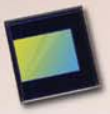
OmniVision



datasheet

PRODUCT SPECIFICATION

1/3.2" color CMOS 8 megapixel (3264 x 2448) image sensor
with improved OmniBSI-2™ technology

OV8865 (rev 1D)

color CMOS 8 megapixel (3264 x 2448) image sensor with OmniBSI-2™ technology

datasheet (COB)
PRODUCT SPECIFICATION

version 2.0
november 2013

To learn more about OmniVision Technologies, visit www.ovt.com.
OmniVision Technologies is publicly traded on NASDAQ under the symbol OVTI.

applications

- cellular phones
- tablets
- PC multimedia

ordering information

- **OV08865-G04A-1D** (color, chip probing, 200 µm backgrinding, reconstructed wafer with good die)

features

- automatic black level calibration (ABLC)
- programmable controls for frame rate, mirror and flip, cropping, and windowing
- static defective pixel canceling
- supports output formats: 10-bit RAW RGB (MIPI)
- supports horizontal and vertical subsampling
- supports images sizes: 3264x2448, 3264x1836, 2816x1584, 1632x1224, 1408x792
- supports 2x2 binning, re-sampling filter
- standard serial SCCB interface
- up to 4-lane MIPI serial output interface
- embedded 1536 bytes one-time programmable (OTP) memory for part identification, etc.
- two on-chip phase lock loops (PLLs)
- programmable I/O drive capability
- built-in temperature sensor

key specifications (typical)

- **active array size:** 3264 x 2448
- **power supply:**
  core: 1.2V
  analog: 2.8V
  I/O: 1.8, 2.8V
- **power requirements:**
  active: 196 mW (full resolution @ 30 fps)
  XSHUTDOWN: 5 µW
- **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-bit RAW
- **lens size:** 1/3.2"
- **lens chief ray angle:** 32.2° non-linear (see **figure 9-2**)
- **input clock frequency:** 6~27 MHz
- **max S/N ratio:** 36.7 dB
- **dynamic range:** 68.8 dB
- **maximum image transfer rate:**
  3264x2448: 30 fps (see **table 2-1**)
  3264x1836: 30 fps (see **table 2-1**)
  2816x1584: 30 fps (see **table 2-1**)
  1632x1224: 30 fps (see **table 2-1**)
  1408x792: 60 fps (see **table 2-1**)
- **sensitivity:** 940 mV/Lux-sec
- **scan mode:** progressive
- **maximum exposure interval:** 2480 x T_{ROW}
- **pixel size:** 1.4 µm x 1.4 µm
- **dark current:** 20e^-/sec @ 60°C junction temperature
- **image area:** 4614.4 µm x 3472 µm
- **die dimensions:** 5850 µm x 5700 µm (COB), 5900 µm x 5750 µm (RW) (see **section 8** for details)

note higher junction temperature degrades image quality

note COB refers to whole wafers with known good die and RW refers to singulated good die on a reconstructed wafer. Die size differs between COB and RW.

table of contents

list of figures

list of tables

1 signal descriptions

table 1-1 lists the signal descriptions and their corresponding pad numbers for the OV8865 image sensor. The die information is shown in section 8.

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

pad number	signal name	pad type	description
1	AGND	ground	analog ground
2	AGND	ground	analog ground
3	AVDD	power	analog power
4	AVDD	power	analog power
5	AVDD	power	analog power
6	AGND	ground	analog ground
7	ATEST	I/O	analog test pin
8	DOGND	ground	I/O ground
9	DOGND	ground	I/O ground
10	DVDD	power	digital circuit power
11	DVDD	power	digital circuit power
12	DOVDD	power	I/O power
13	DOGND	ground	I/O ground
14	NC	–	no connect
15	NC	–	no connect
16	DOGND	ground	I/O ground
17	DVDD	power	digital circuit power
18	DVDD	power	digital circuit power
19	**PWDNB**	input	power down (active low)
20	DOGND	ground	I/O ground
21	DOGND	ground	I/O ground
22	**TM**	input	test mode (active high with pull down resistor)
23	**XSHUTDOWN**	input	reset and power down (active low with pull down resistor)
24	DVDD	power	digital circuit power
25	DVDD	power	digital circuit power

table 1-1 signal descriptions (sheet 2 of 3)

pad number	signal name	pad type	description
26	NC	–	no connect
27	**SIOD**	I/O	SCCB interface data pin
28	**SIOC**	input	SCCB interface input clock
29	**SID**	input	SCCB last bit ID input 0: SCCB address = 0x6C 1: SCCB address = 0x20
30	DOGND	ground	I/O ground
31	DOGND	ground	I/O ground
32	DVDD	power	digital circuit power
33	AVDD	power	analog power
34	AVDD	power	analog power
35	AGND	ground	analog ground
36	AGND	ground	analog ground
37	DVDD	power	digital circuit power
38	**FSIN**	I/O	frame sync
39	DTEST	I/O	internal test pad, tie to ground
40	DOGND	ground	I/O ground
41	**VSYNC**	I/O	video output vertical signal
42	**HREF**	I/O	video output horizontal signal
43	**GPIO**	I/O	general purpose I/O
44	DOVDD	power	I/O power
45	**STROBE**	output	frame exposure output indicator
46	**IL_PWM**	output	mechanical shutter output indicator
47	DVDD	power	digital circuit power
48	**EXTCLK**	input	system clock input
49	DOGND	ground	I/O ground
50	**MDN3**	output	MIPI data negative output
51	**MDP3**	output	MIPI data positive output
52	EGND	ground	MIPI ground
53	**MDN1**	output	MIPI data negative output
54	**MDP1**	output	MIPI data positive output

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

pad number	signal name	pad type	description
55	MCN	output	MIPI clock negative output
56	MCP	output	MIPI clock positive output
57	EVDD	power	MIPI power
58	EGND	ground	MIPI ground
59	PVDD	power	PLL analog power
60	EGND	ground	MIPI ground
61	EVDD	power	MIPI power
62	MDN0	output	MIPI data negative output
63	MDP0	output	MIPI data positive output
64	EGND	ground	MIPI ground
65	MDN2	output	MIPI data negative output
66	MDP2	output	MIPI data positive output
67	DOGND	ground	I/O ground
68	DVDD	power	digital circuit power
69	VN	input	reference
70	VH	input	reference
71	AGND	ground	analog ground
72	AVDD	power	analog power

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

pad	signal name	RESET[a]	after RESET release[b]	software standby[c]	hardware standby[d]
19	PWDNB	input	input	input	input
22	TM	input	input	input	input
23	XSHUTDOWN	input	input	input	input
27	SIOD	open drain	I/O	I/O	open drain
28	SIOC	high-z	input	input	high-z
29	SID	input	input	input	input
38	FSIN	high-z	input	input (configurable)	input (configurable)

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

pad	signal name	RESET[a]	after RESET release[b]	software standby[c]	hardware standby[d]
41	VSYNC	high-z	high-z	high-z by default (configurable)	high-z by default (configurable)
42	HREF	high-z	high-z	high-z by default (configurable)	high-z by default (configurable)
43	GPIO	high-z	high-z	high-z by default (configurable)	high-z by default (configurable)
45	STROBE	low	low	low by default (configurable)	low by default (configurable)
46	IL_PWM	low	low	low by default (configurable)	low by default (configurable)
48	EXTCLK	high-z	input	input	high-z
50	MDN3	high-z	high	high by default (configurable)	high by default (configurable)
51	MDP3	high-z	high	high by default (configurable)	high by default (configurable)
53	MDN1	high-z	high	high by default (configurable)	high by default (configurable)
54	MDP1	high-z	high	high by default (configurable)	high by default (configurable)
55	MCN	high-z	high	high by default (configurable)	high by default (configurable)
56	MCP	high-z	high	high by default (configurable)	high by default (configurable)
62	MDN0	high-z	high	high by default (configurable)	high by default (configurable)
63	MDP0	high-z	high	high by default (configurable)	high by default (configurable)
65	MDN2	high-z	high	high by default (configurable)	high by default (configurable)
66	MDP2	high-z	high	high by default (configurable)	high by default (configurable)

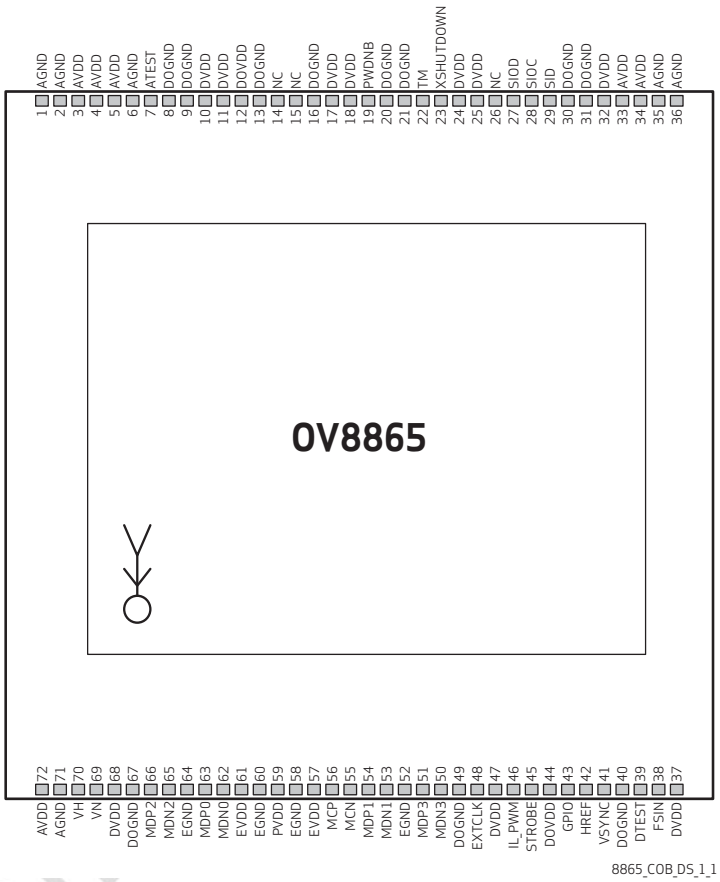
a. XSHUTDOWN = 0

b. XSHUTDOWN from 0 to 1

c. sensor set to sleep from streaming mode

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

figure 1-1 pad diagram



8865_COB_DS_1_1

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

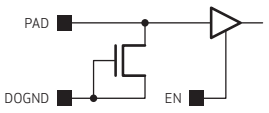
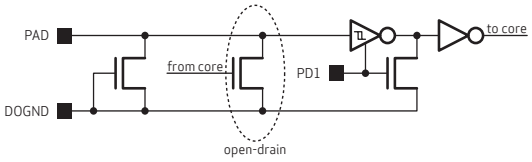
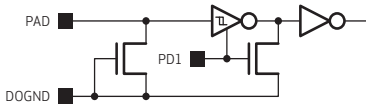
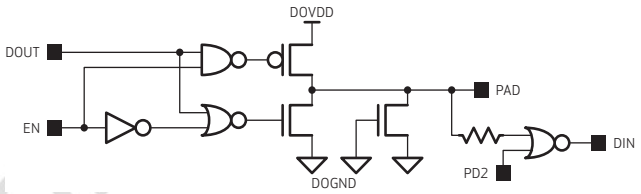
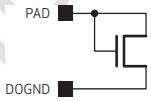
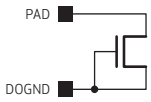
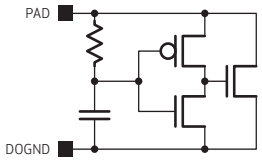
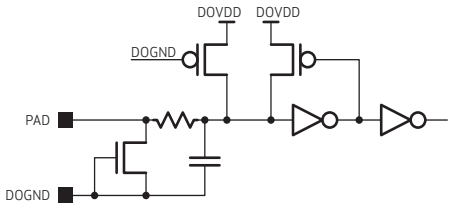
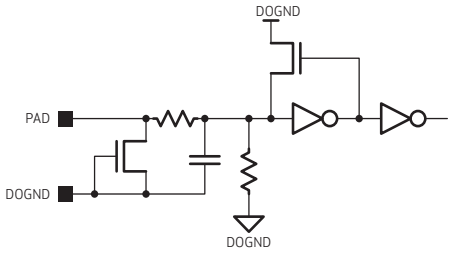
symbol	equivalent circuit
EXTCLK	
SIOD	

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

symbol	equivalent circuit
SIOC	
VSYNC, STROBE, IL_PWM, FSIN, GPIO, HREF	
VN	
MDP3, MDP2, MDP1, MDP0, MDN3, VH, MDN2, MDN1, MDN0, MCP, MCN, EGND, AGND, DOGND	
AVDD, EVDD, DVDD, DOVDD, PVDD	
PWDNB	
XSHUTDOWN, SID, TM	

2 system level description

2.1 overview

The OV8865 color image sensor is a high performance, 8 megapixel RAW image sensor that delivers 3264X2448 at 30 fps using improved OmniBSI-2™ pixel technology. It provides options for multiple resolutions while maintaining full field of view. Users can program image resolution, frame rate, image quality parameters. Camera functions are controlled using the industry standard serial camera control bus (SCCB).

The OV8865 is capable of delivering 30 fps at full resolution allowing burst photography at full 8 megapixel resolution. With a complete 8 megapixel image array, the OV8865 contains all the image management functions to ensure high quality imaging solutions for high resolution digital still camera (DSC), HD camcorders and mobile handsets.

All required image processing functions are programmable through the SCCB interface. In addition, OmniVision image sensors utilize 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, fully stable, color image.

2.2 architecture

The OV8865 sensor core generates streaming pixel data at a constant frame rate. **figure 2-1** shows the functional block diagram of the OV8865 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 10-bit data for each pixel in the array.

figure 2-1 OV8865 block diagram

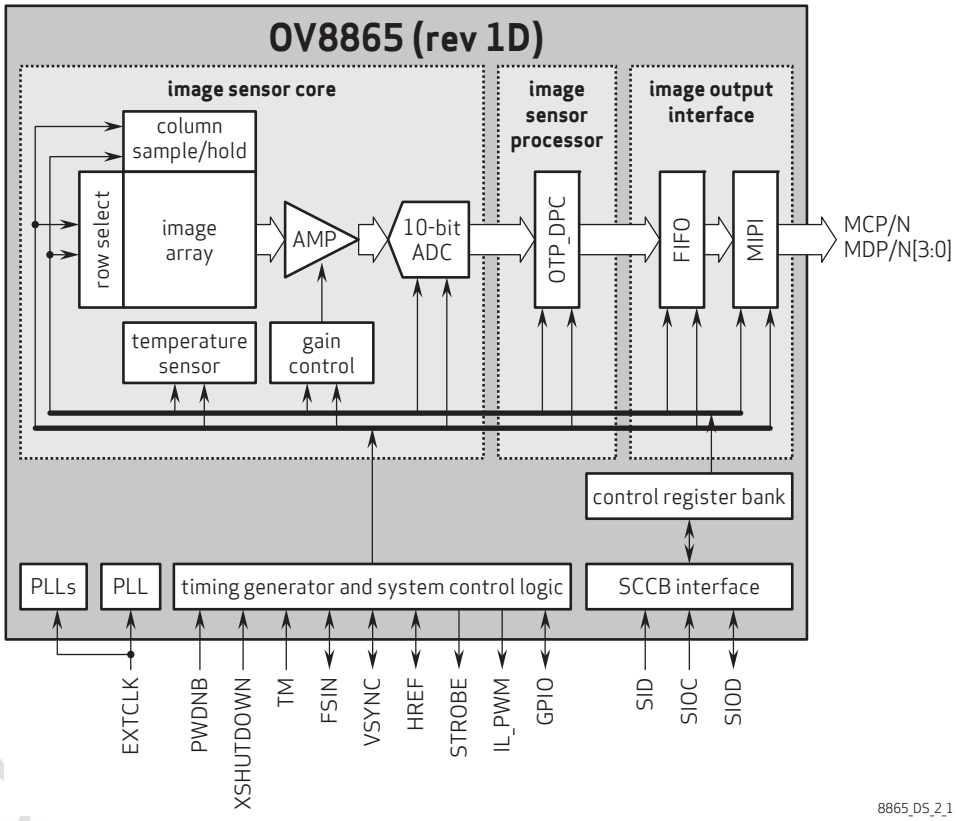
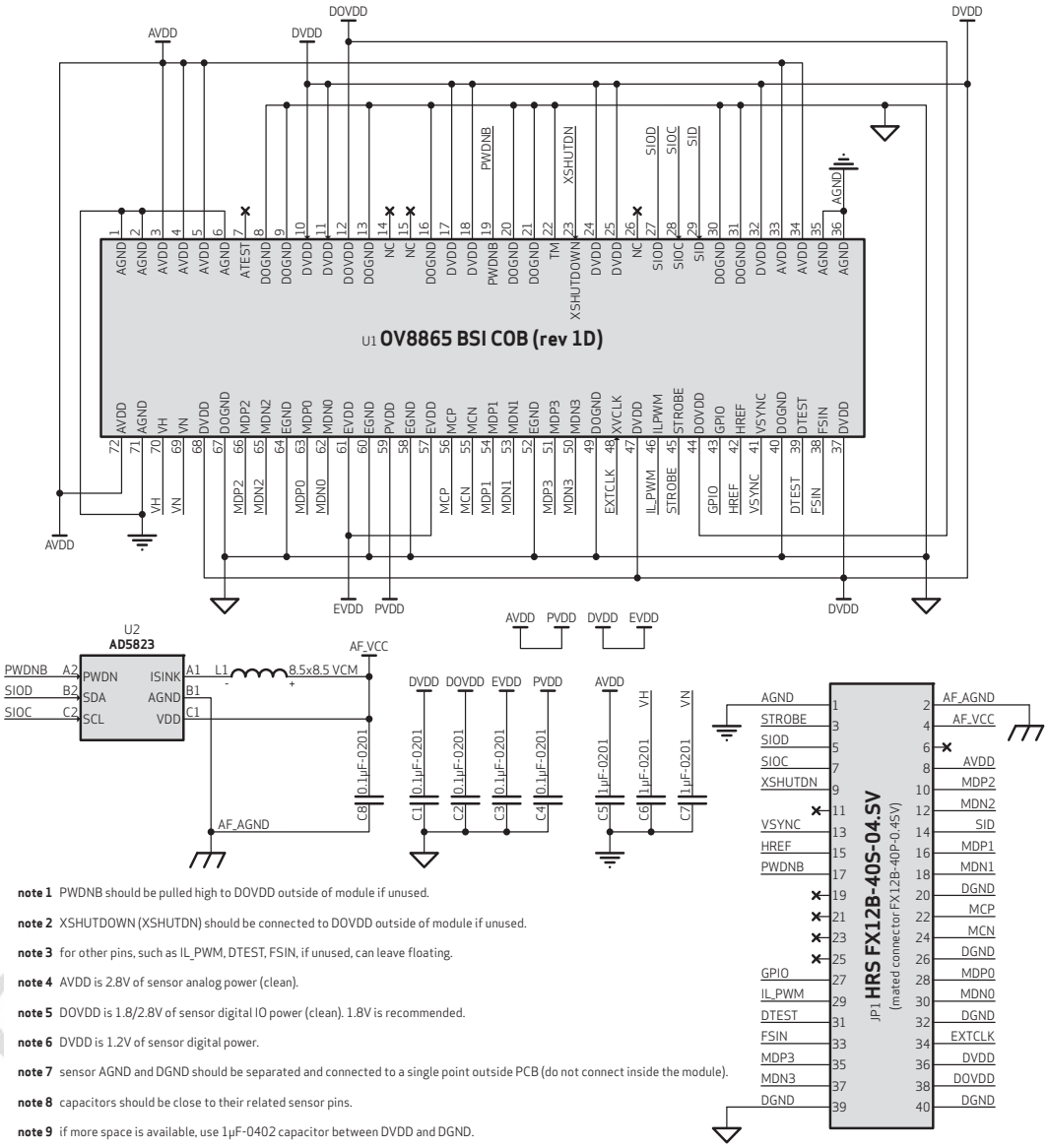


figure 2-2 OV8865 reference schematic



note 1 PWDNB should be pulled high to DOVDD outside of module if unused.

note 2 XSHUTDOWN (XSHUTDN) should be connected to DOVDD outside of module if unused.

note 3 for other pins, such as IL_PWM, DTEST, FSIN, if unused, can leave floating.

note 4 AVDD is 2.8V of sensor analog power (clean).

note 5 DOVDD is 1.8/2.8V of sensor digital IO power (clean). 1.8V is recommended.

note 6 DVDD is 1.2V of sensor digital power.

note 7 sensor AGND and DGND should be separated and connected to a single point outside PCB (do not connect inside the module).

note 8 capacitors should be close to their related sensor pins.

note 9 if more space is available, use 1µF-0402 capacitor between DVDD and DGND.

note 10 EVDD/EGND are power/ground for MIPI core.
MCP and MCN are MIPI clock lane positive and negative output.
MDPx and MDNx are MIPI data lane positive and negative output.

note 11 traces of MCP, MCN, MDPx and MDNx should have the same or similar length.
differential impedance of the clock pair and data pair transmission lines should be controlled under 100 Ohm.

note 12 SID pin should be pulled low for device address 0x20 and pulled high for device address 0x6C.

note 13 all NC pins can be left floating or connected to GND if needed.

note 14 AF_VCC and AF_AGND is the power supply for auto focus related circuitry.
although AF_VCC is 2.8 - 3.3V, it is recommended to use 3.3V to have better auto focus performance.

note 15 AD5823 can be used to replace AD5820.

8865_COB_DS_2_2

2.3 format and frame

The OV8865 supports RAW RGB output with one/two/four lane MIPI interface.

table 2-1 non HDR mode frame rate

resolution	10-bit output	10-bit output MIPI 4 lanes	methodology
3264x2448	30 fps	960 Mbps/lane	full
3264x1836	30 fps	960 Mbps/lane	cropping
2816x1584	30 fps	960 Mbps/lane	cropping
1632x1224	60 fps	960 Mbps/lane	2x2 fast binning
1408x792	60 fps	960 Mbps/lane	2x2 fast binning

2.4 I/O control

The OV8865 can configure its I/O pad 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

function	register	description
output drive capability control	0x3011	Bit[6:5]: pad I/O drive capability 00: 1x 01: 2x 10: 3x 11: 4x
HREF I/O control	0x3002	Bit[6]: HREF output enable 0: input 1: output
HREF output select	0x3010	Bit[6]: enable HREF as GPIO controlled by register
HREF output value	0x300D	Bit[6]: register control HREF output
GPIO I/O control	0x3002	Bit[0]: GPIO0 output enable 0: input 1: output
GPIO output value	0x300D	Bit[0]: register control GPIO output
VSYNC I/O control	0x3002	Bit[7]: VSYNC output enable 0: input 1: output
VSYNC output select	0x3010	Bit[7]: enable VSYNC as GPIO controlled by register
VSYNC output value	0x300D	Bit[7]: register control VSYNC output

2.5 MIPI interface

The OV8865 supports a MIPI interface of up to 4-lanes. The MIPI interface can be configured for 1/2/4-lane and each lane is capable of a data transfer rate of up to 1.2 Gbps.

2.6 LVDS interface

The OV8865 supports a 4-lane LVDS interface. The LVDS interface can be only configured for 4-lane and each lane is capable of a data transfer rate of up to 1.2 Gbps. The LVDS has the same high speed electrical characteristics as MIPI. The main features of the support lanes are:

- supports 4 lanes
- supports only one sync code split in every line or four byte sync code every lane
- SAV or EAV first
- embedded channel id in SAV or EAV mode support

2.6.1 LVDS sync mode

Set register 0x4B00[0] to 1'b1 to enable synchronize code every lane mode. Each lane has 4 bytes of synchronizing.

```
Lane0: FF 00 00 SAV P0        ...      Pn-4 FF 00 00 00 EAV

Lane1: FF 00 00 SAV P1        ...      Pn-3 FF 00 00 00 EAV

Lane2: FF 00 00 SAV P2        ...      Pn-2 FF 00 00 00 EAV

Lane3: FF 00 00 SAV P3        ...      Pn-1 FF 00 00 00 EAV
```

2.6.2 split synchronize code

Set register 0x4B00[0] to 1'b0 to enable split synchronize code mode. The lane amount differs for each lane amount and 4 bytes are inserted for different lane amounts.

```
four lane case

Lane0: FF P0 P4       ...      Pn-4 FF

Lane1: 00 P1 P5       ...      Pn-3 00

Lane2: 00 P2 P6       ...      Pn-2 00

Lane3: SAV P3 P7      ...      Pn-1 EAV
```

2.6.3 embedded channel ID in SAV or EAV

Set 0x4B00[3] to 1'b1 to enable the embedded channel ID mode. In normal mode, the sync code structure is:

SAV/EAV: 1'b1, F, V, H, V^H, F^H, F^V, F^V^H.

F: Field sync control by register 0x4b00[2];

V: 1, vertical blank

H: 1, Horizontal blank.

For 8-bit mode : sync code = SAV/EAV

For 10-bit mode : sync code = SAV/EAV, 2'b00.

For 12-bit mode : sync code = SAV/EAV, 4'b0000.

But in embedded channel ID mode, the last 4 digits of SAV/EAV will be replaced by lane ID:

LANE0 : sync code = 1'b1, F, V, H, 4'b0000;

LANE1 : sync code = 1'b1, F, V, H, 4'b0001;

LANE2 : sync code = 1'b1, F, V, H, 4'b0010;

LANE3 : sync code = 1'b1, F, V, H, 4'b0011;

2.7 power management

Based on the system power configuration (XSHUTDOWN, PWDNB control), the power up sequence will be different. OmniVision recommends cutting off all power supplies, including the external DVDD, when the sensor is not in use.

2.7.1 power up sequence

To avoid any glitch from a strong external noise source, OmniVision recommends controlling XSHUTDOWN or PWDNB by GPIO and tying the other pin to DOVDD.

Whether or not XSHUTDOWN is controlled by GPIO, the XSHUTDOWN rising cannot occur before AVDD and DOVDD.

table 2-3 power up sequence

case	XSHUTDOWN	PWDNB	power up sequence requirement
1	GPIO	DOVDD	Refer to **figure 2-3** 1. DOVDD rising must occur before DVDD rising 2. AVDD rising can occur before or after DOVDD rising 3. AVDD must occur before DVDD 4. XSHUTDOWN rising must occur after AVDD, DOVDD and DVDD are stable
2	DOVDD	GPIO	Refer to **figure 2-4** 1. AVDD rising occurs before DOVDD rising 2. DOVDD rising occurs before DVDD 3. PWDNB rising occurs after DVDD rising

table 2-4 power up sequence timing constraints

constraint	label	min	max	unit
AVDD rising – DOVDD rising	t0	0	∞	ns
DOVDD rising – AVDD rising	t1			ns
XSHUTDOWN rising – first SCCB transaction	t2	8192		EXTCLK cycles
minimum number of EXTCLK cycles prior to the first SCCB transaction	t3	8192		EXTCLK cycles
PLL start up/lock time	t4		0.2	ms
entering streaming mode – first frame start sequence (fixed part)	t5		10	ms
entering streaming mode – first frame start sequence (variable part)	t6	delay is the exposure time value		lines
AVDD or DOVDD, whichever is last – DVDD	t7	0	∞	ns
DVDD – PWDNB rising	t8	0	∞	ns
DVDD – XSHUTDOWN rising	t9	0	∞	ns

figure 2-3 power up sequence (case 1)

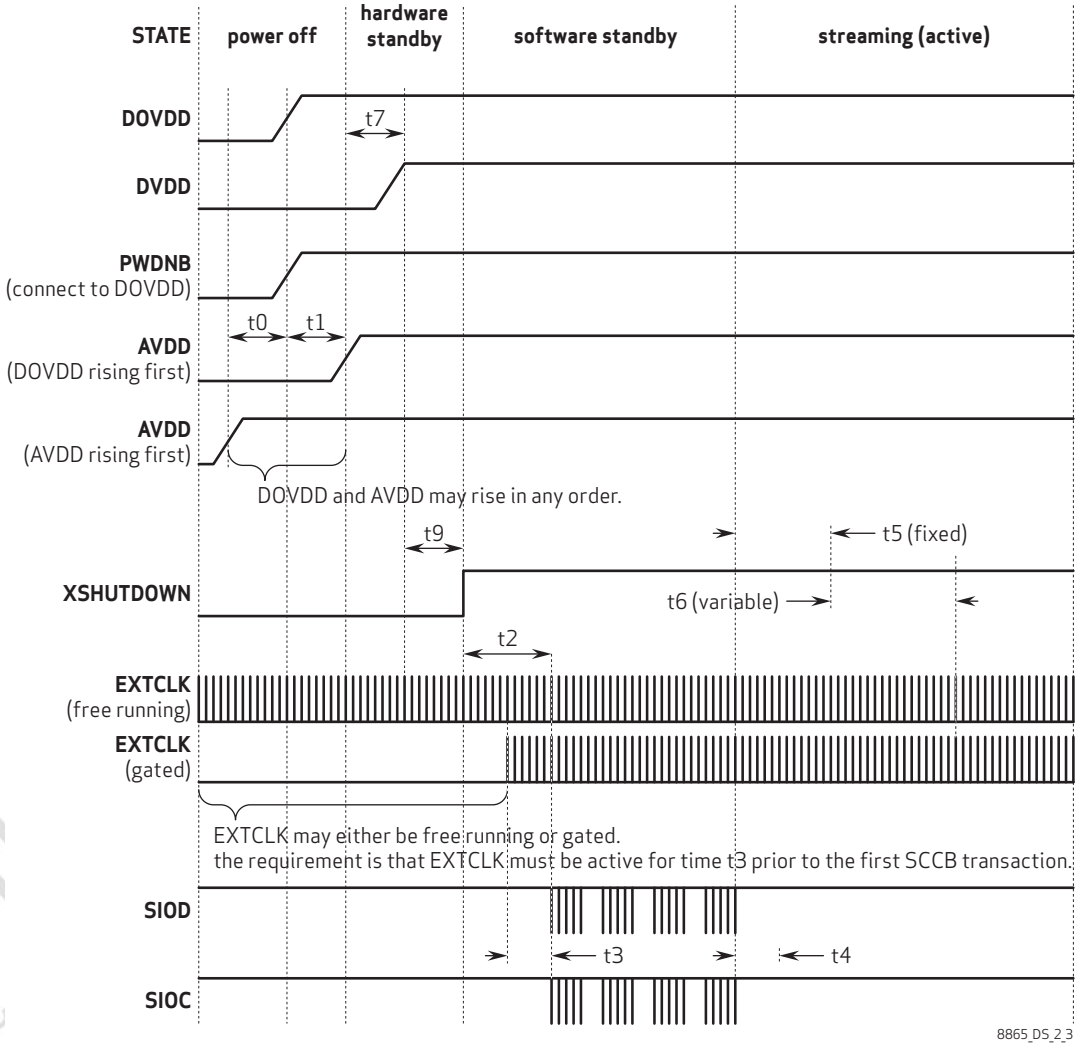
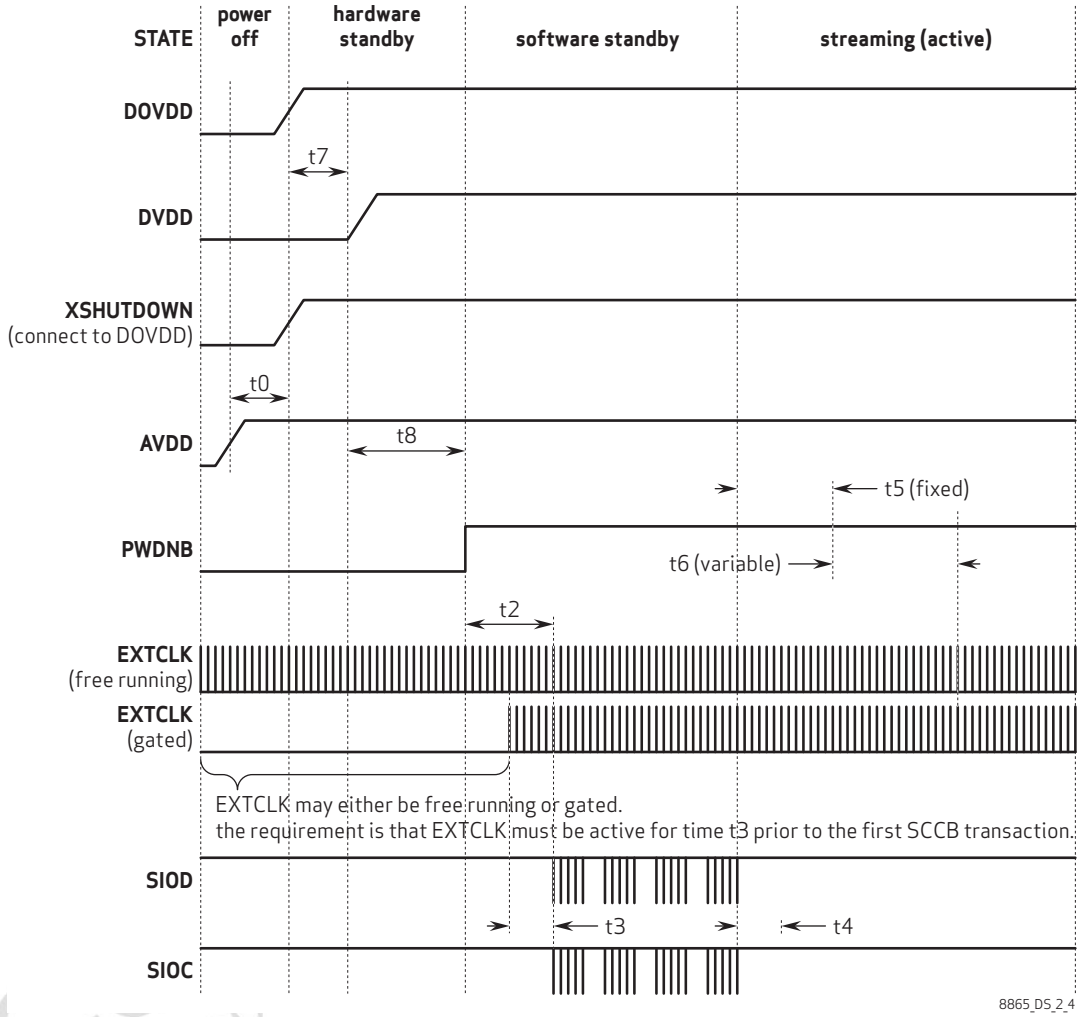


figure 2-4 power up sequence (case 2)



2.7.2 power down sequence

The digital and analog supply voltages can be powered down in any order (e.g. DOVDD, then AVDD or AVDD, then DOVDD). Similar to the power up sequence, the EXTCLK input clock may be either gated or continuous. To avoid bad frames from the MIPI, OmniVision recommends to using group hold to send SCCB sleep command.

table 2-5 power down sequence

case	XSHUTDOWN	PWDNB	power down sequence requirement
1	GPIO	DOVDD	Refer to **figure 2-6** 1. software standby recommended 2. pull XSHUTDOWN low for minimum power consumption 3. cut off DVDD 4. pull AVDD and DOVDD low in any order
2	DOVDD	GPIO	Refer to **figure 2-7** 1. software standby recommended 2. pull PWDNB low for minimum power consumption 3. cut off DVDD 4. pull DOVDD low (XSHUTDOWN connected to DOVDD) 5. pull AVDD low

table 2-6 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 the MIPI end code before entering the software for standby; otherwise, enter the software standby mode immediately		
minimum of EXTCLK cycles after the last SCCB transaction or MIPI frame end	t1	512		EXTCLK cycles
last SCCB transaction or MIPI frame end, XSHUTDOWN falling	t2	512		EXTCLK cycles
XSHUTDOWN falling – AVDD falling or DOVDD falling whichever is first	t3	0.0		ns
AVDD falling – DOVDD falling	t4	AVDD and DOVDD may fall in any order, the falling separation can vary from 0 ns to infinity		ns
DOVDD falling – AVDD falling	t5			ns
PWDNB falling – DOVDD falling	t6	0.0		ns
XSHUTDOWN falling – DVDD falling	t7	0.0		ns
DVDD falling – AVDD falling or DOVDD falling whichever is first	t8	0.0		ns
PWDNB falling – DVDD falling	t9	0.0		ns

figure 2-5 software standby sequence

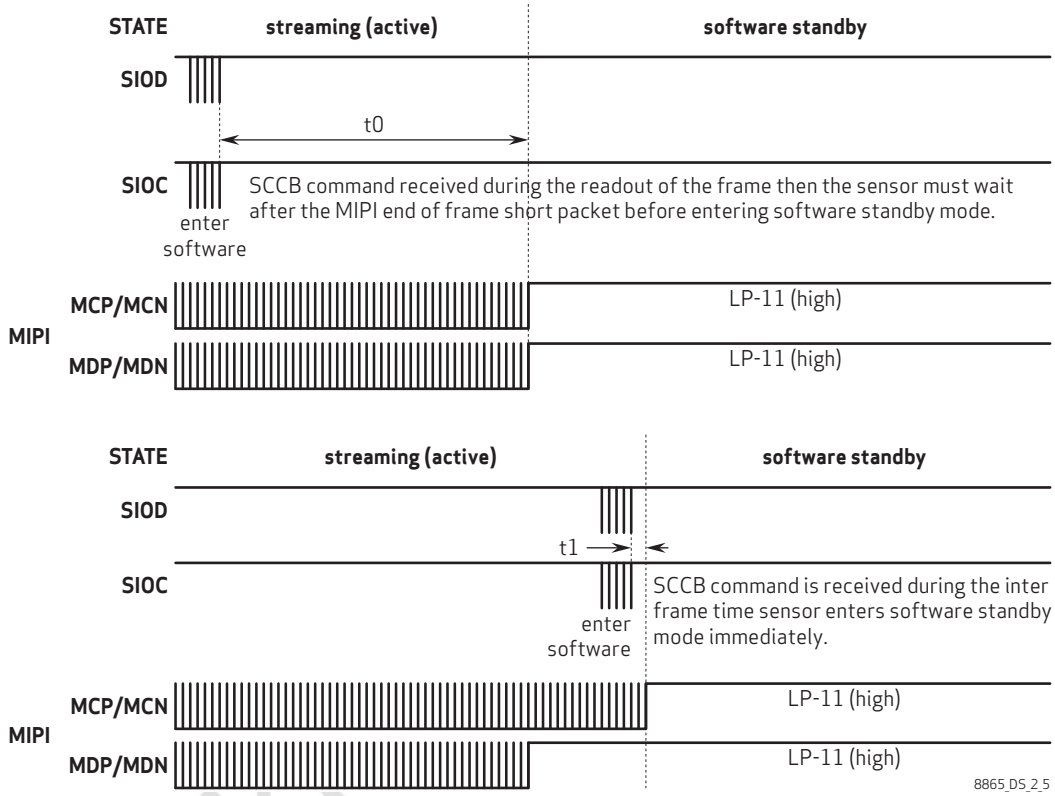


figure 2-6 power down sequence (case 1)

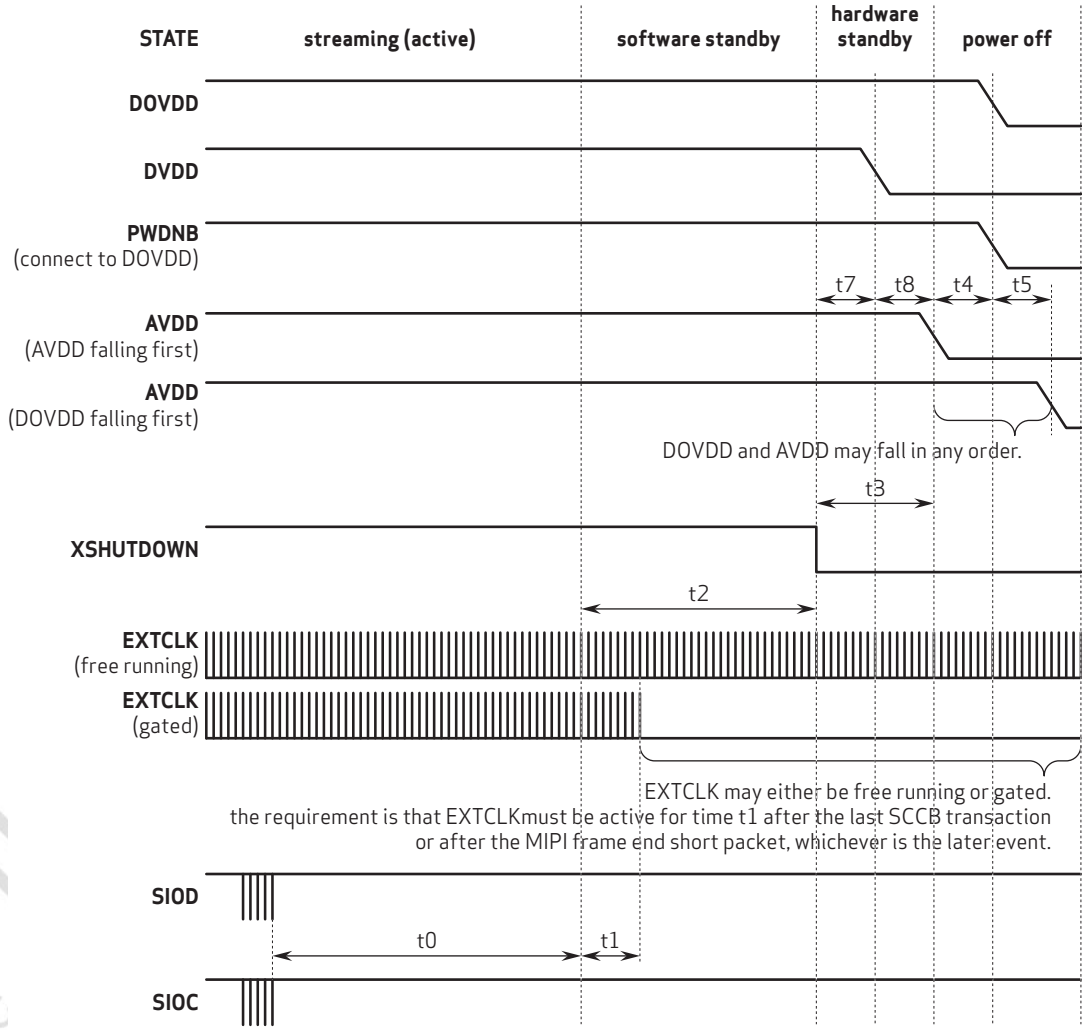
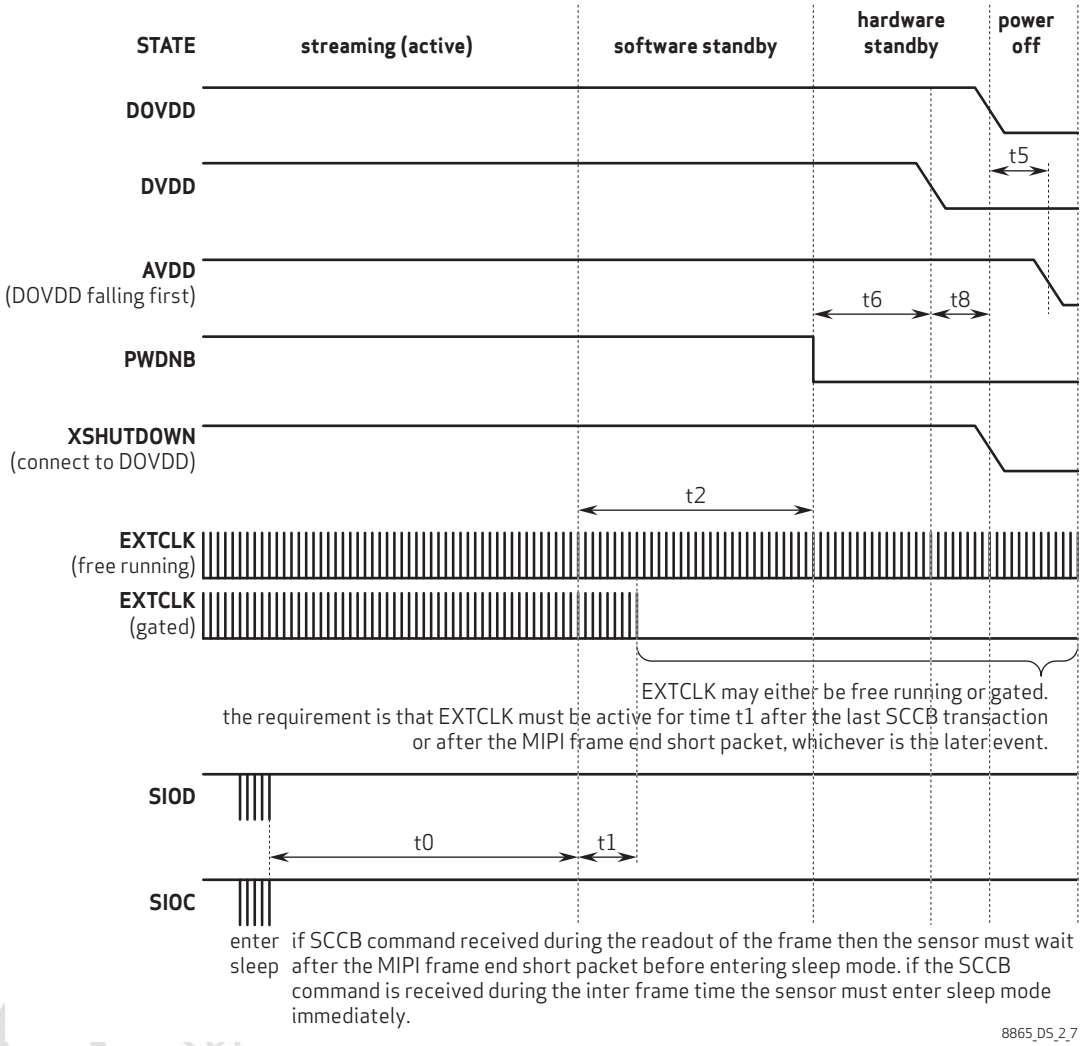


figure 2-7 power down sequence (case 2)



2.8 reset

The whole chip will be reset during power up. Manually applying a hardware reset (XSHUTDOWN=0) upon power up is recommended even though the on-chip power up reset is included. The hardware reset is active low with an asynchronized design. The reset pulse width should be greater than or equal to 2 ms.

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 OV8865:

- hardware standby
- software standby

2.9.1 hardware standby

To initiate hardware standby mode, the XSHUTDOWN or PWDNB pin must be tied to low. When this occurs, the OV8865 internal device clock is halted even when the external clock source is still clocking and all internal counters are reset.

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 2-7 hardware and standby description (sheet 1 of 2)

mode	description
hardware standby with PWDNB	1. enabled by pulling PWDNB low 2. input clock is gated by PWDNB, no SCCB communication 3. register values are maintained 4. power down all blocks and regulator 5. low power consumption 6. GPIO can be configured as high/low/tri-state
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

table 2-7 hardware and standby description (sheet 2 of 2)

mode	description
software standby	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

2.10 system clock control

2.10.1 PLL1

The PLL1 generates a default 75 MHz pixel clock and 600 MHz MIPI serial clock from a 6~27 MHz input clock. The VCO range is from 500 MHz to 1200 MHz. A programmable clock provided is needed to generate different frequencies.

2.10.2 PLL2

The PLL2 generates a default 120 MHz system clock from a 6~27 MHz input clock. The VCO range is from 500 MHz to 1200 MHz. A programmable clock divider is provided to generate different frequencies.

figure 2-8 clock scheme

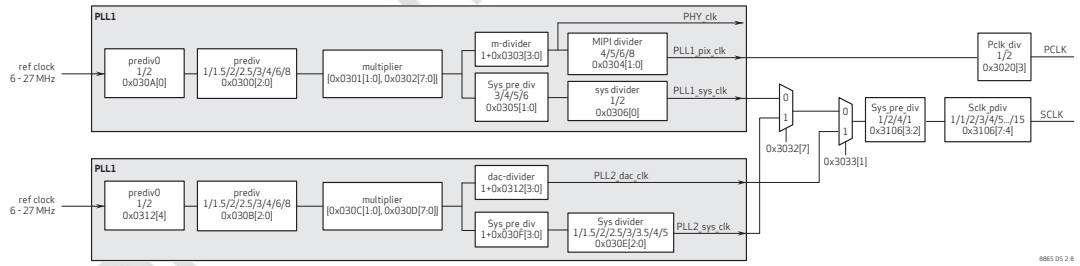


table 2-8 PLL registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x0300	PLL_CTRL_0	0x00	RW	Bit[2:0]: pll1_pre_div
0x0301	PLL_CTRL_1	0x00	RW	Bit[1:0]: pll1_multiplier[9:8]
0x0302	PLL_CTRL_2	0x19	RW	Bit[7:0]: pll1_multiplier[7:0]
0x0303	PLL_CTRL_3	0x00	RW	Bit[3:0]: pll1_divm
0x0304	PLL_CTRL_4	0x03	RW	Bit[1:0]: pll1_div_mipi
0x0305	PLL_CTRL_5	0x01	RW	Bit[1:0]: pll1_div_sp

table 2-8 PLL registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x0306	PLL_CTRL_6	0x01	RW	Bit[0]: pll1_div_s
0x0308	PLL_CTRL_8	0x00	RW	Bit[0]: pll1_bypass
0x0309	PLL_CTRL_9	0x01	RW	Bit[2:0]: pll1_cp
0x030A	PLL_CTRL_A	0x00	RW	Bit[0]: pll1_predivp
0x030B	PLL_CTRL_B	0x00	RW	Bit[2:0]: pll2_pre_div
0x030C	PLL_CTRL_C	0x00	RW	Bit[1:0]: pll2_r_divp[9:8]
0x030D	PLL_CTRL_D	0x1E	RW	Bit[7:0]: pll2_r_divp[7:0]
0x030E	PLL_CTRL_E	0x02	RW	Bit[2:0]: pll2_r_divs
0x030F	PLL_CTRL_F	0x02	RW	Bit[3:0]: pll2_r_divsp
0x0310	PLL_CTRL_10	0x01	RW	Bit[2:0]: pll2_r_cp
0x0311	PLL_CTRL_11	0x00	RW	Bit[0]: pll2_bypass
0x0312	PLL_CTRL_12	0x01	RW	Bit[4]: pll2_pre_div0 Bit[3:0]: pll2_r_divdac

table 2-9 sample PLL configuration (sheet 1 of 2)

control name	address	input clock (EXTCLK)	
		24 MHz	6 MHz
PLL1_PREDIVP	0x030A[0]	0x0	0x0
PLL1_PREDIV	0x0300[2:0]	0x0	0x0
PLL1_MULTIPLIER	{0x0301[1:0], 0x0302[7:0]}	0x19	0x64
PLL1_DIV_MIPI	0x0304[1:0]	0x3	0x3
PLL1_DIVM	0x0303[3:0]	0x0	0x0
PLL1_DIVSP	0x0305[1:0]	0x1	0x1
PLL1_DIVS	0x0306[0]	0x1	0x1
PLL2_PREDIVP	0x0311[0]	0x0	0x0
PLL2_PREDIV	0x030B[2:0]	0x0	0x0
PLL2_MULTIPLIER	{0x030C[1:0], 0x030D[7:0]}	0x1E	0x78
PLL2_DIVSP	0x030F[3:0]	0x2	0x2
PLL2_DIVS	0x030E[2:0]	0x2	0x2

table 2-9 sample PLL configuration (sheet 2 of 2)

control name	address	input clock (EXTCLK)	
		24 MHz	6 MHz
SCLK	–	120MHz	120MHz
PHY_SCLK	–	600MHz	600MHz
MIPI_PCLK	–	75MHz	75MHz

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 OV8865, the SCCB ID is controlled by the SID pin. If SID is low, the sensor's SCCB ID is 0x6C. If SID is high, the sensor's SCCB ID is 0x20. 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.

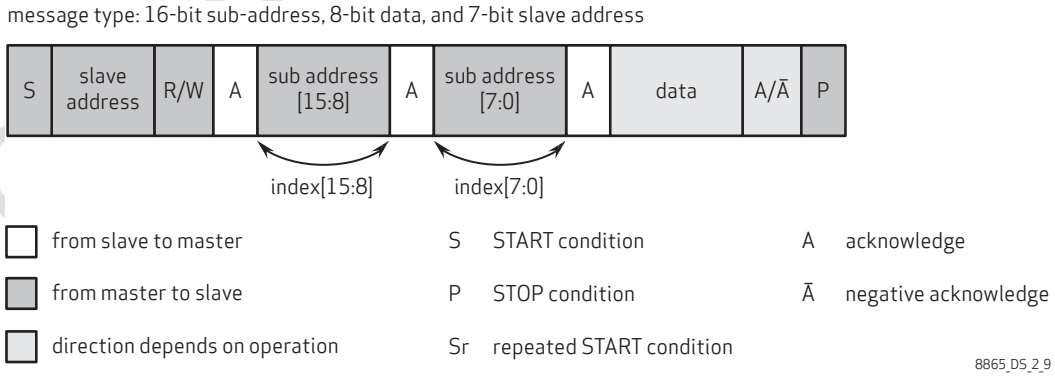
2.11.1 data transfer protocol

The data transfer of the OV8865 follows the SCCB protocol.

2.11.2 message format

The OV8865 supports the message format shown in **figure 2-9**. The repeated START (Sr) condition is not shown in **figure 2-10**, but is shown in **figure 2-11** and **figure 2-12**.

figure 2-9 message type



2.11.3 read / write operation

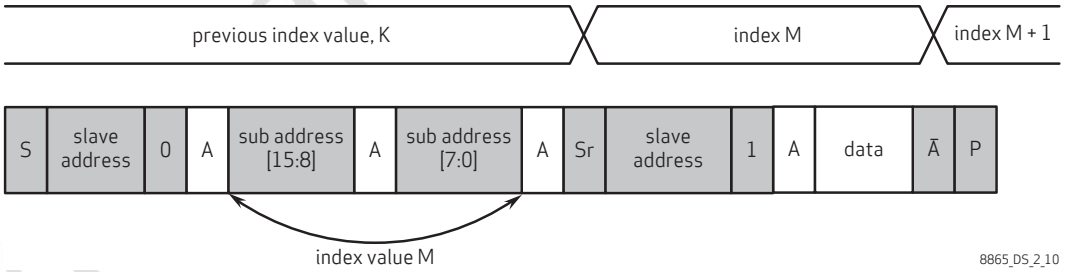
The OV8865 supports four different read operations and two different write operations:

- 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.

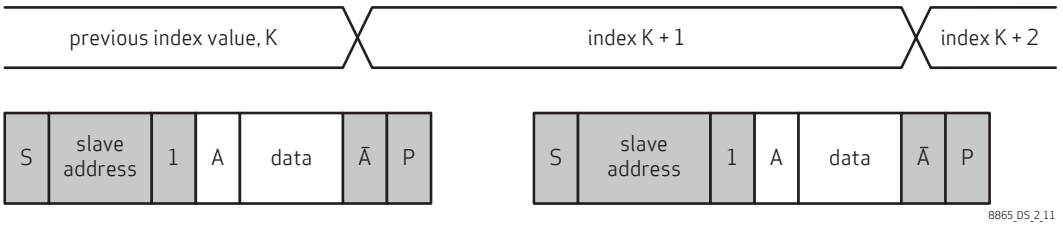
In a single read from random locations, the master does a dummy write operation to desired sub-address, issues a repeated start condition and then addresses the camera again with a read operation. After acknowledging its slave address, the camera starts to output data onto the SIOD line as shown in **figure 2-10**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-10 SCCB single read from random location



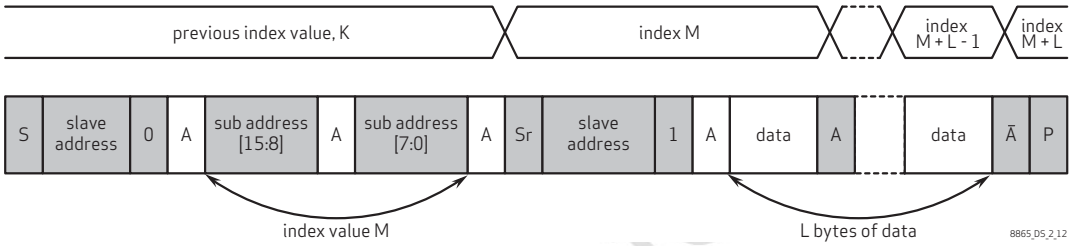
If the host addresses the camera with read operation directly without the dummy write operation, the camera responds by setting the data from last used sub-address to the SIOD line as shown in **figure 2-11**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-11 SCCB single read from current location



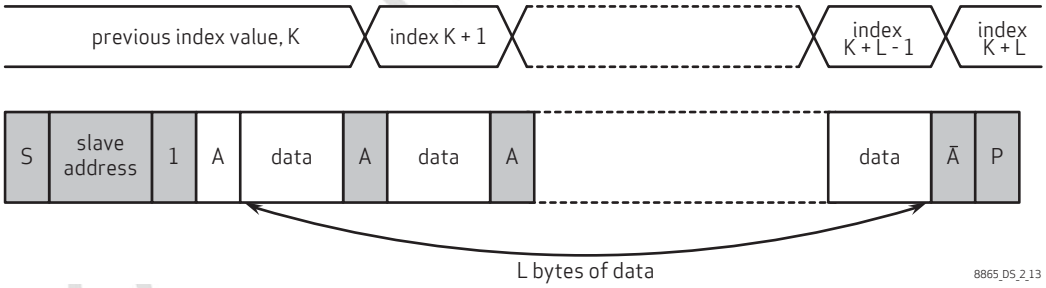
The sequential read from a random location is illustrated in **figure 2-12**. The master does a dummy write to the desired sub-address, issues a repeated start condition after acknowledge from slave and addresses the slave again with read operation. If a master issues an acknowledge after receiving data, 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 2-12 SCCB sequential read from random location



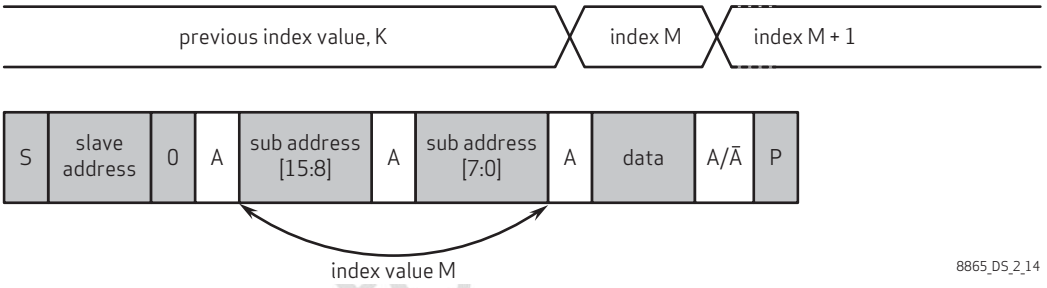
The sequential read from current location is similar to a sequential read from a random location. The only exception is that there is no dummy write operation, as shown in **figure 2-13**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-13 SCCB sequential read from current location



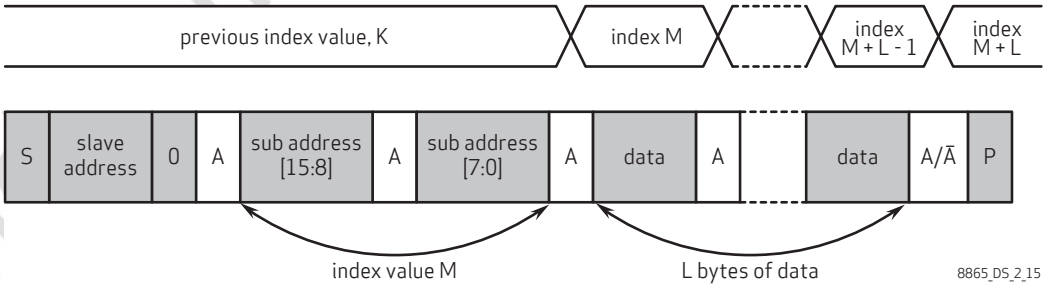
The write operation to a random location is illustrated in **figure 2-14**. The master issues a write operation to the slave, sets the sub-address and data correspondingly after the slave has acknowledged. The write operation is terminated with a stop condition from the master.

figure 2-14 SCCB single write to random location



The sequential write is illustrated in **figure 2-15**. The slave automatically increments the sub-address after each data byte. The sequential write operation is terminated with stop condition from the master.

figure 2-15 SCCB sequential write to random location



2.11.4 SCCB timing

figure 2-16 SCCB interface timing

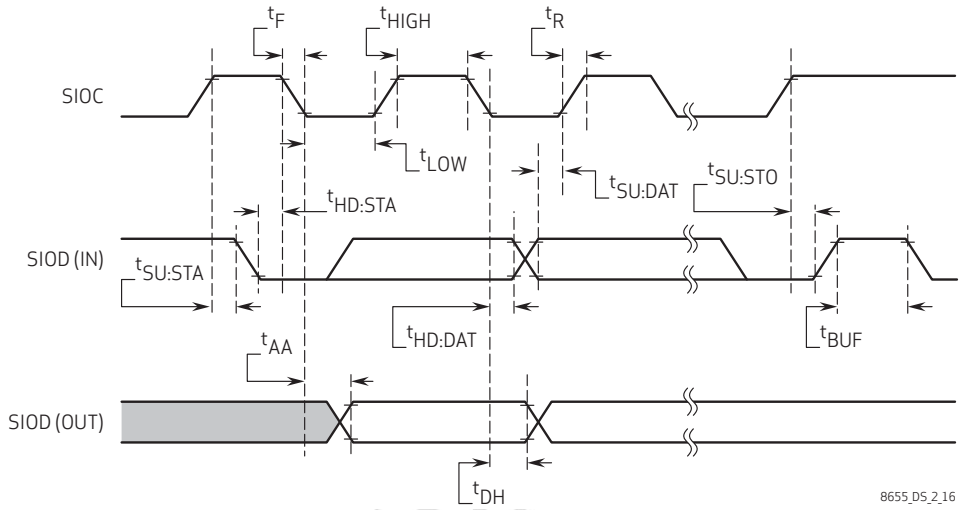


table 2-10 SCCB interface timing specifications[ab]

symbol	parameter	min	typ	max	unit
f_{SIOC}	clock frequency			400	kHz
t_{LOW}	clock low period	1.3			µs
t_{HIGH}	clock high period	0.6			µs
t_{AA}	SIOC 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 the beginning of the rising edge and/or of the falling edge signifies 30%, timing measurement shown in the middle of the rising/falling edge signifies 50%, timing measurement shown at the beginning of the rising edge and/or of the falling edge signifies 70%

2.12 group write

Group write is supported in order to update a group of registers (except 0x31xx) in the same frame. These registers are guaranteed to be written prior to the internal latch at the frame boundary. If more than one group is going to be launched, the second group cannot be recorded or launched before the first group has effectively been launched

The OV8865 supports up to four groups. These groups share 1024 bytes of memory and the size of each group is programmable by adjusting the start address.

table 2-11 context switching control

address	register name	default value	R/W	description
0x3208	GROUP ACCESS	–	W	Group Access Bit[7:4]: group_ctrl 0000: Group hold start 0001: Group hold end 1010: Group delay launch 1110: Group quick launch Others: Debug mode Bit[3:0]: Group ID 0000: Group bank 0, default start from address 0x00 0001: Group bank 1, default start from address 0x40 0010: Group bank 2, default start from address 0x80 0011: Group bank 3, default start from address 0xB0 Others: Debug mode
0x3209	GRP0_PERIOD	0x00	RW	Frames For Staying in First Group (must be Group 0) 0 Means Always Stay in Group 0
0x320A	GRP1_PERIOD	0x00	RW	Frames For Staying in Second Group (can be Group 1-3) 0 Means Always Stay in Group 1
0x320B	GRP_SWCTRL	0x01	RW	Bit[7]: Auto switch Bit[3]: group_switch_repeat_en Enable the first group (group 0) and second group repeatable switch Bit[2]: context_en Enable to switch from second group back to first group (group 0) automatically Bit[1:0]: Second group selection
0x320D	GRP_ACT	–	R	Indicates Which Group is Active
0x320E	FRAME_CNT_GRP0	–	R	frame_cnt_grp0
0x320F	FRAME_CNT_GRP1	–	R	frame_cnt_grp1

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  3208  00    group 0 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  320B  00    manual launch on
6C  3208  E0    quick launch group 0
```

2.12.2.2 launch mode 2 - delay manual launch

Delay manual launch is achieved by writing to the 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  320B  00    manual launch on
6C  3208  A1    delay launch group 1
```

2.12.2.3 launch mode 3 - quick auto launch

Quick auto launch works like the mode 1, the difference is it will return to a specified group automatically. This is controlled by the register 0x3209, where bit[6:5] controls which group to return and bit[4:0] controls how many frames to stay before returning. The auto launch enable bit is the 0x320B[7].

The operation can be better understood with a setting example:

```
6C  3209  44    Bit[6:5]: 2, return to group 2, Bit[4:0]: 4: stay 4 frames
6C  320B  80    auto launch on
6C  3208  E0    quick launch group 0
```

In this example, sensor will quick launch group 0, stay at group 0 for 4 frames, then return to group 2 after that.

2.12.2.4 launch mode 4: delay auto launch

Delay auto launch works like mode 2 in the delay launch part and like the mode 3 in the return part.

The operation can be better understood with a setting example:

```
6C  3209  44    Bit[6:5]: 2, return to group 2, Bit[4:0]: 4: stay 4 frames
6C  320B  80    auto launch on
6C  3208  A0    delay launch group 0
```

In this example, sensor will delay launch group 0, stay at group 0 for 4 frames, then return to group 2 after that.

2.12.2.5 launch mode 5: repeat launch

Repeat launch is controlled by registers 0x3209, 0x320A, and 0x320B. In this mode, the launch is repeated automatically between the first group (must be group 0) and the second group (can be either one of groups 1-3, which is specified by register 0x320B[1:0]). The register 0x3209 defines how many frames remain at group 0, and register 0x320A defines how many frames remain at the second group.

The operation can be better understood with a setting example:

```
6C  3209  02    Bit[7:0]: 2, stay 2 frames in group 0
6C  320A  03    Bit[7]: 3, stay 3 frames in the second group
6C  320B  0E    Bit[3:2]: 3, repeat launch on, Bit[1:0]: 2, second group select:
                group 2
6C  3208  A0    always use a0 for repeat launch
```

In this example, sensor will delay launch group 0, stay at group 0 for 2 frames, then switch to group 2 for 3 frames, then back to group 0 for 2 frames, group 2 for 3 frames and so on.

Below is another example to apply launch mode 2 (delay manual launch) first, sensor stays at group 2 for an indefinite number of frames, then apply launch mode 5 (repeat launch). The sensor will switch to group 0 for 2 frames, then group 2 for 3 frames, and so on.

```
6C  320B  00    manual launch on
6C  3208  A2    delay launch group 2 stay at group 2 for indefinite frames
6C  3209  02    Bit[7:0]: 2, stay 2 frames in group 0
6C  320A  03    Bit[7:0]: 3, stay 3 frames in the second group
6C  320B  0E    Bit[3:2]: 3, repeat launch on, Bit[1:0]: 2, second group select:
                group 2
6C  3208  A0    always use A0 for repeat launch
```

Switch to group 0 for 2 frames, then group 2 for 3 frames, and so on.

2.13 register re-mapping

The OV8865 supports register re-mapping function to re-map a source address to a continuous destination one. One use is to make some discontinuous address to a consecutive address, so the user can use sequential write through SCCB to access these registers. This will speed up the SCCB access. The OV8865 supports up to 32 registers which can be re-mapped to a continuous address.

```
6C  3101  32 ; Bit[5] enable re-mapping function
6C  3108  90 ; Destination start address
6C  3109  00
6C  3110  35 ; source address0
6C  3111  00 ;
6C  3112  35 ; source address1
6C  3113  01 ;
6C  3114  35 ; source address2
6C  3115  02 ;
6C  3116  35 ; source address3
6C  3117  08 ;
6C  3118  35 ; source address4
6C  3119  09 ;
```

Then if the user wants to write 0x3500~0x3503 and 0x3508-0x3509:

```
6C  9000  AA; will write 0x3500 to 0xAA
6C  9001  BB; will write 0x3501 to 0xBB
6C  9002  CC; will write 0x3502 to 0xCC
6C  9003  DD; will write 0x3508 to 0xDD
6C  9004  EE; will write 0x3509 to 0xEE
```

table 2-12 register re-mapping (sheet 1 of 3)

address	register name	default value	R/W	description
0x3101	RE-MAP_EN	0x32	RW	Bit[5]: Re-mapping enable
0x3108	DESTINATION RE-MAPPING ADDR H	0x90	RW	High Byte of Destination Address for Re-mapping Function
0x3109	DESTINATION RE-MAPPING ADDR L	0x00	RW	Low Byte of Destination Address for Re-mapping Function
0x3110	SRC ADDR00 H	0x00	RW	High Byte of Number 00 Source Register Address
0x3111	SRC ADDR00 L	0x00	RW	Low Byte of Number 00 Source Register Address
0x3112	SRC ADDR01 H	0x00	RW	High Byte of Number 01 Source Register Address
0x3113	SRC ADDR01 H	0x00	RW	Low Byte of Number 01 Source Register Address
0x3114	SRC ADDR02 H	0x00	RW	High Byte of Number 02 Source Register Address

table 2-12 register re-mapping (sheet 2 of 3)

address	register name	default value	R/W	description
0x3115	SRC ADDR02 L	0x00	RW	Low Byte of Number 02 Source Register Address
0x3116	SRC ADDR03 H	0x00	RW	High Byte of Number 03 Source Register Address
0x3117	SRC ADDR03 L	0x00	RW	Low Byte of Number 03 Source Register Address
0x3118	SRC ADDR04 H	0x00	RW	High Byte of Number 04 Source Register Address
0x3119	SRC ADDR04 L	0x00	RW	Low Byte of Number 04 Source Register Address
0x311A	SRC ADDR05 H	0x00	RW	High Byte of Number 05 Source Register Address
0x311B	SRC ADDR05 L	0x00	RW	Low Byte of Number 05 Source Register Address
0x311C	SRC ADDR06 H	0x00	RW	High Byte of Number 06 Source Register Address
0x311D	SRC ADDR06 L	0x00	RW	Low Byte of Number 06 Source Register Address
0x311E	SRC ADDR07 H	0x00	RW	High Byte of Number 07 Source Register Address
0x311F	SRC ADDR07 L	0x00	RW	Low Byte of Number 07 Source Register Address
0x3120	SRC ADDR08 H	0x00	RW	High Byte of Number 08 Source Register Address
0x3121	SRC ADDR08 L	0x00	RW	Low Byte of Number 08 Source Register Address
0x3122	SRC ADDR09 H	0x00	RW	High Byte of Number 09 Source Register Address
0x3123	SRC ADDR09 L	0x00	RW	Low Byte of Number 09 Source Register Address
0x3124	SRC ADDR0A H	0x00	RW	High Byte of Number 0A Source Register Address
0x3125	SRC ADDR0A L	0x00	RW	Low Byte of Number 0A Source Register Address
0x3126	SRC ADDR0B H	0x00	RW	High Byte of Number 0b Source Register Address
0x3127	SRC ADDR0B L	0x00	RW	Low Byte of Number 0B Source Register Address
0x3128	SRC ADDR0C H	0x00	RW	High Byte of Number 0C Source Register Address
0x3129	SRC ADDR0C L	0x00	RW	Low Byte of Number 0C Source Register Address
0x312A	SRC ADDR0D H	0x00	RW	High Byte of Number 0D Source Register Address
0x312B	SRC ADDR0D L	0x00	RW	Low Byte of Number 0D Source Register Address
0x312C	SRC ADDR0E H	0x00	RW	High Byte of Number 0E Source Register Address
0x312D	SRC ADDR0E L	0x00	RW	Low Byte of Number 0E Source Register Address
0x312E	SRC ADDR0F H	0x00	RW	High Byte of Number 0F Source Register Address
0x312F	SRC ADDR0F L	0x00	RW	Low Byte of Number 0F Source Register Address
0x3130	SRC ADDR10 H	0x00	RW	High Byte of Number 10 Source Register Address
0x3131	SRC ADDR10 L	0x00	RW	Low Byte of Number 10 Source Register Address
0x3132	SRC ADDR11 H	0x00	RW	High Byte of Number 11 Source Register Address

table 2-12 register re-mapping (sheet 3 of 3)

address	register name	default value	R/W	description
0x3133	SRC ADDR11 L	0x00	RW	Low Byte of Number 11 Source Register Address
0x3134	SRC ADDR12 H	0x00	RW	High Byte of Number 12 Source Register Address
0x3135	SRC ADDR12 L	0x00	RW	Low Byte of Number 12 Source Register Address
0x3136	SRC ADDR13 H	0x00	RW	High Byte of Number 13 Source Register Address
0x3137	SRC ADDR13 L	0x00	RW	Low Byte of Number 13 Source Register Address
0x3138	SRC ADDR14 H	0x00	RW	High Byte of Number 14 Source Register Address
0x3139	SRC ADDR14 L	0x00	RW	Low Byte of Number 14 Source Register Address
0x313A	SRC ADDR15 H	0x00	RW	High Byte of Number 15 Source Register Address
0x313B	SRC ADDR15 L	0x00	RW	Low Byte of Number 15 Source Register Address
0x313C	SRC ADDR16 H	0x00	RW	High Byte of Number 16 Source Register Address
0x313D	SRC ADDR16 L	0x00	RW	Low Byte of Number 16 Source Register Address
0x313E	SRC ADDR17 H	0x00	RW	High Byte of Number 17 Source Register Address
0x313F	SRC ADDR17 L	0x00	RW	Low Byte of Number 17 Source Register Address
0x3140	SRC ADDR18 H	0x00	RW	High Byte of Number 18 Source Register Address
0x3141	SRC ADDR18 L	0x00	RW	Low Byte of Number 18 Source Register Address
0x3142	SRC ADDR19 H	0x00	RW	High Byte of Number 19 Source Register Address
0x3143	SRC ADDR19 L	0x00	RW	Low Byte of Number 19 Source Register Address
0x3144	SRC ADDR1A H	0x00	RW	High Byte of Number 1A Source Register Address
0x3145	SRC ADDR1A L	0x00	RW	Low Byte of Number 1A Source Register Address
0x3146	SRC ADDR1B H	0x00	RW	High Byte of Number 1B Source Register Address
0x3147	SRC ADDR1B L	0x00	RW	Low Byte of Number 1B Source Register Address
0x3148	SRC ADDR1C H	0x00	RW	High Byte of Number 1C Source Register Address
0x3149	SRC ADDR1C L	0x00	RW	Low Byte of Number 1C Source Register Address
0x314A	SRC ADDR1D H	0x00	RW	High Byte of Number 1D Source Register Address
0x314B	SRC ADDR1D L	0x00	RW	Low Byte of Number 1D Source Register Address
0x314C	SRC ADDR1E H	0x00	RW	High Byte of Number 1E Source Register Address
0x314D	SRC ADDR1E L	0x00	RW	Low Byte of Number 1E Source Register Address
0x314E	SRC ADDR1F H	0x00	RW	High Byte of Number 1F Source Register Address
0x314F	SRC ADDR1F L	0x00	RW	Low Byte of Number 1F Source Register Address

3 block level description

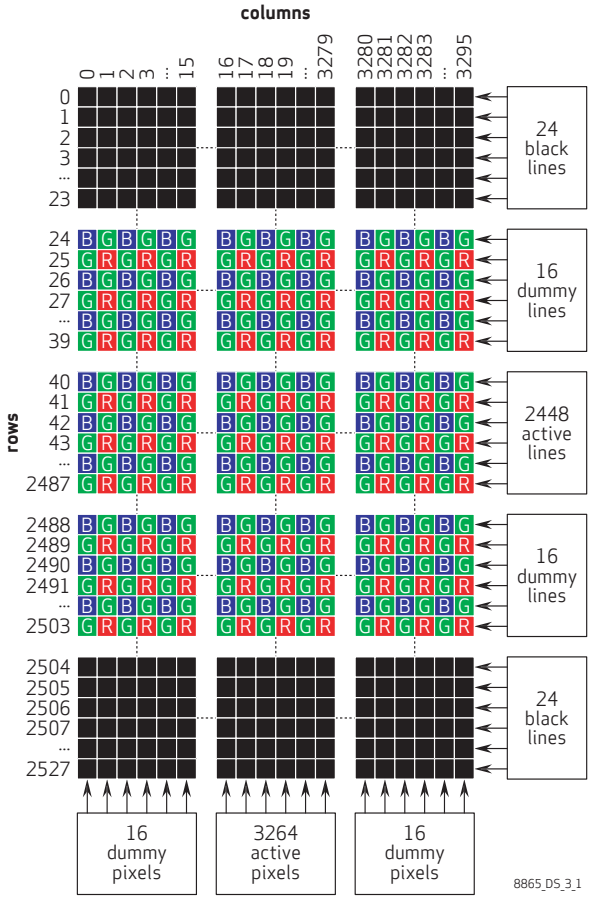
3.1 pixel array structure

The OV8865 sensor has an image array of 3296 columns by 2528 rows (8,332,288 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 8,332,288 pixels, 7,990,272 (3264x2448) are active pixels and can be output. The other pixels are used for black level calibration and interpolation. The center 3264x2448 pixels is suggested to be output from the whole active pixel array. The backend processor can use the boundary pixels for additional processing.

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 array region color filter layout



3.2 subsampling

The OV8865 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 OV8865 supports 2x2 binning, which is illustrated in **figure 3-2**, where the voltage levels of two horizontal (2x1) adjacent same-color pixels are averaged.

figure 3-2 example of 2x2 binning

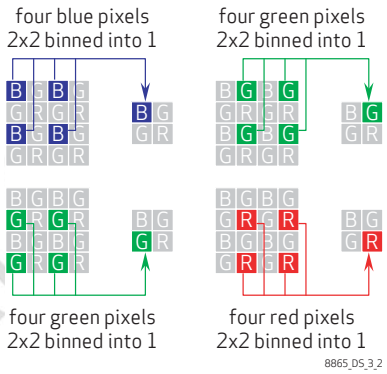


table 3-1 binning-related registers

address	register name	default value	R/W	description
0x3821	TIMING_FORMAT2	0x08	RW	Bit[7]: Vertical sum Bit[5]: Vertical binning Bit[4]: Horizontal binning Bit[3]: ISP horizontal VAR2
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
0x382A	Y_ODD_INC	0x01	RW	Bit[4:0]: Vertical increase number at odd row
0x382B	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 10-bit A/D converters

The balanced signal is then digitized by the on-chip 10-bit ADC.

4 image sensor core digital functions

4.1 mirror and flip

The OV8865 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

original image | mirrored image | flipped image | mirrored and flipped image

8865_DS_4_1

table 4-1 mirror and flip registers

address	register name	default value	R/W	description
0x3820	FORMAT1	0x00	RW	Timing Control Register Bit[2]: Digital vertical flip enable 0: Normal 1: Vertical flip Bit[1]: Array vertical flip enable 0: Normal 1: Vertical flip
0x3821	FORMAT2	0x00	RW	Timing Control Register Bit[2]: Digital horizontal mirror enable 0: Normal 1: Horizontal mirror Bit[1]: Array horizontal mirror enable 0: Normal 1: Horizontal 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.

The OV8865 also supports auto size mode which is controlled by 0x3841[5:0]. Setting it to 0x3F will enable the auto size function and will output the center of the image by default. The user only has to configure the H/V output size (0x3808~0x380B) and registers 0x3842~0x3845 are used to control the offset for auto size mode.

figure 4-2 image windowing

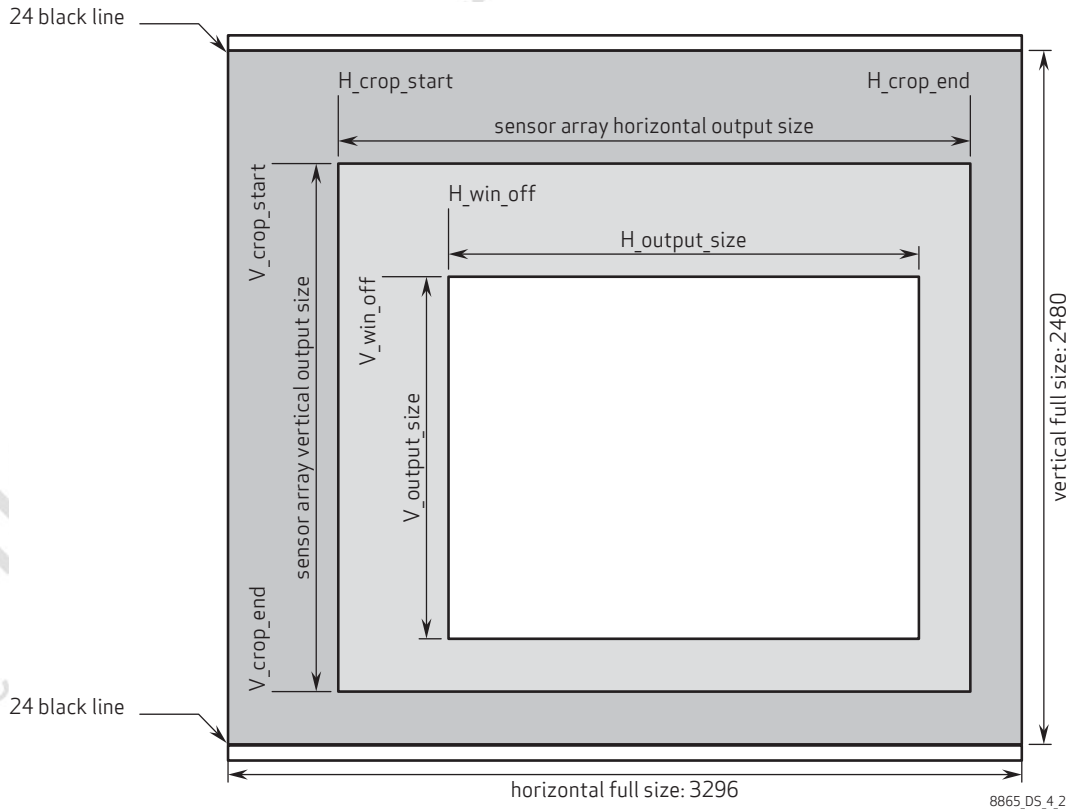


table 4-2 image windowing control functions (sheet 1 of 2)

address	register name	default value	R/W	description
0x3800	H_CROP_START	0x00	RW	Bit[3:0]: Manual horizontal crop start address[11:8]
0x3801	H_CROP_START	0x0C	RW	Bit[7:0]: Manual horizontal crop start address[7:0]
0x3802	V_CROP_START	0x00	RW	Bit[3:0]: Manual vertical crop start address[11:8]
0x3803	V_CROP_START	0x0C	RW	Bit[7:0]: Manual vertical crop start address[7:0]
0x3804	H_CROP_END	0x0C	RW	Bit[3:0]: Manual horizontal crop end address[11:8]
0x3805	H_CROP_END	0xD3	RW	Bit[7:0]: Manual horizontal crop end address[7:0]
0x3806	V_CROP_END	0x09	RW	Bit[3:0]: Manual vertical crop end address[11:8]
0x3807	V_CROP_END	0xA3	RW	Bit[7:0]: Manual vertical crop end address[7:0]
0x3808	H_OURPUT_SIZE	0xDC	RW	Bit[3:0]: Horizontal output size[11:8]
0x3809	H_OUTPUT_SIZE	0xC0	RW	Bit[7:0]: Horizontal output size[7:0]
0x380A	V_OUTPUT_SIZE	0x09	RW	Bit[3:0]: Vertical output size[11:8]
0x380B	V_OUTPUT_SIZE	0x90	RW	Bit[7:0]: Vertical output size[7:0]
0x380C	TIMING_HTS	0x07	RW	Bit[7:0]: Horizontal total size[15:8]
0x380D	TIMING_HTS	0x4C	RW	Bit[7:0]: Horizontal total size[7:0]
0x380E	TIMING_VTS	0x0A	RW	Bit[6:0]: Vertical total size[14:8]
0x380F	TIMING_VTS	0x74	RW	Bit[7:0]: Vertical total size[7:0]
0x3810	H_WIN_OFF	0x00	RW	Bit[7:0]: Manual horizontal windowing offset[15:8]
0x3811	H_WIN_OFF	0x04	RW	Bit[7:0]: Manual horizontal windowing offset[7:0]
0x3812	V_WIN_OFF	0x00	RW	Bit[3:0]: Manual vertical windowing offset[11:8]
0x3813	V_WIN_OFF	0x02	RW	Bit[7:0]: Manual vertical windowing offset[7:0]
0x3814	H_INC_ODD	0x01	RW	Bit[4:0]: Horizontal sub-sample odd increase number
0x3815	H_INC_EVEN	0x01	RW	Bit[4:0]: Horizontal sub-sample even increase number
0x382A	V_INC_ODD	0x01	RW	Bit[4:0]: Vertical sub-sample odd increase number
0x382B	V_INC_EVEN	0x01	RW	Bit[4:0]: Vertical sub-sample even increase number

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

address	register name	default value	R/W	description
0x3841	AUTO_SIZE_CTRL	0xFF	RW	Bit[5]: V window auto enable Bit[4]: H window auto enable Bit[3]: V end size auto enable Bit[2]: H end size auto enable Bit[1]: V start size auto enable Bit[0]: H start size auto enable
0x3842	H_AUTO_OFF_H	0x00	RW	Bit[3:0]: H_offset[11:8] for auto size mode The offset is complemental code 0x0001 is to right shift 1 pixel 0xFFFF is to left shift 1 pixel
0x3843	H_AUTO_OFF_L	0x00	RW	Bit[7:0]: H_offset[7:0] for auto size mode
0x3844	V_AUTO_OFF_H	0x00	RW	Bit[3:0]: V_offset[11:8] for auto size mode The offset is complemental code 0x0001 is to up shift 1 row 0xFFFF is to down shift 1 row
0x3845	V_AUTO_OFF_L	0x00	RW	Bit[7:0]: V_offset[7:0] for auto size mode

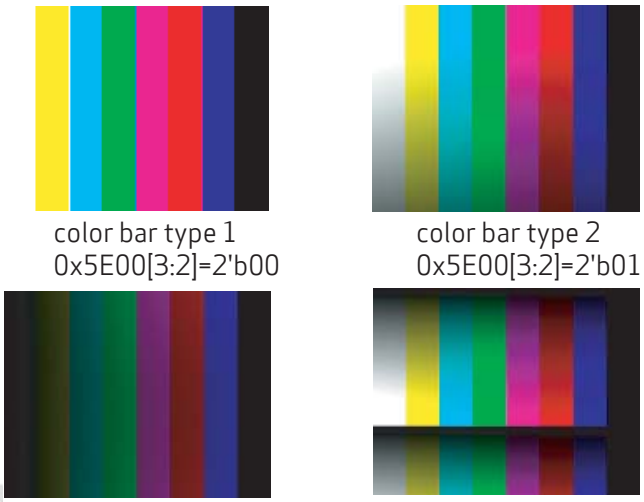
4.3 test pattern

For testing purposes, the OV8865 offers three types of test patterns: color bar, square and random data. The OV8865 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 (0x5E00[3:2]). The digital test pattern function is controlled by register 0x5E00[7].

4.3.1 color bar

There are four types of color bars which are switched by bar-style in register 0x5E00[3:2] (see **figure 4-3**).

figure 4-3 color bar types

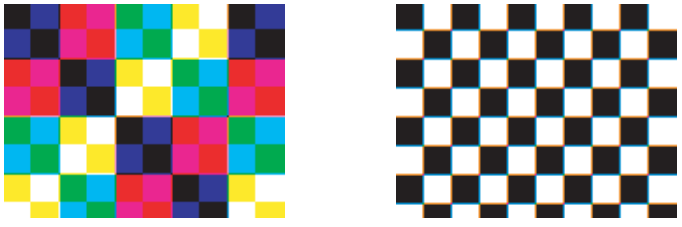


8865_DS_4_3

4.3.2 square

There are two types of squares: color square and black-white square. The squ_bw register (0x5E00[4]) decides which type of square will be output.

figure 4-4 color, black and white square bars



8865 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 (0x5E00[5]). If this register is set, the transparent test pattern will be displayed. The following image is an example showing a transparent color bar image (see **figure 4-5**).

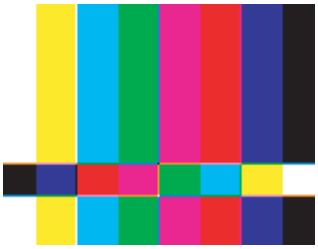
figure 4-5 transparent effect

8865_DS_4_5

4.3.5 rolling bar effect

The rolling bar is set by rolling_bar_en register (0x5E00[6]). If it is set, a inverted-color rolling bar will roll from up to down. The following image is an example showing a rolling bar on color bar image (see **figure 4-6**).

figure 4-6 rolling bar effect



8865_DS_4_6

table 4-3 test pattern registers

address	register name	default value	R/W	description
0x5E00	PRE CTRL00	0x00	RW	Bit[7]: Test pattern enable Bit[6]: Rolling bar function enable Bit[5]: Transparent enable 0: Disable transparent effect function 1: Enable transparent effect function Bit[4]: Square mode 0: Color square 1: Black-white square Bit[3:2]: Color bar style 00: Standard color bar 01: Top-bottom darker color bar 10: Right-left darker color bar 11: Bottom-top darker color bar Bit[1:0]: Test pattern mode 00: Color bar 01: Random data 10: Square pattern 11: Black image
0x5E01	PRE CTRL01	0x41	RW	Bit[6]: Window cut enable 0: Do not cut the redundant pixels 1: Cut the redundant pixels Bit[5]: two_lsb_0_en When set, two LSBs of output data are 0 Bit[4]: Same seed enable When set, the seed used to generate the random data are same which is set in seed register Bit[3:0]: Random seed Seed used in generating random data

4.4 average luminance (YAVG)

Exposure time control is based on a frame brightness average value. The OV8865 supports the average image luminance calculation. By properly setting X_start, Y_start, and window_width and window_height as shown in transparent effect. **table 4-4** lists the corresponding registers.

figure 4-7 average-based window definition

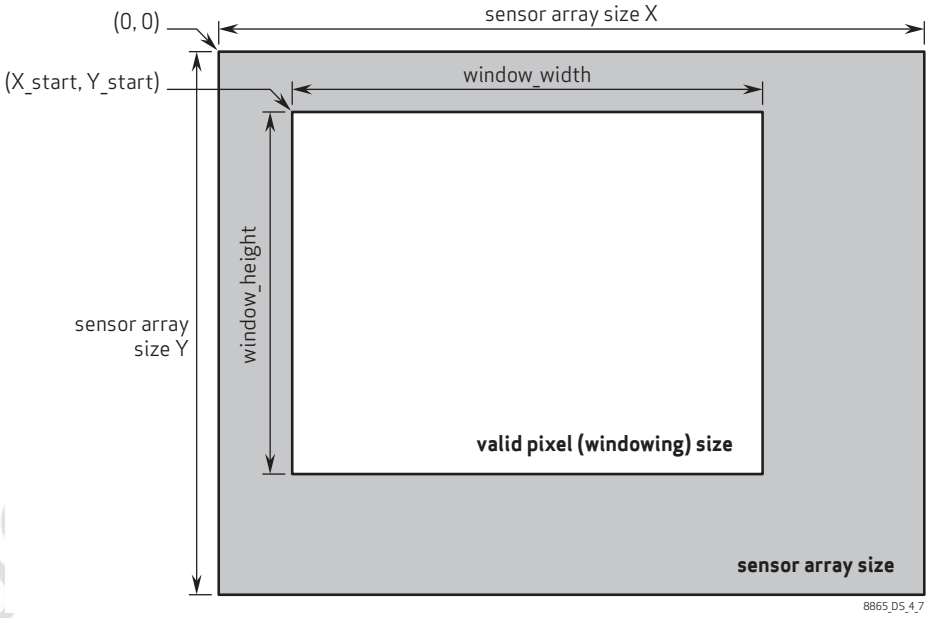


table 4-4 AVG registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5041	ISP CTRL41	0x14	RW	Bit[2]: AVG function enable 0: Disable 1: Enable
0x5680	AVG CTRL00	0x00	RW	Bit[4:0]: X_start_avg[12:8]
0x5681	AVG CTRL01	0x00	RW	Bit[7:0]: X_start_avg[7:0]
0x5682	AVG CTRL02	0x00	RW	Bit[3:0]: Y_start_avg[11:8]
0x5683	AVG CTRL03	0x00	RW	Bit[7:0]: Y_start_avg[7:0]
0x5684	AVG CTRL04	0x0C	RW	Bit[4:0]: Window_width_avg[12:8]
0x5685	AVG CTRL05	0xC0	RW	Bit[7:0]: Window_width_avg[7:0]

table 4-4 AVG registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5686	AVG CTRL06	0x09	RW	Bit[3:0]: Window_height_avg[11:8]
0x5687	AVG CTRL07	0x90	RW	Bit[7:0]: Window_height_avg[7:0]
0x5688	AVG CTRL08	0x02	RW	Bit[1]: Sum option 0: Sum=(4×B+9×G×2 +10×R)/8 1: Sum=B+G×2+R Bit[0]: Sub-window function enable 0: Use whole output window for average 1: Use registers 0x5680~0x5687 to define window for average
0x568A	AVG RO0A	–	R	Bit[7:0]: High 8 bits of whole image average output

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.

Black level adjustments can be made with registers 0x4000, 0x4004, and 0x4005.

table 4-5 BLC registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4000	BLC CTRL00	0xF1	RW	Bit[7]: Offset out of range triggers BLC enable Bit[6]: Format change triggers BLC enable Bit[5]: Gain change triggers BLC enable Bit[4]: Exposure change triggers BLC enable Bit[3]: Manually trigger BLC signal Its rising edge will trigger BLC Bit[2]: BLC freeze function enable When set, BLC will be frozen and the offsets will keep the pre-frame values Bit[1]: BLC always triggered enable When set, the BLC will be triggered every frame unless register bit 0x4000[2] is enabled Bit[0]: Five points median filter function enable

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

address	register name	default value	R/W	description
0x4001	BLC CTRL01	0x86	RW	Bit[7]: Offset dithering function enable Bit[6]: Enable difference between black line column with zero line to cancel horizontal noise Bit[5:4]: Column shift option 00: Left 256 columns 01: Left 128 columns 10: Left 64 columns 11: Left 32 columns Bit[2]: Final BLC offset limitation enable Bit[1]: BLC column cancel function enable
0x4002	BLC CTRL02	0x40	RW	Bit[7:0]: Up threshold of cut range function
0x4004	BLC CTRL04	0x00	RW	Bit[7:0]: Target[15:8] High byte of BLC target
0x4005	BLC CTRL05	0x40	RW	Bit[7:0]: Target[7:0] Low byte of BLC target
0x4009	BLC CTRL09	0x29	RW	Bit[7:4]: Line number for BLC initial function Bit[3]: Bypass cut range function enable Bit[2:0]: BLC column added offset[10:8]
0x400A	BLC CTRL0A	0x00	RW	Bit[7:0]: BLC column added offset[7:0]
0x400B	BLC CTRL0B	0x0C	RW	Bit[7:4]: Start line for BLC initial function Bit[3]: Offset limitation function enable Bit[2]: Cut range function enable Bit[1:0]: BLC last line select
0x4011	BLC CTRL11	0x00	RW	Bit[7:6]: Dithering offset Bit[5]: offset_man_same When it is enabled, the manual offsets will be same. They are all defined by manual_offset00. Bit[4]: Offset manual mode enable
0x401E	BLC CTRL1E	0x20	RW	Bit[7:0]: Down threshold of cut range function
0x401F	BLC CTRL1F	0x06	RW	Bit[3]: Rblue BLC reverse Bit[2]: Interpolation x enable Bit[1]: Interpolation y enable Bit[0]: Anchor one enable
0x4030	DCBLC K1	0x01	RW	Bit[3:0]: Dark current BLC top K coefficient[11:8]
0x4031	DCBLC K1	0x00	RW	Bit[7:0]: Dark current BLC top K coefficient[7:0]
0x4032	DCBLC K2	0x01	RW	Bit[3:0]: Dark current BLC bottom K coefficient[11:8]
0x4033	DCBLC K2	0x00	RW	Bit[7:0]: Dark current BLC bottom K coefficient[7:0]

4.6 one time programmable (OTP) memory

The OV8865 supports a maximum of 1536 bytes of one-time programmable (OTP) memory to store chip identification and manufacturing information, which can be used to update the sensor's default setting and can be controlled through the SCCB (see **table 4-6**). OTP address ranges from 0x7000 to 0x75FF, where 0x7000~0x700F and 0x72D0~0x75FF are reserved for OmniVision use and 0x7010~0x72CF are for customer use.

4.6.1 OTP other functions

OTP loading data can be triggered when power up or writing 0x01 to register 0x3D81. Power up loading data is enabled by register 0x3D85[2], by default it is off. Auto mode and manual mode can be chosen by setting register 0x3D84[6] to 0 and 1, respectively, and by default, it is in auto mode. In auto mode, all data in the OTP will be loaded to the OTP buffer; while in manual mode, part of the data which is defined by the start address ({0x3D88,0x3D89}) and the end address ({0x3D8A,0x3D8B}) of the OTP will be loaded to the OTP buffer.

The OV8865 supports loading setting. When 0xDD as a head byte is read out from the start address, which is set by {0x3D8C, 0x3D8D}, setting is recognized. While the setting is being read out from the OTP, it is being written to the OTP buffer, and at the same time, interpreting to the register write command. Loading setting is controlled by registers 0x3D85[1] and 0x3D85[0], which enable power up loading setting and writing register loading setting, respectively.

There are two types of setting format:

1. AX Start Address MSB, Start Address LSB, data0, data1,.... dataX
2. 5X (X can be 0x0 ~ 0xF) data0, data1,.... dataX

Neither AX nor 5X means the end of the setting. 5X means the start address is from the previous end address. X means number of registers is (x+1).

Example: store the setting table in address 0x0100 of OTP. The table content is: DD A3 30 00 11 22 33 44 53 55 66 77 88

which is: 3000-11, 3001-22, 3002-33, 3003-44, 3004-55,3005-66,3006-77,3007-88

To program the OTP:

```
6C 3D84 40; [6]manual mode enable
6C 3D85 00
6C 3D88 71; manual OTP start address for access
6C 3D89 00
6C 3D8A 71; manual OTP end address for access
6C 3D8B 0C
6C 0100 01; stream mode enable
;delay 20ms
6C 7100 DD
6C 7101 A3
6C 7102 30
6C 7103 00
6C 7104 11
6C 7105 22
6C 7106 33
```

```
6C 7107 44
6C 7108 53
6C 7109 55
6C 710A 66
6C 710B 77
6C 710C 88
6C 3D80 01;[0] program enable
;delay 200ms
6C 3D80 00
```

Setting for loading:

```
...
6C 3D88 71; manual OTP start address for access
6C 3D89 00
6C 3D8A 71; manual OTP end address for access
6C 3D8B 0C
6C 3D85 06; [2] OTP load data enable
; [1] OTP load setting enable
6C 3D8C 01; Start address OTP setting table, the first byte of OTP setting table should
be 0xDD
6C 3D8D 00;
6C 0100 01; stream mode enable, after streaming of the first power up, OV8865 will load
setting from OTP if 3D85[2:1]=2'b11
```

The OV8865 supports OTP BIST. When register 0x3D85[4] is set to 1, the BIST function is enabled. When OTP loading data, the data which is read out from the OTP can be compared with zero or the data with the same address in the register, which can be controlled by setting register 0x3D85[5] to 1 or 0, respectively. After the BIST done, the BIST done flag can be read out from register 0x3D81[4], the BIST error flag can be read out from 0x3D81[5], and the address of the first error can be read out from {0x3D8E, 0x3D8F}.

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

address	register name	default value	R/W	description
0x7000~ 0x75FF	OTP_SRAM	0x00	RW	Bit[7:0]: OTP buffer
0x3D80	OTP_PROGRAM_CTRL	–	RW	Bit[7]: OTP_wr_busy (read only) Bit[0]: OTP_program_enable (write only)

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

address	register name	default value	R/W	description
0x3D81	OTP_LOAD_CTRL	–	RW	Bit[7]: OTP_rd_busy (read only) Bit[5]: OTP_bist_error (read only) Bit[4]: OTP_bist_done (read only) Bit[0]: OTP_load_enable (read and write)
0x3D84	OTP_MODE_CTRL	0x00	RW	Bit[7]: Program disable 1: Disable Bit[6]: Mode select 0: Auto mode 1: Manual mode
0x3D85	OTP_REG85	0x13	RW	Bit[5]: OTP_bist_select 0: Compare with SRAM 1: Compare with zero Bit[4]: OTP_bist_enable Bit[2]: OTP power up load data enable Bit[1]: OTP power up load setting enable Bit[0]: OTP write register load setting enable
0x3D88	OTP_START_ADDRESS	0x00	RW	OTP Start High Address for Manual Mode
0x3D89	OTP_START_ADDRESS	0x00	RW	OTP Start Low Address for Manual Mode
0x3D8A	OTP_END_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_BIST_ERR_ADDRESS	–	R	OTP Check Error Address High
0x3D8F	OTP_BIST_ERR_ADDRESS	–	R	OTP Check Error Address Low

4.7 temperature sensor

TheOV8865 supports an on-chip temperature sensor that covers -40~192°C with an error range of 5°C. It can be controlled through the SCCB interface (see **table 4-7**). When the readout data is lower than 0xC0, the temperature is a positive value.

If the readout data is higher than 0xC0, the temperature is lower than 0°C and the readout data is twos complement code. Before reading the temperature, the temperature sensor should be triggered by a 0 to 1 transition of register 0x4D12[0].

table 4-7 temperature sensor functions

address	register name	R/W	description
0x4D12	TPM TRIGGER	RW	Bit[0]: Temperature sensor trigger
0x4D13	TPM READ	R	Bit[7:0]: Temperature readout

4.8 strobe flash and frame exposure

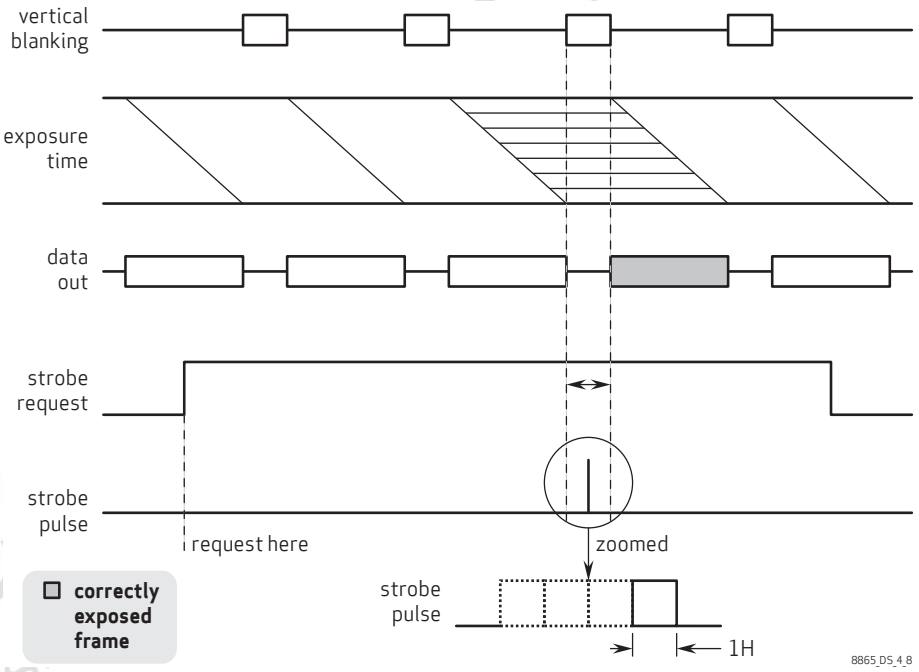
4.8.1 strobe flash control

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.

4.8.1.1 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.

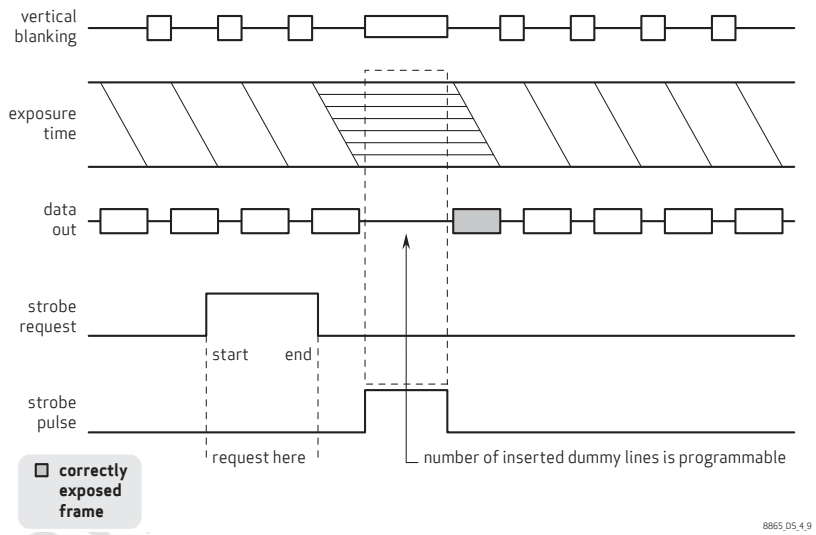
figure 4-8 xenon flash mode



4.8.1.2 LED 1

In LED 1 mode, the strobe signal stays active until the strobe end request is sent (see LED 1 mode).

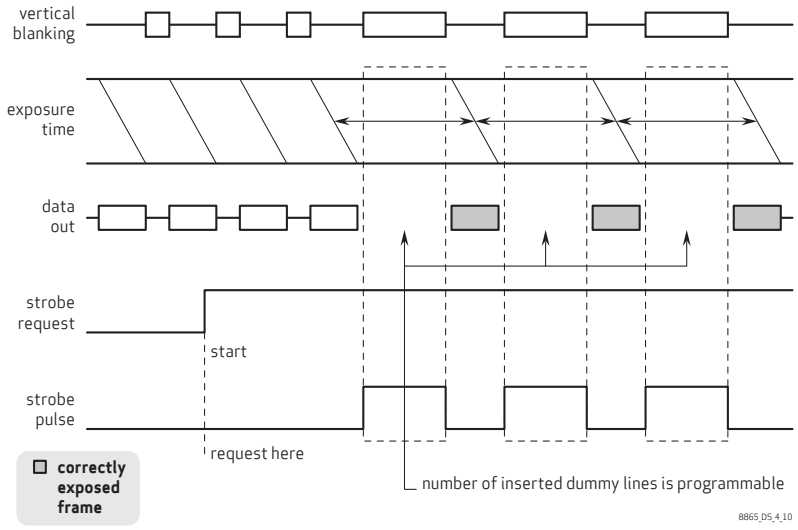
figure 4-9 LED 1



4.8.1.3 LED 2 mode

In LED 2 mode, the strobe signal width can be added by inserting dummy lines which is controlled by register {0x3B02, 0x3B03} (see **section 4-10**).

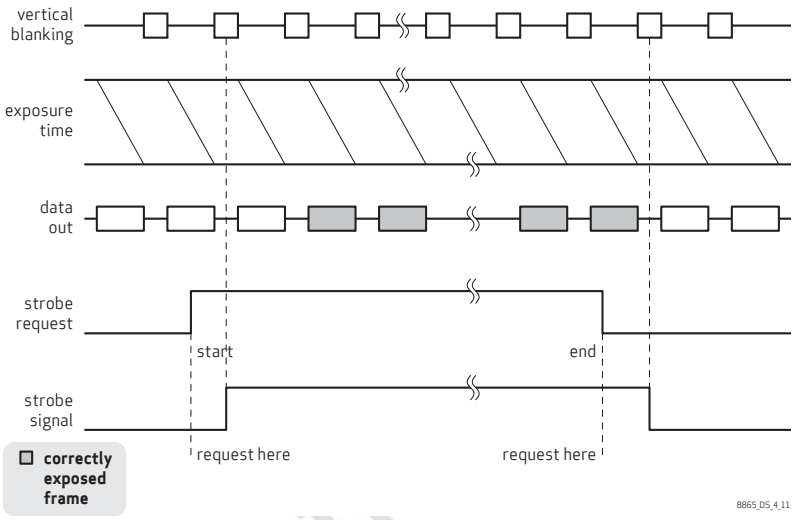
figure 4-10 LED2 mode



4.8.1.4 LED 3 mode

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

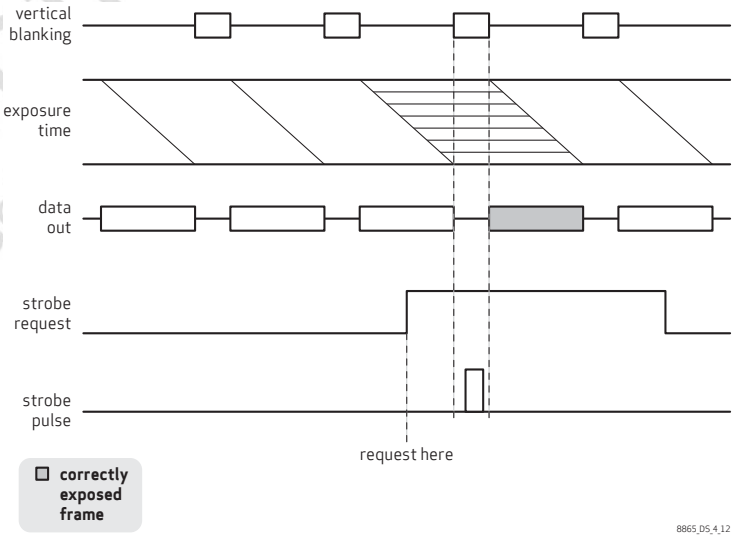
figure 4-11 LED 3 mode



4.8.1.5 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



4.9 embedded data

The MIPI Camera Serial Interface 2 (CSI-2) specifications provide an option to embed sensor internal status information in the picture frame to be delivered to the MIPI host. This feature is especially useful for MIPI host image process. Parameters in sensor, such as ADC gain and exposure time, can help MIPI host to fine tune image processor settings for better image presentation.

The embedded data can be at the beginning or the end of each picture frame by setting register 0x5A08[2]. If embedded information exists, then the lines containing the embedded data must use the embedded data packet data type in the data identifier. It can be configured through register 0x4816 and its default value is 0x13.

table 4-8 embedded data control registers

address	register name	default value	R/W	description
0x4816	EMBEDED DT	0x53	RW	Bit[5:0]: Embedded line data type
0x5A08	EMBEDED FLAG	0x06	RW	Bit[2]: Embedded line flag 0: At start of frame (frame header) 1: At end of frame (frame footer)

table 4-9 describes all data in sensor embedded line for MIPI host image processing. The number in the first column indicates the position of the data with unit in byte, while the second column is the register in that position.

table 4-9 embedded line position data (sheet 1 of 4)

byte number	register name
0	sensor info
1	not used
2	digital gain
3	not used
4	analog gain[10:8]
5	analog gain[7:0]
6	course integration time[15:8]
7	course integration time[7:0]
8	not used
9	not used
10	DPC threshold[9:2]
11	x_addr_start[15:8]

table 4-9 embedded line position data (sheet 2 of 4)

byte number	register name
12	x_addr_start[7:0]
13	y_addr_start[15:8]
14	y_addr_start[7:0]
15	x_output_size[15:8]
16	x_output_size[7:0]
17	y_output_size[15:8]
18	y_output_size[7:0]
19	frame_length_lines[15:8]
20	frame_length_lines[7:0]
21	line_length_pck[15:8]
22	line_length_pck[7:0]
23	MIPI header revision number
24	die temperature
25	x_addr_end[15:8]
26	x_addr_end[7:0]
27	y_addr_end[15:8]
28	y_addr_end[7:0]
29	x_inc
30	y_inc
31	image orientation
32	frame duration A
33	frame duration B
34	context count
35	context select
36	mipi_lane_number
37	clk_hs_prepare
38	ui_clk_hs_prepare
39	clk_hs_zero[15:8]
40	clk_hs_zero[7:0]
41	dat_hs_prepare

table 4-9 embedded line position data (sheet 3 of 4)

byte number	register name
42	ui_dat_hs_prepare
43	dat_hs_zero[15:8]
44	dat_hs_zero[7:0]
45	pclk_period
46	PLL1_multiplier[9:8], PLL1_predivp, PLL1_prediv[4:0]
47	PLL1_multiplier[7:0]
48	PLL1_op_sys_div
49	PLL1_divs[1:0], PLL1_divmipi[1:0], PLL1_op_pix_div[3:0]
50	PLL2_divp[9:8], PLL2_predivp, PLL2_prediv[4:0]
51	PLL2_divp[7:0]
52	PLL2_divdac[3:0], PLL2_divs[3:0]
53	PLL2_divsp[7:0]
54	data pedestal[9:2]
55	blue optical black data[15:8]
56	blue optical black data[7:0]
57	Gb optical black data[15:8]
58	Gb optical black data[7:0]
59	Gr optical black data[15:8]
60	Gr optical black data[7:0]
61	red optical black data[15:8]
62	red optical black data[7:0]
63	frame average[9:2]
64	digital_gain_red
65	digital_gain_red
66	digital_gain_greenR
67	digital_gain_greenR
68	digital_gain_blue
69	digital_gain_blue
70	digital_gain_greenB
71	digital_gain_greenB

table 4-9 embedded line position data (sheet 4 of 4)

byte number	register name
72	horizontal scale ratio[15:8]
73	horizontal scale ratio[7:0]
74	vertical scale ratio[15:8]
75	vertical scale ratio[7:0]
76	DPC on/off
77	strobe control 0
78	strobe control 1
79	strobe control 2
80	test pattern control
81	solid color B hi
82	solid color B lo
83	solid color Gb hi
84	solid color Gb lo
85	solid color R hi
86	solid color R lo
87	solid color Gr hi
88	solid color Gr lo
89	frame counter
90	R channel average[9:2]
91	G channel average[9:2]
92	B channel average[9:2]
93	Y average[9:2]
94	Y, R, G, B average LSBs
95	die temperature
96	temperature decimal

5 image sensor processor digital functions

5.1 DSP top

The main purpose of the DSP top includes:

- integrate all sub-modules
- create necessary control signals

table 5-1 DSP top registers

address	register name	default value	R/W	description
0x5000	DSP CTRL00	0x96	RW	DSP Control 00 (0: disable, 1: enable) Bit[7]: Lens correction (LENC) function enable Bit[4]: Manual WB Gain function enable Bit[2]: Black DPC function enable Bit[1]: White DPC function enable
0x5001	DSP CTRL01	0x01	RW	Bit[0]: BLC function enable
0x5002	DSP CTRL02	0x08	RW	Bit[2]: VarioPixel® function enable
0x5003	DSP CTRL03	0x20	RW	Bit[6]: DSP bypass mode Bit[5]: DPC and DBC buffer control enable
0x501F	DSP CTRL1F	0x00	RW	Bit[5]: Bypass DSP enable Bit[4]: Bit shift enable, when DSP is not bypassed Bit[3]: Bit shift direction 0: Left shift 1: Right shift Bit[2:0]: Bit shift number
0x5025	DSP CTRL25	0x00	RW	Bit[1:0]: Average data select 00: After pre_DSP module 01: After DPC module 10: After MWB module 11: After DBC module
0x5041	DSP CTRL41	0x14	RW	Bit[4]: Digital binning compensation (DBC) function enable Bit[2]: Average function enable
0x5043	DSP CTRL43	0x08	RW	Bit[4:3]: Substrate offset for average module Bit[2]: Horizontal DBC enable Bit[1]: Vertical DBC enable Bit[0]: Manual mode of DBC function enable (In auto mode, DBC will disabled automatically if buffer size is not enough)

5.2 pre_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 image by removing redundant data
- create control signals

table 5-2 pre_DSP registers

address	register name	default value	R/W	description
0x5E00	PRE CTRL00	0x00	RW	Bit[7]: Test pattern enable Bit[6]: Rolling bar function enable Bit[5]: Transparent enable 0: Disable transparent effect function 1: Enable transparent effect function Bit[4]: Square mode 0: Color square 1: Black-white square Bit[3:2]: Color bar style 00: Standard color bar 01: Top-bottom darker color bar 10: Right-left darker color bar 11: Bottom-top darker color bar Bit[1:0]: Test pattern mode 00: Color bar 01: Random data 10: Square pattern 11: Black image
0x5E01	PRE CTRL01	0x41	RW	Bit[6]: Window cut enable 0: Do not cut the redundant pixels 1: Cut the redundant pixels Bit[5]: two_lsb_0_en When set, two LSBs of output data are 0 Bit[4]: Same seed enable When set, the seed used to generate the random data are same which is set in seed register Bit[3:0]: Random seed Seed used in generating random data

5.3 defective pixel cancellation (DPC)

The DPC uses a one line buffer and removes defect pixels. It also supports black/white mode.

table 5-3 DPC control registers

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0x96	RW	Bit[2]: Black DPC function enable Bit[1]: White DPC function 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
0x5A00	WINC CTRL00	0x00	RW	Bit[3:0]: X_start offset[11:8]
0x5A01	WINC CTRL01	0x00	RW	Bit[7:0]: X_start offset[7:0]
0x5A02	WINC CTRL02	0x00	RW	Bit[3:0]: Y_start offset[11:8]
0x5A03	WINC CTRL03	0x00	RW	Bit[7:0]: Y_start offset[7:0]
0x5A04	WINC CTRL04	0x0C	RW	Bit[3:0]: Window width[11:8]
0x5A05	WINC CTRL05	0xE0	RW	Bit[7:0]: Window width[7:0]
0x5A06	WINC CTRL06	0x09	RW	Bit[3:0]: Window height[11:8]
0x5A07	WINC CTRL07	0xB0	RW	Bit[7:0]: Window height[7:0]
0x5A08	WINC CTRL08	0x06	RW	Bit[2]: Select embed line flag 0: Select first line as embed flag 1: Select last line as embed flag Bit[1]: Window enable option 0: Disable window after last valid line 1: Get enable from register Bit[0]: Manual window enable

5.5 manual white balance (MWB)

The manual white balance (MWB) provides digital gain for R,G and B channels. Each channel gain is 14-bit. 0x400 is 1x gain. This function is shared with digital gain.

table 5-5 MWB registers

address	register name	default value	R/W	description
0x5018	ISP CTRL18	0x10	RW	Bit[7:0]: Red MWB gain[13:6]
0x5019	ISP CTRL19	0x00	RW	Bit[5:0]: Red MWB gain[5:0]
0x501A	ISP CTRL1A	0x10	RW	Bit[7:0]: Green MWB gain[13:6]
0x501B	ISP CTRL1B	0x00	RW	Bit[5:0]: Green MWB gain[5:0]
0x501C	ISP CTRL1C	0x10	RW	Bit[7:0]: Blue MWB gain[13:6]
0x501D	ISP CTRL1D	0x00	RW	Bit[5:0]: Blue MWB gain[5:0]
0x501E	ISP CTRL1E	0x00	RW	Bit[1]: Digital gain function enable (MWB gain will be disabled) Bit[0]: Same MWB gain enable (copy red gain to green and blue channel)

5.6 AVG

The main function of the AVG module is to calculate the luminance average using special filters.

table 5-6 AVG control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5041	ISP CTRL41	0x14	RW	Bit[2]: AVG function enable 0: Disable 1: Enable
0x5680	AVG CTRL00	0x00	RW	Bit[4:0]: X_start_avg[12:8]
0x5681	AVG CTRL01	0x00	RW	Bit[7:0]: X_start_avg[7:0]
0x5682	AVG CTRL02	0x00	RW	Bit[3:0]: Y_start_avg[11:8]
0x5683	AVG CTRL03	0x00	RW	Bit[7:0]: Y_start_avg[7:0]
0x5684	AVG CTRL04	0x0C	RW	Bit[4:0]: Window_width_avg[12:8]
0x5685	AVG CTRL05	0xC0	RW	Bit[7:0]: Window_width_avg[7:0]

table 5-6 AVG control registers (sheet 2 of 2)

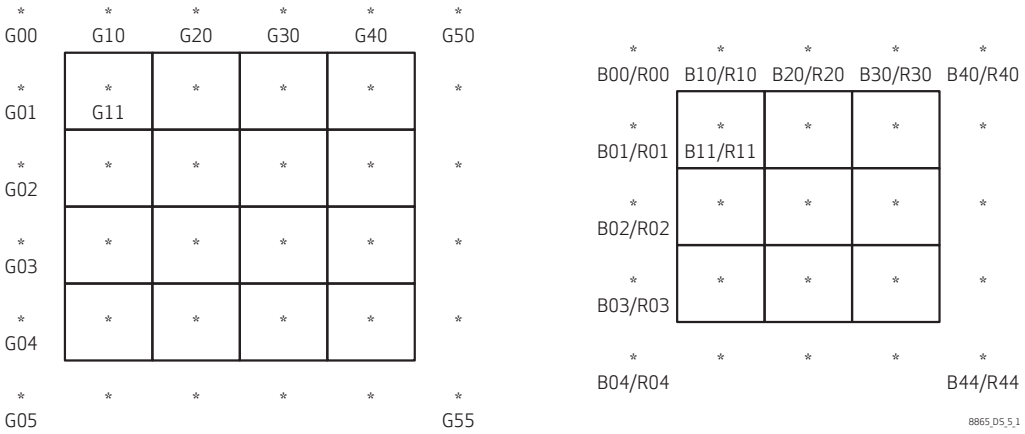
address	register name	default value	R/W	description
0x5686	AVG CTRL06	0x09	RW	Bit[3:0]: Window_height_avg[11:8]
0x5687	AVG CTRL07	0x90	RW	Bit[7:0]: Window_height_avg[7:0]
0x5688	AVG CTRL08	0x02	RW	Bit[1]: Sum option 0: Sum=(4×B+9×G×2 +10×R)/8 1: Sum=B+G×2+R Bit[0]: Sub-window function enable 0: Use whole output window for average 1: Use registers 0x5680~0x5687 to define window for average
0x568A	AVG RO0A	–	R	Bit[7:0]: High 8 bits of whole image average output

5.7 lens correction (LENC)

The LENC algorithm compensates for the illumination drop off in the corners due to the lens. Based on the radius of each pixel to the lens, the algorithm calculates a gain for each pixel and then corrects each pixel with the calculated gain to compensate for the light distribution due to the lens curvature. Additionally, the LENC supports subsampling in both the horizontal and vertical directions. LENC is performed in the RGB domain.

Luminance channel consists of 36 control points while each color channel consists of 25 control points.

figure 5-1 control points of luminance and color channels



note There is a lens calibration tool that can be used for calibrating these settings required for a specific module. Contact your local OmniVision FAE for generating these settings.

figure 5-2 luminance compensation level calculation

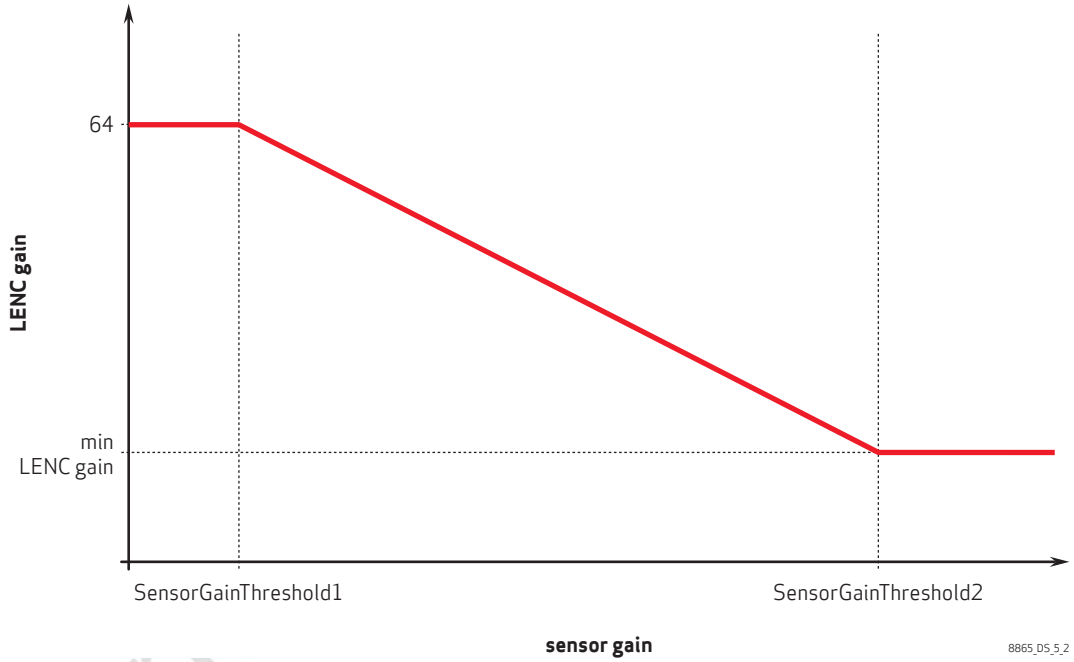


table 5-7 LENC control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0x96	RW	Bit[7]: Lens correction (LENC) function enable 0: Disable 1: Enable
0x5800	LENC G00	0x10	RW	Bit[5:0]: Control point G00 for luminance compensation
0x5801	LENC G01	0x10	RW	Bit[5:0]: Control point G01 for luminance compensation
0x5802	LENC G02	0x10	RW	Bit[5:0]: Control point G02 for luminance compensation
0x5803	LENC G03	0x10	RW	Bit[5:0]: Control point G03 for luminance compensation
0x5804	LENC G04	0x10	RW	Bit[5:0]: Control point G04 for luminance compensation
0x5805	LENC G05	0x10	RW	Bit[5:0]: Control point G05 for luminance compensation

table 5-7 LENC control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x5806	LENC G10	0x10	RW	Bit[5:0]: Control point G10 for luminance compensation
0x5807	LENC G11	0x08	RW	Bit[5:0]: Control point G11 for luminance compensation
0x5808	LENC G12	0x08	RW	Bit[5:0]: Control point G12 for luminance compensation
0x5809~ 0x5822	LENC G13~LENC G54	–	RW	Bit[5:0]: Control point G13~G54 for luminance compensation
0x5823	LENC G55	0x10	RW	Bit[5:0]: Control point G55 for luminance compensation
0x5824	LENC BR00	0xAA	RW	Bit[7:4]: Control point B00 for blue channel compensation Bit[3:0]: Control point R00 for red channel compensation
0x5825	LENC BR01	0xAA	RW	Bit[7:4]: Control point B01 for blue channel compensation Bit[3:0]: Control point R01 for red channel compensation
0x5826	LENC BR02	0xAA	RW	Bit[7:4]: Control point B02 for blue channel compensation Bit[3:0]: Control point R02 for red channel compensation
0x5827	LENC BR03	0xAA	RW	Bit[7:4]: Control point B03 for blue channel compensation Bit[3:0]: Control point R03 for red channel compensation
0x5828	LENC BR04	0xAA	RW	Bit[7:4]: Control point B04 for blue channel compensation Bit[3:0]: Control point R04 for red channel compensation
0x5829~ 0x583C	LENC BR10~ LENC BR44	–	RW	Bit[7:4]: Control point B10~B44 for blue channel compensation Bit[3:0]: Control point R10~R44 for red channel compensation
0x583D	LENC BROFFSET	0x88	RW	Bit[7:4]: Base value for all blue channel control points Bit[3:0]: Base value for all red channel control points

table 5-7 LENC control registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x583E	LENC MAXGAIN	0x40	RW	Bit[7:0]: If AutoLensSwitchEnable is true and sensor gain is larger than this threshold, luminance compensation amplitude will be the minimum value (min LENC gain). Register value is 16 times sensor gain
0x583F	LENC MINGAIN	0x20	RW	Bit[7:0]: If AutoLensSwitchEnable is true and sensor gain is larger than this threshold, luminance compensation amplitude will start to decrease; otherwise, the amplitude will not change. Register value is 16 times sensor gain.
0x5840	LENC MINQ	0x18	RW	Bit[6:0]: This value indicates the minimum amplitude which luminance channel compensates when AutoLensSwitchEnable is true. Value should be in the range [0~64]
0x5841	LENC CTRL	0x0D	RW	Bit[3]: Add BLC target after applying compensation Bit[2]: Enable BLC target for LENC 0: Disable BLC target 1: Enable BLC target Bit[0]: AutoLensSwitchEnable 0: Luminance compensation amplitude does not change with sensor gain 1: Luminance compensation amplitude changes with sensor gain

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

Manual exposure provides exposure time settings and sensor gain. The exposure value in register 0x3500~0x3502 is in units of 1/16 line.

Manual gain provides analog gain settings. The OV8865 has a maximum 16x analog gain.

note
For optimal performance, maximum exposure should be 200ms. For more details, contact your local OmniVision FAE.

table 5-8 MEC/MGC control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3500	AEC EXPO	0x00	RW	Bit[3:0]: Exposure[19:16]
0x3501	AEC EXPO	0x02	RW	Bit[7:0]: Exposure[15:8]
0x3502	AEC EXPO	0x00	RW	Bit[7:0]: Exposure[7:0] Low 4 bits are fraction bits Minimum exposure value is 2 lines
0x3503	AEC MANUAL	0x00	RW	Bit[6]: Digital fraction gain delay option 0: Delay 1 frame 1: Not delay 1 frame Bit[5]: Gain change delay option 0: Delay 1 frame 1: Not delay 1 frame Bit[4]: Gain delay option 0: Delay 1 frame 1: Not delay 1 frame 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[1:0]: Sensor gain option for transferring real gain to sensor gain format
0x3507	AEC GAIN SHIFT	0x00	RW	Bit[1:0]: Gain shift option 00: Not shift 01: Left shift 1 bit 10: Left shift 2 bits 11: Left shift 3 bits
0x3508	AEC GAIN	0x00	RW	Bit[4:0]: Gain[12:8]

table 5-8 MEC/MGC control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3509	AEC GAIN	0x80	RW	Bit[7:0]: Gain[7:0] 0x3503[2]=0, 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, 0x100 is 2x gain if 0x3503[2]=1, gain[12:0] is sensor gain format, gain[12:8] is coarse gain, 00000: 1x, 00001: 2x, 00011: 4x, 00111: 8x, gain[7] is 1, gain[6:3] is fine gain, gain[2:0] is always 0. For example, 0x080 is 1x gain, 0x180 is 2x gain, 0x380 is 4x gain
0x350A	AEC DIGIGAIN	0x08	RW	Bit[7:0]: Digital gain[13:6]
0x350B	AEC DIGIGAIN	0x00	RW	Bit[5:0]: Digital gain[5:0] Low 10 bits are fraction bits

6 register tables

The following tables provide descriptions of the device control registers contained in the OV8865. 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, 0x0300 - 0x031E]

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

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
0x0300	PLL_CTRL_0	0x00	RW	Bit[7:3]: Not used Bit[2:0]: pll1_pre_div 000: /1 001: /1.5 010: /2 011: /2.5 100: /3 101: /4 110: /6 111: /8
0x0301	PLL_CTRL_1	0x00	RW	Bit[7:2]: Not used Bit[1:0]: pll1_multiplier[9:8]
0x0302	PLL_CTRL_2	0x19	RW	Bit[7:0]: pll1_multiplier[7:0]
0x0303	PLL_CTRL_3	0x00	RW	Bit[7:4]: Not used Bit[3:0]: pll1_divm 1+pll1_divm
0x0304	PLL_CTRL_4	0x3	RW	Bit[7:2]: Not used Bit[1:0]: pll1_div_mipi 00: /4 01: /5 10: /6 11: /8
0x0305	PLL_CTRL_5	0x01	RW	Bit[7:2]: Not used Bit[1:0]: pll1_div_sp 00: /3 01: /4 10: /5 11: /6

table 6-1 PLL control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x0306	PLL_CTRL_6	0x01	RW	Bit[7:2]: Not used Bit[0]: pll1_div_s 0: /1 1: /2
0x0308	PLL_CTRL_8	0x00	RW	Bit[7:2]: Not used Bit[0]: pll1_bypass
0x0309	PLL_CTRL_9	0x01	RW	Bit[7:2]: Not used Bit[2:0]: pll1_cp
0x030A	PLL_CTRL_A	0x00	RW	Bit[7:4]: Not used Bit[3:1]: pll1_reserve Bit[0]: pll1_predivp 0: /1 1: /2
0x030B	PLL_CTRL_B	0x00	RW	Bit[7:3]: Not used Bit[2:0]: pll2_pre_div 000: /1 001: /1.5 010: /2 011: /2.5 100: /3 101: /4 110: /6 111: /8
0x030C	PLL_CTRL_C	0x00	RW	Bit[7:2]: Not used Bit[1:0]: pll2_r_divp[9:8]
0x030D	PLL_CTRL_D	0x1E	RW	Bit[7:0]: pll2_r_divp[7:0]
0x030E	PLL_CTRL_E	0x02	RW	Bit[7:3]: Not used Bit[2:0]: pll2_r_divs 000: /1 001: /1.5 010: /2 011: /2.5 100: /3 101: /3.5 110: /4 111: /5
0x030F	PLL_CTRL_F	0x02	RW	Bit[7:4]: Not used Bit[3:0]: pll2_r_divsp 1+pll2_r_divsp
0x0310	PLL_CTRL_10	0x01	RW	Bit[7:3]: Not used Bit[2:0]: pll2_r_cp
0x0311	PLL_CTRL_11	0x00	RW	Bit[7:1]: Not used Bit[0]: pll2_bypass

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

address	register name	default value	R/W	description
0x0312	PLL_CTRL_12	0x01	RW	Bit[7:5]: pll2_reserve Bit[4]: pll2_pre_div0 0: /1 1: /2 Bit[3:0]: pll2_r_divdac 1+pll2_r_divdac
0x031B	PLL_CTRL_1B	0x00	RW	Bit[7:1]: Not used Bit[0]: pll1_rst
0x031C	PLL_CTRL_1C	0x00	RW	Bit[7:1]: Not used Bit[0]: pll2_rst
0x031E	PLL_CTRL_1E	0x04	RW	Bit[7:4]: Not used Bit[3]: pll1_no_lat Bit[2]: Not used Bit[1:0]: mipi_bitsel_man

6.2 system control [0x3000 - 0x3043]

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

address	register name	default value	R/W	description
0x3000	PAD OEN0	0x00	RW	Bit[7:6]: Not used Bit[5]: io_fsin_oen Bit[4:0]: Not used
0x3002	PAD OEN2	0x20	RW	Bit[7]: io_vsync_oen Bit[6]: io_href_oen Bit[5]: Not used Bit[4]: Debug mode Bit[3:1]: Not used Bit[0]: io_gpio_oen
0x3003	GPIO IN	–	R	Bit[7:4]: Not used Bit[3]: io_fsin_i Bit[2]: io_href_i Bit[1]: io_vsync_i Bit[0]: io_gpio_i
0x3004	SCCB ID	0x6C	RW	Bit[7:0]: sccb_id SCCB programmed ID when SID = 0

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

address	register name	default value	R/W	description
0x3005	CLKRST5	0xF0	RW	Bit[7:6]: Not used Bit[5]: sclk_psram Bit[4]: sclk_syncfifo Bit[3:2]: Not used Bit[1]: rst_psram Bit[0]: rst_syncfifo
0x3006	SCCB ID2	0x42	RW	Bit[7:0]: sccb_id2 SCCB ID2, SCCB will ack ID2 and ID1
0x3007	R ISPOUT BITSEL	0x20	RW	Bit[7]: pll12_daclk_sel Bit[6]: r_pump_clk_sel 0: pll2_sclk 1: pll1_sclk Bit[5]: r_ispin_array_addr_sel Bit[4]: r_ilpwm_out_sel Bit[3]: r_rst_pll_sleep_dis Bit[2]: r_db_out_en Bit[1:0]: r_vsync_sel
0x300A	CHIP ID	0x00	R	Bit[7:0]: chip_id[23:16]
0x300B	CHIP ID	0x88	R	Bit[7:0]: chip_id[15:8]
0x300C	CHIP ID	0x65	R	Bit[7:0]: chip_id[7:0]
0x300D	PAD OUT2	0x00	RW	Bit[7]: io_vsync_o Bit[6]: io_href_o Bit[5]: Not used Bit[4]: Chip debug Bit[3]: io_strobe_o Bit[2]: io_sda_o Bit[1]: io_ilpwm_o Bit[0]: io_gpio_o
0x3010	PAD SEL2	0x00	RW	Bit[7]: io_vsync_sel Bit[6]: io_href_sel Bit[5]: Not used Bit[4]: Chip debug Bit[3]: io_strobe_sel Bit[2]: io_sda_sel Bit[1]: io_ilpwm_sel Bit[0]: io_gpio_sel
0x3011	PAD	0x00	RW	Bit[6:5]: Pad drive strength 00: 1x 01: 2x 10: 3x 11: 4x
0x3012	SCCB R12	0x20	RW	Bit[7:0]: sccb_id when SID = 1

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

address	register name	default value	R/W	description
0x3015	PUMP CLK DIV	0x00	RW	Bit[7]: Not used Bit[6:4]: Npump clock div 000: /2 001: /4 010: /8 011: /16 100: /32 1xx: Disable pump_clk Bit[3]: Not used Bit[2:0]: Ppump clock div 000: /2 001: /4 010: /8 011: /16 100: /32 1xx: Disable pump_clk
0x3018	MIPI SC CTRL	0x72	RW	Bit[7:5]: mipi_lane_mode N+1 lane Bit[4]: mipi_lvds_sel 0: LVDS enable 1: MIPI enable Bit[3:2]: r_phy_pd_mipi_man Bit[1]: Reserved Bit[0]: lane_dis option 1: Disable lanes when pd_mipi
0x3019	MIPI SC CTRL	0x00	RW	Bit[7:0]: MIPI lane disable manual
0x301A	CLKRST0	0xF0	RW	Bit[7]: Reserved Bit[6]: sclk_stb Bit[5]: sclk_ac Bit[4]: sclk_tc Bit[3]: mipi_phy_rst_o Bit[2]: rst_stb Bit[1]: rst_ac Bit[0]: rst_tc
0x301B	CLKRST1	0xF0	RW	Bit[7]: sclk_blc Bit[6]: sclk_isp Bit[5]: sclk_testmode Bit[4]: sclk_vfifo Bit[3]: rst_blc Bit[2]: rst_isp Bit[1]: rst_testmode Bit[0]: rst_vfifo

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

address	register name	default value	R/W	description
0x301C	CLKRST2	0xF0	RW	Bit[7]: Not used Bit[6]: sclk_mipi Bit[5]: sclk_dpcm Bit[4]: sclk_otp Bit[3]: Not used Bit[2]: rst_mipi Bit[1]: rst_dpcm Bit[0]: rst_otp
0x301D	CLKRST3	0xF0	RW	Bit[7]: sclk_asram_tst Bit[6]: sclk_grp Bit[5]: sclk_bist Bit[4]: Reserved Bit[3]: rst_asram_tst Bit[2]: rst_grp Bit[1]: rst_bist Bit[0]: rst_aec_pk
0x301E	CLKRST4	0xF0	RW	Bit[7]: sclk_ilpwm Bit[6]: pclk_lvds Bit[5]: pclk_vfifo Bit[4]: pclk_mipi Bit[3]: rst_ilpwm Bit[2]: rst_lvds Bit[1:0]: Not used
0x301F	CHIP DEBUG	–	–	Chip Debug
0x3020	CLOCK SEL	0x93	RW	Bit[7]: Clock switch output 0: Padclk 1: Normal Bit[6:4]: Not used Bit[3]: pclk_div 0: /1 1: /2 Bit[2:0]: Not used
0x3021	MISC CTRL	0x23	RW	Bit[7]: Not used Bit[6]: Sleep no latch option 1: No latch Bit[5]: fst_stby_ctr 0: Software standby enter at v_blk 1: Software standby enter at l_blk or v_blk Bit[4:1]: Not used Bit[0]: cen_global_o Global control all CEN of SRAM

table 6-2 system control registers (sheet 5 of 6)

address	register name	default value	R/W	description
0x3022	MIPI SC CTRL	0x01	RW	Bit[7:4]: Not used Bit[3]: lvds_mode_o Bit[2]: Not used Bit[1]: Clock lane disable when pd_mipi Bit[0]: pd_mipi enable when rst_sync
0x3023	MIPI LPTX SEL	0x00	RW	Bit[7:0]: Not used
0x3024	REG24	0x10	RW	Bit[7:3]: Not used Bit[2]: rst_ana manual Bit[1:0]: Not used
0x302A	SUB ID	0xB0	R	Bit[7:4]: Process Bit[3:0]: Version
0x3030	REG30	0x00	RW	Bit[7:6]: Not used Bit[5]: SCLK inv Bit[4]: PCLK inv Bit[3:0]: Not used
0x3031	REG31	0x0A	RW	Bit[7:5]: Not used Bit[4:0]: mipi_bit_sel 0x8: 8-bit mode 0xA: 10-bit mode 0xC: 12-bit mode Others: Not used
0x3032	REG32	0x80	RW	Bit[7]: pll2_sysclk_sel Bit[6]: asram_clk_sel Bit[5]: array_hskip_man_en Bit[4]: r_rst_otp_sleep_dis Bit[3]: r_rst_ana_sleep_dis Bit[2:0]: array_hskip_man[3:1]
0x3033	REG33	0x24	RW	Bit[7]: array_hskip_man[0] Bit[6]: Not used Bit[5]: r_fmt_eof_sel 0: isp_eof 1: mipi_eof Bit[4]: sync_point_sel Bit[3]: rip_sof_en Bit[2]: Not used Bit[1]: pll_sysclk_sel Bit[0]: lvds_ck_data_sel
0x3037	PLL1 CTR1	0x00	RW	Bit[7:1]: Not used Bit[0]: sid_rev
0x303E	ENCLK_SEL	0x03	RW	Bit[7:3]: Not used Bit[2:0]: Stream on delay 2^9/2^10.../2^16

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

address	register name	default value	R/W	description
0x303F	CTRL3F	0x00	RW	Bit[7:2]: Not used Bit[1]: sccb_id2_nack Nack to ID2 Bit[0]: sccb_pgm_id_en
0x3040	CTRL00	0xF0	RW	Bit[7]: sclk_isp_fc_en Bit[6]: sclk_fc_en Bit[5]: sclk_tpm_en Bit[4]: sclk_fmt_en Bit[3]: rst_isp_fc Bit[2]: rst_fc Bit[1]: rst_tpm Bit[0]: rst_fmt
0x3043	CHIP DEBUG	–	–	Chip Debug

6.3 SCCB control [0x3100 - 0x314F]

table 6-3 SCCB registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x3100	SB_SCCB_CTRL	0x00	RW	Bit[7:0]: Debug mode
0x3101	SCCB OPT	0x32	RW	Bit[7:6]: Not used Bit[5]: Re-mapping enable Bit[4]: en_ss_addr_inc Bit[3:0]: Debug mode
0x3102	SCCB FILTER	0x00	RW	Bit[7:0]: Debug mode
0x3103	SCCB SYSREG	0x00	RW	Bit[7:0]: Not used
0x3104	PWUP DIS	0x01	RW	Bit[7:0]: Not used
0x3105	SB_PADCLK_DIV	0x11	RW	Bit[7:6]: Debug mode Bit[5:0]: Chip debug
0x3106	SRB HOST INPUT DIS	0x01	RW	Bit[7:4]: sclk_div /1/1/2/3.../15 Bit[3:2]: sclk_pre_div 00: /1 01: /2 10: /4 11: /1 Bit[1:0]: Chip debug

table 6-3 SCCB registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x3108	DESTINATION RE-MAPPING ADDR	0x90	RW	High Byte of Destination Address for Re-mapping Function
0x3109	DESTINATION RE-MAPPING ADDR	0x00	RW	Low Byte of Destination Address for Re-mapping Function
0x3110	SRC ADDR00 H	0x00	RW	High Byte of Number 00 Source Register Address
0x3111	SRC ADDR00 L	0x00	RW	Low Byte of Number 00 Source Register Address
0x3112	SRC ADDR01	0x00	RW	High Byte of Number 01 Source Register Address
0x3113	SRC ADDR01	0x00	RW	Low Byte of Number 01 Source Register Address
0x3114	SRC ADDR02 H	0x00	RW	High Byte of Number 02 Source Register Address
0x3115	SRC ADDR02 L	0x00	RW	Low Byte of Number 02 Source Register Address
0x3116	SRC ADDR03 H	0x00	RW	High Byte of Number 03 Source Register Address
0x3117	SRC ADDR03 L	0x00	RW	Low Byte of Number 03 Source Register Address
0x3118	SRC ADDR04 H	0x00	RW	High Byte of Number 04 Source Register Address
0x3119	SRC ADDR04 L	0x00	RW	Low Byte of Number 04 Source Register Address
0x311A	SRC ADDR05 H	0x00	RW	High Byte of Number 05 Source Register Address
0x311B	SRC ADDR05 L	0x00	RW	Low Byte of Number 05 Source Register Address
0x311C	SRC ADDR06 H	0x00	RW	High Byte of Number 06 Source Register Address
0x311D	SRC ADDR06 L	0x00	RW	Low Byte of Number 06 Source Register Address
0x311E	SRC ADDR07 H	0x00	RW	High Byte of Number 07 Source Register Address
0x311F	SRC ADDR07 L	0x00	RW	Low Byte of Number 07 Source Register Address
0x3120	SRC ADDR08 H	0x00	RW	High Byte of Number 08 Source Register Address
0x3121	SRC ADDR08 L	0x00	RW	Low Byte of Number 08 Source Register Address
0x3122	SRC ADDR09 H	0x00	RW	High Byte of Number 09 Source Register Address
0x3123	SRC ADDR09 L	0x00	RW	Low Byte of Number 09 Source Register Address
0x3124	SRC ADDR0A H	0x00	RW	High Byte of Number 0A Source Register Address
0x3125	SRC ADDR0A L	0x00	RW	Low Byte of Number 0A Source Register Address
0x3126	SRC ADDR0B H	0x00	RW	High Byte of Number 0B Source Register Address
0x3127	SRC ADDR0B L	0x00	RW	Low Byte of Number 0B Source Register Address
0x3128	SRC ADDR0C H	0x00	RW	High Byte of Number 0C Source Register Address
0x3129	SRC ADDR0C L	0x00	RW	Low Byte of Number 0C Source Register Address
0x312A	SRC ADDR0D H	0x00	RW	High Byte of Number 0D Source Register Address

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

address	register name	default value	R/W	description
0x312B	SRC ADDR0D L	0x00	RW	Low Byte of Number 0D Source Register Address
0x312C	SRC ADDR0E H	0x00	RW	High Byte of Number 0E Source Register Address
0x312D	SRC ADDR0E L	0x00	RW	Low Byte of Number 0E Source Register Address
0x312E	SRC ADDR0F H	0x00	RW	High Byte of Number 0F Source Register Address
0x312F	SRC ADDR0F L	0x00	RW	Low Byte of Number 0F Source Register Address
0x3130	SRC ADDR10 H	0x00	RW	High Byte of Number 10 Source Register Address
0x3131	SRC ADDR10 L	0x00	RW	Low Byte of Number 10 Source Register Address
0x3132	SRC ADDR11 H	0x00	RW	High Byte of Number 11 Source Register Address
0x3133	SRC ADDR11 L	0x00	RW	Low Byte of Number 11 Source Register Address
0x3134	SRC ADDR12 H	0x00	RW	High Byte of Number 12 Source Register Address
0x3135	SRC ADDR12 L	0x00	RW	Low Byte of Number 12 Source Register Address
0x3136	SRC ADDR13 H	0x00	RW	High Byte of Number 13 Source Register Address
0x3137	SRC ADDR13 L	0x00	RW	Low Byte of Number 13 Source Register Address
0x3138	SRC ADDR14 H	0x00	RW	High Byte of Number 14 Source Register Address
0x3139	SRC ADDR14 L	0x00	RW	Low Byte of Number 14 Source Register Address
0x313A	SRC ADDR15 H	0x00	RW	High Byte of Number 15 Source Register Address
0x313B	SRC ADDR15 L	0x00	RW	Low Byte of Number 15 Source Register Address
0x313C	SRC ADDR16 H	0x00	RW	High Byte of Number 16 Source Register Address
0x313D	RATT ADDR16 L	0x00	RW	Low Byte of Number 16 Source Register Address
0x313E	SRC ADDR17 H	0x00	RW	High Byte of Number 17 Source Register Address
0x313F	SRC ADDR17 L	0x00	RW	Low Byte of Number 17 Source Register Address
0x3140	SRC ADDR18 H	0x00	RW	High Byte of Number 18 Source Register Address
0x3141	SRC ADDR18 L	0x00	RW	Low Byte of Number 18 Source Register Address
0x3142	SRC ADDR19 H	0x00	RW	High Byte of Number 19 Source Register Address
0x3143	SRC ADDR19 L	0x00	RW	Low Byte of Number 19 Source Register Address
0x3144	SRC ADDR1A H	0x00	RW	High Byte of Number 1A Source Register Address
0x3145	SRC ADDR1A L	0x00	RW	Low Byte of Number 1A Source Register Address
0x3146	SRC ADDR1B H	0x00	RW	High Byte of Number 1B Source Register Address
0x3147	SRC ADDR1B L	0x00	RW	Low Byte of Number 1B Source Register Address
0x3148	SRC ADDR1C H	0x00	RW	High Byte of Number 1C Source Register Address

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

address	register name	default value	R/W	description
0x3149	SRC ADDR1C L	0x00	RW	Low Byte of Number 1C Source Register Address
0x314A	SRC ADDR1D H	0x00	RW	High Byte of Number 1D Source Register Address
0x314B	SRC ADDR1D L	0x00	RW	Low Byte of Number 1D Source Register Address
0x314C	SRC ADDR1E H	0x00	RW	High Byte of Number 1E Source Register Address
0x314D	SRC ADDR1E L	0x00	RW	Low Byte of Number 1E Source Register Address
0x314E	SRC ADDR1F H	0x00	RW	High Byte of Number 1F Source Register Address
0x314F	SRC ADDR1F L	0x00	RW	Low Byte of Number 1F Source Register Address

6.4 group hold [0x3200 - 0x320F]

table 6-4 group hold registers (sheet 1 of 2)

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	0x10	RW	Group1 Start Address in SRAM, Actual Address is {0x3201[5:0], 4'h0}
0x3202	GROUP ADR2	0x20	RW	Group2 Start Address in SRAM, Actual Address is {0x3202[5:0], 4'h0}
0x3203	GROUP ADR3	0x30	RW	Group3 Start Address in SRAM, Actual Address is {0x3203[5:0], 4'h0}
0x3204	GROUP LEN0	–	R	Length of Group0
0x3205	GROUP LEN1	–	R	Length of Group1
0x3206	GROUP LEN2	–	R	Length of Group2
0x3207	GROUP LEN3	–	R	Length of Group3

table 6-4 group hold registers (sheet 2 of 2)

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	Bit[7]: Not used Bit[6:5]: Switch back group Bit[4:0]: Number of frames to stay in group 0
0x320A	GROUP1 PERIOD	0x00	RW	Number of Frames to Stay in Group 1
0x320B	GRP_SW_CTRL	0x11	RW	Bit[7]: auto_sw Bit[6:5]: Not used Bit[4]: frame_cnt_trig Bit[3]: group_switch_repeat Bit[2]: context_en Bit[1:0]: Second group select
0x320C	SRAM TEST	0x0A	RW	Bit[7:5]: Not used Bit[4]: Group hold SRAM test enable Bit[3:0]: Group hold SRAM RM[3:0]
0x320D	GRP_ACT	–	R	Active Group Indicator
0x320E	FM_CNT_GRP0	–	R	Group 0 Frame Count
0x320F	FM_CNT_GRP1	–	R	Group 1 Frame Count

6.5 AEC PK control [0x3500 - 0x350B]

table 6-5 AEC PK control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3500	AEC EXPO	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Exposure[19:16]
0x3501	AEC EXPO	0x02	RW	Bit[7:0]: Exposure[15:8]

table 6-5 AEC PK control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3502	AEC EXPO	0x00	RW	Bit[7:0]: Exposure[7:0] Low 4 bits are fraction bits Minimum exposure value is 2 lines
0x3503	AEC MANUAL	0x00	RW	Bit[7]: Not used Bit[6]: Digital fraction gain delay option 0: Delay 1 frame 1: Not delay 1 frame Bit[5]: Gain change delay option 0: Delay 1 frame 1: Not delay 1 frame Bit[4]: Gain delay option 0: Delay 1 frame 1: 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
0x3504	NOT USED	–	–	Not Used
0x3505	GCVT OPTION	0x80	RW	Gain Conversation Option Bit[7]: DAC fixed gain bit Bit[6]: Not used Bit[5:4]: 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	NOT USED	–	–	Not Used
0x3507	AEC GAIN SHIFT	0x00	RW	Bit[7:2]: Not used 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	AEC GAIN	0x00	RW	Bit[7:5]: Not used Bit[4:0]: Gain[12:8]

table 6-5 AEC PK control registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x3509	AEC GAIN	0x80	RW	Bit[7:0]: Gain[7:0] 0x3503[2]=0, 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, 0x100 is 2x gain If 0x3503[2]=1, gain[12:0] is sensor gain format, gain[12:8] is coarse gain, 00000: 1x, 00001: 2x, 00011: 4x, 00111: 8x, gain[7] is 1, gain[6:3] is fine gain, gain[2:0] is always 0. For example, 0x080 is 1x gain, 0x180 is 2x gain, 0x380 is 4x gain
0x350A	AEC DIGIGAIN	0x08	RW	Bit[7:0]: Digital gain[13:6]
0x350B	AEC DIGIGAIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: Digital gain[5:0]

6.6 ANA control [0x3600 - 0x3650]

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

address	register name	default value	R/W	description
0x3600	ASP_CTRL00	0x00	RW	Bit[7:0]: Not used
0x3601~ 0x3638	ANALOG_CTRL	–	–	Analog Control Registers
0x3640	ASP_CTRL40	0x00	RW	Bit[7:3]: Not used Bit[2:0]: mipi_ck0_skew_o
0x3641	ASP_CTRL41	0x00	RW	Bit[7]: Not used Bit[6:4]: mipi_skew1_o Bit[3]: Not used Bit[2:0]: mipi_skew0_o
0x3642	ASP_CTRL42	0x00	RW	Bit[7]: Not used Bit[6:4]: mipi_skew3_o Bit[3]: Not used Bit[2:0]: mipi_skew2_o
0x3643	ASP_CTRL43	0x00	RW	Bit[7:0]: Not used
0x3644	ASP_CTRL44	0x00	RW	Bit[7:0]: Not used

table 6-6 ANA control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3645	ASP_CTRL45	0x03	RW	Bit[7:5]: Not used Bit[4:3]: mipi_r_iref_o Bit[2:0]: mipi_r_lvcm_o
0x3646	ASP_CTRL46	0x82	RW	Bit[7:3]: mipi_vref_o Bit[2:0]: mipi_r_mvcm_o
0x3647	ASP_CTRL47	0x10	RW	Bit[7:6]: Not used Bit[5:4]: mipi_pgm_lptx_o Bit[3:2]: mipi_valid2dck_dly_o Bit[1:0]: mipi_valid2d_dly_o
0x3649	ASP_CTRL49	0x38	RW	Bit[7:0]: Not used
0x364A	ASP_CTRL50	0x05	RW	Bit[7:6]: Not used Bit[5]: mipi_en_reg_o Bit[4]: mipi_pgm_bp_hsen_lat_o Bit[3:0]: mipi_sel_drv_o
0x364B	ASP_CTRL51	0x88	RW	Bit[7:0]: Not used
0x364C	ASP_CTRL52	0x00	RW	Bit[7:3]: Not used Bit[2:0]: mipi_slew_rate_o
0x3650	ASP_CTRL53	0x00	RW	Bit[7:0]: d_tpm_o

6.7 sensor control registers [0x3700 - 0x37B9]

table 6-7 sensor control registers

address	register name	default value	R/W	description
0x3700~ 0x37B9	SENSOR_CTRL	–	–	Sensor Control Registers

6.8 chip debug [0x37C5 - 0x37DF]

table 6-8 chip debug registers

address	register name	default value	R/W	description
0x37C5~ 0x37DF	CHIP DEBUG	–	–	Chip Debug

6.9 timing control [0x3800 - 0x3872]

table 6-9 timing control registers (sheet 1 of 5)

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 high byte
0x3801	X ADDR START	0x0C	RW	Bit[7:0]: x_addr_start[7:0] Array horizontal start point low byte
0x3802	Y ADDR START	0x00	RW	Bit[3:0]: y_addr_start[11:8] Array vertical start point high byte
0x3803	Y ADDR START	0x0C	RW	Bit[7:0]: y_addr_start[7:0] Array vertical start point low byte
0x3804	X ADDR END	0x0C	RW	Bit[3:0]: x_addr_end[11:8] Array horizontal end point high byte
0x3805	X ADDR END	0xD3	RW	Bit[7:0]: x_addr_end[7:0] Array horizontal end point low byte
0x3806	Y ADDR END	0x09	RW	Bit[3:0]: y_addr_end[11:8] Array vertical end point high byte
0x3807	Y ADDR END	0xA3	RW	Bit[7:0]: y_addr_end[7:0] Array vertical end point low byte
0x3808	X OUTPUT SIZE	0x0C	RW	Bit[3:0]: x_output_size[11:8] ISP horizontal output width high byte
0x3809	X OUTPUT SIZE	0xC0	RW	Bit[7:0]: x_output_size[7:0] ISP horizontal output width low byte
0x380A	Y OUTPUT SIZE	0x09	RW	Bit[3:0]: y_output_size[11:8] ISP vertical output height high byte
0x380B	Y OUTPUT SIZE	0x90	RW	Bit[7:0]: y_output_size[7:0] ISP vertical output height low byte
0x380C	HTS	0x07	RW	Bit[7:0]: HTS[15:8] Total horizontal timing size high byte
0x380D	HTS	0x4C	RW	Bit[7:0]: HTS[7:0] Total horizontal timing size low byte
0x380E	VTS	0x0A	RW	Bit[6:0]: VTS[14:8] Total vertical timing size high byte
0x380F	VTS	0x74	RW	Bit[7:0]: VTS[7:0] Total vertical timing size low byte
0x3810	ISP X WIN	0x00	RW	Bit[7:0]: isp_x_win[15:8] ISP horizontal windowing offset high byte

table 6-9 timing control registers (sheet 2 of 5)

address	register name	default value	R/W	description
0x3811	ISP X WIN	0x04	RW	Bit[7:0]: isp_x_win[7:0] ISP horizontal windowing offset low byte
0x3812	ISP Y WIN	0x00	RW	Bit[3:0]: isp_y_win[11:8] ISP vertical windowing offset high byte
0x3813	ISP Y WIN	0x02	RW	Bit[7:0]: isp_y_win[7:0] ISP vertical windowing offset low byte
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	VSYNC START	0x00	RW	Bit[7:0]: vsync_start[15:8] VSYNC start point high byte
0x3817	VSYNC START	0x00	RW	Bit[7:0]: vsync_start[7:0] VSYNC start point low byte
0x3818	VSYNC END	0x00	RW	Bit[7:0]: vsync_end[15:8] VSYNC end point high byte
0x3819	VSYNC END	0x00	RW	Bit[7:0]: vsync_end[7:0] VSYNC end point low byte
0x381A	HSYNC FIRST H	0x04	RW	Bit[7:0]: hsync_first[15:8] HSYNC first active row start position high byte
0x381B	HSYNC FIRST L	0x00	RW	Bit[7:0]: hsync_first[7:0] HSYNC first active row start position low byte
0x3820	FORMAT1	0x00	RW	Format1 Bit[7]: vsub48_blc Bit[6:4]: Not used Bit[3]: byp_isp_o Bit[2]: vflip_dig Bit[1]: vflip_arr Bit[0]: Not used
0x3821	FORMAT2	0x40	RW	Format2 Bit[7]: snr_ctrl_vbin2 Bit[6]: hsync_en_o Bit[5]: fst_vbin Bit[4]: fst_hbin Bit[3]: isp_hvar2 Bit[2]: mirror_dig Bit[1]: mirror_arr Bit[0]: sync_hbin2

table 6-9 timing control registers (sheet 3 of 5)

address	register name	default value	R/W	description
0x3822	REG22	0x88	RW	Bit[7:5]: addr0_num[3:1] Bit[4:0]: ablc_num[5:1]
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
0x3824	CS RST FSIN	0x00	RW	Bit[7:0]: cs_rst_fsin[15:8] CS reset value high byte at vs_ext
0x3825	CS RST FSIN	0x20	RW	Bit[7:0]: cs_rst_fsin[7:0] CS reset value low byte at vs_ext
0x3826	R RST FSIN	0x00	RW	Bit[7:0]: r_rst_fsin[15:8] R reset value high byte at vs_ext
0x3827	R RST FSIN	0x04	RW	Bit[7:0]: r_rst_fsin[7:0] R reset value low byte at vs_ext
0x3828	REG28	0x00	RW	Bit[7]: ext_hs_re Bit[6]: ext_hs_en Bit[5]: asp_start_sel 0: Use sync output 1: Use sensor output Bit[4]: hts_inc_en Bit[3]: r_gate_vs_b Bit[2]: VSYNC polarity Bit[1:0]: href_w
0x382A	Y INC ODD	0x01	RW	Bit[7:5]: Not used Bit[4:0]: y_odd_inc
0x382B	Y INC EVEN	0x01	RW	Bit[7:5]: Not used Bit[4:0]: y_even_inc
0x382C	BLC COL ST L	0x00	RW	Bit[7:0]: blc_col_st_l[7:0] Left black column start address
0x382D	BLC COL END L	0xFF	RW	Bit[7:0]: blc_col_end_l[7:0] Left black column end address
0x382E	BLC COL ST R	0x00	RW	Bit[7]: Not used Bit[6:0]: blc_col_st_r[6:0] Right black column start address
0x382F	BLC COL END R	0x00	RW	Bit[7]: Not used Bit[6:0]: blc_col_end_r[6:0] Right black column end address
0x3830	BLC NUM OPTION	0x04	RW	Bit[7:5]: ablc_adj Bit[4:0]: ablc_use_num[5:1]

table 6-9 timing control registers (sheet 4 of 5)

address	register name	default value	R/W	description
0x3831	BLC NUM MAN	0x00	RW	Bit[7:6]: Not used Bit[5]: man_blc_num_sel Bit[4:0]: man_blc_num[5:1]
0x3832	FRACTION HTS	0x00	RW	Bit[7:0]: fraction_hts[15:8]
0x3833	FRACTION HTS	0x00	RW	Bit[7:0]: fraction_hts[7:0]
0x3834	EXT DIV FACTORS	0x01	RW	Bit[7:4]: ext_vs_div Bit[3:0]: ext_hs_div
0x3835	GROUP WRITE OPTION	0x01	RW	Group_write_option Bit[7:5]: Not used Bit[4]: r_grp_wr_pt_sel 0: Gain trigger 1: End of frame Bit[3:0]: Group write adjust number
0x3836	ZLINE NUM OPTION	0x01	RW	Bit[7:5]: Not used Bit[4:0]: zline_use_num[5:1]
0x3837	RGBC	0x00	RW	Bit[7]: r_rgb_only Bit[6]: r_w_only Bit[5]: r_rgbc_bin Bit[4]: vts_add_dis Bit[3:0]: cexp_gt_vts offs
0x3840	GRP WR MAN	–	W	Bit[7:0]: grp_wr_man Writing to this register will generate manual group command
0x3841	AUTO SIZE CTRL0	0xFF	RW	Bit[7:6]: Not used Bit[5]: r_auto_y_win_en Bit[4]: r_auto_x_win_en Bit[3]: r_auto_y_end_en Bit[2]: r_auto_x_end_en Bit[1]: r_auto_y_start_en Bit[0]: r_auto_x_start_en
0x3842	X ADD OFF H	0x00	RW	Bit[7:4]: Not used Bit[3:0]: x_add_off[11:8] High byte of horizontal offset for auto size mode (complemental code)
0x3843	X ADD OFF L	0x00	RW	Bit[7:0]: x_add_off[7:0] Low byte of horizontal offset for auto size mode (complemental code)
0x3844	Y ADD OFF H	0x00	RW	Bit[7:4]: Not used Bit[3:0]: y_add_off[11:8] High byte of vertical offset for auto size mode (complemental code)

table 6-9 timing control registers (sheet 5 of 5)

address	register name	default value	R/W	description
0x3845	Y ADD OFF L	0x00	RW	Bit[7:0]: y_add_off[7:0] Low byte of vertical offset for auto size mode (complemental code)
0x3846	BOUNDARY PIX NUM	0x48	RW	Bit[7:4]: Y boundary pixel number for auto size mode Bit[3:0]: X boundary pixel number for auto size mode
0x3870	ROW COUNTER LATCH	–	R	Bit[7:0]: row_counter_latch Reading this register will trigger a latch of row counter
0x3871	ROW COUNTER H	–	R	Bit[7:0]: row_counter[15:8]
0x3872	ROW COUNTER L	–	R	Bit[7:0]: row_counter[7:0]

6.10 strobe [0x3B00 - 0x3B05]

table 6-10 strobe control registers (sheet 1 of 2)

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]: 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 high byte
0x3B03	STROBE ADD DUMMY	0x00	RW	Bit[7:0]: strobe_add_dummy[7:0] Dummy line number added at strobe low byte

table 6-10 strobe control registers (sheet 2 of 2)

address	register name	default value	R/W	description
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 frame later 10: Delay one frame, strobe generated 3 frame later 11: Delay one frame, strobe generated 4 frame 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

6.11 illumination PWM [0x3B40 - 0x3B52]

table 6-11 illumination PWN registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3B40	P1 DLY	0x10	RW	First Pulse Delay (0 ~ 31) 0x00: -0.5 frame 0x1F: 0.5 frame
0x3B41	P2 DLY	0x10	RW	Second Pulse Delay (0 ~ 31) 0x00: -0.5 frame 0x1F: 0.5 frame
0x3B42	P3 DLY	0x10	RW	Third Pulse Delay (0 ~ 31) 0x00: -0.5 frame 0x1F: 0.5 frame
0x3B43	P4 DLY	0x10	RW	Fourth Pulse Delay (0 ~ 31) 0x00: -0.5 frame 0x1F: 0.5 frame
0x3B44	DURATION CTRL0	0x11	RW	Bit[7:4]: Duration2 Second pulse duration (0~15 frames) Bit[3:0]: Duration1 First pulse duration (0~15 frames)

table 6-11 illumination PWN registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3B45	DURATION CTRL1	0x11	RW	Bit[7:4]: Duration4 Fourth pulse duration (0~15 frames) Bit[3:0]: Duration3 Third pulse duration (0~15 frames)
0x3B46	P0 DUTY	0x1F	RW	Bit[7:5]: Not used Bit[4:0]: duty_cycle1 first pulse duty cycle 0~31
0x3B47	P1 DUTY STEP	0x1F	RW	Bit[7:5]: Not used Bit[4:0]: duty_step2 Second pulse duty cycle step
0x3B48	P2 DUTY	0x1F	RW	Bit[7:5]: Not used Bit[4:0]: duty_cycle3 Third pulse duty cycle 0~31
0x3B49	P3 DUTY STEP	0x1F	RW	Bit[7:5]: Not used Bit[4:0]: duty_step4 Fourth pulse duty cycle step
0x3B4A	GAP CTRL1	0x00	RW	Bit[7:5]: Not used Bit[4:0]: Gap1 between pulse 0 and pulse 1 0~255 frames
0x3B4B	GAP CTRL2	0x00	RW	Bit[7:0]: Gap2 between pulse 1 and pulse 2
0x3B4C	GAP CTRL3	0x00	RW	Bit[7:0]: Gap3 between pulse 2 and pulse 3
0x3B4D	GAP CTRL4	0x00	RW	Bit[7:0]: Gap4 between pulse 3 and pulse 0
0x3B4E	PWM CTRL	0x00	RW	Bit[7]: pwm_req_r Bit[6]: Not used Bit[5]: illum_sel Bit[4]: duty_no_map Bit[3]: no_gap Bit[2]: sel_slot_out Bit[1]: Manual setting duty cycle for duration1 and duration3 Bit[0]: pwm_repeat
0x3B4F	SLOT WIDTH	0x02	RW	Bit[7:4]: Not used Bit[3:0]: slot_width_r
0x3B50	RAMP2 XSTEP	0x01	RW	Bit[7:4]: Not used Bit[3:0]: ramp2_xstep_r Second pulse duty cycle step
0x3B51	RAMPR4 XSTEP	0x01	RW	Bit[7:4]: Not used Bit[3:0]: rampr4_xstep_r Fourth pulse duty cycle step

table 6-11 illumination PWN registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x3B52	TAIL DUTY CYCLE	0x80	RW	Bit[7]: end_opt 0: No pulse when PWM end 1: Free running at pre-defined duty cycle Bit[6]: tail_stop_toggle Bit[5]: Not used Bit[4:0]: duty_tail Tail pulse duty cycle step

6.12 OTP control registers [0x3D80 - 0x3D8F]

table 6-12 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_LOA_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	0x55	RW	Program Strobe Pulse Width Unit: 8×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[5:0]: Debug mode
0x3D85	OTP_REG85	0x13	RW	Bit[7:3]: Debug mode Bit[2]: OTP power up load data enable Bit[1]: OTP power up load setting enable Bit[0]: OTP write register load setting enable

table 6-12 OTP control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3D86	SRAM_TEST_ SIGNALS	0x02	RW	Bit[7:4]: Debug mode Bit[3]: rst_otp_manual Bit[2]: r_test Bit[1:0]: 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
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_BIST_ERR_ ADDRESS	–	R	OTP Check Error Address High
0x3D8F	OTP_BIT_ERR_ ADDRESS	–	R	OTP Check Error Address Low

6.13 PSRAM control [0x3F00 - 0x3F0F]

table 6-13 PSRAM control registers

address	register name	default value	R/W	description
0x3F00~ 0x3F0F	PSRAM_CTRL	–	–	PSRAM Control Register

6.14 BLC control [0x4000 - 0x4087]

table 6-14 BLC control registers (sheet 1 of 8)

address	register name	default value	R/W	description
0x4000	BLC CTRL00	0xF1	RW	Bit[7]: out_range_trig_en Offset out of range trigger function enable signal Bit[6]: format_chg_en Format change trigger function enable signal Bit[5]: gain_chg_en Gain change trigger function enable signal Bit[4]: exp_chg_en Exposure change trigger function enable signal Bit[3]: manual_trig Manual trigger signal Its rising edge will trigger BLC Bit[2]: freeze_en BLC freeze function enable signal When it is set, the BLC will be frozen. The offsets will keep their previous frame values. Bit[1]: always_do BLC always trigger signal When it is set, the BLC will be triggered every frame unless the freeze_en is enabled. Bit[0]: median_en 5-point median filter function enable signal
0x4001	BLC CTRL01	0x86	RW	Bit[7]: dither_en Offset dithering function enable signal Bit[6]: Blkcol zeroline diff enable Enable difference between black line column with zero line to cancel horizontal noise Bit[5:4]: Column shift option 00: Left 256 columns 01: Left 128 columns 10: Left 64 columns 11: Left 32 columns Bit[3]: Reserved Bit[2]: Final BLC offset limitation enable Bit[1]: BLC column cancel function enable Bit[0]: Chip debug
0x4002	BLC CTRL02	0x40	RW	Bit[7:0]: Cut range up threshold

table 6-14 BLC control registers (sheet 2 of 8)

address	register name	default value	R/W	description
0x4003	BLC CTRL03	0x30	RW	Bit[7:0]: Black line number used to calculate offsets
0x4004	BLC CTRL04	0x00	RW	Bit[7:0]: BLC target[15:8]
0x4005	BLC CTRL05	0x10	RW	Bit[7:0]: BLC target[7:0]
0x4006	BLC CTRL06	0x1F	RW	Bit[7:0]: Format change frame number
0x4007	BLC CTRL07	0x1F	RW	Bit[7:0]: Reset trigger frame number
0x4008	BLC CTRL08	0x01	RW	Bit[7:0]: Manual trigger frame number
0x4009	BLC CTRL09	0x29	RW	Bit[7:4]: Line number for BLC initial function Bit[3]: Bypass cut range function enable Bit[2:0]: BLC column added offset[10:8]
0x400A	BLC CTRL0A	0x00	RW	Bit[7:0]: BLC column added offset[7:0]
0x400B	BLC CTRL0B	0x0C	RW	Bit[7:4]: Start line for BLC initial function Bit[3]: Offset limitation function enable Bit[2]: Cut range function enable Bit[1:0]: BLC last line select
0x400C	BLC CTRL0C	0x00	RW	Bit[7:0]: Offset trigger threshold[15:8] When abs(line_current_offset – blc_line_offset) bigger than offset_trig_thresh, the BLC update will be set
0x400D	BLC CTRL0D	0x20	RW	Bit[7:0]: Offset trigger threshold[7:0]
0x400E	BLC CTRL0E	0x00	RW	Bit[7:0]: BLC bypass offset[15:8]
0x400F	BLC CTRL0F	0x00	RW	Bit[7:0]: BLC bypass offset[7:0]
0x4010	BLC CTRL10	0xFF	RW	Bit[7:0]: max_offset It defines top limitation for offsets. The really used max_offset is {max_offset, {(IM_DW+2-8){1'b1}}}, where IM_DW=10

table 6-14 BLC control registers (sheet 3 of 8)

address	register name	default value	R/W	description
0x4011	BLC CTRL11	0x00	RW	Bit[7:6]: Dithering offset Bit[5]: offset_man_same When it is enabled, manual offsets will be same. They are all defined by manual_offset00. Bit[4]: Offset manual mode enable When it is enabled, the used offsets will be defined manually with registers manual_offset00 ~ manual_offset11 Bit[3]: Chip debug Bit[2]: out_range trigger option Bit[1:0]: Chip debug
0x4012	BLC CTRL12	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: manual_offset00[11:8] Manual offset for normal even-line and even-column pixels
0x4013	BLC CTRL13	0x00	RW	Bit[7:0]: manual_offset00[7:0]
0x4014	BLC CTRL14	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: manual_offset01[11:8] Manual offset for normal even-line and odd-column pixels
0x4015	BLC CTRL15	0x00	RW	Bit[7:0]: manual_offset01[7:0]
0x4016	BLC CTRL16	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: manual_offset10[11:8] Manual offset for normal odd-line and even-column pixels
0x4017	BLC CTRL17	0x00	RW	Bit[7:0]: manual_offset10[7:0]
0x4018	BLC CTRL18	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: manual_offset11[11:8] Manual offset for normal odd-line and odd-column pixels
0x4019	BLC CTRL19	0x00	RW	Bit[7:0]: manual_offset11[7:0]
0x401A	BLC CTRL1A	0x00	RW	Bit[7:0]: Debug mode
0x401B	BLC CTRL1B	0x10	RW	Bit[7:0]: Debug mode
0x401C	BLC CTRL1C	0x00	RW	Bit[7:0]: Debug mode
0x401D	BLC CTRL1D	0x40	RW	Bit[7:0]: Debug mode
0x401E	BLC CTRL1E	0x20	RW	Bit[7:0]: Cut range down threshold

table 6-14 BLC control registers (sheet 4 of 8)

address	register name	default value	R/W	description
0x401F	BLC CTRL1F	0x06	RW	Bit[7:4]: Reserved Bit[3]: Rblue BLC reverse Bit[2]: Interpolation X enable Bit[1]: Interpolation Y enable Bit[0]: Anchor one enable
0x4020	ANCHOR LEFT START	0x02	RW	Bit[7:4]: Reserved Bit[3:0]: Anchor left start[11:8]
0x4021	ANCHOR LEFT START	0x40	RW	Bit[7:0]: Anchor left start[7:0]
0x4022	ANCHOR LEFT END	0x03	RW	Bit[7:4]: Reserved Bit[3:0]: Anchor left end[11:8]
0x4023	ANCHOR LEFT END	0x3F	RW	Bit[7:0]: Anchor left end[7:0]
0x4024	ANCHOR RIGHT START	0x07	RW	Bit[7:4]: Reserved Bit[3:0]: Anchor right start[11:8]
0x4025	ANCHOR RIGHT START	0xC0	RW	Bit[7:0]: Anchor right start[7:0]
0x4026	ANCHOR RIGHT END	0x08	RW	Bit[7:4]: Reserved Bit[3:0]: Anchor right end[11:8]
0x4027	ANCHOR RIGHT END	0xBF	RW	Bit[7:0]: Anchor right end[7:0]
0x4028	TOP ZLINE ST	0x00	RW	Bit[7:6]: Not used Bit[5:0]: Top zero line start
0x4029	TOP ZLINE NUM	0x02	RW	Bit[7:5]: Not used Bit[4:0]: Top zero line number
0x402A	TOP BLKLINE ST	0x06	RW	Bit[7:6]: Not used Bit[5:0]: Top black line start
0x402B	TOP BLKLINE NUM	0x04	RW	Bit[7:5]: Not used Bit[4:0]: Top black line number
0x402C	BOT ZLINE ST	0x02	RW	Bit[7:6]: Not used Bit[5:0]: Bottom zero line start
0x402D	BOT ZLINE NUM	0x02	RW	Bit[7:5]: Not used Bit[4:0]: Bottom zero line number
0x402E	BOT BLKLINE ST	0x0E	RW	Bit[7:6]: Not used Bit[5:0]: Bottom black line start
0x402F	BOT BLKLINE NUM	0x04	RW	Bit[7:5]: Not used Bit[4:0]: Bottom black line number
0x4030	DCBLC K1	0x01	RW	Bit[7:4]: Reserved Bit[3:0]: Dark current BLC top K coefficient[11:8]

table 6-14 BLC control registers (sheet 5 of 8)

address	register name	default value	R/W	description
0x4031	DCBLC K1	0x00	RW	Bit[7:0]: Dark current BLC top K coefficient[7:0]
0x4032	DCBLC K2	0x01	RW	Bit[7:4]: Reserved Bit[3:0]: Dark current BLC bottom K coefficient[11:8]
0x4033	DCBLC K2	0x00	RW	Bit[7:0]: Dark current BLC bottom K coefficient[7:0]
0x4034	OFFSET LIMIT	0x1F	RW	Bit[7:0]: Limitation BLC offset
0x4035~ 0x403F	NOT USED	–	–	Not Used
0x4040	LINE OFFSET C00	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset c00[11:8]
0x4041	LINE OFFSET C00	–	R	Bit[7:0]: Line offset c00[7:0]
0x4042	LINE OFFSET C01	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset c01[11:8]
0x4043	LINE OFFSET C01	–	R	Bit[7:0]: Line offset c01[7:0]
0x4044	LINE OFFSET C10	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset c10[11:8]
0x4045	LINE OFFSET C10	–	R	Bit[7:0]: Line offset c10[7:0]
0x4046	LINE OFFSET C11	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset c11[11:8]
0x4047	LINE OFFSET C11	–	R	Bit[7:0]: Line offset c11[7:0]
0x4048	LINE OFFSET D00	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset d00[11:8]
0x4049	LINE OFFSET D00	–	R	Bit[7:0]: Line offset d00[7:0]
0x404A	LINE OFFSET D01	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset d01[11:8]
0x404B	LINE OFFSET D01	–	R	Bit[7:0]: Line offset d01[7:0]
0x404C	LINE OFFSET D10	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset d10[11:8]
0x404D	LINE OFFSET D10	–	R	Bit[7:0]: Line offset d10[7:0]
0x404E	LINE OFFSET D11	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset d11[11:8]
0x404F	LINE OFFSET D11	–	R	Bit[7:0]: Line offset d11[7:0]

table 6-14 BLC control registers (sheet 6 of 8)

address	register name	default value	R/W	description
0x4050	LINE OFFSET E00	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset e00[11:8]
0x4051	LINE OFFSET E00	–	R	Bit[7:0]: Line offset e00[7:0]
0x4052	LINE OFFSET E01	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset e01[11:8]
0x4053	LINE OFFSET E01	–	R	Bit[7:0]: Line offset e01[7:0]
0x4054	LINE OFFSET E10	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset e10[11:8]
0x4055	LINE OFFSET E10	–	R	Bit[7:0]: Line offset e10[7:0]
0x4056	LINE OFFSET E11	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset e11[11:8]
0x4057	LINE OFFSET E11	–	R	Bit[7:0]: Line offset e11[7:0]
0x4058	LINE OFFSET F00	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset f00[11:8]
0x4059	LINE OFFSET F00	–	R	Bit[7:0]: Line offset f00[7:0]
0x405A	LINE OFFSET F01	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset f01[11:8]
0x405B	LINE OFFSET F01	–	R	Bit[7:0]: Line offset f01[7:0]
0x405C	LINE OFFSET F10	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset f10[11:8]
0x405D	LINE OFFSET F10	–	R	Bit[7:0]: Line offset f10[7:0]
0x405E	LINE OFFSET F11	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset f11[11:8]
0x405F	LINE OFFSET F11	–	R	Bit[7:0]: Line offset f11[7:0]
0x4060	LINE OFFSET A00	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset a00[11:8]
0x4061	LINE OFFSET A00	–	R	Bit[7:0]: Line offset a00[7:0]
0x4062	LINE OFFSET A01	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset a01[11:8]
0x4063	LINE OFFSET A01	–	R	Bit[7:0]: Line offset a01[7:0]
0x4064	LINE OFFSET A10	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset a10[11:8]
0x4065	LINE OFFSET A10	–	R	Bit[7:0]: Line offset a10[7:0]
0x4066	LINE OFFSET A11	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset a11[11:8]

table 6-14 BLC control registers (sheet 7 of 8)

address	register name	default value	R/W	description
0x4067	LINE OFFSET A11	–	R	Bit[7:0]: Line offset a11[7:0]
0x4068	LINE OFFSET B00	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset b00[11:8]
0x4069	LINE OFFSET B00	–	R	Bit[7:0]: Line offset b00[7:0]
0x406A	LINE OFFSET B01	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset b01[11:8]
0x406B	LINE OFFSET B01	–	R	Bit[7:0]: Line offset b01[7:0]
0x406C	LINE OFFSET B10	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset b10[11:8]
0x406D	LINE OFFSET B10	–	R	Bit[7:0]: Line offset b10[7:0]
0x406E	LINE OFFSET B11	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset b11[11:8]
0x406F	LINE OFFSET B11	–	R	Bit[7:0]: Line offset b11[7:0]
0x4070	LINE OFFSET G00	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset g00[11:8]
0x4071	LINE OFFSET G00	–	R	Bit[7:0]: Line offset g00[7:0]
0x4072	LINE OFFSET G01	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset g01[11:8]
0x4073	LINE OFFSET G01	–	R	Bit[7:0]: Line offset g01[7:0]
0x4074	LINE OFFSET G10	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset g10[11:8]
0x4075	LINE OFFSET G10	–	R	Bit[7:0]: Line offset g10[7:0]
0x4076	LINE OFFSET G11	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset g11[11:8]
0x4077	LINE OFFSET G11	–	R	Bit[7:0]: Line offset g11[7:0]
0x4078	LINE OFFSET H00	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset h00[11:8]
0x4079	LINE OFFSET H00	–	R	Bit[7:0]: Line offset h00[7:0]
0x407A	LINE OFFSET H01	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset h01[11:8]
0x407B	LINE OFFSET H01	–	R	Bit[7:0]: Line offset h01[7:0]
0x407C	LINE OFFSET H10	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset h10[11:8]
0x407D	LINE OFFSET H10	–	R	Bit[7:0]: Line offset h10[7:0]

table 6-14 BLC control registers (sheet 8 of 8)

address	register name	default value	R/W	description
0x407E	LINE OFFSET H11	–	R	Bit[7:4]: Not used Bit[3:0]: Line offset h11[11:8]
0x407F	LINE OFFSET H11	–	R	Bit[7:0]: Line offset h11[7:0]
0x4080	LEFT COL OFFSET M	–	R	Bit[7:4]: Not used Bit[3:0]: Left col offset m[11:8]
0x4081	LEFT COL OFFSET M	–	R	Bit[7:0]: Left col offset m[7:0]
0x4082	LEFT COL OFFSET N	–	R	Bit[7:4]: Not used Bit[3:0]: Left col offset n[11:8]
0x4083	LEFT COL OFFSET N	–	R	Bit[7:0]: Left col offset n[7:0]
0x4084	LEFT COL OFFSET O	–	R	Bit[7:4]: Not used Bit[3:0]: Left col offset o[11:8]
0x4085	LEFT COL OFFSET O	–	R	Bit[7:0]: Left col offset o[7:0]
0x4086	LEFT COL OFFSET P	–	R	Bit[7:4]: Not used Bit[3:0]: Left col offset p[11:8]
0x4087	LEFT COL OFFSET P	–	R	Bit[7:0]: Left col offset p[7:0]

6.15 frame control [0x4200 - 0x4203]

table 6-15 frame control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4200	R0	0x08	RW	Bit[7:4]: Not used Bit[3]: sof_after_line0 Bit[2]: fcnt_eof_sel Bit[1]: fcnt_mask_dis Bit[0]: fcnt_reset
0x4201	R1	0x00	RW	Bit[7:4]: Not used Bit[3:0]: frame_on_number
0x4202	R2	0x00	RW	Bit[7:4]: Not used Bit[3:0]: frame_off_number 0x4201, 0x4202= x, 0: on x frames and off 0x4201, 0x4202= 0, x: off x frames and on 0x4201, 0x4202= x, y: on x frames and off y frames

table 6-15 frame control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4203	R3	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

6.16 format control [0x4300 - 0x4329]

table 6-16 format control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x4300	CLIP MAX HI	0xFF	RW	Bit[7:0]: clip_max[11:4]
0x4301	CLIP MIN HI	0x00	RW	Bit[7:0]: clip_min[11:4]
0x4302	CLIP LO	0x0F	RW	Bit[7:4]: clip_min[3:0] Bit[3:0]: clip_max[3:0]
0x4303	FORMAT CTRL3	0x00	RW	Bit[7]: r_inc_en Bit[6]: r_pat_inv Bit[5]: r_pad_lsb Bit[4]: r_bar_mux Bit[3]: r_bar_en Bit[2]: r_moto_tst_en Bit[1]: r_tst_bit8 Bit[0]: r_moto_tst_md
0x4304	FORMAT CTRL4	0x08	RW	Bit[7]: Not used Bit[6:4]: data_bit_swap 000: data_in[9:0] 001: data_in[0:9] 010: data_in[2:9],data_in[1:0] 011: data_in[7:0],data_in[9:8] 100: data_in[9:8],data_in[0:7] 101: data_in[9],data_in[0:8] 110: data_in[1:9],data_in[0] 111: data_in[8:0],data_in[9] Bit[3]: tst_full_win Bit[2:0]: bar_pad

table 6-16 format control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x4305	PAD LOW1	0x40	RW	Bit[7:6]: Pad99 Bit[5:4]: Pad66 Bit[3:2]: Pad33 Bit[1:0]: Pad00
0x4306	PAD LOW2	0x0E	RW	Bit[7:4]: Not used Bit[3:2]: Padff Bit[1:0]: Padcc
0x4307	EMBED CTRL	0x30	RW	Bit[7:2]: Not used Bit[1]: dpc_threshold_opt 0: For black pixel 1: For white pixel Bit[0]: embedded_en
0x4308	TST X START HIGH	0x00	RW	Bit[7:5]: Not used Bit[4:0]: tst_x_start[11:7]
0x4309	TST X START LOW	0x00	RW	Bit[7:1]: tst_x_start[6:0] Bit[0]: Not used
0x430A	TST Y START HIGH	0x00	RW	Bit[7:5]: Not used Bit[4:0]: tst_y_start[11:7]
0x430B	TST Y START LOW	0x00	RW	Bit[7:1]: tst_y_start[6:0] Bit[0]: Not used
0x430C	TST WIDTH HIGH	0x00	RW	Bit[7:4]: Not used Bit[3:0]: tst_width[11:8]
0x430D	TST WIDTH LOW	0x00	RW	Bit[7:0]: tst_width[7:0]
0x430E	TST HIGHT HIGH	0x00	RW	Bit[7:4]: Not used Bit[3:0]: tst_hight[11:8]
0x430F	TST HIGHT LOW	0x00	RW	Bit[7:0]: tst_hight[7:0]
0x4311	CTRL11	0x04	RW	Bit[7:0]: r_hsyvsy_neg_width[15:8]
0x4312	CTRL12	0x00	RW	Bit[7:0]: r_hsyvsy_neg_width[7:0]
0x4313	CTRL13	0x00	RW	Bit[7:5]: Not used Bit[4]: r_vsync_pol Bit[3:2]: r_vsyncout_sel Bit[1]: r_vsync3_mod Bit[0]: r_vsync2_mod
0x4314	CTRL14	0x00	RW	Bit[7:0]: r_seof_vsync_delay[23:16]
0x4315	CTRL15	0x00	RW	Bit[7:0]: r_seof_vsync_delay[15:8]
0x4316	CTRL16	0x00	RW	Bit[7:1]: r_seof_vsync_delay[7:1] Bit[0]: r_dpcm_en

table 6-16 format control registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x4320	TEST PATTERN CTRL	0x80	RW	Bit[7:6]: pixel_order 00: GR/BG 01: RG/GB 10: BG/GR 11: GB/RG Bit[5]: byte_swap Bit[4]: bit_reverse Bit[3:2]: Not used Bit[1]: solid_color_en Bit[0]: pn31_enable
0x4321	PN31 CTRL	0x00	RW	Bit[7:4]: Not used Bit[3]: PN31 LSB first enable Bit[2]: PN31 reset by sof enable Bit[1]: PN31 reset by HREF enable Bit[0]: PN9 enable
0x4322	SOLID COLOR B	0x00	RW	Bit[7:2]: Not used Bit[1:0]: solid_color_b[9:8]
0x4323	SOLID COLOR B	0x00	RW	Bit[7:0]: solid_color_b[7:0]
0x4324	SOLID COLOR GB	0x00	RW	Bit[7:2]: Not used Bit[1:0]: solid_color_gb[9:8]
0x4325	SOLID COLOR GB	0x00	RW	Bit[7:0]: solid_color_gb[7:0]
0x4326	SOLID COLOR R	0x00	RW	Bit[7:2]: Not used Bit[1:0]: solid_color_r[9:8]
0x4327	SOLID COLOR R	0x00	RW	Bit[7:0]: solid_color_r[7:0]
0x4328	SOLID COLOR GR	0x00	RW	Bit[7:2]: Not used Bit[1:0]: solid_color_gr[9:8]
0x4329	SOLID COLOR GR	0x00	RW	Bit[7:0]: solid_color_gr[7:0]

6.17 ADC sync control [0x4500 - 0x4505]

table 6-17 ADC sync control registers

address	register name	default value	R/W	description
0x4500~ 0x4505	ADC_SYNC	–	–	ADC Sync Control Registers

6.18 VFIFO control [0x4600 - 0x4604]

table 6-18 VFIFO control registers

address	register name	default value	R/W	description
0x4600	R VFIFO READ START	0x00	RW	Bit[7:0]: r_vfifo_read_start[15:8] High byte of read_start size
0x4601	R VFIFO READ START	0x10	RW	Bit[7:0]: r_vfifo_read_start[7:0] Low byte of read_start size
0x4602	R2	0x02	RW	Bit[7:4]: r_rm Bit[3]: r_test1 Bit[2]: Not used Bit[1]: Frame reset enable Bit[0]: RAM bypass enable
0x4603	R3	–	R	Bit[7:4]: Not used Bit[3]: ram_full Bit[2]: ram_empty Bit[1]: fo_full Bit[0]: fo_empty

6.19 MIPI control [0x4800 - 0x4851]

table 6-19 MIPI control registers (sheet 1 of 9)

address	register name	default value	R/W	description
0x4800	MIPI CTRL00	0x4C	RW	Bit[7]: Writing '1' to this bit will stop clock lane once at vblk Bit[6]: gate_sc_vblk_en 0: Not used 1: Enable gate clock lane only when vblanking 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]: Enable clock lane stop at hblk when in sleep mode Bit[2:0]: Not used

table 6-19 MIPI control registers (sheet 2 of 9)

address	register name	default value	R/W	description
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 0x05 1: Output 0xAA Bit[4:2]: 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
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_period manual offset SMIA Bit[2]: r_manu_half2one t_period half to 1 SMIA Bit[1]: clk_pre_half Bit[0]: hs_pre_half

table 6-19 MIPI control registers (sheet 3 of 9)

address	register name	default value	R/W	description
0x4804	MIPI CTRL04	0x04	RW	Bit[7:4]: man_lane_num Bit[3]: lane_num_manual_enable Bit[2]: lane4_6b_en 0: Not used 1: Support 4, 7, 8 lane 6-bit Bit[1]: Vsub select 0: Valid in behind 1: Valid in front Bit[0]: vfifo_8x
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
0x4806	MIPI CTRL06	0x10	RW	Bit[7]: Not used Bit[6]: Suspend latch at horizontal blanking Bit[5]: Suspend latch at vertical blanking 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 transmit 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	0x0A	RW	Bit[7:0]: wkup_dly Mark1 wakeup delay/2^10
0x4810	FCNT MAX	0xFF	RW	Bit[7:0]: fcnt_max[15:8] High byte of maximum frame counter of frame sync short packet
0x4811	FCNT MAX	0xFF	RW	Bit[7:0]: fcnt_max[7:0] Low byte of maximum frame counter of frame sync short packet
0x4813	MIPI CTRL13	0x00	RW	Bit[7:3]: Not used Bit[2]: vc_sel Bit[1:0]: VC ID

table 6-19 MIPI control registers (sheet 4 of 9)

address	register name	default value	R/W	description
0x4814	MIPI CTRL14	0x2A	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	MIPI CTRL15	0x00	RW	Bit[7]: Not used Bit[6]: pclk_inv 0: Using falling edge of mipi_pclk_o to generate MIPI bus to PHY 1: Using rising edge of mipi_pclk_o to generate MIPI bus to PHY Bit[5:0]: manu_dt_short Manual type for short packet
0x4816	EMB DT	0x53	RW	Bit[7:6]: Not used Bit[5:0]: emb_dt Manual set embedded data type
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

table 6-19 MIPI control registers (sheet 5 of 9)

address	register name	default value	R/W	description
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
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

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

address	register name	default value	R/W	description
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
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, unit UI 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, unit UI 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	0x08	RW	Bit[7:6]: Not used Bit[5:0]: r_rdy_mark
0x4837	PCLK PERIOD	0x1A	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 9)

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
0x483C	MIPI CTRL3C	0x02	RW	Bit[7:4]: Not used Bit[3:0]: t_clk_pre Unit: pclk2x cycle

table 6-19 MIPI control registers (sheet 8 of 9)

address	register name	default value	R/W	description
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	0x27	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
0x484B	SMIA OPTION	0x07	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 startsafter reset Bit[0]: sof_sel_o 0: Frame starts after HREF occurs 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

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

address	register name	default value	R/W	description
0x484D	TEST PATTEN DATA	0xB6	RW	Bit[7:0]: test_patten_data[7:0] Data lane test pattern
0x484E	FE DLY	0x10	RW	Bit[7:0]: r_fe_dly_o Last packet to frame end delay / 2
0x484F	TEST PATTEN CK DATA	0x55	RW	Bit[7:2]: Not used Bit[1:0]: clk_test_patten_reg
0x4850	LANE SEL01	0x12	RW	Bit[7]: Not used Bit[6:4]: lane1_sel Bit[3]: Not used Bit[2:0]: lane0_sel
0x4851	LANE SEL23	0x03	RW	Bit[7]: Not used Bit[6:4]: lane3_sel Bit[3]: Not used Bit[2:0]: lane2_sel

6.20 ISPFC [0x4900 - 0x4903]

table 6-20 ISPFC control registers

address	register name	default value	R/W	description
0x4900	R0	0x00	RW	Bit[7:4]: Not used Bit[3]: sof_after_line0 Bit[2]: fcnt_eof_sel Bit[1]: fcnt_mask_dis Bit[0]: fcnt_reset
0x4901	R1	0x00	RW	Bit[7:4]: Not used Bit[3:0]: frame_on_number
0x4902	R2	0x00	RW	Bit[7:4]: Not used Bit[3:0]: frame_off_number 0x4901, 0x 4902=x,0: on x frames and off 0x4901, 0x4902=0,x: off x frames and on 0x4901, 0x4902=x,y: on x frames and off y frames
0x4903	R3	0x00	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 LVDS control [0x4B00 - 0x4B0F]

table 6-21 LVDS control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4B00	LVDS R0	0xAA	RW	Bit[7]: r_bit_flip Bit[6]: Sync code manual mode enable Bit[5]: Sync code enable when only 1 lane Bit[4]: vds_pclk_inv Bit[3]: Channel ID enable in sync per lane mode Bit[2]: CCIR parameter F Bit[1]: Sav first enable Bit[0]: Sync code mode 0: Split 1: Per lane
0x4B02	LVDS DUMMY DATA0	0x00	RW	Bit[7:4]: Not used Bit[3:0]: lvds_dummy_data0[11:8] Dummy data0 high byte
0x4B03	LVDS DUMMY DATA0	0x80	RW	Bit[7:0]: lvds_dummy_data0[7:0] Dummy data0 low byte
0x4B04	LVDS DUMMY DATA1	0x00	RW	Bit[7:4]: Not used Bit[3:0]: lvds_dummy_data1[11:8] Dummy data1 high byte
0x4B05	LVDS DUMMY DATA1	0x10	RW	Bit[7:0]: lvds_dummy_data1[7:0] Dummy data1 low byte
0x4B06	LVDS R6	0xAA	RW	Bit[7:0]: frame_st frame_start sync code in manual sync code mode
0x4B07	LVDS R7	0x55	RW	Bit[7:0]: frame_ed frame_end sync code in manual sync code mode
0x4B08	LVDS R8	0x99	RW	Bit[7:0]: line_st line_start sync code in manual sync code mode
0x4B09	LVDS R9	0x66	RW	Bit[7:0]: line_ed line_end sync code in manual sync code mode
0x4B0A	LVDS RA	0x80	RW	Bit[7:4]: r_rdy_start Start point of check pnt mode for LVDS Bit[3]: r_blk_man Bit[2]: r_hts_man_en Bit[1]: r_ln2_sel Bit[0]: r_chk_pcnt

table 6-21 LVDS control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4B0C	LVDS BLK TIMES	0x00	RW	Bit[7:4]: Not used Bit[3:0]: lvds_blk_times[11:8] High byte of r_blk_times
0x4B0D	LVDS BLK TIMES	0x02	RW	Bit[7:0]: lvds_blk_times[7:0] Low byte of r_blk_times
0x4B0E	LVDS HTS MAN	0x00	RW	Bit[7:0]: lvds_hts_man[15:8] High byte of hts_man
0x4B0F	LVDS HTS MAN	0x00	RW	Bit[7:0]: lvds_hts_man[7:0] Low byte hts_man

6.22 temperature sensor [0x4D00 - 0x4D0F]

table 6-22 temperature sensor registers

address	register name	default value	R/W	description
0x4D00~ 0x4D0F	TPM_CTRL_REG	–	–	Temperature Sensor Control Registers
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 the latched temperature value
0x4D13	TPM_CTRL_13	–	R	Latched Temperature Value, Integer Part There is 64 degree offset for the real temperature, for example: 0x4D13=0x50 means 16C
0x4D14	TPM_CTRL_14	–	R	Latched Temperature Value, Decimal Part

6.23 DSP control [0x5000 - 0x5063]

table 6-23 DSP control registers (sheet 1 of 5)

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0x96	RW	Bit[7]: Lens correction (LENC) function enable Bit[6:5]: Not used Bit[4]: Manual white balance (MWB) function enable Bit[3]: Not used Bit[2]: Black DPC function enable Bit[1]: White DPC function enable Bit[0]: Not used
0x5001	ISP CTRL01	0x01	RW	Bit[7:1]: Not used Bit[0]: BLC function enable
0x5002	ISP CTRL02	0x08	RW	Bit[7:4]: Not used Bit[3]: Debug mode Bit[2]: VarioPixel function enable Bit[1]: Not used Bit[0]: Latch enable (using VSYNC to latch ISP module's enable signals)
0x5003	ISP CTRL03	0x20	RW	Bit[7]: Not used Bit[6]: DSP bypass mode Bit[5]: DPC_DBC buffer control enable Bit[4:0]: Not used
0x5004	ISP CTRL04	0x0C	RW	Bit[7:4]: Not used Bit[3]: Auto mode of ISP input size (for manual input size, see registers 0x5006~0x5009) Bit[2]: Not used Bit[1]: Gfirst reverse for the pixel order of ISP input Bit[0]: Rblue reverse for the pixel order of ISP input
0x5005	ISP CTRL05	0x10	RW	Bit[7:5]: Not used Bit[4]: Enable MWB bias (subtract BLC target before MWB gain, and add it back after MWB gain) Bit[3:0]: Post binning option (reverse the mirror / flip / pixel_order signals to post binning module)
0x5006	ISP CTRL06	0x0C	RW	Bit[7:4]: Reserved Bit[3:0]: ISP input width[11:8]
0x5007	ISP CTRL07	0xE0	RW	Bit[7:0]: ISP input width[7:0]

table 6-23 DSP control registers (sheet 2 of 5)

address	register name	default value	R/W	description
0x5008	ISP CTRL08	0x09	RW	Bit[7:4]: Reserved Bit[3:0]: ISP input height[11:8]
0x5009	ISP CTRL09	0xB0	RW	Bit[7:0]: ISP input height[7:0]
0x500A~ 0x500D	NOT USED	–	–	Not Used
0x500E	ISP CTRL0E	0x0C	RW	Bit[7:4]: Reserved Bit[3:0]: DPC input width[11:8]
0x500F	ISP CTRL0F	0xE0	RW	Bit[7:0]: DPC input width[7:0]
0x5010	ISP CTRL10	0x09	RW	Bit[7:4]: Reserved Bit[3:0]: DPC input height[11:8]
0x5011	ISP CTRL11	0xB0	RW	Bit[7:0]: DPC input height[7:0]
0x5012~ 0x5017	NOT USED	–	–	Not Used
0x5018	ISP CTRL18	0x10	RW	Bit[7:0]: Red MWB gain[13:6]
0x5019	ISP CTRL19	0x00	RW	Bit[7:6]: Reserved Bit[5:0]: Red MWB gain[5:0]
0x501A	ISP CTRL1A	0x10	RW	Bit[7:0]: Green MWB gain[13:6]
0x501B	ISP CTRL1B	0x00	RW	Bit[7:6]: Reserved Bit[5:0]: Green MWB gain[5:0]
0x501C	ISP CTRL1C	0x10	RW	Bit[7:0]: Blue MWB gain[13:6]
0x501D	ISP CTRL1D	0x00	RW	Bit[7:6]: Reserved Bit[5:0]: Blue MWB gain[5:0]
0x501E	ISP CTRL1E	0x00	RW	Bit[7:2]: Reserved Bit[1]: Digital gain function enable (MWB gain will be disabled automatically, because they are shared) Bit[0]: Same MWB gain enable (copy red gain setting to green and blue channel)
0x501F	ISP CTRL1F	0x00	RW	Bit[7:6]: Reserved Bit[5]: Bypass DSP enable Bit[4]: Bit shift enable when disable bypass DSP Bit[3]: Bit shift direction 0: Left shift 1: Right shift Bit[2:0]: Bit shift number
0x5020~ 0x5024	NOT USED	–	–	Not Used

table 6-23 DSP control registers (sheet 3 of 5)

address	register name	default value	R/W	description
0x5025	ISP CTRL25	0x00	RW	Bit[7:2]: Not used Bit[1:0]: Average select 00: Use the data after pre_ISP to calculate average 01: Use the data after DPC to calculate average 10: Use the data after MWB to calculate average 11: Use the data after post binning to calculate average
0x5026~ 0x5029	RSVD	–	–	Reserved
0x502A	ISP CTRL2A	0x00	RW	Bit[7:6]: Reserved Bit[5]: Append 0/1 for bit select 0: Append 0 when left shift 1: Append 1 when left shift Bit[4:2]: Debug mode Bit[1]: Manual mode of DPC input size (for manual input size, see registers 0x500E~0x5011) Bit[0]: Reserved
0x502B~ 0x503B	NOT USED	–	–	Not Used
0x503D	ISP CTRL3D	0x18	RW	Bit[7:5]: Reserved Bit[4]: Window cut using the output size from pre_DSP Bit[3]: ISP raw enable (for pre_DSP to adjust window cut module's Y offset) Bit[2:0]: Adjust value of window cut module's Y offset
0x503E	ISP CTRL3E	0x00	RW	Bit[7]: Adjust enable for auto ISP input width 0: Plus adjust value 1: Minus adjust value Bit[6:0]: Adjust value for auto ISP input width
0x503F	ISP CTRL3F	0x00	RW	Bit[7]: Adjust enable for auto ISP input height 0: Plus adjust value 1: Minus adjust value Bit[6:0]: Adjust value for auto ISP input height
0x5040	NOT USED	–	–	Not Used
0x5041	ISP CTRL41	0x14	RW	Bit[7:5]: Reserved Bit[4]: Post binning function enable Bit[3]: Reserved Bit[2]: Average function enable Bit[1:0]: Reserved

table 6-23 DSP control registers (sheet 4 of 5)

address	register name	default value	R/W	description
0x5042	NOT USED	–	–	Not Used
0x5043	ISP CTRL43	0x08	RW	Bit[7:5]: Reserved Bit[4:3]: Subtract offset for average module's height Bit[2]: Horizontal post binning enable Bit[1]: Vertical post binning enable Bit[0]: Manual mode of post binning enable
0x5044	ISP CTRL44	0x10	RW	Bit[7]: Manual mode of input sensor Y offset (for manual value, see registers 0x504C and 0x504D) Bit[6]: Manual mode of input sensor X offset (for manual value, see registers 0x504A and 0x504B) Bit[5]: Reserved Bit[4]: Auto dummy line enable Bit[3:2]: Reserved Bit[1:0]: ISP EOF select 00: Auto mode of EOF 01: Last HREF fall from window cut module 10: EOF from timing control module 11: EOF from window cut module
0x5045	ISP CTRL45	0x01	RW	Bit[7:4]: Reserved Bit[3]: Not used Bit[2]: Manual BLC target enable Bit[1:0]: AEC trigger select 00: EOF from timing control 01: Average done signal from average 10: Last HREF fall from window cut 11: EOF from timing control
0x5046	ISP CTRL46	0x0A	RW	Bit[7:6]: ISP SOF select Bit[5]: Manual enable of post binning's input pixel order Bit[4]: Manual enable of average's input pixel order Bit[3:2]: Manual value of post binning's input pixel order Bit[1:0]: Manual value of average's input pixel order
0x5047	ISP CTRL47	0x02	RW	Bit[7:3]: Reserved Bit[2]: Manual enable of pre_ISP's input pixel order Bit[1:0]: Manual value of pre_ISP's input pixel order
0x5048	ISP CTRL48	0x10	RW	Bit[7:0]: Manual value of BLC target

table 6-23 DSP control registers (sheet 5 of 5)

address	register name	default value	R/W	description
0x5049	ISP CTRL49	0x00	RW	Bit[7]: DPC data switch 0: Not switch 1: Switch even and odd channel of DPC's input data Bit[6]: Manual enable of DPC's input pixel order Bit[5:4]: Manual value of DPC's input pixel order Bit[3:2]: Reserved Bit[1]: Manual enable of bin_mode in post binning module Bit[0]: Manual value of bin_mode in post binning module
0x504A	ISP CTRL4A	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: Manual input sensor X offset[11:8]
0x504B	ISP CTRL4B	0x00	RW	Bit[7:0]: Manual input sensor X offset[7:0]
0x504C	ISP CTRL4C	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: Manual input sensor Y offset[11:8]
0x504D	ISP CTRL4D	0x00	RW	Bit[7:0]: Manual input sensor Y offset[7:0]
0x504E~ 0x505F	NOT USED	–	–	Not Used
0x5060	ISP CTRL60	0x00	RW	Bit[7]: Reserved Bit[6:4]: Average done mask signal Bit[3:0]: Reserved
0x5062	ISP CTRL62	0x00	RW	Bit[7:0]: Adjust sensor X offset for OTP DPC module
0x5063	ISP CTRL63	0x00	RW	Bit[7:0]: Adjust sensor Y offset for OTP DPC module

6.24 AVG control [0x5680 - 0x568A]

table 6-24 AVG control registers

address	register name	default value	R/W	description
0x5680	AVG CTRL00	0x00	RW	Bit[7:5]: Not used Bit[4:0]: x_start_avg[12:8] AVG sub-window horizontal start position high byte
0x5681	AVG CTRL01	0x00	RW	Bit[7:0]: x_start_avg[7:0] AVG sub-window horizontal start position low byte
0x5682	AVG CTRL02	0x00	RW	Bit[7:4]: Not used Bit[3:0]: y_start_avg[11:8] AVG sub-window vertical start position high byte
0x5683	AVG CTRL03	0x00	RW	Bit[7:0]: y_start_avg[7:0] AVG sub-window vertical start position low byte
0x5684	AVG CTRL04	0x0C	RW	Bit[7:5]: Not used Bit[4:0]: window_width_avg[12:8] Sub-window width high byte
0x5685	AVG CTRL05	0xC0	RW	Bit[7:0]: window_width_avg[7:0] Sub-window width low byte
0x5686	AVG CTRL06	0x09	RW	Bit[7:4]: Not used Bit[3:0]: window_height_avg[11:8] Sub-window height high byte
0x5687	AVG CTRL07	0x90	RW	Bit[7:0]: window_height_avg[7:0] Sub-window height low byte
0x5688	AVG CTRL08	0x02	RW	Bit[7:2]: Not used Bit[1]: Sum option 0: Sum = (4×B+9×G×2+10×R)/8 1: Sum = B+G×2+R Bit[0]: Manual mode of sub-window function
0x5689	AVG RO09	–	R	Bit[7:1]: Not used Bit[0]: Average calculated indicating signal for SCCB read
0x568A	AVG RO0A	–	R	Bit[7:0]: High 8 bits of whole image's average output

6.25 DPC control [0x5780 - 0x57A7]

table 6-25 DPC control registers

address	register name	default value	R/W	description
0x5780~ 0x57A7	DPC CTRL	–	–	DPC Control Registers

6.26 LENC control [0x5800 - 0x5860]

table 6-26 LENC control registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x5800	LENC G00	0x10	RW	Bit[7:6]: Not used Bit[5:0]: Control point G00 for luminance compensation
0x5801	LENC G01	0x10	RW	Bit[7:6]: Not used Bit[5:0]: Control point G01 for luminance compensation
0x5802	LENC G02	0x10	RW	Bit[7:6]: Not used Bit[5:0]: Control point G02 for luminance compensation
0x5803	LENC G03	0x10	RW	Bit[7:6]: Not used Bit[5:0]: Control point G03 for luminance compensation
0x5804	LENC G04	0x10	RW	Bit[7:6]: Not used Bit[5:0]: Control point G04 for luminance compensation
0x5805	LENC G05	0x10	RW	Bit[7:6]: Not used Bit[5:0]: Control point G05 for luminance compensation
0x5806	LENC G10	0x10	RW	Bit[7:6]: Not used Bit[5:0]: Control point G10 for luminance compensation
0x5807	LENC G11	0x08	RW	Bit[7:6]: Not used Bit[5:0]: Control point G11 for luminance compensation

table 6-26 LENC control registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x5808	LENC G12	0x08	RW	Bit[7:6]: Not used Bit[5:0]: Control point G12 for luminance compensation
0x5809~ 0x5822	LENC G13~ LENC G54	–	RW	Bit[7:6]: Not used Bit[5:0]: Control point G13~G54 for luminance compensation
0x5823	LENC G55	0x10	RW	Bit[7:6]: Not used Bit[5:0]: Control point G55 for luminance compensation
0x5824	LENC BR00	0xAA	RW	Bit[7:4]: Control point B00 for blue channel compensation Bit[3:0]: Control point R00 for red channel compensation
0x5825	LENC BR01	0xAA	RW	Bit[7:4]: Control point B01 for blue channel compensation Bit[3:0]: Control point R01 for red channel compensation
0x5826	LENC BR02	0xAA	RW	Bit[7:4]: Control point B02 for blue channel compensation Bit[3:0]: Control point R02 for red channel compensation
0x5827	LENC BR03	0xAA	RW	Bit[7:4]: Control point B03 for blue channel compensation Bit[3:0]: Control point R03 for red channel compensation
0x5828	LENC BR04	0xAA	RW	Bit[7:4]: Control point B04 for blue channel compensation Bit[3:0]: Control point R04 for red channel compensation
0x5829~ 0x583C	LENC BR10~ LENC BR44	0xAA	RW	Bit[7:4]: Control point B10~B44 for blue channels compensation Bit[3:0]: Control point R10~R44 for red channels compensation
0x583D	LENC BROFFSET	0x88	RW	Bit[7:4]: Base value for all blue channel control points Bit[3:0]: Base value for all red channel control points
0x583E	LENC MAXGAIN	0x40	RW	Bit[7:0]: If AutoLensSwitchEnable is true and sensor gain is larger than this threshold, luminance compensation amplitude will be the minimum value (min LENC gain). Register value is 16 times sensor gain

table 6-26 LENC control registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x583F	LENC MINGAIN	0x20	RW	Bit[7:0]: If AutoLensSwitchEnable is true and sensor gain is larger than this threshold, luminance compensation amplitude will start to decrease; otherwise, the amplitude will not change. Register value is 16 times sensor gain
0x5840	LENC MINQ	0x18	RW	Bit[7]: Not used Bit[6:0]: This value indicates the minimum amplitude which luminance channel compensates when AutoLensSwitchEnable is true. Value should be in the range [0~64]
0x5841	LENC CTRL	0x0D	RW	Bit[7:4]: Not used Bit[3]: Add BLC target 0: Do not add BLC target after applying compensation 1: Add BLC target after applying compensation Bit[2]: Subtract BLC target 0: Do not subtract BLC target after applying compensation 1: Subtract BLC target after applying compensation Bit[1]: Reserved Bit[0]: AutoLensSwitchEnable 0: Luminance compensation amplitude does not change with sensor gain 1: Luminance compensation amplitude changes with sensor gain
0x5842	LENC BRHSCALE	0x00	RW	Bit[7:3]: Not used Bit[2:0]: BRHScale[10:8] For horizontal color gain calculation, this value indicates the step between two connected horizontal pixels, where BRHScale = 3×2^18 / image width, (In OV8865, image width = 3296, image height = 2480)
0x5843	LENC BRHSCALE	0xEE	RW	Bit[7:0]: BRHScale[7:0]
0x5844	LENC BRVSCALE	0x01	RW	Bit[7:3]: Not used Bit[2:0]: BRVScale[10:8] For vertical color gain calculation, this value indicates the step between two connected vertical pixels, where BRVScale = 3×2^18 / image height

table 6-26 LENC control registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x5845	LENC BRVSCALE	0x3D	RW	Bit[7:0]: BRVScale[7:0]
0x5846	LENC GHSCALE	0x01	RW	Bit[7:3]: Not used Bit[2:0]: GHScale[10:8] For horizontal luminance gain calculation, this value indicates the step between two connected horizontal pixels, where GHScale = 4×2^18 / image width
0x5847	LENC GHSCALE	0x3E	RW	Bit[7:0]: GHScale[7:0]
0x5848	LENC GVSCALE	0x00	RW	Bit[7:3]: Not used Bit[2:0]: GVScale[10:8] For vertical luminance gain calculation, this value indicates the step between two connected horizontal pixels, where GVScale = 4×2^17 / image height
0x5849	LENC GVSCALE	0xD3	RW	Bit[7:0]: GVScale[7:0]
0x584A~ 0x584F	NOT USED	–	–	Not Used
0x5850	LENC XOFFSET	–	R	Bit[7:4]: Not used Bit[3:0]: Input sensor horizontal offset[11:8]
0x5851	LENC XOFFSET	–	R	Bit[7:0]: Input sensor horizontal offset[7:0]
0x5852	LENC YOFFSET	–	R	Bit[7:4]: Not used Bit[3:0]: Input sensor vertical offset[11:8]
0x5853	LENC YOFFSET	–	R	Bit[7:0]: Input sensor vertical offset[7:0]
0x5854	LENC INPUT	–	R	Bit[7:6]: Not used Bit[5]: Input sensor flip Bit[4]: Input sensor mirror Bit[3:2]: Input sensor Y skip Bit[1:0]: Input sensor X skip
0x5855	LENC OVERFLOW	–	R	Bit[7:4]: Not used Bit[3]: GH overflow for debug Bit[2]: BRH overflow for debug Bit[1]: GV overflow for debug Bit[0]: BRV overflow for debug
0x5856	LENC QVALUE	–	R	Bit[7]: Not used Bit[6:0]: Real amplitude Q value

6.27 WINC [0x5A00 - 0x5A0C]

table 6-27 WINC control registers

address	register name	default value	R/W	description
0x5A00	WINC CTRL00	0x00	RW	Bit[7:4]: Not used Bit[3:0]: x_start_offset[11:8] Start address in horizontal
0x5A01	WINC CTRL01	0x00	RW	Bit[7:0]: x_start_offset[7:0]
0x5A02	WINC CTRL02	0x00	RW	Bit[7:4]: Not used Bit[3:0]: y_start_offset[11:8] Start address in vertical
0x5A03	WINC CTRL03	0x00	RW	Bit[7:0]: y_start_offset[7:0]
0x5A04	WINC CTRL04	0x0C	RW	Bit[7:4]: Not used Bit[3:0]: window_width[11:8] Select whole zone width high byte
0x5A05	WINC CTRL05	0xE0	RW	Bit[7:0]: window_width[7:0] Select whole zone width low byte
0x5A06	WINC CTRL06	0x09	RW	Bit[7:4]: Not used Bit[3:0]: window_height[11:8] Select whole zone height high byte
0x5A07	WINC CTRL07	0xB0	RW	Bit[7:0]: window_height[7:0] Select whole zone height low byte
0x5A08	WINC CTRL08	0x06	RW	Bit[7:3]: Reserved Bit[2]: Select embed line flag 0: Select first line as embedded flag 1: Select last line as embedded flag Bit[1]: Window enable option 0: Disable after last valid line 1: Original enable signal from register Bit[0]: Manual window enable 0: Window size from window top 1: Window size from 0x5A00 to 0x5A07
0x5A09	WINC RO09	–	R	Bit[7:4]: Not used Bit[3:0]: Pixel count[11:8] for debug
0x5A0A	WINC RO0A	–	R	Bit[7:0]: Pixel count[7:0] for debug
0x5A0B	WINC RO0B	–	R	Bit[7:4]: Not used Bit[3:0]: Line count[11:8] for debug
0x5A0C	WINC RO0C	–	R	Bit[7:0]: Line count[7:0] for debug

6.28 OTP DPC control [0x5B00 - 0x5B23]

table 6-28 OTP DPC registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5B00	OTP CTRL00	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Memory start address[11:8]
0x5B01	OTP CTRL01	0x00	RW	Bit[7:0]: Memory start address[7:0]
0x5B02	OTP CTRL02	0x01	RW	Bit[7:4]: Not used Bit[3:0]: Memory end address[11:8]
0x5B03	OTP CTRL03	0xFF	RW	Bit[7:0]: Memory end address[7:0]
0x5B04	OTP CTRL04	0x02	RW	Bit[7:5]: Reserved Bit[4]: Manual increase step enable Bit[3]: Disable mirror and flip Bit[2]: Disable OTP offset Bit[1]: Mirror option enable Bit[0]: Disable binning mode
0x5B05	OTP CTRL05	0x6C	RW	Bit[7]: Not used Bit[6:5]: Recover method select 00: Left 1 neighbor pixel (on same channel) 01: Minimum of left 2 neighbor pixels 10: Average of left and right 1 neighbor pixel 11: Maximum between the minimum of left 2 neighbor pixels and the minimum of right 2 neighbor pixels Bit[4]: Use fixed pattern to recover cluster Bit[3]: Fixed pattern mode 0: Use 0x00 to recover cluster 1: Use 0x3FF to recover cluster Bit[2]: Flip option enable Bit[1]: Sensor exposure constrain enable Bit[0]: Sensor gain constrain enable
0x5B06	OTP CTRL06	0x00	RW	Bit[7]: Not used Bit[6:5]: Constrain exposure threshold[9:8] Bit[4:0]: Not used
0x5B07	OTP CTRL07	0x00	RW	Bit[7:0]: Constrain exposure threshold[7:0] (disable OTP function when the sensor exposure is smaller than the constrain exposure threshold)

table 6-28 OTP DPC registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x5B08	OTP CTRL08	0x07	RW	Bit[7:6]: Not used Bit[5:0]: Constrain gain threshold (disable OTP function when the sensor gain is smaller than the constrain gain threshold)
0x5B09	OTP CTRL09	0x08	RW	Bit[7:4]: Not used Bit[3:0]: Recover threshold (recover when the high 8-bits of the recovered data is bigger than the original one by this threshold)
0x5B0A	OTP CTRL0A	0x01	RW	Bit[7:5]: Not used Bit[4:0]: Manual horizontal even increase step
0x5B0B	OTP CTRL0B	0x01	RW	Bit[7:5]: Not used Bit[4:0]: Manual horizontal odd increase step
0x5B0C	OTP CTRL0C	0x01	RW	Bit[7:5]: Not used Bit[4:0]: Manual vertical even increase step
0x5B0D	OTP CTRL0D	0x01	RW	Bit[7:5]: Not used Bit[4:0]: Manual vertical odd increase step
0x5B10	OTP RO10	–	R	Bit[7:4]: Not used Bit[3:0]: Horizontal offset[11:8]
0x5B11	OTP RO11	–	R	Bit[7:0]: Horizontal offset[7:0]
0x5B12	OTP RO12	–	R	Bit[7:4]: Not used Bit[3:0]: Vertical offset[11:8]
0x5B13	OTP RO13	–	R	Bit[7:0]: Vertical offset[7:0]
0x5B14	OTP RO14	–	R	Bit[7:5]: Not used Bit[4:0]: Horizontal even increase step
0x5B15	OTP RO15	–	R	Bit[7:5]: Not used Bit[4:0]: Horizontal odd increase step
0x5B16	OTP RO16	–	R	Bit[7:5]: Not used Bit[4:0]: Vertical even increase step
0x5B17	OTP RO17	–	R	Bit[7:5]: Not used Bit[4:0]: Vertical odd increase step
0x5B18~ 0x5B1F	NOT USED	–	–	Not Used
0x5B20	OTP CTRL20	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Manual X offset[11:8]
0x5B21	OTP CTRL21	0x00	RW	Bit[7:0]: Manual X offset[7:0]

table 6-28 OTP DPC registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x5B22	OTP CTRL22	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Manual Y offset[11:8]
0x5B23	OTP CTRL23	0x00	RW	Bit[7:0]: Manual Y offset[7:0]

6.29 pre_DSP control [0x5E00 - 0x5E2E]

table 6-29 pre_DSP control registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x5E00	PRE CTRL00	0x00	RW	Bit[7]: Test pattern enable Bit[6]: Rolling bar function enable Bit[5]: Transparent enable Bit[4]: Square mode 0: Color square 1: Black-white square Bit[3:2]: Color bar style 00: Standard color bar 01: Top-bottom darker color bar 10: Right-left darker color bar 11: Bottom-top darker color bar Bit[1:0]: Test pattern mode 00: Color bar 01: Random data 10: Square 11: Black image
0x5E01	PRE CTRL01	0x41	RW	Bit[7]: Reserved Bit[6]: Window cut enable Bit[5]: two_lsb_0_en When set, two LSBs of output data are 0 Bit[4]: Same seed enable When set, the seed used to generate the random data are same which is set in seed register Bit[3:0]: Random seed Seed used in generating random data
0x5E02	PRE CTRL02	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: Line number interrupt[11:8]
0x5E03	PRE CTRL03	0x01	RW	Bit[7:0]: Line number interrupt[7:0]
0x5E04~ 0x5E07	RSVD	–	–	Reserved

table 6-29 pre_DSP control registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x5E08	PRE CTRL08	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: Horizontal manual offset[11:8]]
0x5E09	PRE CTRL09	0x00	RW	Bit[7:0]: Horizontal manual offset[7:0]
0x5E0A	PRE CTRL0A	0x00	RW	Bit[7:4]: Reserved Bit[3:0]: Vertical manual offset[11:8]
0x5E0B	PRE CTRL0B	0x00	RW	Bit[7:0]: Vertical manual offset[7:0]
0x5E0C	PRE RO0C	–	R	Bit[7:4]: Reserved Bit[3:0]: Input image pixel number[11:8]
0x5E0D	PRE RO0D	–	R	Bit[7:0]: Input image pixel number[7:0]
0x5E0E	PRE RO0E	–	R	Bit[7:4]: Reserved Bit[3:0]: Input image line number[11:8]
0x5E0F	PRE RO0F	–	R	Bit[7:0]: Input image line number[7:0]
0x5E10	PRE CTRL10	0x3C	RW	Bit[7]: Window X offset option Bit[6]: Window Y offset option Bit[5]: Take the first pixel in the same position with no mirror image enable Bit[4]: Take the first pixel in the same position with no flip image enable Bit[3]: Mirror option from window 0: First pixel is Gb or R with window output 1: First pixel is B or Gr with window output Bit[2]: Flip option from window 0: First line is GR with window output 1: First line is BG with window output Bit[1]: Offset manual enable Bit[0]: Reserved
0x5E11	PRE CTRL11	0x00	RW	Bit[7]: Manual clock/valid ratio enable Bit[6:4]: Manual dummy line number Bit[3]: Reduce HREF low length by half Bit[2:0]: Manual clock/valid ratio for dummy line
0x5E12	PRE RO12	–	R	Bit[7:0]: HREF blank length for dummy line[15:8]
0x5E13	PRE RO13	–	R	Bit[7:0]: HREF blank length for dummy line[7:0]
0x5E14	PRE RO14	–	R	Bit[7:0]: HREF length for dummy line[15:8]
0x5E15	PRE RO15	–	R	Bit[7:0]: HREF length for dummy line[7:0]
0x5E16	PRE RO16	–	R	Bit[7:5]: Reserved Bit[4]: Dummy error indicating signal Bit[3]: Reserved Bit[2:0]: Dummy line clock ratio output

table 6-29 pre_DSP control registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x5E17	PRE RO17	–	R	Bit[7:4]: Horizontal odd increase step Bit[3:0]: Vertical odd increase step
0x5E18	PRE RO18	–	R	Bit[7:4]: Reserved Bit[3:0]: Horizontal sensor offset[11:8]
0x5E19	PRE RO19	–	R	Bit[7:0]: Horizontal sensor offset[7:0]
0x5E1A	PRE RO1A	–	R	Bit[7:4]: Reserved Bit[3:0]: Vertical sensor offset[11:8]
0x5E1B	PRE RO1B	–	R	Bit[7:0]: Vertical sensor offset[7:0]
0x5E1C	PRE RO1C	–	R	Bit[7:4]: Reserved Bit[3:0]: Horizontal window offset[11:8]
0x5E1D	PRE RO1D	–	R	Bit[7:0]: Horizontal window offset[7:0]
0x5E1E	PRE RO1E	–	R	Bit[7:4]: Reserved Bit[3:0]: Vertical window offset[11:8]
0x5E1F	PRE RO1F	–	R	Bit[7:0]: Vertical window offset[7:0]
0x5E20	PRE RO20	–	R	Bit[7:5]: Reserved Bit[4:0]: Horizontal window output size[12:8]
0x5E21	PRE RO21	–	R	Bit[7:0]: Horizontal window output size[7:0]
0x5E22	PRE RO22	–	R	Bit[7:4]: Reserved Bit[3:0]: Vertical window output size[11:8]
0x5E23	PRE RO23	–	R	Bit[7:0]: Vertical window output size[7:0]
0x5E24	PRE RO24	–	R	Bit[7:6]: Reserved Bit[5:4]: Horizontal skip Bit[3:2]: Reserved Bit[1:0]: Vertical skip
0x5E25	PRE RO25	–	R	Bit[7:4]: Horizontal even increase step Bit[3:0]: Vertical even increase step
0x5E26	NOT USED	–	–	Not Used
0x5E27	PRE RO27	–	R	Bit[7:4]: Reserved Bit[3:0]: Cut top offset for bi-linear BLC[11:8]
0x5E28	PRE RO28	–	R	Bit[7:0]: Cut top offset for bi-linear BLC[7:0]
0x5E29	PRE RO29	–	R	Bit[7:4]: Reserved Bit[3:0]: Cut bottom offset for bi-linear BLC[11:8]
0x5E2A	PRE RO2A	–	R	Bit[7:0]: Cut bottom offset for bi-linear BLC[7:0]
0x5E2B	PRE CTRL2B	0x09	RW	Bit[7:4]: Reserved Bit[3:0]: Array height for bi-linear BLC[11:8]

table 6-29 pre_DSP control registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x5E2C	PRE CTRL2C	0xB0	RW	Bit[7:0]: Array height for bi-linear BLC[7:0]
0x5E2D	PRE CTRL2D	0x00	RW	Bit[7:6]: Reserved Bit[5]: Manual horizontal skip enable Bit[4:0]: Manual horizontal skip for RGBC pattern
0x5E2E	PRE CTRL2E	0x00	RW	Bit[7:6]: Reserved Bit[5]: Manual vertical skip enable Bit[4:0]: Manual vertical skip for RGBC pattern

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
	machine model	200V
all input/output voltages (with respect to ground)		-0.3V to V_{DD-IO} + 1V
I/O current on any input or output pin		± 200 mA

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)	2.6	2.8	3.0	V
V_{DD-D}	supply voltage (digital core for 4-lane MIPI up to 1000 Mbps/lane)	1.1	1.2	1.3	V
V_{DD-IO}	supply voltage (digital I/O)	1.7	1.8	3.0	V
I_{DD-A}	active (operating) current[b]		23	30	mA
I_{DD-IO}			3.3	4.5	mA
$I_{DD-CORE}$			105	125	mA
$I_{DDS-SCCB}$	standby current[c]		300	3000	µA
$I_{DDS-PWDN}$			300	3000	µA
$I_{DDS-XSHUTDOWN}$			6	30	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]	SIOC and SIOD	-0.5	0	0.54	V
V_{IH}	SIOC and SIOD	1.28	1.8	3.0	V

a. maximum active current is measured under typical supply voltage

b. operating current is measured at full size and 30 fps with all sensor functions on and DOVDD = 1.8V and EVDD tied to DVDD

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

d. based on DOVDD = 1.8V

7.4 timing characteristics

table 7-4 timing characteristics

symbol	parameter	min	typ	max	unit
oscillator and clock input					
f_{OSC}	frequency (EXTCLK)[a]	6	24	27	MHz
t_r, t_f	clock input rise/fall time			5 (10[b])	ns
	clock input duty cycle	45	50	55	%

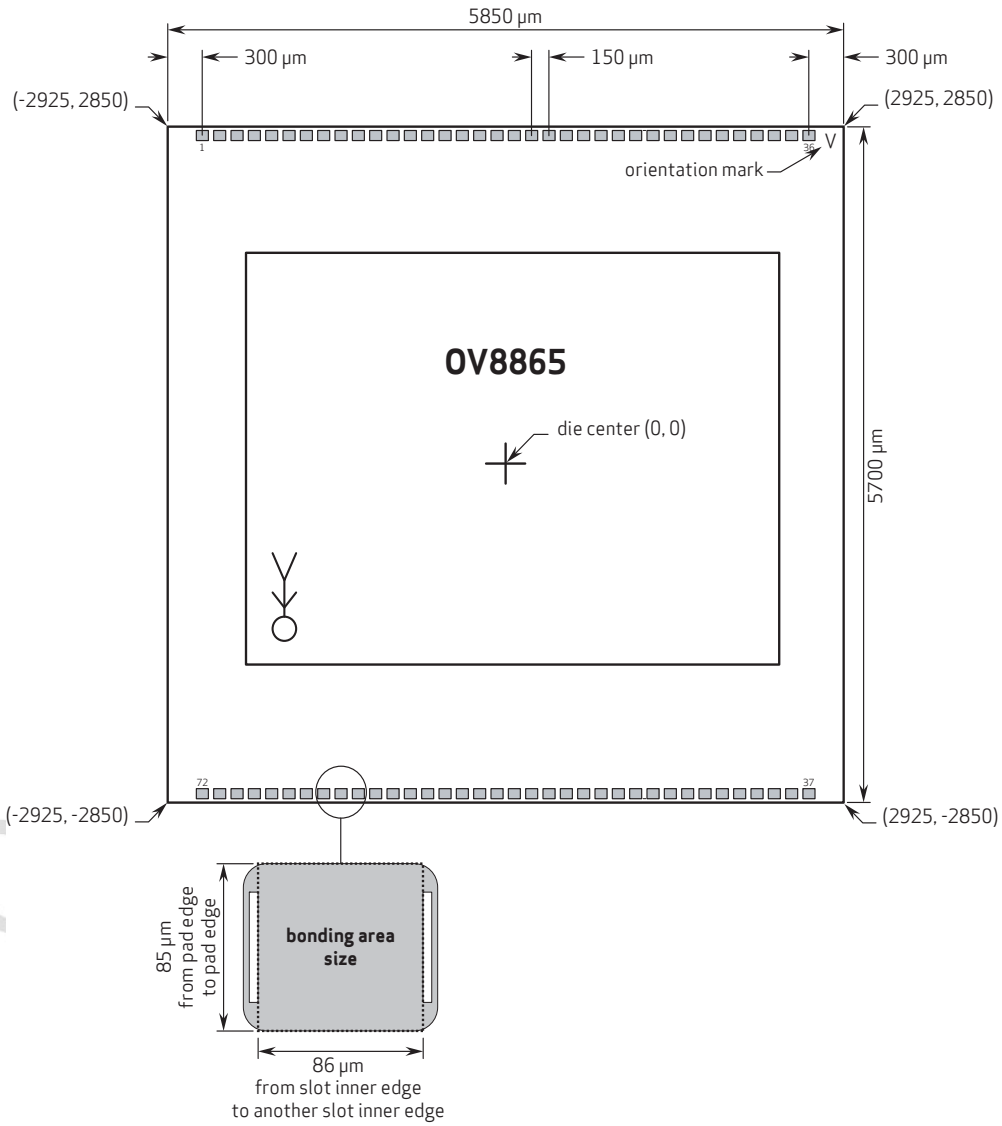
a. for input clock range 6~27MHz, the OV8865 can tolerate input clock period jitter up to 600ps peak-to-peak

b. if using internal PLL

8 mechanical specifications

8.1 COB physical specifications

figure 8-1 COB die specifications



note 1 all dimensions and coordinates are in µm unless otherwise specified

note 2 bonding outside the defined bonding area is prohibited as it may cause failure in reliability or functionality

8865_COB_DS_8_1

table 8-1 pad location coordinates (sheet 1 of 3)

pad number	pad name	x coordinate	y coordinate	bonding area size
1	AGND	-2625	2775	86x85
2	AGND	-2475	2775	86x85
3	AVDD	-2325	2775	86x85
4	AVDD	-2175	2775	86x85
5	AVDD	-2025	2775	86x85
6	AGND	-1875	2775	86x85
7	ATEST	-1725	2775	86x85
8	DOGND	-1575	2775	86x85
9	DOGND	-1425	2775	86x85
10	DVDD	-1275	2775	86x85
11	DVDD	-1125	2775	86x85
12	DOVDD	-975	2775	86x85
13	DOGND	-825	2775	86x85
14	NC	-675	2775	86x85
15	NC	-525	2775	86x85
16	DOGND	-375	2775	86x85
17	DVDD	-225	2775	86x85
18	DVDD	-75	2775	86x85
19	PWDNB	75	2775	86x85
20	DOGND	225	2775	86x85
21	DOGND	375	2775	86x85
22	TM	525	2775	86x85
23	XSHUTDOWN	675	2775	86x85
24	DVDD	825	2775	86x85
25	DVDD	975	2775	86x85
26	NC	1125	2775	86x85
27	SIOD	1275	2775	86x85
28	SIOC	1425	2775	86x85
29	SID	1575	2775	86x85
30	DOGND	1725	2775	86x85

table 8-1 pad location coordinates (sheet 2 of 3)

pad number	pad name	x coordinate	y coordinate	bonding area size
31	DOGND	1875	2775	86x85
32	DVDD	2025	2775	86x85
33	AVDD	2175	2775	86x85
34	AVDD	2325	2775	86x85
35	AGND	2475	2775	86x85
36	AGND	2625	2775	86x85
37	DVDD	2625	-2775	86x85
38	FSIN	2475	-2775	86x85
39	DTEST	2325	-2775	86x85
40	DOGND	2175	-2775	86x85
41	VSYNC	2025	-2775	86x85
42	HREF	1875	-2775	86x85
43	GPIO	1725	-2775	86x85
44	DOVDD	1575	-2775	86x85
45	STROBE	1425	-2775	86x85
46	IL_PWM	1275	-2775	86x85
47	DVDD	1125	-2775	86x85
48	EXTCLK	975	-2775	86x85
49	DOGND	825	-2775	86x85
50	MDN3	675	-2775	86x85
51	MDP3	525	-2775	86x85
52	EGND	375	-2775	86x85
53	MDN1	225	-2775	86x85
54	MDP1	75	-2775	86x85
55	MCN	-75	-2775	86x85
56	MCP	-225	-2775	86x85
57	EVDD	-375	-2775	86x85
58	EGND	-525	-2775	86x85
59	PVDD	-675	-2775	86x85
60	EGND	-825	-2775	86x85

table 8-1 pad location coordinates (sheet 3 of 3)

pad number	pad name	x coordinate	y coordinate	bonding area size
61	EVDD	-975	-2775	86x85
62	MDN0	-1125	-2775	86x85
63	MDP0	-1275	-2775	86x85
64	EGND	-1425	-2775	86x85
65	MDN2	-1575	-2775	86x85
66	MDP2	-1725	-2775	86x85
67	DOGND	-1875	-2775	86x85
68	DVDD	-2025	-2775	86x85
69	VN	-2175	-2775	86x85
70	VH	-2325	-2775	86x85
71	AGND	-2475	-2775	86x85
72	AVDD	-2625	-2775	86x85

8.2 reconstructed wafer (RW) physical specifications

- maximum total die count: 665
- film frame: Disco
- dicing tape: FSL-N6600

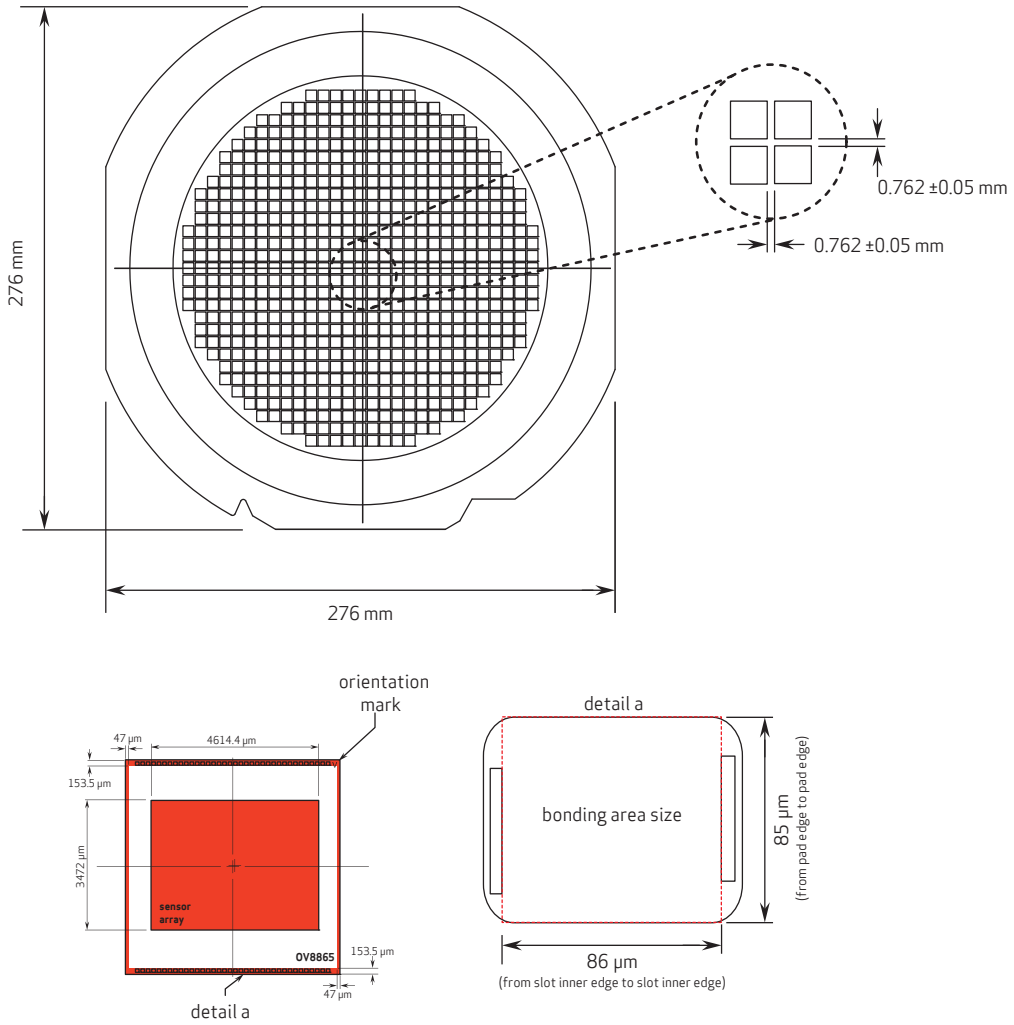
table 8-2 RW physical dimensions

feature	dimensions
RW physical dimensions	8" RW on 12" frame
wafer thickness (OVXXXXX-ABCD)	
C=4	200 µm ± 10 µm (7.9 mil ± 0.4 mil)
reconstructed wafer street width	0.762 mm (30 mil) ± 0.05 mm
placement accuracy x, y, theta	± 50 µm (± 2 mil), <1.0 degree
singulated die size	
width	5900 µm ± 20 µm (232.3 mil ± 0.8 mil)
length	5750 µm ± 20 µm (226.4 mil ± 0.8 mil)
bond pad size	104 µm × 85 µm (4.1 mil × 3.3 mil)
minimum bond pad pitch	150 µm (5.9 mil)
bonding area size	86 µm × 85 µm (3.4 mil × 3.3 mil)
optical array	
die center	(0, 0)
optical center from die center[a]	60 µm, 50 µm (2.4 mil, 2.0 mil)

a. based on die orientation on frame with notch facing down position

note
Actual die count varies and the absent die may be less than 10% of the maximum total die count (excluding the last frame of the wafer lot).

figure 8-2 OV8865 RW physical diagram



note 1 bonding outside the defined bonding area is prohibited, it may potentially induce reliablity issues or functionality failure

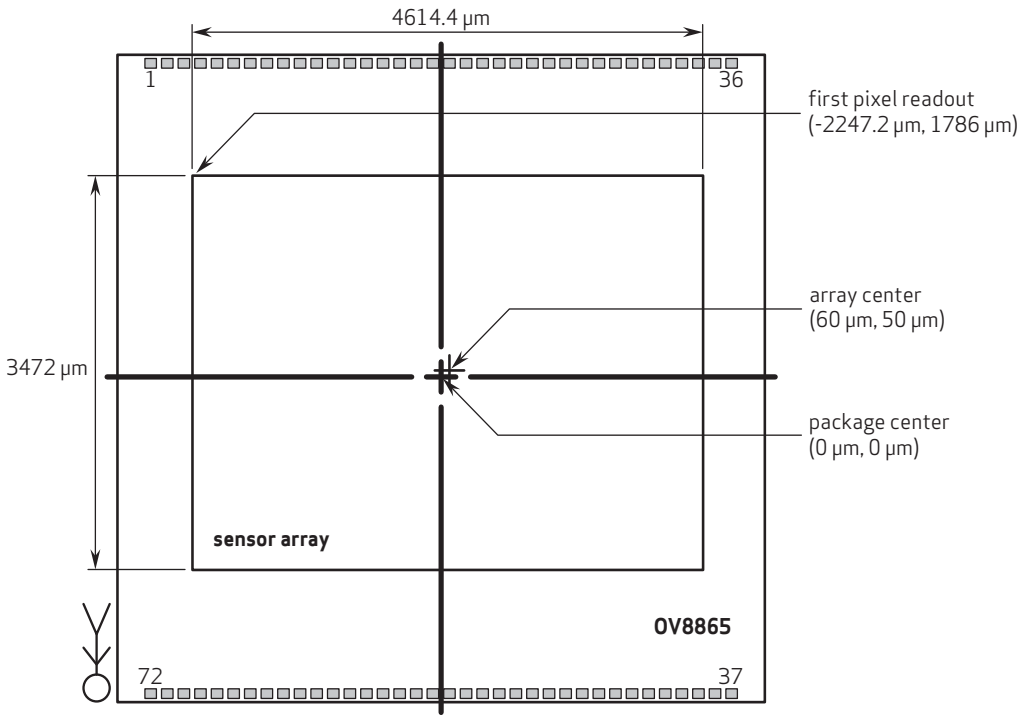
note 2 keep-out-of-contact areas are highlighted in red color for related process fixtures/tools (e.g., nozzle, collets, etc.)

8865_COB_DS_8_2

9 optical specifications

9.1 sensor array center

figure 9-1 sensor array center



top view

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 pad 1 oriented down on the PCB.

8865_DS_9_1

9.2 lens chief ray angle (CRA)

figure 9-2 chief ray angle (CRA)

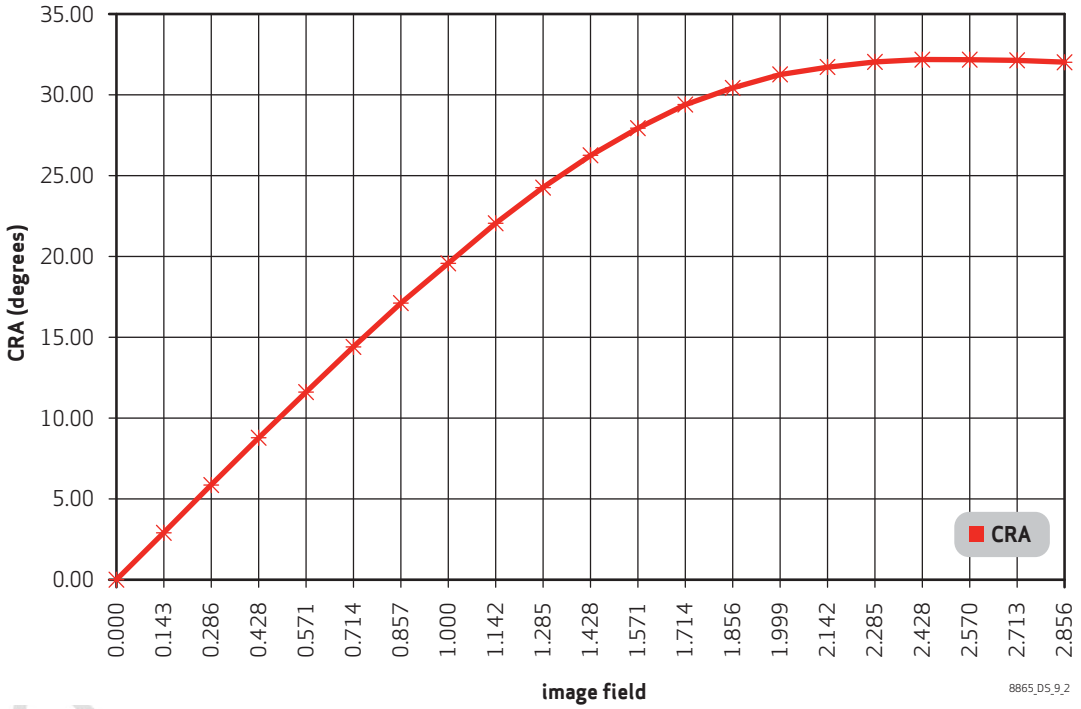


table 9-1 CRA versus image height plot (sheet 1 of 2)

field (%)	image height (mm)	CRA (degrees)
0.00	0.000	0.0
0.05	0.143	3.0
0.10	0.286	5.9
0.15	0.428	8.8
0.20	0.571	11.6
0.25	0.714	14.4
0.30	0.857	17.1
0.35	1.000	19.6
0.40	1.142	22.1
0.45	1.285	24.2

table 9-1 CRA versus image height plot (sheet 2 of 2)

field (%)	image height (mm)	CRA (degrees)
0.50	1.428	26.3
0.55	1.571	27.9
0.6	1.714	29.4
0.65	1.856	30.5
0.70	1.999	31.3
0.75	2.142	31.8
0.80	2.285	32.1
0.85	2.428	32.2
0.90	2.570	32.2
0.95	2.713	32.2
1.00	2.856	32.0

appendix A handling of RW devices

A.1 ESD /EOS prevention

1. Ensure that there is 500V ESD control in all work areas.
2. Use ESD safety shoes, ground strap, and static control smocks in test areas.
3. Use grounded work carts and tables in inspection areas.
4. OmniVision recommends the use of ionized air in all work areas.

A.2 particles and cleanliness of environment

1. All production, inspection and packaging areas should meet Class10 environment requirements.
2. Use optical microscopes with 50X and 100X magnifications for particle inspection.
3. Ensure that there is good cassette sealing for particle protection during storage.
4. OmniVision recommends water cleaning to remove removable particles.
5. RW die should be stored in nitrogen gas purged cabinets with temperature less than 30°C and relative humidity of 60% before assembly.

A.3 other requirements

1. Reliability assurance of RW or COB bare die is certified by product reliability of the bare die in a CLCC, CSP or QFP package form factor. Precautions should be taken if the packaging form factor of the bare die is other than these specified.
2. Avoid exposure to strong sunlight for extended periods of time as the color filter of the image sensor may become discolored.
3. Avoid direct exposure of the sensor bare die to high temperature and/or humidity environment as sensor characteristics will be affected. Extra precautions should be exercised if the bare die experiences temperatures exceeding 260°C for more than 75 seconds.

revision history

version 1.0 06.04.2013

- initial release

version 1.01 07.18.2013

- in table 1-1, changed description for pad 29 from "... = 0x20" to "...= 0x6C" and "... = 0x6C" to "...= 0x20"
- in section 2, updated figure 2-2

version 1.1 07.24.2013

- in key specifications, changed active power requirements to 210 mW (full resolution @ 30 fps), removed standby power requirements, and changed XSHUTDOWN power requirements to 5 µW
- in key specifications, changed max S/N ratio to 36.7 dB, dynamic range to 68.8 dB, sensitivity to 940 mV/Lux-sec, maximum exposure interval to 2480 x T_{ROW}, and dark current to 20e^-/sec @ 60°C junction temperature
- in table 6-23, changed default value of register 0x5047 to 0x02
- in table 7-3, changed typ and max values for I_{DD-A} to 28mA and 35mA, respectively
- in table 7-3, changed typ and max values for I_{DD-IO} to 3.3mA and 4.5mA, respectively
- in table 7-3, changed typ and max values for $I_{DD-CORE}$ to 105mA and 125mA, respectively
- in table 7-3, changed typ and max values for $I_{DD-SCCB}$ to 300µA and 3000µA, respectively
- in table 7-3, changed typ and max values for $I_{DD-PWDN}$ to 300µA and 3000µA, respectively
- in table 7-3, changed typ and max values for $I_{DD-XSHUTDOWN}$ to 6µA and 30µA, respectively
- in table 7-3, changed table footnote b to "operating current is measured at full size and 30 fps with all sensor functions on and DOVDD = 1.8V and EVDD tied to DVDD"
- in table 7-4, added table footnote a, "for input clock range 6~27MHz, the OV8865 can tolerate input clock period jitter up to 600ps peak-to-peak" and changed max value for clock input rise/fall time to 5 (10^b) with table footnote b, "if using internal PLL"

version 2.0 11.07.2013

- changed datasheet from Preliminary Specification to Product Specification
- in key specifications, changed active power requirements to 196 mW
- in table 7-3, changed typ and max values for active (operating) current (I_{DD-A}) to 23 and 30, respectively

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: salesamerican@ovt.com

UNITED KINGDOM

Hampshire + 44 1256 744 610

GERMANY

Munich +49 89 63 81 99 88

INDIA

Bangalore +91 988 008 0140

CHINA

Beijing + 86 10 6580 1690
Shanghai + 86 21 6175 9888
Shenzhen + 86 755 8384 9733
Hong Kong + 852 2403 4011

JAPAN

Yokohama +81 45 478 7977
Osaka +81 6 4964 2606

KOREA

Seoul + 82 2 3478 2812

SINGAPORE +65 6220 1335

TAIWAN

Taipei +886 2 2657 9800
Hsinchu +886 3 6110933

website: **www.ovt.com**