datasheet

PRODUCT SPECIFICATION

1/2" color CMOS 8 megapixel (3840 x 2160) image sensor
with PureCel® technology

OS08A10

color CMOS 8 megapixel (3840 x 2160) image sensor with PureCel® technology

datasheet (CSP)
PRODUCT SPECIFICATION

version 2.11
april 2018

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

applications

- security camera
- action camera
- high resolution consumer camera
- digital still cameras (DSC)
- digital video camcorders (DVC)

ordering information

- **OS08A10-H92A** (color, lead-free)
  92-pin CSP

features

- 2 µm x 2 µm pixel
- optical size of 1/2"
- programmable controls for frame rate, mirror and flip, cropping, and windowing
- supports output formats: 12-/10-bit RAW RGB
- supports images sizes: 4K2K (3840x2160), 2560x1440, 1080p (1920x1080), 720p (1280x720)
- supports 2x2 binning
- standard serial SCCB interface
- 12-bit ADC
- up to 4-lane MIPI/LVDS serial output interface (supports maximum speed up to 1500 Mbps/lane)
- 2-exposure staggered HDR support
- programmable I/O drive capability
- light sensing mode (LSM)
- PLL with SSC support
- support for FSIN

key specifications (typical)

- **active array size:** 3840 x 2160
- **power supply:**
  core: 1.2V
  analog: 2.8V
  I/O: 1.8V
- **power requirements:**
  active: 240 mA
  XSHUTDOWN: <10 µA
- **temperature range:**
  operating: -30°C to +85°C junction temperature (see **table 7-2**)
  stable image: 0°C to +60°C junction temperature (see **table 7-2**)
- **output formats:** 10/12-bit RAW RGB
- **lens size:** 1/2"
- **input clock frequency:** 6~27 MHz
- **lens chief ray angle:** 11° linear (see **figure 9-3**)
- **max S/N ratio**: 39 dB
- **dynamic range:** 74 dB @ 16x gain
- **maximum image transfer rate:**
  4K2K: 60 fps (see **table 2-1**)
  2250x1440: 60 fps (see **table 2-1**)
- **sensitivity:** 13,000 e⁻/Lux-sec
- **scan mode:** progressive
- **maximum exposure interval:** VTS-8
- **pixel size:** 2.0 µm x 2.0 µm
- **image area:** 7736.256 µm x 4379.616 µm
- **package dimensions:** 8939.2 µm x 6340 µm

note Higher junction temperature degrades image quality

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 OS08A10 image sensor. The package information is shown in section 8.

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

pin number	signal name	pin type	description
A1	NC	–	no connect
A2	DVDD	power	digital circuit power
A3	DOVDD	power	I/O power
A4	DVDD	power	digital circuit power
A5	DOVDD	power	I/O power
A6	DOGND	ground	I/O ground
A7	DOGND	ground	I/O ground
A8	DOVDD	power	I/O power
A9	DOGND	ground	I/O ground
A10	DOVDD	power	I/O power
A11	NC	–	no connect
B1	AGND	ground	analog ground
B2	DOGND	ground	I/O ground
B3	ATEST	output	analog test pin
B4	DOGND	ground	I/O ground
B7	DVDD	power	digital circuit power
B8	DVDD	power	digital circuit power
B9	DVDD	power	digital circuit power
B10	DOGND	ground	I/O ground
B11	AGND	ground	analog ground
C1	AVDD	power	analog power
C2	AVDD	power	analog power
C3	AVDD	power	analog power
C4	DOGND	ground	I/O ground
C7	DOGND	ground	I/O ground

table 1-1 signal descriptions (sheet 2 of 4)

pin number	signal name	pin type	description
C8	DOGND	ground	I/O ground
C9	AGND	ground	analog ground
C10	DVDD	power	digital circuit power
C11	AVDD	power	analog power
D1	AGND	ground	analog ground
D2	VN2	input	reference
D3	VN1	input	reference
D4	PVDD	power	PLL analog power
D7	DVDD	power	digital circuit power
D8	DVDD	power	digital circuit power
D9	SCL	input	SCCB interface input clock
D10	AVDD	power	analog power
D11	DOVDD	power	I/O power
E1	DOGND	ground	I/O ground
E2	VH1	input	reference
E3	DOGND	ground	I/O ground
E4	DOGND	ground	I/O ground
E5	DOGND	ground	I/O ground
E6	DOGND	ground	I/O ground
E7	DOGND	ground	I/O ground
E8	XVCLK	input	system clock input
E9	XSHUTDOWN2	input	reset and power down (active low with pull down resistor)
E10	DOVDD	power	I/O power
E11	DVDD	power	digital circuit power
F1	DVDD	power	digital circuit power
F2	AVDD	power	analog power
F3	DOGND	ground	I/O ground
F4	MDP0	output	MIPI data positive output
F5	EVDD	power	MIPI/PLL digital circuit power
F6	EVDD	power	MIPI/PLL digital circuit power

table 1-1 signal descriptions (sheet 3 of 4)

pin number	signal name	pin type	description
F7	**MDP3**	output	MIPI data positive output
F8	**GPIO0**	I/O	general purpose I/O
F9	**VSYNC**	I/O	video output vertical signal
F10	**TM**	input	test mode (active high with pull down resistor)
F11	DOGND	ground	I/O ground
G1	VH2	input	reference
G2	PGND	ground	analog ground
G3	**MDP2**	output	MIPI data positive output
G4	**MDN0**	output	MIPI data negative output
G5	**MCN**	output	MIPI clock negative output
G6	EGND	ground	ground for MIPI TX circuit
G7	**MDN3**	output	MIPI data negative output
G8	**GPIO3**	I/O	general purpose I/O
G9	HREF	I/O	video output horizontal signal
G10	**SID**	input	SCCB last bit ID input 0: SCCB ID address = 0x6C 1: SCCB ID address = 0x20
G11	DVDD	power	digital circuit power
H1	AGND	ground	analog ground
H2	DOVDD	power	I/O power
H3	**MDN2**	output	MIPI data negative output
H4	EGND	ground	ground for MIPI TX circuit
H5	**MCP**	output	MIPI clock positive output
H6	**MDP1**	output	MIPI data positive output
H7	DOVDD	power	I/O power
H8	**GPIO2**	I/O	general purpose I/O
H9	**PWDNB**	input	power down (active low)
H10	**STROBE**	output	frame exposure output indicator
H11	**SDA**	I/O	SCCB interface data pin
J1	NC	–	no connect
J2	DVDD	power	digital circuit power

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

pin number	signal name	pin type	description
J3	EVDD	power	MIPI/PLL digital circuit power
J4	DVDD	power	digital circuit power
J6	**MDN1**	output	MIPI data negative output
J7	DVDD	power	digital circuit power
J8	**GPIO1**	I/O	general purpose I/O
J9	**XSHUTDOWN**	input	reset and power down (active low with pull down resistor)
J10	DOGND	ground	I/O ground
J11	NC	–	no connect

table 1-2 configuration under various conditions (sheet 1 of 2)

pin	signal name	XSHUTDOWN[a]	after XSHUTDOWN with XSHUTDOWN2 high[b]	after XSHUTDOWN with XSHUTDOWN2 low ALS mode	software standby[c]	hardware standby[d]
D9	SCL	high-z	input	input	input	high-z
E8	XVCLK	high-z	input	input	input	high-z
E9	XSHUTDOWN2	input	input	input	input	input
F10	TM	input	input	input	input	input
F4	MDP0	high-z	high	high-z	high by default (configurable)	high by default (configurable)
F7	MDP3	high-z	high	high-z	high by default (configurable)	high by default (configurable)
F8	GPIO0	high-z	input	low by default (configurable)	input by default (configurable)	input by default (configurable)
F9	VSYNC	high-z	input	high-z	input (configurable)	input (configurable)
G10	SID	input	input	input	input	input
G3	MDP2	high-z	high	high-z	high by default (configurable)	high by default (configurable)
G4	MDN0	high-z	high	high-z	high by default (configurable)	high by default (configurable)

table 1-2 configuration under various conditions (sheet 2 of 2)

pin	signal name	XSHUTDOWN[a]	after XSHUTDOWN with XSHUTDOWN2 high[b]	after XSHUTDOWN with XSHUTDOWN2 low ALS mode	software standby[c]	hardware standby[d]
G5	MCN	high-z	high	high-z	high by default (configurable)	high by default (configurable)
G7	MDN3	high-z	high	high-z	high by default (configurable)	high by default (configurable)
G8	GPIO3	high-z	high	high-z	input by default (configurable)	input by default (configurable)
H3	MDN2	high-z	high	high-z	high by default (configurable)	high by default (configurable)
H5	MCP	high-z	high	high-z	high by default (configurable)	high by default (configurable)
H6	MDP1	high-z	high	high-z	high by default (configurable)	high by default (configurable)
H8	GPIO2	high-z	high	input	input by default (configurable)	input by default (configurable)
H9	PWDNB	input	input	input	input	input
H10	STROBE	low	low	low	low by default (configurable)	low by default (configurable)
H11	SDA	open drain	I/O	open drain	I/O	open drain
J6	MDN1	high-z	high	high-z	high by default (configurable)	high by default (configurable)
J8	GPIO1	high-z	high	input	input by default (configurable)	input by default (configurable)
J9	XSHUTDOWN	input	input	input	input	input

a. XSHUTDOWN = 0

b. XSHUTDOWN from 0 to 1

c. sensor set to standby from streaming mode

d. sensor set to hardware standby from streaming mode by pulling PWDNB = 0

figure 1-1 pin diagram

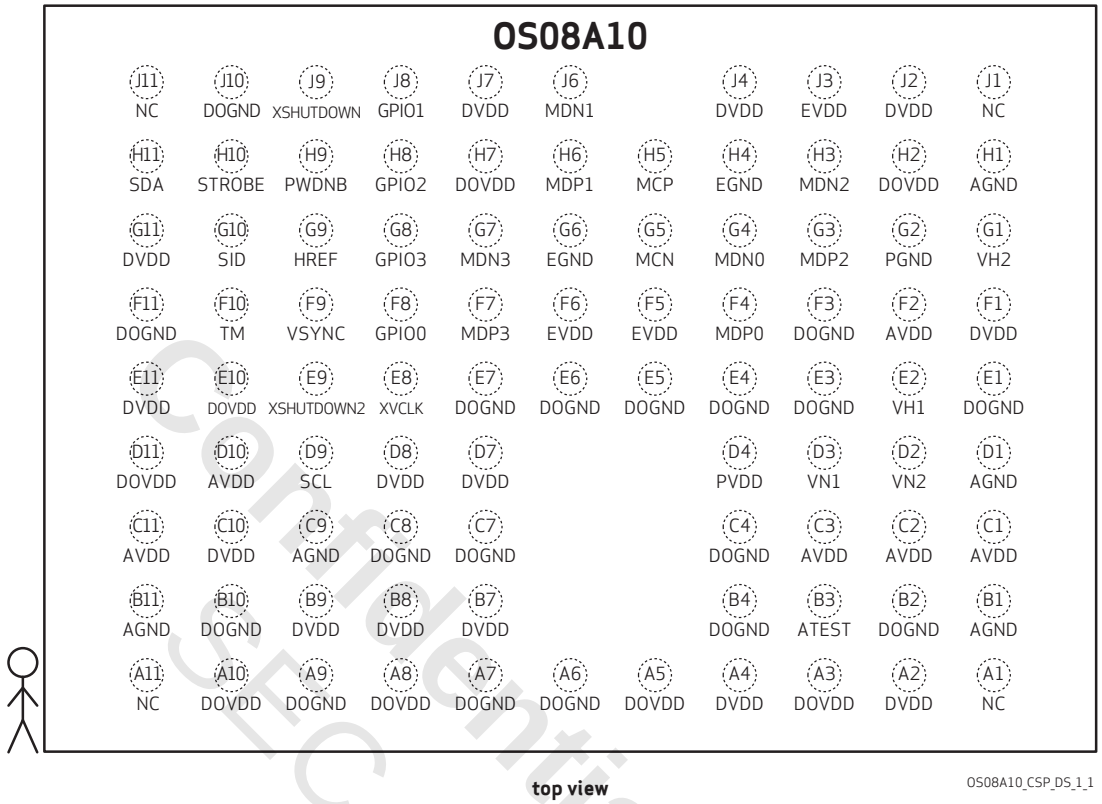


table 1-3 pad symbol and equivalent circuit (sheet 1 of 2)

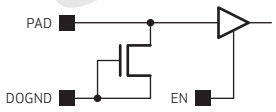
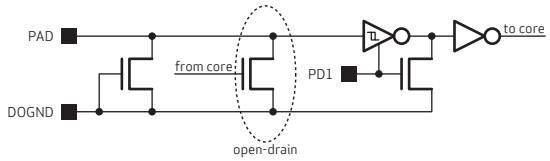
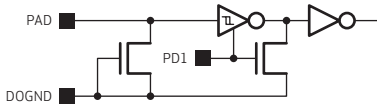
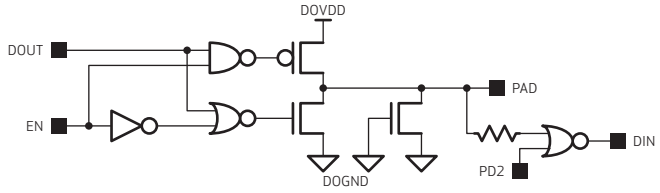
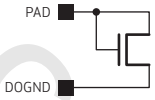
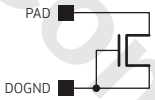
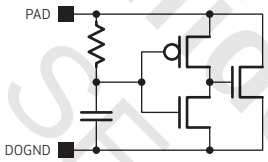
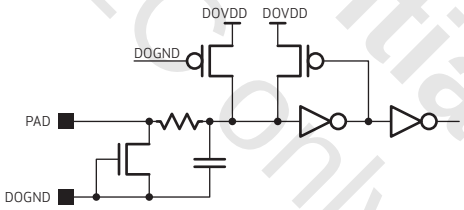
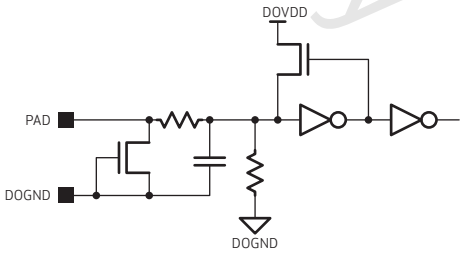
symbol	equivalent circuit
XVCLK	
SDA	
SCL	

table 1-3 pad symbol and equivalent circuit (sheet 2 of 2)

symbol	equivalent circuit
STROBE, VSYNC, GPIO0, GPIO1, GPIO2, GPIO3	
VN1, VN2	
MDP3, MDN3, MDP2, MDN2, MDP1, MDN1, MDP0, MDN0, MCP, MCN, EGND, AGND, DOGND, VH1, VH2	
AVDD, DVDD, DOVDD, PVDD	
PWDNB	
XSHUTDOWN, SID, TM, XSHUTDOWN2	

2 system level description

2.1 overview

The OS08A10 color RAW RGB PureCel® image sensors are high performance, 1/2-inch, 8 megapixel, CMOS, image sensors that deliver 3840x2160 at 60 fps. They provide full-frame, sub-sampled, and windowed 10/12-bit MIPI/LVDS images in various formats via the control of the serial camera control bus (SCCB) interface.

The OS08A10 has an 8 megapixel image array capable of operating at up to 60 frames per second (fps) in 10-bit resolution with complete user control over image quality, formatting and output data transfer. Some image processing functions, such as defective pixel canceling, etc., are programmable through the SCCB interface.

In addition, OmniVision image sensors use proprietary sensor technology to improve image quality by reducing or eliminating common lighting/electrical sources of image contamination, such as fixed pattern noise, smearing, etc., to produce a clean and fully stable color image.

For customized information purposes, the OS08A10 includes 8k bits of one-time programmable (OTP) memory (6k bits are reserved for OmniVision and 2k bits are reserved for customers). The OS08A10 has a MIPI interface of up to four lanes.

2.2 architecture

The OS08A10 sensor core generates streaming pixel data at a constant frame rate. **figure 2-1** shows the functional block diagram of the OS08A10 image sensor.

The timing generator outputs clocks to access the rows of the imaging array, pre-charging and sampling the rows of the array sequentially. In the time between pre-charging and sampling a row, the charge in the pixels decreases with exposure to incident light. This is the exposure time in rolling shutter architecture.

The exposure time is controlled by adjusting the time interval between pre-charging and sampling. After the data of the pixels in the row has been sampled, it is processed through analog circuitry to correct the offset and multiply the data with corresponding gain. Following analog processing is the ADC which outputs up to 12-bit data for each pixel in the array.

figure 2-1 OS08A10 block diagram

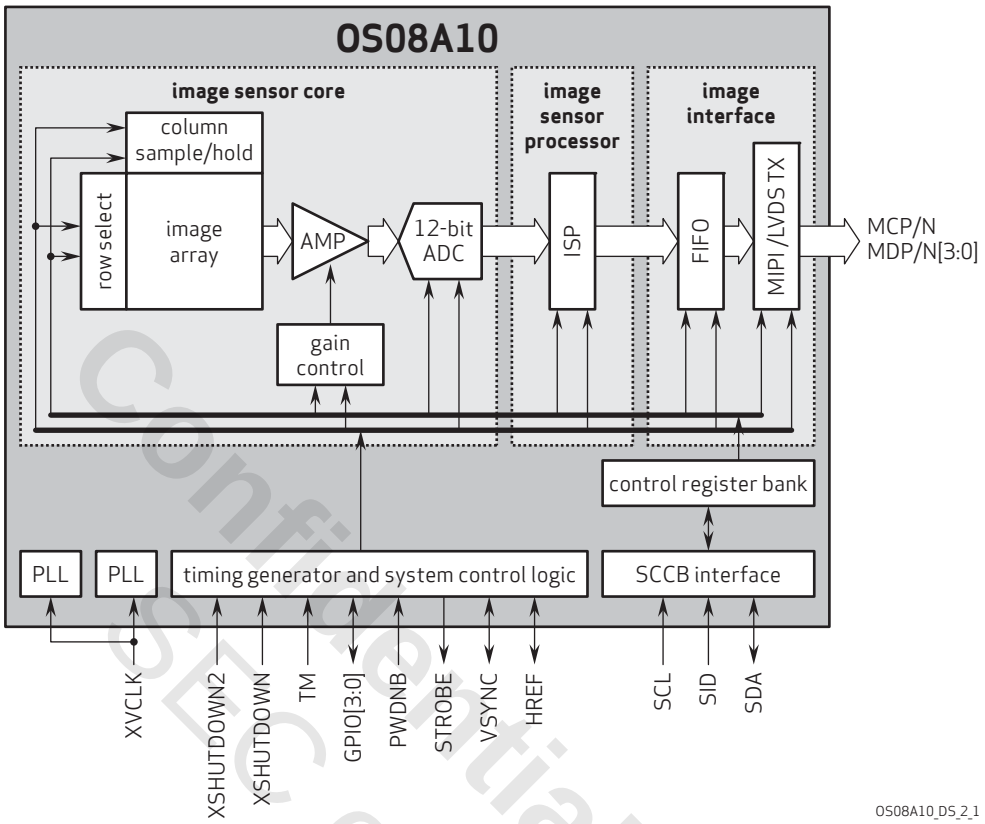
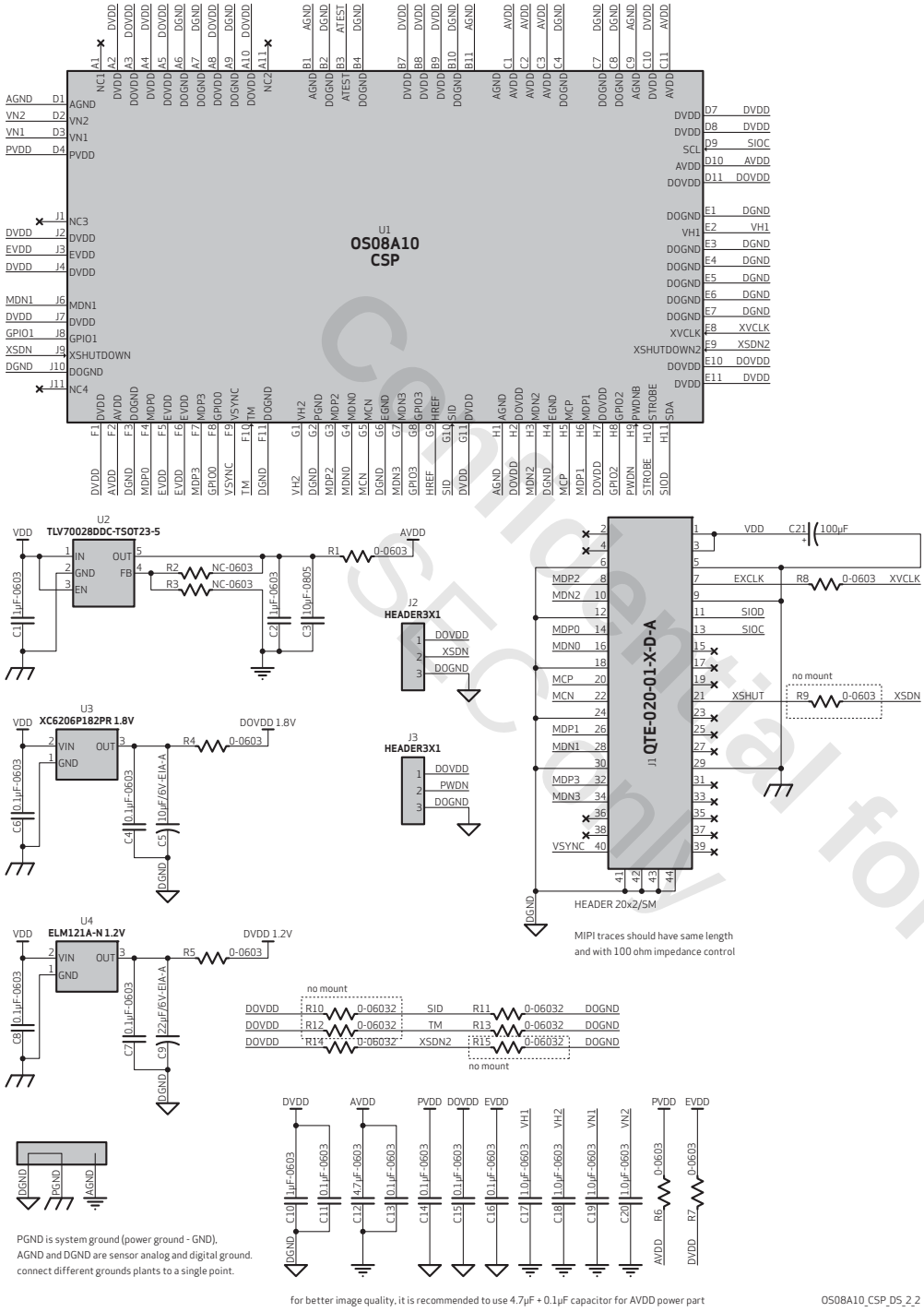


figure 2-2 OS08A10 reference schematic



2.3 format and frame

The OS08A10 supports RAW RGB output with one/two/four MIPI or LVDS interface.

table 2-1 format and frame rate

format	resolution	max frame rate	methodology	10-bit output MIPI data rate
4K2K	3840 x 2160	60 fps	full resolution (16:9)	1440 Mbps/lane
4K2K (12bit)	3840 x 2160	30 fps	full resolution (16:9)	1440 Mbps/lane
2560 x 1440	2360 x 1440	60 fps	cropping	700 Mbps/lane
1920 x 1440	1920 x 1440	60 fps	cropping	500 Mbps/lane
1080p	1920 x 1080	120 fps	2x binning	800 Mbps/lane
1280 x 960	1280 x 960	120 fps	cropping + 2x binning	700 Mbps/lane
720p	1280 x 720	180 fps	cropping + 2x binning	600 Mbps/lane

2.4 I/O control

The OS08A10 can configure its I/O pins as an input or output. For the output signal, it follows one of two paths: either from the data path or from register control.

table 2-2 I/O control registers (sheet 1 of 2)

function	register	description
VSYNC I/O control	0x3002	Bit[7]: VSYNC output enable 0: input 1: output
VSYNC output select	0x3008	Bit[7]: Enable VSYNC as GPIO controlled by register
VSYNC output value	0x3005	Bit[7]: Register control VSYNC output
GPIO0 I/O control	0x3002	Bit[0]: GPIO0 output enable 0: input 1: output
GPIO0 output select	0x3008	Bit[0]: Enable GPIO0 as GPIO controlled by register
GPIO0 output value	0x3005	Bit[0]: Register control GPIO0 output
GPIO1 I/O control	0x3002	Bit[0]: GPIO1 output enable 0: input 1: output
GPIO1 output select	0x3008	Bit[0]: Enable GPIO1 as GPIO controlled by register

table 2-2 I/O control registers (sheet 2 of 2)

function	register	description
GPIO1 output value	0x3005	Bit[1]: Register control GPIO1 output
GPIO2 I/O control	0x3001	Bit[0]: GPIO2 output enable 0: input 1: output
GPIO2 output select	0x3007	Bit[0]: Enable GPIO2 as GPIO controlled by register
GPIO2 output value	0x3004	Bit[0]: Register control GPIO2 output
GPIO3 I/O control	0x3001	Bit[1]: GPIO3 output enable 0: input 1: output
GPIO3 output select	0x3007	Bit[1]: Enable GPIO3 as GPIO controlled by register
GPIO3 output value	0x3004	Bit[1]: Register control GPIO3 output

2.5 MIPI interface

The OS08A10 supports an MIPI interface of up to four lanes. The MIPI interface can be configured for 1/2/4-lanes. Each lane is capable of a data transfer rate of up to 1500 Mbps.

2.6 LVDS interface

The OS08A10 supports a 1/2/4 LVDS interface with a data transfer rate of up to 1500 Mbps. For details of LVDS, please contact local OmniVision FAE.

2.7 power management

OmniVision recommends cutting off all power supplies, including the external DVDD, when the sensor is not in use. There is no requirement for the power sequence. Cutting off any power source (AVDD/DVDD/DOVDD) is equivalent to XSHUTDOWN going low and the OS08A10 will enter hardware standby mode, which uses very low power.

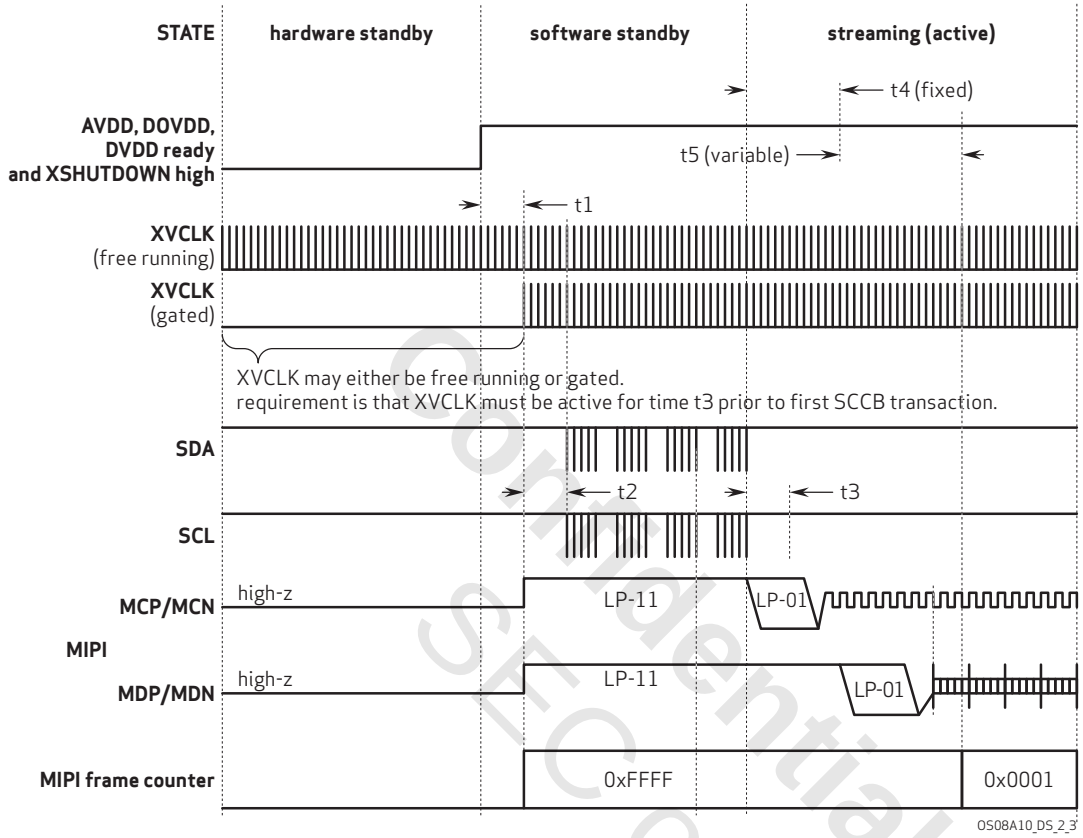
2.7.1 power up sequence

table 2-3 power up sequence timing constraints [a]

constraint	label	min	max	unit
XSHUTDOWN rising – system ready	t1	5		ms
minimum number of XVCLK cycles prior to first SCCB transaction	t2	8192		XVCLK cycles
PLL start up/lock time	t3		0.2	ms
entering streaming mode – first frame start sequence (fixed part)	t4		10	ms
entering streaming mode – first frame start sequence (variable part)	t5	delay is exposure time value		lines

a. if not used, tie both PWDNB and XSHUTDOWN2 pin to DOVDD

figure 2-3 power up sequence



2.7.2 power down sequence

To avoid bad frames from the MIPI, OmniVision recommends using group hold to send SCCB sleep command before sending the sensor into power down mode. To set the sensor into hardware power down mode, pull the XSHUTDOWN signal to low. Any power cut is equivalent to XSHUTDOWN being driven low.

table 2-4 power down sequence timing constraints

constraint	label	min	max	unit
enter software standby SCCB command device in software standby mode	t0	when a frame of MIPI data is output, wait for MIPI end code before entering software for standby; otherwise, enter software standby mode immediately		
minimum of XVCLK cycles after last SCCB transaction or MIPI frame end	t1	512		XVCLK cycles
last SCCB transaction or MIPI frame end, XSHUTDOWN falling	t2	512		XVCLK cycles
XSHUTDOWN falling – AVDD falling or DOVDD falling whichever is first	t3	0.0		ns

figure 2-4 software standby sequence

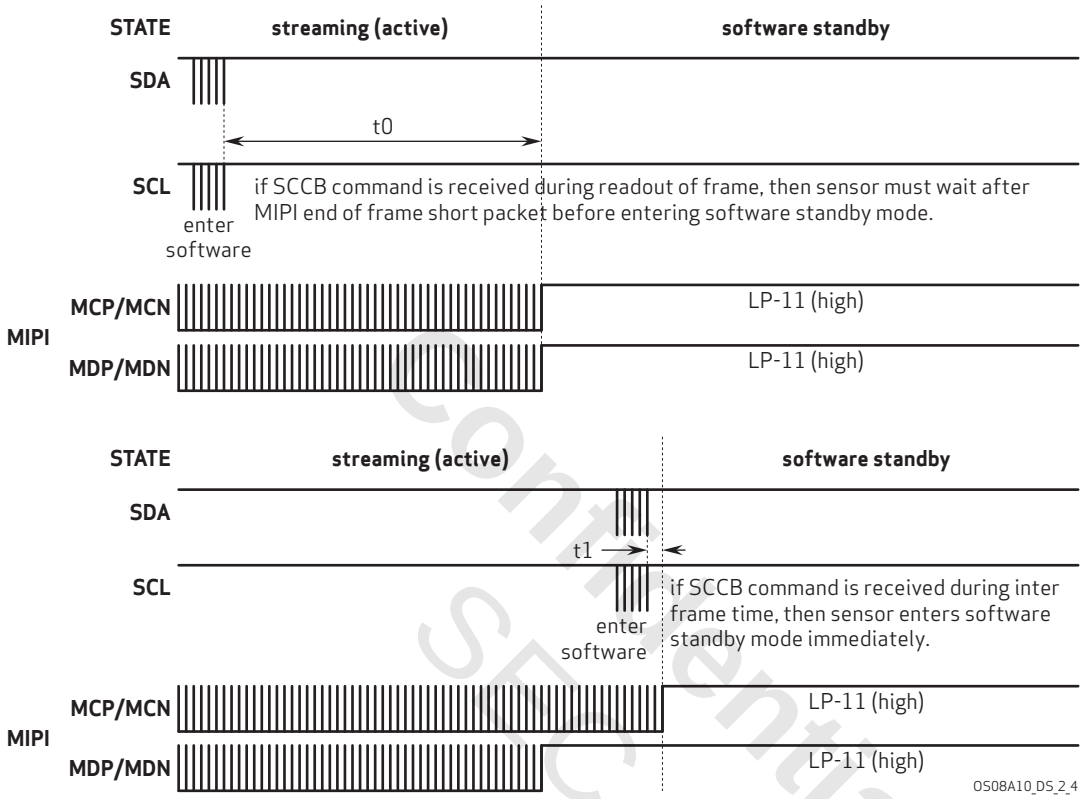
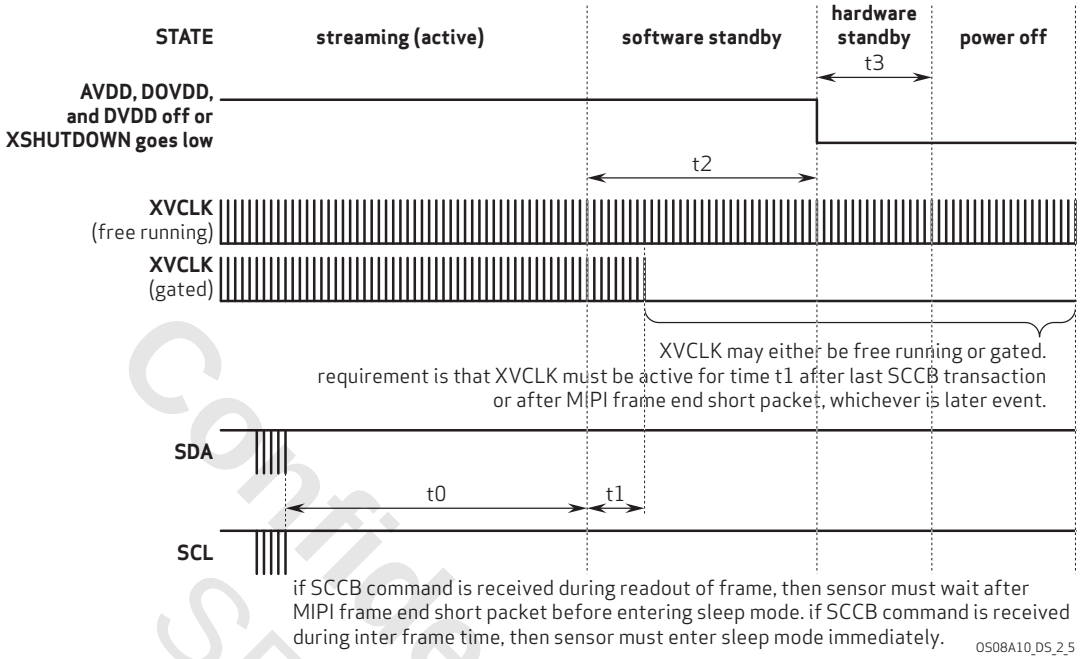


figure 2-5 power down sequence



2.8 reset

The whole chip will be reset during power up.

2.8.1 power ON reset

The power on reset can be controlled from an external pin. Additionally, in this sensor, a power on reset is generated after the core power becomes stable.

2.8.2 software reset

When register 0x0103[0] is configured as 1, all registers are reset to default value.

2.9 hardware and software standby

Two suspend modes are available for the OS08A10:

- hardware standby
- software standby

2.9.1 hardware standby

Dropping any power source (AVDD/DOVDD/DVDD) or if XSHUTDOWN is tied to low, will initiate hardware standby mode. In this mode, the total power consumption will be less than 100 µW.

2.9.2 software standby

Executing a software power down (0x0100[0]) through the SCCB interface suspends internal circuit activity, but does not halt the device clock. All register content is maintained in standby mode. During the resume state, all the registers are restored to their original values.

table 2-5 hardware and standby description

mode	description
hardware standby with XSHUTDOWN	1. enabled by pulling XSHUTDOWN low 2. power down all blocks 3. register values are reset to default values 4. no SCCB communication 5. minimum power consumption
hardware standby with PWDNB	1. default mode after power on reset 2. power down all blocks except SCCB 3. register values are maintained 4. SCCB communication is available 5. low power consumption 6. GPIO can be configured as high/low/tri-state
hardware standby with XSHUTDOWN2	1. enabled by pulling XSHUTDOWN2 low 2. power down digital 3. register values are reset to default values 4. SCCB only access LSM block 5. extreme low power consumption

2.10 system clock control

PLL settings can only be changed during sensor standby mode (0x0100 = 0).

2.10.1 PLL1

The PLL1 generates a default 184 MHz pixel clock and 1500 MHz MIPI serial clock from a 6~64 MHz input clock. The VCO range is from 500 MHz to 1600 MHz. A programmable clock is provided to generate different frequencies.

2.10.2 PLL2

The PLL2 generates a default 144 MHz system clock from a 6~64 MHz input clock. The VCO range is from 500 MHz to 1200 MHz. A programmable clock divider is provided to generate different frequencies.

figure 2-6 clock scheme

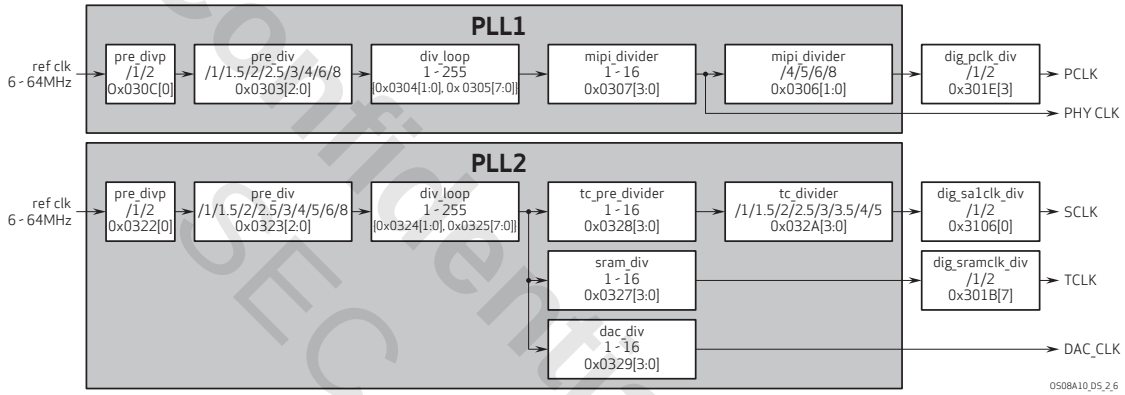


table 2-6 PLL registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x0303	PLL_CTRL_03	0x01	RW	Bit[2:0]: pll1_prediv
0x0304	PLL_CTRL_04	0x00	RW	Bit[1:0]: pll1_divp[9:8]
0x0305	PLL_CTRL_05	0x5C	RW	Bit[7:0]: pll1_divp[7:0]
0x0306	PLL_CTRL_06	0x00	RW	Bit[1:0]: pll1_divmipi
0x0307	PLL_CTRL_07	0x00	RW	Bit[3:0]: pll1_divm
0x0308	PLL_CTRL_08	0x03	RW	Bit[3:0]: pll1_divsp
0x0309	PLL_CTRL_09	0x04	RW	Bit[2:0]: pll1_divs
0x030C	PLL_CTRL_0C	0x00	RW	Bit[0]: pll1_predivp
0x0322	PLL_CTRL_22	0x00	RW	Bit[0]: pll2_predivp

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

address	register name	default value	R/W	description
0x0323	PLL_CTRL_23	0x02	RW	Bit[2:0]: pll2_prediv
0x0324	PLL_CTRL_24	0x00	RW	Bit[1:0]: pll2_divp[9:8]
0x0325	PLL_CTRL_25	0x48	RW	Bit[7:0]: pll2_divp[7:0]
0x0326	PLL_CTRL_26	0x00	RW	Bit[1:0]: pll2_reserve1
0x0327	PLL_CTRL_27	0x05	RW	Bit[3:0]: pll2_divsram
0x0328	PLL_CTRL_28	0x05	RW	Bit[3:0]: pll2_divst
0x0329	PLL_CTRL_29	0x01	RW	Bit[3:0]: pll2_divdac
0x032a	PLL_CTRL_2A	0x00	RW	Bit[2:0]: pll2_divt

2.11 serial camera control bus (SCCB) interface

The Serial Camera Control Bus (SCCB) interface controls the image sensor operation. Refer to the *OmniVision Technologies Serial Camera Control Bus (SCCB) Specification* for detailed usage of the serial control port.

In the OS08A10, the SCCB ID is controlled by the SID pin. If SID is low, the sensor's SCCB ID is 0x6C for write (0x6D for read). If SID is high, the sensor's SCCB ID is 0x20 for write (0x21 fore read). The SCCB ID can also be programmed by registers. When 0x303F[0] is 1, the ID comes from register 0x3004 when SID=0 and register 0x3012 when SID=1.

2.11.1 data transfer protocol

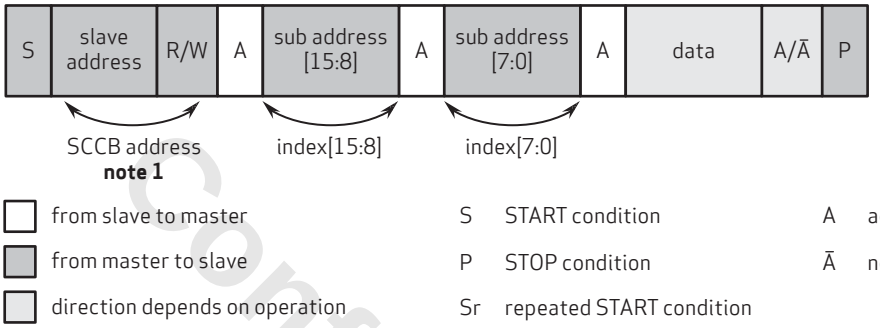
The data transfer of the OS08A10 follows the SCCB protocol.

2.11.2 message format

The OS08A10 supports the message format shown in **figure 2-7**. The repeated START (Sr) condition is not shown in **figure 2-8**, but is shown in **figure 2-9** and **figure 2-10**.

figure 2-7 message type

message type: 16-bit sub-address, 8-bit data, and 7-bit slave address



note 1 slave address must be 0x36 for SCCB write address to be 0x6C and for SCCB read address to be 0x6D

OS08A10_DS_2.7

2.11.3 read / write operation

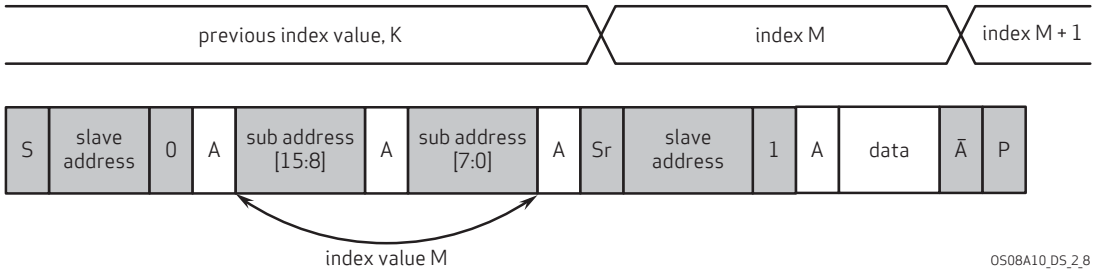
The OS08A10 supports four different read operations and two different write operations:

- a single read from random locations
- a sequential read from random locations
- a single read from current location
- a sequential read from current location
- single write to random locations
- sequential write starting from random location

The sub-address in the sensor automatically increases by one after each read/write operation.

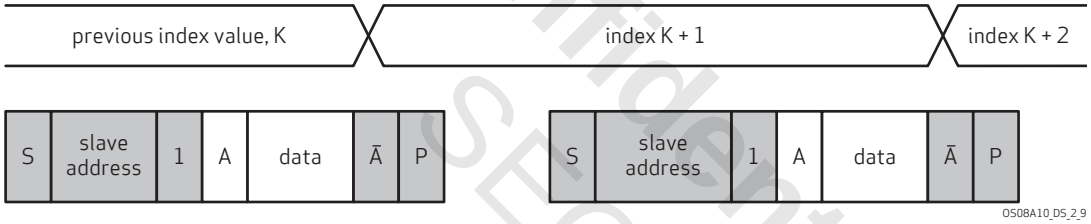
In a single read from random locations, the master does a dummy write operation to desired sub-address, issues a repeated start condition and then addresses the camera again with a read operation. After acknowledging its slave address, the camera starts to output data onto the SDA line as shown in **figure 2-8**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-8 SCCB single read from random location



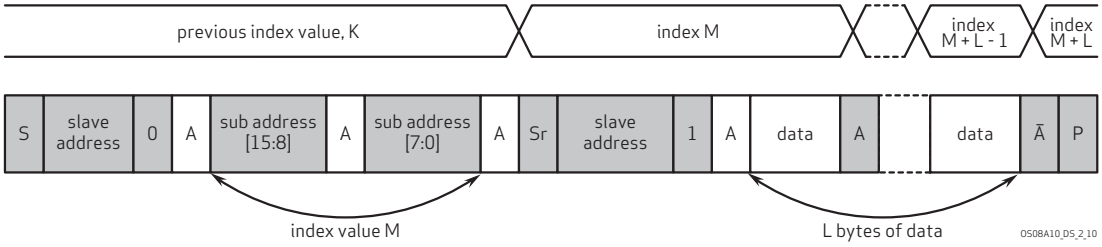
If the host addresses the camera with read operation directly without the dummy write operation, the camera responds by setting the data from last used sub-address to the SDA line as shown in **figure 2-9**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-9 SCCB single read from current location



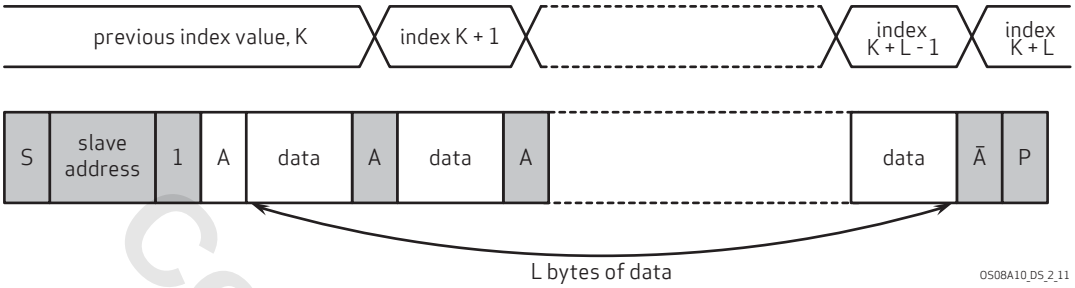
The sequential read from a random location is illustrated in **figure 2-10**. The master does a dummy write to the desired sub-address, issues a repeated start condition after acknowledge from slave and addresses the slave again with read operation. If a master issues an acknowledge after receiving data, it acts as a signal to the slave that the read operation shall continue from the next sub-address. When master has read the last data byte, it issues a negative acknowledge and stop condition.

figure 2-10 SCCB sequential read from random location



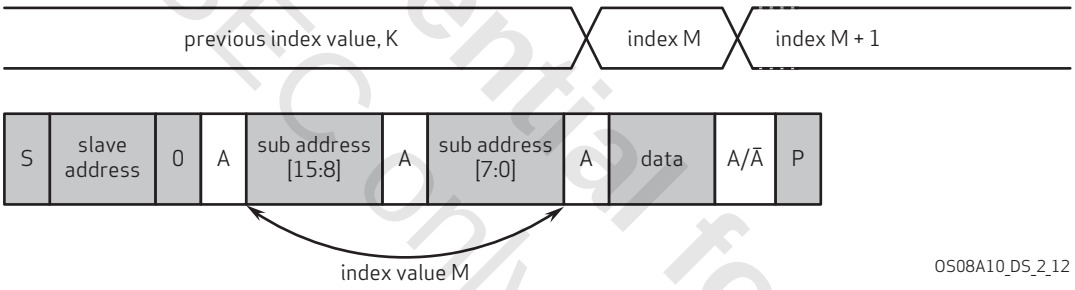
The sequential read from current location is similar to a sequential read from a random location. The only exception is that there is no dummy write operation as shown in **figure 2-11**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-11 SCCB sequential read from current location



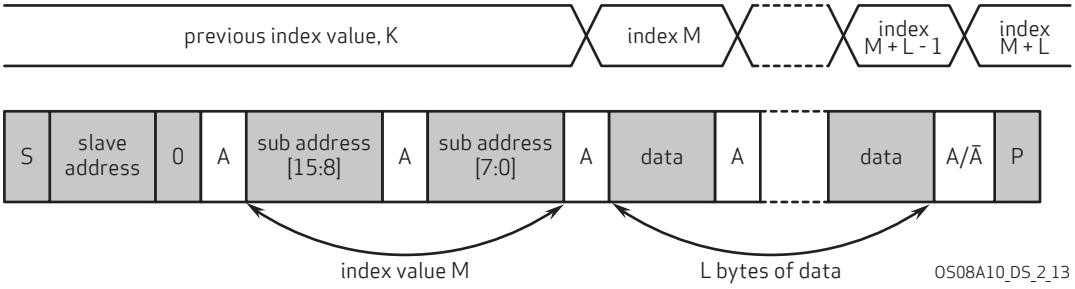
The write operation to a random location is illustrated in **figure 2-12**. The master issues a write operation to the slave, sets the sub-address and data correspondingly after the slave has acknowledged. The write operation is terminated with a stop condition from the master.

figure 2-12 SCCB single write to random location



The sequential write is illustrated in **figure 2-13**. The slave automatically increments the sub-address after each data byte. The sequential write operation is terminated with stop condition from the master.

figure 2-13 SCCB sequential write to random location



2.11.4 SCCB timing

figure 2-14 SCCB interface timing

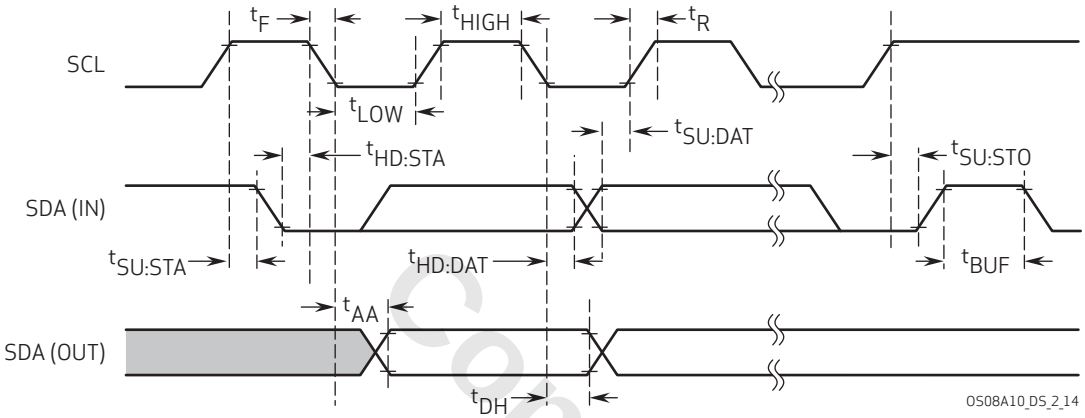


table 2-7 SCCB interface timing specifications[ab]

symbol	parameter	min	typ	max	unit
f_{SCL}	clock frequency			400	kHz
t_{LOW}	clock low period	1.3			µs
t_{HIGH}	clock high period	0.6			µs
t_{AA}	SCL low to data out valid	0.1		0.9	µs
t_{BUF}	bus free time before new start	1.3			µs
$t_{HD:STA}$	start condition hold time	0.6			µs
$t_{SU:STA}$	start condition setup time	0.6			µs
$t_{HD:DAT}$	data in hold time	0			µs
$t_{SU:DAT}$	data in setup time	0.1			µs
$t_{SU:STO}$	stop condition setup time	0.6			µs
t_R, t_F	SCCB rise/fall times			0.3	µs
t_{DH}	data out hold time	0.05			µs

a. SCCB timing is based on 400kHz mode

b. timing measurement shown at beginning of rising edge and/or end of falling edge signifies 30%,
timing measurement shown in middle of rising/falling edge signifies 50%,
timing measurement shown at end of rising edge and/or beginning of falling edge signifies 70%

2.12 group write

Group write is supported in order to update a group of registers in the same frame. These registers are guaranteed to be written prior to the internal latch at the frame boundary.

The OS08A10 supports up to four groups and can be recorded in the same time. These groups share 1024 bytes of memory and the size of each group is programmable by adjusting the start address.

table 2-8 context switching control

address	register name	default value	R/W	description
0x3208	GROUP ACCESS	–	W	Bit[7:4]: group_ctrl 0000: Group hold start 0001: Group hold end 1010: Group launch 1110: Fast group launch Others: Reserved Bit[3:0]: Group ID 0000: Group bank 0 0001: Group bank 1 0010: Group bank 2 0011: Group bank 3 Others: Reserved
0x3209	GROUP0 PERIOD	0x00	RW	Number of Frames to Stay in Group 0
0x320A	GROUP1 PERIOD	0x00	RW	Number of Frames to Stay in Group 1
0x320B	GROUP2 PERIOD	0x00	RW	Number of Frames to Stay in Group 2
0x320C	GROUP3 PERIOD	0x00	RW	Number of Frames to Stay in Group 3
0x320D	GRP_SWCTRL	0x01	RW	Bit[7:6]: Reserved Bit[5]: Repeat switch mode Bit[4]: context_en Bit[3:2]: Reserved Bit[1:0]: Switch back group
0x321A	GRP_ACT	–	R	Active Group Indicator
0x321C	FRAME_CNT_GRP0	–	R	Group 0 Frame Count
0x321D	FRAME_CNT_GRP1	–	R	Group 1 Frame Count
0x321E	FRAME_CNT_GRP2	–	R	Group 2 Frame Count
0x321F	FRAME_CNT_GRP3	–	R	Group 3 Frame Count

2.12.1 hold

After the groups are configured, users can perform a hold operation to store register settings into the SRAM of each group. The hold of each group starts and ends with the control register 0x3208. The lower 4 bits of register 0x3208 control which group to access, and the upper 4 bits control the start (0x0: hold start) and end (0x1: hold end) of the hold operation.

The example setting below shows the sequence to hold group 0:

```
6C  320D  00    clear for context switching
6C  3208  00    group 0 hold start
6C  0808  00    declare each group hold start
6C  3800  11    first register into group 0
6C  3911  22    second register into group 0
6C  3208  10    group 0 hold end
```

2.12.2 launch

After the contents of each group are defined in the hold operation, all registers belonging to each group are stored in SRAM, and ready to be written into target registers (i.e., the launch of that group).

There are five launch modes as described in sections **section 2.12.2.1** to **section 2.12.2.5**.

2.12.2.1 launch mode 1 – quick manual launch

Manual launch is enabled by setting the register 0x320B to 0.

Quick manual launch is achieved by writing to control register 0x3208. The value written into this register is 0xEX, the upper 4 bits (0xE) are the quick launch command and the lower 4 bits (0xX) are the group number. For example, if users want to launch group 0, they just write the value 0xE0 to 0x3208, then the contents of group 0 will be written to the target registers immediately after the sensor gets this command through the SCCB. Below is a setting example.

```
6C  320D  00    manual launch on
6C  3208  E0    delay launch group 1
```

2.12.2.2 launch mode 2 – delay manual launch

Delay manual launch is achieved by writing to register 0x3208. The value written into this register is 0xAX, where the upper 4 bits (0xA) are the delay launch command and the lower 4 bits (0xX) are the group number. For example, if users want to launch group 1, they just write the value 0xA1 to 0x3208, then the contents of group 1 will be written to the target registers. The difference with mode 1 is that the writing will wait for some internally defined time spot in vertical blanking, thus delayed. Below is a setting example.

```
6C  320D  00    manual launch on
6C  3208  A1    delay launch group 1
```

2.12.2.3 launch mode 3 – quick auto launch

In addition to quick launch features like in mode 1, quick auto launch will switch groups automatically. This is controlled by the register 0x320D, where bit[4] enables mode and bit[1:0] controls which group to return. The auto launch will start from a user specified launch group. Then, it will increment group numbers until it reaches the return group. If the highest group has been reached before the return group, it will switch to the lowest index group and resume incrementing until th return group is reached (i.e., start group = 2, return group = 1). For each group, users can specify how many frames to stay. If the user specifies 0 frames to stay, then auto launch will skip that group(s). Frame numbers 0x3209, 0x320A, 0x320B, and 0x320C for groups 0, 1, 2, and 3, respectively, control how many frames to stay before returning. The operation can be better understood with a setting example:

```
6C  320D  12    [1:0]: 1, return to group 2, [4]: auto switch enable
6C  3209  04    stay in group 0 for 4 frames
6C  320A  01    stay in group 1 for 1 frame
6C  320B  02    stay in group 2 for 2 frames
6C  320C  05    stay in group 3 for 5 frames
6C  3208  E3    quick launch group 3
```

In this example, the sensor will quick launch group 3 for 5 frames, switch to group 1 for 1 frame, and then switch to group 2.

Launching quick auto launch in stagger HDR mode has additional requirements, such as:

1. User needs to write "R0x3833[0]=1" prior to group launch.
2. User needs to drop first short channel frame after launching context switching.

2.12.2.4 launch mode 4 – delay auto launch

In addition to delay launch features like in mode 2, delay auto launch will switch groups automatically using incrementing indices until the return group number is reached. Similar to mode 3, specifying 0 frames to stay will skip that group(s).

The operation can be better understood with a setting example:

```
6C  320D  11    [1:0]: 1, return to group 1, [4]: enable auto switch
6C  3209  04    stay in group 0 for 4 frames
6C  320A  01    number of frames to stay in group 1
6C  3208  A0    delay launch group 0
```

In this example, the sensor will delay launch group 0, stay at group 0 for 4 frames, then return to group 1.

Launching delay auto launch in stagger HDR mode has additional requirements, such as:

1. User needs to write "R0x3833[0]=1" prior to group launch.
2. User needs to drop first short channel frame after launching context switching.

2.12.2.5 launch mode 5 – repeat launch

In repeat launch, up to four groups can be launched in incrementing order, looping indefinitely, until a new register setting is written. The user can specify how many frames to stay for each group. Groups can be skipped by writing zero to the stay frames register. In order to run a repeat launch, the first group hold needs to be set along with stay frame numbers. Then, the user can enable repeat launch using register bit 0x320D[5]. Stay frame numbers can be specified by registers 0x3209, 0x320A, 0x320B, and 0x320C for groups 0, 1, 2, and 3, respectively.

The operation can be better understood with a setting example:

```
6C  320D  00    clear for context switching
6C  3208  00    group 0 hold start
6C  0808  00    declare each group hold start
6C  3508  00    first register into group 0
6C  3509  80    second register into group 0
6C  3208  10    group 0 hold end

6C  3208  01    group 1 hold start
6C  0808  00    declare each group hold start
...
6C  3208  11    group 1 hold end

6C  3208  02    group 2 hold start
6C  0808  00    declare each group hold start
...
6C  3208  12    group 2 hold end

6C  3208  03    group 3 hold start
6C  0808  00    declare each group hold start
...
6C  3208  13    group 3 hold end

6C  3209  64    stay 100 frames in group 0
6C  320A  00    stay 0 frames in group 1 (=skip group 1)
6C  320B  32    stay 50 frames in group 2
6C  320C  64    stay 100 frames in group 3
6C  320D  30    [5]: repeat switch enable, [4] context switch enable
6C  3208  A0    always use a0 for repeat launch
```

In this example, the sensor will repeat launch group 0, stay at group 0 for 100 frames, skip group 1 since stay frames for group 1 is set to 0, switch to group 2 for 50 frames, switch to group 3 for 100 frames, then back to group 0 for 100 frames, and so on until the user sends a register write.

Launching delay auto launch in stagger HDR mode has additional requirements, such as:

1. User needs to write "R0x3833[0]=1" prior to group launch.
2. User needs to drop first short channel frame after launching context switching.

2.13 power saving mode

In power saving mode, the sensor operates in low power mode during vertical blanking. The sensor will enter power saving mode using register 0x3406 × 4 rows before entering vblanking and exit out of power saving mode using register 0x3409[3:0] × 4 rows after vblanking ends.

table 2-9 power saving mode

address	register name	default value	R/W	description
0x3406	R STREAM ST OFFS	0x08	RW	Bit[7:0]: r_stream_st_offs
0x3409	CTRL09	0x22	RW	Bit[3:0]: r_strm_rear_offs

3 block level description

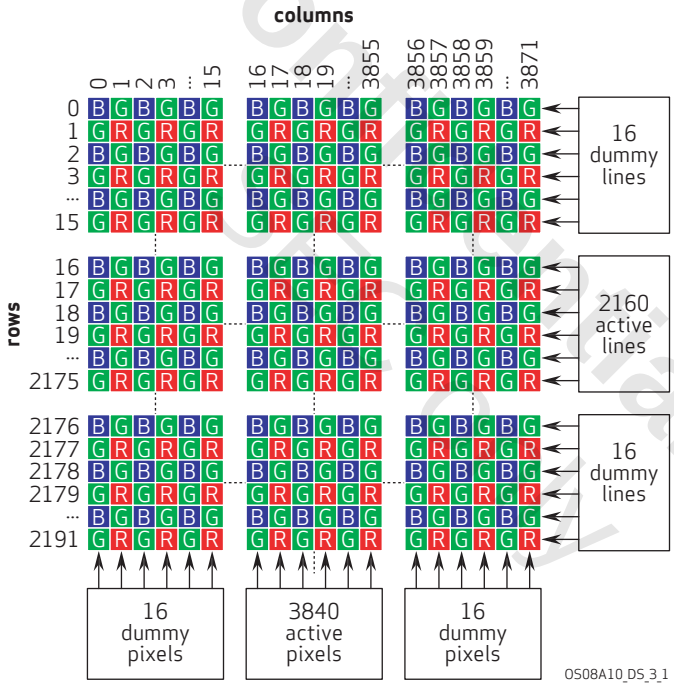
3.1 pixel array structure

The OS08A10 sensor has an image array of 3872 columns by 2192 rows (8,487,424 pixels). **figure 3-1** shows a cross-section of the image sensor array.

The color filters are arranged in a Bayer pattern. The primary color BG/GR array is arranged in line-alternating fashion. Of the 8,487,424 pixels, 8,294,400 (3840x2160) are active pixels and can be output.

The sensor array design is based on a field integration readout system with line-by-line transfer and an electronic shutter with a synchronous pixel readout scheme.

figure 3-1 sensor output pattern (3840x2160)



3.2 subsampling

The OS08A10 supports a binning mode to provide a lower resolution output while maintaining the field of view. With binning mode ON, the voltage levels of adjacent pixels (of the same color) are averaged before being sent to the ADC. The OS08A10 supports 2x2 binning, which is illustrated in **figure 3-2** and **figure 3-3**, where the voltage levels of two horizontal (2x1) adjacent same-color pixels are averaged.

figure 3-2 example of 2x2 RGB binning

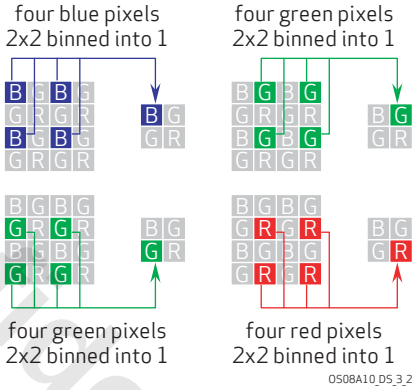


figure 3-3 example of 2x2 mono binning

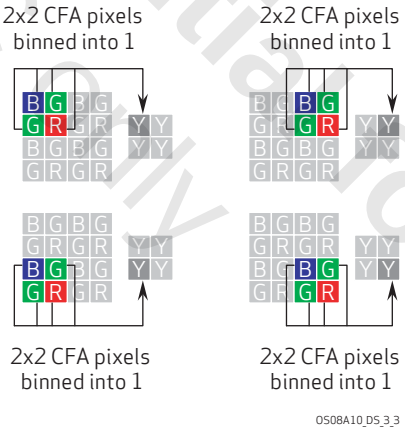


table 3-1 binning-related registers

address	register name	default value	R/W	description
0x3814	X_ODD_INC	0x01	RW	Bit[4:0]: Horizontal increase number at odd pixel
0x3815	X_EVEN _INC	0x01	RW	Bit[4:0]: Horizontal increase number at even pixel
0x3820	FORMAT1	0x80	RW	Bit[2]: Flip Bit[1]: Vertical mono binning Bit[0]: Vertical binning
0x3821	FORMAT2	0x00	RW	Bit[7:6]: Digital in digital domain Bit[1]: Horizontal mono binning Bit[0]: Horizontal binning
0x3816	Y_ODD_INC	0x01	RW	Bit[4:0]: Vertical increase number at odd row
0x3817	Y_EVEN_INC	0x01	RW	Bit[4:0]: Vertical increase number at even row

3.3 analog amplifier

When the column sample/hold circuit has sampled one row of pixels, the pixel data will shift out one-by-one into an analog amplifier.

3.4 12-bit A/D converters

The balanced signal is then digitized by the on-chip 12-bit ADC.

4 image sensor core digital functions

4.1 mirror and flip

The OS08A10 provides mirror and flip readout modes, which respectively reverse the sensor data readout order horizontally and vertically (see **figure 4-1**).

figure 4-1 mirror and flip samples

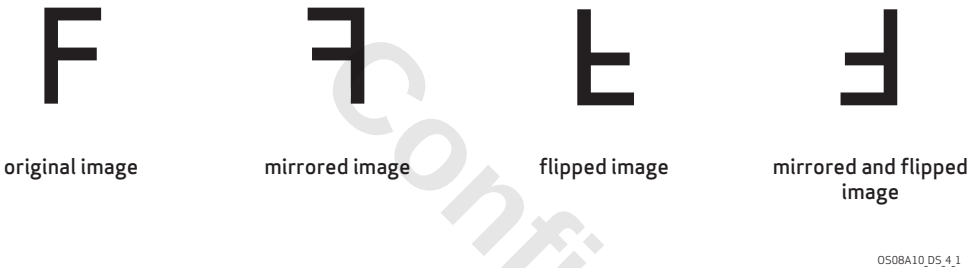


table 4-1 mirror and flip registers

address	register name	default value	R/W	description
0x3820	FORMAT1	0x00	RW	Bit[2]: Flip
0x3821	FORMAT2	0x00	RW	Bit[2]: Mirror

4.2 image windowing

An image windowing area is defined by four parameters, horizontal start (HS), horizontal end (HE), vertical start (VS), and vertical end (VE). By properly setting the parameters, any portion within the sensor array size can output as a visible area. Windowing is achieved by masking off the pixels outside of the window; thus, the original timing is not affected.

figure 4-2 image windowing

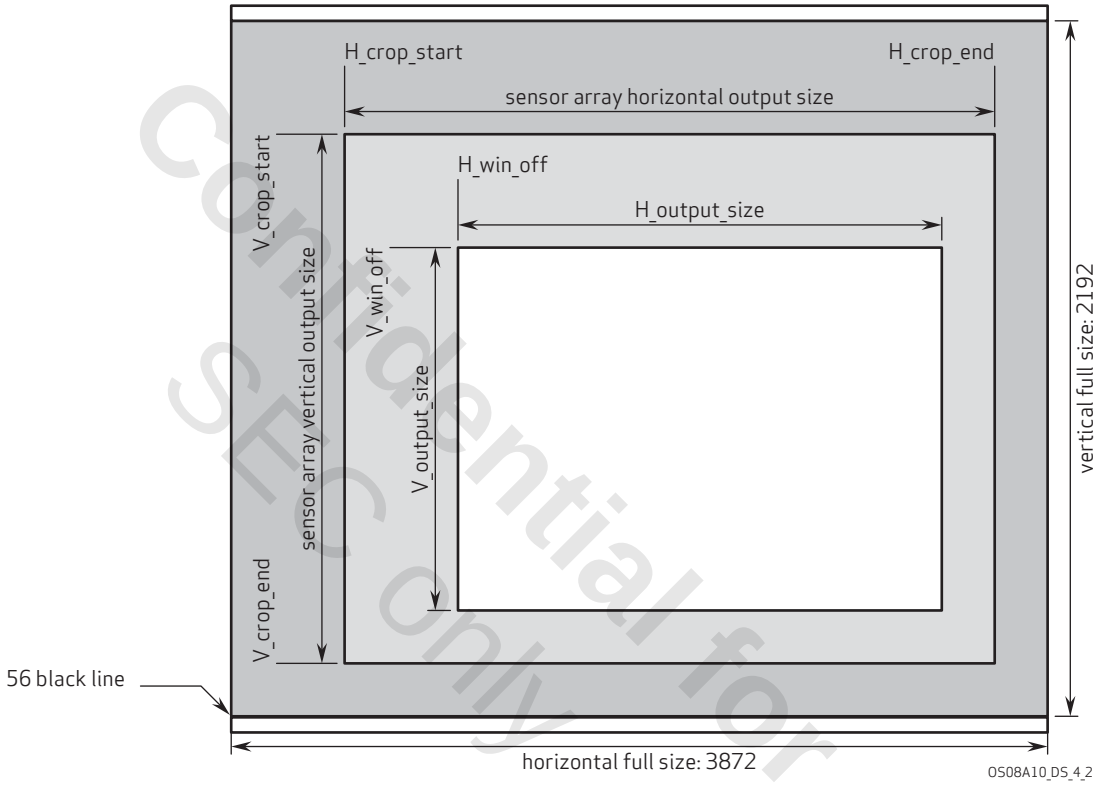


table 4-2 image windowing control functions (sheet 1 of 2)

address	register name	default value	R/W	description
0x3800	X ADDR START	0x00	RW	Bit[3:0]: x_addr_start[11:8] Array horizontal start point
0x3801	X ADDR START	0x00	RW	Bit[7:0]: x_addr_start[7:0] Array horizontal start point
0x3802	Y ADDR START	0x00	RW	Bit[3:0]: y_addr_start[11:8] Array vertical start point
0x3803	Y ADDR START	0x04	RW	Bit[7:0]: y_addr_start[7:0] Array vertical start point
0x3804	X ADDR END	0x07	RW	Bit[3:0]: x_addr_end[11:8] Array horizontal end point
0x3805	X ADDR END	0x8F	RW	Bit[7:0]: x_addr_end[7:0] Array horizontal end point
0x3806	Y ADDR END	0x04	RW	Bit[3:0]: y_addr_end[11:8] Array vertical end point
0x3807	Y ADDR END	0x43	RW	Bit[7:0]: y_addr_end[7:0] Array vertical end point
0x3808	X OUTPUT SIZE	0x07	RW	Bit[3:0]: x_output_size[11:8] ISP horizontal output width
0x3809	X OUTPUT SIZE	0x80	RW	Bit[7:0]: x_output_size[7:0] ISP horizontal output width
0x380A	Y OUTPUT SIZE	0x04	RW	Bit[3:0]: y_output_size[11:8] ISP vertical output height
0x380B	Y OUTPUT SIZE	0x38	RW	Bit[7:0]: y_output_size[7:0] ISP vertical output height
0x380C	HTS	0x02	RW	Bit[7:0]: HTS[15:8] Total horizontal timing size
0x380D	HTS	0x78	RW	Bit[7:0]: HTS[7:0] Total horizontal timing size
0x380E	VTS	0x04	RW	Bit[7:0]: VTS[15:8] Total vertical timing size
0x380F	VTS	0xA0	RW	Bit[7:0]: VTS[7:0] Total vertical timing size
0x3810	ISP X WIN	0x00	RW	Bit[7:0]: isp_x_win[15:8] ISP horizontal windowing offset
0x3811	ISP X WIN	0x08	RW	Bit[7:0]: isp_x_win[7:0] ISP horizontal windowing offset

table 4-2 image windowing control functions (sheet 2 of 2)

address	register name	default value	R/W	description
0x3812	ISP Y WIN	0x00	RW	Bit[3:0]: isp_y_win[11:8] ISP vertical windowing offset
0x3813	ISP Y WIN	0x04	RW	Bit[7:0]: isp_y_win[7:0] ISP vertical windowing offset
0x3814	X INC ODD	0x01	RW	Bit[4:0]: x_odd_inc
0x3815	X INC EVEN	0x01	RW	Bit[4:0]: x_even_inc
0x3816	Y_INC_ODD	0x01	RW	Bit[4:0]: y_odd_inc
0x3817	Y_INC_EVEN	0x01	RW	Bit[4:0]: y_even_inc

4.3 test pattern

For testing purposes, the OS08A10 offers three types of test patterns: color bar, square and random data. The OS08A10 also offers two digital effects: transparent effect and rolling bar effect. The output type of digital test pattern is controlled by the test_pattern_type register (0x5081[3:2]). The digital test pattern function is controlled by register 0x5081[7].

4.3.1 color bar

There are four types of color bars, which are switched by bar-style in register 0x5081[3:2] (see **figure 4-3**).

figure 4-3 color bar types

4.3.2 square

There are two types of squares: color square and black-white square. The squ_bw register (0x5081[4]) determines which type of square will be output.

figure 4-4 color, black and white square bars

OS08A10_DS_4_4

4.3.3 random data

There are two types of random data test patterns: frame-changing and frame-fixed random data.

4.3.4 transparent effect

The transparent effect is enabled by transparent_en register (0x5081[5]). If this register is set, the transparent test pattern will be displayed. **figure 4-5** is an example showing a transparent color bar image.

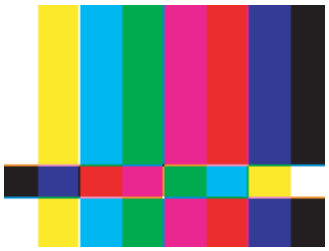
figure 4-5 transparent effect

OS08A10_DS_4_5

4.3.5 rolling bar effect

The rolling bar is set by rolling_bar_en register (0x5081[6]). If it is set, an inverted-color rolling bar will roll from up to down. **figure 4-6** is an example showing a rolling bar on color bar image.

figure 4-6 rolling bar effect



OS08A10_DS_4_6

table 4-3 test pattern register

address	register name	default value	R/W	description
0x5081	ISP CTRL 1	0x00	RW	Bit[7]: test_en Bit[6]: rolling Bit[5]: trans Bit[4]: squ_bw Bit[3:2]: bar_style Bit[1:0]: test_sel

4.4 average luminance (YAVG)

Exposure time control is based on a frame brightness average value. The OS08A10 supports the average image luminance calculation. By properly setting X_start, Y_start, and window_width and window_height, the user can adjust the average based window.

figure 4-7 average based window definition

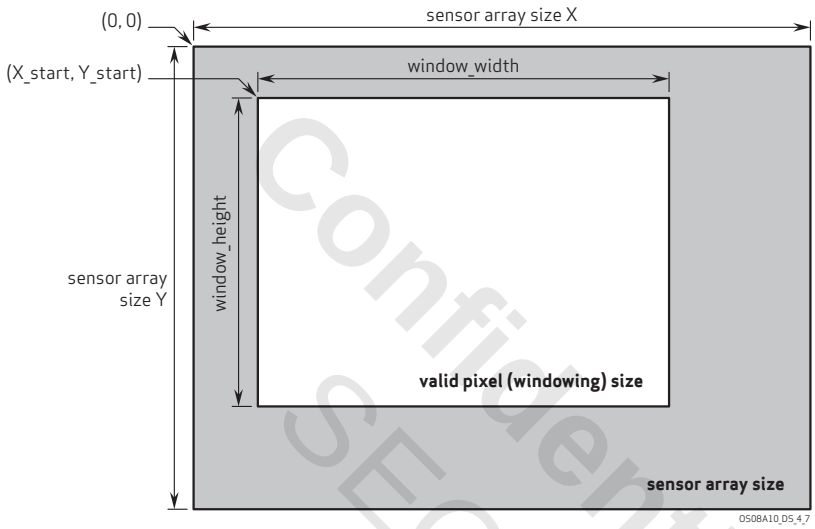


table 4-4 average control registers

address	register name	default value	R/W	description
0x4B00	AVG CTRL00	0x00	RW	Bit[0]: X_start_avg[8]
0x4B01	AVG CTRL01	0x00	RW	Bit[7:0]: X_start_avg[7:0]
0x4B03	AVG CTRL03	0x00	RW	Bit[7:0]: Y_start_avg[7:0]
0x4B04	AVG CTRL04	0x01	RW	Bit[0]: Window_width_avg[8]
0x4B05	AVG CTRL05	0x40	RW	Bit[7:0]: Window_width_avg[7:0]
0x4B07	AVG CTRL07	0xF0	RW	Bit[7:0]: Window_height_avg[7:0]
0x4B13	AVG RO13	–	R	Bit[7:0]: High 8 bits of whole image's average output
0x4B14	AVG G	–	R	Average G Channel Read Out
0x4B15	AVG R	–	R	Average R Channel Read Out
0x4B16	AVG Y	–	R	Average Y Channel Read Out

4.5 black level calibration (BLC)

The pixel array contains several optically shielded (black) lines and optically shielded (black) pixels on the right side. These lines and columns are used as reference for black level calibration. The main function of the BLC is to adjust all normal pixel values based on the values of the black levels.

table 4-5 BLC control registers (sheet 1 of 5)

address	register name	default value	R/W	description
0x4000	BLC CTRL00	0xF8	RW	Bit[7]: offset_trigger_en Bit[6]: exp_trig_en Bit[5]: gain_trig_en Bit[4]: fmt_trig_en Bit[3]: rst_trig_en Bit[2]: man_trig_en Bit[1]: blc_freeze Bit[0]: blc_always_update
0x4001	BLC CTRL01	0x2B	RW	Bit[7]: zero_in_out_en Bit[6]: blc_In_out_en Bit[5]: blc_dithering_en Bit[4]: man_offset_en Bit[3]: median_filter_en Bit[2]: v15_one_chnl_en Bit[1]: dc_blc_en Bit[0]: blc_en
0x4004	BLK LVL TARGET	0x00	RW	Bit[3:0]: blc_lvl_target[11:8]
0x4005	BLK LVL TARGET	0x40	RW	Bit[7:0]: blc_lvl_target[7:0]
0x4006	HWIN PAD	0x00	RW	Bit[3:0]: hwin_pad[11:8]
0x4007	HWIN PAD	0x04	RW	Bit[7:0]: hwin_pad[7:0]
0x4008	BLC CTRL08	0x00	RW	Bit[7:0]: bl_start
0x4009	BLC CTRL09	0x0F	RW	Bit[7:0]: bl_end
0x400A	OFF LIM TH	0x02	RW	Bit[4:0]: off_lim_th[12:8]
0x400B	OFF LIM TH	0x00	RW	Bit[7:0]: off_lim_th[7:0]
0x400C	HWIN OFF	0x00	RW	Bit[3:0]: hwin_off[11:8]
0x400D	HWIN OFF	0x00	RW	Bit[7:0]: hwin_off[7:0]
0x400E	BLC CTRL0E	0x00	RW	Bit[7:0]: mf_th

table 4-5 BLC control registers (sheet 2 of 5)

address	register name	default value	R/W	description
0x400F	BLC CTRL0F	0xA0	RW	Bit[7]: r_threshold_en Bit[6]: r_set_zb Bit[5:4]: output_bits 00: 10 bits 01: 11 bits 10: 12 bits Bit[2]: r_en_adp_k Bit[1]: r_dc_offset_mode Bit[0]: r_compute_offset_v15
0x4010	BLC CTRL10	0x12	RW	Bit[7]: r_img_gfirst_rvs Bit[6]: r_blk_rblue_rvs Bit[5]: r_img_rblue_rvs Bit[4]: r_dc_man Bit[3]: target_adj_dis Bit[2]: cmp_en Bit[1]: kcoef_man_en Bit[0]: man_avg_en
0x4011	BLC CTRL11	0xFF	RW	Bit[7]: r_off_cmp_man_en Bit[6]: off_chg_mf_en Bit[5]: fmt_chg_mf_en Bit[4]: gain_chg_mf_en Bit[3]: rst_mf_mode Bit[2]: off_chg_mf_mode Bit[1]: fmt_chg_mf_mode Bit[0]: gain_chg_mf_mode
0x4012	BLC CTRL12	0x08	RW	Bit[7:0]: rst_trig_fn
0x4013	BLC CTRL13	0x02	RW	Bit[7:0]: fmt_trig_fn
0x4014	BLC CTRL14	0x02	RW	Bit[7:0]: gain_trig_fn
0x4015	BLC CTRL15	0x02	RW	Bit[7:0]: off_trig_fn
0x4016	OFF TRIG TH	0x00	RW	Bit[1:0]: off_trig_th[9:8] off_trig_th high 2 bits
0x4017	OFF TRIG TH	0x04	RW	Bit[7:0]: off_trig_th[7:0] off_trig_th low 8 bits
0x4019	BLC CTRL19	0x04	RW	Bit[7:0]: r_blk_ln_num
0x401A	BLC CTRL1A	0x48	RW	Bit[7]: r_h_size_man_en Bit[6]: r_kcoef_mirror Bit[5]: r_adp_dc_switch_en Bit[4]: gain_trig_beh Bit[3]: format_trig_beh Bit[2]: r_ln_man Bit[1:0]: byp_mode CZ V1.54 change start

table 4-5 BLC control registers (sheet 3 of 5)

address	register name	default value	R/W	description
0x4020	BLC CTRL20	0x00	RW	Bit[7:0]: off_cmp_th0000
0x4021	BLC CTRL21	0x00	RW	Bit[7:0]: off_cmp_th0001
0x4022	BLC CTRL22	0x00	RW	Bit[7:0]: off_cmp_th0010
0x4023	BLC CTRL23	0x00	RW	Bit[7:0]: off_cmp_th0011
0x4024	BLC CTRL24	0x00	RW	Bit[7:0]: off_cmp_th0100
0x4025	BLC CTRL25	0x00	RW	Bit[7:0]: off_cmp_th0101
0x4026	BLC CTRL26	0x00	RW	Bit[7:0]: off_cmp_th0110
0x4027	BLC CTRL27	0x00	RW	Bit[7:0]: off_cmp_th0111
0x4028	THRES CTRL	0x4F	RW	Bit[6]: r_lim_bits_en Bit[5]: r_thres_en Bit[4]: r_thres_man_en Bit[3:0]: r_thres_hist_cutoff
0x4029	R THRES DELTA CUTOFF	0x01	RW	Bit[4:0]: r_thres_delta_cutoff[4:0]
0x402A	R THRES DELTA	0x00	RW	Bit[7:0]: r_thres_delta[7:0]
0x402B	R THRES LINE	0x00	RW	Bit[5:0]: r_thres_line[5:0]
0x402C	R THRES MAN	0x00	RW	Bit[3:0]: r_thres_man[11:8]
0x402D	R THRES MAN	0x00	RW	Bit[7:0]: r_thres_man[7:0]
0x402E~ 0x402F	NOT USED	–	–	Not Used
0x4040	OFF MAN0000	0x00	RW	Bit[4:0]: off_man0000[12:8]
0x4041	OFF MAN0000	0x00	RW	Bit[7:0]: off_man0000[7:0]
0x4042	OFF MAN0001	0x00	RW	Bit[4:0]: off_man0001[12:8]
0x4043	OFF MAN0001	0x00	RW	Bit[7:0]: off_man0001[7:0]
0x4044	OFF MAN0010	0x00	RW	Bit[4:0]: off_man0010[12:8]
0x4045	OFF MAN0010	0x00	RW	Bit[7:0]: off_man0010[7:0]
0x4046	OFF MAN0011	0x00	RW	Bit[4:0]: off_man0011[12:8]
0x4047	OFF MAN0011	0x00	RW	Bit[7:0]: off_man0011[7:0]
0x4048	OFF MAN0100	0x00	RW	Bit[4:0]: off_man0100[12:8]
0x4049	OFF MAN0100	0x00	RW	Bit[7:0]: off_man0100[7:0]
0x404A	OFF MAN0101	0x00	RW	Bit[4:0]: off_man0101[12:8]
0x404B	OFF MAN0101	0x00	RW	Bit[7:0]: off_man0101[7:0]

table 4-5 BLC control registers (sheet 4 of 5)

address	register name	default value	R/W	description
0x404C	OFF MAN0110	0x00	RW	Bit[4:0]: off_man0110[12:8]
0x404D	OFF MAN0110	0x00	RW	Bit[7:0]: off_man0110[7:0]
0x404E	OFF MAN0111	0x00	RW	Bit[4:0]: off_man0111[12:8]
0x404F	OFF MAN0111	0x00	RW	Bit[7:0]: off_man0111[7:0]
0x4050	BLC CTRL50	0x04	RW	Bit[7:0]: zl_start
0x4051	BLC CTRL51	0x0B	RW	Bit[7:0]: zl_end
0x4052	KCOEF B MAN	0x01	RW	Bit[1:0]: kcoef_b_man[9:8]
0x4053	KCOEF B MAN	0x00	RW	Bit[7:0]: kcoef_b_man[7:0]
0x4054	KCOEF GB MAN	0x01	RW	Bit[1:0]: kcoef_gb_man[9:8]
0x4055	KCOEF GB MAN	0x00	RW	Bit[7:0]: kcoef_gb_man[7:0]
0x4056	KCOEF GR MAN	0x01	RW	Bit[1:0]: kcoef_gr_man[9:8]
0x4057	KCOEF GR MAN	0x00	RW	Bit[7:0]: kcoef_gr_man[7:0]
0x4058	KCOEF R MAN	0x01	RW	Bit[1:0]: kcoef_r_man[9:8]
0x4059	KCOEF R MAN	0x00	RW	Bit[7:0]: kcoef_r_man[7:0]
0x405A	BLC CTRL5A	0x30	RW	Bit[7:0]: r_l_dc_th_1_0
0x405B	BLC CTRL5B	0x18	RW	Bit[7:0]: r_l_dc_th_2_0
0x405C	BLC CTRL5C	0x00	RW	Bit[5:0]: avg_weight
0x405D	RND GAIN TH	0x00	RW	Bit[1:0]: rnd_gain_th[9:8] rnd_gain_th high 2 bits
0x405E	RND GAIN TH	0x00	RW	Bit[7:0]: rnd_gain_th[7:0] rnd_gain_th low 8 bits
0x405F	BLC CTRL5F	0x00	RW	Bit[7:0]: r_s_dc_th_1_0
0x4060	BLC CTRL60	0x00	RW	Bit[7:0]: r_s_dc_th_2_0
0x4061	THRES L	–	R	Bit[4:0]: thres_l[12:8]
0x4062	THRES L	–	R	Bit[7:0]: thres_l[7:0]
0x4063	THRES S	–	R	Bit[4:0]: thres_s[12:8]
0x4064	THRES S	–	R	Bit[7:0]: thres_s[7:0]
0x4065	ZERO LN NUM	0x00	RW	Bit[1:0]: zero_ln_num[9:8] Zero line number high 2 bits
0x4066	ZERO LN NUM	0x02	RW	Bit[7:0]: zero_ln_num[7:0] Zero line number low 8 bits

table 4-5 BLC control registers (sheet 5 of 5)

address	register name	default value	R/W	description
0x4067	COL WIN	0x18	RW	Bit[7:0]: col_win
0x4068	R COL LOW GAIN	0x00	RW	Bit[1:0]: r_col_low_gain[9:8] r_col_low_gain high 2 bits
0x4069	R COL LOW GAIN	0x20	RW	Bit[7:0]: r_col_low_gain[7:0] r_col_low_gain low 8 bits
0x406A	R COL HIGH GAIN	0x00	RW	Bit[1:0]: r_col_high_gain[9:8] r_col_high_gain high 2 bits
0x406B	R COL HIGH GAIN	0x40	RW	Bit[7:0]: r_col_high_gain[7:0] r_col_high_gain low 8 bits
0x406C	BLC CTRL6C	0x02	RW	Bit[4]: Calibration no mirror select Bit[3]: col_real_gain_sel Bit[2:0]: r_col_div_cnt
0x406D	BLC CTRL6D	0x00	RW	Bit[7:0]: mf_col_th
0x406E	R H SIZE MAN	0x00	RW	Bit[4:0]: r_h_size_man[12:8]
0x406F	R H SIZE MAN	0x00	RW	Bit[7:0]: r_h_size_man[7:0]

4.6 strobe flash

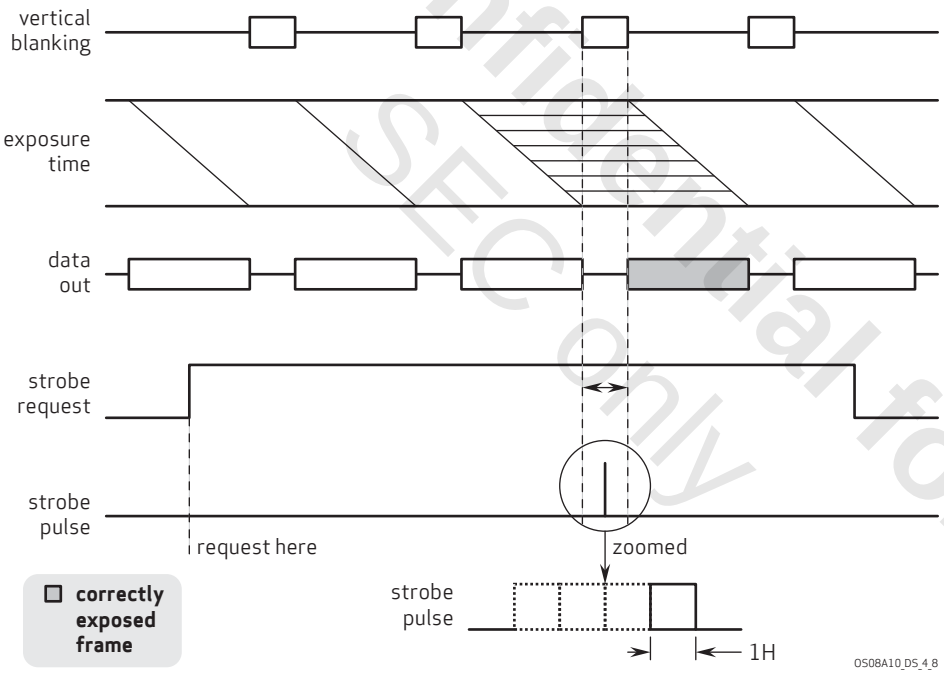
4.6.1 strobe flash control

The strobe signal is programmable using register 0x3B00[2:0]. It supports both LED and Xenon modes. The polarity of the pulse can be changed. The strobe signal is enabled (turned high/low depending on the pulse's polarity) by requesting the signal via the SCCB interface using register bit 0x3B00[7]. Flash modules are triggered by the rising edge by default or by the falling edge if the signal polarity is changed. it supports the following flashing modes: Xenon flash control, LED mode 1, LED mode 2, LED mode 3, and LED mode 4.

4.6.1.1 xenon flash control

After a strobe request is submitted, the strobe pulse will be activated at the beginning of the third frame (see **figure 4-8**). The third frame will be correctly exposed. The pulse width can be changed in Xenon mode between 1H and 4H using register 0x3B00[5:4], where H is one row period.

figure 4-8 xenon flash mode



4.6.1.2 LED 1 & 2 mode

In LED 1 & 2 mode, the strobe signal width can be added by inserting dummy lines, which is controlled by registers {0x3802, 0x3803} (see **figure 4-9**). The strobe will be toggled during vertical blanking for every frame until the user disables the strobe request using register bit 0x3800[7].

figure 4-9 LED 1 & 2 mode (part a)

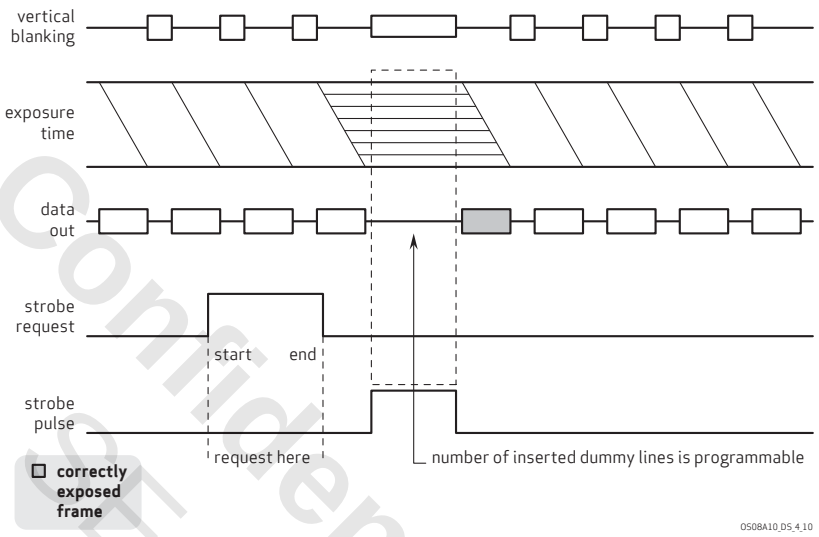
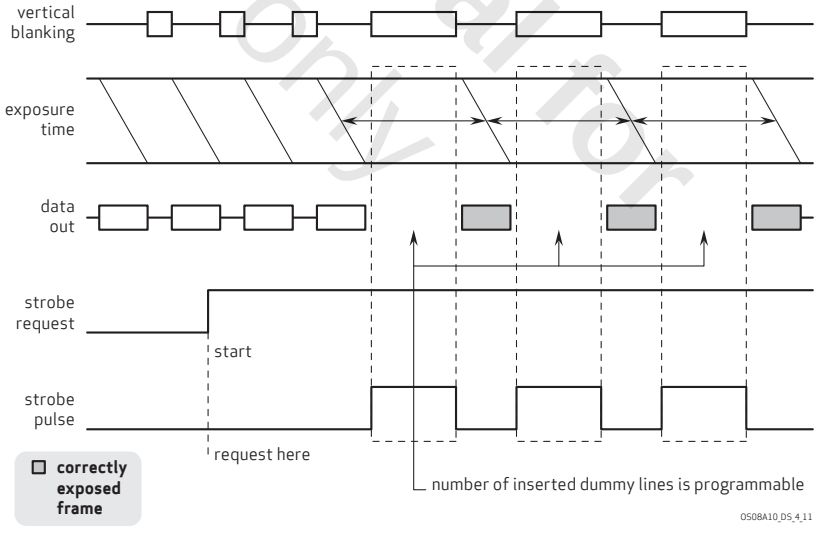


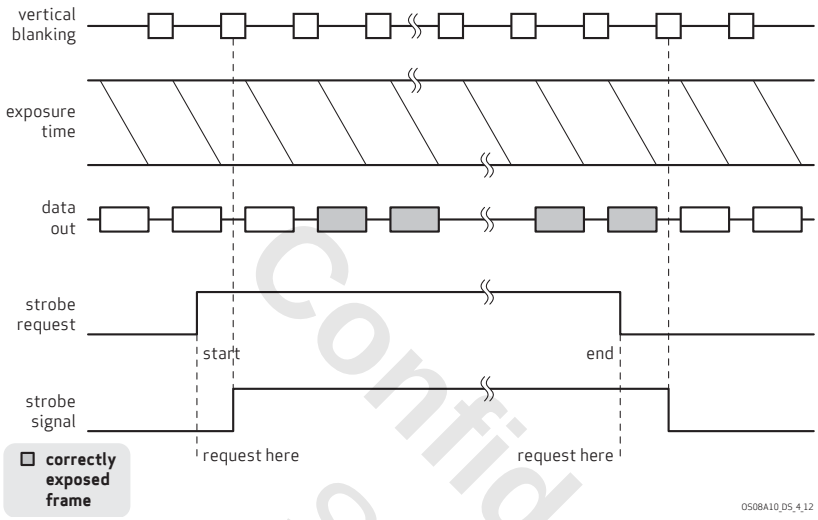
figure 4-10 LED 1 & 2 mode (part b)



4.6.1.3 LED 3 mode

In LED 3 mode, the strobe signal stays active until the strobe end request is sent (see **figure 4-11**).

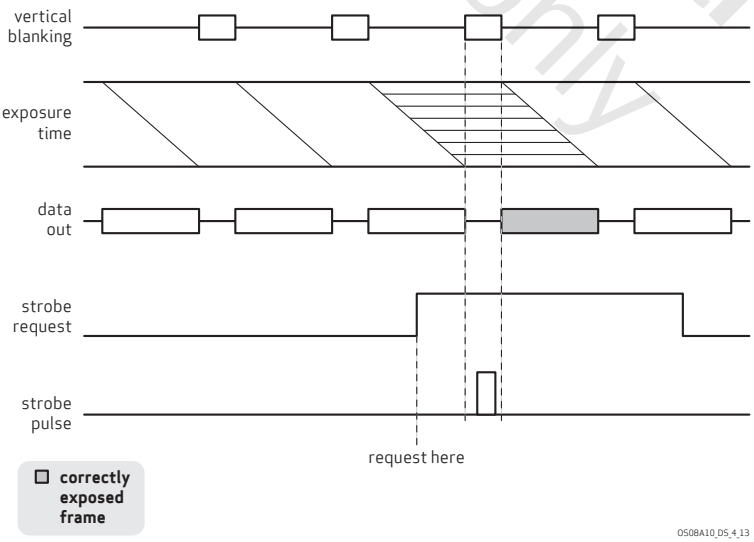
figure 4-11 LED 3 mode



4.6.1.4 LED 4 mode

In LED 4 mode, the strobe signal width is controlled by register 0x3B05 (see **figure 4-12**). Strobe width = 128 × (2^0x3B05[1:0]) × (0x3B05[7:2] + 1) × sclk_period. The maximum value of 0x3B05[7:2] is 6'b111110.

figure 4-12 LED 4 mode



See **table 4-6** for strobe control functions.

table 4-6 strobe control registers

address	register name	default value	R/W	description
0x3B00	STROBE CTRL	0x00	RW	Bit[7]: Strobe on/off Bit[6]: Strobe polarity 0: Active high 1: Active low Bit[5:4]: width_in_xenon Bit[2:0]: Strobe mode 000: Xenon 001: LED1 010: LED2 011: LED3 100: LED4
0x3B02	STROBE H	0x00	RW	Dummy Lines Added at Strobe Mode, MSB
0x3B03	STROBE L	0x00	RW	Dummy Lines Added at Strobe Mode, LSB
0x3B04	STROBE CTRL	0x00	RW	Bit[3]: start_point_sel Bit[2]: Strobe repeat enable Bit[1:0]: Strobe latency 00: Strobe generated at next frame 01: Strobe generated 2 frames later 10: Strobe generated 3 frames later 11: Strobe generated 4 frames later
0x3B05	STROBE WIDTH	0x00	RW	Bit[7:2]: Strobe pulse width step Bit[1:0]: Strobe pulse width gain strobe_pulse_width = 128×(2^gain)×(step+1)×Tsclk

4.7 embedded line

In the OS08A10, the user can configure the registers being output in embedded line. One or more rows of embedded data can be output at the beginning or end of image frame. The embedded line number is controlled by register 0x3216[3:0]. In total, embedded line can support up to 512 bytes of data.

4.7.1 embedded line setup example

For example: output data from register addresses 0x3800~0x3802 and 0x4800~0x4801:

```
6C  3665  AB;   [7] emb_en, [6] emb_lsb, [5:0] emb_dt
6C  3216  02;   [3:0] emb_line_number
6C  3208  04;   [3:0] embedded line record start using group hold function
6C  3500  10;   record register 0x3500~0x350F, total of sixteen registers
6C  3700  10;   record register 0x3700~0x370F, total of sixteen registers
6C  3740  10;   record register 0x3740~0x374F, total of sixteen registers
6C  3208  14;   embedded line record end
6C  3218  02;   emb_line at sof
6C  3660  42;   vfifo sof rip disable
```

4.7.2 embedded line output

Embedded line is output in front of image line. Valid embedded line content width is programmable. Invalid embedded line content is fixed and filled with dummy data from one programmable byte. Embedded line supports tag and no tags. Tag value is programmable.

table 4-7 embedded data control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3216	EMB_LINE_NUM	0x00	RW	Bit[7:0]: Embedded line number
0x3217	EMB_PADDING	0x00	RW	Bit[7:0]: Padding data value

table 4-7 embedded data control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3218	EMB_CTRL	0x00	RW	Bit[7:6]: emb_line_blk_ctrl 00: 32 sclk cycles 01: 64 sclk cycles 10: 128 sclk cycles 11: Wait for external trigger signal Bit[5]: embline_addr_en Bit[4]: r_padding_md2 0: No dummy data mixing with valid data 1: Dummy data output with valid data for every cycle Bit[3]: frame_trig_sel 0: tc_grp_wr 1: EOF Bit[2]: embline_eof_en Bit[1]: embline_sof_en Bit[0]: embline_tag_en
0x3219	EMB_TAG	0x00	RW	Bit[7:0]: emb_tag
0x3665	CORE_REG	0x01	RW	Bit[7]: Embedded lines enable at system level Bit[6]: Embedded line in low byte or high byte

4.8 light sensing mode (LSM)

The OS08A10 provides a new operating mode called light sensing mode (LSM). While it features extremely low power consumption, its functionality is very different from conventional video streaming modes. In LSM mode, all digital functions are shut down. The image sensor serves as an ambient light sensor. Therefore, there is no video streaming output.

The following are characteristics of LSM:

- no input clock is required
- no video streaming output (light level is stored in register)
- ambient light detection and generate interrupt (through GPIO pin) to wake up the system
- light level sampling rate is 30 Hz

4.8.1 entering/exiting LSM mode

Entering LSM mode:

1. Pull XSHUTDOWN2 pin to low.
2. Configure sensor setting as mode 1 configuration or mode 2 configuration.
3. Wait for interrupt generation.
4. Host reads register 0x2021 (MSB) and 0x2022 (LSB) for light sensing condition to decide the next step

Exiting LSM mode:

1. Pull XSHUTDOWN2 pin to high
2. Re-initialize sensor setting to enter ULPM mode or normal mode for video streaming

figure 4-13 shows an example flow chart for unlocking a device using LSM mode.

figure 4-13 low power mode example

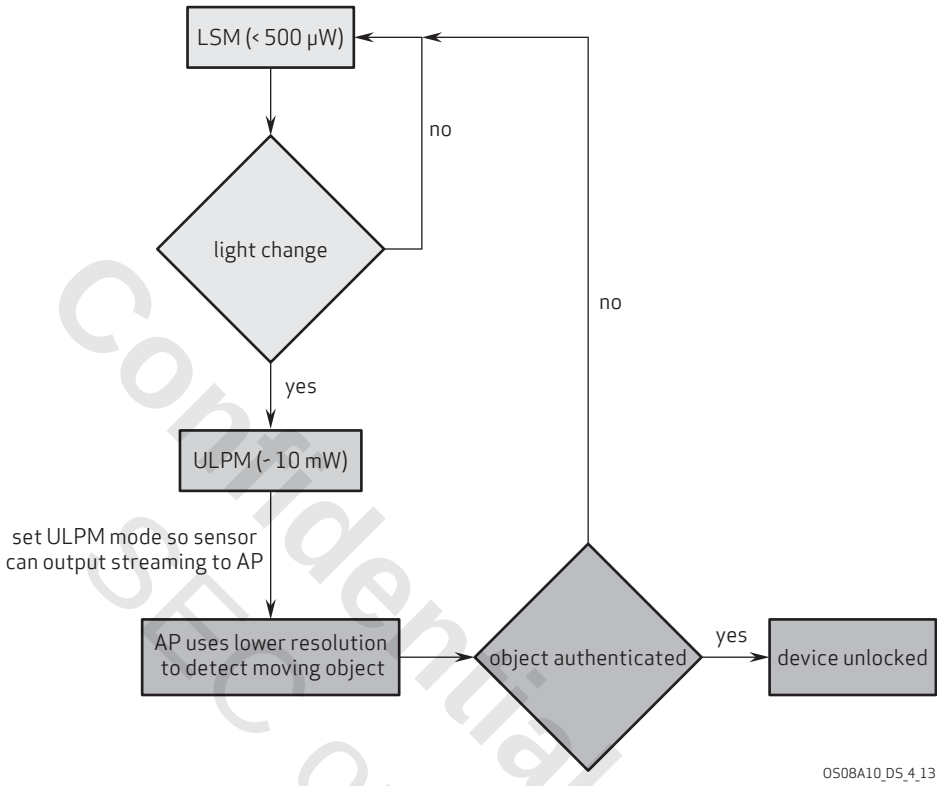
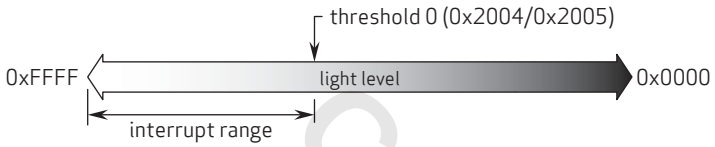


figure 4-14 LSM interrupt mode 1 configuration

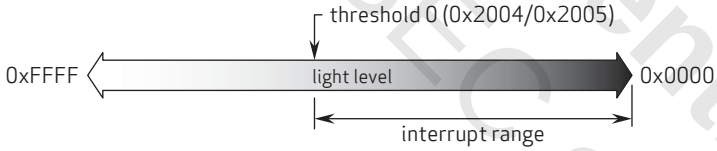
The configuration below is used to generate interrupt when the light level is above the threshold.

7c 2000 01 ; enable LSM mode
7c 2004 06 ; interrupt threshold high byte
7c 2005 00 ; interrupt threshold low byte
7c 200e 86 ; LSM control
7c 2010 9e ; interrupt polarity control
;;; read register 0x2021 (MSB) and 0x2022 (LSB) for light sensing condition



The configuration below is used to generate interrupt when the light level is below the threshold.

7c 2000 01 ; enable LSM Mode
7c 2004 06 ; interrupt threshold high byte
7c 2005 00 ; interrupt threshold low byte
7c 200e 86 ; LSM control
7c 2010 96 ; interrupt polarity control
;;; read register 0x2021 (MSB) and 0x2022 (LSB) for light sensing condition



OS08A10_DS_4_14

figure 4-15 LSM interrupt mode 2 configuration

The configuration below is used to generate interrupt if the light level is above threshold 1 or between the light zone boundary and threshold 0.

The interrupt range ceiling is fixed as 0xFFFF and "threshold 1" > "light zone boundary" > " threshold 0".

```
7c 2000 01 ; enable LSM mode
7c 2006 a0 ; interrupt threshold 1 high byte
7c 2007 00 ; interrupt threshold 1 low byte
7c 200a 80 ; light zone boundary S high byte
7c 200b 00 ; light zone boundary S low byte
7c 200c 80 ; light zone boundary L high byte
7c 200d 00 ; light zone boundary L low byte
7c 2004 06 ; interrupt threshold 0 high byte
7c 2005 00 ; interrupt threshold 0 low byte

7c 200e 86 ; LSM control
7c 2010 9e ; interrupt polarity control
;;; read register 0x2021 (MSB) and 0x2022 (LSB) for light sensing condition
```

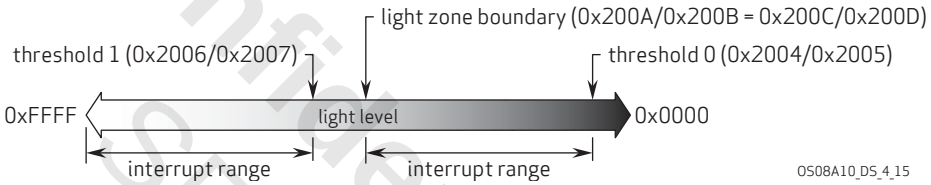


table 4-8 LSM control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x2000	LSM CTRL00	0x20	RW	Bit[0]: LSM enable 0: Disable LSM 1: Enable LSM
0x2001	LSM CTRL01	0x01	RW	Bit[7]: clk_sel_o Bit[6:0]: frame_interval Minimum number of frames between two LSM interrupt signals
0x2004	THRESHOLD0 HIGH	0x40	RW	Bit[7:0]: Threshold0[15:8]
0x2005	THRESHOLD0 LOW	0x00	RW	Bit[7:0]: Threshold0[7:0]
0x2006	THRESHOLD1 HIGH	0x00	RW	Bit[7:0]: Threshold1[15:8]
0x2007	THRESHOLD1 LOW	0x00	RW	Bit[7:0]: Threshold1[7:0]
0x200A	LM SMALL0	0xFF	RW	Bit[7:0]: lm_small[15:8] Small zone light level boundary
0x200B	LM SMALL1	0xFF	RW	Bit[7:0]: lm_small[7:0] Small zone light level boundary

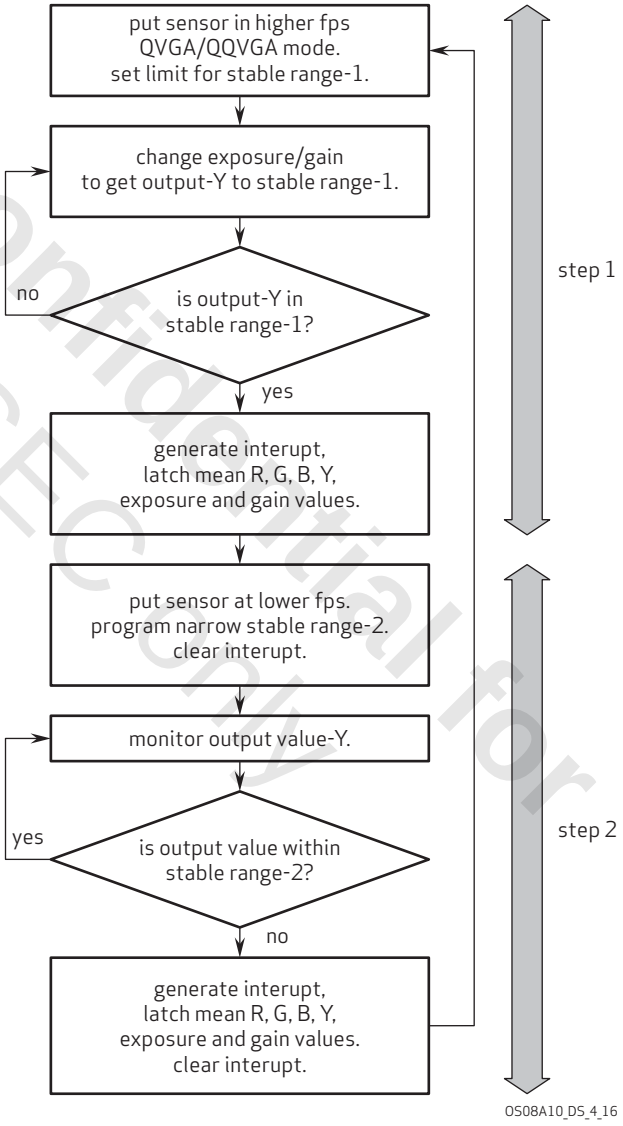
table 4-8 LSM control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x200C	LM LARGE0	0xFF	RW	Bit[7:0]: lm_large[15:8] Small zone light level boundary
0x200D	LM LARGE1	0xFF	RW	Bit[7:0]: lm_large[7:0] Small zone light level boundary
0x2010	LSM CTRL10	0x9A	RW	Bit[7]: Interrupt enable Bit[3]: Interrupt polarity select 0: Active low 1: Active high
0x2013	LSM CTRL13	0x00	RW	Bit[7:6]: Interrupt calculation mode select 00: Compare light level count of current frame 01: Compare absolute difference against previous frame 10: Combine mode0 and mode1 11: Compare change percentage against previous frame Bit[5:4]: lsm_output_sel 00: Level interrupt (last one frame (65536 cycles)) 01: Debug mode 10: Pulse interrupt 11: Debug mode Bit[3:0]: Debug mode
0x2016	LSM INTR CLEAR	–	W	Bit[7:1]: Debug mode Bit[0]: LSM interrupt clear
0x2021	RO LM1 COUNT0	–	R	Bit[7:0]: lm1_count[15:8] Light meter readout value
0x2022	RO LM1 COUNT1	–	R	Bit[7:0]: lm1_count[7:0] Light meter readout value

4.9 ULPM interrupt operations procedures

The OS08A10 supports an ULPM interrupt function that can output the current frame's Y (or R/G/B channel) average and generate a hardware interrupt when sensor luminance level is within or out of pre-set target range. **figure 4-16** shows a description of the ULPM interrupt operations.

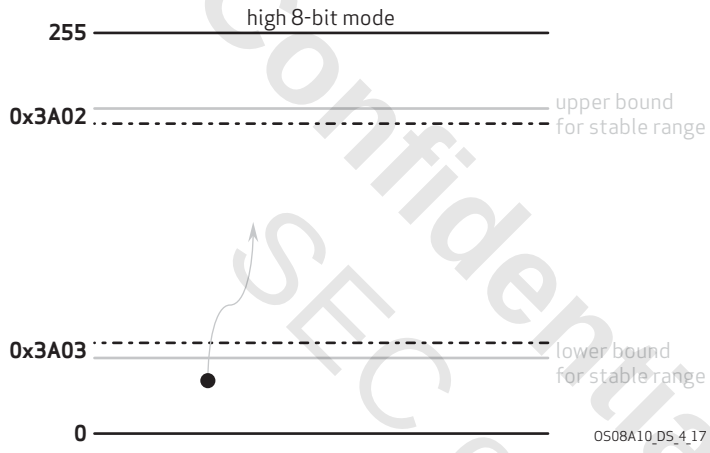
figure 4-16 procedure description



4.9.1 step 1 (detecting Y value)

1. Sensor is powered up in sleep condition.
2. Host programs a sequence of SCCB addresses to get the sensor operational in the high frame rate QVGA/QQVGA mode. The following registers will need to be written: QVGA/QQVGA setting, exposure/gain settings, window, and stable range.
3. Sensor gain and exposure values change to get to the stable range defined above.
4. As soon as the sensor achieves stable range, it triggers an output on ULPM pin and 0x4A20[0]. The values of gain, exposure, mean R, G, B, Y, are latched onto separate registers (0x4A21 ~ 0x4A2B).
5. Host is interrupted by this trigger and takes this as a signal to go and read the values in the above registers.
6. Host needs to clear the trigger (rewrite 0x4A10 = 0) to go to step 2.

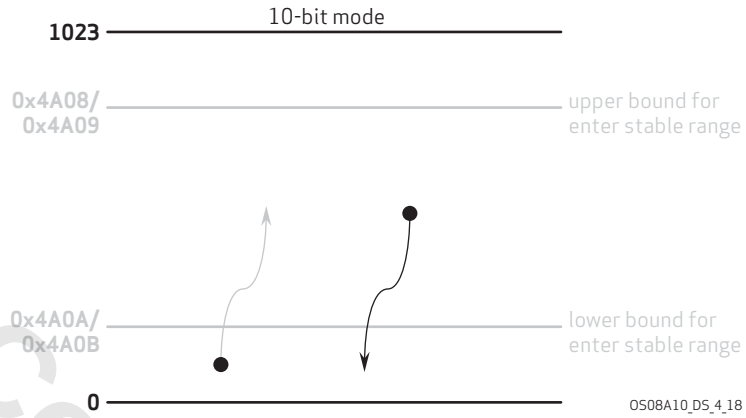
figure 4-17 stable range-1 diagram



4.9.2 step 2 (detecting any change in Y value from an earlier state)

1. Host programs a sequence of SCCB addresses to get the sensor into a lower fps mode. The host will need to use the gain and exposure values latched in the previous step 1.
2. Host programs a new stable range-2 (0x4A08~0x4A0B) around the value that the sensor had reached when it gave a trigger to the host in step 1.
3. If the Y output of the sensor goes outside its stable range programmed above in step 2, it will generate another interrupt on the same pin (ULPM) and 0x4A20[1] latched the gain, exposure, mean (R, G, B, Y) values on registers (0x4A21~0x4A2B).
4. Host needs to clear the interrupt (rewrite 0x4A10=0) to come back to step 1.
5. Host needs to program the sensor back to the step 1 to get the new Y value as per the procedure in step 1.

figure 4-18 stable range-2 diagram



4.10 stable range/window registers for ULPM interrupt

table 4-9 stable range/window registers

address	register name	default value	R/W	description
0x3A02	EXP_CTRL02	0x78	RW	Bit[7:0]: Top stable range-1
0x3A03	EXP_CTRL03	0x68	RW	Bit[7:0]: Bottom stable range-1
0x4B04	AVG_CTRL04	0x01	RW	Bit[0]: window_width_avg[8]
0x4B05	AVG_CTRL05	0x40	RW	Bit[7:0]: window_width_avg[7:0]
0x4B07	AVG_CTRL07	0xF0	RW	Bit[7:0]: window_height_avg[7:0]
0x4B10	AVG_CTRL10	0x12	RW	Bit[0]: average_man_en

4.11 ULPM interrupt

table 4-10 ULPM interrupt registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4A00	ULPM_CTRL00	0xA4	RW	Bit[7]: Debug mode (always set to 1'b0) Bit[6]: Use average method to do ULPM 1: Average mode Bit[5]: r_opt Bit[3]: r_ratio_man_en Bit[1]: r_pol_inv_en Bit[0]: r_abs_en 0: Use other method 1: Use absolute value to make decision
0x4A01	ULPM_CTRL01	0x0A	RW	Bit[3]: gpio_auto_clear_en Bit[2]: gpio_host_clear_en Bit[1]: reg_auto_clear_en Bit[0]: reg_host_clear_en
0x4A04	ULPM_CTRL04	0x00	RW	Bit[2:0]: r_diff_ratio0
0x4A05	ULPM_CTRL05	0x00	RW	Bit[7:4]: r_diff_ratio1 Bit[3:0]: r_diff_ratio2
0x4A06	ULPM_CTRL06	0x04	RW	Bit[3:0]: Trigger length
0x4A07	ULPM_CTRL07	0x10	RW	Bit[5:0]: r_no_intr_threshold
0x4A08	ULPM_CTRL08	0xF0	RW	Bit[7:0]: r_threshold_high[15:8]
0x4A09	ULPM_CTRL09	0x00	RW	Bit[7:0]: r_threshold_high[7:0]
0x4A0A	ULPM_CTRL0A	0x00	RW	Bit[7:0]: r_threshold_low[15:8]
0x4A0B	ULPM_CTRL0B	0x64	RW	Bit[7:0]: r_threshold_low[7:0]
0x4A0C	ULPM_CTRL0C	0x80	RW	Bit[7:0]: r_ratio_manual[15:8]
0x4A0D	ULPM_CTRL0D	0x00	RW	Bit[7:0]: r_ratio_manual[7:0]
0x4A10	ULPM_CTRL10	–	W	Bit[1]: Host register clear This bit is always kept high until it is written to 0. After writing to zero operation is finished, this bit will return to high state. 0: reg_host_clear Bit[0]: Host GPIO clear 0: gpio_host_clear
0x4A20	ULPM_CTRL20	–	R	Bit[2]: Interrupt flag combination of interrupt 1 and interrupt 2 flags Bit[1]: Interrupt 2 flag Bit[0]: Interrupt 1 flag

table 4-10 ULPM interrupt registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4A21	ULPM_CTRL21	–	R	Bit[7:0]: Current gain value
0x4A22	ULPM_CTRL22	–	R	Bit[7:0]: Current exposure[15:8]
0x4A23	ULPM_CTRL23	–	R	Bit[7:0]: Current exposure[7:0]
0x4A24	ULPM_CTRL24	–	R	Bit[1:0]: avg_y[9:8]
0x4A25	ULPM_CTRL25	–	R	Bit[7:0]: avg_y[7:0]
0x4A26	ULPM_CTRL26	–	R	Bit[1:0]: avg_r[9:8]
0x4A27	ULPM_CTRL27	–	R	Bit[7:0]: avg_r[7:0]
0x4A28	ULPM_CTRL28	–	R	Bit[1:0]: avg_g[9:8]
0x4A29	ULPM_CTRL29	–	R	Bit[7:0]: avg_g[7:0]
0x4A2A	ULPM_CTRL2A	–	R	Bit[1:0]: avg_b[9:8]
0x4A2B	ULPM_CTRL2B	–	R	Bit[7:0]: avg_b[7:0]

4.11.1 ULPM interrupt example

1. Send higher fps QQVGA sensor setting
2. Setup stable range, window setting

```
6c 0100 00; Disabling streaming

6c 3a02 78; stable range-1 upper bound in high 8-bit format

6c 3a03 68; stable range-1 lower bound in high 8-bit format

6c 4b04 00; avg window width

6c 4b05 a0; avg window width

6c 4b07 78; avg window height

6c 4b10 13; manual avg window
```

3. Set up ULPM interrupt setting

```
6c 3016 00; disable ULPM controlled by register

6c 4a00 64; ULPM average mode

6c 4a01 05; disable auto clearing flag
```

4. Monitor ULPM interrupt flag via register 0x4A20 or ULPM pin.
5. Enable streaming with the setting shown below. Once AEC/AGC adjusts Y value reaching step 1 stable range, ULPM interrupt can be observed at 0x4A20 (=0x05) and ULPM pin.

```
6c 0100 01
```

6. Reduce frame rate for step 2, if desired, and set the stable range for step 2.

```
6c 380e 20; reduce fps

6c 4a08 01; stable range-2 upper bound in 10-bit format

6c 4a09 e0; stable range-2 upper bound in 10-bit format

6c 4a0a 01; stable range-2 lower bound in 10-bit format

6c 4a0b a0; stable range-2 lower bound in 10-bit format
```

7. Clear flag by writing 0x00 to register 0x4A10, and then monitor register 0x4A20 for step 2 flag. Once image Y value is outside stable range-2, ULPM interrupt can be observed at 0x4A20 (=0x06) and ULPM pin.

```
6c 4a10 00
```

8. After step 2 interrupt, frame rate can be adjusted by clearing step 2 flag by writing 0x00 to register 0x4A10, and then go back to step 1 again.

4.12 high dynamic range (HDR)

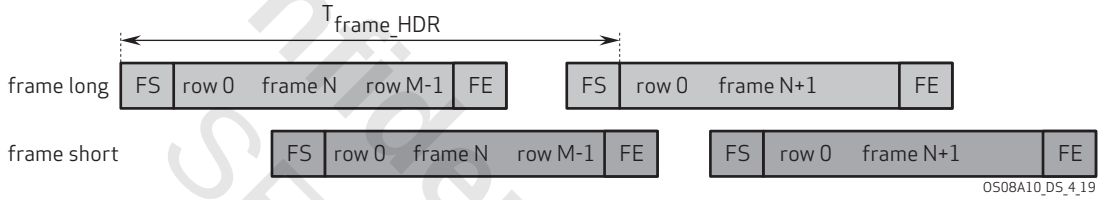
HDR mode increases image dynamic range by capturing multiple exposures of a similar scene and then combining them into one single image. The OS08A10 supports staggered HDR mode. Sequential HDR can be achieved by using context switching.

4.12.1 staggered HDR

In staggered HDR mode, long/short exposure frames overlap each other. Data is read line by line alternatively between long exposure and short exposure frames. The overlapping reduces the timing delay between various exposure frames that combine to form one HDR frame. The overlapping also reduces the frame/line buffer needed for backend sensors. The frame timing for staggered HDR is shown in **figure 4-19**.

In this mode, the max exposure line number: T_long + T_short < frame_length (VTS) - 4

figure 4-19 set staggered HDR

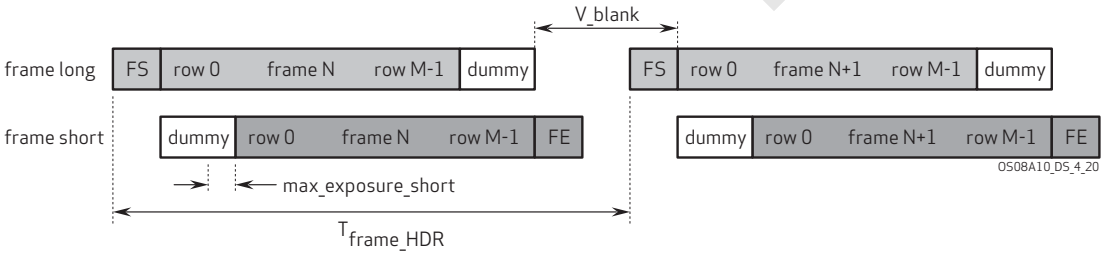


4.12.2 staggered HDR output modes

The first staggered mode uses a MIPI channel to differentiate different exposure frames. Long/short frames use MIPI virtual channel VC0/VC1, respectively, in order for different exposure frames to not mix signals on the MIPI receiver side.

The second output mode is the new staggered HDR mode with dummy lines, where no VC is needed, and only on FS and on FE for each new staggered HDR frame, supported by MIPI and LVDS.

figure 4-20 set staggered HDR with no VC



The third output mode is the LVDS axis mode. LVDS axis mode uses two independent LVDS, long exposure transmits on lane0/1, short exposure transmits on lane2/3. LVDS axis mode supports two 2-lane modes and two 1-lane modes.

figure 4-21 LVDS axis mode

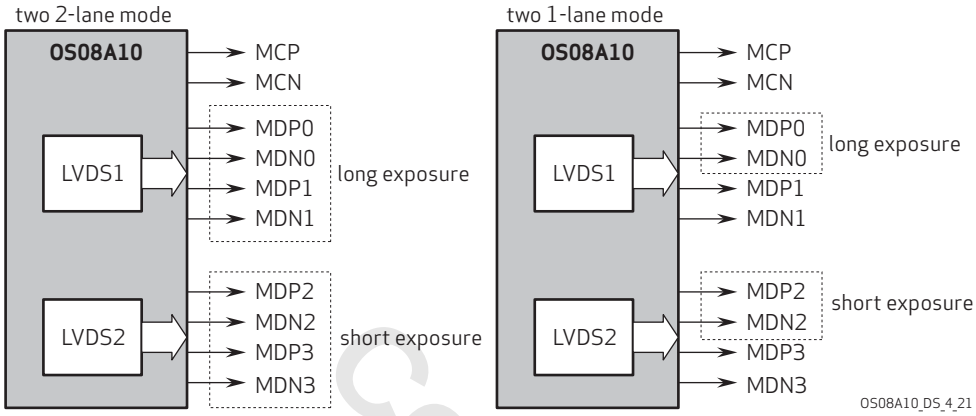


table 4-11 staggered HDR control register (sheet 1 of 2)

address	register name	default value	R/W	description
0x3501	LONG EXPO	0x00	RW	Bit[7:0]: Long exposure[15:8]
0x3502	LONG EXPO	0x40	RW	Bit[7:0]: Long exposure[7:0]
0x3508	LONG GAIN	0x00	RW	Bit[5:0]: Gain[13:8]
0x3509	LONG GAIN	0x80	RW	Bit[7:0]: Long gain[7:0] If 0x3503[2] = 0, then Gain[12:0] is real gain format, where low 7 bits are fraction bits. real_gain = Gain[12:0]/128 For example, 0x080 is 1x gain, 0x140 is 2.5x gain. Maximum gain is 15.5x; 0x7C0. If 0x3503[2]=1, then Gain[12:0] is sensor gain format, where: Gain[12:8]: Coarse gain 00000: 1x 00001: 2x 00011: 4x 00111: 8x Gain[7]: 1'b1 Gain[6:3]: Fine gain Gain[2:0]: Always 0 For example, 0x080 is 1x gain, 0x180 is 2x gain, and 0x380 is 4x gain.
0x350C	SHORT GAIN	0x00	RW	Bit[5:0]: Short gain[13:8]
0x350D	SHORT GAIN	0x80	RW	Bit[7:0]: Short gain[7:0]
0x3511	SHORT EXPO	0x00	RW	Bit[7:0]: Short exposure[15:8]

table 4-11 staggered HDR control register (sheet 2 of 2)

address	register name	default value	R/W	description
0x3512	SHORT EXPO	0x20	RW	Bit[7:0]: Short exposure[7:0]
0x3661	CORE 1	0x06	RW	Bit[0]: stg_hdr_align_en
0x381C	BLC_NUM_ OPTION	0x0E	RW	Bit[1]: new_stg_hdr_en
0x3821	FORMAT2	0x00	RW	Bit[5]: Stagger HDR enable

5 image sensor processor digital functions

5.1 ISP top

The main purpose of the ISP top includes:

- integrate all sub-modules
- create necessary control signals

table 5-1 ISP top registers

address	register name	default value	R/W	description
0x5000	ISP CTRL 0	0xA9	RW	Bit[7]: awb_en Bit[3]: otp_en Bit[0]: isp_en
0x5001	ISP CTRL 1	0x09	RW	Bit[6]: win_en Bit[1]: dpc_en

5.2 DSP

The main purposes of the pre_DSP module include:

- adjust HREF, valid, RBlue signals and data
- create color bar image
- determine the sizes of input images by removing redundant data
- create control signals

table 5-2 pre_DSP register

address	register name	default value	R/W	description
0x5081	ISP CTRL 1	0x04	RW	Bit[7]: test_en Bit[6]: rolling Bit[5]: trans Bit[4]: squ_bw Bit[3:2]: bar_style Bit[1:0]: test_sel

5.3 defective pixel cancellation (DPC)

The main purpose of the DPC function is to remove white/black defect pixels. If the pixel is defective, DPC will use a value calculated from the neighboring normal pixels to replace it.

table 5-3 DPC control register

address	register name	default value	R/W	description
0x5200	ISP_CTRL_0	0x1B	RW	Bit[7]: Enable tail Bit[6]: Enable tailing cluster Bit[5]: Enable 3x3 cluster Bit[4]: Enable saturate cross cluster Bit[3]: Enable cross cluster Bit[2]: Enable manual mode Bit[1]: Enable black pixel Bit[0]: Enable white enable

5.4 window cut (WINC)

The main purpose of the WINC module is to make the image size to be real size by removing offset.

table 5-4 WINC registers

address	register name	default value	R/W	description
0x5800	ISP_CTRL_0	0x00	RW	Bit[3:0]: Window X start[11:8]
0x5801	ISP_CTRL_1	0x00	RW	Bit[7:0]: Window X start[7:0]
0x5802	ISP_CTRL_2	0x00	RW	Bit[3:0]: Window Y start[11:8]
0x5803	ISP_CTRL_3	0x00	RW	Bit[7:0]: Window Y start[7:0]
0x5804	ISP_CTRL_4	0x02	RW	Bit[3:0]: Window width[11:8]
0x5805	ISP_CTRL_5	0x80	RW	Bit[7:0]: Window width[7:0]
0x5806	ISP_CTRL_6	0x01	RW	Bit[3:0]: Window height[11:8]
0x5807	ISP_CTRL_7	0xE0	RW	Bit[7:0]: Window height[7:0]
0x5808	ISP_CTRL_8	0x00	RW	Bit[7:4]: emb_num Bit[2]: emb_flag_sel 0: Start line 1: End line Bit[0]: win_man_en 0: Window size from window top 1: Window size from register

5.5 manual exposure compensation/ manual gain compensation (MEC/MGC)

Manual exposure provides exposure time settings and sensor gain. Manual gain provides analog gain settings. For optimal performance, maximum exposure should be 200 ms. For more details, contact your local OmniVision FAE.

table 5-5 MEC/MGC control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3501	LONG EXPO	0x00	RW	Bit[7:0]: Long exposure[15:8]
0x3502	LONG EXPO	0x40	RW	Bit[7:0]: Long exposure[7:0]
0x3503	AEC MANUAL	0x88	RW	Bit[7]: Not used Bit[6]: Digital fraction gain delay option 0: Delay 1 frame 1: Do not delay 1 frame Bit[5]: Not used Bit[4]: Gain delay option 0: Delay 1 frame 1: Do not delay 1 frame Bit[3]: Not used Bit[2]: Gain manual as sensor gain 0: Input gain as real gain format 1: Input gain as sensor gain format Bit[1]: Exposure delay option (must be 0) 0: Delay 1 frame 1: Not used Bit[0]: Exposure change delay option (must be 0) 0: Delay 1 frame 1: Not used
0x3505	GCVT OPTION	0x80	RW	Gain Conversation Option Bit[7]: DAC fixed gain bit Bit[6:4]: Sensor gain fixed bit Bit[3:2]: Sensor gain pregain option (debug only, always set it to 0) Bit[1:0]: Sensor gain option for transferring real gain to sensor gain format
0x3507	GAIN SHIFT	0x08	RW	Bit[3]: fine_snr_gain_16_en Back gain enable Bit[2]: Priority_6 switch_snr_gain_en 0: dac[2:0]_comp[1:0] 1: comp[1:0]_dac[2:0] Bit[1:0]: Gain shift option 00: Do not shift 01: Left shift 1 bit 10: Left shift 2 bits 11: Left shift 3 bits

note For optimal performance, minimum exposure should be 8, maximum exposure should be VTS - 8. For more details, contact your local OmniVision FAE.

table 5-5 MEC/MGC control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3508	LONG GAIN	0x00	RW	Bit[5:0]: Gain[13:8]
0x3509	LONG GAIN	0x80	RW	Bit[7:0]: Long gain[7:0] If 0x3503[2] = 0, then Gain[12:0] is real gain format, where low 7 bits are fraction bits. real_gain = Gain[12:0]/128 For example, 0x080 is 1x gain, 0x140 is 2.5x gain. Maximum gain is 15.5x; 0x7C0. If 0x3503[2]=1, then Gain[12:0] is sensor gain format, where: Gain[12:8]: Coarse gain 00000: 1x 00001: 2x 00011: 4x 00111: 8x Gain[7]: 1'b1 Gain[6:3]: Fine gain Gain[2:0]: Always 0 For example, 0x080 is 1x gain, 0x180 is 2x gain, and 0x380 is 4x gain.
0x350A	LONG DIGIGAIN	0x04	RW	Bit[5:0]: Long digital gain[13:8] 4.10 format
0x350B	LONG DIGIGAIN	0x00	RW	Bit[7:0]: Long digital gain[7:0] 4.10 format
0x350C	SHORT GAIN	0x00	RW	Bit[5:0]: Short gain[13:8]
0x350D	SHORT GAIN	0x80	RW	Bit[7:0]: Short gain[7:0]
0x350E	SHORT GAIN	0x04	RW	Bit[5:0]: Short digital gain[13:8]
0x350F	SHORT GAIN	0x00	RW	Bit[7:0]: Short digital gain[7:0]
0x3511	SHORT EXPO	0x00	RW	Bit[7:0]: Short exposure[15:8]
0x3512	SHORT EXPO	0x20	RW	Bit[7:0]: Short exposure[7:0]
0x3513	SNR_GAIN_L	–	R	Bit[5:0]: Long sensor gain[13:8]
0x3514	SNR_GAIN_L	–	R	Bit[7:0]: Long sensor gain[7:0]
0x3515	FINE_SNR_GAIN_L	–	R	Bit[5:0]: Long fine sensor gain
0x3516	SNR_GAIN_S	–	R	Bit[5:0]: Short sensor gain[13:8]
0x3517	SNR_GAIN_S	–	R	Bit[7:0]: Short sensor gain[7:0]
0x3518	FINE_SNR_GAIN_S	–	R	Bit[5:0]: Short fine sensor gain

table 5-5 MEC/MGC control registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x3519	GAIN_DELAY_ SWAP_OPTION	0x00	RW	Bit[2]: Debug mode Bit[1]: digital_gain apply select 0: Digital gain apply in BLC 1: Digital gain apply in MWB Bit[0]: Auto gain delay option 0: Real gain and digital gain delay manual mode 1: Real gain and digital gain delay will auto sync with expo. When expo and gain change at same time, both gain will delay 1 frame to sync with expo; otherwise, gain will have no frame delay
0x351A	BLC_DIG_GAIN_ CUR_L	–	R	Bit[5:0]: Current BLC long digital gain[13:8]
0x351B	BLC_DIG_GAIN_ CUR_L	–	R	Bit[7:0]: Current BLC long digital gain[7:0]
0x351C	BLC_DIG_GAIN_ CUR_S	–	R	Bit[5:0]: Current BLC short digital gain[13:8]
0x351D	BLC_DIG_GAIN_ CUR_S	–	R	Bit[7:0]: Current BLC short digital gain[7:0]
0x351E	MWB_DIG_CUR_L	–	R	Bit[5:0]: Current MWB long digital gain[13:8]
0x351F	MWB_DIG_CUR_L	–	R	Bit[7:0]: Current MWB long digital gain[7:0]
0x3520	MWB_DIG_CUR_S	–	R	Bit[5:0]: Current MWB long digital gain[13:8]
0x3521	MWB_DIG_CUR_S	–	R	Bit[7:0]: Current MWB long digital gain[7:0]
0x3522	REAL_GAIN_L_BLC	–	R	Bit[1:0]: Current long BLC gain[9:8]
0x3523	REAL_GAIN_L_BLC	–	R	Bit[7:0]: Current long BLC gain[7:0]
0x3524	REAL_GAIN_S_BLC	–	R	Bit[1:0]: Current short BLC gain[9:8]
0x3525	REAL_GAIN_S_BLC	–	R	Bit[7:0]: Current short BLC gain[7:0]
0x3526	REAL_GAIN_L_ISP	–	R	Bit[3:0]: Current long ISP gain[11:8]
0x3527	REAL_GAIN_L_ISP	–	R	Bit[7:0]: Current long ISP gain[7:0]
0x3528	REAL_GAIN_S_ISP	–	R	Bit[3:0]: Current short ISP gain[11:8]
0x3529	REAL_GAIN_S_ISP	–	R	Bit[7:0]: Current short ISP gain[7:0]

5.6 MWB gain

The RAW R/G/B values of a gray object vary with the spectrum of the illumination and the sensor spectral response. The illumination spectrum is usually described by "color temperature", which is the surface temperature of a black body radiating equivalent spectrum. In the real world, the light color temperature ranges from very low (reddish) to very high (bluish) value. For example, the color temperature of an incandescent lamp is around 2850K, while the color temperature of an overcast day is around 6500K.

To make sure that a gray image is truly gray, the sensor needs to adjust the gain for each color channel according to the color temperature. The process is called white balance (WB).

White balanced gain is enabled by default and can be disabled by register 0x5000[7]. The manual white balance (MWB), controlled by registers 0x5100~0x5105, provides gain for R, G, and B channels. Each channel gain is 12-bit. 0x400 is 1x gain. Maximum 4x gain is 0xFFF.

White balance gain does not have internal latch function. Once it is set, it will take effect immediately.

table 5-6 MWB gain registers

address	register name	default value	R/W	description
0x5100	ISP CTRL 0	0x04	RW	Bit[3:0]: awb_gain_b[11:8]
0x5101	ISP CTRL 1	0x00	RW	Bit[7:0]: awb_gain_b[7:0]
0x5102	ISP CTRL 2	0x04	RW	Bit[3:0]: awb_gain_g[11:8]
0x5103	ISP CTRL 3	0x00	RW	Bit[7:0]: awb_gain_g[7:0]
0x5104	ISP CTRL 4	0x04	RW	Bit[3:0]: awb_gain_r[11:8]
0x5105	ISP CTRL 5	0x00	RW	Bit[7:0]: awb_gain_r[7:0]
0x5106	ISP CTRL 6	0x02	RW	Bit[5:0]: digigain_man[13:8]
0x5107	ISP CTRL 7	0x00	RW	Bit[7:0]: digigain_man[7:0]

6 register tables

The following tables provide descriptions of the device control registers contained in the OS08A10. For all register enable/disable bits, ENABLE = 1 and DISABLE = 0. The device slave addresses are 0x6C for write and 0x6D for read (when SID=1, 0x20 for write and 0x21 for read).

6.1 PLL control [0x0100, 0x0103, 0x0303 - 0x0309, 0x030C, 0x0322 - 0x032A]

table 6-1 PLL control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x0100	SC_CTRL0100	0x00	RW	Bit[7:1]: Debug mode Bit[0]: software_standby 0: software_standby 1: Streaming
0x0103	SC_CTRL0103	–	W	Bit[7:1]: Debug mode Bit[0]: software_reset
0x0303	PLL_CTRL_03	0x02	RW	Bit[7:3]: Not used Bit[2:0]: pll1_prediv
0x0304	PLL_CTRL_04	0x00	RW	Bit[7:2]: Not used Bit[1:0]: pll1_divp[9:8]
0x0305	PLL_CTRL_05	0x3C	RW	Bit[7:0]: pll1_divp[7:0]
0x0306	PLL_CTRL_06	0xH0	RW	Bit[7:2]: Not used Bit[1:0]: pll1_divmipi
0x0307	PLL_CTRL_07	0x00	RW	Bit[7:4]: Not used Bit[3:0]: pll1_divm
0x0308	PLL_CTRL_08	0x03	RW	Bit[7:4]: Not used Bit[3:0]: pll1_divsp
0x0309	PLL_CTRL_09	0x03	RW	Bit[7:3]: Not used Bit[2:0]: pll1_divs
0x030C	PLL_CTRL_0C	0x00	RW	Bit[7:1]: Not used Bit[0]: pll1_predivp
0x0322	PLL_CTRL_22	0x00	RW	Bit[7:1]: Not used Bit[0]: pll2_predivp
0x0323	PLL_CTRL_23	0x02	RW	Bit[7:3]: Not used Bit[2:0]: pll2_prediv
0x0324	PLL_CTRL_24	0x00	RW	Bit[7:2]: Not used Bit[1:0]: pll2_divp[9:8]
0x0325	PLL_CTRL_25	0x3C	RW	Bit[7:0]: pll2_divp[7:0]

table 6-1 PLL control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x0326	PLL_CTRL_26	0x00	RW	Bit[7:2]: Not used Bit[1:0]: pll2_reserve1
0x0327	PLL_CTRL_27	0x07	RW	Bit[7:4]: Not used Bit[3:0]: pll2_divsram
0x0328	PLL_CTRL_28	0x07	RW	Bit[7:4]: Not used Bit[3:0]: pll2_divst
0x0329	PLL_CTRL_29	0x01	RW	Bit[7:4]: Not used Bit[3:0]: pll2_divdac
0x032A	PLL_CTRL_2A	0x00	RW	Bit[7:3]: Not used Bit[2:0]: pll2_divt

6.2 LSM control [0x2000 - 0x2013, 0x2016 - 0x2018, 0x2020 - 0x2029]

The SCCB ID in LSM is 0x7C when SID is 0, and 0x30 when SID is 1.

table 6-2 LSM control registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x2000	LSM CTRL00	0x20	RW	Bit[7:6]: Debug mode Bit[5]: level_adj_enable Bit[4]: merge_avg Bit[3]: merge_en Bit[2]: Reserved Bit[1]: sel_abs_diff 0: Using absolute level of current frames 1: Using absolute level difference between current and previous frames Bit[0]: lsm_en
0x2001	LSM CTRL01	0x01	RW	Bit[7]: clk_sel_o Bit[6:0]: frame_interval Minimum number of frames between two LSM interrupt signals
0x2002	LSM CTRL02	0x00	RW	Bit[7:0]: thres_man
0x2003	LSM CTRL03	0x00	RW	Bit[7:0]: thres_man
0x2004	THRESHOLD0 HIGH	0x40	RW	Bit[7:0]: Threshold0[15:8]
0x2005	THRESHOLD0 LOW	0x00	RW	Bit[7:0]: Threshold0[7:0]

table 6-2 LSM control registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x2006	THRESHOLD1 HIGH	0x00	RW	Bit[7:0]: Threshold1[15:8]
0x2007	THRESHOLD1 LOW	0x00	RW	Bit[7:0]: Threshold1[7:0]
0x2008	LSM CTRL08	0x00	RW	Bit[7:1]: Not used Bit[0]: thres_man_en
0x2009	STEP NORM	0x00	RW	Bit[7:0]: step_norm
0x200A	LM SMALL0	0xFF	RW	Bit[7:0]: lm_small[15:8] Small zone light level boundary
0x200B	LM SMALL1	0xFF	RW	Bit[7:0]: lm_small[7:0] Small zone light level boundary
0x200C	LM LARGE0	0xFF	RW	Bit[7:0]: lm_large[15:8] Small zone light level boundary
0x200D	LM LARGE1	0xFF	RW	Bit[7:0]: lm_large[7:0] Small zone light level boundary
0x200E	ANA REG0	0x82	RW	Bit[7:0]: reg_ana[15:8]
0x200F	ANA REG1	0x10	RW	Bit[7:0]: reg_ana[7:0]
0x2010	LSM CTRL10	0x9A	RW	Bit[7]: Interrupt enable Bit[6:4]: wait_reading_en Bit[3]: Interrupt polarity 0: Active low 1: Active high Bit[2]: rdy_mask_auto_only Bit[1]: out_of_range_intr_en Bit[0]: host_clear_frm
0x2011	LSM CTRL11	0x02	RW	Bit[7]: rdy_opt Bit[6:0]: mask_period Mask first N frames after reset
0x2012	LSM CTRL12	0x01	RW	Bit[7:0]: intr_num

table 6-2 LSM control registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x2013	LSM CTRL13	0x00	RW	Bit[7:6]: Interrupt calculation mode select 00: Compare light level count of current frame 01: Compare absolute difference against previous frame 10: Combine mode0 and mode1 11: Compare change percentage against previous frame Bit[5:4]: lsm_output_sel 00: Level interrupt (last one frame (65536 cycles)) 01: out_range (debug) 10: Pulse interrupt 11: lm_in1_i (debug) Bit[3:0]: Debug mode
0x2016	LSM INTR CLEAR	–	W	Bit[7:1]: Debug mode Bit[0]: LSM interrupt clear
0x2017	LSM CTRL17	0xC4	RW	Bit[7:4]: level_adj_ht Higher threshold for automatic level adjustment Bit[3:0]: level_adj_lt Lower threshold for automatic level adjustment
0x2018	LSM CTRL18	0x00	RW	Bit[7:4]: Debug mode Bit[3:1]: lsm_pulse_length LSM pulse width = Nx16 clock cycles Bit[0]: als_pulse_sel 0: Detect negative edge of als_trigger 1: Detect positive edge of als_trigger
0x2020	RO LEVEL ADJ	–	R	Bit[7]: Not used Bit[6:2]: level_adj Bit[1]: lm_in2 Bit[0]: lm_in1
0x2021	RO LM1 COUNT0	–	R	Bit[7:0]: lm1_count[15:8]
0x2022	RO LM1 COUNT1	–	R	Bit[7:0]: lm1_count[7:0]
0x2022	RO LM2 COUNT0	–	R	Bit[7:0]: lm2_count[15:8]
0x2023	RO LM2 COUNT1	–	R	Bit[7:0]: lm2_count[7:0]
0x2024	RSVD	–	–	Reserved
0x2025	RO LM1 DIFF0	–	R	Bit[7:0]: lm1_diff[15:8]

table 6-2 LSM control registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x2026	RO LM1 DIFF1	–	R	Bit[7:0]: lm1_diff[7:0]
0x2027	RO LM2 DIFF0	–	R	Bit[7:0]: lm2_diff[15:8]
0x2028	RO LM2 DIFF1	–	R	Bit[7:0]: lm2_diff[7:0]
0x2029	RO LSM CTRL29	–	R	Bit[7:4]: Debug mode Bit[3]: out_range_i Bit[2]: lsm_trig_i Bit[1]: lsm_trig2_i Bit[0]: lsm_trig1_i

6.3 system control [0x3000 - 0x302A, 0x3031 - 0x3038, 0x303E - 0x303F]

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

address	register name	default value	R/W	description
0x3000	NOT USED	–	–	Not Used
0x3001	IO PAD OEN	0x83	RW	Bit[7:2]: Not used Bit[1]: gpio3_oen Bit[0]: gpio2_oen
0x3002	IO PAD OEN	0x00	RW	Bit[7]: io_vsync_oen Bit[6]: io_href_oen Bit[5:2]: Not used Bit[1]: io_gpio1_oen Bit[0]: io_gpio0_oen
0x3003	NOT USED	–	–	Not Used
0x3004	IO PAD OUT	0x04	RW	Bit[7:2]: Not used Bit[1]: io_gpio3_o Bit[0]: io_gpio2_o
0x3005	IO PAD OUT	0x00	RW	Bit[7]: io_vsync_o Bit[6]: io_href_o Bit[5]: Not used Bit[4]: io_strobe Bit[3]: Not used Bit[2]: io_sda Bit[1]: io_gpio1_o Bit[0]: io_gpio0_o
0x3006	NOT USED	–	–	Not Used

table 6-3 system control registers (sheet 2 of 6)

address	register name	default value	R/W	description
0x3007	IO PAD SEL	0x04	RW	Bit[7:2]: Not used Bit[1]: ip_gpio3_sel Bit[0]: ip_gpio2_sel
0x3008	IO PAD SEL	0x00	RW	Bit[7]: io_vsync_sel Bit[6]: io_href_sel Bit[5]: Not used Bit[4]: io_strobe_sel Bit[3]: Not used Bit[2]: io_sda_sel Bit[1]: io_gpio1_sel Bit[0]: io_gpio0_sel
0x3009	PAD CTRL	0x06	RW	Bit[7:3]: Not used Bit[2]: pad_vsync_enb Bit[1:0]: P2X3v
0x300A	CHIP ID	0x53	R	Bit[7:0]: chip_id[23:16]
0x300B	CHIP ID	0x08	R	Bit[7:0]: chip_id[15:8]
0x300C	CHIP ID	0x41	R	Bit[7:0]: chip_id[7:0]
0x300D	PUMP CLK CTRL	0x13	RW	Bit[7]: Not used Bit[6:4]: p_pump_clk_div Bit[3]: Not used Bit[2:0]: n_pump_clk_div
0x300E	PUMP CLK CTRL	0x11	RW	Bit[7:3]: Not used Bit[2:0]: n_pump2_clk_div
0x300F	MIPI SC	0x11	RW	Bit[7]: mipi_cphy_dis2 Bit[6]: mipi_cphy_dis1 Bit[5]: mipi_cphy_dis0 Bit[4]: mipi_en Bit[3:0]: Not used
0x3010	MIPI PK	0x00	RW	Bit[7:6]: mipi_decap Bit[5]: mipi_hsen_skew Bit[4]: dis_hsen_skew Bit[3]: mipi_pclk_debug Bit[2:0]: pgm_vcm
0x3011	MIPI PK	0x04	RW	Bit[7:4]: sel_drv Bit[3:2]: pgm_lptx Bit[1:0]: r_iref

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

address	register name	default value	R/W	description
0x3012	MIPI SC	0x41	RW	Bit[7:4]: lane_num 0000: 0 lane 0001: 1 lane 0010: 2 lane 0100: 4 lane Bit[3]: Not used Bit[2]: r_phy_pd_mipi manual 0: Not used 1: Power down PHY HS TX Bit[1]: mipi_ck_lp_dir Bit[0]: phy_pad_en
0x3013	MIPI SC	0x00	RW	Bit[7:6]: mipi_d4_skew Bit[5:4]: mipi_d3_skew Bit[3:2]: mipi_d2_skew Bit[1:0]: mipi_d1_skew
0x3014	NOT USED	–	–	Not Used
0x3015	MIPI SC	0x00	RW	Bit[7]: mipi_lane_dis4 Bit[6]: mipi_lane_dis3 Bit[5]: mipi_lane_dis2 Bit[4]: mipi_lane_dis1 Bit[3]: mipi_ck_lane_dis Bit[2]: Not used Bit[1:0]: mipi_ck0_skew_o
0x3016	CLKRST0	0xF0	RW	Bit[7]: sclk_ac Bit[6]: sclk_stb Bit[5]: ULPM clock enable Bit[4]: sclk_tc Bit[3]: rst_ac Bit[2]: rst_stb Bit[1]: ULPM reset 1: Reset Bit[0]: rst_tc
0x3017	CLKRST1	0xF0	RW	Bit[7]: sclk_tpm Bit[6]: sclk_isp Bit[5]: sclk_arb Bit[4]: sclk_vfifo Bit[3]: rst_tpm Bit[2]: rst_isp Bit[1]: rst_arb Bit[0]: rst_vfifo

table 6-3 system control registers (sheet 4 of 6)

address	register name	default value	R/W	description
0x3018	CLKRST2	0xF0	RW	Bit[7]: Not used Bit[6]: sclk_mipi Bit[5]: Not used Bit[4]: sclk_efuse Bit[3]: rst_lvds Bit[2]: rst_mipi Bit[1]: rst_hdns Bit[0]: rst_efuse
0x3019	CLKRST3	0xF0	RW	Bit[7]: sclk_blc Bit[6]: sclk_ispfc Bit[5]: sclk_fmt Bit[4]: sclk_embline Bit[3]: rst_blc Bit[2]: rst_ispfc Bit[1]: rst_fmt Bit[0]: rst_embline
0x301A	CLKRST4	0xF0	RW	Bit[7]: sclk_grp Bit[6]: padclk_mipi_sc Bit[5]: pclk_vfifo Bit[4]: pclk_mipi Bit[3]: rst_grp Bit[2]: rst_mipi_sc Bit[1]: Not used Bit[0]: rst_emline manual
0x301B	CLKRST5	0x34	RW	Bit[7:6]: dac_clk_sel Bit[5]: sclk_bist20 Bit[4]: sclk_snr_sync Bit[3]: sclk_grp_fix Bit[2]: dacclk_en Bit[1]: rst_bist20 Bit[0]: rst_snr_sync
0x301C~ 0x301D	RSVD	–	–	Reserved
0x301E	CLOCK SEL	0x98	RW	Bit[7:4]: Not used Bit[3]: pclk2x_sel Bit[2:0]: Not used
0x301F	MISC CTRL	0x83	RW	Bit[7:1]: Not used Bit[0]: cen_global_o

table 6-3 system control registers (sheet 5 of 6)

address	register name	default value	R/W	description
0x3020	LOW PWR CTRL	0x00	RW	Bit[7]: Not used Bit[6]: phy_pd_mipi_pwdn_dis Bit[5]: Not used Bit[4]: stb_rst_dis 0: Reset all blocks at software standby mode 1: TC, sensor_control, ISP are reset, others not Bit[3:2]: Not used Bit[1]: phy_pd_mipi_slppd_dis Bit[0]: Not used
0x3021	A PWC PK O	0x00	RW	Bit[7:0]: a_pwc_pk_o[7:0]
0x3022	CLKRST6	0xF8	RW	Bit[7:0]: Not used
0x3023	CLKRST7	0xF0	RW	Bit[7]: sclk_psv Bit[6:5]: Not used Bit[4]: sclk_sccb Bit[3]: rst_psv Bit[2:0]: Not used
0x3024	RSVD	–	–	Reserved
0x3025	PSV MODE OPT	0x02	RW	Bit[7:0]: Not used
0x3026	CLK GATE MASK	0x00	RW	Bit[7]: sclk_avg Bit[6]: sclk_ulpm Bit[5]: pclk_mipi Bit[4]: pclk_vfifo Bit[3]: pclk_pdfifo Bit[2]: sclk_snr_sync Bit[1]: sclk_emb Bit[0]: Not used
0x3027	CLK GATE MASK	0x00	RW	Bit[7]: sclk_fmt Bit[6]: sclk_ispfc Bit[5]: sclk_blc Bit[4:3]: Not used Bit[2]: sclk_vfifo Bit[1]: sclk_isp Bit[0]: sclk_ulpm_aec
0x3028	CTRL28	0xF0	RW	Bit[7:6]: Not used Bit[5]: sclk_avg Bit[4]: sclk_ulpm Bit[3:2]: Not used Bit[1]: rst_avg Bit[0]: rst_ulpm
0x3029	RSVD	–	–	Reserved
0x302A	CHIP REVISION	0xB0	R	Bit[7:0]: chip_revision

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

address	register name	default value	R/W	description
0x3031	MIPI PK	0xA9	RW	Bit[7]: n3_vcnt_set[0] Bit[6]: discon_elvd_sl Bit[5]: open_term_en Bit[4]: pll_mipiclk_debug Bit[3]: mipi_mode 1 0: CPHY 1: DPHY Bit[2]: n3_dis_term Bit[1]: n3_testmode_en Bit[0]: lp_sr
0x3032	MIPI PK	0x00	RW	Bit[7:2]: Not used Bit[1:0]: n3_vcnt_set[2:1]
0x3033~ 0x3034	NOT USED	–	–	Not Used
0x3035	SCCB ID	0x6C	RW	Bit[7:0]: sccb_ID
0x3036	SCCB ID	0x42	RW	Bit[7:0]: sccb_ID2
0x3037	SCCB ID	0x20	RW	Bit[7:0]: sccb_alternate_ID
0x3038	SCCB ID CTRL	0x00	RW	Bit[7:5]: Not used Bit[4]: ssc_en Bit[3:1]: Not used Bit[0]: sccb_id2_no_ack
0x303E	GP IO IN1	–	R	Bit[7:3]: Not used Bit[2]: d_tpm_db_i Bit[1]: p_vsync_i Bit[0]: p_href_i
0x303F	GP IO IN2	–	R	Bit[7]: p_pwdn_i Bit[6]: p_tm_i Bit[5]: p_rst_n_i Bit[4]: p_sid_i Bit[3]: p_gpio3_i Bit[2]: p_gpio2_i Bit[1]: p_gpio1_i Bit[0]: p_gpio0_i

6.4 SCCB control [0x3100 - 0x3108]

table 6-4 SCCB registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3100	SCCB CTRL	0x00	RW	Bit[7:4]: Not used Bit[3]: r_sda_dly_en Bit[2:0]: r_sda_dly
0x3101	SCCB OPT	0x32	RW	Bit[7:6]: Not used Bit[5]: Enable register address translating table Bit[4]: en_ss_addr_inc Bit[3]: r_sda_byp_sync 0: Two clock stage sync for sda_i 1: No sync for sda_i Bit[2]: r_scl_byp_sync 0: Two clock stage sync for scl_i 1: No sync for scl_i Bit[1]: r_msk_glitch Bit[0]: r_msk_stop
0x3102	SCCB FILTER	0x00	RW	Bit[7:4]: r_sda_num Bit[3:0]: r_scl_num
0x3103	SCCB SYSREG	0x00	RW	Bit[7]: Not used Bit[6]: ctrl_rst_mipisc Bit[5]: ctrl_rst_srb Bit[4]: ctrl_rst_sccb_s Bit[3]: ctrl_rst_pon_sccb_s Bit[2]: ctrl_rst_clkmod Bit[1]: ctrl_mipi_phy_rst_o Bit[0]: ctrl_pll_rst_o
0x3104	PWUP DIS	0x01	RW	Bit[7:5]: Not used Bit[4]: r_srb_clk_syn_en Bit[3]: pwup_dis2 Bit[2]: pwup_dis1 Bit[1]: pll_clk_sel Bit[0]: pwup_dis0
0x3105	PADCLK DIV	0x11	RW	Bit[7:6]: Not used Bit[5:0]: padclk_div
0x3106	SRB HOST INPUT DIS	0x15	RW	Bit[7]: sramclk_cutoff_byp Bit[6]: sa1clk_cutoff_byp Bit[5:4]: Not used Bit[3:2]: sclk_sel Bit[1]: rst_arb Bit[0]: bypass_arb

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

address	register name	default value	R/W	description
0x3107	SRB_CTRL	0x01	RW	Bit[7]: npump_clk_ausw_dis Bit[6]: npump_clk_sw Bit[5]: auto_sleep_en Bit[4]: pd_mipi_dis_aslp Bit[3]: pumpclk_cutoff_byp Bit[2]: pclk_cutoff_byp Bit[1]: sclk_cutoff_byp Bit[0]: sclk_inv
0x3108	SRB_CTRL	0x01	RW	Bit[7:2]: Not used Bit[1]: pll_sclk_selection 0: pll_sclk_i 1: p_clk_i Bit[0]: Sleep control

6.5 group hold [0x3200 - 0x3206, 0x3208 - 0x320D, 0x3210 - 0x321A, 0x321C - 0x321F]

table 6-5 group hold registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3200	GROUP ADR0	0x00	RW	Group0 Start Address in SRAM Actual Address is {0x3200[5:0], 4'h0}
0x3201	GROUP ADR1	0x08	RW	Group1 Start Address in SRAM Actual Address is {0x3201[5:0], 4'h0}
0x3202	GROUP ADR2	0x10	RW	Group2 Start Address in SRAM Actual Address is {0x3202[5:0], 4'h0}
0x3203	GROUP ADR3	0x18	RW	Group3 Start Address in SRAM Actual Address is {0x3203[5:0], 4'h0}
0x3204	GROUP ADR4	0x20	RW	Group4 Start Address in SRAM Actual Address is {0x3204[5:0], 4'h0}
0x3205	GROUP ADR5	0x30	RW	Group5 Start Address in SRAM Actual Address is {0x3205[5:0], 4'h0}
0x3206	FSIN_CTRL	0x10	RW	Bit[7:5]: Not used Bit[4]: fsin_en Bit[3:2]: Not used Bit[1:0]: fsin_grp_num

table 6-5 group hold registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3208	GROUP ACCESS	–	W	Bit[7:4]: group_ctrl 0000: Group hold start 0001: Group hold end 1010: Group launch 1110: Fast group launch Others: Reserved Bit[3:0]: Group ID 0000: Group bank 0 0001: Group bank 1 0010: Group bank 2 0011: Group bank 3 Others: Reserved
0x3209	GRP0_PERIOD	0x00	RW	Number of Frames to Stay in Group 0
0x320A	GRP0_PERIOD	0x00	RW	Number of Frames to Stay in Group 1
0x320B	GRP0_PERIOD	0x00	RW	Number of Frames to Stay in Group 2
0x320C	GRP0_PERIOD	0x00	RW	Number of Frames to Stay in Group 3
0x320D	GRP_SWCTRL	0x01	RW	Bit[7:6]: Reserved Bit[5]: Repeat switch mode Bit[4]: context_en Bit[3:2]: Reserved Bit[1:0]: Switch back group
0x3210	GROUP_LEN0	–	R	Bit[7:0]: r_len0
0x3211	GROUP_LEN1	–	R	Bit[7:0]: r_len0
0x3212	GROUP_LEN2	–	R	Bit[7:0]: r_len0
0x3213	GROUP_LEN3	–	R	Bit[7:0]: r_len0
0x3214	GROUP_LEN4	–	R	Bit[7:0]: r_len0
0x3215	GROUP_LEN0	–	R	Bit[7:0]: r_len0
0x3216	EMB_LINE_NUM	0x01	RW	Bit[7:4]: Not used Bit[3:0]: emb_line_num
0x3217	PADDING_DATA	0x00	RW	Bit[7:0]: padding_data

table 6-5 group hold registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x3218	EMB_LINE_ CONTROL	0x00	RW	Bit[7:6]: emb_line_blk_ctrl 00: 32 sclk cycles 01: 64 sclk cycles 10: 128 sclk cycles 11: Wait for external trigger signal Bit[5]: embline_addr_en Bit[4]: r_padding_md2 0: No dummy data mixing with valid data 1: Dummy data output with valid data for every cycle Bit[3]: frame_trig_sel 0: tc_grp_wr 1: EOF Bit[2]: embline_eof_en Bit[1]: embline_sof_en Bit[0]: embline_tag_en
0x3219	EMB_TAG	0x55	RW	Bit[7:0]: emb_tag
0x321A	GRP_ACT	–	R	Active Group Indicator
0x321C	FM_CNT_GRP0	–	R	Group 0 Frame Count
0x321D	FM_CNT_GRP1	–	R	Group 1 Frame Count
0x321E	FM_CNT_GRP2	–	R	Group 2 Frame Count
0x321F	FM_CNT_GRP3	–	R	Group 3 Frame Count

6.6 power saving mode [0x3400 - 0x340D, 0x3410 - 0x3417, 0x3420 - 0x342B]

table 6-6 power saving mode registers (sheet 1 of 5)

address	register name	default value	R/W	description
0x3400	PSV CTRL	0x00	RW	Bit[7:4]: Not used Bit[3]: psv_auto_on_dis 0: PSV mode auto enable if VTS > threshold 1: Disable PSV auto on mode only (depends on r_psv_mode_en) Bit[2]: r_psv_mode_en Bit[1]: Not used Bit[0]: psv_mode 0: Keep SCLK on frame timing based on SCLK 1: Shut off SCLK at blanking frame timing switch to pad_clk domain
0x3401	AUTO SLEEP CTRL	0x00	RW	Bit[7:5]: Not used Bit[4]: stream_clk_allon Bit[3]: tc_sof_sync_en Bit[2]: blkp_sync_dis Bit[1:0]: blank_retime_opt
0x3402	HTS PAD CLK	0x00	RW	Bit[7:0]: hts_pad_clk[15:8]
0x3403	HTS PAD CLK	0xB1	RW	Bit[7:0]: hts_pad_clk[7:0]
0x3404	CS CNT INTIAL	0x00	RW	Bit[7:0]: cs_cnt_intial[15:8]
0x3405	CS CNT INTIAL	0x0F	RW	Bit[7:0]: cs_cnt_intial[7:0]
0x3406	R STREAM ST OFFS	0x08	RW	Bit[7:0]: r_stream_st_offs
0x3407	R PCHG ST OFFS	0x08	RW	Bit[7:0]: r_pchg_st_offs
0x3408	CTRL08	0x01	RW	Bit[7:4]: Not used Bit[3:0]: r_clk_winp_off
0x3409	CTRL09	0x22	RW	Bit[7:4]: Not used Bit[3:0]: r_strm_rear_offs
0x340A	CTRL0A	0x02	RW	Bit[7:4]: Not used Bit[3:0]: r_pchg_rear_offs
0x340B	RSVD	–	–	Reserved
0x340C	CTRL0C	0x10	RW	Bit[7:0]: psv_auto_on_thresh[15:8]

table 6-6 power saving mode registers (sheet 2 of 5)

address	register name	default value	R/W	description
0x340D	CTRL0D	0x00	RW	Bit[7:0]: psv_auto_on_thresh[7:0]
0x3410	CTRL10	0x00	RW	Bit[7:2]: Not used Bit[1]: vsm_stream_on Bit[0]: long_expo_mode_en
0x3411	RSVD	–	–	Reserved
0x3412	LONG EXPOSURE CONTROL	0x00	RW	Bit[7:0]: long_expo_time[31:24]
0x3413	LONG EXPOSURE CONTROL	0x00	RW	Bit[7:0]: long_expo_time[23:16]
0x3414	LONG EXPOSURE CONTROL	0x00	RW	Bit[7:0]: long_expo_time[15:8]
0x3415	LONG EXPOSURE CONTROL	0x00	RW	Bit[7:0]: long_expo_time[7:0]
0x3416	CTRL16	0x00	RW	Bit[7:2]: Not used Bit[1]: r_ramp_psv_en Bit[0]: r_ramp_psv_mode
0x3417	CTRL17	0x00	RW	Bit[7:4]: r_ramp_psv_start_offs Bit[3:0]: r_ramp_psv_end_offs
0x3420	CTRL20	0x00	RW	Bit[7:0]: r_mipi_pll_pd_sel
0x3421	R ASP PD	0x00	RW	Bit[7]: Bpg2 N-pump2 bypass to AGND Bit[6]: sreg_lp_npump N-pump low power mode Bit[5]: a_pd_pump Pump power down (all pumps) Bit[4]: a_pd_vref_core VREF core pump down Bit[3]: a_pd_column_r Right column power down Bit[2]: a_pd_column_l Left column power down Bit[1]: a_pd_XDEC XDEC power down Bit[0]: d_pd_SRAM SRAM power down

table 6-6 power saving mode registers (sheet 3 of 5)

address	register name	default value	R/W	description
0x3422	R ASP PD	0x00	RW	Bit[7]: sreg_bp_ppump1 (high-z) P_pump1 power down Bit[6]: sreg_pwrdn_vrd VRFD buffer power down Bit[5]: ulpm_vref VREF power down Bit[4]: ulpm_dac DAC power down Bit[3]: ulpm_v2i V2I power down Bit[2]: sreg_bpA_pp1 P-pump1 bypass to AVDD Bit[1]: sreg_bpA_pp2 P-pump2 bypass to AVDD Bit[0]: bpg1 N-pump1 bypass to AGND
0x3423	R ASP PD	0x00	RW	Bit[7]: d_col_pwrdn_l Left column power down (vln/rampbuf/dcomp) Bit[6]: d_col_cbar_pwrdn Color bar power down Bit[5]: d_col_pwrdn_r Right column power down (vln/rampbuf/dcomp) Bit[4]: DAC clock divider power down Bit[3]: ATEST power down Bit[2]: sreg_bp_npump2 (high-z) N-pump2 power down Bit[1]: sreg_bp_npump1 (pull down through resistor) N-pump1 power down Bit[0]: sreg_bp_ppump2 (high-z) P-pump2 power down
0x3424	R ASP PD	0x48	RW	Bit[7:1]: Not used Bit[0]: SCCB clock power down

table 6-6 power saving mode registers (sheet 4 of 5)

address	register name	default value	R/W	description
0x3425	R ASP PD SEL	0x00	RW	Bit[7:6]: bpg2 N-pump2 bypass to AGND 00: Register 01: Stream blanking 10: Stream and PCHG blanking 11: Stream to PCHG blanking Bit[5:4]: sreg_lp_npump N-pum low power mode 00: Register 01: Stream blanking 10: Stream and PCHG blanking 11: Stream to PCHG blanking Bit[3:2]: a_pd_pump Pump power down (all pumps) 00: Register 01: Stream blanking 10: Stream and PCHG blanking 11: Stream to PCHG blanking Bit[1:0]: a_pd_vref_core VREF core pump down 00: Register 01: Stream blanking 10: Stream and PHCG blanking 11: Stream to PCHG blanking
0x3426	R ASP PD SEL	0x00	RW	Bit[7:6]: A_pd_column_r Right column power down 00: Register 01: Stream blanking 10: Stream and PCHG blanking 11: Stream to PCHG blanking Bit[5:4]: A_pd_column_l Left column power down 00: Register 01: Stream blanking 10: Stream and PCHG blanking 11: Stream to PCHG blanking Bit[3:2]: A_pd_XDEC XDEC power down 00: Register 01: Stream blanking 10: Stream and PCHG blanking 11: Stream to PCHG blanking Bit[1:0]: d_pd_SRAM SRAM power down 00: Register 01: Stream blanking 10: Stream and PCHG blanking 11: Stream to PCHG blanking

table 6-6 power saving mode registers (sheet 5 of 5)

address	register name	default value	R/W	description
0x3427	R ASP PD SEL	0x00	RW	Bit[7:6]: sreg_bp_ppump1 (high-z) P_pump1 power down 00: Register 01: Stream blanking 10: Stream and PCHG blanking 11: Stream to PCHG blanking Bit[5:4]: sreg_pwrdn_vrd VRFD buffer power down 00: Register 01: Stream blanking 10: Stream and PCHG blanking 11: Stream to PCHG blanking Bit[3:2]: ULPM_vref VREF power down Bit[1:0]: ULPM_dac DAC power down
0x3428	R ASP PD SEL	0x00	RW	Bit[7:6]: ULPM_v2i V2I power down Bit[5:4]: sreg_bpA_pp1 P-pump1 bypass to AVDD Bit[3:2]: sreg_bpA_pp2 P-pump2 bypass to AVDD Bit[1:0]: Bpg1 N-pump1 bypass to AGND
0x3429	R ASP PD SEL	0x00	RW	Bit[7:6]: d_col_pwrdn_l Left column power down (vln/rampbuf/dcomp) Bit[5:4]: d_col_cbar_pwrdn Color bar power down Bit[3:2]: d_col_pwrdn_r Right column power down (vln/rampbuf/dcomp) Bit[1:0]: DAC clock divider power down
0x342A	R ASP PD SEL	0x00	RW	Bit[7:6]: ATEST power down Bit[5:4]: sreg_bp_npump2 (high-z) N-pump2 power down Bit[3:2]: sreg_bp_npump1 (pull down through resistor) N-pump1 power down Bit[1:0]: sreg_bp_ppump2 (high-z) P-pump2 power down
0x342B	R ASP PD SEL	0x00	RW	Bit[7:2]: Not used Bit[1:0]: SCCB clock power down 00: Register 01: Stream blanking 10: Stream and PCHG blanking 11: Stream to PCHG blanking

6.7 MEC/MGC control [0x3501 - 0x350F, 0x3511 - 0x352F]

table 6-7 MEC/MGC control registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x3501	LONG EXPO	0x00	RW	Bit[7:0]: Long exposure[15:8]
0x3502	LONG EXPO	0x40	RW	Bit[7:0]: Long exposure[7:0]
0x3503	AEC MANUAL	0x88	RW	Bit[7]: Not used Bit[6]: Digital fraction gain delay option 0: Delay 1 frame 1: Do not delay 1 frame Bit[5]: Gain change delay option 0: Delay 1 frame 1: Do not delay 1 frame Bit[4]: Gain delay option 0: Delay 1 frame 1: Do not delay 1 frame Bit[3]: Not used Bit[2]: Gain manual as sensor gain 0: Input gain in real gain format 1: Input gain in sensor gain format Bit[1]: Exposure delay option (must be 0) 0: Delay 1 frame 1: Not used Bit[0]: Exposure change delay option (must be 0) 0: Delay 1 frame 1: Not used
0x3504	RSVD	–	–	Reserved
0x3505	GCVT OPTION	0x80	RW	Gain Conversation Option Bit[7]: DAC fixed gain bit Bit[6:4]: Sensor gain fixed bit Bit[3:2]: Sensor gain pregain option (debug only, always set it to 0) Bit[1:0]: Sensor gain option for transferring real gain to sensor gain format
0x3506	RSVD	–	–	Reserved

table 6-7 MEC/MGC control registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x3507	GAIN SHIFT	0x00	RW	Bit[7:4]: Not used Bit[3]: fine_snr_gain_16_en Back gain enable Bit[2]: Priority_6 switch_snr_gain_en 0: dac[2:0]_comp[1:0] 1: comp[1:0]_dac[2:0] Bit[1:0]: Gain shift option 00: No shift 01: Left shift 1 bit 10: Left shift 2 bit 11: Left shift 3 bit
0x3508	LONG GAIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: Gain[13:8]
0x3509	LONG GAIN	0x80	RW	Bit[7:0]: Long gain[7:0] If 0x3503[2] = 0, then Gain[12:0] is real gain format, where low 7 bits are fraction bits. real_gain = Gain[12:0]/128 For example, 0x080 is 1x gain, 0x140 is 2.5x gain. Maximum gain is 15.5x; 0x7C0. If 0x3503[2]=1, then Gain[12:0] is sensor gain format, where: Gain[12:8]: Coarse gain 00000: 1x 00001: 2x 00011: 4x 00111: 8x Gain[7]: 1'b1 Gain[6:3]: Fine gain Gain[2:0]: Always 0 For example, 0x080 is 1x gain, 0x180 is 2x gain, and 0x380 is 4x gain.
0x350A	LONG DIGIGAIN	0x04	RW	Bit[7:6]: Not used Bit[5:0]: Long digital gain[13:8] 4.10 format
0x350B	LONG DIGIGAIN	0x00	RW	Bit[7:0]: Long digital gain[7:0] 4.10 format
0x350C	SHORT GAIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: Short gain[13:8]
0x350D	SHORT GAIN	0x80	RW	Bit[7:0]: Short gain[7:0]
0x350E	SHORT DIGIGAIN	0x04	RW	Bit[7:6]: Not used Bit[5:0]: Short digital gain[13:8]
0x350F	SHORT DIGIGAIN	0x00	RW	Bit[7:0]: Short digital gain[7:0]

table 6-7 MEC/MGC control registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x3511	SHORT EXPO	0x00	RW	Bit[7:0]: Short exposure[15:8]
0x3512	SHORT EXPO	0x20	RW	Bit[7:0]: Short exposure[7:0]
0x3513	SNR_GAIN_L	–	R	Bit[7:6]: Not used Bit[5:0]: Long sensor gain[13:8]
0x3514	SNR_GAIN_L	–	R	Bit[7:0]: Long sensor gain[7:0]
0x3515	FINE_SNR_ GAIN_L	–	R	Bit[7:6]: Not used Bit[5:0]: Long fine sensor gain
0x3516	SNR_GAIN_S	–	R	Bit[7:6]: Not used Bit[5:0]: Short sensor gain[13:8]
0x3517	SNR_GAIN_S	–	R	Bit[7:0]: Short sensor gain[7:0]
0x3518	FINE_SNR_ GAIN_S	–	R	Bit[7:6]: Not used Bit[5:0]: Short fine sensor gain
0x3519	GAIN_DELAY_ SWAP_OPTION	0x00	RW	Bit[7:3]: Not used Bit[2]: Debug mode (do not change) Bit[1]: digital_gain Apply select 00: Digital gain apply in BLC 01: Digital gain apply in MWB Bit[0]: Auto gain delay option 0: Real gain and digital gain delay manual mode 1: Real gain and digital gain will auto sync with exposure. When exposure and auto gain change at the same time, both gain will delay 1 frame to sync with exposure; otherwise, there will be no frame delay in gain.
0x351A	BLC_DIG_GAIN _CUR_L	–	R	Bit[7:6]: Not used Bit[5:0]: Current BLC long digital gain[13:8]
0x351B	BLC_DIG_GAIN _CUR_L	–	R	Bit[7:0]: Current BLC long digital gain[7:0]
0x351C	BLC_DIG_GAIN _CUR_S	–	R	Bit[7:6]: Not used Bit[5:0]: Current BLC short digital gain[13:8]
0x351D	BLC_DIG_GAIN _CUR_S	–	R	Bit[7:0]: Current BLC short digital gain[7:0]
0x351E	MWB_DIG_ CUR_L	–	R	Bit[7:6]: Not used Bit[5:0]: Current MWB long digital gain[13:8]
0x351F	MWB_DIG_ CUR_L	–	R	Bit[7:0]: Current MWB long digital gain[7:0]

table 6-7 MEC/MGC control registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x3520	MWB_DIG_CUR_S	–	R	Bit[7:6]: Not used Bit[5:0]: Current MWB long digital gain[13:8]
0x3521	MWB_DIG_CUR_S	–	R	Bit[7:0]: Current MWB long digital gain[7:0]
0x3522	BLC_REAL_CUR_L	–	R	Bit[7:2]: Not used Bit[1:0]: Current BLC long real gain[9:8]
0x3523	BLC_REAL_CUR_L	–	R	Bit[7:0]: Current BLC long real gain[7:0]
0x3524	BLC_REAL_CUR_S	–	R	Bit[7:2]: Not used Bit[1:0]: Current BLC short real gain[9:8]
0x3525	BLC_REAL_CUR_S	–	R	Bit[7:0]: Current BLC short real gain[7:0]
0x3526	ISP_REAL_CUR_L	–	R	Bit[7:4]: Not used Bit[3:0]: Current ISP long real gain[11:8]
0x3527	ISP_REAL_CUR_L	–	R	Bit[7:0]: Current ISP long real gain[7:0]
0x3528	ISP_REAL_CUR_S	–	R	Bit[7:4]: Not used Bit[3:0]: Current ISP short real gain[11:8]
0x3529	ISP_REAL_CUR_S	–	R	Bit[7:0]: Current ISP short real gain[7:0]
0x352A	EXPO_CUR_L	–	R	Bit[7:4]: Not used Bit[3:0]: Current long exposure[19:16]
0x352B	EXPO_CUR_L	–	R	Bit[7:0]: Current long exposure[15:8]
0x352C	EXPO_CUR_L	–	R	Bit[7:0]: Current long exposure[7:0]
0x352D	EXPO_CUR_S	–	R	Bit[7:4]: Not used Bit[3:0]: Current short exposure[19:16]
0x352E	EXPO_CUR_S	–	R	Bit[7:0]: Current short exposure[15:8]
0x352F	EXPO_CUR_S	–	R	Bit[7:0]: Current short exposure[7:0]

6.8 analog control [0x3600 - 0x3637, 0x3660 - 0x3672, 0x3680 - 0x36A5]

table 6-8 analog control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3600~ 0x3637	ANA CTRL	–	–	Analog Control Registers
0x3660	CORE 0	0x40	RW	Bit[7]: pseudo_12bit_en Bit[6]: pseudo_12bit_rnd_option Bit[5]: mipi_pclk_sel 0: MIPI 1: LVDS Bit[4]: raw12_en Bit[3]: Not used Bit[2]: ramp_bit 0: 10-bit 1: 11-bit Bit[1]: rip_seof_ispfc_en Bit[0]: rip_sof_vfifo_en
0x3661	CORE 1	0x06	RW	Bit[7]: byp_isp Bit[6]: sof_s_img For stagger HDR sof_s after image_href Bit[5]: testmode_sel 0: Long exposure 1: Short exposure Bit[4]: stg_hdr_dt_mode Bit[3]: new_stg_hdr_en Bit[2]: stg_hdr_short_en Debug mode Bit[1]: stg_hdr_long_en Debug mode Bit[0]: stg_hdr_align_en
0x3662	CORE 2	0x00	RW	Bit[7]: raw_bit_shift_dis Bit[6]: sram_tm Bit[5]: Not used Bit[4]: col_vln_sig_clamp_en_disable Bit[3]: bit_shift_clip_en Bit[2:0]: bit_shift_mode

table 6-8 analog control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3663	CORE 3	0x00	RW	Bit[7]: rst_opt Bit[6]: rst_tpm Bit[5:4]: Not used Bit[3]: lvds_vsync_sel Bit[2]: lvds_ck_data_sel Bit[1]: split_lvds Two LVDS to transmit long and short exposure data Bit[0]: split_lane One LVDS, two lanes for long, and other two lanes for short, new STG HDR
0x3664	CORE 4	0x00	RW	Bit[7]: r_fix_timing_en Bit[6]: Not used Bit[5]: fix_timing_sof_reset Bit[4:0]: sof2ready_adj
0x3665	CORE 5	0x12	RW	Bit[7]: emb_en Bit[6]: emb_data_lsb Bit[5:0]: emb_dt
0x3666~ 0x3669	RSVD	–	–	Reserved
0x366A	CORE A	0x00	RW	Bit[7:0]: hts_pclk[15:8] Fix timing mode
0x366B	CORE B	0x00	RW	Bit[7:0]: hts_pclk[7:0] Fix timing mode
0x366C	CORE C	0x00	RW	Bit[7:0]: y_output_size[15:8] Manual for MIPI
0x366D	CORE D	0x00	RW	Bit[7:0]: y_output_size[7:0] Manual for MIPI
0x366E	CORE E	0x00	RW	Bit[7]: rlong_reverse Bit[6:5]: ln_mask_opt_l Bit[4:3]: ln_mask_opt_s Bit[2]: new_stg_hdr_blc_line_en Bit[1]: hdr_vsize_out_sel Bit[0]: hdr_vsize_man_en
0x366F	CORE F	0x0F	RW	Bit[7:4]: Not used Bit[3:0]: Pseudo data noise mask l[11:8]
0x3670	CORE 10	0xFF	RW	Bit[7:0]: Pseudo data noise mask l[7:0]
0x3671	CORE 11	0x0F	RW	Bit[7:4]: Not used Bit[3:0]: Pseudo data noise mask s[11:8]
0x3672	CORE 12	0xFF	RW	Bit[7:0]: Pseudo data noise mask s[7:0]

table 6-8 analog control registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x3680~ 0x36A5	ANA CTRL	–	–	Analog Control Registers

6.9 sensor control [0x3700 - 0x37FF]

table 6-9 sensor control registers

address	register name	default value	R/W	description
0x3700~ 0x37FF	SNR CTRL	–	–	Sensor Timing Control Registers

6.10 timing control [0x3800 - 0x381D, 0x3820 - 0x3829, 0x3832 - 0x3833, 0x3835]

table 6-10 timing control registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x3800	X ADDR START	0x00	RW	Bit[7:4]: Not used Bit[3:0]: x_addr_start[11:8] Array horizontal start point
0x3801	X ADDR START	0x00	RW	Bit[7:0]: x_addr_start[7:0] Array horizontal start point
0x3802	Y ADDR START	0x00	RW	Bit[7:4]: Not used Bit[3:0]: y_addr_start[11:8] Array vertical start point
0x3803	Y ADDR START	0x04	RW	Bit[7:0]: y_addr_start[7:0] Array vertical start point
0x3804	X ADDR END	0x07	RW	Bit[7:4]: Not used Bit[3:0]: x_addr_end[11:8] Array horizontal end point
0x3805	X ADDR END	0x8F	RW	Bit[7:0]: x_addr_end[7:0] Array horizontal end point

table 6-10 timing control registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x3806	Y ADDR END	0x04	RW	Bit[7:4]: Not used Bit[3:0]: y_addr_end[11:8] Array vertical end point
0x3807	Y ADDR END	0x43	RW	Bit[7:0]: y_addr_end[7:0] Array vertical end point
0x3808	X OUTPUT SIZE	0x07	RW	Bit[7:4]: Not used Bit[3:0]: x_output_size[11:8] ISP horizontal output width
0x3809	X OUTPUT SIZE	0x80	RW	Bit[7:0]: x_output_size[7:0] ISP horizontal output width
0x380A	Y OUTPUT SIZE	0x04	RW	Bit[7:4]: Not used Bit[3:0]: y_output_size[11:8] ISP vertical output height
0x380B	Y OUTPUT SIZE	0x38	RW	Bit[7:0]: y_output_size[7:0] ISP vertical output height
0x380C	HTS	0x02	RW	Bit[7:0]: HTS[15:8] Total horizontal timing size
0x380D	HTS	0x78	RW	Bit[7:0]: HTS[7:0] Total horizontal timing size
0x380E	VTS	0x04	RW	Bit[7:0]: VTS[15:8] Total vertical timing size
0x380F	VTS	0xA0	RW	Bit[7:0]: VTS[7:0] Total vertical timing size
0x3810	ISP X WIN	0x00	RW	Bit[7:0]: isp_x_win[15:8] ISP horizontal windowing offset
0x3811	ISP X WIN	0x08	RW	Bit[7:0]: isp_x_win[7:0] ISP horizontal windowing offset
0x3812	ISP Y WIN	0x00	RW	Bit[7:4]: Not used Bit[3:0]: isp_y_win[11:8] ISP vertical windowing offset
0x3813	ISP Y WIN	0x04	RW	Bit[7:0]: isp_y_win[7:0] ISP vertical windowing offset
0x3814	X INC ODD	0x01	RW	Bit[7:5]: Not used Bit[4:0]: x_odd_inc
0x3815	X INC EVEN	0x01	RW	Bit[7:5]: Not used Bit[4:0]: x_even_inc
0x3816	Y_INC_ODD	0x01	RW	Bit[7:5]: Not used Bit[4:0]: y_odd_inc

table 6-10 timing control registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x3817	Y_INC_EVEN	0x01	RW	Bit[7:5]: Not used Bit[4:0]: y_even_inc
0x3818	VSYNC START	0x00	RW	Bit[7:0]: vsync_start[15:8] VSYNC start point
0x3819	VSYNC START	0x00	RW	Bit[7:0]: vsync_start[7:0] VSYNC start point
0x381A	VSYNC END	0x00	RW	Bit[7:0]: vsync_end[15:8] VSYNC end point
0x381B	VSYNC END	0x01	RW	Bit[7:0]: vsync_end[7:0] VSYNC end point
0x381C	BLC_NUM_OPTION	0x0E	RW	Bit[7:5]: ablc_adj Bit[4]: Not used Bit[3]: tc_sof_sel Bit[2]: vref_new_stg_hdr_s_rev Bit[1]: new_stg_hdr_en Bit[0]: vref_img_rev
0x381D	BLC_OPTION	0x40	RW	Bit[7]: Reserved Bit[6]: r_tc_sof_dly Bit[5]: r_man_blc_num_sel Bit[4:0]: r_man_blc_num[5:1] r_man_blc_num[0] = 1'b0
0x3820	FORMAT1	0x80	RW	Bit[7]: vsub48_blc Bit[6]: blc_flip Bit[5:3]: Not used Bit[2]: Flip Bit[1]: Vertical mono binning Bit[0]: Vertical binning
0x3821	FORMAT2	0x00	RW	Bit[7:6]: Digital in digital domain Bit[5]: Stagger HDR enable Bit[4:3]: Not used Bit[2]: Mirror Bit[1]: Horizontal mono binning Bit[0]: Horizontal binning
0x3822	REG22	0x04	RW	Bit[7:6]: Reserved Bit[5]: r_fix_cnt_en Bit[4]: vts_add_dis Bit[3:0]: r_grp_adj
0x3823	REG23	0x08	RW	Bit[7]: ext_vs_re Bit[6]: ext_vs_en Bit[5]: vts_no_latch Bit[4]: init_man Bit[3:0]: r_grp_adj

table 6-10 timing control registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x3824	CS RST FSIN	0x00	RW	Bit[7:0]: cs_rst_fsin[15:8] CS reset value at vs_ext
0x3825	CS RST FSIN	0x20	RW	Bit[7:0]: cs_rst_fsin[7:0] CS reset value at vs_ext
0x3826	R RST FSIN	0x00	RW	Bit[7:0]: r_rst_fsin[15:8] R reset value at vs_ext
0x3827	R RST FSIN	0x08	RW	Bit[7:0]: r_rst_fsin[7:0] R reset value at vs_ext
0x3828	REG28	0x30	RW	Bit[7]: Not used Bit[6:0]: r_ablc_num_o
0x3829	MIRROR_FLIP_CUT	0x03	RW	Bit[7:3]: Not used Bit[2]: r_flip_cut_opt Bit[1]: r_flip_cut Bit[0]: r_mirr_cut
0x3832	REG32	0x08	RW	Bit[7:4]: vsync_width Bit[3:0]: r_init_offset
0x3833	REG33	0x05	RW	Bit[7:3]: Reserved Bit[2]: r_stg_grphold_nomask Bit[1]: r_stg_sleep_nomask Bit[0]: r_stg_hdr_grp_wr_opt
0x3835	REG35	0x00	RW	Bit[7:6]: Reserved Bit[5:4]: r_hbin Bit[3:0]: r_array_skip

6.11 AEC control [0x3A01 - 0x3A0F]

table 6-11 AEC control registers

address	register name	default value	R/W	description
0x3A01	AEC CTRL01	0x01	RW	Bit[7]: Debug mode Bit[6:4]: long_short_ration Bit[3:0]: min_expo
0x3A02	EXP CTRL02	0x78	RW	Bit[7:0]: Top stable range-1
0x3A03	EXP CTRL03	0x68	RW	Bit[7:0]: Bottom stable range-1
0x3A04~ 0x3A0F	RSVD	–	–	Reserved

6.12 strobe [0x3B00, 0x3B02 - 0x3B05]

table 6-12 strobe control registers

address	register name	default value	R/W	description
0x3B00	RSTRB	0x00	RW	Bit[7]: Strobe on/off Bit[6]: Strobe polarity 0: Active high 1: Active low Bit[5:4]: width_in_xenon Bit[3]: Not used Bit[2:0]: Strobe mode 000: Xenon 001: LED1 010: LED2 011: LED3 100: LED4
0x3B02	STROBE ADD DUMMY	0x00	RW	Bit[7:0]: strobe_add_dummy[15:8] Dummy line number added at strobe
0x3B03	STROBE ADD DUMMY	0x00	RW	Bit[7:0]: strobe_add_dummy[7:0] Dummy line number added at strobe
0x3B04	STROBE CTL1	0x00	RW	Bit[7]: strobe_valid (read only) Bit[6:4]: Not used Bit[3]: start_point_sel Bit[2]: Strobe repeat enable Bit[1:0]: Strobe lantency 00: Strobe generated at next frame 01: Delay one frame Strobe generated 2 frames later 10: Delay one frame Strobe generated 3 frames later 11: Delay one frame Strobe generated 4 frames later
0x3B05	STROBE WIDTH	0x00	RW	Bit[7:2]: Strobe pulse width step Bit[1:0]: Strobe pulse width gain Strobe pulse width = 128 × (2^gain) × (step+1) × sclk_period

6.13 OTP control [0x3D80 - 0x3D95]

table 6-13 OTP control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3D80	OTP_PROGRAM_CTRL	–	RW	Bit[7]: OTP_wr_busy (read only) Bit[6:1]: Not used Bit[0]: OTP_program_enable (write only)
0x3D81	OTP_LOAD_CTRL	–	RW	Bit[7]: OTP_rd_busy (read only) Bit[6]: Not used Bit[5]: OTP_bist_error (read only) Bit[4]: OTP_bist_done (read only) Bit[3:1]: Not used Bit[0]: OTP_load_enable (write only)
0x3D82	OTP_PGM_PULSE	0x5E	RW	Program Strobe Pulse Width Unit: 8x System Clock Period
0x3D83	OTP_LOAD_PULSE	0x08	RW	Load Strobe Pulse Width Unit: System Clock Period
0x3D84	OTP_MODE_CTRL	0x00	RW	Bit[7]: Program disable 0: Not used 1: Disable Bit[6]: Mode select 0: Auto mode 1: Manual mode Bit[5:1]: Reserved Bit[0]: bank_sram_switch
0x3D85	OTP_REG85	0x1B	RW	Bit[7]: Reserved Bit[6]: r_otp_bist_comp_val Bit[5]: otp_bist_select 0: Compare with SRAM 1: Compare with zero Bit[4]: OTP_bist_enable Bit[3]: OTP power up load data enable Bit[2]: Reserved Bit[1]: OTP power up load setting enable Bit[0]: OTP writer register load setting enable
0x3D86	SRAM_TEST_SIGNALS	0x00	RW	Bit[7:6]: Reserved Bit[5]: r_test Bit[4]: r_rme Bit[3:0]: r_rm
0x3D87	OTP_PS2CS	0x0A	RW	OTP PS to CSB Delay Unit: System Clock Period
0x3D88	OTP_START_ADDRESS	0x00	RW	OTP Start High Address for Manual Mode

table 6-13 OTP control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3D89	OTP_START_ ADDRESS	0x00	RW	OTP Start Low Address for Manual Mode
0x3D8A	OTP_EN_ADDRESS	0x00	RW	OTP End High Address for Manual Mode
0x3D8B	OTP_END_ADDRESS	0x00	RW	OTP End Low Address for Manual Mode
0x3D8C	OTP_SETTING_STT_ ADDRESS	0x00	RW	OTP Start High Address for Load Setting
0x3D8D	OTP_SETTING_STT_ ADDRESS	0x00	RW	OTP Start Low Address for Load Setting
0x3D8E	OTP_ERR_ADDRESS	–	R	OTP Error Address High
0x3D8F	OTP_ERR_ADDRESS	–	R	OTP Error Address Low
0x3D90	R_BASE_ADR_H	0x00	RW	Gap Between Strobe Pulse When Programming
0x3D91	R_BASE_ADR_H	0x00	RW	Gap Between Strobe Pulse When Loading
0x3D92	PGENB_TIMING	0xE2	RW	Bit[7:4]: t_pgenb_end Bit[3:0]: t_pgenb_start
0x3D93	VDDQ_TIMING	0x46	RW	Bit[7:4]: t_vddq_end Bit[3:0]: t_vddq_start
0x3D94	OTP_CTRL	0x0A	RW	Bit[7:0]: Gap between strobe pulse when programming
0x3D95	OTP_CTRL	0x06	RW	Bit[7:0]: Gap between strobe pulse when loading

6.14 PSRAM control [0x3F00 - 0x3F03, 0x3F0E - 0x3F0F]

table 6-14 PSRAM control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3F00	PSRAM REG0	0x04	RW	Bit[7]: sread_st_opt 1 Manual mode 2 at asp_start Bit[6]: sread_d1r Bit[5]: sread_img_d1r Bit[4]: srclk_inv Bit[3]: Not used Bit[2]: srclk_div2 enable Bit[1:0]: PSRAM address increase number

table 6-14 PSRAM control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3F01	PSRAM REG1	0x00	RW	Bit[7:0]: sread_man_st_pt[15:8]
0x3F02	PSRAM REG2	0x00	RW	Bit[7:0]: sread_mam_st_pt[7:0]
0x3F03	NOT USED	–	–	Not Used
0x3F0E	SRAM READPOINT	–	R	Bit[7:0]: sram_readpoint[15:8] SRAM readout point at CS
0x3F0F	SRAM READPOINT	–	R	Bit[7:0]: sram_readpoint[7:0] SRAM readout point at CS

6.15 BLC control [0x4000 - 0x402F, 0x4040 - 0x409F, 0x40C0 - 0x40CF]

table 6-15 BLC control registers (sheet 1 of 8)

address	register name	default value	R/W	description
0x4000	BLC CTRL00	0xF8	RW	Bit[7]: offset_trigger_en Bit[6]: exp_trig_en Bit[5]: gain_trig_en Bit[4]: fmt_trig_en Bit[3]: rst_trig_en Bit[2]: man_trig_en Bit[1]: blc_freeze Bit[0]: blc_always_update
0x4001	BLC CTRL01	0x2B	RW	Bit[7]: zero_In_out_en Bit[6]: blc_In_out_en Bit[5]: blc_dithering_en Bit[4]: man_offset_en Bit[3]: median_filter_en Bit[2]: v15_one_chnl_en Bit[1]: dc_blc_en Bit[0]: blc_en
0x4002~ 0x4003	RSVD	–	–	Reserved
0x4004	BLK LVL TARGET	0x00	RW	Bit[7:4]: Not used Bit[3:0]: blc_lvl_target[11:8]
0x4005	BLK LVL TARGET	0x40	RW	Bit[7:0]: blc_lvl_target[7:0]
0x4006	HWIN PAD	0x00	RW	Bit[7:4]: Not used Bit[3:0]: hwin_pad[11:8]
0x4007	HWIN PAD	0x04	RW	Bit[7:0]: hwin_pad[7:0]

table 6-15 BLC control registers (sheet 2 of 8)

address	register name	default value	R/W	description
0x4008	BLC CTRL08	0x00	RW	Bit[7:0]: bl_start
0x4009	BLC CTRL09	0x0F	RW	Bit[7:0]: bl_end
0x400A	OFF LIM TH	0x02	RW	Bit[7:5]: Not used Bit[4:0]: off_lim_th[12:8]
0x400B	OFF LIM TH	0x00	RW	Bit[7:0]: off_lim_th[7:0]
0x400C	HWIN OFF	0x00	RW	Bit[7:4]: Not used Bit[3:0]: hwin_off[11:8]
0x400D	HWIN OFF	0x00	RW	Bit[7:0]: hwin_off[7:0]
0x400E	BLC CTRL0E	0x00	RW	Bit[7:0]: mf_th
0x400F	BLC CTRL0F	0xA0	RW	Bit[7]: r_threshold_en Bit[6]: r_set_zb Bit[5:4]: output_bits 00: 10 bits 01: 11 bits 10: 12 bits Bit[3]: Not used Bit[2]: r_en_adp_k Bit[1]: r_dc_offset_mode Bit[0]: r_compute_offset_v15
0x4010	BLC CTRL10	0x12	RW	Bit[7]: r_img_gfirst_rvs Bit[6]: r_blk_rblue_rvs Bit[5]: r_img_rblue_rvs Bit[4]: r_dc_man Bit[3]: target_adj_dis Bit[2]: cmp_en Bit[1]: kcoef_man_en Bit[0]: man_avg_en
0x4011	BLC CTRL11	0xFF	RW	Bit[7]: r_off_cmp_man_en Bit[6]: off_chg_mf_en Bit[5]: fmt_chg_mf_en Bit[4]: gain_chg_mf_en Bit[3]: rst_mf_mode Bit[2]: off_chg_mf_mode Bit[1]: fmt_chg_mf_mode Bit[0]: gain_chg_mf_mode
0x4012	BLC CTRL12	0x08	RW	Bit[7:0]: rst_trig_fn
0x4013	BLC CTRL13	0x02	RW	Bit[7:0]: fmt_trig_fn
0x4014	BLC CTRL14	0x02	RW	Bit[7:0]: gain_trig_fn
0x4015	BLC CTRL15	0x02	RW	Bit[7:0]: off_trig_fn

table 6-15 BLC control registers (sheet 3 of 8)

address	register name	default value	R/W	description
0x4016	OFF TRIG TH	0x00	RW	Bit[7:2]: Not used Bit[1:0]: off_trig_th[9:8] off_trig_th high 2 bits
0x4017	OFF TRIG TH	0x04	RW	Bit[7:0]: off_trig_th[7:0] off_trig_th low 8 bits
0x4018	NOT USED	–	–	Not Used
0x4019	BLC CTRL19	0x04	RW	Bit[7:0]: r_blk_ln_num
0x401A	BLC CTRL1A	0x48	RW	Bit[7]: r_h_size_man_en Bit[6]: r_kcoef_mirror Bit[5]: r_adp_dc_switch_en Bit[4]: gain_trig_beh Bit[3]: format_trig_beh Bit[2]: r_ln_man Bit[1:0]: byp_mode CZ V1.54 change start
0x401B~ 0x401F	RSVD	–	–	Reserved
0x4020	BLC CTRL20	0x00	RW	Bit[7:0]: off_cmp_th0000
0x4021	BLC CTRL21	0x00	RW	Bit[7:0]: off_cmp_th0001
0x4022	BLC CTRL22	0x00	RW	Bit[7:0]: off_cmp_th0010
0x4023	BLC CTRL23	0x00	RW	Bit[7:0]: off_cmp_th0011
0x4024	BLC CTRL24	0x00	RW	Bit[7:0]: off_cmp_th0100
0x4025	BLC CTRL25	0x00	RW	Bit[7:0]: off_cmp_th0101
0x4026	BLC CTRL26	0x00	RW	Bit[7:0]: off_cmp_th0110
0x4027	BLC CTRL27	0x00	RW	Bit[7:0]: off_cmp_th0111
0x4028	THRES CTRL	0x4F	RW	Bit[7]: Not used Bit[6]: r_lim_bits_en Bit[5]: r_thres_en Bit[4]: r_thres_man_en Bit[3:0]: r_thres_hist_cutoff
0x4029	R THRES DELTA CUTOFF	0x01	RW	Bit[7:5]: Not used Bit[4:0]: r_thres_delta_cutoff[4:0]
0x402A	R THRES DELTA	0x00	RW	Bit[7:0]: r_thres_delta[7:0]
0x402B	R THRES LINE	0x00	RW	Bit[7:6]: Not used Bit[5:0]: r_thres_line[5:0]
0x402C	R THRES MAN	0x00	RW	Bit[7:4]: Not used Bit[3:0]: r_thres_man[11:8]

table 6-15 BLC control registers (sheet 4 of 8)

address	register name	default value	R/W	description
0x402D	R THRES MAN	0x00	RW	Bit[7:0]: r_thres_man[7:0]
0x402E~ 0x402F	NOT USED	–	–	Not Used
0x4040	OFF MAN0000	0x00	RW	Bit[7:5]: Not used Bit[4:0]: off_man0000[12:8]
0x4041	OFF MAN0000	0x00	RW	Bit[7:0]: off_man0000[7:0]
0x4042	OFF MAN0001	0x00	RW	Bit[7:5]: Not used Bit[4:0]: off_man0001[12:8]
0x4043	OFF MAN0001	0x00	RW	Bit[7:0]: off_man0001[7:0]
0x4044	OFF MAN0010	0x00	RW	Bit[7:5]: Not used Bit[4:0]: off_man0010[12:8]
0x4045	OFF MAN0010	0x00	RW	Bit[7:0]: off_man0010[7:0]
0x4046	OFF MAN0011	0x00	RW	Bit[7:5]: Not used Bit[4:0]: off_man0011[12:8]
0x4047	OFF MAN0011	0x00	RW	Bit[7:0]: off_man0011[7:0]
0x4048	OFF MAN0100	0x00	RW	Bit[7:5]: Not used Bit[4:0]: off_man0100[12:8]
0x4049	OFF MAN0100	0x00	RW	Bit[7:0]: off_man0100[7:0]
0x404A	OFF MAN0101	0x00	RW	Bit[7:5]: Not used Bit[4:0]: off_man0101[12:8]
0x404B	OFF MAN0101	0x00	RW	Bit[7:0]: off_man0101[7:0]
0x404C	OFF MAN0110	0x00	RW	Bit[7:5]: Not used Bit[4:0]: off_man0110[12:8]
0x404D	OFF MAN0110	0x00	RW	Bit[7:0]: off_man0110[7:0]
0x404E	OFF MAN0111	0x00	RW	Bit[7:5]: Not used Bit[4:0]: off_man0111[12:8]
0x404F	OFF MAN0111	0x00	RW	Bit[7:0]: off_man0111[7:0]
0x4050	BLC CTRL50	0x04	RW	Bit[7:0]: zl_start
0x4051	BLC CTRL51	0x0B	RW	Bit[7:0]: zl_end
0x4052	KCOEF B MAN	0x01	RW	Bit[7:2]: Not used Bit[1:0]: kcoef_b_man[9:8]
0x4053	KCOEF B MAN	0x00	RW	Bit[7:0]: kcoef_b_man[7:0]
0x4054	KCOEF GB MAN	0x01	RW	Bit[7:2]: Not used Bit[1:0]: kcoef_gb_man[9:8]

table 6-15 BLC control registers (sheet 5 of 8)

address	register name	default value	R/W	description
0x4055	KCOEF GB MAN	0x00	RW	Bit[7:0]: kcoef_gb_man[7:0]
0x4056	KCOEF GR MAN	0x01	RW	Bit[7:2]: Not used Bit[1:0]: kcoef_gr_man[9:8]
0x4057	KCOEF GR MAN	0x00	RW	Bit[7:0]: kcoef_gr_man[7:0]
0x4058	KCOEF R MAN	0x01	RW	Bit[7:2]: Not used Bit[1:0]: kcoef_r_man[9:8]
0x4059	KCOEF R MAN	0x00	RW	Bit[7:0]: kcoef_r_man[7:0]
0x405A	BLC CTRL5A	0x30	RW	Bit[7:0]: r_l_dc_th_1_0
0x405B	BLC CTRL5B	0x18	RW	Bit[7:0]: r_l_dc_th_2_0
0x405C	BLC CTRL5C	0x00	RW	Bit[7:6]: Not used Bit[5:0]: avg_weight
0x405D	RND GAIN TH	0x00	RW	Bit[7:2]: Not used Bit[1:0]: rnd_gain_th[9:8] rnd_gain_th high 2 bits
0x405E	RND GAIN TH	0x00	RW	Bit[7:0]: rnd_gain_th[7:0] rnd_gain_th low 8 bits
0x405F	BLC CTRL5F	0x00	RW	Bit[7:0]: r_s_dc_th_1_0
0x4060	BLC CTRL60	0x00	RW	Bit[7:0]: r_s_dc_th_2_0
0x4061	THRES L	–	R	Bit[7:5]: Not used Bit[4:0]: thres_l[12:8]
0x4062	THRES L	–	R	Bit[7:0]: thres_l[7:0]
0x4063	THRES S	–	R	Bit[7:5]: Not used Bit[4:0]: thres_s[12:8]
0x4064	THRES S	–	R	Bit[7:0]: thres_s[7:0]
0x4065	ZERO LN NUM	0x00	RW	Bit[7:2]: Not used Bit[1:0]: zero_ln_num[9:8] Zero line number high 2 bits
0x4066	ZERO LN NUM	0x02	RW	Bit[7:0]: zero_ln_num[7:0] Zero line number low 8 bits
0x4067	COL WIN	0x18	RW	Bit[7:0]: col_win
0x4068	R COL LOW GAIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: r_col_low_gain[9:8] r_col_low_gain high 2 bits
0x4069	R COL LOW GAIN	0x20	RW	Bit[7:0]: r_col_low_gain[7:0] r_col_low_gain low 8 bits

table 6-15 BLC control registers (sheet 6 of 8)

address	register name	default value	R/W	description
0x406A	R COL HIGH GAIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: r_col_high_gain[9:8] r_col_high_gain high 2 bits
0x406B	R COL HIGH GAIN	0x40	RW	Bit[7:0]: r_col_high_gain[7:0] r_col_high_gain low 8 bits
0x406C	BLC CTRL6C	0x02	RW	Bit[7:5]: Not used Bit[4]: Calibration no mirror select Bit[3]: col_real_gain_sel Bit[2:0]: r_col_div_cnt
0x406D	BLC CTRL6D	0x00	RW	Bit[7:0]: mf_col_th
0x406E	R H SIZE MAN	0x00	RW	Bit[7:5]: Not used Bit[4:0]: r_h_size_man[12:8]
0x406F	R H SIZE MAN	0x00	RW	Bit[7:0]: r_h_size_man[7:0]
0x4070	BLC OFFSET0000	–	R	Bit[7:1]: Not used Bit[0]: blc_offset0000[24]
0x4071	BLC OFFSET0000	–	R	Bit[7:0]: blc_offset0000[23:16]
0x4072	BLC OFFSET0000	–	R	Bit[7:0]: blc_offset0000[15:8]
0x4073	BLC OFFSET0000	–	R	Bit[7:0]: blc_offset0000[7:0]
0x4074	BLC OFFSET0001	–	R	Bit[7:1]: Not used Bit[0]: blc_offset0001[24]
0x4075	BLC OFFSET0001	–	R	Bit[7:0]: blc_offset0001[23:16]
0x4076	BLC OFFSET0001	–	R	Bit[7:0]: blc_offset0001[15:8]
0x4077	BLC OFFSET0001	–	R	Bit[7:0]: blc_offset0001[7:0]
0x4078	BLC OFFSET0010	–	R	Bit[7:1]: Not used Bit[0]: blc_offset0010[24]
0x4079	BLC OFFSET0010	–	R	Bit[7:0]: blc_offset0010[23:16]
0x407A	BLC OFFSET0010	–	R	Bit[7:0]: blc_offset0010[15:8]
0x407B	BLC OFFSET0010	–	R	Bit[7:0]: blc_offset0010[7:0]
0x407C	BLC OFFSET0011	–	R	Bit[7:1]: Not used Bit[0]: blc_offset0011[24]
0x407D	BLC OFFSET0011	–	R	Bit[7:0]: blc_offset0011[23:16]
0x407E	BLC OFFSET0011	–	R	Bit[7:0]: blc_offset0011[15:8]
0x407F	BLC OFFSET0011	–	R	Bit[7:0]: blc_offset0011[7:0]
0x4080	BLC OFFSET0100	–	R	Bit[7:1]: Not used Bit[0]: blc_offset0100[24]

table 6-15 BLC control registers (sheet 7 of 8)

address	register name	default value	R/W	description
0x4081	BLC OFFSET0100	–	R	Bit[7:0]: blc_offset0100[23:16]
0x4082	BLC OFFSET0100	–	R	Bit[7:0]: blc_offset0100[15:8]
0x4083	BLC OFFSET0100	–	R	Bit[7:0]: blc_offset0100[7:0]
0x4084	BLC OFFSET0101	–	R	Bit[7:1]: Not used Bit[0]: blc_offset0101[24]
0x4085	BLC OFFSET0101	–	R	Bit[7:0]: blc_offset0101[23:16]
0x4086	BLC OFFSET0101	–	R	Bit[7:0]: blc_offset0101[15:8]
0x4087	BLC OFFSET0101	–	R	Bit[7:0]: blc_offset0101[7:0]
0x4088	BLC OFFSET0110	–	R	Bit[7:1]: Not used Bit[0]: blc_offset0110[24]
0x4089	BLC OFFSET0110	–	R	Bit[7:0]: blc_offset0110[23:16]
0x408A	BLC OFFSET0110	–	R	Bit[7:0]: blc_offset0110[15:8]
0x408B	BLC OFFSET0110	–	R	Bit[7:0]: blc_offset0110[7:0]
0x408C	BLC OFFSET0111	–	R	Bit[7:1]: Not used Bit[0]: blc_offset0111[24]
0x408D	BLC OFFSET0111	–	R	Bit[7:0]: blc_offset0111[23:16]
0x408E	BLC OFFSET0111	–	R	Bit[7:0]: blc_offset0111[15:8]
0x408F	BLC OFFSET0111	–	R	Bit[7:0]: blc_offset0111[7:0]
0x4090	CMP 0000	–	R	Bit[7:3]: Not used Bit[2:0]: cmp_0000[10:8]
0x4091	CMP 0000	–	R	Bit[7:0]: cmp_0000[7:0]
0x4092	CMP 0001	–	R	Bit[7:3]: Not used Bit[2:0]: cmp_0001[10:8]
0x4093	CMP 0001	–	R	Bit[7:0]: cmp_0001[7:0]
0x4094	CMP 0010	–	R	Bit[7:3]: Not used Bit[2:0]: cmp_0010[10:8]
0x4095	CMP 0010	–	R	Bit[7:0]: cmp_0010[7:0]
0x4096	CMP 0011	–	R	Bit[7:3]: Not used Bit[2:0]: cmp_0011[10:8]
0x4097	CMP 0011	–	R	Bit[7:0]: cmp_0011[7:0]
0x4098	CMP 0100	–	R	Bit[7:3]: Not used Bit[2:0]: cmp_0100[10:8]
0x4099	CMP 0100	–	R	Bit[7:0]: cmp_0100[7:0]

table 6-15 BLC control registers (sheet 8 of 8)

address	register name	default value	R/W	description
0x409A	CMP 0101	–	R	Bit[7:3]: Not used Bit[2:0]: cmp_0101[10:8]
0x409B	CMP 0101	–	R	Bit[7:0]: cmp_0101[7:0]
0x409C	CMP 0110	–	R	Bit[7:3]: Not used Bit[2:0]: cmp_0110[10:8]
0x409D	CMP 0110	–	R	Bit[7:0]: cmp_0110[7:0]
0x409E	CMP 0111	–	R	Bit[7:3]: Not used Bit[2:0]: cmp_0111[10:8]
0x409F	CMP 0111	–	R	Bit[7:0]: cmp_0111[7:0]
0x40C0	OFF CMP K0000	0x00	RW	Bit[7:4]: Not used Bit[3:0]: off_cmp_k0000[11:8]
0x40C1	OFF CMP K0000	0x00	RW	Bit[7:0]: off_cmp_k0000[7:0]
0x40C2	OFF CMP K0001	0x00	RW	Bit[7:4]: Not used Bit[3:0]: off_cmp_k0001[11:8]
0x40C3	OFF CMP K0001	0x00	RW	Bit[7:0]: off_cmp_k0001[7:0]
0x40C4	OFF CMP K0010	0x00	RW	Bit[7:4]: Not used Bit[3:0]: off_cmp_k0010[11:8]
0x40C5	OFF CMP K0010	0x00	RW	Bit[7:0]: off_cmp_k0010[7:0]
0x40C6	OFF CMP K0011	0x00	RW	Bit[7:4]: Not used Bit[3:0]: off_cmp_k0011[11:8]
0x40C7	OFF CMP K0011	0x00	RW	Bit[7:0]: off_cmp_k0011[7:0]
0x40C8	OFF CMP K0100	0x00	RW	Bit[7:4]: Not used Bit[3:0]: off_cmp_k0100[11:8]
0x40C9	OFF CMP K0100	0x00	RW	Bit[7:0]: off_cmp_k0100[7:0]
0x40CA	OFF CMP K0101	0x00	RW	Bit[7:4]: Not used Bit[3:0]: off_cmp_k0101[11:8]
0x40CB	OFF CMP K0101	0x00	RW	Bit[7:0]: off_cmp_k0101[7:0]
0x40CC	OFF CMP K0110	0x00	RW	Bit[7:4]: Not used Bit[3:0]: off_cmp_k0110[11:8]
0x40CD	OFF CMP K0110	0x00	RW	Bit[7:0]: off_cmp_k0110[7:0]
0x40CE	OFF CMP K0111	0x00	RW	Bit[7:4]: Not used Bit[3:0]: off_cmp_k0111[11:8]
0x40CF	OFF CMP K0111	0x00	RW	Bit[7:0]: off_cmp_k0111[7:0]

6.16 CADC sync [0x4500 - 0x4502, 0x4507]

table 6-16 CADC sync control registers

address	register name	default value	R/W	description
0x4500	CTRL	0x16	RW	Bit[7:0]: FIFO read delay
0x4501	R1	0x18	RW	Bit[7]: Hbin manual enable Bit[6]: hbin4_man Bit[5]: hbin2_man Bit[4]: PSRAM address Bit[3]: hbin_avg Bit[2]: Option to select odd pixels for skip4 Bit[1]: Option to select odd pixels for skip2 Bit[0]: srclk_inv
0x4502	R2	0x00	RW	Bit[7]: rblue_disable Bit[6:2]: Not used Bit[1]: Digital mono_hbin Bit[0]: rblue_reversed
0x4507	CTRL07	0x00	RW	Bit[7]: Hsub manual enable Bit[6]: quarter_sclk_man Bit[5]: half_sclk_man Bit[4:2]: Not used Bit[1:0]: hsub_man

6.17 VFIFO control [0x4600 - 0x4606, 0x4608]

table 6-17 VFIFO control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4600	R VFIFO READ START	0x01	RW	Bit[7:0]: r_vfifo_read_start[15:8] read_start size high byte
0x4601	R VFIFO READ START	0x04	RW	Bit[7:0]: r_vfifo_read_start[7:0] read_start size low byte
0x4602	R2	0x02	RW	Bit[7:4]: r_rm Bit[3]: r_test1 Bit[2]: bist_clk_sw Bit[1]: Frame reset enable Bit[0]: RAM bypass enable
0x4603	R3	0x00	RW	Bit[7:2]: Not used Bit[1]: sram_rme Bit[0]: man_start_mode

table 6-17 VFIFO control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4604	VFIFO MIRR CTRL	0x00	RW	Bit[7:4]: Not used Bit[3]: r_sram_hfq_mode 1 Use SRAM readout data after 2 clock cycles Bit[2]: r_fo_mirr_4x_fix_dis Bit[1]: mirror_fi_swap_dis Bit[0]: vfifo_mirror_dis
0x4605	R5	0x03	RW	Bit[7:5]: Not used Bit[4]: sram_always_on Bit[3:0]: Ready low tp number
0x4606	R6	–	R	Bit[7:4]: Not used Bit[3]: ram_full Bit[2]: ram_empty Bit[1]: fo_full Bit[0]: fo_empty
0x4608	NOT USED	–	–	Not Used

6.18 LVDS control [0x4700, 0x4702 - 0x4713]

table 6-18 LVDS control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4700	LVDS R0	0x2A	RW	Bit[7]: Two sync code enable in 8-lane mode Bit[6]: Sync code manual mode enable Bit[5]: Sync code enable when there is only 1 lane Bit[4]: lvds_pclk_inv Bit[3]: Channel ID enable in sync per lane mode Bit[2]: CCIR parameter Bit[1]: Sav first enable Bit[0]: Sync code mode 0: Split 1: Per lane
0x4702	LVDS DUMMY DATA0	0x00	RW	Bit[7:0]: lvds_dummy_data0[15:8] Dummy data0
0x4703	LVDS DUMMY DATA0	0x80	RW	Bit[7:0]: lvds_dummy_data0[7:0] Dummy data0
0x4704	LVDS DUMMY DATA1	0x00	RW	Bit[7:0]: lvds_dummy_data1[15:8] Dummy data1

table 6-18 LVDS control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4705	LVDS DUMMY DATA1	0x10	RW	Bit[7:0]: lvds_dummy_data1[7:0] Dummy data1
0x4706	LVDS R6	0xAA	RW	Bit[7:0]: lvds_r6 frame_start sync code in manual sync code mode
0x4707	LVDS R7	0x55	RW	Bit[7:0]: lvds_r7 frame_end sync code in manual sync code mode
0x4708	LVDS R8	0x99	RW	Bit[7:0]: lvds_r8 line_start sync code in manual sync code mode
0x4709	LVDS R9	0x66	RW	Bit[7:0]: lvds_r9 line_end sync code in manual sync code mode
0x470A	LVDS RA	0x08	RW	Bit[7:3]: Not used Bit[2]: r_hts_man_en Bit[1]: r_ln2_sel Bit[0]: r_chk_pcnt
0x470B	LVDS SLEEP CTRL	0x88	RW	Bit[7]: sleep_en Bit[4]: frame_rst_en Bit[3:0]: ln_end_dly
0x470C	LVDS BLK TIMES	0x00	RW	Bit[7:4]: Not used Bit[3:0]: lvds_blk_times[11:8] High 4 bits of r_blk_times
0x470D	LVDS BLK TIMES	0x02	RW	Bit[7:0]: lvds_blk_times[7:0] Low byte of r_blk_times
0x470E	LVDS HTS MAN	0x00	RW	Bit[7:0]: lvds_hts_man[15:8] hts_man high byte
0x470F	LVDS HTS MAN	0x00	RW	Bit[7:0]: lvds_hts_man[7:0] hts_man low byte
0x4710	LVDS CHID01	0x00	RW	Bit[7:4]: man_id_lane1 Bit[3:0]: man_id_lane0
0x4711	LVDS CHID23	0x00	RW	Bit[7:4]: man_id_lane3 Bit[3:0]: man_id_lane2
0x4712	LVDS CHID45	0x00	RW	Bit[7:4]: man_id_lane5 Bit[3:0]: man_id_lane4
0x4713	LVDS CHID67	0x00	RW	Bit[7:4]: man_id_lane7 Bit[3:0]: man_id_lane6

6.19 MIPI control [0x4800 - 0x4809, 0x4810 - 0x483D, 0x484A - 0x48A1]

table 6-19 MIPI control registers (sheet 1 of 14)

address	register name	default value	R/W	description
0x4800	MIPI CTRL00	0x04	RW	Bit[7:6]: Not used Bit[5]: gate_sc_en 0: Clock lane is free running 1: Gate clock lane when there is no packet to transmit Bit[4]: line_sync_en 0: Do not send line short packet for each line 1: Send line short packet for each line Bit[3:0]: Not used
0x4801	MIPI CTRL01	0x00	RW	Bit[7]: Not used Bit[6]: spkt_dt_sel 0: Not used 1: Use dt_spkt as short packet data Bit[5]: first_bit Change clk_lane first bit 0: Output 8'h5 1: Output 8'hAA Bit[4]: Reserved Bit[3:2]: Not used Bit[1]: LPX_select for pclk domain 0: Auto calculate t_lpx_p Unit pclk2x cycle 1: Use lpx_p_min[7:0] Bit[0]: Not used

table 6-19 MIPI control registers (sheet 2 of 14)

address	register name	default value	R/W	description
0x4802	MIPI CTRL02	0x00	RW	Bit[7]: hs_prepare_sel 0: Auto calculate T_hs_prepare, unit pclk2x 1: Use hs_prepare_min_o[7:0] Bit[6]: clk_prepare_sel 0: Auto calculate T_clk_prepare, unit pclk2x 1: Use clk_prepare_min_o[7:0] Bit[5]: clk_post_sel 0: Auto calculate T_clk_post, unit pclk2x 1: Use clk_post_min_o[7:0] Bit[4]: clk_trail_sel 0: Auto calculate T_clk_trail, unit pclk2x 1: Use clk_trail_min_o[7:0] Bit[3]: hs_exit_sel 0: Auto calculate T_hs_exit, unit pclk2x 1: Use hs_exit_min_o[7:0] Bit[2]: hs_zero_sel 0: Auto calculate T_hs_zero, unit pclk2x 1: Use hs_zero_min_o[7:0] Bit[1]: hs_trail_sel 0: Auto calculate T_hs_trail, unit pclk2x 1: Use hs_trail_min_o[7:0] Bit[0]: clk_zero_sel 0: Auto calculate T_clk_zero, unit pclk2x 1: Use clk_zero_min_o[7:0]
0x4803	MIPI CTRL03	0x00	RW	Bit[7:4]: Not used Bit[3]: manu_ofset_o t_perio manual offset SMIA Bit[2]: r_manu_half2one t_period half to 1 SMIA Bit[1:0]: Not used
0x4804	MIPI CTRL04	0x44	RW	Bit[7:4]: man_lane_num Bit[3]: lane_num_manual_enable Bit[2]: lane4_6b_en 0: Not used 1: Supports 4, 7, and 8-lane 6-bit Bit[1:0]: Not used
0x4805	MIPI CTRL05	0x00	RW	Bit[7:4]: Not used Bit[3]: lpda_retim_manu_o Bit[2]: lpda_retim_sel_o 0: Not used 1: Manual Bit[1]: lpck_retim_manu_o Bit[0]: lpck_retim_sel_o 0: Not used 1: Manual

table 6-19 MIPI control registers (sheet 3 of 14)

address	register name	default value	R/W	description
0x4806	MIPI CTRL06	0x00	RW	Bit[7:5]: Not used Bit[4]: pu_mark_en_o Power up mark1 enable Bit[3]: mipi_remot_rst Bit[2]: mipi_susp Bit[1]: smia_lane_ch_en Bit[0]: tx_lsb_first 0: High bit first 1: Low power TX low bit first
0x4807	MIPI CTRL07	0x03	RW	Bit[7:4]: Not used Bit[3:0]: sw_t_lpx ul_tx t_lpx
0x4808	MIPI CTRL08	0x18	RW	Bit[7:0]: wkup_dly Mark1 wakeup delay/2^10
0x4809	MIPI CTRL09	0x2B	RW	Bit[7:6]: Not used Bit[5:0]: dt_man1 Manual data type
0x4810	FCNT MAX	0xFF	RW	Bit[7:0]: fcnt_max[15:8] High byte of max frame counter of frame sync short packet
0x4811	FCNT MAX	0xFF	RW	Bit[7:0]: fcnt_max[7:0] Low byte of max frame counter of frame sync short packet
0x4812	MIPI CTRL12	0x2B	RW	Bit[7:6]: Not used Bit[5:0]: dt_man2 Manual data type
0x4813	MIPI CTRL13	0x90	RW	Bit[7:6]: Virtual channel2 of MIPI Bit[5:4]: Virtual channel1 of MIPI Bit[3]: Not used Bit[2]: vc_sel Input VC or register VC Bit[1:0]: Virtual channel0 of MIPI
0x4814	MIPI CTRL14	0x2B	RW	Bit[7]: Not used Bit[6]: lpkt_dt_sel 0: Use mipi_dt 1: Use dt_man_o as long packet data Bit[5:0]: dt_man Manual data type
0x4815~ 0x4816	RSVD	–	–	Reserved

table 6-19 MIPI control registers (sheet 4 of 14)

address	register name	default value	R/W	description
0x4817	YUV420 FUN	0x04	RW	Bit[7:2]: Not used Bit[1]: yuv420_2x YUV420 2x in odd line, lcnt[0]=1 0: Not used 1: Use emb_dt as data in first emb_line_nu Bit[0]: yuv420_en
0x4818	HS ZERO MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: hs_zero_min[9:8] High byte of minimum value of hs_zero, unit ns
0x4819	HS ZERO MIN	0x70	RW	Bit[7:0]: hs_zero_min[7:0] Low byte of minimum value of hs_zero hs_zero_real = hs_zero_min_o + Tui*ui_hs_zero_min_o
0x481A	HS TRAIL MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: hs_trail_min[9:8] High byte of minimum value of hs_trail, unit ns
0x481B	HS TRAIL MIN	0x3C	RW	Bit[7:0]: hs_trail_min[7:0] Low byte of minimum value of hs_trail hs_trail_real = hs_trail_min_o + Tui*ui_hs_trail_min_o
0x481C	CLK ZERO MIN	0x01	RW	Bit[7:2]: Not used Bit[1:0]: clk_zero_min[9:8] High byte of minimum value of clk_zero, unit ns
0x481D	CLK ZERO MIN	0x2C	RW	Bit[7:0]: clk_zero_min[7:0] Low byte of minimum value of clk_zero clk_zero_real = clk_zero_min_o + Tui*ui_clk_zero_min_o
0x481E	CLK PREPARE MAX	0x5F	RW	Bit[7:0]: clk_prepare_max[7:0] Maximum value of clk_prepare, unit ns
0x481F	CLK PREPARE MIN	0x26	RW	Bit[7:0]: clk_prepare_min[7:0] Minimum value of clk_prepare clk_prepare_real = clk_prepare_min_o + Tui*ui_clk_prepare_min_o
0x4820	CLK POST MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: clk_post_min[9:8] High byte of minimum value of clk_post, unit ns

table 6-19 MIPI control registers (sheet 5 of 14)

address	register name	default value	R/W	description
0x4821	CLK POST MIN	0x3C	RW	Bit[7:0]: clk_post_min[7:0] Low byte of minimum value of clk_post clk_post_real = clk_post_min_o + Tui*ui_clk_post_min_o
0x4822	CLK TRAIL MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: clk_trail_min[9:8] High byte of minimum value of clk_trail, unit ns
0x4823	CLK TRAIL MIN	0x3C	RW	Bit[7:0]: clk_trail_min[7:0] Low byte of minimum value of clk_trail clk_trail_real = clk_trail_min_o + Tui*ui_clk_trail_min_o
0x4824	LPX P MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: lpx_p_min[9:8] High byte of minimum value of lpx_p, unit ns
0x4825	LPX P MIN	0x32	RW	Bit[7:0]: lpx_p_min[7:0] Low byte of minimum value of lpx_p lpx_p_real = lpx_p_min_o + Tui*ui_lpx_p_min_o
0x4826	HS PREPARE MIN	0x32	RW	Bit[7:0]: hs_prepare_min[7:0] Minimum value of hs_prepare, unit ns
0x4827	HS PREPARE MAX	0x55	RW	Bit[7:0]: hs_prepare_max[7:0] Maximum value of hs_prepare hs_prepare_real = hs_prepare_max_o + Tui*ui_hs_prepare_max_o
0x4828	HS EXIT MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: hs_exit_min[9:8] High byte of minimum value of hs_exit, unit ns
0x4829	HS EXIT MIN	0x64	RW	Bit[7:0]: hs_exit_min[7:0] Low byte of minimum value of hs_exit hs_exit_real = hs_exit_min_o + Tui*ui_hs_exit_min_o
0x482A	UI HS ZERO MIN	0x06	RW	Bit[7:6]: Not used Bit[5:0]: ui_hs_zero_min[5:0] Minimum UI value of hs_zero, unit UI
0x482B	UI HS TRAIL MIN	0x04	RW	Bit[7:6]: Not used Bit[5:0]: ui_hs_trail_min[5:0] Minimum UI value of hs_trail, unit UI

table 6-19 MIPI control registers (sheet 6 of 14)

address	register name	default value	R/W	description
0x482C	UI CLK ZERO MIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: ui_clk_zero_min[5:0] Minimum UI value of clk_zero, unit UI
0x482D	UI CLK PREPARE	0x00	RW	Bit[7:4]: ui_clk_prepare_max Maximum UI value of clk_prepare Bit[3:0]: ui_clk_prepare_min Minimum UI value of clk_prepare, unit UI
0x482E	UI CLK POST MIN	0x34	RW	Bit[7:6]: Not used Bit[5:0]: ui_clk_post_min[5:0] Minimum UI value of clk_post, unit UI
0x482F	UI CLK TRAIL MIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: ui_clk_trail_min[5:0] Minimum UI value of clk_trail, unit UI
0x4830	UI LPX P MIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: ui_lpx_p_min[5:0] Minimum UI value of lpx_p(pclk2x domain), unit UI
0x4831	UI HS PREPARE	0x64	RW	Bit[7:4]: ui_hs_prepare_max Maximum UI value of hs_prepare Bit[3:0]: ui_hs_prepare_min Minimum UI value of hs_prepare, unit UI
0x4832	UI HS EXIT MIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: ui_hs_exit_min[5:0] Minimum UI value of hs_exit, unit UI
0x4833	MIPI PKT STAR SIZE	0x18	RW	Bit[7:0]: mipi_pkt_star_size
0x4834~ 0x4835	RSVD	–	–	Reserved
0x4836	GLB MODE SEL	0x00	RW	Bit[7:1]: Not used Bit[0]: smia_cal_en 0: Use period to calculate 1: Use SMIA bitrate to calculate
0x4837	PCLK PERIOD	0x08	RW	Bit[7:0]: pclk_period[7:0] Period of pclk2x, pclk_div=1, and 1-bit decimal

table 6-19 MIPI control registers (sheet 7 of 14)

address	register name	default value	R/W	description
0x4838	MIPI LP GPIO0	0x00	RW	Bit[7]: lp_sel0 0: Auto generate mipi_lp_dir0_o 1: Use lp_dir_man0 to be mipi_lp_dir0_o Bit[6]: lp_dir_man0 0: Input 1: Output Bit[5]: lp_p0_o Bit[4]: lp_n0_o Bit[3]: lp_sel1 0: Auto generate mipi_lp_dir1_o 1: Use lp_dir_man1 to be mipi_lp_dir1_o Bit[2]: lp_dir_man1 0: Input 1: Output Bit[1]: lp_p1_o Bit[0]: lp_n1_o
0x4839	MIPI LP GPIO1	0x00	RW	Bit[7]: lp_sel2 0: Auto generate mipi_lp_dir2_o 1: Use lp_dir_man2 to be mipi_lp_dir2_o Bit[6]: lp_dir_man2 0: Input 1: Output Bit[5]: lp_p2_o Bit[4]: lp_n2_o Bit[3]: lp_sel3 0: Auto generate mipi_lp_dir3_o 1: Use lp_dir_man3 to be mipi_lp_dir3_o Bit[2]: lp_dir_man3 0: Input 1: Output Bit[1]: lp_p3_o Bit[0]: lp_n3_o
0x483A	MIPI LP GPIO2	0x00	RW	Bit[7]: lp_sel4 0: Auto generate mipi_lp_dir4_o 1: Use lp_dir_man4 to be mipi_lp_dir4_o Bit[6]: lp_dir_man4 0: Input 1: Output Bit[5]: lp_p4_o Bit[4]: lp_n4_o Bit[3]: lp_sel5 0: Auto generate mipi_lp_dir5_o 1: Use lp_dir_man5 to be mipi_lp_dir5_o Bit[2]: lp_dir_man5 0: Input 1: Output Bit[1]: lp_p5_o[0] lp_n5_o Bit[0]: Not used

table 6-19 MIPI control registers (sheet 8 of 14)

address	register name	default value	R/W	description
0x483B	MIPI LP GPIO3	0x00	RW	Bit[7]: lp_sel6 0: Auto generate mipi_lp_dir6_o 1: Use lp_dir_man6 to be mipi_lp_dir6_o Bit[6]: lp_dir_man6 0: Input 1: Output Bit[5]: lp_p6_o Bit[4]: lp_n6_o Bit[3]: lp_sel7 0: Auto generate mipi_lp_dir7_o 1: Use lp_dir_man7 to be mipi_lp_dir7_o Bit[2]: lp_dir_man7 0: Input 1: Output Bit[1]: lp_p7_o Bit[0]: lp_n7_o
0x483C	MIPI CTRL3C	0x02	RW	Bit[7:4]: Not used Bit[3:0]: t_clk_pre Unit pclk2x cycle
0x483D	MIPI LP GPIO4	0x00	RW	Bit[7]: lp_ck_sel0 0: Auto generate mipi_ck_lp_dir0_o 1: Use lp_ck_dir_man0 to be mipi_ck_lp_dir0_o Bit[6]: lp_ck_dir_man0 0: Input 1: Output Bit[5]: lp_ck_p0_o Bit[4]: lp_ck_n0_o Bit[3]: lp_ck_sel1 0: Auto generate mipi_ck_lp_dir1_o 1: Use lp_ck_dir_man1 to be mipi_ck_lp_dir1_o Bit[2]: lp_ck_dir_man1 0: Input 1: Output Bit[1]: lp_ck_p1_o Bit[0]: lp_ck_n1_o
0x484A	SEL MIPI CTRL4A	0x3F	RW	Bit[7:6]: Not used Bit[5]: slp_lp_pon_man_o Set for power up Bit[4]: slp_lp_pon_da Bit[3]: slp_lp_pon_ck Bit[2]: mipi_slp_man_st MIPI bus status manual control enable in sleep mode Bit[1]: clk_lane_state Bit[0]: data_lane_state

table 6-19 MIPI control registers (sheet 9 of 14)

address	register name	default value	R/W	description
0x484B	SMIA OPTION	0x01	RW	Bit[7:3]: Not used Bit[2]: line_st_sel_o 0: Line starts after HREF 1: Line starts after fifo_st Bit[1]: clk_start_sel_o 0: Clock starts after SOF 1: Clock starts after reset Bit[0]: sof_sel_o 0: Frame starts after HREF starts 1: Frame starts after SOF
0x484C	SEL MIPI CTRL4C	0x03	RW	Bit[7]: Not used Bit[6]: smia_fcnt_i select Bit[5]: prbs_enable Bit[4]: hs_test_only MIPI high speed only test mode enable Bit[3]: set_frame_cnt_0 Set frame count to inactive mode (keep 0) Bit[2:0]: Not used
0x484D	TEST PATTERN DATA	0xB6	RW	Bit[7:0]: test_patten_data[7:0] Data lane test pattern register
0x484E	FE DLY	0x10	RW	Bit[7:0]: r_fe_dly_o Last packet to frame end delay/2
0x484F	TEST PATTERN CK DATA	0x55	RW	Bit[7:0]: clk_test_patten_reg
0x4850	R DE SKEW	0x5C	RW	Bit[7:2]: r_de_skew_dly Bit[1]: r_de_skew_manu1 Bit[0]: r_de_skew_manu0
0x4851	R SKEW COMMOND	0x01	RW	Bit[7:0]: r_skew_commond
0x4852	R DE SKEW DLY	0x27	RW	Bit[7:0]: r_de_skew_dly
0x4853	R SW RDY	–	R	Bit[7:4]: MIPI state Bit[3]: Not used Bit[2]: frame_act Bit[1]: mipi_busy Bit[0]: SW ready
0x4854	R SKEW CNT START0	0x1E	RW	Bit[7:0]: r_skew_cnt_start0
0x4855	R SKEW CNT START1 L	0x90	RW	Bit[7:0]: r_skew_cnt_start1_l

table 6-19 MIPI control registers (sheet 10 of 14)

address	register name	default value	R/W	description
0x4856	R MODE	0x5C	RW	Bit[7:5]: Not used Bit[4]: r_skew_cnt_start1_h Bit[3]: r_trail_same_start Bit[2]: r_mode12_tradat Bit[1:0]: r_mode
0x4857	R MODE12 HSDAT	0x55	RW	Bit[7:0]: r_mode12_hsdat
0x4858	R MODE2 SYNCDAT1	0xFF	RW	Bit[7:0]: r_mode2_syncdat1
0x4859	R MODE2 SYNCDAT2	0xFF	RW	Bit[7:0]: r_mode2_syncdat2
0x485A	R MODE3 PREADAT1	0xFF	RW	Bit[7:0]: r_mode3_preadat1
0x485B	R MODE3 PREADAT2	0x3F	RW	Bit[7:0]: r_mode3_preadat2
0x485C	R MODE3 PREADAT SEL	0x2A	RW	Bit[7:0]: r_mode3_preadat_sel
0x485D	R PROGDAT1	0x66	RW	Bit[7:0]: r_progdat1
0x485E	R PROGDAT2	0x99	RW	Bit[7:0]: r_progdat2
0x485F	R PROGDAT3	0x88	RW	Bit[7:0]: r_progdat3
0x4860	R PROGDAT4	0xAA	RW	Bit[7:0]: r_progdat4
0x4861	R MODE3 PROGDAT SEL L	0xAA	RW	Bit[7:0]: r_mode3_progdat_sel_l
0x4862	R MODE3 PROGDAT SEL H	0x0A	RW	Bit[7:0]: r_mode3_progdat_sel_h
0x4863	R MODE3 SYNCDAT1	0x84	RW	Bit[7:0]: r_mode3_syncdat1
0x4864	R MODE3 SYNCDAT2	0x36	RW	Bit[7:0]: r_mode3_syncdat2
0x4865	R MODE3 SYNCDAT SEL	0x2A	RW	Bit[7:0]: r_mode3_syncdat_sel
0x4866	TPREBEGIN	0x01	RW	Bit[7:0]: Tprebegin
0x4867	TPROGSEQ	0x02	RW	Bit[7:0]: Tprogseq
0x4868	TPREEND	0x01	RW	Bit[7:0]: Tpreend

table 6-19 MIPI control registers (sheet 11 of 14)

address	register name	default value	R/W	description
0x4869	MODE3 CTR	0x18	RW	Bit[7:5]: Not used Bit[4]: Mode3 PH order select Bit[3]: eot_same Bit[2]: crc_mode3_sel Bit[1]: mode3_tradat Bit[0]: mode3_trail_man
0x486A	R MODE3 PH SEL	0xAA	RW	Bit[7:0]: r_mode3_ph_sel
0x486B	R MODE3 RESERDAT	0x00	RW	Bit[7:0]: r_mode3_reserdat
0x486C	R MODE3 ESC DAT1	0x84	RW	Bit[7:0]: r_mode3_esc_dat1
0x486D	R MODE3 ESC DAT2	0x36	RW	Bit[7:0]: r_mode3_esc_dat2
0x486E	MIPI CTRL6E	0x03	RW	Bit[7:4]: Not used Bit[3]: Data type enable Bit[2]: Virtual channel enable Bit[1]: clk_trail_data Bit[0]: mode3_cphy
0x486F	R MODE12 CLK DATA	0x55	RW	Bit[7:0]: r_mode12_clk_data
0x4870	MIPI CTRL70	0x48	RW	Bit[7]: Not used Bit[6]: r_skew_man_sel Bit[5:3]: r_lane1_swap Bit[2:0]: r_lane0_swap
0x4871	MIPI CTRL71	0x1A	RW	Bit[7:6]: Not used Bit[5:3]: r_lane3_swap Bit[2:0]: r_lane2_swap
0x4872	MIPI CTRL72	0x2C	RW	Bit[7]: r_slp_change_en Bit[6]: r_pkt_slp_en Bit[5:3]: r_lane5_swap Bit[2:0]: r_lane4_swap
0x4873	MIPI CTRL73	0x02	RW	Bit[7:4]: Not used Bit[3]: r_clk_start_early Bit[2:0]: r_mode2_sync_cnt
0x4874	R MODE2 SYNCDAT3	0xFF	RW	Bit[7:0]: r_mode2_syncdat3
0x4875	R LANE1 START	0xF0	RW	Bit[7:0]: r_lane1_start
0x4876	R CPHY MANUAL	0x08	RW	Bit[7:0]: r_cphy_manual

table 6-19 MIPI control registers (sheet 12 of 14)

address	register name	default value	R/W	description
0x4877	R DUMMY DATA15	0x00	RW	Bit[7:0]: r_dummy_data15
0x4878	R DUMMY DATA14	0x00	RW	Bit[7:0]: r_dummy_data14
0x4879	R DUMMY DATA13	0xE7	RW	Bit[7:0]: r_dummy_data13
0x487A	R DUMMY DATA12	0x24	RW	Bit[7:0]: r_dummy_data12
0x487B	R DUMMY DATA11	0x01	RW	Bit[7:0]: r_dummy_data11
0x487C	R DUMMY DATA10	0x00	RW	Bit[7:0]: r_dummy_data10
0x487D	R DUMMY DATA9	0x34	RW	Bit[7:0]: r_dummy_data9
0x487E	R DUMMY DATA8	0x00	RW	Bit[7:0]: r_dummy_data8
0x487F	R DUMMY DATA7	0x84	RW	Bit[7:0]: r_dummy_data7
0x4880	R DUMMY DATA6	0x36	RW	Bit[7:0]: r_dummy_data6
0x4881	R DUMMY DATA5	0x00	RW	Bit[7:0]: r_dummy_data5
0x4882	R DUMMY DATA4	0x00	RW	Bit[7:0]: r_dummy_data4
0x4883	R DUMMY DATA3	0xE7	RW	Bit[7:0]: r_dummy_data3
0x4884	R DUMMY DATA2	0x24	RW	Bit[7:0]: r_dummy_data2
0x4885	R DUMMY DATA1	0x01	RW	Bit[7:0]: r_dummy_data1
0x4886	R DUMMY DATA0	0x00	RW	Bit[7:0]: r_dummy_data0
0x4887	R MODE3 PROGDAT5	0x00	RW	Bit[7:0]: r_mode3_progdat5
0x4888	R MODE3 PROGDAT6	0x00	RW	Bit[7:0]: r_mode3_progdat6
0x4889	R MODE3 PROGDAT7	0x55	RW	Bit[7:0]: r_mode3_progdat7

table 6-19 MIPI control registers (sheet 13 of 14)

address	register name	default value	R/W	description
0x488A	R MODE3 PROGDAT8	0x15	RW	Bit[7:0]: r_mode3_progdat8
0x488B	R MODE3 PROGDAT9	0xAA	RW	Bit[7:0]: r_mode3_progdat9
0x488C	R MODE3 PROGDAT10	0x2A	RW	Bit[7:0]: r_mode3_progdat10
0x488D	R MODE3 PROGDAT11	0xFF	RW	Bit[7:0]: r_mode3_progdat11
0x488E	R MODE3 PROGDAT12	0x3F	RW	Bit[7:0]: r_mode3_progdat12
0x488F	R MODE3 PROGDAT13	0x00	RW	Bit[7:0]: r_mode3_progdat13
0x4890	R MODE3 PROGDAT14	0x00	RW	Bit[7:0]: r_mode3_progdat14
0x4891	R MODE3 PROGDAT15	0x55	RW	Bit[7:0]: r_mode3_progdat15
0x4892	R MODE3 PROGDAT16	0x15	RW	Bit[7:0]: r_mode3_progdat16
0x4893	R MODE3 PROGDAT17	0xAA	RW	Bit[7:0]: r_mode3_progdat17
0x4894	R MODE3 PROGDAT18	0x2A	RW	Bit[7:0]: r_mode3_progdat18
0x4895	R MODE3 PROGDAT19	0xFF	RW	Bit[7:0]: r_mode3_progdat19
0x4896	R MODE3 PROGDAT20	0x3F	RW	Bit[7:0]: r_mode3_progdat20
0x4897	R MODE3 PROGDAT21	0x00	RW	Bit[7:0]: r_mode3_progdat21
0x4898	R MODE3 PROGDAT22	0x00	RW	Bit[7:0]: r_mode3_progdat22
0x4899	R MODE3 PROGDAT23	0x55	RW	Bit[7:0]: r_mode3_progdat23
0x489A	R MODE3 PROGDAT24	0x15	RW	Bit[7:0]: r_mode3_progdat24
0x489B	R MODE3 PROGDAT25	0xAA	RW	Bit[7:0]: r_mode3_progdat25
0x489C	R MODE3 PROGDAT26	0x2A	RW	Bit[7:0]: r_mode3_progdat26

table 6-19 MIPI control registers (sheet 14 of 14)

address	register name	default value	R/W	description
0x489D	R MODE3 PROGDAT27	0xFF	RW	Bit[7:0]: r_mode3_progdat27
0x489E	R MODE3 PROGDAT28	0x3F	RW	Bit[7:0]: r_mode3_progdat28
0x489F	R MODE2 SYNCDAT4	0xFF	RW	Bit[7:0]: r_mode2_syncdat4
0x48A0	R MODE2 SYNCDAT5	0xFF	RW	Bit[7:0]: r_mode2_syncdat5
0x48A1	R MODE2 SYNCDAT6	0xFF	RW	Bit[7:0]: r_mode2_syncdat6

6.20 ISPFC [0x4900 - 0x4903]

table 6-20 ISPFC control registers

address	register name	default value	R/W	description
0x4900	FC REG0	0x08	RW	Bit[7:4]: Reserved Bit[3]: sof_after_line0 Bit[2]: fcnt_eof_sel Bit[1]: fcnt_mask_dis Bit[0]: fcnt_reset
0x4901	FC REG1	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: frame_on_number
0x4902	FC REG2	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: frame_off_number
0x4903	FC REG3	0x80	RW	Bit[7]: zero_line_mask_dis Bit[6]: rblue_mask_dis Bit[5]: data_mask_dis Bit[4]: valid_mask_dis Bit[3]: href_mask_dis Bit[2]: eof_mask_dis Bit[1]: sof_mask_dis Bit[0]: all_mask_dis

6.21 ULPM interrupt [0x4A00 - 0x4A10, 0x4A20 - 0x4A2B]

table 6-21 ULPM interrupt registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4A00	ULPM CTRL00	0xA4	RW	Bit[7]: Debug mode (always set to 1'b0) Bit[6]: Use average method to do ULPM 1: Average mode Bit[5]: r_opt Bit[3]: r_ratio_man_en Bit[1]: r_pol_inv_en Bit[0]: r_abs_en 0: Use other method 1: Use absolute value to make decision
0x4A01	ULPM CTRL01	0x0A	RW	Bit[7:4]: Reserved Bit[3]: gpio_auto_clear_en Bit[2]: gpio_host_clear_en Bit[1]: reg_auto_clear_en Bit[0]: reg_host_clear_en
0x4A02~ 0x4A03	RSVD	–	–	Reserved
0x4A04	ULPM CTRL04	0x00	RW	Bit[7:3]: Reserved Bit[2:0]: r_diff_ratio0
0x4A05	ULPM CTRL05	0x00	RW	Bit[7:4]: r_diff_ratio1 Bit[3:0]: r_diff_ratio2
0x4A06	ULPM CTRL06	0x04	RW	Bit[7:4]: Reserved Bit[3:0]: Trigger length
0x4A07	ULPM CTRL07	0x10	RW	Bit[7:6]: Reserved Bit[5:0]: r_no_intr_threshold
0x4A08	ULPM CTRL08	0xF0	RW	Bit[7:0]: r_threshold_high[15:8]
0x4A09	ULPM CTRL09	0x00	RW	Bit[7:0]: r_threshold_high[7:0]
0x4A0A	ULPM CTRL0A	0x00	RW	Bit[7:0]: r_threshold_low[15:8]
0x4A0B	ULPM CTRL0B	0x64	RW	Bit[7:0]: r_threshold_low[7:0]
0x4A0C	ULPM CTRL0C	0x80	RW	Bit[7:0]: r_ratio_manual[15:8]
0x4A0D	ULPM CTRL0D	0x00	RW	Bit[7:0]: r_ratio_manual[7:0]
0x4A0E~ 0x4A0F	RSVD	–	–	Reserved

table 6-21 ULPM interrupt registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4A10	ULPM CTRL10	–	W	Bit[7:2]: Reserved Bit[1]: Host register clear This bit is always kept high until it is written to 0. After writing to zero operation is finished, this bit will return to high state. 0: reg_host_clear Bit[0]: Host GPIO clear 0: gpio_host_clear
0x4A20	ULPM CTRL20	–	R	Bit[2]: Interrupt flag combination of interrupt 1 and interrupt 2 flags Bit[1]: Interrupt 2 flag Bit[0]: Interrupt 1 flag
0x4A21	ULPM CTRL21	–	R	Bit[7:0]: Current gain value
0x4A22	ULPM CTRL22	–	R	Bit[7:0]: Current exposure[15:8]
0x4A23	ULPM CTRL23	–	R	Bit[7:0]: Current exposure[7:0]
0x4A24	ULPM CTRL24	–	R	Bit[7:2]: Reserved Bit[1:0]: avg_y[9:8]
0x4A25	ULPM CTRL25	–	R	Bit[7:0]: avg_y[7:0]
0x4A26	ULPM CTRL26	–	R	Bit[7:2]: Reserved Bit[1:0]: avg_r[9:8]
0x4A27	ULPM CTRL27	–	R	Bit[7:0]: avg_r[7:0]
0x4A28	ULPM CTRL28	–	R	Bit[7:2]: Reserved Bit[1:0]: avg_g[9:8]
0x4A29	ULPM CTRL29	–	R	Bit[7:0]: avg_g[7:0]
0x4A2A	ULPM CTRL2A	–	R	Bit[7:2]: Reserved Bit[1:0]: avg_b[9:8]
0x4A2B	ULPM CTRL2B	–	R	Bit[7:0]: avg_b[7:0]

6.22 AVG [0x4B00 - 0x4B01, 0x4B03 - 0x4B05, 0x4B07 - 0x4B17]

table 6-22 AVG (sheet 1 of 2)

address	register name	default value	R/W	description
0x4B00	AVG CTRL00	0x00	RW	Bit[7:1]: Not used Bit[0]: x_start_avg[8] AVG sub-window horizontal start position
0x4B01	AVG CTRL01	0x00	RW	Bit[7:0]: x_start_avg[7:0] AVG sub-window horizontal start position
0x4B03	AVG CTRL03	0x00	RW	Bit[7:0]: y_start_avg[7:0] AVG sub-window vertical start position
0x4B04	AVG CTRL04	0x01	RW	Bit[7:1]: Not used Bit[0]: window_width_avg[8] Sub-window width
0x4B05	AVG CTRL05	0x40	RW	Bit[7:0]: window_width_avg[7:0] Sub-window width
0x4B07	AVG CTRL07	0xF0	RW	Bit[7:0]: window_height_avg[7:0] Sub-window height
0x4B08	AVG CTRL08	0x11	RW	Bit[7:4]: Weight01 Weight of zone01 Bit[3:0]: Weight00 Weight of zone00
0x4B09	AVG CTRL09	0x11	RW	Bit[7:4]: Weight03 Weight of zone03 Bit[3:0]: Weight02 Weight of zone02
0x4B0A	AVG CTRL0A	0x11	RW	Bit[7:4]: Weight5 Weight of zone5 Bit[3:0]: Weight4 Weight of zone4
0x4B0B	AVG CTRL0B	0x11	RW	Bit[7:4]: Weight7 Weight of zone7 Bit[3:0]: Weight6 Weight of zone6
0x4B0C	AVG CTRL0C	0x11	RW	Bit[7:4]: Weight9 Weight of zone 9 Bit[3:0]: Weight8 Weight of zone8

table 6-22 AVG (sheet 2 of 2)

address	register name	default value	R/W	description
0x4B0D	AVG CTRL0D	0x11	RW	Bit[7:4]: Weight11 Weight of zone11 Bit[3:0]: Weight10 Weight of zone10
0x4B0E	AVG CTRL0E	0x11	RW	Bit[7:4]: Weight13 Weight of zone13 Bit[3:0]: Weight12 Weight of zone12
0x4B0F	AVG CTRL0F	0x11	RW	Bit[7:4]: Weight15 Weight of zone15 Bit[3:0]: Weight14 Weight of zone14
0x4B10	AVG CTRL10	0x12	RW	Bit[7:5]: Not used Bit[4:3]: px_order manual Bit[2]: px_order manual enable Bit[1]: avg_opt Bit[0]: avg_man_en
0x4B11	AVG RO11	–	R	Bit[7:0]: weight-sum Sum of weight
0x4B12	AVG RO12	–	R	Bit[7:1]: Reserved Bit[0]: Average calculated indicating signal for SCCB read
0x4B13	AVG RO13	–	R	Bit[7:0]: High 8 bits of whole image's average output
0x4B14	AVG G	–	R	Average G Channel Read Out
0x4B15	AVG R	–	R	Average R Channel Read Out
0x4B16	AVG Y	–	R	Average Y Channel Read Out
0x4B17	DEBUG MODE	–	–	Debug Mode

6.23 temperature sensor control [0x4D00 - 0x4D14]

table 6-23 temperature sensor control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4D00~ 0x4D0F	TPM_CTRL_ REG	–	–	Temperature Sensor Control Registers

table 6-23 temperature sensor control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4D10	TPM_CTRL_10	0x00	RW	Bit[7:0]: r_tpm_min
0x4D11	TPM_CTRL_11	0xFF	RW	Bit[7:0]: r_tpm_max
0x4D12	TPM_CTRL_12	–	W	Writing 0x4D12[0] to '1' Will Trigger Temperature Calculating, Then 0x4D12 and 0x4D13 Will Be Latched Temperature Value
0x4D13	TPM_CTRL_13	–	R	Latched Temperature Value, Integer Part
0x4D14	TPM_CTRL_14	–	R	Latched Temperature Value, Decimal Part

6.24 ISP control [0x5000 - 0x5024]

table 6-24 ISP control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5000	ISP CTRL 0	0xA9	RW	Bit[7]: awb_en Bit[6:4]: Not used Bit[3]: otp_en Bit[2:1]: Not used Bit[0]: isp_en
0x5001	ISP CTRL 1	0x09	RW	Bit[7]: Not used Bit[6]: win_en Bit[5:2]: Not used Bit[1]: dpc_en Bit[0]: Not used
0x5002	ISP CTRL 2	0x00	RW	Bit[7]: Not used Bit[6]: pd_byp_en Bit[5]: man_gain_en Bit[4]: man_digigain_en Bit[3]: ctrl_latch_en Bit[2]: pd_before_lenc_en Bit[1]: pd_chk_en Bit[0]: rst_protect_en
0x5003	ISP CTRL 3	0x00	RW	Bit[7]: man_flip Bit[6]: man_mirror Bit[5]: man_hdr_ptn Bit[4]: man_work_mode Bit[3]: man_flip_en Bit[2]: man_mirror_en Bit[1]: man_hdr_ptn_en Bit[0]: man_work_mode_en

table 6-24 ISP control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x5004	ISP CTRL 4	0x00	RW	Bit[7]: sof_sel Bit[6]: eof_sel Bit[5:4]: man_cfa_ptn Bit[3]: man_expo_en Bit[2]: man_real_gain_en Bit[1]: man_blc_en Bit[0]: man_cfa_ptn_en
0x5005	ISP CTRL 5	0x02	RW	Bit[7]: raw10_out_sel Bit[6]: man_raw10_en_out_sel Bit[5]: man_raw10_en Bit[4]: pd_chk_l_man_en Bit[3]: pd_chk_s_man_en Bit[2]: man_pd_loc_en Bit[1]: man_size_en Bit[0]: man_pipe_en
0x5006	ISP CTRL 6	0x00	RW	Bit[7:4]: Not used Bit[3:0]: man_real_gain_0[11:8]
0x5007	ISP CTRL 7	0x10	RW	Bit[7:0]: man_real_gain_0[7:0]
0x5008	ISP CTRL 8	0x00	RW	Bit[7:4]: Not used Bit[3:0]: man_real_gain_1[11:8]
0x5009	ISP CTRL 9	0x10	RW	Bit[7:0]: man_real_gain_1[7:0]
0x500A	ISP CTRL A	0x00	RW	Bit[7:4]: Not used Bit[3:0]: man_blc_0[11:8]
0x500B	ISP CTRL B	0x10	RW	Bit[7:0]: man_blc_0[7:0]
0x500C	ISP CTRL C	0x00	RW	Bit[7:4]: Not used Bit[3:0]: man_blc_1[11:8]
0x500D	ISP CTRL D	0x10	RW	Bit[7:0]: man_blc_1[7:0]
0x500E	ISP CTRL E	0x00	RW	Bit[7:4]: Not used Bit[3:0]: man_expo_0[19:16]
0x500F	ISP CTRL F	0x00	RW	Bit[7:0]: man_expo_0[15:8]
0x5010	ISP CTRL 10	0x10	RW	Bit[7:0]: man_expo_0[7:0]
0x5011	ISP CTRL 11	0x00	RW	Bit[7:2]: Not used Bit[1:0]: pd_byp_mf_opt
0x5012	ISP CTRL 12	0x02	RW	Bit[7:5]: Not used Bit[4:0]: man_hsize[12:8]
0x5013	ISP CTRL 13	0x80	RW	Bit[7:0]: man_hsize[7:0]
0x5014	ISP CTRL 14	0x01	RW	Bit[7:4]: Not used Bit[3:0]: man_vsize[11:8]

table 6-24 ISP control registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x5015	ISP CTRL 15	0xE0	RW	Bit[7:0]: man_vsize[7:0]
0x5016	ISP CTRL 16	0x00	RW	Bit[7:4]: Not used Bit[3:0]: man_expo_1[19:16]
0x5017	ISP CTRL 17	0x00	RW	Bit[7:0]: man_expo_1[15:8]
0x5018	ISP CTRL 18	0x10	RW	Bit[7:0]: man_expo_1[7:0]
0x5019	ISP CTRL 19	0x00	RW	Bit[7:2]: Not used Bit[1]: man_bw_mode Bit[0]: man_bw_mode_en
0x501A	ISP CTRL 1A	0x00	RW	Bit[7:6]: Not used Bit[5:0]: man_digi_gain_0[13:8]
0x501B	ISP CTRL 1B	0x10	RW	Bit[7:0]: man_digi_gain_0[7:0]
0x501C	ISP CTRL 1C	0x00	RW	Bit[7:6]: Not used Bit[5:0]: man_digi_gain_1[13:8]
0x501D	ISP CTRL 1D	0x10	RW	Bit[7:0]: man_digi_gain_1[7:0]
0x501E	ISP CTRL 1E	0x00	RW	Bit[7:4]: Not used Bit[3:0]: man_gain_0[11:8]
0x501F	ISP CTRL 1F	0x10	RW	Bit[7:0]: man_gain_0[7:0]
0x5020	ISP CTRL 20	0x00	RW	Bit[7:4]: Not used Bit[3:0]: man_gain_1[11:8]
0x5021	ISP CTRL 21	0x10	RW	Bit[7:0]: man_gain_1[7:0]
0x5022	ISP CTRL 22	0x00	RW	Bit[7]: dis_ck_gt_win Bit[6]: dis_ck_gt_dns Bit[5]: dis_ck_gt_dpc Bit[4]: dis_ck_gt_pd_byp Bit[3]: dis_ck_gt_otp Bit[2]: dis_ck_gt_awbg Bit[1]: dis_ck_gt_lenc Bit[0]: dis_ck_gt_top
0x5023	ISP CTRL 23	0x00	RW	Bit[7:3]: Not used Bit[2]: dis_ck_gt_vblank Bit[1]: dis_ck_gt_hblank Bit[0]: dis_ck_gt_all
0x5024	ISP CTRL 24	0x00	RW	Bit[7:6]: Not used Bit[5:2]: sram_rm Bit[1]: sram_rme Bit[0]: sram_test1

6.25 pre_ISP control [0x5081 - 0x5093, 0x50A0 - 0x50AF]

table 6-25 pre_ISP control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5081	ISP CTRL 1	0x04	RW	Bit[7]: test_en Bit[6]: rolling Bit[5]: trans Bit[4]: squ_bw Bit[3:2]: bar_style Bit[1:0]: test_sel
0x5082	ISP CTRL 2	0x01	RW	Bit[7]: Not used Bit[6]: win_cut_en Bit[5]: isp_test Bit[4]: rnd_same Bit[3:0]: rnd_seed
0x5083	ISP CTRL 3	0x30	RW	Bit[7]: Not used Bit[6]: end_man Bit[5]: mirror_opt Bit[4]: flip_opt Bit[3]: mirror_bg Bit[2]: flip_br Bit[1]: offset_man Bit[0]: scale_size_man
0x5084	ISP CTRL 4	0x0F	RW	Bit[7:0]: l_ratio
0x5085	ISP CTRL 5	0x0F	RW	Bit[7:0]: s_ratio
0x5086	ISP CTRL 6	0x00	RW	Bit[7:6]: Not used Bit[5]: x_skip_man_en Bit[4:0]: x_skip_man
0x5087	ISP CTRL 7	0x00	RW	Bit[7:6]: Not used Bit[5]: y_skip_man_en Bit[4:0]: y_skip_man
0x5088	ISP CTRL 8	0x00	RW	Bit[7:4]: Not used Bit[3:0]: In_number0[IV_SW-1:8]
0x5089	ISP CTRL 9	0x01	RW	Bit[7:0]: In_number0[7:0]
0x508A	ISP CTRL A	0x00	RW	Bit[7:4]: Not used Bit[3:0]: In_number1[IV_SW-1:8]
0x508B	ISP CTRL B	0x01	RW	Bit[7:0]: In_number1[7:0]
0x508C	ISP CTRL C	0x00	RW	Bit[7:5]: Not used Bit[4:0]: man_x_offset[IH_SW-1:8]
0x508D	ISP CTRL D	0x00	RW	Bit[7:0]: man_x_offset[7:0]

table 6-25 pre_ISP control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x508E	ISP CTRL E	0x00	RW	Bit[7:4]: Not used Bit[3:0]: man_y_offset[IV_SW-1:8]
0x508F	ISP CTRL F	0x00	RW	Bit[7:0]: man_y_offset[7:0]
0x5090	ISP CTRL 10	0x00	RW	Bit[7:5]: Not used Bit[4:0]: man_x_end[IH_SW-1:8]
0x5091	ISP CTRL 11	0x00	RW	Bit[7:0]: man_x_end[7:0]
0x5092	ISP CTRL 12	0x00	RW	Bit[7:4]: Not used Bit[3:0]: man_y_end[IV_SW-1:8]
0x5093	ISP CTRL 13	0x00	RW	Bit[7:0]: man_y_end[7:0]
0x50A0	ISP CTRL 20	–	R	Bit[7:5]: Not used Bit[4:0]: x_even_inc
0x50A1	ISP CTRL 21	–	R	Bit[7:5]: Not used Bit[4:0]: x_odd_inc
0x50A2	ISP CTRL 22	–	R	Bit[7:5]: Not used Bit[4:0]: y_even_inc
0x50A3	ISP CTRL 23	–	R	Bit[7:5]: Not used Bit[4:0]: y_odd_inc
0x50A4	ISP CTRL 24	–	R	Bit[7:5]: Not used Bit[4:0]: x_offset[IH_SW-1:8]
0x50A5	ISP CTRL 25	–	R	Bit[7:0]: x_offset[7:0]
0x50A6	ISP CTRL 26	–	R	Bit[7:4]: Not used Bit[3:0]: y_offset[IV_SW-1:8]
0x50A7	ISP CTRL 27	–	R	Bit[7:0]: y_offset[7:0]
0x50A8	ISP CTRL 28	–	R	Bit[7:5]: Not used Bit[4:0]: l_px_cnt[IH_SW-1:8]
0x50A9	ISP CTRL 29	–	R	Bit[7:0]: l_px_cnt[7:0]
0x50AA	ISP CTRL 2A	–	R	Bit[7:4]: Not used Bit[3:0]: l_ln_cnt[IV_SW-1:8]
0x50AB	ISP CTRL 2B	–	R	Bit[7:0]: l_ln_cnt[7:0]
0x50AC	ISP CTRL 2C	–	R	Bit[7:5]: Not used Bit[4:0]: s_px_cnt[IH_SW-1:8]
0x50AD	ISP CTRL 2D	–	R	Bit[7:0]: s_px_cnt[7:0]
0x50AE	ISP CTRL 2E	–	R	Bit[7:4]: Not used Bit[3:0]: s_ln_cnt[IV_SW-1:8]
0x50AF	ISP CTRL 2F	–	R	Bit[7:0]: s_ln_cnt[7:0]

6.26 AWB gain control [0x5100 - 0x510B]

table 6-26 AWB gain control registers

address	register name	default value	R/W	description
0x5100	ISP CTRL 0	0x04	RW	Bit[7:4]: Not used Bit[3:0]: awb_gain_b[11:8]
0x5101	ISP CTRL 1	0x00	RW	Bit[7:0]: awb_gain_b[7:0]
0x5102	ISP CTRL 2	0x04	RW	Bit[7:4]: Not used Bit[3:0]: awb_gain_g[11:8]
0x5103	ISP CTRL 3	0x00	RW	Bit[7:0]: awb_gain_g[7:0]
0x5104	ISP CTRL 4	0x04	RW	Bit[7:4]: Not used Bit[3:0]: awb_gain_r[11:8]
0x5105	ISP CTRL 5	0x00	RW	Bit[7:0]: awb_gain_r[7:0]
0x5106	ISP CTRL 6	0x02	RW	Bit[7:6]: Not used Bit[5:0]: digigain_man[13:8]
0x5107	ISP CTRL 7	0x00	RW	Bit[7:0]: digigain_man[7:0]
0x5108	ISP CTRL 8	0x00	RW	Bit[7:3]: Not used Bit[2]: blc_man_en Bit[1]: digigain_man_en Bit[0]: cfa_ptn_man_en
0x5109	ISP CTRL 9	0x00	RW	Bit[7:2]: Not used Bit[1:0]: cfa_ptn_man
0x510A	ISP CTRL A	0x02	RW	Bit[7:4]: Not used Bit[3:0]: blc_man[11:8]
0x510B	ISP CTRL B	0x00	RW	Bit[7:0]: blc_man[7:0]

6.27 OTP control [0x5180 - 0x518B, 0x518E - 0x5193, 0x5195 - 0x51A3]

table 6-27 OTP control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5180	ISP CTRL 0	0x00	RW	Bit[7:0]: start_addr[15:8]
0x5181	ISP CTRL 1	0x00	RW	Bit[7:0]: start_addr[7:0]
0x5182	ISP CTRL 2	0x0F	RW	Bit[7:0]: end_addr[15:8]

table 6-27 OTP control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x5183	ISP CTRL 3	0xFF	RW	Bit[7:0]: end_addr[7:0]
0x5184	ISP CTRL 4	0x02	RW	Bit[7:5]: Not used Bit[4]: man_inc_en Bit[3]: disable_mf Bit[2]: disable_offset Bit[1]: mirror_opt Bit[0]: disable_bin
0x5185	ISP CTRL 5	0x6C	RW	Bit[7]: Not used Bit[6:5]: recov_method Bit[4]: fixed_replace Bit[3]: fixed_ptn Bit[2]: flip_opt Bit[1]: expo_en Bit[0]: gain_en
0x5186	ISP CTRL 6	0x00	RW	Bit[7]: Not used Bit[6:0]: expo_constrain_l[14:8]
0x5187	ISP CTRL 7	0x00	RW	Bit[7:0]: expo_constrain_l[7:0]
0x5188	ISP CTRL 8	0x07	RW	Bit[7:0]: gain_constrain_l
0x5189	ISP CTRL 9	0x08	RW	Bit[7:5]: Not used Bit[4]: thre_en Bit[3:0]: thre_l
0x518A	ISP CTRL A	0x01	RW	Bit[7:5]: Not used Bit[4:0]: man_x_even_inc
0x518B	ISP CTRL B	0x01	RW	Bit[7:5]: Not used Bit[4:0]: man_x_odd_inc
0x518E	ISP CTRL E	0x01	RW	Bit[7:5]: Not used Bit[4:0]: man_y_even_inc
0x518F	ISP CTRL F	0x01	RW	Bit[7:5]: Not used Bit[4:0]: man_y_odd_inc
0x5190	ISP CTRL 10	0x00	RW	Bit[7:0]: man_x_offset[15:8]
0x5191	ISP CTRL 11	0x00	RW	Bit[7:0]: man_x_offset[7:0]
0x5192	ISP CTRL 12	0x00	RW	Bit[7:0]: man_y_offset[15:8]
0x5193	ISP CTRL 13	0x00	RW	Bit[7:0]: man_y_offset[7:0]

table 6-27 OTP control registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x5195	ISP CTRL 15	0x00	RW	Bit[7]: Not used Bit[6]: dis_proc_gate Bit[5]: dis_mem_rd_gate Bit[4]: dis_start_gate Bit[3]: dis_buf3_gate Bit[2]: dis_buf2_gate Bit[1]: dis_buf1_gate Bit[0]: dis_buf0_gate
0x5196	ISP CTRL 16	0x0F	RW	Bit[7:0]: man_x_end[15:8]
0x5197	ISP CTRL 17	0x00	RW	Bit[7:0]: man_x_end[7:0]
0x5198	ISP CTRL 18	0x00	RW	Bit[7]: Not used Bit[6:0]: expo_constrain_s[14:8]
0x5199	ISP CTRL 19	0x00	RW	Bit[7:0]: expo_constrain_s[7:0]
0x519A	ISP CTRL 1A	0x07	RW	Bit[7:0]: gain_constrain_s
0x519B	ISP CTRL 1B	0x08	RW	Bit[7:4]: Not used Bit[3:0]: thre_s
0x519C	ISP CTRL 1C	–	R	Bit[7:5]: Not used Bit[4:0]: ro_x_offset[IH_SW-1:8]
0x519D	ISP CTRL 1D	–	R	Bit[7:0]: ro_x_offset[7:0]
0x519E	ISP CTRL 1E	–	R	Bit[7:4]: Not used Bit[3:0]: ro_y_offset[IV_SW-1:8]
0x519F	ISP CTRL 1F	–	R	Bit[7:0]: ro_y_offset[7:0]
0x51A0	ISP CTRL 20	–	R	Bit[7:5]: Not used Bit[4:0]: ro_x_even_inc
0x51A1	ISP CTRL 21	–	R	Bit[7:5]: Not used Bit[4:0]: ro_x_odd_inc
0x51A2	ISP CTRL 22	–	R	Bit[7:5]: Not used Bit[4:0]: ro_y_even_inc
0x51A3	ISP CTRL 23	–	R	Bit[7:5]: Not used Bit[4:0]: ro_y_odd_inc

6.28 DPC control [0x5200 - 0x5210, 0x5214, 0x5218 - 0x521F]

table 6-28 DPC control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5200	ISP CTRL 0	0x1B	RW	Bit[7]: Enable tail Bit[6]: Enable general tail Bit[5]: Enable 33 clusters Bit[4]: Enable saturated cross cluster Bit[3]: Enable cross cluster Bit[2]: Manual mode Bit[1]: Black pixel Bit[0]: White pixel
0x5201	ISP CTRL 1	0x94	RW	Bit[7:6]: VNumList_2 Bit[5:4]: VNumList_1 Bit[3:2]: VNumList_0 Bit[1]: Clip interp G Bit[0]: Share buffer
0x5202	ISP CTRL 2	0x2E	RW	Bit[7:6]: Not used Bit[5:4]: Pxorder Bit[3:2]: Edge option Bit[1:0]: VNumList_3
0x5203	ISP CTRL 3	0x24	RW	Bit[7]: Not used Bit[6:3]: WThresList_0 Bit[2:0]: MaxVNum
0x5204	ISP CTRL 4	0x12	RW	Bit[7:4]: WThresList_2 Bit[3:0]: WThresList_1
0x5205	ISP CTRL 5	0x41	RW	Bit[7:4]: BThresRatio Bit[3:0]: WThresList_3
0x5206	ISP CTRL 6	0x48	RW	Bit[7:4]: Status threshold Bit[3:0]: More connection case thre
0x5207	ISP CTRL 7	0x84	RW	Bit[7:4]: Matching threshold Bit[3:0]: Status threshold step
0x5208	ISP CTRL 8	0x40	RW	Bit[7:4]: Adaptive pattern step Bit[3:0]: Adaptive pattern thre
0x5209	ISP CTRL 9	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Saturate
0x520A	ISP CTRL A	0x03	RW	Bit[7:0]: GainList_0
0x520B	ISP CTRL B	0x0F	RW	Bit[7:0]: GainList_1
0x520C	ISP CTRL C	0x3F	RW	Bit[7:0]: GainList_2
0x520D	ISP CTRL D	0x0F	RW	Bit[7:0]: DPCLevelList_0

table 6-28 DPC control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x520E	ISP CTRL E	0xFD	RW	Bit[7:0]: DPCLevelList_1
0x520F	ISP CTRL F	0xF5	RW	Bit[7:0]: DPCLevelList_2
0x5210	ISP CTRL 10	0xF5	RW	Bit[7:0]: DPCLevelList_3
0x5214	ISP CTRL 14	0x00	RW	Bit[7]: ClkGateDisRec2 Bit[6]: ClKGateDisRec1 Bit[5]: ClkGateDisRec0 Bit[4]: ClkGateDisSram Bit[3]: ClKGateDisBuf Bit[2]: ClkGateDisSF Bit[1]: ClkGateDis Bit[0]: ManPxOrder
0x5218	ISP CTRL 18	–	R	Bit[7]: Not used Bit[6:0]: Bthre
0x5219	ISP CTRL 19	–	R	Bit[7:5]: Not used Bit[4:0]: Wthre
0x521A	ISP CTRL 1A	–	R	Bit[7:5]: Not used Bit[4:0]: Thre1
0x521B	ISP CTRL 1B	–	R	Bit[7:0]: Thre2
0x521C	ISP CTRL 1C	–	R	Bit[7]: Not used Bit[6:0]: Thre3
0x521D	ISP CTRL 1D	–	R	Bit[7]: Not used Bit[6:0]: Thre4
0x521E	ISP CTRL 1E	–	R	Bit[7:4]: Level Bit[3:0]: Pconnected
0x521F	ISP CTRL 1F	–	R	Bit[7:3]: Not used Bit[2:0]: Vnum

6.29 window control [0x5800 - 0x580B, 0x5810 - 0x5813]

table 6-29 window control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5800	ISP_CTRL_0	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Window X start[11:8]

table 6-29 window control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5801	ISP_CTRL_1	0x00	RW	Bit[7:0]: Window X start[7:0]
0x5802	ISP_CTRL_2	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Window Y start[11:8]
0x5803	ISP_CTRL_3	0x00	RW	Bit[7:0]: Window Y start[7:0]
0x5804	ISP_CTRL_4	0x02	RW	Bit[7:4]: Not used Bit[3:0]: Window width[11:8]
0x5805	ISP_CTRL_5	0x80	RW	Bit[7:0]: Window width[7:0]
0x5806	ISP_CTRL_6	0x01	RW	Bit[7:4]: Not used Bit[3:0]: Window height[11:8]
0x5807	ISP_CTRL_7	0xE0	RW	Bit[7:0]: Window height[7:0]
0x5808	ISP_CTRL_8	0x00	RW	Bit[7:4]: emb_num Bit[3]: Not used Bit[2]: emb_flag_sel 0: Start line 1: End line Bit[1]: Not used Bit[0]: win_man_en 0: Window size from window top 1: Window size from register
0x5809	Y EXT LN	0x00	RW	Bit[7:4]: Not used Bit[3:0]: y_ext_ln[3:0] Window height extent line number
0x580A	Y OFFSET MINUS	0x00	RW	Bit[7:0]: y_offset_minus[7:0] Minus the Y offset
0x580B	REG0B	0x0F	RW	Bit[7:4]: Not used Bit[3]: mirror_en Bit[2]: flip_en Bit[1]: hwin_en Bit[0]: vwin_en
0x5810	PX CNT	–	R	Bit[7:4]: Not used Bit[3:0]: px_cnt[11:8]
0x5811	PX CNT	–	R	Bit[7:0]: px_cnt[7:0]
0x5812	LN CNT	–	R	Bit[7:4]: Not used Bit[3:0]: ln_cnt[11:8]
0x5813	LN CNT	–	R	Bit[7:0]: ln_cnt[7:0]

6.30 OTP_SRAM [0x6000 - 0x63FF]

table 6-30 OTP_SRAM registers

address	register name	default value	R/W	description
0x6000~ 0x63FF	OTP_SRAM	0x00	RW	Bit[7:0]: OTP buffer

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)	V_{DD-A}	4.5V
	V_{DD-D}	3V
	V_{DD-IO}	4.5V
electro-static discharge (ESD)	human body model	2000V
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 temparature (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 (for applications up to 90 fps)[a]	-30°C to +85°C junction temperature
stable image temperature[b]	0°C to +60°C junction temperature

a. sensor functions but image quality may be noticeably different at temperatures outside of stable image range

b. image quality remains stable throughout this temperature range

7.3 DC characteristics

table 7-3 DC characteristics (-30°C < T_J < 85°C)

symbol	parameter	min	typ	max[a]	unit
supply					
V_{DD-A}	supply voltage (analog)	2.7	2.8	2.9	V
V_{DD-D}	supply voltage (digital core)	1.14	1.2	1.26	V
V_{DD-IO}	supply voltage (digital I/O)	1.7	1.8	1.9	V
I_{DD-A}	active (operating) current[b]		50	62	mA
I_{DD-IO}			1.5	2.1	mA
I_{DD-D}			122	162	mA
$I_{DDS-SCCB}$	standby current[c]		4.5	15	mA
$I_{DDS-PWDNB}$			3.5	15	mA
$I_{DDS-XSHUTDOWN}$			1.5	10	µA
digital inputs (typical conditions: AVDD = 2.8V, DVDD = 1.2V, DOVDD = 1.8V)					
V_{IL}	input voltage LOW			0.54	V
V_{IH}	input voltage HIGH	1.26			V
C_{IN}	input capacitor			10	pF
digital outputs (standard loading 25 pF)					
V_{OH}	output voltage HIGH	1.62			V
V_{OL}	output voltage LOW			0.18	V
serial interface inputs					
V_{IL}[d]	SCL and SDA	-0.5	0	0.54	V
V_{IH}	SCL and SDA	1.28	1.8	3.0	V

a. maximum active current is measured under typical supply voltage

b. DVDD is provided by external regulator for lower power consumption. DOVDD = 1V

c. standby current is measured at room temperature with external clock off

d. based on DOVDD = 1V

7.4 timing characteristics

figure 7-1 reference clock input timing diagram

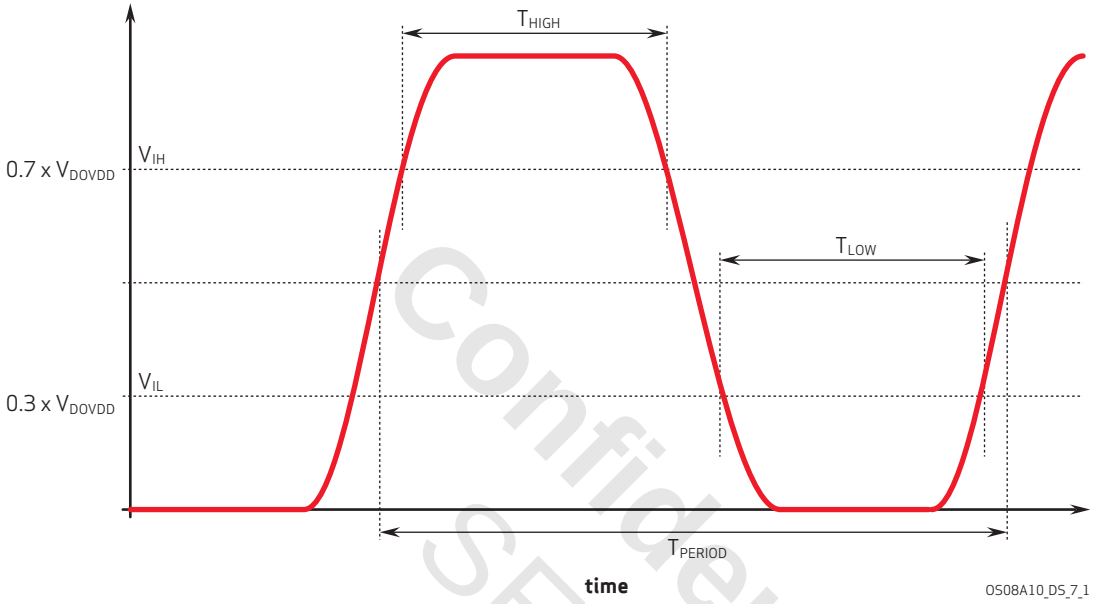


table 7-4 timing characteristics

symbol	parameter	min	typ	max	unit
oscillator and clock input					
f_{XVCLK}	frequency (XVCLK)[a]	6	24	64	MHz
T_{PERIOD}	period (XVCLK)	15.6	41.7	166.7	ns
T_{LOW}	low level width (XVCLK)	0.35×T_{PERIOD}		0.65×T_{PERIOD}	ns
T_{HIGH}	high level width (XVCLK)	0.35×T_{PERIOD}		0.65×T_{PERIOD}	ns

a. for input clock range 6~27MHz, the OS08A10 can tolerate input clock period jitter up to 600ps peak-to-peak

8 mechanical specifications

8.1 physical specifications

figure 8-1 package specifications

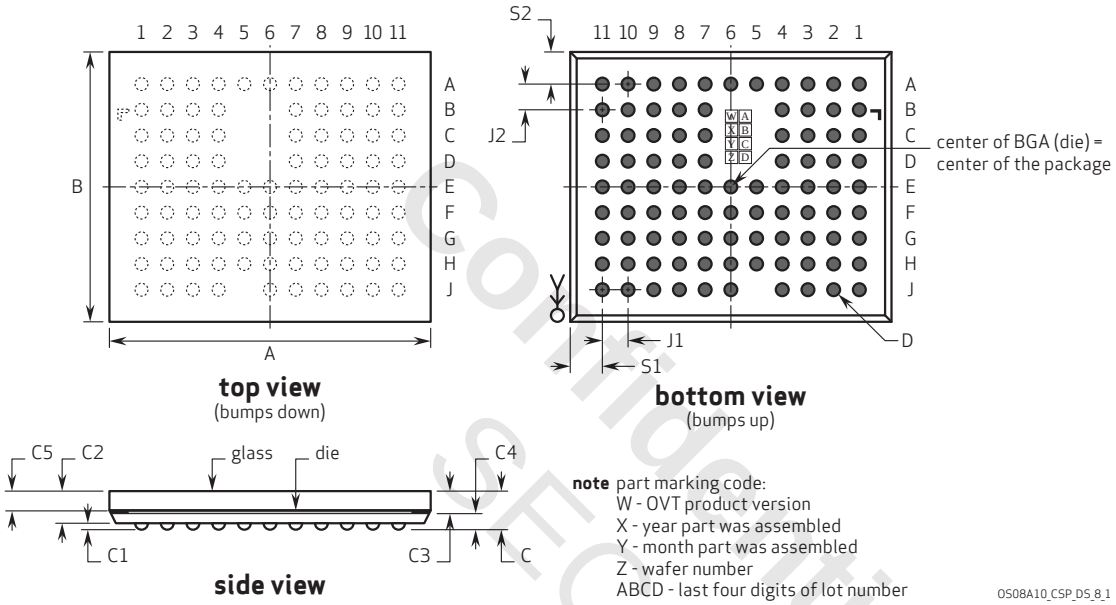


table 8-1 package dimensions (sheet 1 of 2)

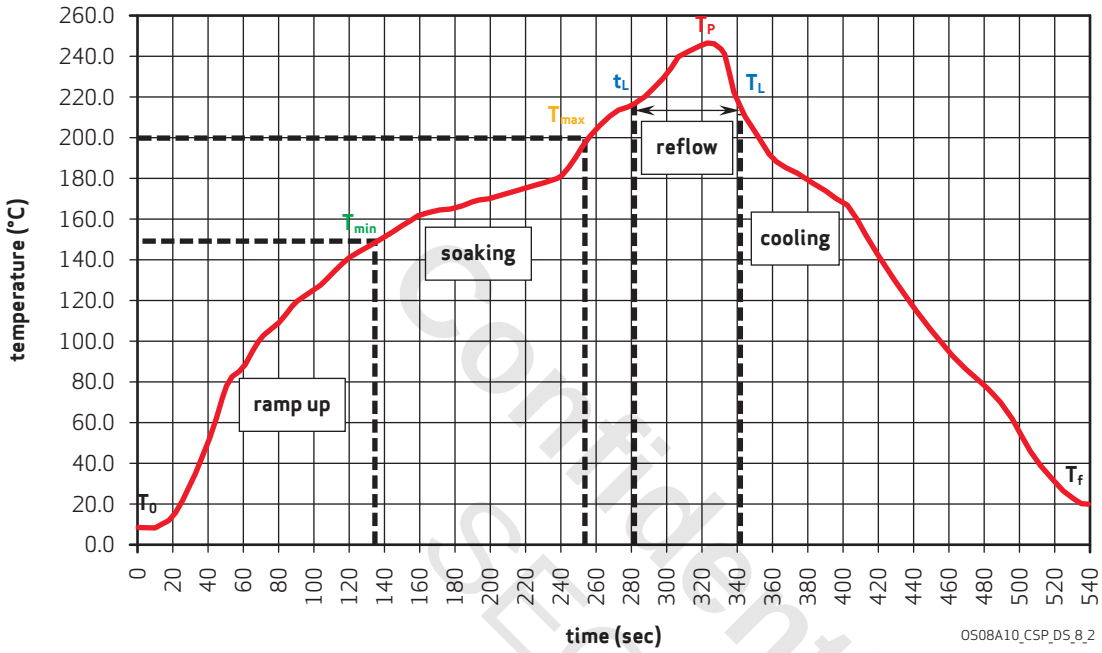
parameter	symbol	min	typ	max	unit
package body dimension x	A	8914.2	8939.2	8964.2	µm
package body dimension y	B	6315	6340	6365	µm
package height	C	675	735	795	µm
ball height	C1	120	150	180	µm
package body thickness	C2	540	585	630	µm
thickness from top glass surface to die	C3	425	445	465	µm
image plane height	C4	235	290	345	µm
glass thickness	C5	390	400	410	µm
ball diameter	D	250	280	310	µm
total pin count	N		92 (4 NC)		

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

parameter	symbol	min	typ	max	unit
pin count x-axis	N1		11		
pin count y-axis	N2		9		
pins pitch x-axis	J1		720		µm
pins pitch y-axis	J2		630		µm
edge-to-pin center distance along x	S1	839.6	869.6	899.6	µm
edge-to-pin center distance along y	S2	620	650	680	µm
air gap between die and glass		40	45	50	µm

8.2 IR reflow specifications

figure 8-2 IR reflow ramp rate requirements



note The OS08A10 uses a lead-free package.

note To reduce image artifacts from infrared light and to provide the best image quality, OmniVision recommends an IR cut filter.

table 8-2 reflow conditions[ab]

zone	description	exposure
ramp up A (T_0 to T_{min})	heating from room temperature to 150°C	temperature slope ≤ 3°C per second
soaking	heating from 150°C to 200°C	90 ~ 150 seconds
ramp up B (t_L to T_P)	heating from 217°C to 245°C	temperature slope ≤ 3°C per second
peak temperature	maximum temperature in SMT	245°C +0/-5° (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 down from 245°C to 217°C	temperature slope ≤ 3°C per second
ramp down B (T_L to T_f)	cooling down from 217°C to room temperature	temperature slope ≤ 2°C per second
T_0 to T_P	room temperature to peak temperature	≤ 8 minutes

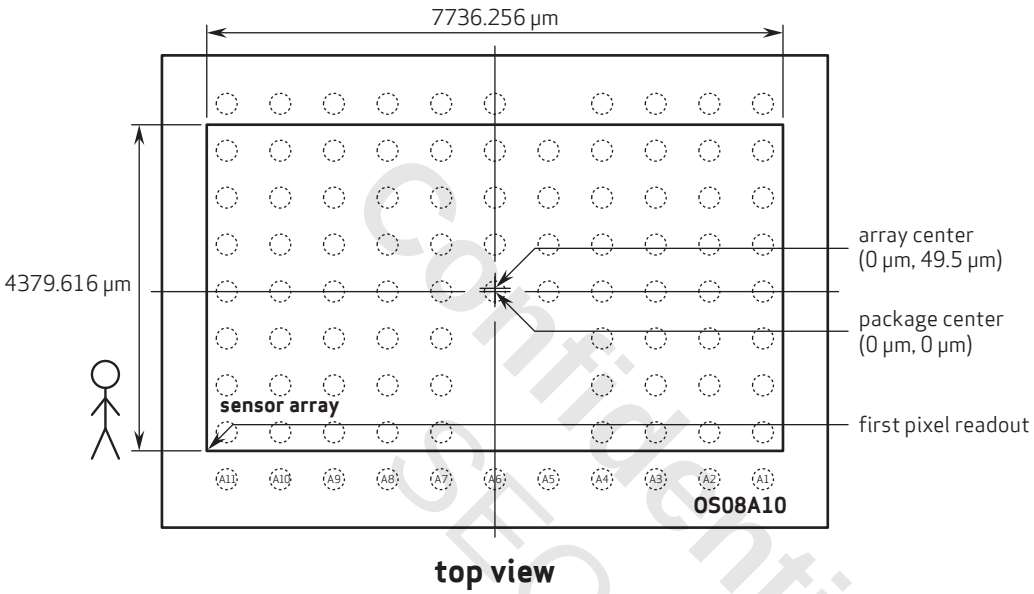
a. maximum number of reflow cycles = 3

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

9 optical specifications

9.1 sensor array center

figure 9-1 sensor array center

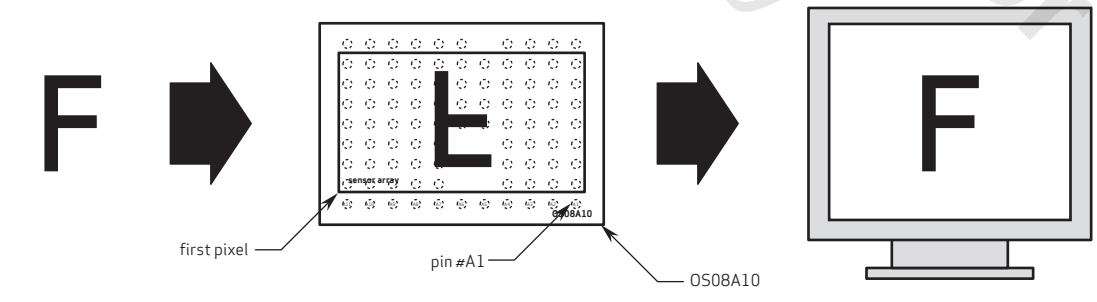


note 1 this drawing is not to scale and is for reference only.

note 2 as most optical assemblies invert and mirror the image, the chip is typically mounted with pin A1 oriented down on the PCB.

OS08A10_CSP_DS_9_1

figure 9-2 final image output



OS08A10_CSP_DS_9_2

9.2 lens chief ray angle (CRA)

figure 9-3 chief ray angle (CRA)

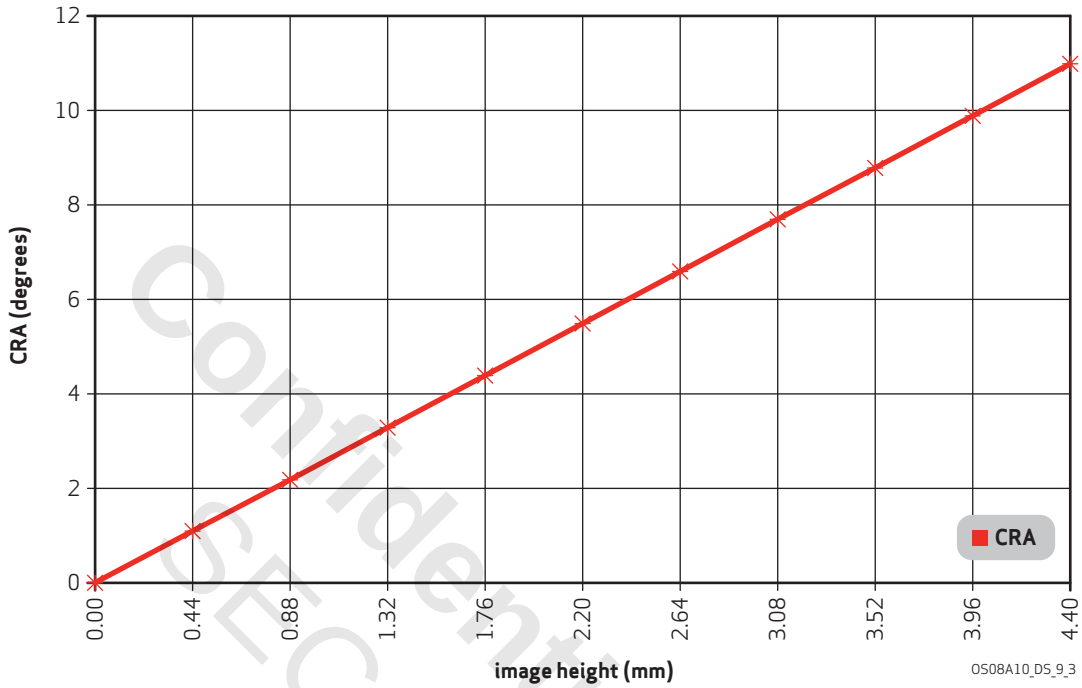


table 9-1 CRA versus image height plot

field (%)	image height (mm)	CRA (degrees)
0.00	0.00	0.0
0.10	0.44	1.1
0.20	0.88	2.2
0.30	1.32	3.3
0.40	1.76	4.4
0.50	2.2	5.5
0.60	2.64	6.6
0.70	3.08	7.7
0.80	3.52	8.8
0.90	3.96	9.9
1.00	4.4	11

revision history

version 1.0 09.13.2016

- initial release

version 1.01 10.21.2016

- in key specifications, changed active power requirements from 90 mW to 221 mW
- in table 7-3, changed values for I_{DD-A}, I_{DD-IO}, I_{DD-D} typical active (operating) current from 50 mA, 3 mA, 90 mA, respectively to 46 mA, 1.1 mA, 54 mA, respectively

version 1.02 01.25.2017

- in section 2.2, updated figure 2-2

version 1.03 01.30.2017

- in chapter 4, removed section 4.6

version 1.1 04.04.2017

- in key specifications, changed active power requirements to 280mW, changed standby power requirements to 1000 µA, changed max S/N ratio to 39 dB, changed dynamic range to 74 dB @ 16x gain, and changed sensitivity to 13,000e⁻/Lux-sec
- in chapter 2, updated figure 2-2
- in section 4.9, changed last sentence of first paragraph to "figure 4-16 shows a description of the ULPM interrupt operations." and updated figure 4-16
- in section 4.9.1, changed second step to "...sensor operational in the high frame rate QVGA/QQVGA mode. The following registers will need to be written: QVGA/QQVGA setting, exposure/gain settings, window, and stable range.", changed fourth step to "...on ULPM pin and 0x4A20[0]. The values of gain, exposure, mean R, G, B, Y, are latched onto separate registers (0x4A21 ~ 0x4A2B).", changed sixth step to "...trigger (rewrite 0x4A10 = 0) to go to step 2.", and changed title of figure 4-17 to "stable range-1 diagram"
- in section 4.9.2, changed second step to "...new stable range-2 (0x4A08~0x4A0B) around...", changed third step to "...and 0x4A20[1] latched the gain, exposure, mean (R, G, B, Y) values on registers (0x4A21~0x4A2B).", changed fourth step to "...clear the interrupt (rewrite 0x4A10=0)...", changed title of figure 4-18 to "stable range-2 diagram", and updated figure 4-18
- in section 4.10, changed section title to "stable range/window registers for ULPM interrupt and changed title of table 4-9 to "stable range/window registers"
- in table 4-9, removed rows for registers 0x3504, 0x3A01, 0x3A04, 0x3A05, 0x3A07, and 0x3A0F, changed register name for registers 0x3A02 and 0x3A03 to "EXP_CTRL02" and "EXP_CTRL03", respectively, changed description for registers 0x3A02 and 0x3A03 to "Bit[7:0]: Top stable range-1" and "Bit[7:0]: Bottom stable range-1", respectively, changed register address 0x4F04 to 0x4B04,

changed register address 0x4F05 to 0x4B05, changed register address 0x4F07 to 0x4B07, and changed register address 0x4F10 to 0x4B10

- in table 4-10, changed all register addresses from 0x47xx to 0x4Axx
- in section 4.11.1, removed second code line of item 2, changed third and fourth code lines of item 2 to "...stable range-1 upper bound in high 8-bit format" and "...stable range-1 lower bound in high 8-bit format", respectively, removed fifth and sixth code lines of item 2, changed seventh to tenth code lines of item 2 from "...4f04..." to "...4b04...", from "...4f05..." to "...4b05...", from "...4f07..." to "...4b07...", and from "...4f10..." to "...4b10...", respectively, changed first code line of item 3 from "...3010..." to "...3016...", changed second code line of item 3 from "...4700..." to "...4A00...", changed third code line of item 3 from "...4701..." to "...4A01...", changed item 4 to "...via register 0x4A20 or ULPM pin.", changed item 5 to "...observed at 0x4A20 (=0x05) and ULPM pin.", changed second to fifth code lines of item 6 to "6c 4a08 01; stable range-2 upper bound in 10-bit format", "6c 4a09 e0; stable range-2 upper bound in 10-bit format", "6c 4a0a 01; stable range-2 lower bound in 10-bit format", and "6c 4a0b a0; stable range-2 upper bound in 10-bit format", respectively, changed item 7 to "...register 0x4A10, and then monitor register 0x4A20 for step 2 flag. Once image Y value is outside stable range-2, ULPM interrupt can be observed at 0x4A20 (=0x06) and ULPM pin.", changed first code line of item 7 to "6c 4a10 00", and changed item 8 to "...register 0x4A10, and then go back to step 1 again."
- in table 5-5, changed description of register bits 0x3503[7] and 0x3503[3] to "Not used"
- in table 6-3, changed description of register bits 0x3016[5] to "Bit[5]: ULPM clock enable" and 0x3016[1] to "Bit[1]: ULPM reset, 1: Reset"
- in table 6-7, changed description of register bits 0x3503[7] and 0x3503[3] to "Not used"
- in table 7-3, changed typ value of I_{DD-A} to 50mA, changed typ value of I_{DD-IO} to 2mA and changed typ value of I_{DD-D} to 117mA

version 1.11 04.18.2017

- in chapter 9, updated figure 9-1 and added figure 9-2

version 1.12 05.17.2017

- in section 2.10.2, updated figure 2-6
- in table 5-5, updated description of register 0x3502
- in chapter 7, updated section 7.4, including adding figure 7-1

version 2.0 09.19.2017

- changed datasheet from Preliminary Specification to Product Specification
- in ordering information, removed "-Z" from ordering part number
- in key specifications, changed active power requirements to 240 mA, removed standby power requirements, and changed XSHUTDOWN power requirements to <10 µA
- in table 7-3, changed max value for V_{DD-A} to 2.9V, changed max value for I_{DD-A} to 62mA, changed typ and max values for I_{DD-IO} to 1.5mA and 2.1mA, respectively, changed typ and max values for I_{DD-D} to 122mA and 162mA, respectively, changed typ and max values for $I_{DDS-SCCB}$ to 4.5mA and 15mA, respectively, changed typ and max values for $I_{DDS-PWDNB}$ to 3.5mA and 15mA, respectively, and changed typ and max values for $I_{DDS-XSHUTDOWN}$ to 1.5µA and 10µA, respectively

version 2.01 10.26.2017

- in section 2.2, updated figure 2-2

version 2.1 02.22.2018

- in features, changed thirteenth bullet to "PLL with SSC support"
- in key specifications, changed maximum exposure interval to VTS-8
- in section 2.10.1, changed first sentence to "...and 1500 MHz MIPI serial clock from a 6~64MHz input clock."
- in section 2.10.2, changed second sentence to "The VCO range is from 500 MHz to 1200 MHz." and updated figure 2-6
- in chapter 2, updated sections 2.12.1, 2.12.2.3, 2.12.2.4, and 2.12.2.5
- in chapter 4, updated section 4.7.1
- in section 4.12, updated first paragraph
- in section 4.12.1, updated figure 4-19
- in section 4.12.2, updated figure 4-20
- in table 5-5, removed row for 0x3510
- in table 7-4, changed max value for f_{XVCLK} to 64MHz and changed min value for T_{PERIOD} to 15.6ns

version 2.11 04.09.2018

- in section 4.6.1.2, changed title to "LED 1 & 2 mode", changed first paragraph completely, changed title of figure 4-9 to "LED 1 & 2 mode (part a)", moved figure 4-10 from section 4.6.1.3 to section 4.6.1.2, and changed title of figure 4-10 to "LED 1 & 2 mode (part b)
- in chapter 4, removed section 4.6.1.3
- in table 6-19, changed description of register bits 0x4809[5:0] to dt_man1 and changed description of registers 0x4815 and 0x4816 to Reserved

defining the future of digital imaging™

OmniVision Technologies, Inc.

UNITED STATES

4275 Burton Drive
Santa Clara, CA 95054
tel: +1 408 567 3000
fax: +1 408 567 3001
email: sales@ovt.com

UNITED KINGDOM

Hook, Hampshire +44 1256 744 610

GERMANY

Munich +49 89 63 81 99 88

INDIA

Bangalore +91 80 4112 8966

CHINA

Beijing + 86 10 6580 1690
Shanghai + 86 21 6175 9888
+86 21 5774 9288
Shenzhen + 86 755 8384 9733

JAPAN

Yokohama +81 45 478 7977
Kyoto +81 75 708 5352

KOREA

Seoul + 82 2 3472 3769

SINGAPORE +65 6933 1933

TAIWAN

Taipei +886 2 2657 9800
Hsin-chu +886 3 5656688

website: **www.ovt.com**