUM S	–	R	Bit[7:0]: b_sum_s_i[15:8]
0x0F	B SUM S	–	R	Bit[7:0]: b_sum_s_i[7:0]
0x10	G SUM S	–	R	Bit[7:0]: g_sum_s_i[31:24]
0x11	G SUM S	–	R	Bit[7:0]: g_sum_s_i[23:16]
0x12	G SUM S	–	R	Bit[7:0]: g_sum_s_i[15:8]
0x13	G SUM S	–	R	Bit[7:0]: g_sum_s_i[7:0]
0x14	R SUM S	–	R	Bit[7:0]: r_sum_s_i[31:24]
0x15	R SUM S	–	R	Bit[7:0]: r_sum_s_i[23:16]
0x16	R SUM S	–	R	Bit[7:0]: r_sum_s_i[15:8]
0x17	R SUM S	–	R	Bit[7:0]: r_sum_s_i[7:0]
0x18	RSVD	–	–	Reserved
0x19	ALL CNT L	–	R	Bit[7:6]: Reserved Bit[5:0]: all_cnt_l_i[21:16]
0x1A	ALL CNT L	–	R	Bit[7:0]: all_cnt_l_i[15:8]
0x1B	ALL CNT L	–	R	Bit[7:0]: all_cnt_l_i[7:0]
0x1C	RSVD	–	–	Reserved
0x1D	ALL CNT S	–	R	Bit[7:6]: Reserved Bit[5:0]: all_cnt_s_i[21:16]
0x1E	ALL CNT S	–	R	Bit[7:0]: all_cnt_s_i[15:8]
0x1F	ALL CNT S	–	R	Bit[7:0]: all_cnt_s_i[7:0]
0x20	B_SUM_A_L	–	R	Bit[7:0]: b_sum_a_l_i[31:24]
0x21	B_SUM_A_L	–	R	Bit[7:0]: b_sum_a_l_i[23:16]
0x22	B_SUM_A_L	–	R	Bit[7:0]: b_sum_a_l_i[15:8]
0x23	B_SUM_A_L	–	R	Bit[7:0]: b_sum_a_l_i[7:0]
0x24	G_SUM_A_L	–	R	Bit[7:0]: g_sum_a_l_i[31:24]
0x25	G_SUM_A_L	–	R	Bit[7:0]: g_sum_a_l_i[23:16]
0x26	G_SUM_A_L	–	R	Bit[7:0]: g_sum_a_l_i[15:8]
0x27	G_SUM_A_L	–	R	Bit[7:0]: g_sum_a_l_i[7:0]
0x28	R_SUM_A_L	–	R	Bit[7:0]: r_sum_a_l_i[31:24]
0x29	R_SUM_A_L	–	R	Bit[7:0]: r_sum_a_l_i[23:16]

table 6-10 AWB_CT read only registers (sheet 3 of 5)

address offset	register name	reset value	R/W	description
0x2A	R_SUM_A_L	–	R	Bit[7:0]: r_sum_a_l_i[15:8]
0x2B	R_SUM_A_L	–	R	Bit[7:0]: r_sum_a_l_i[7:0]
0x2C	B_SUM_A_S	–	R	Bit[7:0]: b_sum_a_s_i[31:24]
0x2D	B_SUM_A_S	–	R	Bit[7:0]: b_sum_a_s_i[23:16]
0x2E	B_SUM_A_S	–	R	Bit[7:0]: b_sum_a_s_i[15:8]
0x2F	B_SUM_A_S	–	R	Bit[7:0]: b_sum_a_s_i[7:0]
0x30	G_SUM_A_S	–	R	Bit[7:0]: g_sum_a_s_i[31:24]
0x31	G_SUM_A_S	–	R	Bit[7:0]: g_sum_a_s_i[23:16]
0x32	G_SUM_A_S	–	R	Bit[7:0]: g_sum_a_l_i[15:8]
0x33	G_SUM_A_S	–	R	Bit[7:0]: g_sum_a_l_i[7:0]
0x34	R_SUM_A_L	–	R	Bit[7:0]: r_sum_a_l_i[31:24]
0x35	R_SUM_A_L	–	R	Bit[7:0]: r_sum_a_l_i[23:16]
0x36	R_SUM_A_L	–	R	Bit[7:0]: r_sum_a_l_i[15:8]
0x37	R_SUM_A_L	–	R	Bit[7:0]: r_sum_a_l_i[7:0]
0x38	RSVD	–	–	Reserved
0x39	CNT_A_L	–	R	Bit[7:6]: Reserved Bit[5:0]: cnt_a_l_i[21:16]
0x3A	CNT_A_L	–	R	Bit[7:0]: cnt_a_l_i[15:8]
0x3B	CNT_A_L	–	R	Bit[7:0]: cnt_a_l_i[7:0]
0x3C	RSVD	–	–	Reserved
0x3D	CNT_A_S	–	R	Bit[7:6]: Reserved Bit[5:0]: cnt_a_s_i[21:16]
0x3E	CNT_A_S	–	R	Bit[7:0]: cnt_a_s_i[15:8]
0x3F	CNT_A_S	–	R	cnt_a_s_i[7:0]Bit[7:0]:
0x40	B_SUM_DAY_L	–	R	Bit[7:0]: b_sum_day_l_i[31:24]
0x41	B_SUM_DAY_L	–	R	Bit[7:0]: b_sum_day_l_i[23:16]
0x42	B_SUM_DAY_L	–	R	Bit[7:0]: b_sum_day_l_i[15:8]
0x43	B_SUM_DAY_L	–	R	Bit[7:0]: b_sum_day_l_i[7:0]
0x44	G_SUM_DAY_L	–	R	Bit[7:0]: g_sum_day_l_i[31:24]
0x45	G_SUM_DAY_L	–	R	Bit[7:0]: g_sum_day_l_i[23:16]
0x46	G_SUM_DAY_L	–	R	Bit[7:0]: g_sum_day_l_i[15:8]

table 6-10 AWB_CT read only registers (sheet 4 of 5)

address offset	register name	reset value	R/W	description
0x47	G_SUM_DAY_L	–	R	Bit[7:0]: g_sum_day_l_i[7:0]
0x48	R_SUM_DAY_L	–	R	Bit[7:0]: r_sum_day_l_i[31:24]
0x49	R_SUM_DAY_L	–	R	Bit[7:0]: r_sum_day_l_i[23:16]
0x4A	R_SUM_DAY_L	–	R	Bit[7:0]: r_sum_day_l_i[15:8]
0x4B	R_SUM_DAY_L	–	R	Bit[7:0]: r_sum_day_l_i[7:0]
0x4C	B_SUM_DAY_S	–	R	Bit[7:0]: b_sum_day_s_i[31:24]
0x4D	B_SUM_DAY_S	–	R	Bit[7:0]: b_sum_day_s_i[23:16]
0x4E	B_SUM_DAY_S	–	R	Bit[7:0]: b_sum_day_s_i[15:8]
0x4F	B_SUM_DAY_S	–	R	Bit[7:0]: b_sum_day_s_i[7:0]
0x50	G_SUM_DAY_S	–	R	Bit[7:0]: g_sum_day_s_i[31:24]
0x51	G_SUM_DAY_S	–	R	Bit[7:0]: g_sum_day_s_i[23:16]
0x52	G_SUM_DAY_S	–	R	Bit[7:0]: g_sum_day_s_i[15:8]
0x53	G_SUM_DAY_S	–	R	Bit[7:0]: g_sum_day_s_i[7:0]
0x54	R_SUM_DAY_S	–	R	Bit[7:0]: r_sum_day_s_i[31:24]
0x55	R_SUM_DAY_S	–	R	Bit[7:0]: r_sum_day_s_i[23:16]
0x56	R_SUM_DAY_S	–	R	Bit[7:0]: r_sum_day_s_i[15:8]
0x57	R_SUM_DAY_S	–	R	Bit[7:0]: r_sum_day_s_i[7:0]
0x58	RSVD	–	–	Reserved
0x59	ALL_DAY_CNT_L	–	R	Bit[7:6]: Reserved Bit[5:0]: all_day_cnt_l_i[21:16]
0x5A	ALL_DAY_CNT_L	–	R	Bit[7:0]: all_day_cnt_l_i[15:8]
0x5B	ALL_DAY_CNT_L	–	R	Bit[7:0]: all_day_cnt_l_i[7:0]
0x5C	RSVD	–	–	Reserved
0x5D	ALL_DAY_CNT_S	–	R	Bit[7:6]: Reserved Bit[5:0]: all_day_cnt_s_i[21:16]
0x5E	ALL_DAY_CNT_S	–	R	Bit[7:0]: all_day_cnt_s_i[15:8]
0x5F	ALL_DAY_CNT_S	–	R	Bit[7:0]: all_day_cnt_s_i[7:0]
0x60	B_SUM_CWF_L	–	R	Bit[7:0]: b_sum_cwf_l_i[31:24]
0x61	B_SUM_CWF_L	–	R	Bit[7:0]: b_sum_cwf_l_i[23:16]
0x62	B_SUM_CWF_L	–	R	Bit[7:0]: b_sum_cwf_l_i[15:8]
0x63	B_SUM_CWF_L	–	R	Bit[7:0]: b_sum_cwf_l_i[7:0]

table 6-10 AWB_CT read only registers (sheet 5 of 5)

address offset	register name	reset value	R/W	description
0x64	G_SUM_CWF_L	–	R	Bit[7:0]: g_sum_cwf_l_i[31:24]
0x65	G_SUM_CWF_L	–	R	Bit[7:0]: g_sum_cwf_l_i[23:16]
0x66	G_SUM_CWF_L	–	R	Bit[7:0]: g_sum_cwf_l_i[15:8]
0x67	G_SUM_CWF_L	–	R	Bit[7:0]: g_sum_cwf_l_i[7:0]
0x68	R_SUM_CWF_L	–	R	Bit[7:0]: r_sum_cwf_l_i[31:24]
0x69	R_SUM_CWF_L	–	R	Bit[7:0]: r_sum_cwf_l_i[23:16]
0x6A	R_SUM_CWF_L	–	R	Bit[7:0]: r_sum_cwf_l_i[15:8]
0x6B	R_SUM_CWF_L	–	R	Bit[7:0]: r_sum_cwf_l_i[7:0]
0x6C	B_SUM_CWF_S	–	R	Bit[7:0]: b_sum_cwf_s_i[31:24]
0x6D	B_SUM_CWF_S	–	R	Bit[7:0]: b_sum_cwf_s_i[23:16]
0x6E	B_SUM_CWF_S	–	R	Bit[7:0]: b_sum_cwf_s_i[15:8]
0x6F	B_SUM_CWF_S	–	R	Bit[7:0]: b_sum_cwf_s_i[7:0]
0x70	G_SUM_CWF_S	–	R	Bit[7:0]: g_sum_cwf_s_i[31:24]
0x71	G_SUM_CWF_S	–	R	Bit[7:0]: g_sum_cwf_s_i[23:16]
0x72	G_SUM_CWF_S	–	R	Bit[7:0]: g_sum_cwf_s_i[15:8]
0x73	G_SUM_CWF_S	–	R	Bit[7:0]: g_sum_cwf_s_i[7:0]
0x74	R_SUM_CWF_S	–	R	Bit[7:0]: r_sum_cwf_s_i[31:24]
0x75	R_SUM_CWF_S	–	R	Bit[7:0]: r_sum_cwf_s_i[23:16]
0x76	R_SUM_CWF_S	–	R	Bit[7:0]: r_sum_cwf_s_i[15:8]
0x77	R_SUM_CWF_S	–	R	Bit[7:0]: r_sum_cwf_s_i[7:0]
0x78	RSVD	–	–	Reserved
0x79	ALL_CWF_CNT_L	–	R	Bit[7:6]: Reserved Bit[5:0]: all_cwf_cnt_l_i[21:16]
0x7A	ALL_CWF_CNT_L	–	R	Bit[7:0]: all_cwf_cnt_l_i[15:8]
0x7B	ALL_CWF_CNT_L	–	R	Bit[7:0]: all_cwf_cnt_l_i[7:0]
0x7C	RSVD	–	–	Reserved
0x7D	ALL_CWF_CNT_S	–	R	Bit[7:6]: Reserved Bit[5:0]: all_cwf_cnt_s_i[21:16]
0x7E	ALL_CWF_CNT_S	–	R	Bit[7:0]: all_cwf_cnt_s_i[15:8]
0x7F	ALL_CWF_CNT_S	–	R	Bit[7:0]: all_cwf_cnt_s_i[7:0]

6.10 AWB_PK

- base address: 0x80162A80 (ISP0_AWBPK)
- base address: 0x80163A80 (ISP1_AWBPK)

table 6-11 AWB_PK registers

address offset	register name	reset value	R/W	description
0x00	AWB_R_GAIN H	0x04	RW	Bit[7:4]: Reserved Bit[3:0]: awb_r_gain[11:8]
0x01	AWB_R_GAIN L	0x00	RW	Bit[7:0]: awb_r_gain[7:0]
0x02	AWB_G_GAIN H	0x04	RW	Bit[7:4]: Reserved Bit[3:0]: awb_g_gain[11:8]
0x03	AWB_G_GAIN L	0x00	RW	Bit[7:0]: awb_g_gain[7:0]
0x04	AWB_B_GAIN H	0x04	RW	Bit[7:4]: Reserved Bit[3:0]: awb_b_gain[11:8]
0x05	AWB_B_GAIN L	0x00	RW	Bit[7:0]: awb_b_gain[7:0]
0x06	AWB_MAN_CTRL	1'b0	RW	Bit[0]: awb_man_ctrl

6.11 EDR

- base address: 0x80162300 (ISP0_EDR)
- base address: 0x80163300 (ISP1_EDR)

table 6-12 EDR registers (sheet 1 of 5)

address offset	register name	reset value	R/W	description
0x00	EDR_00	0x02	RW	Bit[7:3]: Reserved Bit[2]: clear update Clear update signal to be 1'b0 Bit[1]: r_gainctrl Level adjustment with gain enable Bit[0]: smooth_en Enable curve smooth function
0x01	EDR_01	0x50	RW	Bit[7:0]: Stretch level
0x02	EDR_02	0x08	RW	Bit[7:5]: Reserved Bit[4:0]: Stretch step
0x03	EDR_03	0x39	RW	Bit[7:0]: Light0 Max value of ylist0

table 6-12 EDR registers (sheet 2 of 5)

address offset	register name	reset value	R/W	description
0x04	EDR_04	0x4A	RW	Bit[7:0]: Light1 Max value of ylist1
0x05	EDR_05	0x5D	RW	Bit[7:0]: Light2 Max value of ylist2
0x06	EDR_06	0x72	RW	Bit[7:0]: Light3 Max value of ylist3
0x07	EDR_07	0x7A	RW	Bit[7:0]: Light4 Max value of ylist4
0x08	EDR_08	0x81	RW	Bit[7:0]: Light5 Max value of ylist5
0x09	EDR_09	0x88	RW	Bit[7:0]: Light6 Max value of ylist6
0x0A	EDR_0A	0x8F	RW	Bit[7:0]: Light7 Max value of ylist7
0x0B	EDR_0B	0x97	RW	Bit[7:0]: Light8 Max value of ylist8
0x0C	EDR_0C	0x9E	RW	Bit[7:0]: Light9 Max value of ylist9
0x0D	EDR_0D	0xAE	RW	Bit[7:0]: Light10 Max value of ylist10
0x0E	EDR_0E	0xBC	RW	Bit[7:0]: Light11 Max value of ylist11
0x0F	EDR_0F	0xCF	RW	Bit[7:0]: Light12 Max value of ylist12
0x10	EDR_10	0xDE	RW	Bit[7:0]: Light13 Max value of ylist13
0x11	EDR_11	0xEC	RW	Bit[7:0]: Light14 Max value of ylist14
0x12	EDR_12	0x12	RW	Bit[7:0]: Dark0 Min value of ylist0
0x13	EDR_13	0x20	RW	Bit[7:0]: Dark1 Min value of ylist1
0x14	EDR_14	0x3B	RW	Bit[7:0]: Dark2 Min value of ylist2
0x15	EDR_15	0x5D	RW	Bit[7:0]: Dark3 Min value of ylist3
0x16	EDR_16	0x6A	RW	Bit[7:0]: Dark4 Min value of ylist4

table 6-12 EDR registers (sheet 3 of 5)

address offset	register name	reset value	R/W	description
0x17	EDR_17	0x76	RW	Bit[7:0]: Dark5 Min value of ylist5
0x18	EDR_18	0x81	RW	Bit[7:0]: Dark6 Min value of ylist6
0x19	EDR_19	0x8B	RW	Bit[7:0]: Dark7 Min value of ylist7
0x1A	EDR_1A	0x96	RW	Bit[7:0]: Dark8 Min value of ylist8
0x1B	EDR_1B	0x9E	RW	Bit[7:0]: Dark9 Min value of ylist9
0x1C	EDR_1C	0xAE	RW	Bit[7:0]: Dark10 Min value of ylist10
0x1D	EDR_1D	0xBC	RW	Bit[7:0]: Dark11 Min value of ylist11
0x1E	EDR_1E	0xCF	RW	Bit[7:0]: Dark12 Min value of ylist12
0x1F	EDR_1F	0xDE	RW	Bit[7:0]: Dark13 Min value of ylist13
0x20	EDR_20	0xEC	RW	Bit[7:0]: Dark14 Min value of ylist14
0x21~ 0x3A	RSVD	–	–	Reserved
0x40	EDR_40	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 00[22:16]
0x41	EDR_41	–	R	Bit[7:0]: Histogram 00[15:8]
0x42	EDR_42	–	R	Bit[7:0]: Histogram 00[7:0]
0x43	EDR_43	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 01[22:16]
0x44	EDR_44	–	R	Bit[7:0]: Histogram 01[15:8]
0x45	EDR_45	–	R	Bit[7:0]: Histogram 01[7:0]
0x46	EDR_46	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 02[22:16]
0x47	EDR_47	–	R	Bit[7:0]: Histogram 02[15:8]
0x48	EDR_48	–	R	Bit[7:0]: Histogram 02[7:0]
0x49	EDR_49	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 03[22:16]

table 6-12 EDR registers (sheet 4 of 5)

address offset	register name	reset value	R/W	description
0x4A	EDR_4A	–	R	Bit[7:0]: Histogram 03[15:8]
0x4B	EDR_4B	–	R	Bit[7:0]: Histogram 03[7:0]
0x4C	EDR_4C	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 04[22:16]
0x4D	EDR_4D	–	R	Bit[7:0]: Histogram 04[15:8]
0x4E	EDR_4E	–	R	Bit[7:0]: Histogram 04[7:0]
0x4F	EDR_4F	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 05[22:16]
0x50	EDR_50	–	R	Bit[7:0]: Histogram 05[15:8]
0x51	EDR_51	–	R	Bit[7:0]: Histogram 05[7:0]
0x52	EDR_52	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 06[22:16]
0x53	EDR_53	–	R	Bit[7:0]: Histogram 06[15:8]
0x54	EDR_54	–	R	Bit[7:0]: Histogram 06[7:0]
0x55	EDR_55	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 07[22:16]
0x56	EDR_56	–	R	Bit[7:0]: Histogram 07[15:8]
0x57	EDR_57	–	R	Bit[7:0]: Histogram 07[7:0]
0x58	EDR_58	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 08[22:16]
0x59	EDR_59	–	R	Bit[7:0]: Histogram 08[15:8]
0x5A	EDR_5A	–	R	Bit[7:0]: Histogram 08[7:0]
0x5B	EDR_5B	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 09[22:16]
0x5C	EDR_5C	–	R	Bit[7:0]: Histogram 09[15:8]
0x5D	EDR_5D	–	R	Bit[7:0]: Histogram 09[7:0]
0x5E	EDR_5E	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 10[22:16]
0x5F	EDR_5F	–	R	Bit[7:0]: Histogram 10[15:8]
0x60	EDR_60	–	R	Bit[7:0]: Histogram 10[7:0]
0x61	EDR_61	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 11[22:16]
0x62	EDR_62	–	R	Bit[7:0]: Histogram 11[15:8]

table 6-12 EDR registers (sheet 5 of 5)

address offset	register name	reset value	R/W	description
0x63	EDR_63	–	R	Bit[7:0]: Histogram 11[7:0]
0x64	EDR_64	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 12[22:16]
0x65	EDR_65	–	R	Bit[7:0]: Histogram 12[15:8]
0x66	EDR_66	–	R	Bit[7:0]: Histogram 12[7:0]
0x67	EDR_67	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 13[22:16]
0x68	EDR_68	–	R	Bit[7:0]: Histogram 13[15:8]
0x69	EDR_69	–	R	Bit[7:0]: Histogram 13[7:0]
0x6A	EDR_6A	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 14[22:16]
0x6B	EDR_6B	–	R	Bit[7:0]: Histogram 14[15:8]
0x6C	EDR_6C	–	R	Bit[7:0]: Histogram 14[7:0]
0x6D	EDR_6D	–	R	Bit[7]: Reserved Bit[6:0]: Histogram 15[22:16]
0x6E	EDR_6E	–	R	Bit[7:0]: Histogram 15[15:8]
0x6F	EDR_6F	–	R	Bit[7:0]: Histogram 15[7:0]

6.12 stretch

- base address: 0x80162380 (ISP0_STRETCH)
- base address: 0x80163380 (ISP1_STRETCH)

table 6-13 stretch registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	STRETCH_CTRL0	0x09	RW	Bit[7:4]: Reserved Bit[3:2]: Stretch minimum register Real stretch minimum value 00: 1024 01: 2048 10: 3096 11: 8192 Bit[1]: Stretch manual enable 0: Manual disable 1: manual_en Bit[0]: Stretch auto enable 0: Auto disable 1: Auto enable
0x01	STRETCH_CTRL1	0x60	RW	Bit[7:4]: Stretch max low level Bit[3:0]: Stretch min low level
0x02	STRETCH_CTRL2	0x10	RW	Bit[7:5]: Reserved Bit[4:0]: Stretch min high level
0x03	STRETCH_CTRL3	0x41	RW	Bit[7:4]: Stretch step2 Bit[3:0]: Stretch step1
0x04	STRETCH_CTRL4	0x10	RW	Bit[7:0]: Stretch cur low level If min low level is 0, cur low level must be greater or equal step1
0x05	STRETCH_CTRL5	0x0E	RW	Bit[7:4]: Reserved Bit[3:0]: Stretch cur high level[11:8]
0x06	STRETCH_CTRL6	0xFF	RW	Bit[7:0]: Stretch cur high level[7:0]
0x07	STRETCH_CTRL7	0x10	RW	Bit[7:0]: Stretch current low level manual value
0x08	STRETCH_CTRL8	0x03	RW	Bit[7:2]: Reserved Bit[1:0]: Stretch current high level manual value[9:8]
0x09	STRETCH_CTRL9	0xFF	RW	Bit[7:0]: Stretch current high level manual value[7:0]
0x0A	STRETCH_CTRL10	0x00	RW	Bit[7:0]: Stretch thres1[23:16]
0x0B	STRETCH_CTRL11	0x3B	RW	Bit[7:0]: Stretch_thres1[15:8]

table 6-13 stretch registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x0C	STRETCH_CTRL12	0xC4	RW	Bit[7:0]: Stretch_thres1[7:0] stretch_thre1 = width * height * m_nPsThres1 >> 10
0x0D	STRETCH_CTRL13	0x00	RW	Bit[7:0]: Stretch thres2[23:16]
0x0E	STRETCH_CTRL14	0x3B	RW	Bit[7:0]: Stretch_thres2[15:8]
0x0F	STRETCH_CTRL15	0xC4	RW	Bit[7:0]: Stretch_thres2[7:0] stretch_thre2 = width * height * m_nPsThres2>>10
0x10	STRETCH_CTRL16	–	R	Bit[7:0]: Stretch current low level read only
0x11	STRETCH_CTRL17	–	R	Bit[7:4]: Reserved Bit[3:0]: Stretch current high level read only[11:8]
0x12	STRETCH_CTRL18	–	R	Bit[7:0]: Stretch current high level read only[7:0]
0x13	STRETCH_CTRL19	–	R	Bit[7:4]: Reserved Bit[3:0]: Stretch gain read only[11:8]
0x14	STRETCH_CTRL20	–	R	Bit[7:0]: Stretch gain read only[7:0]

6.13 DPC

- base address: 0x80162400 (ISP0_DPC)
- base address: 0x80163400 (ISP1_DPC)

table 6-14 DPC registers (sheet 1 of 3)

address offset	register name	reset value	R/W	description
0x00	DPC_00	0x18	RW	Bit[7]: tail_en Enable removing tail pattern Bit[6]: sat_en Enable saturate Bit[5]: cluster_en Enable removing cluster pattern Bit[4]: sc_en Enable same channel detection Bit[3]: dc_en Enable different channel detection Bit[2]: smooth_en Enable using average G values when doing recovery Bit[1]: bwsnr_en Enable b&w sensor Bit[0]: man_mode_en Enable manual mode
0x01	DPC_01	0xDF	RW	Bit[7]: Manual thresholds for T type of cluster Bit[6]: Enable comparison in cfa3 line for tluster Bit[5]: Enable VBP mode Bit[4]: Enable color line Bit[3]: Enable single defect pixel mode Bit[2]: Enable tcluster mode Bit[1:0]: Option for padding boundary pixel
0x02	DPC_02	0x3F	RW	Bit[7:6]: Reserved Bit[5:4]: Number of unsaturate pixels around current pixels in single defect pixel mode Bit[3:2]: Number of unsaturate pixels around currnet pixels in tcluster defect pixels mode Bit[2:0]: Vertical line number

table 6-14 DPC registers (sheet 2 of 3)

address offset	register name	reset value	R/W	description
0x03	DPC_03	0x08	RW	Bit[6:0]: When man_mode_en is on, it means wthre[6:0] (threshold value for detecting white pixel). When man_mode_en is off, it means glist1[6:0].
0x04	DPC_04	0x20	RW	Bit[6:0]: When man_mode_en is on, it means bthre[6:0] (threshold value for detecting black pixel). When man_mode_en is off, it means glist2[6:0].
0x05	DPC_05	0x10	RW	Bit[6:0]: thre1[6:0] Used in recovery
0x06	DPC_06	0x20	RW	Bit[6:0]: thre2[6:0] Used in recovery
0x07	DPC_07	0x10	RW	Bit[6:0]: thre3[6:0] Used in checkbelow
0x08	DPC_08	0x18	RW	Bit[6:0]: thre4[6:0] Used in detection
0x09	DPC_09	0x08	RW	Bit[6:0]: wthre_list0[6:0] Used for wthre calculation in auto mode
0x0A	DPC_0A	0x04	RW	Bit[6:0]: wthre_list1[6:0] Used for wthre calculation in auto mode
0x0B	DPC_0B	0x02	RW	Bit[6:0]: wthre_list2[6:0] Used for wthre calculation in auto mode
0x0C	DPC_0C	0x02	RW	Bit[6:0]: wthre_list3[6:0] Used for wthre calculation in auto mode
0x0D	DPC_0D	0x0C	RW	Bit[7:0]: bthre_list0[7:0] Used for bthre calcluation in auto mode
0x0E	DPC_0E	0x06	RW	Bit[7:0]: bthre_list1[7:0] Used for bthre calcluation in auto mode
0x0F	DPC_0F	0x02	RW	Bit[7:0]: bthre_list2[7:0] Used for bthre calcluation in auto mode

table 6-14 DPC registers (sheet 3 of 3)

address offset	register name	reset value	R/W	description
0x10	DPC_10	0x02	RW	Bit[7:0]: bthre_list3[7:0] Used for bthre calcluation in auto mode
0x11	DPC_11	0xFF	RW	Bit[7:0]: Thre saturate value
0x12	DPC_12	0x07	RW	Bit[6:0]: vb_gain_th1
0x13	DPC_13	0x03	RW	Bit[6:0]: vb_gain_th2
0x14	DPC_13	0x03	RW	Bit[6:0]: smooth_gainliist0
0x15	DPC_14	0x07	RW	Bit[6:0]: smooth_gainlist1
0x16	DPC_15	0x0F	RW	Bit[6:0]: smooth_gainlist2
0x17	DPC_16	0x07	RW	Bit[6:0]: smooth_gain_th1
0x18	DPC_17	0x03	RW	Bit[6:0]: smooth_gain_th2
0x19	DPC_19	0xF0	RW	Bit[7:0]: Thresholds of unsaturate pixel
0x1A	DPC_1A	0x08	RW	Bit[7:0]: Thresholds of T type of cluster
0x20	DPC_20	–	R	Bit[7:3]: Reserved Bit[2]: vbp_en Bit[1]: color_en Bit[0]: smooth_en
0x21	DPC_21	–	R	Bit[6:0]: WThre of DPC
0x22	DPC_22	–	R	Bit[6:0]: BThre of DPC
0x23	DPC_23	–	R	Bit[6:0]: Thre1[6:0] of DPC
0x24	DPC_24	–	R	Bit[6:0]: Thre2[6:0] of DPC
0x25	DPC_25	–	R	Bit[6:0]: Thre3[6:0] of DPC
0x26	DPC_26	–	R	Bit[6:0]: Thre4[6:0] of DPC
0x27	DPC_27	–	R	Bit[7:0]: Thre[7:0] of DPC in Tcluster mode

6.14 RAW_gamma

- base address: 0x80162480 (ISP0_RAW_GAMMA)
- base address: 0x80163480 (ISP1_RAW_GAMMA)

table 6-15 RAW gamma registers

address offset	register name	reset value	R/W	description
0x00	GAMMA_CTRL	0x00	RW	Bit[7:2]: Reserved Bit[1]: yslp15_man Enable manual set yst15 slope 0: yst15 slope input disable 1: yst15 slope input enable Bit[0]: bias_plus Enable bias 0: Bias disable 1: Bias enable
0x01	YST1	0x26	RW	Parameter of yst1
0x02	YST2	0x35	RW	Parameter of yst2
0x03	YST3	0x48	RW	Parameter of yst3
0x04	YST4	0x57	RW	Parameter of yst4
0x05	YST5	0x63	RW	Parameter of yst5
0x06	YST6	0x6E	RW	Parameter of yst6
0x07	YST7	0x77	RW	Parameter of yst7
0x08	YST8	0x80	RW	Parameter of yst8
0x09	YST9	0x88	RW	Parameter of yst9
0x0A	YST10	0x96	RW	Parameter of yst10
0x0B	YST11	0xA3	RW	Parameter of yst11
0x0C	YST12	0xAF	RW	Parameter of yst12
0x0D	YST13	0xC5	RW	Parameter of yst1 3
0x0E	YST14	0xD7	RW	Parameter of yst1 4
0x0F	YST15	0xE8	RW	Parameter of yst1 5
0x10	YSLP15	0x0F	RW	Slope of yst15
0x11	BIAS	0x00	RW	Bias Offset

6.15 RAW_DNS

- base address: 0x80162500 (ISP0_RAW_DNS)
- base address: 0x80163500 (ISP1_RAW_DNS)

table 6-16 RAW_DNS registers

address	register name	default value	R/W	description
0x00	RAW_DNS_CTRL_00	0x11	RW	Bit[7:6]: Reserved Bit[5:2]: raw_dns_y_slop Bit[1]: raw_dns manual mode 0: Control parameters auto mode 1: Control parameters manual mode Bit[0]: Enable G pixel process
0x01	RAW_DNS_CTRL_01	0x04	RW	Bit[7:0]: Y noise list0
0x02	RAW_DNS_CTRL_02	0x08	RW	Bit[7:0]: Y noise list1
0x03	RAW_DNS_CTRL_03	0x10	RW	Bit[7:0]: Y noise list2
0x04	RAW_DNS_CTRL_04	0x18	RW	Bit[7:0]: Y noise list3
0x05	RAW_DNS_CTRL_05	0x20	RW	Bit[7:0]: Y noise list4
0x06	RAW_DNS_CTRL_06	0x30	RW	Bit[7:0]: Y noise list5
0x07	RAW_DNS_CTRL_07	0x40	RW	Bit[7:0]: Y noise list6
0x08	RAW_DNS_CTRL_08	0x40	RW	Bit[7:0]: Y noise list7
0x09	RAW_DNS_CTRL_09	0xFF	RW	Bit[7:0]: Max edgethre
0x0A	RAW_DNS_CTRL_10	0x08	RW	Bit[7:0]: Raw DNS noise parameter in manual mode
0x0B	RAW_DNS_CTRL_11	0x18	RW	Bit[7:0]: Raw DNS edgethre parameter in manual mode
0x0C	RAW_DNS_CTRL_12	0x08	R	Bit[7:0]: Y noise value read only register
0x0D	RAW_DNS_CTRL_13	0x18	R	Bit[7:0]: Edgethre value read only register

6.16 CIP

- base address: 0x80162580 (ISP0_CIP)
- base address: 0x80163580 (ISP1_CIP)

table 6-17 CIP registers

address offset	register name	reset value	R/W	description
0x00	CIP_CTRL_00	0x80	RW	Bit[7:4]: noise_y_slp Bit[3:0]: local_sharpen
0x01	CIP_CTRL_01	0x00	RW	Bit[7:5]: Reserved Bit[4:1]: gbgr_shift Bit[0: gbgr_rem_enable
0x02	CIP_CTRL_02	0x00	RW	noise_list0
0x03	CIP_CTRL_03	0x00	RW	noise_list1
0x04	CIP_CTRL_04	0x00	RW	noise_list2
0x05	CIP_CTRL_05	0x00	RW	noise_list3
0x06	CIP_CTRL_06	0x00	RW	noise_list4
0x07	CIP_CTRL_07	0x00	RW	noise_list5
0x08	CIP_CTRL_08	0x00	RW	noise_list6
0x09	CIP_CTRL_09	0x00	RW	noise_list7
0x0A	CIP_CTRL_0A	0x00	RW	min_sharpen
0x0B	CIP_CTRL_0B	0x0C	RW	max_sharpen
0x0C	CIP_CTRL_0C	0x08	RW	min_detail
0x0D	CIP_CTRL_0D	0x0C	RW	max_detail
0x0E	CIP_CTRL_0E	0x02	RW	min_sharpen_gain
0x0F	CIP_CTRL_0F	0x10	RW	max_sharpen_gain
0x10	CIP_CTRL_10	0x60	RW	Bit[7]: Reserved Bit[6]: r_anti_alias Bit[5]: r_fix_h_en Bit[4]: r_cip_option Bit[3:2]: r_binning_patt Bit[1:0]: r_binning_mode
0x11	CIP_CTRL_11	0x40	RW	CIP EOF Delay Number

6.17 CMX

- base address: 0x80162600 (ISP0_CMX)
- base address: 0x80163600 (ISP1_CMX)

table 6-18 CMX registers (sheet 1 of 3)

address offset	register name	reset value	R/W	description
0x00	CMX_CTRL_00	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_d00[11:8]
0x01	CMX_CTRL_01	0x00	RW	Bit[7:0]: cmx_d00[7:0]
0x02	CMX_CTRL_02	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_d01[11:8]
0x03	CMX_CTRL_03	0x00	RW	Bit[7:0]: cmx_d01[7:0]
0x04	CMX_CTRL_04	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_d10[11:8]
0x05	CMX_CTRL_05	0x00	RW	Bit[7:0]: cmx_d10[7:0]
0x06	CMX_CTRL_06	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_d11[11:8]
0x07	CMX_CTRL_07	0x00	RW	Bit[7:0]: cmx_d11[7:0]
0x08	CMX_CTRL_08	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_d20[11:8]
0x09	CMX_CTRL_09	0x00	RW	Bit[7:0]: cmx_d20[7:0]
0x0A	CMX_CTRL_0A	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_d21[11:8]
0x0B	CMX_CTRL_0B	0x00	RW	Bit[7:0]: cmx_d21[7:0]
0x0C	CMX_CTRL_0C	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_a00[11:8]
0x0D	CMX_CTRL_0D	0x00	RW	Bit[7:0]: cmx_a00[7:0]
0x0E	CMX_CTRL_0E	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_a01[11:8]
0x0F	CMX_CTRL_0F	0x00	RW	Bit[7:0]: cmx_a01[7:0]
0x10	CMX_CTRL_10	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_a10[11:8]
0x11	CMX_CTRL_11	0x00	RW	Bit[7:0]: cmx_a10[7:0]
0x12	CMX_CTRL_12	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_a11[11:8]
0x13	CMX_CTRL_13	0x00	RW	Bit[7:0]: cmx_a11[7:0]

table 6-18 CMX registers (sheet 2 of 3)

address offset	register name	reset value	R/W	description
0x14	CMX_CTRL_14	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_a20[11:8]
0x15	CMX_CTRL_15	0x00	RW	Bit[7:0]: cmx_a20[7:0]
0x16	CMX_CTRL_16	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_a21[11:8]
0x17	CMX_CTRL_17	0x00	RW	Bit[7:0]: cmx_a21[7:0]
0x18	CMX_CTRL_18	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_c00[11:8]
0x19	CMX_CTRL_19	0x00	RW	Bit[7:0]: cmx_c00[7:0]
0x1A	CMX_CTRL_1A	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_c01[11:8]
0x1B	CMX_CTRL_1B	0x00	RW	Bit[7:0]: cmx_c01[7:0]
0x1C	CMX_CTRL_1C	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_c10[11:8]
0x1D	CMX_CTRL_1D	0x00	RW	Bit[7:0]: cmx_c10[7:0]
0x1E	CMX_CTRL_1E	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_c11[11:8]
0x1F	CMX_CTRL_1F	0x00	RW	Bit[7:0]: cmx_c11[7:0]
0x20	CMX_CTRL_20	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_c20[11:8]
0x21	CMX_CTRL_21	0x00	RW	Bit[7:0]: cmx_c20[7:0]
0x22	CMX_CTRL_22	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: cmx_c21[11:8]
0x23	CMX_CTRL_23	0x00	RW	Bit[7:0]: cmx_c21[7:0]
0x24	CMX_CTRL_24	0x01	RW	Bit[7:5]: Reserved Bit[4]: r_cmx_br_gain_man Bit[3:0]: r_cmx_awb_bgain[11:8]
0x25	CMX_CTRL_25	0x00	RW	Bit[7:0]: r_cmx_awb_bgain[7:0]
0x26	CMX_CTRL_26	0x01	RW	Bit[7:4]: Reserved Bit[3:0]: r_cmx_awb_rgain[11:8]
0x27	CMX_CTRL_27	0x00	RW	Bit[7:0]: r_cmx_awb_rgain[7:0]
0x28	CMX_CTRL_28	0x00	RW	Bit[7:2]: Reserved Bit[1:0]: r_cmx_ct_list0[9:8]
0x29	CMX_CTRL_29	0x80	RW	Bit[7:0]: r_cmx_ct_list0[7:0]

table 6-18 CMX registers (sheet 3 of 3)

address offset	register name	reset value	R/W	description
0x2A	CMX_CTRL_2A	0x00	RW	Bit[7:2]: Reserved Bit[1:0]: r_cmx_ct_list1[9:8]
0x2B	CMX_CTRL_2B	0x80	RW	Bit[7:0]: r_cmx_ct_list1[7:0]
0x2C	CMX_CTRL_2C	0x00	RW	Bit[7:2]: Reserved Bit[1:0]: r_cmx_ct_list2[9:8]
0x2D	CMX_CTRL_2D	0x80	RW	Bit[7:0]: r_cmx_ct_list2[7:0]
0x2E	CMX_CTRL_2E	0x00	RW	Bit[7:6]: Reserved Bit[5]: Sign of cmx_d00 Bit[4]: Sign of cmx_d01 Bit[3]: Sign of cmx_d10 Bit[2]: Sign of cmx_d11 Bit[1]: Sign of cmx_d20 Bit[0]: Sign of cmx_d
0x2F	CMX_CTRL_2F	0x00	RW	Bit[7:6]: Reserved Bit[5]: Sign of cmx_a00 Bit[4]: Sign of cmx_a01 Bit[3]: Sign of cmx_a10 Bit[2]: Sign of cmx_a11 Bit[1]: Sign of cmx_a20 Bit[0]: Sign of cmx_a21
0x30	CMX_CTRL_30	0x00	RW	Bit[7:6]: Reserved Bit[5]: Sign of cmx_c00 Bit[4]: Sign of cmx_c01 Bit[3]: Sign of cmx_c10 Bit[2]: Sign of cmx_c11 Bit[1]: Sign of cmx_c20 Bit[0]: Sign of cmx_c21
0x31	CMX_CTRL_31	0x01	RW	Bit[7:2]: Reserved Bit[1]: r_cmx_auto_ct_en Bit[0]: r_cmx_auto_factor_en
0x32	CMX_CTRL_32	0x40	RW	Bit[7]: Reserved Bit[6:0]: r_cmx_max_factor
0x33	CMX_CTRL_33	0x40	RW	Bit[7]: Reserved Bit[6:0]: r_cmx_min_factor
0x34	CMX_CTRL_34	0x07	RW	Bit[7:5]: Reserved Bit[4:0]: r_cmx_min_gain real_cmx_min_gain = (r_cmx_min_gain+1)*16
0x35	CMX_CTRL_35	0x0F	RW	Bit[7:5]: Reserved Bit[4:0]: r_cmx_max_gain real_cmx_max_gain = (r_cmx_max_gain+1)*16

6.18 RGB gamma

- base address: 0x80162680 (ISP0_RGB_GAMMA)
- base address: 0x80163680 (ISP1_RGB_GAMMA)

table 6-19 RGB gamma registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	GAMMA_00	0x04	RW	Bit[7:0]: List00 for Y
0x01	GAMMA_01	0x08	RW	Bit[7:0]: List01 for Y
0x02	GAMMA_02	0x10	RW	Bit[7:0]: List02 for Y
0x03	GAMMA_03	0x20	RW	Bit[7:0]: List03 for Y
0x04	GAMMA_04	0x28	RW	Bit[7:0]: List04 for Y
0x05	GAMMA_05	0x30	RW	Bit[7:0]: List05 for Y
0x06	GAMMA_06	0x38	RW	Bit[7:0]: List06 for Y
0x07	GAMMA_07	0x40	RW	Bit[7:0]: List07 for Y
0x08	GAMMA_08	0x48	RW	Bit[7:0]: List08 for Y
0x09	GAMMA_09	0x50	RW	Bit[7:0]: List09 for Y
0x0A	GAMMA_0A	0x60	RW	Bit[7:0]: List0a for Y
0x0B	GAMMA_0B	0x70	RW	Bit[7:0]: List0b for Y
0x0C	GAMMA_0C	0x90	RW	Bit[7:0]: List0c for Y
0x0D	GAMMA_0D	0xB0	RW	Bit[7:0]: List0d for Y
0x0E	GAMMA_0E	0xD0	RW	Bit[7:0]: List0e for Y
0x0F	GAMMA_0F	0xFF	RW	Bit[7:0]: gamma_max_shd_h_gain
0x10	GAMMA_10	0x01	RW	Bit[7:1]: Reserved Bit[0]: gamma_man_en
0x40	GAMMA_40	0x0E	RW	Bit[7:0]: RISC read or write list00 for Y
0x41	GAMMA_41	0x1A	RW	Bit[7:0]: RISC read or write list01 for Y
0x42	GAMMA_42	0x31	RW	Bit[7:0]: RISC read or write list02 for Y
0x43	GAMMA_43	0x5A	RW	Bit[7:0]: RISC read or write list03 for Y
0x44	GAMMA_44	0x69	RW	Bit[7:0]: RISC read or write list04 for Y
0x45	GAMMA_45	0x75	RW	Bit[7:0]: RISC read or write list05 for Y
0x46	GAMMA_46	0x7E	RW	Bit[7:0]: RISC read or write list06 for Y
0x47	GAMMA_47	0x88	RW	Bit[7:0]: RISC read or write list07 for Y

table 6-19 RGB gamma registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x48	GAMMA_48	0x8F	RW	Bit[7:0]: RISC read or write list08 for Y
0x49	GAMMA_49	0x96	RW	Bit[7:0]: RISC read or write list09 for Y
0x4A	GAMMA_4A	0xA3	RW	Bit[7:0]: RISC read or write list0a for Y
0x4B	GAMMA_4B	0xAF	RW	Bit[7:0]: RISC read or write list0b for Y
0x4C	GAMMA_4C	0xC4	RW	Bit[7:0]: RISC read or write list0c for Y
0x4D	GAMMA_4D	0xD7	RW	Bit[7:0]: RISC read or write list0d for Y
0x4E	GAMMA_4E	0xE8	RW	Bit[7:0]: RISC read or write list0e for Y

6.19 RGB_DNS

- base address: 0x80162700 (ISP0_RGB_DNS)
- base address: 0x80163700 (ISP1_RGB_DNS)

table 6-20 RGB_DNS registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	RGB_DNS_CTRL00	0x0D	RW	Bit[7:6]: Reserved Bit[5:1]: Shadow extra noise Bit[0]: Smooth y enable
0x01	RGB_DNS_CTRL01	0x08	RW	Bit[7:4]: Reserved Bit[3:0]: RGB DNS edgethre
0x02	RGB_DNS_CTRL02	0x08	RW	Bit[7:4]: Y noise list1 Bit[3:0]: Y noise list0
0x03	RGB_DNS_CTRL03	0x1A	RW	Bit[7:4]: Y noise list3 Bit[3:0]: Y noise list2
0x04	RGB_DNS_CTRL04	0x24	RW	Bit[7:4]: Y noise list5 Bit[3:0]: Y noise list4
0x05	RGB_DNS_CTRL05	0x24	RW	Bit[7:4]: Y noise list7 Bit[3:0]: Y noise list6
0x06	RGB_DNS_CTRL06	0x02	RW	Bit[5:0]: UV noise list0
0x07	RGB_DNS_CTRL07	0x04	RW	Bit[5:0]: UV noise list1
0x08	RGB_DNS_CTRL08	0x06	RW	Bit[5:0]: UV noise list2
0x09	RGB_DNS_CTRL09	0x08	RW	Bit[5:0]: UV noise list3
0x0A	RGB_DNS_CTRL0A	0x0A	RW	Bit[5:0]: UV noise list4

table 6-20 RGB_DNS registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x0B	RGB_DNS_CTRL0B	0x0A	RW	Bit[5:0]: UV noise list5
0x0C	RGB_DNS_CTRL0C	0x0A	RW	Bit[5:0]: UV noise list6
0x0D	RGB_DNS_CTRL0D	0x0A	RW	Bit[5:0]: UV noise list7
0x0E	RGB_DNS_CTRL0E	0x02	RW	Bit[7:4]: Reserved Bit[3:1]: Y noise manual value Bit[0]: RGB DNS manual mode
0x0F	RGB_DNS_CTRL0F	0x04	RW	Bit[5:0]: UV noise manual value
0x10	RGB_DNS_CTRL10	–	R	Y Noise Value Read Only
0x11	RGB_DNS_CTRL11	–	R	UV Noise Value Read Only

6.20 SDE

- base address: 0x80162780 (ISP0_SDE)
- base address: 0x80163780 (ISP1_SDE)

table 6-21 SDE registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	R_SDE_CTRL0	0x00	RW	Bit[7]: Fixy enable Bit[6]: Neg enable Bit[5]: Gray enable Bit[4]: fix_v enable Bit[3]: fix_u enable Bit[2]: Contrast enable Bit[1]: Saturation enable Bit[0]: Hue enable
0x01	R_SDE_CTRL1	0x80	RW	Bit[7:0]: hue_cos Cb' = cos(A)*(Cb-128)+sin(A)*(Cr-128)+128
0x02	R_SDE_CTRL2	0x00	RW	Bit[7:0]: hue_sin Cr' = cos(A)*(Cr-128)-sin(A)*(Cb-128)+128
0x03	R_SDE_CTRL3	0x40	RW	Bit[7:0]: sat_u or ureg Saturation coefficient for U
0x04	R_SDE_CTRL4	0x40	RW	Bit[7:0]: sat_v or vreg Saturation coefficient for V

table 6-21 SDE registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x05	R_SDE_CTRL5	0x00	RW	Bit[7:0]: Yoffset or fixy Offset = YOffset*(-1)^sign2
0x06	R_SDE_CTRL6	0x20	RW	Bit[7:0]: Ygain Y2 = (Y1+Bright-Offset)*YGain+Offset
0x07	R_SDE_CTRL7	0x00	RW	Bit[7:0]: Ybright Britht = YBright*(-1)^sign3
0x08	R_SDE_CTRL8	0x01	RW	Bit[7:6]: Reserved Bit[5:0]: Signset
0x09	R_SDE_CTRL9	0x00	RW	Bit[7:0]: uvadj_th1 UV adjust threshold parameter
0x0A	R_SDE_CTRLA	0xFF	RW	Bit[7:0]: uvadj_th2 UV adjust threshold parameter
0x0B	R_SDE_CTRLB	0x00	RW	Bit[7:1]: Reserved Bit[0]: uvadj_man_en UV adjust manual enable

6.21 AEC/AGC

- base address: 0x80162800 (ISP0_AECAGC)
- base address: 0x80163800 (ISP1_AECAGC)

table 6-22 AEC/AGC registers (sheet 1 of 6)

address offset	register name	reset value	R/W	description
0x00	AEC_00	0x78	RW	Bit[7]: Reserved Bit[6]: Enable function that exposure can be less than 1 line Bit[5]: Enable banding effect remove Bit[4]: Enable function that exposure can be less than 1 band Bit[3]: 0: Select tc_eof_blc_i as start signal 1: Select avg_done_i as start signal Bit[2]: Enable night mode Bit[1]: Select new balance method Bit[0]: Enable freeze mode

table 6-22 AEC/AGC registers (sheet 2 of 6)

address offset	register name	reset value	R/W	description
0x01	AEC_01	0x04	RW	Bit[7:0]: Minimum exposure (Set 1, high 4 bits represent integer lines, low 4 bits represent fraction lines)
0x02	AEC_02	0x0F	RW	Bit[7:5]: Reserved Bit[4:0]: Maximum exposure for 60Hz (Set 1)
0x03	AEC_03	0xE0	RW	Bit[7:0]: Maximum exposure for 60Hz (Set 1, no fraction lines)
0x05	AEC_05	0x30	RW	Bit[7]: Reverse flag signal of 50/60Hz Bit[6]: Enable frame insertion in night mode Bit[5]: Enable automatic step calculation Bit[4:0]: Step ratio in automatic step calculation
0x06	AEC_06	0x10	RW	Bit[7:5]: Reserved Bit[4:0]: Step 1 in manual step calculation
0x07	AEC_07	0x18	RW	Bit[7:4]: Step 2 in manual step calculation Bit[3:0]: Step 3 in manual step calculation
0x08	AEC_08	0x00	RW	Bit[1:0]: Band step for 50Hz light
0x09	AEC_09	0x98	RW	Bit[7:0]: Band step for 50Hz light
0x0A	AEC_0A	0x00	RW	Bit[1:0]: Band step for 60Hz light
0x0B	AEC_0B	0x7F	RW	Bit[7:0]: Band step for 60Hz light
0x0C	AEC_0C	0xE4	RW	Bit[7:4]: Maximum fraction exposure Bit[3:0]: Minimum fraction exposure
0x0D	AEC_0D	0x04	RW	Bit[5:0]: Maximum band number in one frame for 60 Hz light source
0x0E	AEC_0E	0x03	RW	Bit[5:0]: Maximum band number in one frame for 50 Hz light source
0x0F	AEC_0F	0x78	RW	Bit[7:0]: Upper bound of stable range (set 1)
0x10	AEC_10	0x68	RW	Bit[7:0]: Lower bound of stable range (set 1)
0x11	AEC_11	0xD0	RW	Bit[7:0]: Upper bound of range to determine step value in manual step calculation

table 6-22 AEC/AGC registers (sheet 3 of 6)

address offset	register name	reset value	R/W	description
0x12	AEC_12	0x50	RW	Bit[7:0]: Manual average value
0x13	AEC_13	0x90	RW	Bit[7]: Reserved Bit[6]: Enable pre sensor gain Bit[4]: Value of pre-sensor gain Pre-sensor gain means output real gain must be larger than pre-gain
0x14	AEC_14	0x0E	RW	Bit[7:5]: Reserved Bit[4:0]: Maximum exposure for 50Hz (Set 1)
0x15	AEC_15	0x40	RW	Bit[7:0]: Maximum exposure for 50Hz (Set 1, no fraction lines)
0x17	AEC_17	0x01	RW	Bit[1:0]: Threshold value of gain_night
0x18	AEC_18	0x07	RW	Bit[2:0]: Upper bound of output real gain
0x19	AEC_19	0xC0	RW	Bit[7:0]: Upper bound of output real gain
0x1A	AEC_1A	0x06	RW	Bit[7:0]: Minimum difference between VTS and max one frame exposure
0x1B	AEC_1B	0x76	RW	Bit[7:0]: Upper bound of stable range (Set 2)
0x1C	AEC_1C	0x06	RW	Bit[7:0]: Added rows number for LED mode
0x1D	AEC_1D	0x18	RW	Bit[7:0]: Added rows number for LED mode
0x1E	AEC_1E	0x68	RW	Bit[7:0]: Lower bound of stable range (Set 2)
0x1F	AEC_1F	0x40	RW	Bit[7:0]: Lower bound of range to determine step value in manual step calculation
0x20	AEC_20	0x20	RW	Bit[7]: Enable function of subtracting black level Bit[6:3]: Black level Bit[2]: Enable strobe option Bit[1]: Enable manual average Bit[0]: Enable calculation of case current exposure does not change

table 6-22 AEC/AGC registers (sheet 4 of 6)

address offset	register name	reset value	R/W	description
0x21	AEC_21	0x78	RW	Bit[7]: Reserved Bit[6:4]: Maximum inserted frame number in frame insertion method of night mode Bit[3:1]: Reserved Bit[0]: gain_adj_opt 0: Add 1 to calculated gain if AEC is increasing and exposure and gain remain unchanged. Subtract 1 to calculated gain if AEC is decreasing and exposure and gain remain unchanged. 1: No operations of adding 1 or subtracting 1
0x25	AEC_25	0x00	RW	Bit[7:5]: Reserved Bit[4:2]: freeze_cnt AEC update once per freeze_cnt frames Bit[1]: Enable fraction constraint in less 1 line state Bit[0]: Enable same auto_step speed in automatic step mode
0x26	AEC_26	8'H02	RW	Bit[7:0]: expo_line If exposure is less than expo_line, speed of auto_step is step_man2; otherwise, speed is r_step_auo
0x50	AEC_50	–	R	Bit[7:0]: Average of current frame image
0x51	AEC_51	–	R	Bit[7:2]: Reserved Bit[1:0]: Current state of exposure FSM
0x52	AEC_52	–	R	Bit[7:3]: Reserved Bit[2]: Increase status of current frame Bit[1]: Decrease status of current frame Bit[0]: Balance status of current frame

table 6-22 AEC/AGC registers (sheet 5 of 6)

address offset	register name	reset value	R/W	description
0x53	AEC_53	–	R	Bit[7:6]: Reserved Bit[5]: Update exposure and gain in current frame Bit[4]: Total condition for adjust gain only Bit[3]: One condition for adjust gain only Bit[2]: One condition for adjust gain only Bit[1]: One condition for adjust gain only Bit[0]: One condition for adjust gain only
0x54	AEC_54	–	R	Bit[7:4]: Reserved Bit[3:0]: Calculated step value
0x55	AEC_55	–	R	Bit[7:0]: Calculated step value
0x56	AEC_56	–	R	Bit[7:6]: Reserved Bit[5:0]: e_tmp for debug purpose[29:24]
0x57	AEC_57	–	R	Bit[7:0]: e_tmp for debug purpose[23:16]
0x58	AEC_58	–	R	Bit[7:0]: e_tmp for debug purpose[15:8]
0x59	AEC_59	–	R	Bit[7:0]: e_tmp for debug purpose[7:0]
0x5A	AEC_5A	–	R	Bit[7:6]: Reserved Bit[5:0]: e_tmp_pg for debug purpose[29:24]
0x5B	AEC_5B	–	R	Bit[7:0]: e_tmp_pg for debug purpose[23:16]
0x5C	AEC_5C	–	R	Bit[7:0]: e_tmp_pg for debug purpose[15:8]
0x5D	AEC_5D	–	R	Bit[7:0]: e_tmp_pg for debug purpose[7:0]
0x5E	AEC_5E	–	R	Bit[7:0]: VTS increment of current frame[15:8]
0x5F	AEC_5F	–	R	Bit[7:0]: VTS increment of current frame[7:0]

table 6-22 AEC/AGC registers (sheet 6 of 6)

address offset	register name	reset value	R/W	description
0x60	AEC_60	–	R	Bit[7:4]: Reserved Bit[3]: Signal indicating LED add dummy lines Bit[2]: Signal indicating LED working mode Bit[1]: Input flag signal of 50/60Hz light source Bit[0]: New flag signal of 50/60Hz light source
0x61	AEC_61	–	R	Bit[7:3]: Reserved Bit[2:0]: Exposure of maximum 1 frame[10:8]
0x62	AEC_62	–	R	Bit[7:0]: Exposure of maximum 1 frame[7:0]

6.22 UV_sum

- base address: 0x80162880 (ISP0_UV_SUM)
- base address: 0x80163880 (ISP1_UV_SUM)

table 6-23 UV_sum registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	UREF_L	0x08	RW	Bit[7:0]: uref[7:0]
0x01	UREF_H	0xC0	RW	Bit[7:0]: uref[15:8]
0x02	VREF_L	0x08	RW	Bit[7:0]: vref[7:0]
0x03	VREF_H	0xC0	RW	Bit[7:0]: vref_h[15:8]
0x10	U_SUM	–	R	Bit[7:0]: u_sum[29:24]
0x11	U_SUM	–	R	Bit[7:0]: u_sum[23:16]
0x12	U_SUM	–	R	Bit[7:0]: u_sum[15:8]
0x13	U_SUM	–	R	Bit[7:0]: u_sum[7:0]
0x14	V_SUM	–	R	Bit[7:0]: v_sum[29:24]
0x15	V_SUM	–	R	Bit[7:0]: v_sum[23:16]
0x16	V_SUM	–	R	Bit[7:0]: v_sum[15:8]
0x17	V_SUM	–	R	Bit[7:0]: v_sum[7:0]

table 6-23 UV_sum registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x18	CNT	–	R	Bit[7:0]: cnt[21:16]
0x19	CNT	–	R	Bit[7:0]: cnt[15:8]
0x1A	CNT	–	R	Bit[7:0]: cnt[7:0]

6.23 UV_DNS

- base address: 0x80162900 (ISP0_UV_DNS)
- base address: 0x80163900 (ISP1_UV_DNS)

table 6-24 UV_DNS registers

address offset	register name	reset value	R/W	description
0x00	NS_LIST0	0x00	RW	Bit[7:0]: Noise list 0
0x01	NS_LIST1	0x01	RW	Bit[7:0]: Noise list 1
0x02	NS_LIST2	0x04	RW	Bit[7:0]: Noise list 2
0x03	NS_LIST3	0x10	RW	Bit[7:0]: Noise list 3
0x04	NS_LIST4	0x20	RW	Bit[7:0]: Noise list 4
0x05	NS_LIST5	0x30	RW	Bit[7:0]: Noise list 5
0x06	NS_CTRL0	0x06	RW	Bit[7]: Noise manual enable Bit[6]: Shadow extra noise disable Bit[5]: Not used Bit[4:0]: Shadow extra noise
0x07	NOISE_MAN	0x00	RW	Bit[7:0]: Manual noise
0x08	NOISE	–	R	Noise

6.24 WINC

- base address: 0x80162980 (ISP0_WINC)
- base address: 0x80163980 (ISP1_WINC)

table 6-25 WINC registers

address offset	register name	reset value	R/W	description
0x00	XSTART H	0x00	RW	Bit[7:5]: Reserved Bit[4:0]: xstart[12:8] Start address in horizontal
0x01	XSTART L	0x00	RW	Bit[7:0]: xstart[7:0]
0x02	YSTART H	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: ystart[11:8] Start address in vertical
0x03	YSTART L	0x00	RW	Bit[7:0]: ystart[7:0]
0x04	X_WINH	0x10	RW	Bit[7:5]: Reserved Bit[4:0]: x_win[12:8] Select whole zone width
0x05	X_WINL	0xA0	RW	Bit[7:0]: x_win[7:0]
0x06	Y_WINH	0x0C	RW	Bit[7:4]: Reserved Bit[3:0]: y_win[11:8] Select whole zone height
0x07	Y_WINL	0x78	RW	Bit[7:0]: y_win[7:0]
0x08	WIN_MAN_EN	0x00	RW	Bit[7:1]: Reserved Bit[0]: win_man_en 0: Window size from window top 1: Window size from register 0x00 to register 0x07
0x09	PX_CNT	–	R	Bit[7:5]: Reserved Bit[4:0]: px_cnt[12:8] Pixel count from input image in horizontal
0x0A	PX_CNT	–	R	Bit[7:0]: px_cnt[7:0]
0x0B	LN_CNT	–	R	Bit[7:4]: Reserved Bit[3:0]: ln_cnt[11:8] Line count from input image in vertical
0x0C	LN_CNT	–	R	Bit[7:0]: ln_cnt[7:0]

6.25 AECPK

- base address: 0x80162A00 (ISP0_AECPK)
- base address: 0x80163A00 (ISP1_AECPK)

table 6-26 AECPK registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	PK_CTRL00	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: Manual exposure[19:16]
0x01	PK_CTRL01	0x02	RW	Bit[7:0]: Manual exposure[15:8]
0x02	PK_CTRL02	0x00	RW	Bit[7:0]: Manual exposure[7:0]
0x03	PK_CTRL03	0x00	RW	Bit[7:6]: Reserved Bit[5]: Gain delay option 0: 1 frame latch 1: Delay 1 frame latch Bit[4]: Choose delay option 0: Delay disable 1: Delay enable Bit[3]: Reserved Bit[2]: Enable VTS manual mode Bit[1]: Enable AGC manual mode Bit[0]: Enable AEC manual mode
0x04	PK_CTRL04	0x00	RW	Bit[7:2]: Reserved Bit[1:0]: Manual sensor gain[9:8]
0x05	PK_CTRL05	0x00	RW	Bit[7:0]: Manual sensor gain[7:0]
0x09	PK_CTRL09	0x10	RW	Bit[7]: Reserved Bit[6:5]: Real gain output sel 00: For gain no delay 01: For delay 1 frame latch 10: For delay 2 frame latch 11: For no delay Bit[4]: Enable gain convert Bit[3]: Enable manual gain mode Bit[2]: Reserved Bit[1:0]: Real gain to gain convert module select 00 For gain no delay 01: For delay 1 frame latch 10: For delay 2 frame latch 11: For no delay
0x0A	PK_CTRL0A	0x00	RW	Bit[7:3]: Reserved Bit[2:0]: Manual real gain[10:8]
0x0B	PK_CTRL0B	0x10	RW	Bit[7:0]: Manual real gain[7:0]

table 6-26 AECPK registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x10	PK_CTRL10	–	R	Bit[7:3]: Reserved Bit[2:0]: Real gain for reading out
0x11	PK_CTRL11	–	R	Bit[7:0]: Real gain for reading out
0x12	PK_CTRL12	–	R	Bit[1:0]: Sensor gain for reading out
0x13	PK_CTRL13	–	R	Bit[7:0]: Sensor gain for reading out

6.26 frame control

- base address: 0x80166000 (DP0_FC0)
- base address: 0x80166100 (DP1_FC0)
- base address: 0x80166200 (MSNR_FC0)
- base address: 0x80166300 (DP0_FC1)
- base address: 0x80166400 (DP1_FC1)

table 6-27 frame control registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	R0	0x08	RW	Bit[7:4]: Reserved Bit[3]: SOF after line0 Bit[2]: Fcnt EOF select Bit[1]: Fcnt mask disable Bit[0]: Fcnt reset
0x01	R1	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: Frame on number
0x02	R2	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: Frame off number
0x03	R3	0x80	RW	Bit[7]: Zero line mask disable Bit[6]: Rblue mask disable Bit[5]: Data mask disable Bit[4]: Valid mask disable Bit[3]: HREF mask disable Bit[2]: EOF mask disable Bit[1]: SOF mask disable Bit[0]: All mask disable

table 6-27 frame control registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x04	R4	0x00	RW	Bit[1]: Mask first Bit[0]: Once mode (drop or burst frames) 8 mode descriptions: frame_on_number = m frame_off_number = n (1) mode0 m>0, n=0, mask first =0, once mode=0 (2) mode1 m=0, n>0, mask first =0, once mode=0 (3) mode2 m>0, n>0, mask first =0, once mode=1 (4) mode3 m>0, n>0, mask first =1, once mode=1 (5) mode4 m>0, n>0, mask first =1, once mode=0 (6) mode5 m>0, n>0, mask first =0, once mode=0 (7) mode 6 m=0, n=0, mask first =0, once mode=0 (8) mode 7 m=0, n=f, mask first =0, once mode=0

6.27 MIPI_RX

- base address: 0x80166600 (DP0_MIPI_RX)
- base address: 0x80166700 (DP1_MIPI_RX)
- base address: 0x80166800 (MSNR_MIPI_RX)

table 6-28 MIPI_RX registers (sheet 1 of 6)

address offset	register name	reset value	R/W	description
0x00	R0	0x80	RW	Bit[7]: Four lane select Bit[6]: Bypass ECC Bit[5]: Two lane select Bit[4]: Packet header bit order Bit[3:2]: Packet header byte order Bit[1]: Href select for virtual channel1 Bit[0]: Href select for virtual channel0
0x01	R1	0x04	RW	Control Read Start Bytes for Virtual Channel0
0x02	R2	0x00	RW	Bit[7]: SOT 1 bit wrong match enable for lane0 Bit[6]: Data type select for virtual channel0 Bit[5:0]: Data type set by registers for virtual channel0
0x03	R3	0x04	RW	Counter for Asserting VSYNC Signal After Frame End Packet in Virtual Channel0
0x04	R4	0x04	RW	Counter for De-asserting VSYNC Signal After Frame Start Packet in Virtual Channel0
0x05	R5	0x04	RW	Control Read Start Bytes for Virtual Channel1
0x06	R6	0x00	RW	SOT 1 Bit Wrong Match Enable for Lane1 Data Type Select for Virtual Channel1 Data Type Set by Registers for Virtual Channel1
0x07	R7	0x04	RW	Counter for Asserting VSYNC Signal After Frame End Packet in Virtual Channel1
0x08	R8	0x04	RW	Counter for De-asserting VSYNC Signal After Frame Start Packet in Virtual Channel1
0x09	R9	0x04	RW	Bit[7]: Sync data valid disable Bit[6]: Last line data clear to 0 enable Bit[5:4]: Reserved Bit[3:2]: VC1 value Bit[1:0]: VC0 value
0x0A~ 0x0F	RSVD	–	–	Reserved

table 6-28 MIPI_RX registers (sheet 2 of 6)

address offset	register name	reset value	R/W	description
0x10	R10	0x0F	RW	Bit[7:6]: Reserved Bit[5]: Disable VSYNC control for HREF and valid in virtual channel1 Bit[4]: Disable VSYNC control for HREF and valid in virtual channel0 Bit[3]: Enable VSYNC posedge reset in virtual channel1 async FIFO Bit[2]: Enable VSYNC negedge reset in virtual channel1 async FIFO Bit[1]: Enable VSYNC posedge reset in virtual channel0 async FIFO Bit[0]: Enable VSYNC negedge reset in virtual channel0 async FIFO
0x11	R11	0xE4	R	Bit[7:6]: Lane swap control for lane 3 Bit[5:4]: Lane swap control for lane 3 Bit[3:2]: Lane swap control for lane 3 Bit[1:0]: Lane swap control for lane 3
0x12	R12	–	R	Bit[7:5]: Reserved Bit[4]: Out swap control Bit[3]: Reserved Bit[2]: Four word output control Bit[1]: Reserved Bit[0]: Two word output control
0x13	R13	0x13	R	Bit[7:3]: Reserved Bit[2]: Shift 4 bits Bit[1]: Reserved Bit[0]: shift 2 bits
0x14	R14	–	R	PRBS Error Counter for Lane0
0x15	R15	–	R	PRBS Error Counter for Lane1
0x16	R16	–	R	PRBS Error Counter for Lane2
0x17	R17	–	R	PRBS Error Counter for Lane3
0x18	R18	–	R	High Byte of Pixel Number For Virtual Channel2
0x19	R19	–	R	Low Byte of Pixel Number for Virtual Channel2
0x1A	R1A	–	R	High Byte of Line Number for Virtual Channel2
0x1B	R1B	–	R	Low Byte of Line Number for Virtual Channel2
0x1C	R1C	–	R	High Byte of Pixel Number For Virtual Channel2
0x1D	R1D	–	R	Low Byte of Pixel Number for Virtual Channel2
0x1E	R1E	–	R	High Byte of Line Number for Virtual Channel2
0x1F	R1F	–	R	Low Byte of Line Number for Virtual Channel2

table 6-28 MIPI_RX registers (sheet 3 of 6)

address offset	register name	reset value	R/W	description
0x20	R20	0x00	RW	Bit[7:3]: Reserved Bit[2]: PRBS error clear Bit[1]: PRBS in/out select Bit[0]: PRBS enable
0x21	R21	0x00	RW	Bit[7:0]: Clear error status[7:0]
0x22	R22	0x00	RW	Bit[7:0]: Clear error status[15:8]
0x23	R23	0x00	RW	Bit[7:0]: PRBS test counter[7:0]
0x24	R24	0x00	RW	Bit[7:0]: PRBS test counter[15:8]
0x25	R25	0x00	RW	Bit[7:0]: PRBS test counter[23:16]
0x26	R26	0x40	RW	PRBS Error Counter
0x27	R27	0x10	RW	Bit[7:0]: Error detected enable[7:0]
0x28	R28	0x00	RW	Bit[7:0]: Error detected enable[15:8]
0x29	R29	0x00	RW	Bit[7:0]: Clear error status[23:16]
0x2A	R2A	0x00	RW	Bit[7:0]: Error detected enable[23:16]
0x2B	R2B	0x00	RW	Bit[7:0]: Clear error status[25:24]
0x2C	R2C	0x48	RW	Bit[7:2]: Reserved Bit[1:0]: Error interrupt enable[25:24]
0x2D	R2D	0x12	RW	Bit[7]: Embed line enable Bit[6]: VSYNC change to 1 only for SOF function enable Bit[5:0]: Embed data ID
0x2E	R2E	0x2C	RW	Bit[7:6]: Reserved Bit[5:0]: Embed data type
0x2F	RSVD	–	–	Reserved
0x30	R30	–	R	Bit[7:0]: Error status[7:0]
0x31	R31	–	R	Bit[7:0]: Error status[15:8]
0x32	R32	–	R	Bit[7:0]: Error status[23:16]
0x33	R33	–	R	Bit[7]: PRBS fail in lane0 Bit[6]: PRBS fail in lane1 Bit[5]: PRBS fail in lane2 Bit[4]: PRBS fail in lane3 Bit[3]: PRBS succeed in lane0 Bit[2]: PRBS succeed in lane1 Bit[1]: PRBS succeed in lane2 Bit[0]: PRBS succeed in lane3

table 6-28 MIPI_RX registers (sheet 4 of 6)

address offset	register name	reset value	R/W	description
0x34	R34	–	R	Bit[7:2]: Reserved Bit[1:0]: PRBS test status
0x35~ 0x37	RSVD	–	–	Reserved
0x38	R38	–	R	Bit[7:0]: Pixel number in one line for virtual channel 0[7:0]
0x39	R39	–	R	Bit[7:0]: Pixel number in one line for virtual channel 0[15:8]
0x3A	R3A	–	R	Bit[7:0]: Line number in one frame for virtual channel 0[7:0]
0x3B	R3B	–	R	Bit[7:0]: Line number in one frame for virtual channel 0[15:8]
0x3C	R3C	–	R	Bit[7:0]: Pixel number in one line for virtual channel 1[7:0]
0x3D	R3D	–	R	Bit[7:0]: Pixel number in one line for virtual channel 1[15:8]
0x3E	R3E	–	R	Bit[7:0]: Line number in one frame for virtual channel 1[7:0]
0x3F	R3F	–	R	Bit[7:0]: Line number in one frame for virtual channel 1[15:8]
0x40	R40	0x12	RW	Bit[7]: Embed counter mode enable Bit[6]: Reserved Bit[5:3]: Header embedded line number Bit[2:0]: Reserved
0x41	RSVD	–	–	Reserved
0x42	R42	0x00	RW	Bit[7:0]: Image Vsize[15:8]
0x43	R43	0x00	RW	Bit[7:0]: Image Vsize[7:0]
0x44	R44	0x00	RW	Bit[7:0]: Interrupt enable[7:0]
0x45	R45	0x00	RW	Bit[7:0]: Interrupt enable[15:8]
0x46	R46	0x00	RW	Bit[7:0]: Interrupt enable[23:16]
0x47	R47	0x00	RW	Bit[7:0]: Interrupt enable[31:24]
0x48	R48	–	W	Bit[7:0]: Interrupt clear[7:0]
0x49	R49	–	W	Bit[7:0]: Interrupt clear[15:8]
0x4A	R4A	–	W	Bit[7:0]: Interrupt clear[23:16]
0x4B	R4B	–	W	Bit[7:0]: Interrupt clear[31:24]

table 6-28 MIPI_RX registers (sheet 5 of 6)

address offset	register name	reset value	R/W	description
0x4C	R4C	–	R	Bit[7:0]: Interrupt value[7:0]
0x4D	R4D	–	R	Bit[7:0]: Interrupt value[15:8]
0x4E	R4E	–	R	Bit[7:0]: Interrupt value[23:16]
0x4F	R4F	–	R	Bit[7:0]: Interrupt value[31:24]
0x50	R50	0xB0	RW	Bit[7:6]: VC2 value Bit[5:4]: VC3 value Bit[3]: Fast AF data enable Bit[2]: Reserved Bit[1]: Row data enable Bit[0]: Column data enable
0x51	R51	0x2F	RW	Bit[7:6]: Fast AF VC value Bit[5:0]: Fast AF ID value
0x52	R52	0x2B	RW	Bit[7]: Fast AF ID match enable Bit[6]: Fast AF VC match enable Bit[5:0]: Fast AF data type
0x53	R53	0x30	RW	Bit[7:6]: Row VC value Bit[5:0]: Row ID value
0x54	R54	0x2B	RW	Bit[7]: Row ID match enable Bit[6]: Row VC match enable Bit[5:0]: Row data type
0x55	R55	0x31	RW	Bit[7:6]: Column VC value Bit[5:0]: Column ID value
0x56	R56	0x2B	RW	Bit[7]: Column ID match enable Bit[6]: Column VC match enable Bit[5:0]: Column data type
0x57	R57	0x00	RW	Bit[7:4]: Reserved Bit[3]: VC2 HREF start or end determined by line sync short packet enable Bit[2]: VC3 HREF start or end determined by line sync short packet enable Bit[1]: Disable VSYNC control for HREF and valid in virtual channel1 Bit[0]: Disable VSYNC control for HREF and valid in virtual channel0
0x58	R58	0x04	RW	Counter for Asserting VSYNC Signal After Frame End Packet in Virtual Channel2
0x59	R59	0x04	RW	Counter for Asserting VSYNC Signal After Frame End Packet in Virtual Channel2
0x5A	R5A	0x04	RW	Counter for Asserting VSYNC Signal After Frame End Packet in Virtual Channel3

table 6-28 MIPI_RX registers (sheet 6 of 6)

address offset	register name	reset value	R/W	description
0x5B	R5B	0x04	RW	Counter for Asserting VSYNC Signal After Frame End Packet in Virtual Channel3
0x5C	R5C	0x00	RW	Bit[7]: Embed data ID match disable Bit[6]: Embed data VC match disable Bit[5:4]: Embed data VC value Bit[3]: Image data must match enable Bit[2]: Reserved Bit[1:0]: Image VC value
0x5D	R5D	0x10	RW	Low Byte of Line Number Flag
0x5E	R5E	0x00	RW	High Byte of Line Number Flag
0x5F	R5F	0x11	RW	Bit[7:4]: SOF number in one frame Bit[3:0]: EOF number in one frame
0x60	R60	0x00	RW	Bit[7:4]: Debug bus select Bit[3]: EOF output select 0: VSYNC posedge after SOF 1: VSYNC posedge Bit[2]: Information write early in async FIFO Bit[1]: Long HREF length Bit[0]: SOF output select 0: Image SOF and EOF 1: Counter SOF and EOF
0x61	R61	0x01	RW	Low Byte of Pixel Number Flag
0x62	R62	0x01	RW	High Byte of Pixel Number Flag

6.28 format

- base address: 0x80166900 (DP0_FORMAT)
- base address: 0x80166A00 (DP1_FORMAT)

table 6-29 format registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	FORMAT CTRL	0x1E	RW	Bit[7]: JPEG format input Bit[6]: Reverse Y/U order Bit[5:0]: Image format 0x1E: YUV422 0x2A: RAW8 0x2B: RAW10 0x2C: RAW12

table 6-29 format registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x01	FORMAT CTRL1	0x03	RW	Bit[7:3]: Reserved Bit[2]: 0: Use internal register as hsize and vsize 1: Use ISP input as hsize and vsize Bit[1]: Gate bypass 0: Disable bypass 1: Enable bypass Bit[0]: Crop bypass 0: Disable bypass 1: Enable bypass
0x02	H_START_CROP_LOW	0x00	RW	Bit[7:0]: Crop horizontal start[7:0]
0x03	H_START_CROP_HIGH	0x00	RW	Bit[7:0]: Crop horizontal start[15:8]
0x04	H_END_CROP_LOW	0x7F	RW	Bit[7:0]: Crop horizontal end[7:0]
0x05	H_END_CROP_HIGH	0x02	RW	Bit[7:0]: Crop horizontal end high
0x06	V_START_CROP_LOW	0x00	RW	Bit[7:0]: Crop vertical start[7:0]
0x07	V_START_CROP_HIGH	0x00	RW	Bit[7:0]: Crop vertical start[15:8]
0x08	V_END_CROP_LOW	0xDF	RW	Bit[7:0]: Crop verical end[7:0]
0x09	V_END_CROP_HIGH	0x01	RW	Bit[7:0]: Crop verical end[15:8]
0x0A	H_START_GATE_LOW	0x00	RW	Bit[7:0]: Gate horizontal start[7:0]
0x0B	H_START_GATE_HIGH	0x00	RW	Bit[7:0]: Gate horizontal start[15:8]
0x0C	H_END_GATE_LOW	0x7F	RW	Bit[7:0]: Gate horizontal end[7:0]
0x0D	H_END_GATE_HIGH	0x02	RW	Bit[7:0]: Gate horizontal end[15:8]
0x0E	V_START_GATE_LOW	0x00	RW	Bit[7:0]: Gate vertical start[7:0]
0x0F	V_START_GATE_HIGH	0x00	RW	Bit[7:0]: Gate vertical start[15:8]
0x10	V_END_GATE_LOW	0xDF	RW	Bit[7:0]: Gate verical end[7:0]
0x11	V_END_GATE_HIGH	0x01	RW	Bit[7:0]: Gate verical end[15:8]
0x12	GATE_PATTERN_LOW	0x00	RW	Bit[7:0]: Gate pattern[7:0]
0x13	GATE_PATTERN_HIGH	0x00	RW	Bit[7:0]: Gate pattern[15:8]
0x14	IMG_HSIZE_LOW	0x80	RW	Bit[7:0]: Internal image hsize[7:0]
0x15	IMG_HSIZE_HIGH	0x02	RW	Bit[7:0]: Internal image hsize[15:8]
0x16	IMG_VSIZE_LOW	0xE0	RW	Bit[7:0]: Internal image vsize[7:0]
0x17	IMG_VSIZE_HIGH	0x01	RW	Bit[7:0]: Internal image vsize[15:8]

6.29 VFIFO

- base address: 0x80166C00 (DP0_VFIFO)
- base address: 0x80166D00 (DP1_VFIFO)
- base address: 0x80166E00 (MSNR_VFIFO)

table 6-30 VFIFO registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	IMG_START_SIZEH	0x02	RW	Bit[7:0]: Image start size[15:8]
0x01	IMG_START_SIZEL	0x80	RW	Bit[7:0]: Image start size[7:0]
0x02	VFIFO CTRL0	0x02	RW	Bit[7:6]: Reserved Bit[5]: Underflow interrupt enable Bit[4]: Overflow interrupt enable Bit[3]: Debug bus selection Bit[2]: Reserved Bit[1]: Frame reset enable 0: Disable 1: Enable Bit[0]: Reserved
0x03	VFIFO CTRL1	0x01	RW	Bit[7]: JPEG image enable 0: Input image is no JPEG 1: Input image is JPEG Bit[6]: JPEG output mode 0: No add dummy byte 1: Add dummy byte in last long packet Bit[5]: Reserved Bit[4]: Manual start size mode 0: Auto set start size 1: Manual set start size Bit[3:2]: Reserved Bit[1:0]: Start offset
0x04	VFIFO STAT	–	R	Bit[7:4]: Reserved Bit[3]: VFIFO RAM full Bit[2]: VFIFO RAM empty Bit[1]: FO full Bit[0]: FO empty
0x05	JPEG PACKET LOW	0x40	RW	Bit[7:0]: JPEG long packet size[7:0]
0x06	JPEG PACKET HIGH	0x00	RW	Bit[7:0]: JPEG long packet size[15:8]
0x07	SNR-EMB START SIZE LOW	0x80	RW	Bit[7:0]: Sensor embline start size[7:0]
0x08	SNR-EMB START SIZE HIGH	0x02	RW	Bit[7:0]: Sensor embline start size[15:8]
0x09	683-EMB START SIZE LOW	0x80	RW	Bit[7:0]: OV683 embline start size[7:0]
0x0A	683-EMB START SIZE HIGH	0x02	RW	Bit[7:0]: OV683 embline start size[15:8]

table 6-30 VFIFO registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x0B	FAF START SIZE LOW	0x80	RW	Bit[7:0]: Fast AF start size[7:0]
0x0C	FAF START SIZE HIGH	0x02	R	Bit[7:0]: Fast AF start size[15:8]
0x0D~ 0x0E	RSVD	–	–	Reserved
0x0F	PDAF PACKET SIZE HIGH	–	R	Bit[7:0]: PDAF long packet size[15:8]
0x10	PDAF PACKET SIZE LOW	–	R	Bit[7:0]: PDAF long packet size[7:0]

6.30 virtual channel control

- base address: 0x80166F00 (VC_CTRL0)
- base address: 0x80167000 (VC_CTRL1)

table 6-31 virtual channel control registers (sheet 1 of 8)

address offset	register name	reset value	R/W	description
0x00	CHN MUX CTRL0	0x00	RW	Bit[7]: Count PDAF Bit[6:4]: Debug bus selection Bit[3]: Reserved Bit[2]: EOF error auto clear enable 0: Disable 1: Enable Bit[1]: EOF error detection enable 0: Disable 1: Enable Bit[0]: Frame error detection enable 0: Disable 1: Enable

table 6-31 virtual channel control registers (sheet 2 of 8)

address offset	register name	reset value	R/W	description
0x01	CHN MUX CTRL1	0x00	RW	Bit[7]: VFIFO 3 enable 0: Disable 1: Enable Bit[6]: VFIFO 2 enable 0: Disable 1: Enable Bit[5]: VFIFO 1 enable 0: Disable 1: Enable Bit[4]: VFIFO 0 enable 0: Disable 1: Enable Bit[3]: VFIFO 3 test mode 0: Source from VFIFO 1: Source from colorbar Bit[2]: VFIFO 2 test mode 0: Source from VFIFO 1: Source from colorbar Bit[1]: VFIFO 1 test mode 0: Source from VFIFO 1: Source from colorbar Bit[0]: VFIFO 0 test mode 0: Source from VFIFO 1: Source from colorbar

table 6-31 virtual channel control registers (sheet 3 of 8)

address offset	register name	reset value	R/W	description
0x02	CHN MUX CTRL2	0x00	RW	Bit[7]: Channel 3 frame error interrupt enable 0: Disable 1: Enable Bit[6]: Channel 2 frame error interrupt enable 0: Disable 1: Enable Bit[5]: Channel 1 frame error interrupt enable 0: Disable 1: Enable Bit[4]: Channel 0 frame error interrupt enable 0: Disable 1: Enable Bit[3]: Channel 3 EOF error interrupt enable 0: Disable 1: Enable Bit[2]: Channel 2 EOF error interrupt enable 0: Disable 1: Enable Bit[1]: Channel 1 EOF error interrupt enable 0: Disable 1: Enable Bit[0]: Channel 0 EOF error interrupt enable 0: Disable 1: Enable
0x03	CHN MUX CTRL3	0xE4	RW	Bit[7:6]: VFIFO 3 image channel selection Bit[5:4]: VFIFO 2 image channel selection Bit[3:2]: VFIFO 1 image channel selection Bit[1:0]: VFIFO 0 image channel selection
0x04	CHN MUX CTRL4	0xE4	RW	Bit[7:6]: VFIFO 3 snr-emb channel selection Bit[5:4]: VFIFO 2 snr-emb channel selection Bit[3:2]: VFIFO 1 snr-emb channel selection Bit[1:0]: VFIFO 0 snr-emb channel selection
0x05	CHN MUX CTRL5	0xE4	RW	Bit[7:6]: VFIFO 3 683-emb channel selection Bit[5:4]: VFIFO 2 683-emb channel selection Bit[3:2]: VFIFO 1 683-emb channel selection Bit[1:0]: VFIFO 0 683-emb channel selection

table 6-31 virtual channel control registers (sheet 4 of 8)

address offset	register name	reset value	R/W	description
0x06	CHN MUX CTRL6	0xE4	RW	Bit[7:6]: VFIFO 3 fast AF raw channel selection Bit[5:4]: VFIFO 2 fast AF raw channel selection Bit[3:2]: VFIFO 1 fast AF raw channel selection Bit[1:0]: VFIFO 0 fast AF raw channel selection
0x07	CHN MUX CTRL7	0xE4	RW	Bit[7:6]: VFIFO 3 fast AF statistics channel selection Bit[5:4]: VFIFO 2 fast AF statistics channel selection Bit[3:2]: VFIFO 1 fast AF statistics channel selection Bit[1:0]: VFIFO 0 fast AF statistics channel selection
0x08	CHN MUX CTRL8	0xE4	RW	Bit[7:6]: VFIFO 3 PDAF row channel selection Bit[5:4]: VFIFO 2 PDAF row channel selection Bit[3:2]: VFIFO 1 PDAF row channel selection Bit[1:0]: VFIFO 0 PDAF row channel selection
0x09	CHN MUX CTRL9	0xE4	RW	Bit[7:6]: VFIFO 3 PDAF colum channel selection Bit[5:4]: VFIFO 2 PDAF colum channel selection Bit[3:2]: VFIFO 1 PDAF colum channel selection Bit[1:0]: VFIFO 0 PDAF colum channel selection
0x0A	CHN MUX CTRL10	0x00	RW	Bit[7:4]: VFIFO3 image data type ID Bit[3:0]: VFIFO2 image data type ID
0x0B	CHN MUX CTRL11	0x00	RW	Bit[7:4]: VFIFO1 image data type ID Bit[3:0]: VFIFO0 image data type ID
0x0C	CHN MUX CTRL12	0x11	RW	Bit[7:4]: VFIFO3 snr-emb data type ID Bit[3:0]: VFIFO2 snr-emb data type ID
0x0D	CHN MUX CTRL13	0x11	RW	Bit[7:4]: VFIFO1 snr-emb data type ID Bit[3:0]: VFIFO0 snr-emb data type ID
0x0E	CHN MUX CTRL14	0x22	RW	Bit[7:4]: VFIFO3 683-emb data type ID Bit[3:0]: VFIFO2 683-emb data type ID
0x0F	CHN MUX CTRL15	0x22	RW	Bit[7:4]: VFIFO1 683-emb data type ID Bit[3:0]: VFIFO0 683-emb data type ID

table 6-31 virtual channel control registers (sheet 5 of 8)

address offset	register name	reset value	R/W	description
0x10	CHN MUX CTRL16	0x33	RW	Bit[7:4]: VFIFO3 fast AF raw data type ID Bit[3:0]: VFIFO2 fast AF raw data type ID
0x11	CHN MUX CTRL17	0x33	RW	Bit[7:4]: VFIFO1 fast AF raw data type ID Bit[3:0]: VFIFO0 fast AF raw data type ID
0x12	CHN MUX CTRL18	0x44	RW	Bit[7:4]: VFIFO3 fast AF statistics data type ID Bit[3:0]: VFIFO2 fast AF statistics data type ID
0x13	CHN MUX CTRL19	0x44	RW	Bit[7:4]: VFIFO1 fast AF statistics data type ID Bit[3:0]: VFIFO0 fast AF statistics data type ID
0x14	CHN MUX CTRL20	0x55	RW	Bit[7:4]: VFIFO3 PDAF row data type ID Bit[3:0]: VFIFO2 PDAF row data type ID
0x15	CHN MUX CTRL21	0x55	RW	Bit[7:4]: VFIFO1 PDAF row data type ID Bit[3:0]: VFIFO0 PDAF row data type ID
0x16	CHN MUX CTRL22	0x66	RW	Bit[7:4]: VFIFO3 PDAF colum data type ID Bit[3:0]: VFIFO2 PDAF colum data type ID
0x17	CHN MUX CTRL23	0x66	RW	Bit[7:4]: VFIFO1 PDAF colum data type ID Bit[3:0]: VFIFO0 PDAF colum data type ID
0x18	CHN MUX CTRL24	0x01	RW	Bit[7:0]: Channel 0 vertical size[15:8]
0x19	CHN MUX CTRL25	0xE0	RW	Bit[7:0]: Channel 0 vertical size[7:0]
0x1A	CHN MUX CTRL26	0x01	RW	Bit[7:0]: Channel 1 vertical size[15:8]
0x1B	CHN MUX CTRL27	0xE0	RW	Bit[7:0]: Channel 1 vertical size[7:0]
0x1C	CHN MUX CTRL28	0x01	RW	Bit[7:0]: Channel 2 vertical size[15:8]
0x1D	CHN MUX CTRL29	0xE0	RW	Bit[7:0]: Channel 2 vertical size[7:0]
0x1E	CHN MUX CTRL30	0x01	RW	Bit[7:0]: Channel 3 vertical size[15:8]
0x1F	CHN MUX CTRL31	0xE0	RW	Bit[7:0]: Channel 3 vertical size[7:0]

table 6-31 virtual channel control registers (sheet 6 of 8)

address offset	register name	reset value	R/W	description
0x20	SBS CTRL0	0x00	RW	Bit[7:5]: Reserved Bit[4]: SBS error send extra ready enable 0: Disable 1: Enable Bit[3]: SBS first image 0: VFIFO 1 image output first 1: VFIFO 2 image output first Bit[2]: SBS embline sof selection 0: VFIFO 1 embline SOF 1: VFIFO 2 embline SOF Bit[1]: SBS sof selection 0: VFIFO 1 SOF 1: VFIFO 2 SOF Bit[0]: SBS enable 0: Disable SBS function 1: Enable SBS function
0x21	SBS CTRL1	0x00	RW	Bit[7:5]: Reserved Bit[4]: SBS output single image enable 0: Output each single image 1: Output SBS image only Bit[3]: Insert dummy data 2 in SBS image 0: Do not insert dummy data 2 in SBS line 1: Insert dummy data 2 in SBS line Bit[2]: Insert dummy data 1 in SBS image 0: Do not insert dummy data 1 in SBS line 1: Insert dummy data 1 in SBS line Bit[1]: Dispose SBS error 0: Don't care SBS error 1: Enable to dispose SBS error Bit[0]: SBS error detect enable 0: Don't detect SBS error 1: Enable to SBS error
0x22	SBS CTRL2	0x02	RW	Bit[7:0]: Image width[15:8]
0x23	SBS CTRL3	0x80	RW	Bit[7:0]: Image width[7:0]
0x24	SBS CTRL4	0x01	RW	Bit[7:0]: Image height[15:8]
0x25	SBS CTRL5	0xE0	RW	Bit[7:0]: Image height[7:0]
0x26	SBS CTRL6	0xEA	RW	Bit[7:6]: SBS channel number Bit[5:0]: SBS data identifier
0x27	SBS CTRL7	0x00	RW	Bit[7:0]: SOF delay cycle number[15:8]
0x28	SBS CTRL8	0x0A	RW	Bit[7:0]: SOF delay cycle number[7:0]
0x29	SBS CTRL9	0x00	RW	Bit[7:0]: Embline SOF delay cycle number[15:8]

table 6-31 virtual channel control registers (sheet 7 of 8)

address offset	register name	reset value	R/W	description	
0x2A	SBS CTRL10	0x0A	RW	Bit[7:0]:	Embline SOF delay cycle number[7:0]
0x2B	SBS CTRL11	0x00	RW	Bit[7:0]:	Dummy data start point 1[15:8]
0x2C	SBS CTRL12	0x00	RW	Bit[7:0]:	Dummy data start point 1[7:0]
0x2D	SBS CTRL13	0x00	RW	Bit[7:0]:	Dummy data start point 2[15:8]
0x2E	SBS CTRL14	0x00	RW	Bit[7:0]:	Dummy data start point 2[7:0]
0x2F	SBS CTRL15	0x00	RW	Bit[7:0]:	Dummy length 1
0x30	SBS CTRL16	0x00	RW	Bit[7:0]:	Dummy length 2
0x31	SBS CTRL17	0x00	RW	Bit[7:0]:	Dummy data 1[47:40]
0x32	SBS CTRL18	0x00	RW	Bit[7:0]:	Dummy data 1[39:32]
0x33	SBS CTRL19	0x00	RW	Bit[7:0]:	Dummy data 1[31:24]
0x34	SBS CTRL20	0x00	RW	Bit[7:0]:	Dummy data 1[23:16]
0x35	SBS CTRL21	0x00	RW	Bit[7:0]:	Dummy data 1[15:8]
0x36	SBS CTRL22	0x00	RW	Bit[7:0]:	Dummy data 1[7:40]
0x37	SBS CTRL23	0x00	RW	Bit[7:0]:	Dummy data 2[47:40]
0x38	SBS CTRL24	0x00	RW	Bit[7:0]:	Dummy data 2[39:32]
0x39	SBS CTRL25	0x00	RW	Bit[7:0]:	Dummy data 2[31:24]
0x3A	SBS CTRL26	0x00	RW	Bit[7:0]:	Dummy data 2[23:16]
0x3B	SBS CTRL27	0x00	RW	Bit[7:0]:	Dummy data 2[15:8]
0x3C	SBS CTRL28	0x00	RW	Bit[7:0]:	Dummy data 2[7:40]
0x3D	CHN MUX INT	0x00	RW	Bit[7]: Bit[6]: Bit[5]: Bit[4]: Bit[3]: Bit[2]: Bit[1]: Bit[0]:	Channel 3 frame error interrupt status / clear Channel 2 frame error interrupt status / clear Channel 1 frame error interrupt status / clear Channel 0 frame error interrupt status / clear Channel 3 EOF error interrupt status / clear Channel 2 EOF error interrupt status / clear Channel 1 EOF error interrupt status / clear Channel 0 EOF error interrupt status / clear

table 6-31 virtual channel control registers (sheet 8 of 8)

address offset	register name	reset value	R/W	description
0x3E	CHN MUX EOF ERR		RW	Bit[7:4]: Reserved Bit[3]: Detect channel 3 EOF less than expected before next SOF Bit[2]: Detect channel 2 EOF less than expected before next SOF Bit[1]: Detect channel 1 EOF less than expected before next SOF Bit[0]: Detect channel 0 EOF less then expected before next SOF

6.31 colorbar

- base address: 0x80167100 (DP0_CB)
- base address: 0x80167200 (DP1_CB)
- base address: 0x80167300 (MSNR_CB)

table 6-32 colorbar registers

address	register name	default value	R/W	description
0x00	CB STYLE CONTROL	0x00	RW	Colorbar Control Bit[7]: Data value increasing Bit[6]: All data value is 1 Bit[5:4]: Pixel rate Bit[3]: Reserved Bit[2]: Colorbar roll enable Bit[1]: Colorbar 75% select 0: 100% 1: 75% Bit[0]: Colorbar enable
0x01	CB HSIZE	0x80	RW	Bit[7:0]: Colorbar hsize[7:0]
0x02	CB HSIZE	0x02	RW	Bit[7:4]: Reserved Bit[3:0]: Colorbar hsize[11:8]
0x03	CB VSIZEL	0xE0	RW	Bit[7:0]: Colorbar vsize[7:0]
0x04	CB VSIZEH	0x01	RW	Bit[7:4]: Reserved Bit[3:0]: Colorbar vsize[11:8]
0x05	VDELAY	0x10	RW	Line Number from VSYNC to First HREF
0x06	VDMY	0x10	RW	Line Number from Last HREF to EOF
0x07	VBLANK	0x10	RW	Line Number from EOF to VSYNC
0x08	HBLANK	0x14	RW	Cycle Number When HREF Is Low

6.32 SPI_MS

- base address: 0x80167500 (set 0x80160200[16] to 1'b1)

table 6-33 SPI_MS registers (sheet 1 of 2)

address offset	register name	reset value	R/W	description
0x00	SPI_CTRL	0x18	RW	Bit[7:6]: Reserved Bit[5]: SPI master enable Bit[4]: SPI RX LSB first Bit[3]: SPI TX LSB first Bit[2]: SPI CPHA Bit[1]: SPI CPOL Bit[0]: SPI enable
0x01	SPI_START_CTRL	0x00	RW	Bit[7:3]: Reserved Bit[2]: SPI start Bit[1]: 1: SPI CS is not allowed to keep low during one block when TX FIFO empty or RX FIFO full 0: SPI CS is allowed to keep low during one block Bit[0]: 1: SPI CS is not allowed to keep low between blocks when TX FIFO empty or RX FIFO full 0: SPI CS is allowed to keep low between blocks
0x02	SPI_TX_SIZE LOW	0x00	RW	Bit[1:0]: SPI TX size in one block low byte
0x03	SPI_TX START SIZE LOW	0x00	RW	Bit[1:0]: SPI TX start size in one block low byte
0x04	SPI_RX_SIZE LOW	0x00	RW	Bit[1:0]: SPI RX size in one block low byte
0x05	SPI_RX START SIZE LOW	0x00	RW	Bit[1:0]: SPI RX start size in one block low byte
0x06	SPI_DIV_RATE	0x00	RW	Bit[7:0]: SPI clock dividing rate based on spi_clk_i, dividing from spi_clk_i/2 -> spi_clk_i/4 -> spi_clk_i/6 ...
0x07	SPI_CS_COUNTER	0x00	RW	Bit[3:0]: SPI CS counter Counting from half one cycle -> one cycle -> one by half cycle ...

table 6-33 SPI_MS registers (sheet 2 of 2)

address offset	register name	reset value	R/W	description
0x08	SPI_SLAVE_SEL	0x00	RW	Bit[2:0]: SPI slave selection
0x09	SPI_TX_SIZE HIGH	0x00	RW	Bit[1:0]: SPI TX size in one block high byte
0x0A	SPI_TX START SIZE HIGH	0x00	RW	Bit[1:0]: SPI TX start size in one block high byte
0x0B	SPI_RX_SIZE HIGH	0x00	RW	Bit[1:0]: SPI RX size in one block high byte
0x0C	SPI_RX START SIZE HIGH	0x00	RW	Bit[1:0]: SPI RX start size in one block high byte
0x10	SPI_TX WDATA	0x00	RW	Bit[7:0]: SPI TX FIFO write data
0x11	SPI_RX RDATA	–	R	Bit[7:0]: SPI RX FIFO read data
0x12	SPI_FIFOSTATUS	–	R	Bit[3]: RX FIFO full Bit[2]: RX FIFO empty Bit[1]: TX FIFO full Bit[0]: TX FIFO empty
0x20	SPI_ERR_INT_EN	0x00	RW	Bit[5]: SPI RX finish interrupt enable Bit[4]: SPI TX finish interrupt enable Bit[3]: SPI RX FIFO full interrupt enable Bit[2]: SPI RX FIFO empty interrupt enable Bit[1]: SPI TX FIFO full interrupt enable Bit[0]: SPI TX FIFO empty interrupt enable
0x21	SPI_ERR_INT_ST	–	R	Bit[5:0]: SPI error interrupt status
0x22	SPI_TX_FIFO_CNT	–	R	Bit[5:0]: Number of bytes exist in TX FIFO
0x23	SPI_RX_FIFO_CNT	–	R	Bit[5:0]: Number of bytes exist in RX FIFO

6.33 MIPI_TX

- base address: 0x80167600 (MIPI_TX0)
- base address: 0x80167700 (MIPI_TX1)

table 6-34 MIPI_TX registers (sheet 1 of 25)

address	register name	default value	R/W	description
0x00	CLK_POST_CONST_MIN	0x41	RW	T(clk_post) Constant Minimum Value
0x01	CLK_POST_UI_MIN	0x34	RW	T(clk_post) UI Minimum Value
0x02	CLK_TRAIL_CONST_MIN	0x41	RW	T(clk_trail) Constant Minimum Value
0x03	CLK_TRAIL_UI_MIN	0x00	RW	T(clk_trail) UI Minimum Value
0x04	CLK_PREPARE_CONST_MIN	0x32	RW	T(clk_prepare) Constant Minimum Value
0x05	CLK_PREPARE_UI_MIN	0x00	RW	T(clk_prepare) UI Minimum Value
0x06	CLK_ZERO_CONST_MIN1	2'b01	RW	T(clk_zero) Constant Minimum Value (high 2 bits)
0x07	CLK_ZERO_CONST_MIN2	0x2C	RW	T(clk_zero) Constant Minimum Value (low 8 bits)
0x08	CLK_ZERO_UI_MIN	0x00	RW	T(clk_zero) UI Minimum Value
0x09	HS_EXIT_CONST_MIN	0x6E	RW	T(hs_exit) Constant Minimum Value
0x0A	HS_EXIT_UI_MIN	0x00	RW	T(hs_exit) UI Minimum Value
0x0B	HS_PREPARE_CONST_MIN	0x37	RW	T(hs_prepare) Constant Minimum Value
0x0C	HS_PREPARE_UI_MIN	0x04	RW	T(hs_prepare) UI Minimum Value
0x0D	HS_ZERO_CONST_MIN	0x6E	RW	T(hs_zero) Constant Minimum Value
0x0E	HS_ZERO_UI_MIN	0x06	RW	T(hs_zero) UI Minimum Value
0x0F	HS_TRAIL_CONST_MIN	0x41	RW	T(hs_trail) Constant Minimum Value
0x10	HS_TRAIL_UI_MIN	0x04	RW	T(hs_trail) UI Minimum Value
0x11	LPX_CONST_MIN	0x37	RW	T(lpx) Constant Minimum Value
0x12	LPX_UI_MIN	0x00	RW	T(lpx) UI Minimum Value
0x13	MIPI_CLK_PERIOD1	2'b00	RW	Clock Period of mipi_clk Used to calculate timing parameters of MIPI TX (low 2 bits) This register should be updated when mipi_clk clock cycle is changed. MIPI_CLK default is 100MHz

table 6-34 MIPI_TX registers (sheet 2 of 25)

address	register name	default value	R/W	description
0x14	MIPI_CLK_PERIOD2	0x28	RW	Clock Period of mipi_clk Used to calculate timing parameters of MIPI TX (high 8 bits) This register should be updated when mipi_clk clock cycle is changed. MIPI_CLK default is 100MHz
0x15	MIPI_LANE_CTRL0	0x92	RW	Bit[7]: Finish lane transfer long packet data exactly 0: Maybe send any dummy data in data lane when image line byte number (add 2 CRC bytes) cannot be divided by lane number, meaning there will be one dummy byte in some data lanes 1: Do not send any dummy data in data lane, meaning trail data will be behind image data in each data lane Bit[6]: Clock lane data change 2'b01 -> 2'b10 0: Bit[1:0] 1: {bit[0],bit[1]} Bit[5]: dis_clk_lane (active high) Disable clock lane Bit[4]: Line_sync_en Insert LS/LE in MIPI TX stream if this bit is set Bit[3]: frame_cnt_zero_c1 MIPI TX channel 1 will keep frame counter zero if this bit is set Bit[2]: frame_cnt_zero_c0 MIPI TX channel 0 will keep frame counter zero if this bit is set Bit[1]: gate_clk_en2 (active high) Gate clock about clock lane when frame blanking time Bit[0]: gate_clk_en1 (active high) Gate clock about clock lane when line/frame blanking time

table 6-34 MIPI_TX registers (sheet 3 of 25)

address	register name	default value	R/W	description
0x16	MIPI_LPKT_MAN	0x00	RW	Bit[7]: MIPI data type tag enable (active high) Bit[6]: lpkt_man_en (active high) Long packet manual input mode enable Bit[5:0]: Manual data type
0x17	MIPI_SRC0_DI0	0x1E	RW	Bit[7:6]: Virtual channel ID of source 0 for virtual channel 0 Bit[5:0]: Data type of source 0 for virtual channel 0
0x18	MIPI_SRC0_DI1	0x5E	RW	Bit[7:6]: Virtual channel ID of source 0 for virtual channel 1 Bit[5:0]: Data type of source 0 for virtual channel 1
0x19	MIPI_SRC0_DI2	0xAA	RW	Bit[7:6]: Virtual channel ID of source 0 for virtual channel 2 Bit[5:0]: Data type of source 0 for virtual channel 2
0x1A	MIPI_SRC0_DI3	0xEA	RW	Bit[7:6]: Virtual channel ID of source 0 for virtual channel 3 Bit[5:0]: Data type of source 0 for virtual channel 3
0x1B	MIPI_SRC0_WIDTHH0	0x02	RW	Bit[7:0]: src0_hsize[15:8] for virtual channel 0
0x1C	MIPI_SRC0_WIDTHL0	0x80	RW	Bit[7:0]: src0_hsize[7:0] for virtual channel 0
0x1D	MIPI_SRC0_HEIGHTH0	0x01	RW	Bit[7:0]: src0_vsize[15:8] for virtual channel 0
0x1E	MIPI_SRC0_HEIGHTL0	0xE0	RW	Bit[7:0]: src0_vsize[7:0] for virtual channel 0
0x1F	MIPI_SRC0_WIDTHH1	0x02	RW	Bit[7:0]: src0_hsize[15:8] for virtual channel 1
0x20	MIPI_SRC0_WIDTHL1	0x80	RW	Bit[7:0]: src0_hsize[7:0] for virtual channel 1
0x21	MIPI_SRC0_HEIGHTH1	0x01	RW	Bit[7:0]: src0_vsize[15:8] for virtual channel 1
0x22	MIPI_SRC0_HEIGHTL1	0xE0	RW	Bit[7:0]: src0_vsize[7:0] for virtual channel 1
0x23	MIPI_SRC0_WIDTHH2	0x02	RW	Bit[7:0]: src0_hsize[15:8] for virtual channel 2

table 6-34 MIPI_TX registers (sheet 4 of 25)

address	register name	default value	R/W	description
0x24	MIPI_SRC0_WIDTHL2	0x80	RW	Bit[7:0]: src0_hsize[7:0] for virtual channel 2
0x25	MIPI_SRC0_HEIGHTH2	0x01	RW	Bit[7:0]: src0_vsize[15:8] for virtual channel 2
0x26	MIPI_SRC0_HEIGHTL2	0xE0	RW	Bit[7:0]: src0_vsize[7:0] for virtual channel 2
0x27	MIPI_SRC0_WIDTHH3	0x02	RW	Bit[7:0]: src0_hsize[15:8] for virtual channel 3
0x28	MIPI_SRC0_WIDTHL3	0x80	RW	Bit[7:0]: src0_hsize[7:0] for virtual channel 3
0x29	MIPI_SRC0_HEIGHTH3	0x01	RW	Bit[7:0]: src0_vsize[15:8] for virtual channel 3
0x2A	MIPI_SRC0_HEIGHTL3	0xE0	RW	Bit[7:0]: src0_vsize[7:0] for virtual channel 3
0x2B	MIPI_STATUS	–	R	Bit[7:2]: Reserved Bit[1]: mipi_ph_done MIPI has transfered long packet header data. User can modify data type. Bit[0]: mipi_busy MIPI is transmitting data if this bit is asserted

table 6-34 MIPI_TX registers (sheet 5 of 25)

address	register name	default value	R/W	description
0x2C	MIPI_LANE_CTRL1	0x0B	RW	Bit[7]: hs_zero_sync_en 0: Send 'hs_en' one cycle ahead of hs_zero state 1: Send 'hs_en' sync with hs_zero state Bit[6]: Send "FS" packet after MIPI receives SOF 0: Send "FS" packet when VFIFO data is ready 1: Send "FS" packet when MIPI receives SOF Bit[5]: Long packet chksum byte exchange enable 0: Chksum = CRC[15:0] 1: Chksum = {CRC[7:0], CRC[15:8]} Bit[4]: Clock/data lane valid signal adjust enable 0: Disable valid adjust, valid is same as hs_en 1: Enable valid adjust, valid will ahead/delay than hs_en signal Bit[3]: Low power state when data lane is idle 0: Low power signal for each data lane may be controlled by lane_en 1: lp_p or lp_n will stay "1" if current data lane is not active Bit[2]: pclk_inv_en (active high) PCLK inverse enable (output to PHY) Bit[1]: gen_fe1_en Force to generate frame end short packet in channel 1 when MIPI TX has transmitted one line if VFIFO of this channel is overflow Bit[0]: gen_fe0_en Force to generate frame end short packet in channel 0 when MIPI TX has transmitted one line if VFIFO of this channel is overflow

table 6-34 MIPI_TX registers (sheet 6 of 25)

address	register name	default value	R/W	description
0x2D	MIPI_LANE_CTRL2	0x00	RW	Bit[7]: Data lane 4 inverse enable Bit[6]: Data lane 3 inverse enable Bit[5]: Data lane 2 inverse enable Bit[4]: Data lane 1 inverse enable Bit[3]: lane4_en Data lane 4 enable Bit[2]: lane3_en Data lane 3 enable Bit[1]: lane2_en Data lane 2 enable Bit[0]: lane1_en Data lane 1 enable
0x2E	MIPI_LANE_CTRL3	0x03	RW	Bit[7]: Channel 3 crop enable 0: Channel 3 crop disable 1: Channel 3 crop enable Bit[6]: Channel 2 crop enable 0: Channel 2 crop disable 1: Channel 2 crop enable Bit[5]: Channel 1 crop enable 0: Channel 1 crop disable 1: Channel 1 crop enable Bit[4]: Channel 0 crop enable 0: Channel 0 crop disable 1: Channel 0 crop enable Bit[3]: frame_cnt_zero_c3 MIPI TX channel 3 will keep frame counter zero if this bit is set Bit[2]: frame_cnt_zero_c2 MIPI TX channel 2 will keep frame counter zero if this bit is set Bit[1]: gen_fe3_en Force to generate frame end short packet in channel 3 when MIPI TX has transmitted one line if VFIFO of this channel is overflow Bit[0]: gen_fe2_en Force to generate frame end short packet in channel 2 when MIPI TX has transmitted one line if VFIFO of this channel is overflow

table 6-34 MIPI_TX registers (sheet 7 of 25)

address	register name	default value	R/W	description
0x2F	MIPI_TEST	0x00	RW	Bit[7:6]: Reserved Bit[5]: lp_n_man_data Output manual lower power data in MIPI TX test mode Bit[4]: lp_p_man_data Output manual lower power data in MIPI TX test mode Bit[3]: lp_man_en Output manual low power data enable in low power test mode and de-assert high speed signal (hs_en or valid) Bit[2]: hs_man_en Manual test data will output to PHY when this bit is set and MIPI TX in high speed test mode (register 0x25 must be set in this case) Bit[1]: test_mode 0: Test start point sync by mipi_test 1: Test start point sync by MIPI RX prbs_en Bit[0]: mipi_test (active high) Test MIPI TX and RX PHY
0x30	MAN_DATA	0xFF	RW	Bit[7:0]: Manual test data for MIPI PHY
0x31	MAX_FCNT_HC0	0xFF	RW	Bit[7:0]: Maximum frame counter of channel 0[15:8]
0x32	MAX_FCNT_LC0	0xFF	RW	Bit[7:0]: Maximum frame counter of channel 0[7:0]
0x33	MAX_FCNT_HC1	0xFF	RW	Bit[7:0]: Maximum frame counter of channel 1[15:8]
0x34	MAX_FCNT_LC1	0xFF	RW	Bit[7:0]: Maximum frame counter of channel 1[7:0]
0x35	MAX_FCNT_HC2	0xFF	RW	Bit[7:0]: Maximum frame counter of channel 2[15:8]
0x36	MAX_FCNT_LC2	0xFF	RW	Bit[7:0]: Maximum frame counter of channel 2[7:0]
0x37	MAX_FCNT_HC3	0xFF	RW	Bit[7:0]: Maximum frame counter of channel 3[15:8]
0x38	MAX_FCNT_LC3	0xFF	RW	Bit[7:0]: Maximum frame counter of channel 3[7:0]
0x39	YUV422_12B_DT	0x1B	RW	Bit[5:0]: YUV422_12b data type

table 6-34 MIPI_TX registers (sheet 8 of 25)

address	register name	default value	R/W	description
0x3A	YUV422_10B_DT	0x1F	RW	Bit[5:0]: YUV422_10b data type
0x3B	YUV422_8B_DT	0x1E	RW	Bit[5:0]: YUV422_8b data type
0x3C	RAW12_DT	0x2C	RW	Bit[5:0]: Raw12 data type
0x3D	RAW10_DT	0x2B	RW	Bit[5:0]: Raw10 data type
0x3E	RAW8_DT	0x2A	RW	Bit[5:0]: Raw8 data type
0x3F	RAW6_DT	0x28	RW	Bit[5:0]: Raw6 data type
0x40	RGB888_DT	0x24	RW	Bit[5:0]: RGB888 data type
0x41	RGB565_DT	0x22	RW	Bit[5:0]: RGB565 data type
0x42	JPEG_DT	0x30	RW	Bit[5:0]: JPEG data type
0x43	EMB_LINE_DT	0x12	RW	Bit[5:0]: emb_line data type
0x46	DATA_SEQ_CTRL	0x00	RW	Bit[7:5]: RGB sequence 000: 'BGR' sequence 001: 'BRG' sequence 010: 'GBR' sequence 011: 'GRB' sequence 100: 'RGB' sequence 101: 'RBG' sequence 110: Same as 3'b000 111: Same as 3'b000 Bit[4:3]: Quarter sequence Used to adjust each 12-bit data sequence 00: Data[11:0] 01: {data[9:0], data[11:10]} 10: {data[7:0], data[11:8]} 11: Same as 2'b00 Bit[2]: Low half sequence Used to adjust low 24-bit data sequence 0: Data[23:0] 1: {data[11:0], data[23:12]} Bit[1]: High half sequence Used to adjust low 24-bit data sequence 0: Data[47:24] 1: {data[35:24], data[47:36]} Bit[0]: Total sequence 0: Data[47:0] 1: {data[11:0], data[23:12], data[35:24], data[47:36]}
0x47	LP_DELAY	8'H04	RW	Bit[7:0]: LP00 to LP11 delay cycle when hs_zero sync enable

table 6-34 MIPI_TX registers (sheet 9 of 25)

address	register name	default value	R/W	description
0x48	INTER_CTRL0_CH0	0x00	RW	Bit[7]: Source 7 interleave enable in channel 0 Bit[6]: Source 6 interleave enable in channel 0 Bit[5]: Source 5 interleave enable in channel 0 Bit[4]: Source 4 interleave enable in channel 0 Bit[3]: Source 3 interleave enable in channel 0 Bit[2]: Source 2 interleave enable in channel 0 Bit[1]: Source 1 interleave enable in channel 0 Bit[0]: Source 0 interleave enable in channel 0
0x49	INTER_CTRL1_CH0	0x00	RW	Bit[7]: Interleave total enable in channel 0 Bit[6:3]: Reserved Bit[2]: Source a interleave enable in channel 0 Bit[1]: Source 9 interleave enable in channel 0 Bit[0]: Source 8 interleave enable in channel 0
0x4A	INTER_CTRL_CH1	0x00	RW	Bit[7]: Interleave total enable in channel 1 Bit[6:3]: Reserved Bit[2]: Source 2 interleave enable in channel 1 Bit[1]: Source 1 interleave enable in channel 1 Bit[0]: Source 0 interleave enable in channel 1
0x4B	INTER_CTRL_CH2	0x00	RW	Bit[7]: Interleave total enable in channel 2 Bit[6:3]: Reserved Bit[2]: Source 2 interleave enable in channel 2 Bit[1]: Source 1 interleave enable in channel 2 Bit[0]: Source 0 interleave enable in channel 2

table 6-34 MIPI_TX registers (sheet 10 of 25)

address	register name	default value	R/W	description
0x4C	INTER_CTRL_CH3	0x00	RW	Bit[7]: Interleave total enable in channel 3 Bit[6:3]: Reserved Bit[2]: Source 2 interleave enable in channel 3 Bit[1]: Source 1 interleave enable in channel 3 Bit[0]: Source 0 interleave enable in channel 3
0x4D	MIPI_SRC1_DI0	0x12	RW	Bit[7:6]: Virtual channel ID of source 1 for virtual channel 0 Bit[5:0]: Data type of source 1 for virtual channel 0
0x4E	MIPI_SRC1_DI1	0x52	RW	Bit[7:6]: Virtual channel ID of source 1 for virtual channel 1 Bit[5:0]: Data type of source 1 for virtual channel 1
0x4F	MIPI_SRC1_DI2	0x92	RW	Bit[7:6]: Virtual channel ID of source 1 for virtual channel 2 Bit[5:0]: Data type of source 1 for virtual channel 2
0x50	MIPI_SRC1_DI3	0xD2	RW	Bit[7:6]: Virtual channel ID of source 1 for virtual channel 3 Bit[5:0]: Data type of source 1 for virtual channel 3
0x51	MIPI_SRC1_WIDTHH0	0x02	RW	Bit[7:0]: src1_hsize[15:8] for virtual channel 0
0x52	MIPI_SRC1_WIDTHL0	0x80	RW	Bit[7:0]: src1_hsize[7:0] for virtual channel 0
0x53	MIPI_SRC1_HEIGHTH0	0x01	RW	Bit[7:0]: src1_vsize[15:8] for virtual channel 0
0x54	MIPI_SRC1_HEIGHTL0	0xE0	RW	Bit[7:0]: src1_vsize[7:0] for virtual channel 0
0x55	MIPI_SRC1_WIDTHH1	0x02	RW	Bit[7:0]: src1_hsize[15:8] for virtual channel 1
0x56	MIPI_SRC1_WIDTHL1	0x80	RW	Bit[7:0]: src1_hsize[7:0] for virtual channel 1
0x57	MIPI_SRC1_HEIGHTH1	0x01	RW	Bit[7:0]: src1_vsize[15:8] for virtual channel 1
0x58	MIPI_SRC1_HEIGHTL1	0xE0	RW	Bit[7:0]: src1_vsize[7:0] for virtual channel 1

table 6-34 MIPI_TX registers (sheet 11 of 25)

address	register name	default value	R/W	description
0x59	MIPI_SRC1_WIDTHH2	0x02	RW	Bit[7:0]: src1_hsize[15:8] for virtual channel 2
0x5A	MIPI_SRC1_WIDTHL2	0x80	RW	Bit[7:0]: src1_hsize[7:0] for virtual channel 2
0x5B	MIPI_SRC1_HEIGHTH2	0x01	RW	Bit[7:0]: src1_vsize[15:8] for virtual channel 2
0x5C	MIPI_SRC1_HEIGHTL2	0xE0	RW	Bit[7:0]: src1_vsize[7:0] for virtual channel 2
0x5D	MIPI_SRC1_WIDTHH3	0x02	RW	Bit[7:0]: src1_hsize[15:8] for virtual channel 3
0x5E	MIPI_SRC1_WIDTHL3	0x80	RW	Bit[7:0]: src1_hsize[7:0] for virtual channel 3
0x5F	MIPI_SRC1_HEIGHTH3	0x01	RW	Bit[7:0]: src1_vsize[15:8] for virtual channel 3
0x60	MIPI_SRC1_HEIGHTL3	0xE0	RW	Bit[7:0]: src1_vsize[7:0] for virtual channel 3
0x61	MIPI_SRC2_DI0	0x12	RW	Bit[7:6]: Virtual channel ID of source 2 for virtual channel 0 Bit[5:0]: Data type of source 2 for virtual channel 0
0x62	MIPI_SRC2_DI1	0x52	RW	Bit[7:6]: Virtual channel ID of source 2 for virtual channel 1 Bit[5:0]: Data type of source 2 for virtual channel 1
0x63	MIPI_SRC2_DI2	0x92	RW	Bit[7:6]: Virtual channel ID of source 2 for virtual channel 2 Bit[5:0]: Data type of source 2 for virtual channel 2
0x64	MIPI_SRC2_DI3	0xD2	RW	Bit[7:6]: Virtual channel ID of source 2 for virtual channel 3 Bit[5:0]: Data type of source 2 for virtual channel 3
0x65	MIPI_SRC2_WIDTHH0	0x02	RW	Bit[7:0]: src2_hsize[15:8] for virtual channel 0
0x66	MIPI_SRC2_WIDTHL0	0x80	RW	Bit[7:0]: src2_hsize[7:0] for virtual channel 0
0x67	MIPI_SRC2_HEIGHTH0	0x01	RW	Bit[7:0]: src2_vsize[15:8] for virtual channel 0
0x68	MIPI_SRC2_HEIGHTL0	0xE0	RW	Bit[7:0]: src2_vsize[7:0] for virtual channel 0

table 6-34 MIPI_TX registers (sheet 12 of 25)

address	register name	default value	R/W	description
0x69	MIPI_SRC2_WIDTHH1	0x02	RW	Bit[7:0]: src2_hsize[15:8] for virtual channel 1
0x6A	MIPI_SRC2_WIDTHL1	0x80	RW	Bit[7:0]: src2_hsize[7:0] for virtual channel 1
0x6B	MIPI_SRC2_HEIGHTH1	0x01	RW	Bit[7:0]: src2_vsize[15:8] for virtual channel 1
0x6C	MIPI_SRC2_HEIGHTL1	0xE0	RW	Bit[7:0]: src2_vsize[7:0] for virtual channel 1
0x6D	MIPI_SRC2_WIDTHH2	0x02	RW	Bit[7:0]: src2_hsize[15:8] for virtual channel 2
0x6E	MIPI_SRC2_WIDTHL2	0x80	RW	Bit[7:0]: src2_hsize[7:0] for virtual channel 2
0x6F	MIPI_SRC2_HEIGHTH2	0x01	RW	Bit[7:0]: src2_vsize[15:8] for virtual channel 2
0x70	MIPI_SRC2_HEIGHTL2	0xE0	RW	Bit[7:0]: src2_vsize[7:0] for virtual channel 2
0x71	MIPI_SRC2_WIDTHH3	0x02	RW	Bit[7:0]: src2_hsize[15:8] for virtual channel 3
0x72	MIPI_SRC2_WIDTHL3	0x80	RW	Bit[7:0]: src2_hsize[7:0] for virtual channel 3
0x73	MIPI_SRC2_HEIGHTH3	0x01	RW	Bit[7:0]: src2_vsize[15:8] for virtual channel 3
0x74	MIPI_SRC2_HEIGHTL3	0xE0	RW	Bit[7:0]: src2_vsize[7:0] for virtual channel 3
0x75	MIPI_SRC3_DI0	0x12	RW	Bit[7:6]: Virtual channel ID of source 2 for virtual channel 0 Bit[5:0]: Data type of source 2 for virtual channel 0
0x76	MIPI_SRC3_WIDTHH0	0x02	RW	Bit[7:0]: src3_hsize[15:8] for virtual channel 0
0x77	MIPI_SRC3_WIDTHL0	0x80	RW	Bit[7:0]: src3_hsize[7:0] for virtual channel 0
0x78	MIPI_SRC3_HEIGHTH0	0x01	RW	Bit[7:0]: src3_vsize[15:8] for virtual channel 0
0x79	MIPI_SRC3_HEIGHTL0	0xE0	RW	Bit[7:0]: src3_vsize[7:0] for virtual channel 0

table 6-34 MIPI_TX registers (sheet 13 of 25)

address	register name	default value	R/W	description
0x7A	MIPI_SRC4_DI0	0x12	RW	Bit[7:6]: Virtual channel ID of source 4 for virtual channel 0 Bit[5:0]: Data type of source 4 for virtual channel 0
0x7B	MIPI_SRC4_WIDTHH0	0x02	RW	Bit[7:0]: src4_hsize[15:8] for virtual channel 0
0x7C	MIPI_SRC4_WIDTHL0	0x80	RW	Bit[7:0]: src4_hsize[7:0] for virtual channel 0
0x7D	MIPI_SRC4_HEIGHTH0	0x01	RW	Bit[7:0]: src4_vsize[15:8] for virtual channel 0
0x7E	MIPI_SRC4_HEIGHTL0	0xE0	RW	Bit[7:0]: src4_vsize[7:0] for virtual channel 0
0x7F	MIPI_SRC5_DI0	0x12	RW	Bit[7:6]: Virtual channel ID of source 5 for virtual channel 0 Bit[5:0]: Data type of source 5 for virtual channel 0
0x80	MIPI_SRC5_WIDTHH0	0x02	RW	Bit[7:0]: src5_hsize[15:8] for virtual channel 0
0x81	MIPI_SRC5_WIDTHL0	0x80	RW	Bit[7:0]: src5_hsize[7:0] for virtual channel 0
0x82	MIPI_SRC5_HEIGHTH0	0x01	RW	Bit[7:0]: src5_vsize[15:8] for virtual channel 0
0x83	MIPI_SRC5_HEIGHTL0	0xE0	RW	Bit[7:0]: src5_vsize[7:0] for virtual channel 0
0x84	MIPI_SRC6_DI0	0x12	RW	Bit[7:6]: Virtual channel ID of source 6 for virtual channel 0 Bit[5:0]: Data type of source 6 for virtual channel 0
0x85	MIPI_SRC6_WIDTHH0	0x02	RW	Bit[7:0]: src6_hsize[15:8] for virtual channel 0
0x86	MIPI_SRC6_WIDTHL0	0x80	RW	Bit[7:0]: src6_hsize[7:0] for virtual channel 0
0x87	MIPI_SRC6_HEIGHTH0	0x01	RW	Bit[7:0]: src6_vsize[15:8] for virtual channel 0
0x88	MIPI_SRC6_HEIGHTL0	0xE0	RW	Bit[7:0]: src6_vsize[7:0] for virtual channel 0
0x89	MIPI_SRC7_DI0	0x12	RW	Bit[7:6]: Virtual channel ID of source 7 for virtual channel 0 Bit[5:0]: Data type of source 7 for virtual channel 0

table 6-34 MIPI_TX registers (sheet 14 of 25)

address	register name	default value	R/W	description
0x8A	MIPI_SRC7_WIDTHH0	0x02	RW	Bit[7:0]: src7_hsize[15:8] for virtual channel 0
0x8B	MIPI_SRC7_WIDTHL0	0x80	RW	Bit[7:0]: src7_hsize[7:0] for virtual channel 0
0x8C	MIPI_SRC7_HEIGHTH0	0x01	RW	Bit[7:0]: src7_vsize[15:8] for virtual channel 0
0x8D	MIPI_SRC7_HEIGHTL0	0xE0	RW	Bit[7:0]: src7_vsize[7:0] for virtual channel 0
0x8E	MIPI_SRC8_DI0	0x12	RW	Bit[7:6]: Virtual channel ID of source 8 for virtual channel 0 Bit[5:0]: Data type of source 8 for virtual channel 0
0x8F	MIPI_SRC8_WIDTHH0	0x02	RW	Bit[7:0]: src8_hsize[15:8] for virtual channel 0
0x90	MIPI_SRC8_WIDTHL0	0x80	RW	Bit[7:0]: src8_hsize[7:0] for virtual channel 0
0x91	MIPI_SRC8_HEIGHTH0	0x01	RW	Bit[7:0]: src8_vsize[15:8] for virtual channel 0
0x92	MIPI_SRC8_HEIGHTL0	0xE0	RW	Bit[7:0]: src8_vsize[7:0] for virtual channel 0
0x93	MIPI_SRC9_DI0	0x12	RW	Bit[7:6]: Virtual channel ID of source 9 for virtual channel 0 Bit[5:0]: Data type of source 9 for virtual channel 0
0x94	MIPI_SRC9_WIDTHH0	0x02	RW	Bit[7:0]: src9_hsize[15:8] for virtual channel 0
0x95	MIPI_SRC9_WIDTHL0	0x80	RW	Bit[7:0]: src9_hsize[7:0] for virtual channel 0
0x96	MIPI_SRC9_HEIGHTH0	0x01	RW	Bit[7:0]: src9_vsize[15:8] for virtual channel 0
0x97	MIPI_SRC9_HEIGHTL0	0xE0	RW	Bit[7:0]: src9_vsize[7:0] for virtual channel 0
0x98	MIPI_SRCA_DI0	0x12	RW	Bit[7:6]: Virtual channel ID of source a for virtual channel 0 Bit[5:0]: Data type of source a for virtual channel 0
0x99	MIPI_SRCA_WIDTHH0	0x02	RW	Bit[7:0]: srca_hsize[15:8] for virtual channel 0

table 6-34 MIPI_TX registers (sheet 15 of 25)

address	register name	default value	R/W	description
0x9A	MIPI_SRCA_WIDTHL0	0x80	RW	Bit[7:0]: srca_hsize[7:0] for virtual channel 0
0x9B	MIPI_SRCA_HEIGHTH0	0x01	RW	Bit[7:0]: srca_vsize[15:8] for virtual channel 0
0x9C	MIPI_SRCA_HEIGHTL0	0xE0	RW	Bit[7:0]: srca_vsize[7:0] for virtual channel 0
0x9D	MIPI_SRC0_TAG0	0x1E	RW	Bit[7:6]: Channel_ID tag transmitted in virtual channel 0 Bit[5:0]: Source 0 data_ID tag transmitted in virtual channel 0
0x9E	MIPI_SRC0_TAG1	0x9E	RW	Bit[7:6]: Channel_ID tag transmitted in virtual channel 1 Bit[5:0]: Source 0 data_ID tag transmitted in virtual channel 1
0x9F	MIPI_SRC0_TAG2	0xAA	RW	Bit[7:6]: Channel_ID tag transmitted in virtual channel 2 Bit[5:0]: Source 0 data_ID tag transmitted in virtual channel 2
0xA0	MIPI_SRC0_TAG3	0xEA	RW	Bit[7:6]: Channel_ID tag transmitted in virtual channel 3 Bit[5:0]: Source 0 data_ID tag transmitted in virtual channel 3
0xA1	MIPI_SRC1_TAG0	0x12	RW	Bit[7:6]: Reserved Bit[5:0]: Source 1 data_ID tag transmitted in virtual channel 0
0xA2	MIPI_SRC1_TAG1	0x12	RW	Bit[7:6]: Reserved Bit[5:0]: Source 1 data_ID tag transmitted in virtual channel 1
0xA3	MIPI_SRC1_TAG2	0x12	RW	Bit[7:6]: Reserved Bit[5:0]: Source 1 data_ID tag transmitted in virtual channel 2
0xA4	MIPI_SRC1_TAG3	0x12	RW	Bit[7:6]: Reserved Bit[5:0]: Source 1 data_ID tag transmitted in virtual channel 3
0xA5	MIPI_SRC2_TAG0	0x12	RW	Bit[7:6]: Reserved Bit[5:0]: Source 2 data_ID tag transmitted in virtual channel 0
0xA6	MIPI_SRC2_TAG1	0x12	RW	Bit[7:6]: Reserved Bit[5:0]: Source 2 data_ID tag transmitted in virtual channel 1

table 6-34 MIPI_TX registers (sheet 16 of 25)

address	register name	default value	R/W	description
0xA7	MIPI_SRC2_TAG2	0x12	RW	Bit[7:6]: Reserved Bit[5:0]: Source 2 data_ID tag transmitted in virtual channel 2
0xA8	MIPI_SRC2_TAG3	0x12	RW	Bit[7:6]: Reserved Bit[5:0]: Source 2 data_ID tag transmitted in virtual channel 3
0xA9	MIPI_SRC3_TAG0	0x31	RW	Bit[7:6]: Reserved Bit[5:0]: Source 3 data_ID tag transmitted in virtual channel 0
0xAA	MIPI_SRC4_TAG0	0x32	RW	Bit[7:6]: Reserved Bit[5:0]: Source 4 data_ID tag transmitted in virtual channel 0
0xAB	MIPI_SRC5_TAG0	0x33	RW	Bit[7:6]: Reserved Bit[5:0]: Source 5 data_ID tag transmitted in virtual channel 0
0xAC	MIPI_SRC6_TAG0	0x34	RW	Bit[7:6]: Reserved Bit[5:0]: Source 6 data_ID tag transmitted in virtual channel 0
0xAD	MIPI_SRC7_TAG0	0x35	RW	Bit[7:6]: Reserved Bit[5:0]: Source 7 data_ID tag transmitted in virtual channel 0
0xAE	MIPI_SRC8_TAG0	0x36	RW	Bit[7:6]: Reserved Bit[5:0]: Source 8 data_ID tag transmitted in virtual channel 0
0xAF	MIPI_SRC9_TAG0	0x37	RW	Bit[7:6]: Reserved Bit[5:0]: Source 9 data_ID tag transmitted in virtual channel 0
0xB0	MIPI_SRCA_TAG0	0x38	RW	Bit[7:6]: Reserved Bit[5:0]: Source A data_ID tag transmitted in virtual channel 0

table 6-34 MIPI_TX registers (sheet 17 of 25)

address	register name	default value	R/W	description
0xB1	MIPI_TX_EXT_CTRL	0xC0	RW	Bit[7:6]: SBS channel select 00: Select channel 0 for SBS image output 01: Select channel 1 for SBS image output 10: Select channel 2 for SBS image output 11: Select channel 3 for SBS image output Bit[5]: SBS width configuration mode 0: SBS width (byte) by SBS module 1: SBS width (pixel) by MIPI_TX module Bit[4]: Reserved Bit[3:0]: mipi_tx_debug mux 0000: mipi_tx default debug bus output 0001: mipi_tx 1st debug bus output 0010: mipi_tx 22nd debug bus output 0011: mipi_tx 3rd debug bus output 0100: mipi_tx 4th debug bus output 0101: mipi_tx 5th debug bus output 0110: mipi_tx 6th debug bus output 0111: mipi_tx 7th debug bus output 1000: mipi_tx 8th debug bus output 1001: mipi_tx 9th debug bus output 1010: mipi_tx 10th debug bus output 1011: mipi_tx 11th debug bus output 1100: mipi_tx 12th debug bus output 1101: mipi_tx 13th debug bus output 1110: mipi_tx 14th debug bus output 1111: Reserved

table 6-34 MIPI_TX registers (sheet 18 of 25)

address	register name	default value	R/W	description
0xB2	MIPI_TX_INTR_EN1	0x00	RW	Bit[7]: mipi_tx_intr enable 15 Bit[6]: mipi_tx_intr enable 14 Bit[5]: mipi_tx_intr enable 13 Bit[4]: mipi_tx_intr enable 12 Bit[3]: mipi_tx_intr enable 11 Bit[2]: mipi_tx_intr enable 10 Bit[1]: mipi_tx_intr enable 9 Bit[0]: mipi_tx_intr enable 8
0xB3	MIPI_TX_INTR_EN0	0x00	RW	Bit[7]: mipi_tx_intr enable 7 Bit[6]: mipi_tx_intr enable 6 Bit[5]: mipi_tx_intr enable 5 Bit[4]: mipi_tx_intr enable 4 Bit[3]: mipi_tx_intr enable 3 Bit[2]: mipi_tx_intr enable 2 Bit[1]: mipi_tx_intr enable 1 Bit[0]: mipi_tx_intr enable 0
0xB4	MIPI_TX_INTR_CLR1	0x00	RW	Bit[7]: mipi_tx_intr clear 15 Bit[6]: mipi_tx_intr clear 14 Bit[5]: mipi_tx_intr clear 13 Bit[4]: mipi_tx_intr clear 12 Bit[3]: mipi_tx_intr clear 11 Bit[2]: mipi_tx_intr clear 10 Bit[1]: mipi_tx_intr clear 9 Bit[0]: mipi_tx_intr clear 8
0xB5	MIPI_TX_INTR_CLR0	0x00	RW	Bit[7]: mipi_tx_intr clear 7 Bit[6]: mipi_tx_intr clear 6 Bit[5]: mipi_tx_intr clear 5 Bit[4]: mipi_tx_intr clear 4 Bit[3]: mipi_tx_intr clear 3 Bit[2]: mipi_tx_intr clear 2 Bit[1]: mipi_tx_intr clear 1 Bit[0]: mipi_tx_intr clear 0
0xB6	MIPI_TX_INTR_STATUS1	–	R	Bit[7]: mipi_tx_intr 15 status Bit[6]: mipi_tx_intr 14 status Bit[5]: mipi_tx_intr 13 status Bit[4]: mipi_tx_intr 12 status Bit[3]: mipi_tx_intr 11 status Bit[2]: mipi_tx_intr 10 status Bit[1]: mipi_tx_intr 9 status Bit[0]: mipi_tx_intr 8 status

table 6-34 MIPI_TX registers (sheet 19 of 25)

address	register name	default value	R/W	description
0xB7	MIPI_TX_INTR_STATUS	–	R	Bit[7]: mipi_tx_intr 7 status Bit[6]: mipi_tx_intr 6 status Bit[5]: mipi_tx_intr 5 status Bit[4]: mipi_tx_intr 4 status Bit[3]: mipi_tx_intr 3 status Bit[2]: mipi_tx_intr 2 status Bit[1]: mipi_tx_intr 1 status Bit[0]: mipi_tx_intr 0 status
0xB8	MIPI_LINE_CNT_VC0_H	–	R	Bit[7:0]: mipi_line_cnt_vc0[15:8]
0xB9	MIPI_LINE_CNT_VC0_L	–	R	Bit[7:0]: mipi_line_cnt_vc0[7:0]
0xBA	MIPI_LINE_CNT_VC1_H	–	R	Bit[7:0]: mipi_line_cnt_vc1[15:8]
0xBB	MIPI_LINE_CNT_VC1_L	–	R	Bit[7:0]: mipi_line_cnt_vc1[7:0]
0xBC	MIPI_LINE_CNT_VC2_H	–	R	Bit[7:0]: mipi_line_cnt_vc2[15:8]
0xBD	MIPI_LINE_CNT_VC2_L	–	R	Bit[7:0]: mipi_line_cnt_vc2[7:0]
0xBE	MIPI_LINE_CNT_VC3_H	–	R	Bit[7:0]: mipi_line_cnt_vc3[15:8]
0xBF	MIPI_LINE_CNT_VC3_L	–	R	Bit[7:0]: mipi_line_cnt_vc3[7:0]
0xC0	MIPI_BYTE_CNT_VC0_H	–	R	Bit[7:0]: mipi_line_cnt_vc0[15:8]
0xC1	MIPI_BYTE_CNT_VC0_L	–	R	Bit[7:0]: mipi_line_cnt_vc0[7:0]
0xC2	MIPI_BYTE_CNT_VC1_H	–	R	Bit[7:0]: mipi_line_cnt_vc1[15:8]
0xC3	MIPI_BYTE_CNT_VC1_L	–	R	Bit[7:0]: mipi_line_cnt_vc1[7:0]
0xC4	MIPI_BYTE_CNT_VC2_H	–	R	Bit[7:0]: mipi_line_cnt_vc2[15:8]
0xC5	MIPI_BYTE_CNT_VC2_L	–	R	Bit[7:0]: mipi_line_cnt_vc2[7:0]
0xC6	MIPI_BYTE_CNT_VC3_H	–	R	Bit[7:0]: mipi_line_cnt_vc3[15:8]
0xC7	MIPI_BYTE_CNT_VC3_L	–	R	Bit[7:0]: mipi_line_cnt_vc3[7:0]

table 6-34 MIPI_TX registers (sheet 20 of 25)

address	register name	default value	R/W	description
0xC8	MIPI_HS_EN_CTRL	0x00	RW	Bit[7]: hs_en_adjust_en 0: Disable hs_en adjust for VCM noise concern 1: Enable hs_en adjust for VCM noise concern Bit[6]: hs_en_adjust_mode 0: Only delay hs_en pull up sequence 1: Delay hs_en pull up sequence and pull down sequence Bit[5]: dl_valid_adjust_en 0: Disable dl_valid adjust for VCM noise concern 1: Enable dl_valid adjust for VCM noise concern Bit[4]: dl_valid_adjust_mode 0: Only delay dl_valid pull up sequence 1: Delay dl_valid pull up sequence and pull down sequence Bit[3:0]: Reserved
0xC9	MIPI_VALID_CTRL	0x22	RW	Bit[7:4]: valid_ahead Ahead cycle for clock/data lane valid signal (related to hs_en signal) Bit[3:0]: valid_delay Delay cycle for clock/data lane valid signal (related to hs_en signal)
0xD0	DS_CTRL0	0x00	RW	Bit[7]: De-skew calibration enable (total control) 0: Disable de-skew calibion function 1: Enable de-skew calibion function Bit[6]: De-skew err flag report enable 0: Disable de-skew error report when an error occurs 1: Enable de-skew error report when an error occurs

table 6-34 MIPI_TX registers (sheet 21 of 25)

address	register name	default value	R/W	description
				Bit[5]: De-skew err flag interrupt enable 0: Disable ds_err_flag interrupt 1: Enable ds_err_flag interrupt Bit[4]: De-skew calibration insertation enable when power up 0: Disable de-skew calibration insertation when power up 1: Enable de-skew calibration insertation when power up Bit[3]: De-skew calibration power up insertation mode select 0: Hardware mode, after reset, before first SOF, insert de-skew calibration 1: Software mode, when power up TX PHY, user needs to configure this bit to insert de-skew calibration Bit[2]: Software mode enable for de-skew calibration insertation when power up 0: Disable de-skew calibration power up insertaion software mode 1: Enable de-skew calibration power up insertaion software mode Bit[1]: De-skew calibration insertation enable when frame blanking 0: Disable de-skew calibration insertation when frame blanking 1: Enable de-skew calibration insertation when frame blanking Bit[0]: Mode of de-skew calibration insertation when frame blanking 0: Each frame blanking all do de-skew calibration 1: Select some special frame blanking to do de-skew

table 6-34 MIPI_TX registers (sheet 22 of 25)

address	register name	default value	R/W	description
0xD1	DS_CTRL1	0x00	RW	Bit[7]: De-skew calibration manual mode enable When CCI control CSI-2 to do de-skew calibration, aka manual mode 0: Disable de-skew calibration manual mode 1: Enable de-skew calibration manual mode Bit[6]: De-skew calibration manual stop mode 0: All de-skew calibration start and stop all controlled manually by CCI 1: Only de-skew calibtration start controlled manually by CCI, stop need count Tde-skew-cal to finish Bit[5]: De-skew manual stop enable (only for stop mode 0) 0: Do not stop manual mode by CCI 1: Stop manual mode by CCI Bit[4]: De-skew calibration manual start mode 0: After manual control from CCI enable,then start de-skew calibration 1: Only after manual control from CCI enable and active frame blanking, then start de-skew calibration Bit[3]: De-skew manual start enable (only for start mode 0) 0: Do not start manual mode by CCI 1: Start manual mode by CCI Bit[2:0]: Frame blanking interval of manual mode for doing de-skew calibration when frame blanking 000: Do de-skew calibration in special frame blanking 001: Do de-skew calibration 2 frame blanking interval

table 6-34 MIPI_TX registers (sheet 23 of 25)

address	register name	default value	R/W	description
				010: Do de-skew calibration 4 frame blanking interval 011: Do de-skew calibration 8 frame blanking interval 100: Do de-skew calibration 16 frame blanking interval 101: Do de-skew calibration 32 frame blanking interval 110: Do de-skew calibration 64 frame blanking interval 111: Do de-skew calibration 128 frame blanking interval
0xD2	DS_CTRL2	0x00	RW	Bit[7]: De-skew err flag interrupt clear 0: Do not clear ds_err_flag interrupt 1: Clear ds_err_flag interrupt Bit[6]: De-skew calibration one time one line (OTOL) after power up enable 0: Disable OTOL feature after power up 1: Enable OTOL feature after power up Bit[5]: De-skew calibration one frame one line (OFOL) enable 0: Disable OFOL feature 1: Enable OFOL feature Bit[4]: De-skew calibration OTOL enable Sequence is D0+clk-->D1+clk-->D2+clk-->D3+clk 0: Disable OTOL feature 1: Enable OTOL feature Bit[3:0]: Reserved
0xD3	DS_SYNC_BYTE1	0xFF	RW	Bit[7:0]: De-skew 1st sync byte for DPHY1.2 calibration
0xD4	DS_SYNC_BYTE2	0xFF	RW	Bit[7:0]: De-skew 2nd sync byte for DPHY1.2 calibration

table 6-34 MIPI_TX registers (sheet 24 of 25)

address	register name	default value	R/W	description
0xD5	DS_LANE12_BYTE	0x55	RW	Bit[7:4]: Lane2 data format Bit[3:0]: Lane1 data format
0xD6	DS_LANE34_BYTE	0x55	RW	Bit[7:4]: Lane4 data format Bit[3:0]: Lane3 data format
0xD7	DS_PU_LENGTHH	0x17	RW	Bit[7:0]: De-skew calibration sequence length when power up[15:8]
0xD8	DS_PU_LENGTHL	0x70	RW	Bit[7:0]: De-skew calibration sequence length when power up[7:0]
0xD9	DS_FB_LENGTHH	0x03	RW	Bit[7:0]: De-skew calibration sequence length when vertical frame blanking[15:8]
0xDA	DS_FB_LENGTHL	0x20	RW	Bit[7:0]: De-skew calibration sequence length when vertical frame blanking[7:0]
0xDB	DS_START_DELAYH	0x00	RW	Bit[7:0]: Delay time between frame_end and first de-skew sequence[15:8]
0xDC	DS_START_DELAYL	0x00	RW	Bit[7:0]: Delay time between frame_end and first de-skew sequence[7:0]
0xDD	DS_INTER_DELAYH	0x00	RW	Bit[7:0]: Delay time between each lane de-skew sequence[15:8]
0xDE	DS_INTER_DELAYL	0x00	RW	Bit[7:0]: Delay time between each lane de-skew sequence[7:0]
0xDF	DS_FB_CNTH	0x00	RW	Bit[7:0]: Frame blanking count[15:8]
0xE0	DS_FB_CNTL	0x00	RW	Bit[7:0]: Frame blanking count[7:0]
0xE1	DS_STATUS	–	R	Bit[7:4]: De-skew calibration lane number when happen error case Bit[3]: Reserved Bit[2]: De-skew calibration have error status when send pattern Bit[1]: De-skew calibration is in process Bit[0]: De-skew calibration in busy process
0xE2	DS_ERR_FRAME_H	–	R	Bit[7:0]: De-skew calibration frame count high byte when error case happens

table 6-34 MIPI_TX registers (sheet 25 of 25)

address	register name	default value	R/W	description
0xE3	DS_ERR_FRAME_L	–	R	Bit[7:0]: De-skew calibration frame count low byte when error case happens
0xE4	DS_ERR_PAT_LEN_H	–	R	Bit[7:0]: De-skew calibration has done cycle high byte in error lane when error case happens
0xE5	DS_ERR_PAT_LEN_L	–	R	Bit[7:0]: De-skew calibration has done cycle low byte in error lane when error case happens

7 operating specifications

7.1 absolute maximum ratings

table 7-1 absolute maximum ratings

parameter		absolute maximum rating[a]
ambient storage temperature		-40°C to +125°C
supply voltage (with respect to ground)	MTX0_AVDD, PADVDD18, MRX0AVDD, MRX1_AVDD, VRI	1.8V
	MTX0_DVDD, MRX0DVDD, MRX1_DVDD, VROPD, VROPK, CPDVDD12, CPKVDD12	1.2V
electro-static discharge (ESD)	human body model	2000V
	machine model	200V
all input/output voltages (with respect to ground)		0.3V to V_{DD-IO} +1V
I/O current on any input or output pin		± 200 mA
peak solder temperature (10 second dwell time)		245°C

a. exceeding the absolute maximum ratings shown above invalidates all AC and DC electrical specifications and may result in permanent damage to the device. Exposure to absolute maximum rated conditions for extended periods may affect device reliability.

7.2 functional temperature

table 7-2 functional temperature

parameter	range
operating temperature	-25°C to +70°C junction temperature

7.3 DC characteristics

table 7-3 DC characteristics (sheet 1 of 2)

symbol	parameter	condition	min	typ	max	unit
supply						
VRI	supply voltage (regulator analog)	1.8V ±10%	1.62	1.8	1.98	V
MTX0_AVDD	supply voltage (mipi_tx0 analog)	1.8V ±10%	1.62	1.8	1.98	V
MRX0AVDD	supply voltage (mipi_rx0 analog)	1.8V ±10%	1.62	1.8	1.98	V
MRX1_AVDD	supply voltage (mipi_rx1 analog)	1.8V ±10%	1.62	1.8	1.98	V
PADVDD18	supply voltage (digital I/O)	1.8V ±10%	1.62	1.8	1.98	V
MTX0_DVDD	supply voltage (mipi_tx0 digital)	1.2V ±10%	1.08	1.2	1.32	V
MRX0DVDD	supply voltage (mipi_rx0 digital)	1.2V ±10%	1.08	1.2	1.32	V
MRX1_DVDD	supply voltage (mipi_rx1 digital)	1.2V ±10%	1.08	1.2	1.32	V
CPDVDD12	supply voltage (digital core power down domain power)	1.2V ±10%	1.08	1.2	1.32	V
CPKVDD12	supply voltage (digital core power keep domain power)	1.2V ±10%	1.08	1.2	1.32	V
VROPD	supply voltage (power down domain regulator power supply)	1.2V ±10%	1.08	1.2	1.32	V
VROPK	supply voltage (power keep domain regulator power supply)	1.2V ±10%	1.08	1.2	1.32	V
digital inputs (typical conditions: AVDD = 1.8V, PADVDD18 =1.8V)						
V_{IH}	input voltage HIGH	CMOS	0.7 × PADVDD18			V
V_{IL}	input voltage LOW	CMOS			0.3 × PADVDD18	V

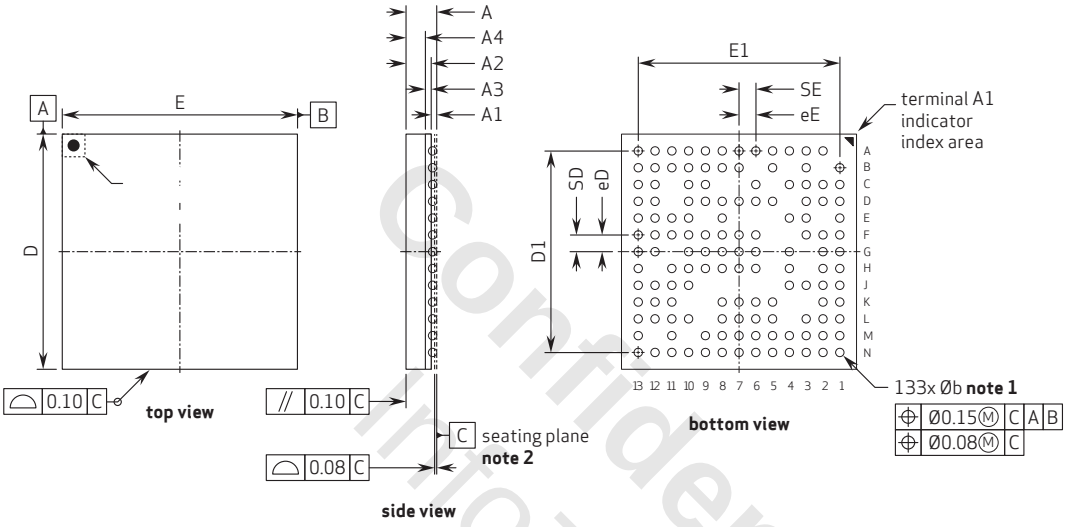
table 7-3 DC characteristics (sheet 2 of 2)

symbol	parameter	condition	min	typ	max	unit
digital outputs (standard loading 25 pF)						
V_{OH}	output voltage HIGH	CMOS	0.9 × PADVDD18			V
V_{OL}	output voltage LOW	CMOS			0.1 × PADVDD18	V

8 mechanical specifications

8.1 physical specifications

figure 8-1 package specifications



note 1 dimension b is measured at the maximum solder ball diameter, parallel to datum plane c

note 2 datum c (seating plane) is defined by the spherical crowns of the solder ball

683_BDS_6_1

table 8-1 package dimensions (sheet 1 of 2)

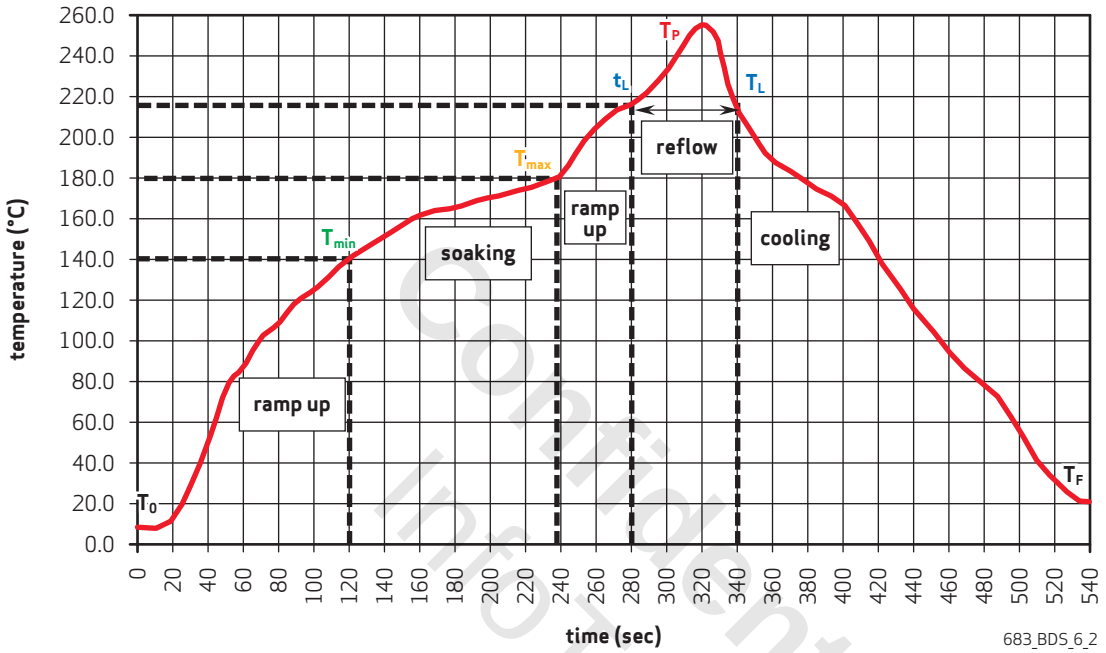
parameter	symbol	min	typ	max	unit
total thickness	A			1.17	mm
stand off	A1	0.16		0.26	mm
substrate thickness	A2			0.84	mm
mold thickness	A3	0.22	0.26	0.30	mm
	A4	0.51	0.53	0.54	mm
body size y-axis	D	6.95	7.00	7.05	mm
body size x-axis	E	6.95	7.00	7.05	mm
edge ball center to center (y-axis)	D1		6.00 BSC		mm
edge ball center to center (x-axis)	E1		6.00 BSC		mm
ball width	b	0.20	0.25	0.30	mm

table 8-1 package dimensions (sheet 2 of 2)

parameter	symbol	min	typ	max	unit
ball pitch	eE		0.50 BSC		mm
	eD		0.50 BSC		mm
body center to contact ball (y-axis)	SD		0.50 BSC		mm
body center to contact ball (x-axis)	SE		0.50 BSC		mm
ball opening			0.25 BSC		mm

8.2 IR reflow specifications

figure 8-2 IR reflow ramp rate requirements



note

The OV683 uses a lead free package.

table 8-2 reflow conditions[ab]

zone	description	exposure
ramp up A (T_0 to T_{min})	heating from room temperature to 140°C	temperature slope ≤ 3°C per second
soaking	heating from 140°C to 180°C	90 ~ 150 seconds
ramp up B (t_L to T_P)	heating from 217°C to 260°C	temperature slope ≤ 3°C per second
peak temperature	maximum temperature in SMT	260°C +0/-5°C (duration max 30 sec)
reflow (t_L to T_L)	temperature higher than 217°C	30 ~ 120 seconds
ramp down A (T_P to T_L)	cooling from 260°C to 217°C	temperature slope ≤ 3°C per second
ramp down B (T_L to T_f)	cooling from 217°C to room temperature	temperature slope ≤ 6°C per second
T_0 to T_P	room temperature to peak temperature	≤ 8 minutes

a. maximum number of reflow cycles = 3

b. N2 gas reflow or control O2 gas PPM<500 as recommendation

revision history

version 1.0 02.27.2015

- initial release

version 1.01 06.04.2015

- in chapter 2, added section 2.7

version 1.1 06.17.2016

- added new chapter 6 and moved subsequent chapters accordingly
- in chapter 7, added section 7.2

version 2.0 06.29.2016

- changed datasheet from Preliminary Specification to Product Specification
- added new chapter 6 and moved subsequent chapters accordingly

defining the future of digital imaging™

OmniVision Technologies, Inc.

UNITED STATES

4275 Burton Drive
Santa Clara, CA 95054

tel: +1 408 567 0000

fax: +1 408 567 0001

email: sale@ovt.com

UNITED KINGDOM

Hook, Hampshire +44 1256 744 666

GERMANY

Munich +49 89 63 81 99 99

INDIA

Bangalore +91 80 4112 8888

CHINA

Beijing + 86 10 6580 1666

Shanghai + 86 21 6175 8888

+86 21 5774 9888

Shenzhen + 86 755 8304 5321

JAPAN

Yokohama +81 45 478 7777

Osaka +81 6 4964 2666

KOREA

Seoul + 82 2 3472 3333

SINGAPORE +65 6933 1333

TAIWAN

Taipei +886 2 2657 9888

Hsin-chu +886 3 5658888

Sales: Shenzhen Sunnywale Inc, www.sunnywale.com , awin@sunnywale.com , Wechat: 9308762