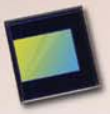
OmniVision



datasheet

PRELIMINARY SPECIFICATION

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

OV8830

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

datasheet (COB)
PRELIMINARY SPECIFICATION

version 1.41
june 2011

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

applications

- cellular and mobile phones
- digital still cameras (DSC)
- digital video camcorders (DVC)
- PC multimedia

ordering information

- **OV08830-G04A** (color, chip probing, 200 µm backgrinding, reconstructed wafer with good die)
- **OV08830-G14A** (color, chip probing, 200 µm backgrinding, uncut die, cut into four quarters)
- **OV08830-G20A** (color, chip probing, no backgrinding, no die-saw, whole wafer)
- **OV08830-G24A** (color, chip probing, 200 µm backgrinding, no die-saw, whole wafer)

features

- automatic black level calibration (ABLC)
- programmable controls for frame rate, mirror and flip, cropping, windowing, and scaling
- image quality controls: lens correction and defective pixel canceling
- supports output formats: 10-bit RAW RGB (MIPI)
- supports horizontal and vertical subsampling
- supports images sizes: 8 Mpixel, EIS1080p, 1080p, EIS720p, EISQ 1080p, Q1080p, EISVGA, VGA, QVGA, etc.
- fast mode switching
- support 2x2 binning, re-sampling filter
- standard serial SCCB interface
- up to 4-lane MIPI serial output interface
- up to 4-lane LVDS serial output interface
- embedded 4K bits one-time programmable (OTP) memory for part identification, etc.
- two on-chip phase lock loop (PLL)
- programmable I/O drive capability
- built-in 1.2V regulator for core
- built-in temperature sensor
- supports alternate row HDR timing

key specifications (typical)

- **active array size:** 3264 x 2448
- **power supply:**
  core: 1.14~1.32V for up to 700Mbps/lane or 1.27~1.32V for up to 1Gbps/lane MIPI (internal regulator optional)
  analog: 2.6 ~ 3.0V
  I/O: 1.7 ~ 3.0V
- **power requirements:**
  active: 155 mA (339mW, see **sidebar note**)
  standby: 300 µA
  XSHUTDOWN: 10 µA
- **temperature range:**
  operating: -30°C to 70°C junction temperature (see **table 8-2**)
  stable image: 0°C to 50°C junction temperature (see **table 8-2**)
- **output formats:** 10-bit RGB RAW
- **lens size:** 1/3.2"
- **input clock frequency:** 6~27 MHz
- **lens chief ray angle:** 27° non-linear (see **figure 10-2**)
- **max S/N ratio:** 36 dB
- **dynamic range:** 67 dB @ 8x gain
- **maximum image transfer rate:**
  8Mpixel: 24 fps (see **table 2-1**)
  EIS1080p: 30 fps (see **table 2-1**)
  EIS720p: 60 fps (see **table 2-1**)
- **sensitivity:** 864 mV/Lux-sec
- **scan mode:** progressive
- **maximum exposure interval:** 2480 x t_{ROW}
- **pixel size:** 1.4 µm x 1.4 µm
- **dark current:** 0.26mV/s @ 50°C junction temp
- **image area:** 4592 µm x 3450 µm
- **die dimensions:** 6410 µm x 5940 µm

note pixel performance shown are target values. These values are subject to change based on real measurements.

note
to achieve 339mW, use internal regulator, DOVDD=1.8V

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 OV8830 image sensor. The die information is shown in section 9.

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

pad number	signal name	pad type	description
01	SIOD	I/O	SCCB data I/O
02	NC	–	no connect
03	NC	–	no connect
04	FSO/VSYNC	I/O	frame sync output / VSYNC output
05	FREX	I/O	frame exposure control
06	IL_PWM	I/O	illumination control output
07	AVDD	power	power for analog circuit
08	AGND	ground	ground for analog circuit
09	STROBE	I/O	strobe control
10	PWDNB	input	power down (active low with internal pull up resistor)
11	XSHUTDOWN	input	reset and power down (active low with internal pull down resistor)
12	TM	input	test mode (active high)
13	AGND	ground	ground for analog circuit
14	DVDD	reference	power for digital circuit
15	DGND	ground	ground for I/O circuit
16	AVDD	power	power for analog circuit
17	AVDD	power	power for analog circuit
18	AGND	ground	ground for analog circuit
19	AGND	ground	ground for analog circuit
20	VN2	reference	internal analog reference (requires a 0.1 µF capacitor between VN2 and AGND)
21	VN	reference	internal analog reference (requires a 0.1 µF capacitor between VN and AGND)
22	VH	reference	internal analog reference (requires a 0.1 µF capacitor between VH and AGND)
23	AVDD	power	power for analog circuit

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

pad number	signal name	pad type	description
24	AVDD	power	power for analog circuit
25	AGND	ground	ground for analog circuit
26	AGND	ground	ground for analog circuit
27	DOVDD	power	power for I/O circuit
28	DOGND	ground	ground for I/O circuit
29	GPIO_2	I/O	general purpose I/O 2
30	NC	–	no connect
31	**EXTCLK**	input	system input clock
32	EXTREG_EN	input	external DVDD enable (active high)
33	VTST2	reference	analog test 2
34	EGND	ground	ground for MIPI circuit
35	**MDP2**	I/O	MIPI TX/LVDS data lane 2 positive output
36	**MDN2**	I/O	MIPI TX/LVDS data lane 2 negative output
37	**MDP0**	I/O	MIPI TX/LVDS data lane 0 positive output
38	**MDN0**	I/O	MIPI TX/LVDS data lane 0 negative output
39	**MCP**	I/O	MIPI TX/LVDS clock lane positive output
40	**MCN**	I/O	MIPI TX/LVDS clock lane negative output
41	EGND	ground	ground for MIPI circuit
42	EGND	ground	ground for MIPI circuit
43	EVDD	power	power for MIPI circuit (connect to DVDD)
44	EVDD	power	power for MIPI circuit (connect to DVDD)
45	**MDP1**	I/O	MIPI TX/LVDS data lane 1 positive output
46	**MDN1**	I/O	MIPI TX/LVDS data lane 1 negative output
47	**MDP3**	I/O	MIPI TX/LVDS data lane 3 positive output
48	**MDN3**	I/O	MIPI TX/LVDS data lane 3 negative output
49	PVDD	power	power for analog circuit
50	AGND	ground	ground for analog circuit
51	DOGND	ground	ground for I/O circuit
52	DOVDD	power	power for I/O circuit
53	DOVDD	power	power for I/O circuit

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

pad number	signal name	pad type	description
54	DVDD	reference	power for digital circuit
55	DVDD	reference	power for digital circuit
56	DGND	ground	ground for digital circuit
57	DGND	ground	ground for digital circuit
58	DGND	ground	ground for digital circuit
59	DGND	ground	ground for digital circuit
60	DVDD	reference	power for digital circuit
61	DVDD	reference	power for digital circuit
62	AVDD	power	power for analog circuit
63	AGND	ground	ground for analog circuit
64	AGND	ground	ground for analog circuit
65	AGND	ground	ground for analog circuit
66	AVDD	power	power for analog circuit
67	AVDD	power	power for analog circuit
68	VTST	reference	internal analog reference
69	DGND	ground	ground for digital circuit
70	DVDD	reference	power for digital circuit
71	DOVDD	power	power for I/O circuit
72	GPIO_0	I/O	general purpose I/O 0
73	GPIO_1	I/O	general purpose I/O 1
74	SID	input	SCCB address selection (0x6C if SID=1 and 0x20 if SID=0)
75	FSIN	I/O	frame sync input
76	SIOC	input	SCCB input clock

table 1-2 configuration under various conditions

pin	signal name	RESET[a]	after RESET release[b]	software standby	hardware standby[c]
01	SIOD	input	input	input	high-z
04	VSYNC	input	input by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
05	FREX	input	input by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
06	IL_PWM	input	input by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
09	STROBE	input	input by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
10	PWDNB	input	input	input	input
11	XSHUTDOWN	input	input	input	input
12	TM	input	input	input	input
31	EXTCLK	input	input	input	high-z
35	MDP2	high-z	high-z by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
36	MDN2	high-z	high-z by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
37	MDP0	high-z	high-z by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
38	MDN0	high-z	high-z by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
39	MCP	high-z	high-z by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
40	MCN	high-z	high-z by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
45	MDP1	high-z	high-z by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
46	MDN1	high-z	high-z by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
47	MDP3	high-z	high-z by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
48	MDN3	high-z	high-z by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
76	SIOC	input	input	input	high-z

a. XSHUTDOWN = 0

b. XSHUTDOWN from 0 to 1

c. PWDNB = 0

figure 1-1 pad diagram

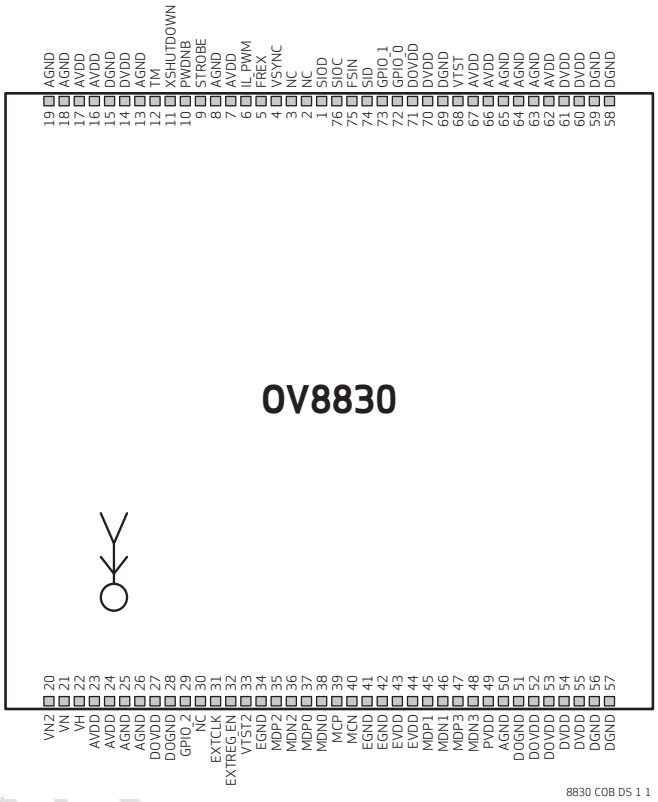


table 1-3 pad symbol and equivalent circuit (sheet 1 of 2)

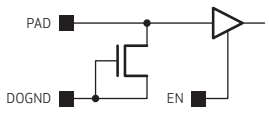
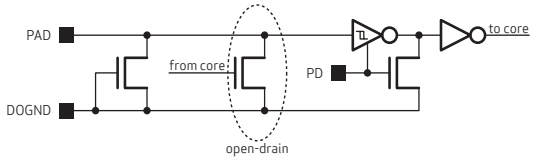
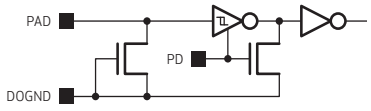
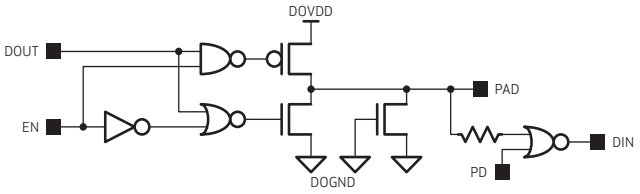
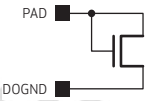
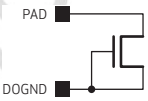
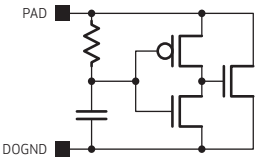
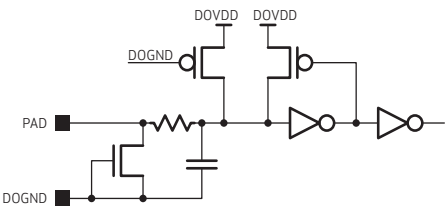
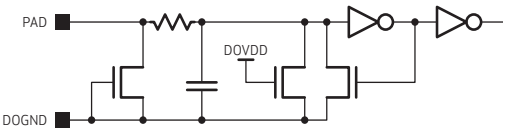
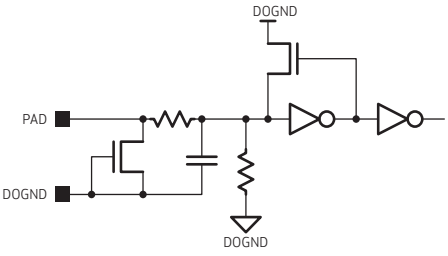
symbol	equivalent circuit
EXTCLK	
SIOD	
SIOC	

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

symbol	equivalent circuit
FSO/VSYNC, STROBE, IL_PWM, FREX, FSIN, GPIO_0, GPIO_1, GPIO_2	
VN, VN2, VH	
MDP3, MDP2, MDP1, MDP0, MDN3, MDN2, MDN1, MDN0, MCP, MCN, EGND, AGND, DOGND, DGND	
AVDD, EVDD, DVDD, DOVDD, PVDD	
PWDNB	
TM	
XSHUTDOWN, SID, EXTREG_EN	

2 system level description

2.1 overview

The OV8830 (color) image sensor is a low voltage, high performance 1/3.2-inch 8 megapixel CMOS image sensor that provides the functionality of a single 8 megapixel (3264x2448) camera using OmniBSI-2™ technology. It provides full-frame, sub-sampled, windowed, and scaled 10-bit MIPI images in various formats via the control of the Serial Camera Control Bus (SCCB) interface.

The OV8830 has an image array capable of operating at up to 24 frames per second (fps) in 10-bit 8 megapixel resolution with complete user control over image quality, formatting and output data transfer. All required image processing functions, including exposure control, white balance, defective pixel canceling, etc., are programmable through the SCCB interface.

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

For customized information purposes, the OV8830 includes one-time programmable (OTP) memory. The OV8830 has up to four lanes of MIPI interface.

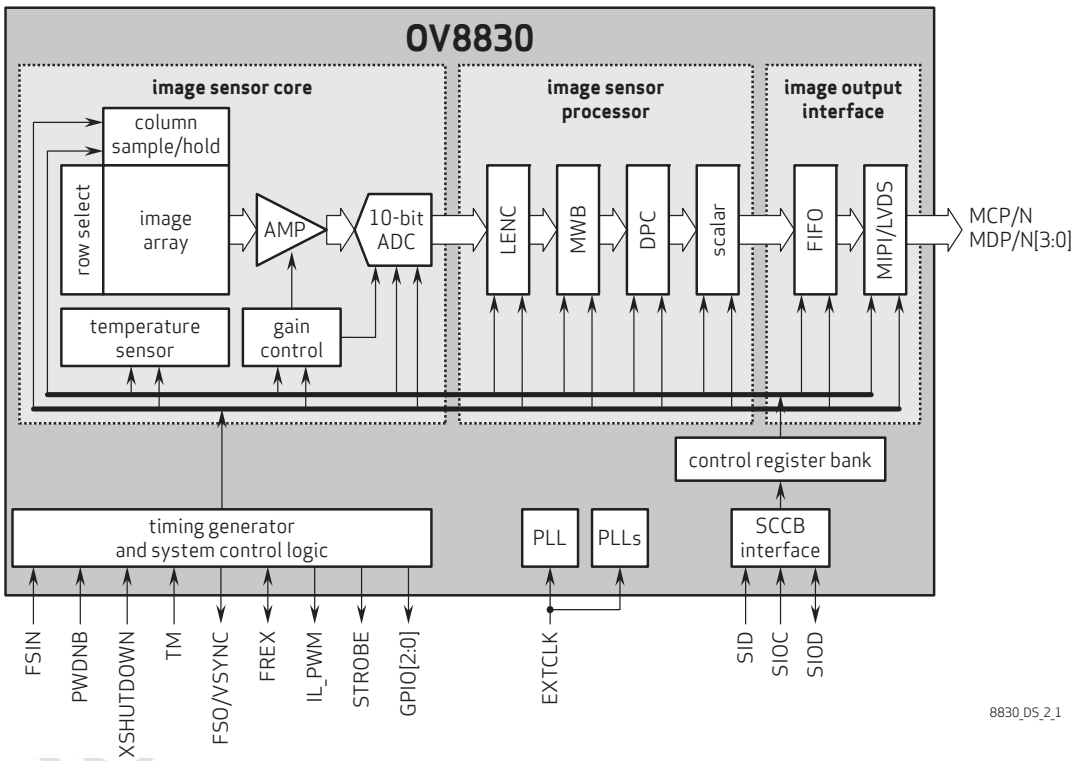
2.2 architecture

The OV8830 sensor core generates streaming pixel data at a constant frame rate. **figure 2-1** shows the functional block diagram of the OV8830 image sensor.

The timing generator outputs signals to access the rows of the image array, precharging and sampling the rows of array in series. In the time between precharging and sampling a row, the charge in the pixels decreases with the time exposed to incident light is known as exposure time.

The exposure time is controlled by adjusting the time interval between precharging 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 10-bit data for each pixel in the array.

figure 2-1 OV8830 block diagram



2.3 format and frame

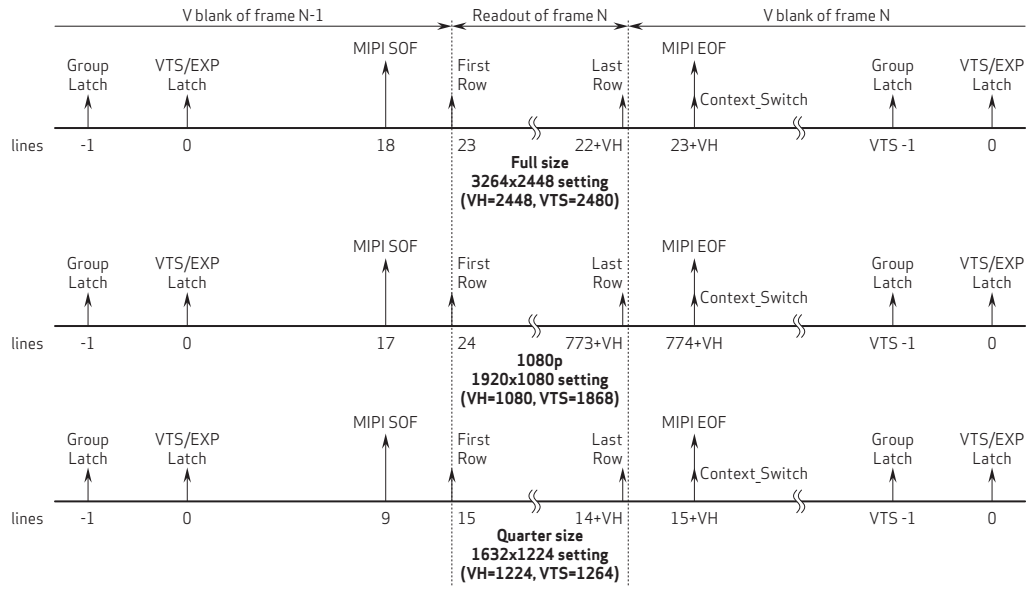
The OV8830 supports RAW RGB output with 1/2/4 lane MIPI interface.

table 2-1 supported resolution and frame rate

format[a]	resolution	max frame rate with MIPI	technology
full resolution (see **figure 2-2** and **table 2-3**)	3264x2448	24 fps	full
16:9 6Mpixel	3264x1836	30 fps	crop
EIS1080p (see **figure 2-3** and **table 2-3**)	2112x1188	30 fps	crop+scale 1.5 (3168x1782)
1080p (see **figure 2-2** and **table 2-3**)	1920x1080	30 fps	crop+scale 1.7 (3264x1836)
EIS720p (see **figure 2-3** and **table 2-3**)	1408x792	60 fps	crop+binningx2+scale 1.5 (3238x1822)
720p (see **figure 2-3** and **table 2-3**)	1280x720	60 fps	crop+binningx2+scale 1.25 (3200x1800)
EISQ1080p	1056x594	60 fps	crop+binningx2+scale 1.5 (3168x1782)
Q1080p	960x540	60 fps	crop+binningx2+scale 1.7 (3264x1836)
EISVGA	704x528	60 fps	crop+binningx2+skipx2+scale 1.15 (3238x2428)
VGA (see **figure 2-4** and **table 2-3**)	640x480	90 fps / (200 fps)	crop+binningx2+skipx2+scale 1.27 / (crop+skipx4) (3264x2448)
QVGA	320x240	400 fps	crop+skip

a. all formats with minimum 4 dummy lines and 4 dummy pixels

figure 2-2 exposure/gain latch points for full resolution, 1080p, and quarter size

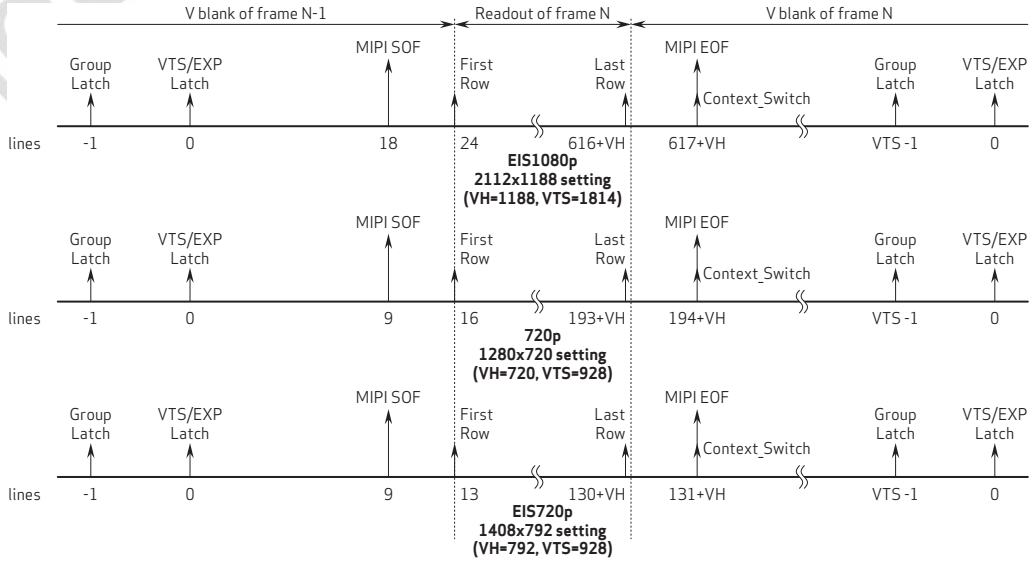


note 1 VTS = Total Vertical Size in units of lines (refer to registers 0x380E, 0x380F)

note 2 VH = Vertical Endpoint (refer to registers 0x3806, 0x3807)

8830_DS_2_2

figure 2-3 exposure/gain latch points for EIS1080p, EIS720p, and 720p

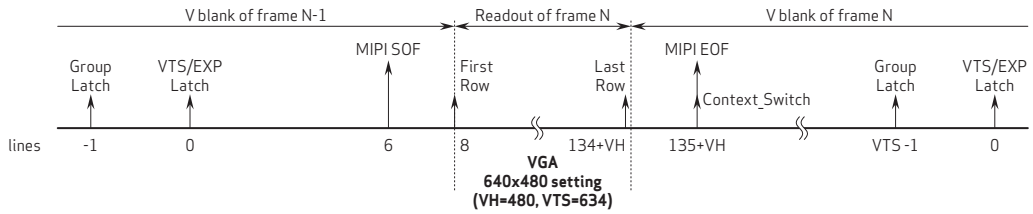


note 1 VTS = Total Vertical Size in units of lines (refer to registers 0x380E, 0x380F)

note 2 VH = Vertical Endpoint (refer to registers 0x3806, 0x3807)

8830_DS_2_3

figure 2-4 exposure/gain latch points for VGA



note 1 VTS = Total Vertical Size in units of lines (refer to registers 0x380E, 0x380F)

note 2 VH = Vertical Endpoint (refer to registers 0x3806, 0x3807)

8830_DS_2_4

2.4 I/O control

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

function	register	description
output drive capability control	0x3001	Bit[6:5]: pad I/O drive capability 00: 1x 01: 2x 10: 3x 11: 4x
VSYNC I/O control	0x3002	Bit[7]: input/output control for VSYNC pad 0: input 1: output
VSYNC output select	0x3027	Bit[7]: output selection for VSYNC pad 0: normal data path (vertical sync signal) 1: register control value
VSYNC output value	0x3009	Bit[7]: VSYNC output value
IL_PWM I/O control	0x3005	Bit[2]: input/output control for IL_PWM pad 0: input 1: output
IL_PWM output select	0x3024	Bit[0]: output selection for IL_PWM pad 0: normal data path (illumination control signal) 1: register control value
IL_PWM output value	0x3006	Bit[0]: IL_PWM output value
FREX I/O control	0x3002	Bit[4]: input/output control for FREX pad 0: input 1: output
FREX output select	0x3027	Bit[4]: output selection for FREX pad 0: normal data path 1: register control value

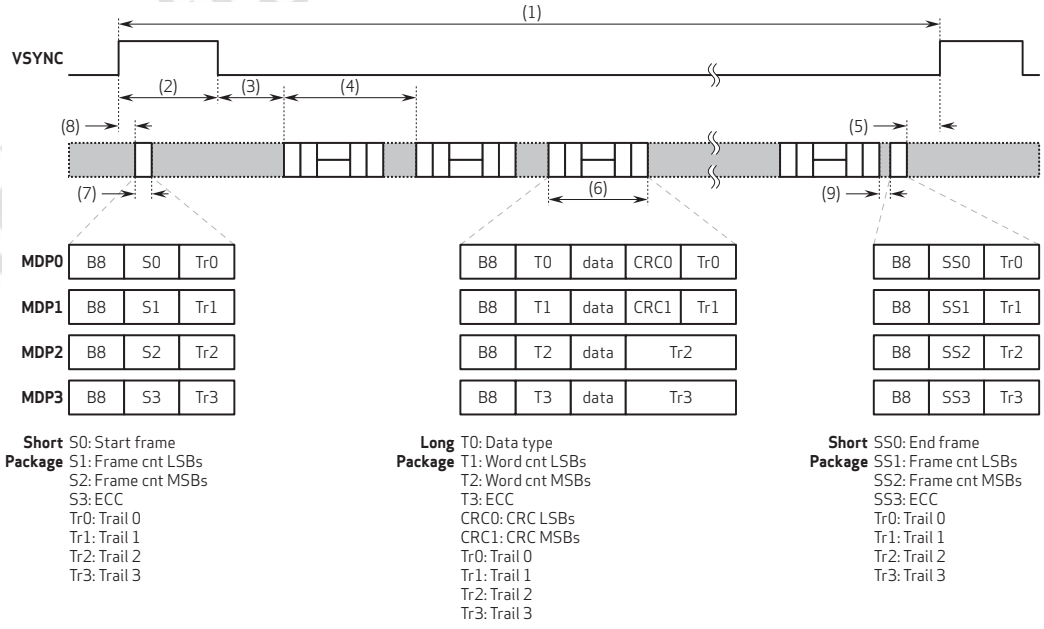
table 2-2 I/O control registers (sheet 2 of 2)

function	register	description
FREX output value	0x3009	Bit[4]: FREX output value
STROBE I/O control	0x3002	Bit[3]: input/output control for STROBE pad 0: input 1: output
STROBE output select	0x3027	Bit[3]: output selection for STROBE pad 0: normal data path 1: register control value
STROBE output value	0x3009	Bit[3]: STROBE output value

2.5 MIPI interface

The OV8830 supports 1/2/4-lane MIPI transmitter interface with up to 800 Mbps per lane MIPI interface.

figure 2-5 MIPI timing



Short Package
S0: Start frame
S1: Frame cnt LSBs
S2: Frame cnt MSBs
S3: ECC
Tr0: Trail 0
Tr1: Trail 1
Tr2: Trail 2
Tr3: Trail 3

Long Package
T0: Data type
T1: Word cnt LSBs
T2: Word cnt MSBs
T3: ECC
CRC0: CRC LSBs
CRC1: CRC MSBs
Tr0: Trail 0
Tr1: Trail 1
Tr2: Trail 2
Tr3: Trail 3

Short Package
SS0: End frame
SS1: Frame cnt LSBs
SS2: Frame cnt MSBs
SS3: ECC
Tr0: Trail 0
Tr1: Trail 1
Tr2: Trail 2
Tr3: Trail 3

note 1 input clock is 24 MHz

note 2 in 8 megapixel and VGA formats, MIPI 4-lane mode, MIPI data rate is 528.5 Mbps/lane; tp is one MIPI clock lane clock cycle, which in this case is 2/528.5 MHz

note 3 in 1080p, EIS1080p, 720p, EIS720p, and quartersize formats, MIPI 4-lane mode, MIPI data rate is 407.8 Mbps/lane; tp is one MIPI clock lane clock cycle, which in this case is 2/407.8 MHz

note 4 1080p, EIS1080p, 720p, EIS720p, and VGA are generated using raw scale function, in which value (4) will vary

8830_DS_2_5

table 2-3 MIPI timing specifications (sheet 1 of 2)

mode	timing
8 Megapixel 3264x2448 24 fps	(1) 4,481,136 tps (2) 1,024 tps (3) 43,834 tpp (4) 2,102 tpp (5) 31,748 tpp (6) 1,032 tpp (7) 2 tpp (8) 136 tpp (9) 84 tpp where tps = 1 Tsclk, tpp = 1 Tpclk
1080p 3264x1836 => 1920x1080 (scale) 30 fps	(1) 3,377,088 tps (2) 1,024 tps (3) 45,940 tpp (4) 2,104 or 4,210 tpp (5) 23,743 tpp (6) 612 tpp (7) 2 tpp (8) 136 tpp (9) 76 tpp where tps = 1 Tsclk, tpp = 1 Tpclk
EIS1080p 3168x1782 => 2112x1188 (scale) 30 fps	(1) 1,674,112 tps (2) 1,024 tps (3) 32,347 tpp (4) 2,104 or 4,210 tpp (5) 37,611 tpp (6) 342 tpp (7) 2 tpp (8) 136 tpp (9) 75 tpp where tps = 1 Tsclk, tpp = 1 Tpclk
720p 3200x1800 => 1280x720 (binning + scale) 30 fps	(1) 1,688,544 tps (2) 1,024 tps (3) 28,147 tpp (4) 2,104 or 4,210 tpp (5) 50,159 tpp (6) 412 tpp (7) 2 tpp (8) 136 tpp (9) 75 tpp where tps = 1 Tsclk, tpp = 1 Tpclk

table 2-3 MIPI timing specifications (sheet 2 of 2)

mode	timing
EIS720p 2816x1584 => 1408x792 (binning + scale) 30 fps	(1) 1,710,192 tps (2) 1,024 tps (3) 32,367 tpp (4) 2,104 or 4,210 tpp (5) 37,483 tpp (6) 452 tpp (7) 2 tpp (8) 136 tpp (9) 75 tpp where tps = 1 Tsclk, tpp = 1 Tpclk
VGA 1280x960 => 640x480 (skip + scale) 90 fps	(1) 1,147,344 tps (2) 1,024 tps (3) 31,891 tpp (4) 2,104 or 4,210 tpp (5) 38,196 tpp (6) 212 tpp (7) 2 tpp (8) 135 tpp (9) 76 tpp where tps = 1 Tsclk, tpp = 1 Tpclk

2.6 VSYNC timing

2.6.1 VSYNC modes

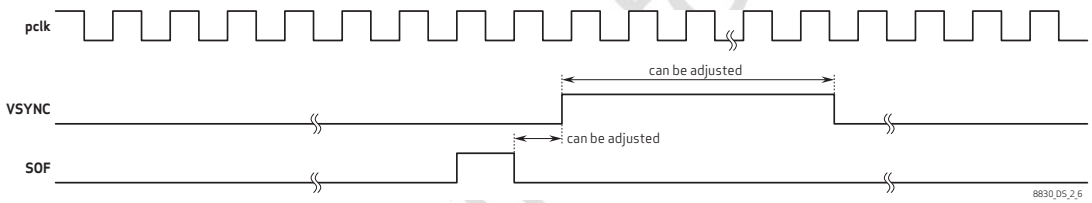
The VSYNC rising edge delay is controlled by register vsync_delay ({0x4314, 0x4315, 0x4316}) in all three VSYNC modes. VSYNC width is controlled by register vsync_width_pixel ({0x4311, 0x4312}) for VSYNC modes 1 and 2. The steps of both registers vsync_delay and vsync_width_pixel are 1 pclk cycle.

Note that VSYNC timing in mode 3 is a long VSYNC mode. The register vsync_width_pixel ({0x4311, 0x4312}) controls VSYNC falling edge differently.

2.6.1.1 VSYNC mode 1

In mode 1, VSYNC is generated by the internal start of frame (SOF) signal (see **figure 2-6**).

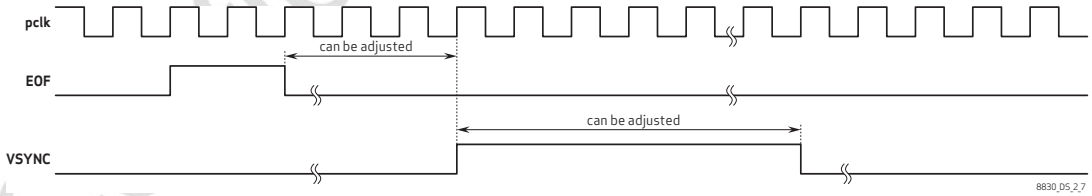
figure 2-6 VSYNC timing in mode 1



2.6.1.2 VSYNC mode 2

In mode 2, VSYNC is generated by the internal end of frame (EOF) signal (see **figure 2-7**).

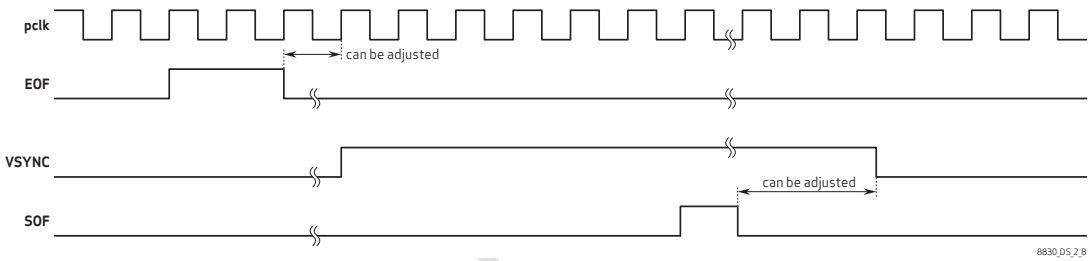
figure 2-7 VSYNC timing in mode 2



2.6.1.3 VSYNC mode 3

In mode 3, VSYNC is generated by EOF and SOF (see **figure 2-8**).

figure 2-8 VSYNC timing in mode 3



2.6.2 VSYNC control

There are two registers used to control VSYNC width. They are organized as follows: {0x4311, 0x4312} which controls the VSYNC width in units of pixel cycles.

For example, if registers {0x4311, 0x4312} = 0x08, VSYNC width is 8 pixel cycles in full size mode.

2.6.2.1 adjusting VSYNC position

There are three registers used to control the VSYNC position in reference to EOF/SOF. They are 0x4314, 0x4315, and 0x4316. These registers control the latency between EOF/SOF and VSYNC.

table 2-4 VSYNC control registers

address	register name	default value	R/W	description
0x4311	VSYNC_WIDTH_H	0x04	RW	Bit[7:0]: VSYNC width[15:8] (in terms of pixel numbers) high byte
0x4312	VSYNC_WIDTH_L	0x00	RW	Bit[7:0]: VSYNC width[7:0] (in terms of pixel numbers) low byte
0x4313	VSYNC_CTRL	0x00	RW	Bit[4]: VSYNC polarity Bit[3:2]: VSYNC output select 00: VSYNC output 11: Bypass EOF to output Bit[1]: VSYNC mode 2 Bit[0]: VSYNC mode 1
0x4314	VSYNC_DELAY1	0x00	RW	Bit[7:0]: VSYNC trigger signal to VSYNC output delay[23:16]
0x4315	VSYNC_DELAY2	0x01	RW	Bit[7:0]: VSYNC trigger signal to VSYNC output delay[15:8]
0x4316	VSYNC_DELAY3	0x00	RW	Bit[7:0]: VSYNC trigger signal to VSYNC output delay[7:0]

2.7 power management

2.7.1 power up sequence

The OV8830 can use either the internal regulator or an external power supply to provide digital core 1.2V DVDD. When an external 1.2V is used to provide DVDD power, EXTREG_EN must be pulled to DOVDD which is used to disable the internal regulator to avoid any unstable conflict between the external DVDD and output of the internal regulator. At the same time, the internal regulator must be turned off by a control register.

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

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

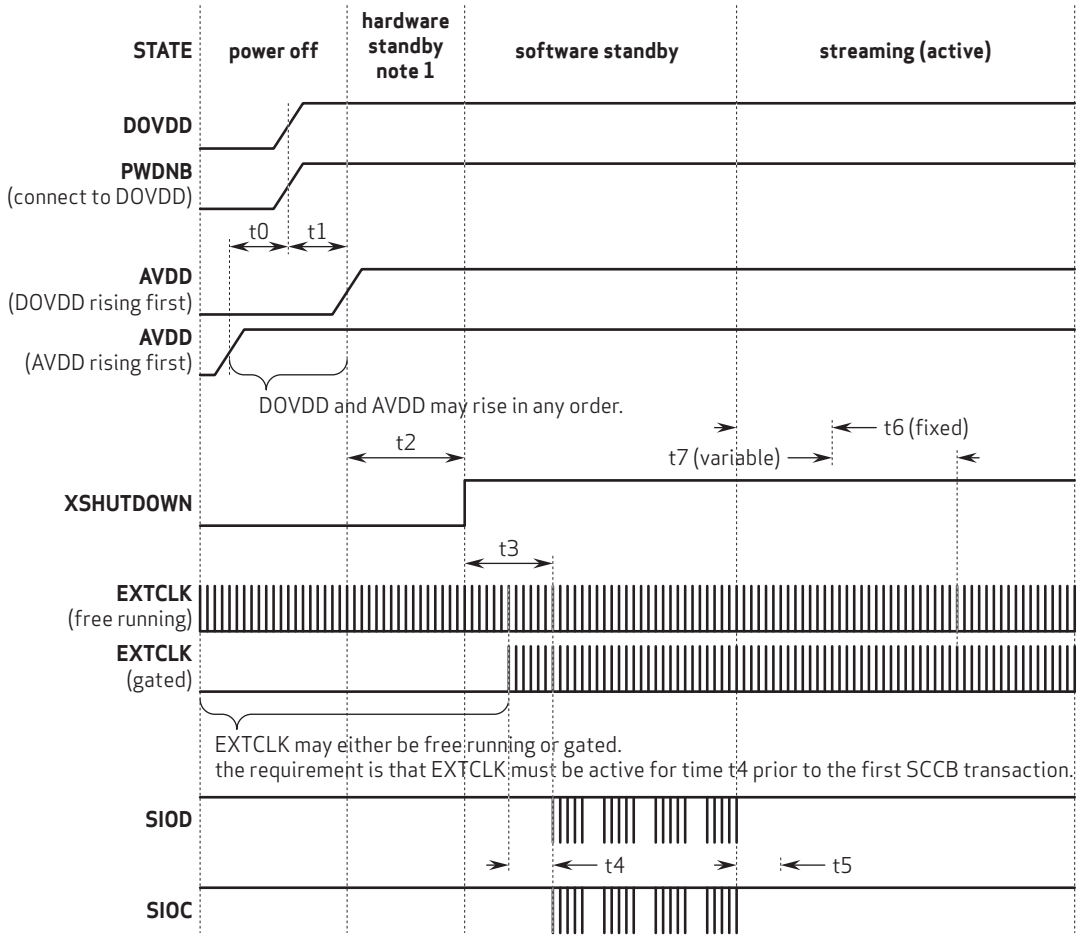
table 2-5 power up sequence

case	DVDD	EXTREG_EN	XSHUTDOWN	PWDNB	power up sequence requirement
1	internal	DGND	GPIO	DOVDD	Refer to **figure 2-9** 1. AVDD rising can occur before or after DOVDD rising as long as they are before XSHUTDOWN rising 2. XSHUTDOWN is pulled up after AVDD and DOVDD are stable
2	internal	DGND	DOVDD	GPIO	Refer to **figure 2-10** 1. AVDD rising occurs before DOVDD rising 2. PWDNB rising occurs after DOVDD rising
3	external	DOVDD	GPIO	DOVDD	Refer to **figure 2-11** 1. DOVDD rising must occur before external DVDD rising 2. AVDD rising can occur before or after DOVDD rising 3. XSHUTDOWN rising must occur after AVDD, DOVDD and DVDD are stable
4	external	DOVDD	DOVDD	GPIO	Refer to **figure 2-12** 1. AVDD rising occurs before DOVDD rising 2. DOVDD rising occurs before DVDD 3. PWDNB rising occurs after DVDD rising

table 2-6 power up sequence timing constraints

constraint	label	min	max	unit
AVDD rising – DOVDD rising	t0	0	∞	ns
DOVDD rising – AVDD rising	t1			ns
AVDD or DOVDD rising, whichever is last – XSHUTDOWN rising	t2	0.0		ns
XSHUTDOWN rising – first SCCB transaction	t3	8192 EXT CLKS		EXTCLK cycles
minimum number of EXTCLK cycles prior to the first SCCB transaction	t4	8192 EXTCLKS		EXTCLK cycles
PLL start up/lock time	t5		0.2	ms
entering streaming mode – first frame start sequence (fixed part)	t6		10	ms
entering streaming mode – first frame start sequence (variable part)	t7	delay is the exposure time value		lines
XSHUTDOWN rising – PWDNB rising	t8	0	∞	ns
AVDD or DOVDD, whichever is last – external DVDD	t9	0	∞	ns
external DVDD – PWDNB rising	t10	0	∞	ns
external DVDD – XSHUTDOWN rising	t11	0	∞	ns

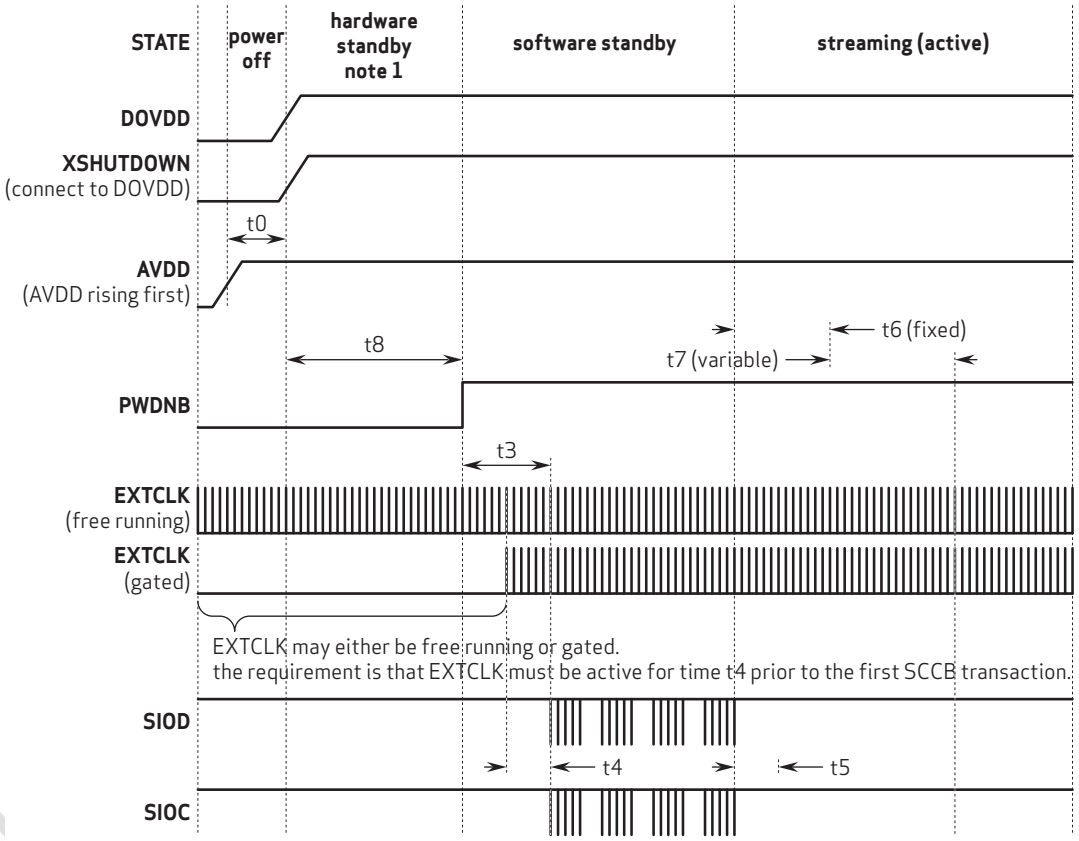
figure 2-9 power up sequence (case 1)



note 1 with minimum power consumption

8830_DS_2_9

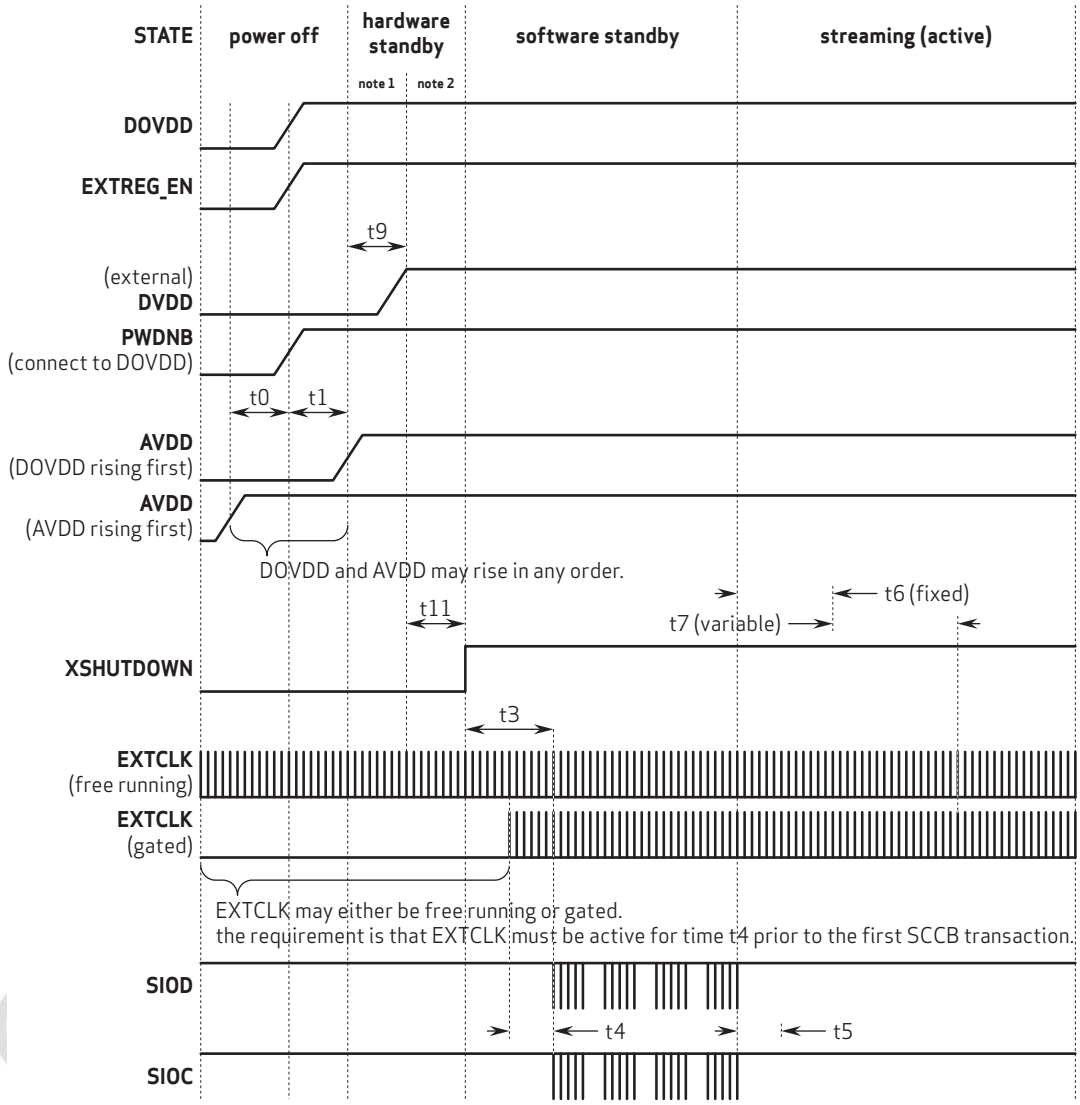
figure 2-10 power up sequence (case 2)



note 1 with low power consumption

8830_DS_2_10

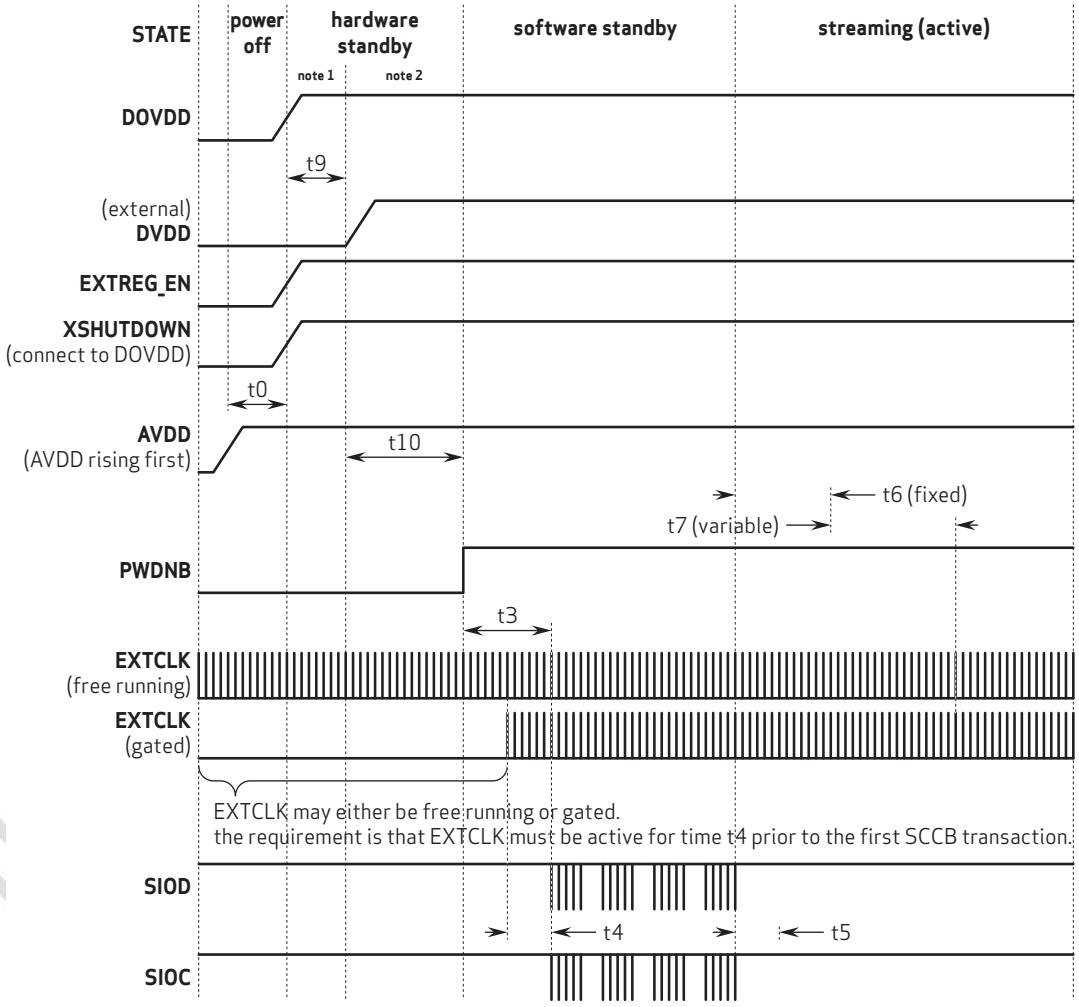
figure 2-11 power up sequence (case 3)



note 1 with minimum power consumption
note 2 with low power consumption

8830_DS_2_11

figure 2-12 power up sequence (case 4)



note 1 with minimum power consumption
note 2 with low power consumption

8830_DS_2_12

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). Similarly to the power up sequence, the XVCLK input clock may be either gated or continuous. If the SCCB command to exit streaming is received while a frame of MIPI data is being output, then the sensor must wait to the MIPI frame end code before entering software standby mode.

If the SCCB command to exit streaming mode is received during the inter frame time, then the sensor must enter software standby mode immediately.

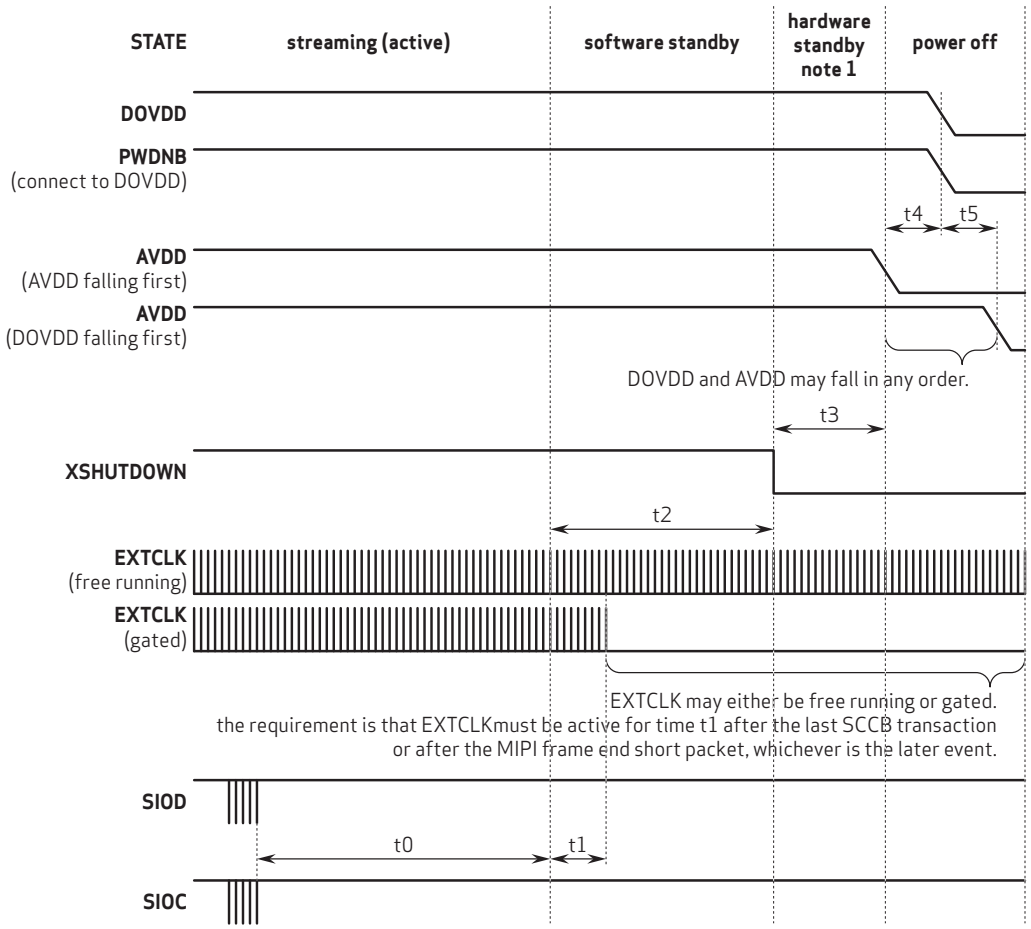
table 2-7 power down sequence

case	DVDD	XSHUTDOWN	PWDNB	power down sequence requirement
1	internal	GPIO	DOVDD	Refer to **figure 2-13** 1. software standby recommended 2. pull XSHUTDOWN low for minimum power consumption 3. AVDD and DOVDD may fall in any order
2	internal	DOVDD	GPIO	Refer to **figure 2-14** 1. software standby recommended 2. pull PWDNB low for low power consumption 3. pull DOVDD low for minimum power consumption or power off (XSHUTDOWN is connected to DOVDD) 4. pull AVDD low
3	external	GPIO	DOVDD	Refer to **figure 2-15** 1. software standby recommended 2. pull XSHUTDOWN low for low power consumption 3. cut off DVDD, then it will be in hardware standby state for minimum power consumption 4. pull AVDD and DOVDD low in any order
4	external	DOVDD	GPIO	Refer to **figure 2-16** 1. software standby recommended 2. pull PWDNB low for low power consumption 3. cut off DVDD, then it will be in hardware standby mode with minimum power consumption 4. pull DOVDD low (XSHUTDOWN connected to DOVDD) 5. pull AVDD low

table 2-8 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 indefinite		ns
DOVDD falling – AVDD falling	t5			ns
PWDNB falling – DOVDD falling	t6	0.0		ns
XSHUTDOWN falling – external DVDD falling	t7	0.0		ns
external DVDD falling – AVDD falling or DOVDD falling whichever is first	t8	0.0		ns
PWDNB falling – external DVDD falling	t9	0.0		ns

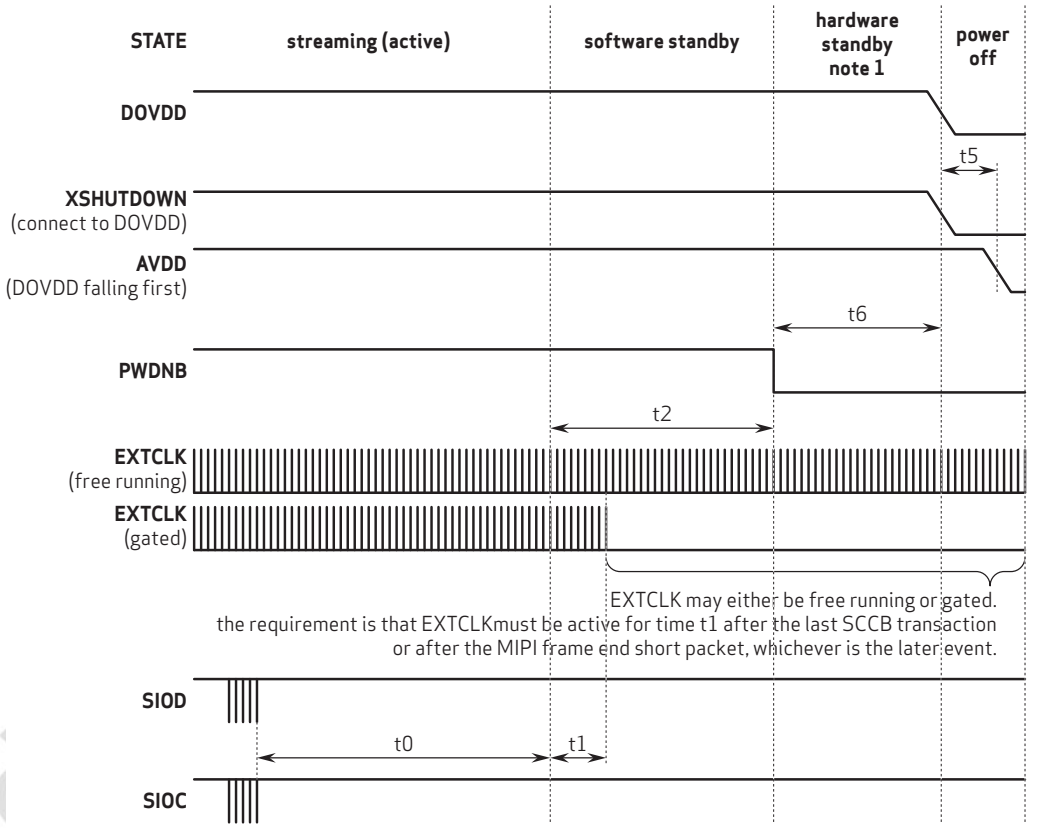
figure 2-13 power down sequence (case 1)



note 1 with minimum power consumption

8830_DS_2_13

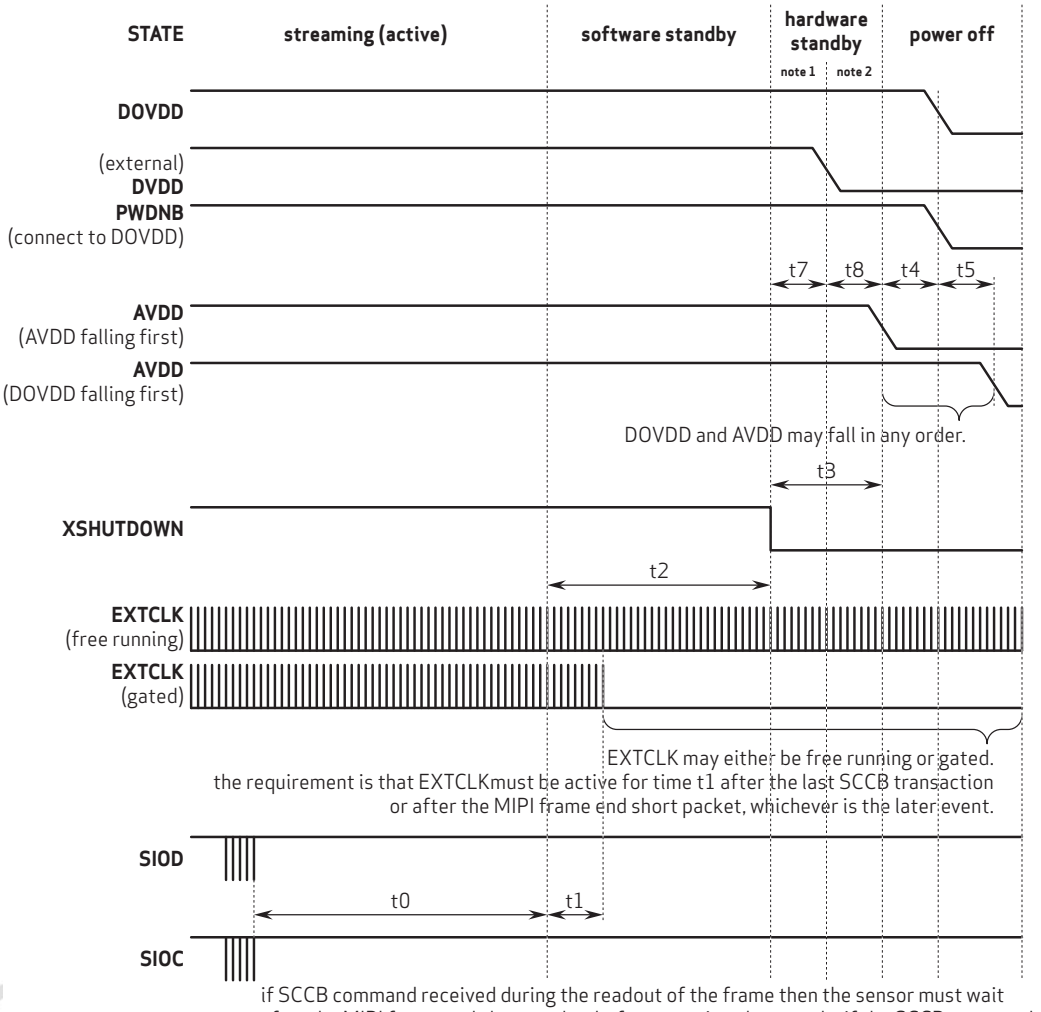
figure 2-14 power down sequence (case 2)



note 1 with low power consumption

8830_DS_2_14

figure 2-15 power down sequence (case 3)

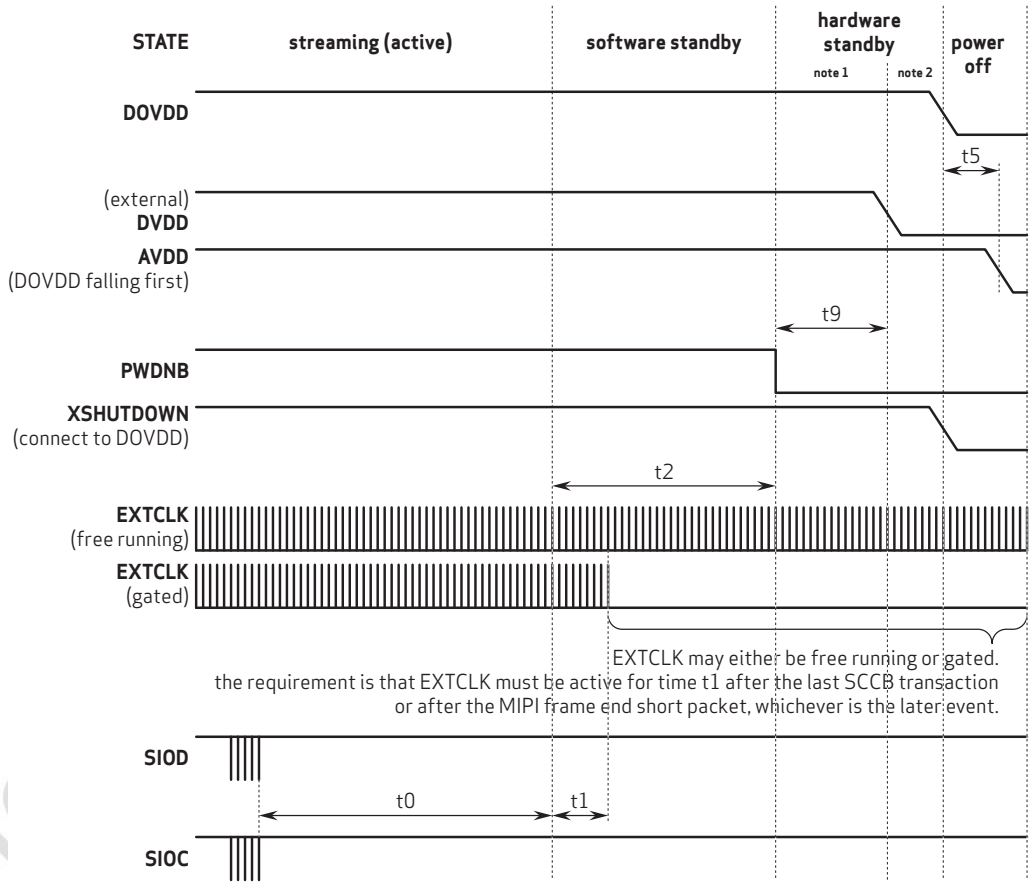


note 1 with low power consumption

note 2 with minimum power consumption

8830_DS_2_15

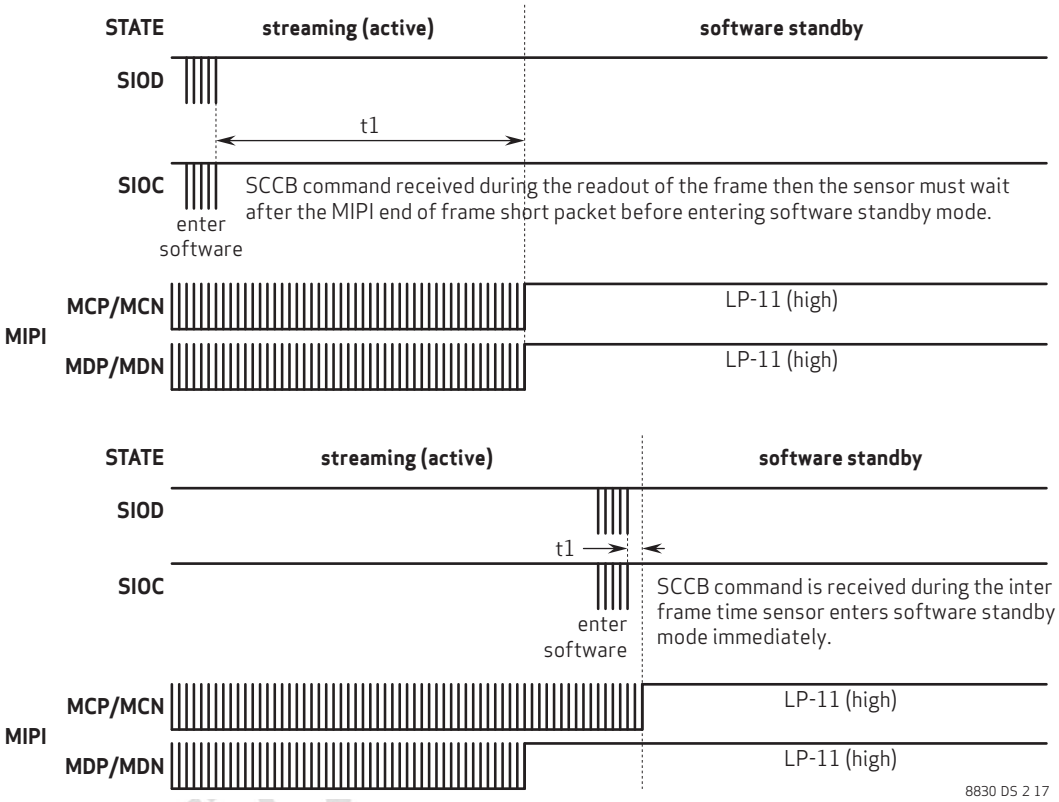
figure 2-16 power down sequence (case 4)



note 1 with low power consumption
note 2 with minimum power consumption

8830_DS_2_16

figure 2-17 standby sequence



2.8 reset

The OV8830 sensor includes a **XSHUTDOWN** pad (pad **11**) that forces a complete hardware reset when it is pulled low (GND). The OV8830 clears all registers and resets them to their default values when a hardware reset occurs. Reset requires ~2ms settling time.

2.8.1 power ON reset generation

The power on reset can be controlled from external pin. However, inside this chip, a power on reset is generated after core power becomes stable.

2.9 hardware and software standby

Two suspend modes are available for the OV8830:

- hardware standby
- software standby

table 2-9 hardware and standby description

mode	description
hardware standby with PWDNB	1. enabled by pull PWDNB low 2. input clock is gated by PWDNB, no I2C communication 3. register value keeps 4. power down all blocks and regulator 5. low power consumption 6. GPIO can be configured as hi/lo/tri-state
hardware standby with XSHUTDOWN	1. enabled by pull XSHUTDOWN low 2. power down all blocks 3. register value is reset to default 4. no I2C communication 5. minimum power consumption
software standby	1. default mode after power on reset 2. power down all blocks except I2C and regulator 3. register value keeps 4. I2C communication is available 5. low power consumption 6. GPIO can be configured as hi/lo/tri-state

2.10 system clock control

The OV8830 has two on-chip PLLs which generates the system clock with 6~27 MHz input clock. A programmable clock divider is provided to generate different frequency for the system.

2.10.1 PLL configuration

figure 2-18 OV8830 PLL1 clock diagram

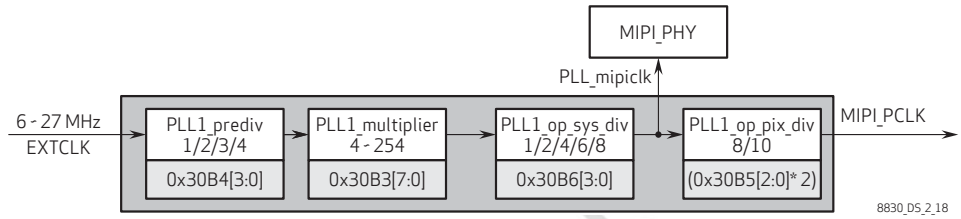


figure 2-19 OV8830 PLL2 clock diagram

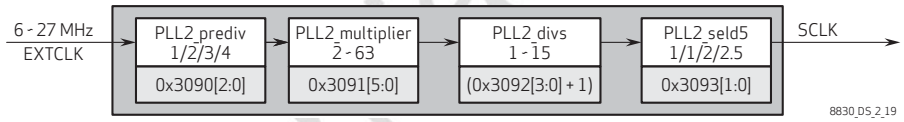
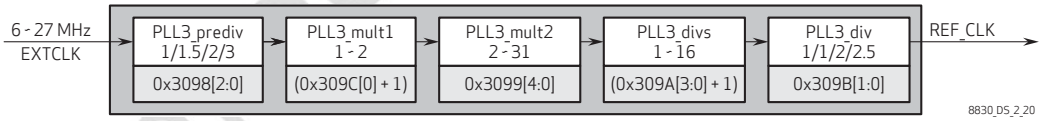


figure 2-20 OV8830 PLL3 clock diagram



note
Contact your local OVT FAE for additional assistance on PLL configuration.

table 2-10 sample PLL configuration[a] (sheet 1 of 2)

name	address	value
PLL2_prediv	0x3090[2:0]	3'h2
PLL2_multiplier	0x3091[5:0]	6'h12
PLL2_divs	0x3092[3:0]	4'h0
PLL2_seld5	0x3093[1:0]	2'b00
PLL1_multiplier[7:0]	0x30B3[7:0]	8'h54
PLL1_prediv	0x30B4[3:0]	4'h02
PLL1_op_pix_div	0x30B5[2:0]	3'h4
PLL1_op_sys_div	0x30B6[3:0]	4'h1

table 2-10 sample PLL configuration[a] (sheet 2 of 2)

name	address	value
PLL3_prediv	0x3098[2:0]	3'h03
PLL3_mult1	0x309C[0]	1'b0
PLL3_mult2	0x3099[4:0]	5'h1E
PLL3_divs	0x309A[3:0]	4'h00
PLL3_div	0x309B[1:0]	2'h00
SCLK		216 MHz
MIPI_PCLK		126 MHz
PLL_mipiclk		1008 MHz
EXTCLK		24 MHz
REF_CLK		240 MHz

a. PLL control for 8 Mpixel @ 24 fps with 4-lane, 10-bit output

figure 2-21 clock connection diagram

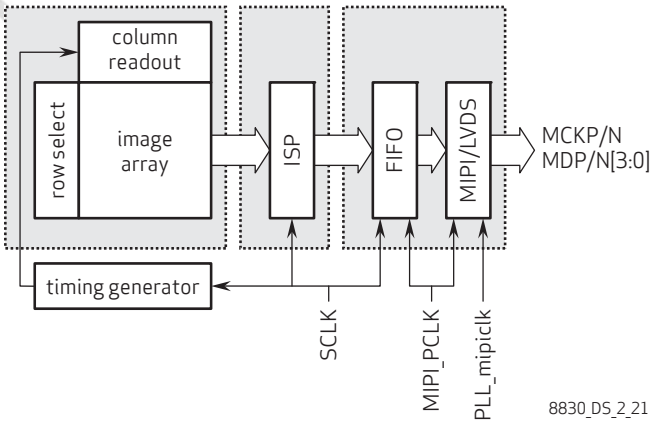


table 2-11 PLL speed limitation (sheet 1 of 2)

parameter	value
PLL1_multiplier input	4~27 MHz
PLL1_multiplier output	500~1008 MHz
PLL2_multiplier input	4~27 MHz

table 2-11 PLL speed limitation (sheet 2 of 2)

parameter	value
PLL2_multiplier output	200~400 MHz
PLL3_mult2 output	150~300 MHz
SCLK	max 222 MHz
REF_CLK	240~252 MHz

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 OV8830, the SCCB ID is controlled by the SID pin. If SID is low, the sensor's SCCB address comes from register 0x300C which has a default value of 0x20. If SID is high, the sensor's SCCB address comes from register 0x3661 which has a default value of 0x6C.

2.11.1 data transfer protocol

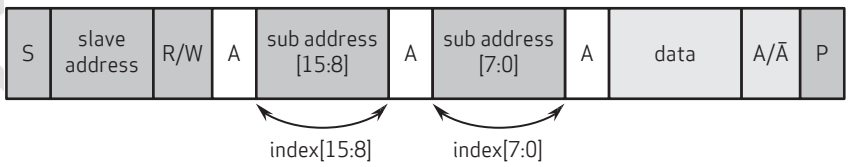
The data transfer of the OV8830 follows the SCCB protocol.

2.11.2 message format

The OV8830 supports the message format shown in **figure 2-22**. The the 7-bit address of OV8830 is 0x10 or 0x36 depending on SID, but can be programmed. If SID is low, the sensor's SCCB address comes from register 0x300C. If SID is high, the sensor's SCCB address comes from register 0x3661. The repeated START (Sr) condition is not shown in **figure 2-23**, but is shown in **figure 2-24** and **figure 2-25**.

figure 2-22 message type

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



from slave to master
from master to slave
direction depends on operation

S START condition
P STOP condition
Sr repeated START condition

A acknowledge
Ā negative acknowledge

8830_DS_2_22

2.11.3 read / write operation

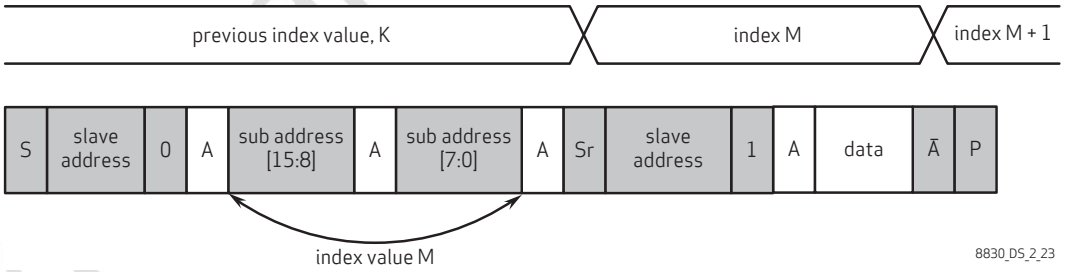
The OV8830 supports four different read operations and two different write operations:

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

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

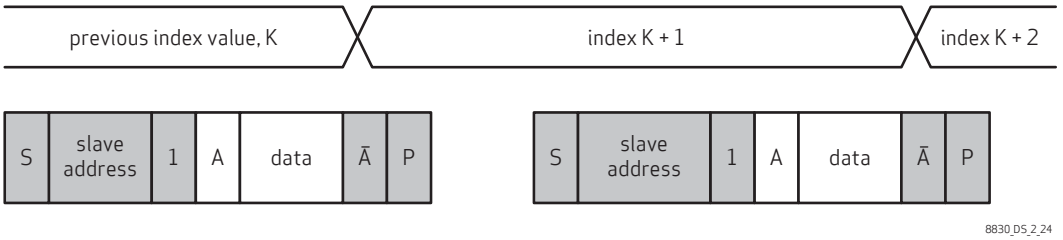
In a single read from random locations, the master does a dummy write operation to desired sub-address, issues a repeated start condition and then addresses the camera again with a read operation. After acknowledging its slave address, the camera starts to output data onto the SDA line as shown in **figure 2-23**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-23 SCCB single read from random location



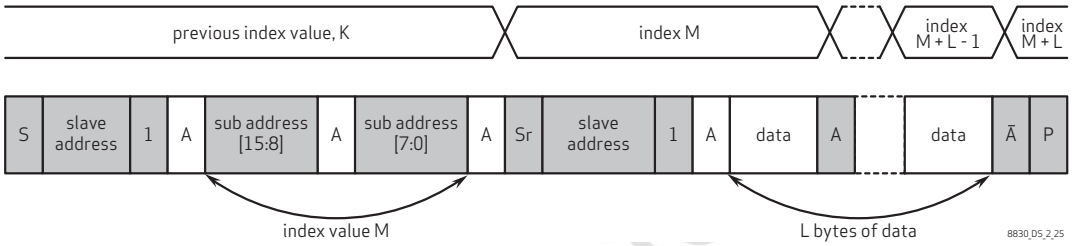
If the host addresses the camera with read operation directly without the dummy write operation, the camera responds by setting the data from last used sub-address to the SDA line as shown in **figure 2-24**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-24 SCCB single read from current location



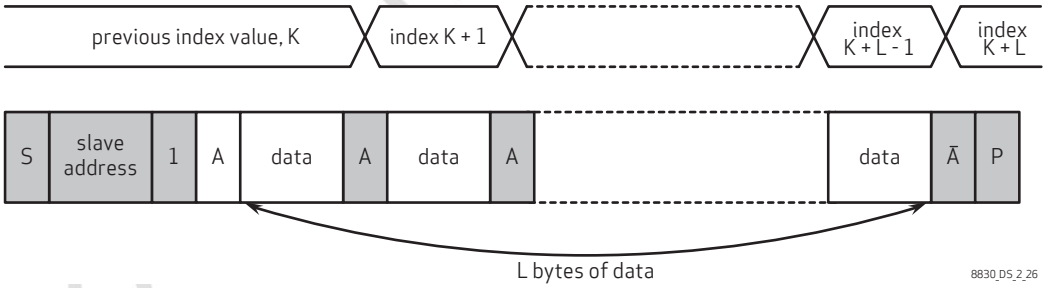
The sequential read from a random location is illustrated in **figure 2-25**. The master does a dummy write to the desired sub-address, issues a repeated start condition after acknowledge from slave and addresses the slave again with read operation. If a master issues an acknowledge after receiving data, 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 2-25 SCCB sequential read from random location



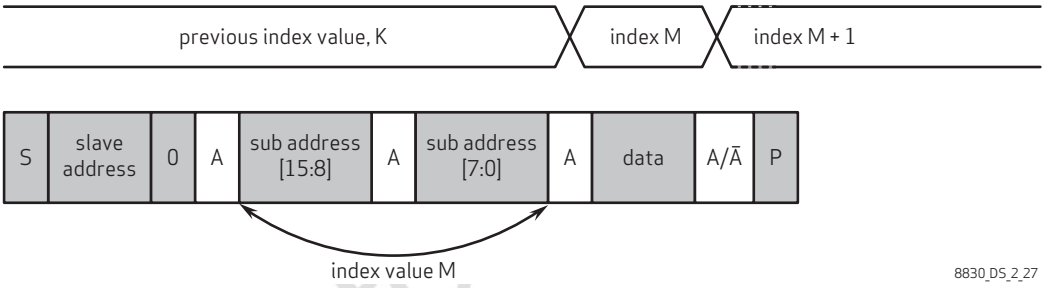
The sequential read from current location is similar to a sequential read from a random location. The only exception is that there is no dummy write operation, as shown in **figure 2-26**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-26 SCCB sequential read from current location



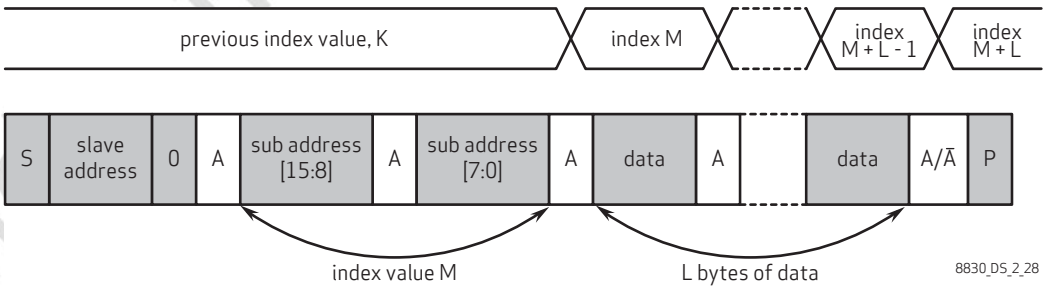
The write operation to a random location is illustrated in **figure 2-27**. The master issues a write operation to the slave, sets the sub-address and data correspondingly after the slave has acknowledged. The write operation is terminated with a stop condition from the master.

figure 2-27 SCCB single write to random location



The sequential write is illustrated in **figure 2-28**. The slave automatically increments the sub-address after each data byte. The sequential write operation is terminated with stop condition from the master.

figure 2-28 SCCB sequential write to random location



2.11.4 SCCB timing

figure 2-29 SCCB interface timing

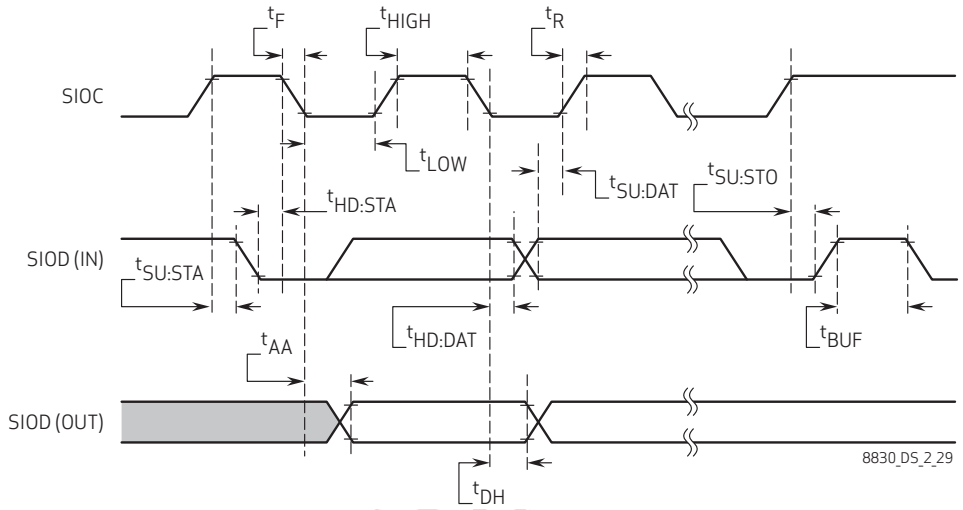


table 2-12 SCCB interface timing specifications[ab]

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 or/and of the falling edge signifies 10%, timing measurement shown in the middle of the rising/falling edge signifies 50%, timing measurement shown at the beginning of the rising edge or/and of the falling edge signifies 90%

2.11.5 group write

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

The OV8830 supports up to four groups. These groups share 512 x 8 bits or 512 bytes and the size of each group is programmable by adjusting the start address. The maximum size for each group is 63 registers.

table 2-13 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: Reserved 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: Reserved
0x3209	GRP0_PERIOD	0x00	RW	Frames For Staying in Group 0
0x320A	GRP1_PERIOD	0x00	RW	Frames For Staying in Group 1
0x320B	GRP_SWCTRL	0x01	RW	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

The OV8830 supports four launch modes:

1. Quick Manual Launch mode
   a. Quick manual launch mode is activated by writing register 0x3208[7:4] to be 4'b1110. Then the contents of group defined by register 0x3208[3:0] will be written into the target registers immediately after the sensor receives the SCCB command.
   b. Quick manual launch mode will stop current frame immediately, which may cause an incomplete broken image, but will dramatically reduce shutter lag (the time interval between resolution switch command to the start of exposure of the first line of the desired frame) to zero.
2. Delay Manual Launch mode
   a. Delay manual launch mode is achieved by writing 0x3208[7:4] to be 4'b1010. The writing of the contents from the group defined by register 0x3208[3:0] will be delayed. The delay is defined internally by a time set in vertical blanking.
   b. Delay manual launch mode will keep current frame complete without a broken image as opposed to Quick Manual Launch Mode. Depending when the SCCB command is issued, the average shutter lag is one half of a frame
3. Auto Launch mode
   a. Auto launch enable system returns automatically from second group (programmed by register 0x320B[1:0]) to first group (group 0). This function is enabled by register 0x320B[2]. Register 0x320A controls how many frames to stay before returning.
4. Repeat Launch mode
   a. Launch mode is repeated automatically between group 0 and second group. The number of frames to stay in those groups are controlled by register 0x3209 and 0x320A.

3 block level description

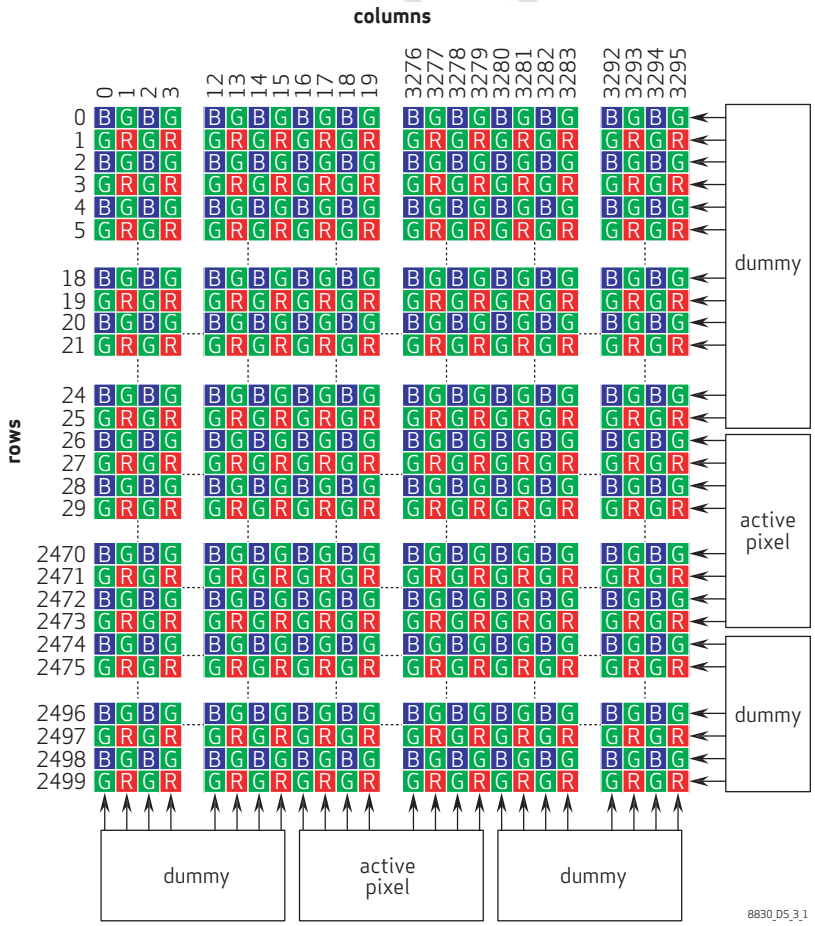
3.1 pixel array structure

The OV8830 sensor has an image array of 3296 columns by 2500 rows (8,240,000 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 8,240,000 pixels, 7,990,272 (3264x2448) are active pixels and can be output. The other pixels are used for black level calibration and interpolation.

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 binning

Binning mode is usually used for low resolution. When the binning function is ON, voltage levels of adjacent pixels are averaged before being sent to the ADC. If the binning function is OFF, the pixels, which are not output, are merely skipped. The OV8830 supports 2x2 binning. **figure 3-2** illustrates 2x2 binning, where the voltage levels of two horizontal (2x1) adjacent same-color pixels are averaged before entering the ADC.

figure 3-2 example of 2x2 binning

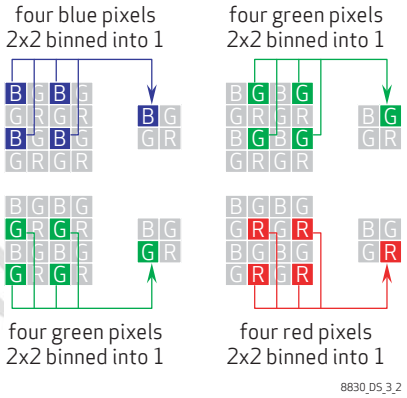


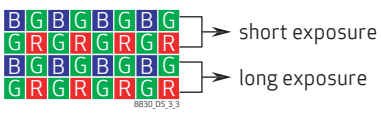
table 3-1 binning-related registers

address	register name	default value	R/W	description
0x3820	TIMING_FORMAT1	0x10	RW	Bit[0]: Vertical binning
0x3821	TIMING_FORMAT2	0x08	RW	Bit[0]: Horizontal binning
0x4512	INPUT_SWAP_MAN_EN	0x01	RW	Bit[0]: Vertical binning option 0: Sum 1: Average

3.3 alternate row HDR

In HDR mode, the exposure is still controlled by a rolling shutter. However, short and long exposure time is switched every two rows. Output sequence is line by line (see **figure 3-3**).

figure 3-3 alternate row HDR



4 image sensor core digital functions

4.1 mirror and flip

The OV8830 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

8830_DS_4_1

table 4-1 mirror and flip registers

address	register name	default value	R/W	description
0x3820	TIMING_FORMAT1	0x10	RW	Timing Control Register Bit[6]: Digital vertical flip enable 0: Normal 1: Vertical flip Bit[1]: Array vertical flip enable 0: Normal 1: Vertical flip
0x3821	TIMING_FORMAT2	0x08	RW	Timing Control Register Bit[2:1]: Horizontal mirror enable 00: Normal 11: 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. This windowing is achieved by simply masking the pixels outside the window; thus, it will not affect original timing.

figure 4-2 image windowing

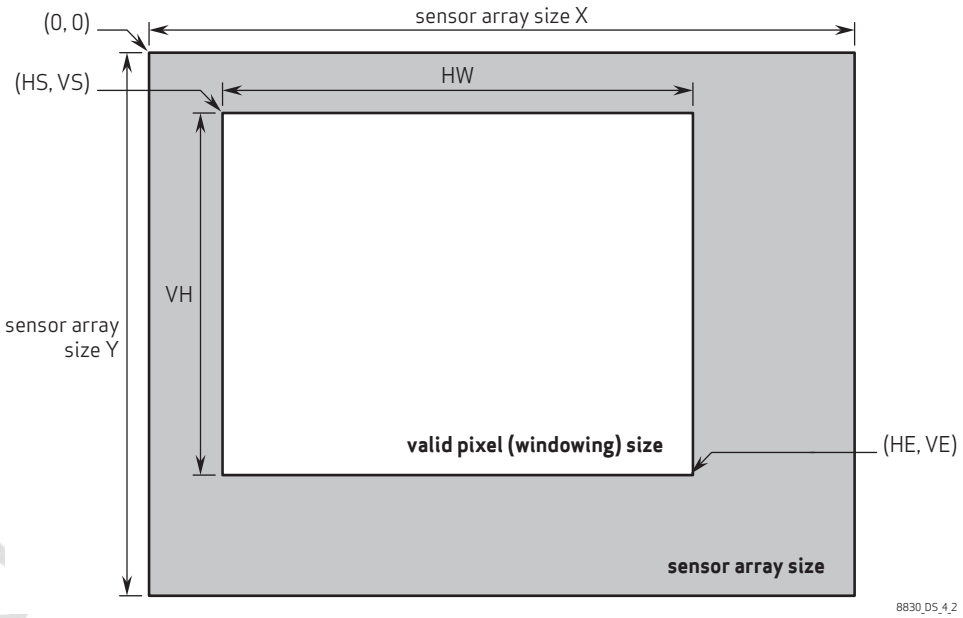


table 4-2 image windowing control functions

function	register	R/W	description
horizontal start	{0x3800, 0x3801}	RW	HS[12:8] = 0x3800 HS[7:0] = 0x3801
vertical start	{0x3802, 0x3803}	RW	VS[11:8] = 0x3802 VS[7:0] = 0x3803
horizontal end	{0x3804, 0x3805}	RW	HW[12:8] = 0x3804 HW[7:0] = 0x3805
vertical end	{0x3806, 0x3807}	RW	VH[11:8] = 0x3806 VH[7:0] = 0x3807

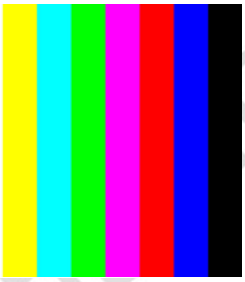
4.3 test pattern

For testing purposes, the OV8830 offers five types of test patterns:

- **general color bar**
- **test pattern I and II (16 bar)**
- **test pattern III and IV (horizontal fading)**

4.3.1 general color bar

figure 4-3 test pattern



color bar

table 4-3 general color bar selection control

function	register	default value	R/W	description
general color bar	0x5E00	0x00	RW	Bit[7]: test_enable Bit[3:2]: color_bar style 00: horizontal bar 01: vertical bar 10: horizontal fading bar 11: vertical fading bar

4.3.2 test pattern I and II (16 bar)

table 4-4 test pattern I and II selection control

function	register	default value	R/W	description
test pattern I & & II	0x4303	0x00	RW	Bit[4]: 16 color bar inverse 0: normal 1: inverse Bit[3]: 16 color bar enable 0: 16 color bar OFF 1: 16 color bar enable

4.3.3 test pattern III and IV (horizontal fading)

table 4-5 test pattern III and IV selection control

function	register	default value	R/W	description
test pattern III & & IV	0x4303	0x00	RW	Bit[7]: fading enable 0: disable 1: enable Bit[6]: horizontal fading inverse 0: normal 1: inverse

4.4 black level calibration (BLC)

The pixel array contains several optically shielded (black) lines. These lines are used as reference for black level calibration.

Black level adjustments can be made with registers 0x4000, 0x4002, 0x4003, and 0x4009.

table 4-6 BLC control functions (sheet 1 of 2)

address	register name	default value	R/W	description
0x4000	BLC_BYPASS	0x10	RW	Bit[7]: BLC bypass enable 0: Disable BLC bypass (BLC enabled) 1: Enable BLC bypass (no BLC) Bit[4]: BLC zero line window select 0: Choose full width as window 1: Window select as BLC does Bit[3:0]: Debug mode
0x4002	BLC_AUTO	0x45	RW	Bit[7]: Format change enable 0: BLC will remain the same after format change 1: BLC will redo after format change Bit[6]: BLC auto enable 0: Get black level manually from register 1: Calculate black level from auto statistics Bit[5:0]: Reset frame number[5:0] Frames that will continue to go through BLC after reset

table 4-6 BLC control functions (sheet 2 of 2)

address	register name	default value	R/W	description
0x4003	BLC_FREEZE	0x08	RW	Bit[7]: BLC redo enable 0: Normal 1: Force BLC to redo N frames (where N=0x4003[5:0]) when this bit is set Bit[6]: Freeze enable 0: Normal 1: BLC black level will not update. Priority lower than always update. Bit[5:0]: Manual frame number BLC redo frame number
0x4009	BLC_TARGET	0x10	RW	Bit[7:0]: Black target level[7:0]

4.5 one time programmable (OTP) memory

The OV8830 supports a maximum of 4K bits of one-time programmable (OTP) memory to store chip identification and manufacturing information. It can be controlled through the SCCB (see **table 4-7**). OmniVision reserves the last 59 bytes and first 5 bytes in OTP memory.

table 4-7 OTP control functions

function	register	description
OTP dump / program	0x3D80	Trigger Program OTP
OTP dump / program	0x3D81	Trigger Load / dump OTP
OTP bank select	0x3D84	Bit[7]: program_disable 0: OTP program enable 1: OTP program disable Bit[6]: Memory bank enable 0: Manual memory bank disable 1: Manual memory bank mode Bit[5:0]: Memory bank select
OTP start address	0x3D85	Bit[3:0]: Start byte index for the 16 bytes in memory bank
OTP end address	0x3D86	Bit[3:0]: End byte index for the 16 bytes in memory bank
OTP PS to CSB time	0x3D87	Bit[3:0]: PS to CSB time control by sclk
dump / program data n	0x3D00 ~ 0x3D0F	Bit[7:0]: Data dumped or data to be programmed for bits[8*(n+1)-1:8*n]

4.6 temperature sensor

The OV8830 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 4-8**). 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 bit 0x4D0B[7].

table 4-8 temperature sensor functions

function	register	R/W	description
TPM trigger / read	0x4D0B	RW	Bit[7]: temperature sensor trigger Bit[6:0]: measured temperature

4.7 strobe flash and frame exposure

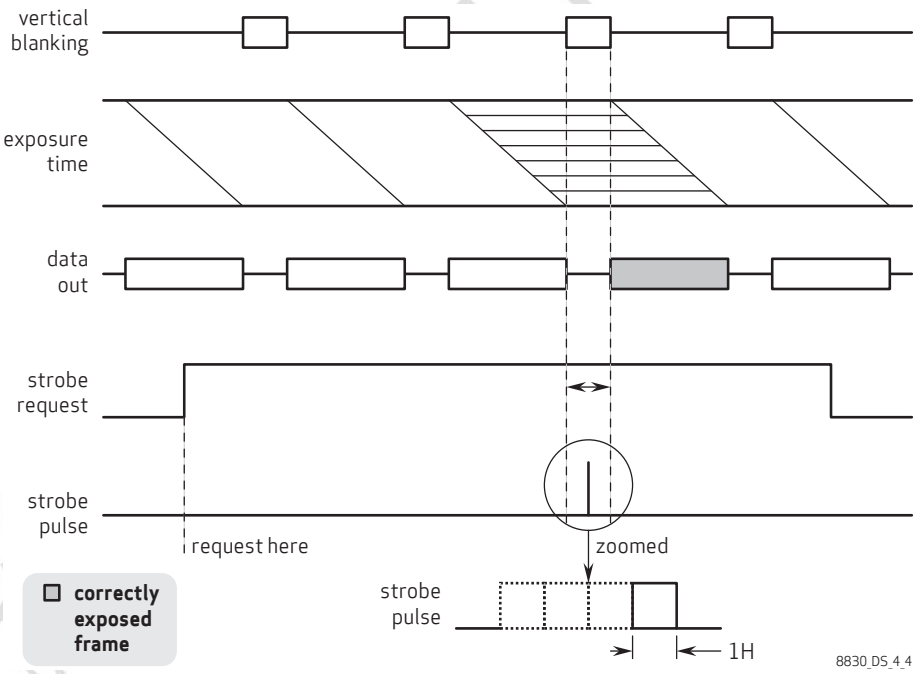
4.7.1 strobe flash control

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 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 flashlight modes.

4.7.1.1 xenon flash control

After a strobe request is submitted, the strobe pulse will be activated at the beginning of the third frame (see figure 4-4). 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 4-4 xenon flash mode



4.7.1.2 LED 1 & 2 mode

In LED 1 & 2 modes, the strobe pulse is active two frames after the strobe request is submitted and the third frame is correctly exposed. The strobe pulse will be activated only one time if the strobe end request is set as shown in **figure 4-5**. If end request has not been sent, the strobe signal is activated intermittently until the strobe end request is set (see **figure 4-6**). For longer strobe pulse, number of dummy lines can be inserted by controlling {0x3B02, 0x3B03}. When the strobe request is cleared, the dummy lines will be removed automatically.

figure 4-5 LED 1 & 2 mode - one pulse output

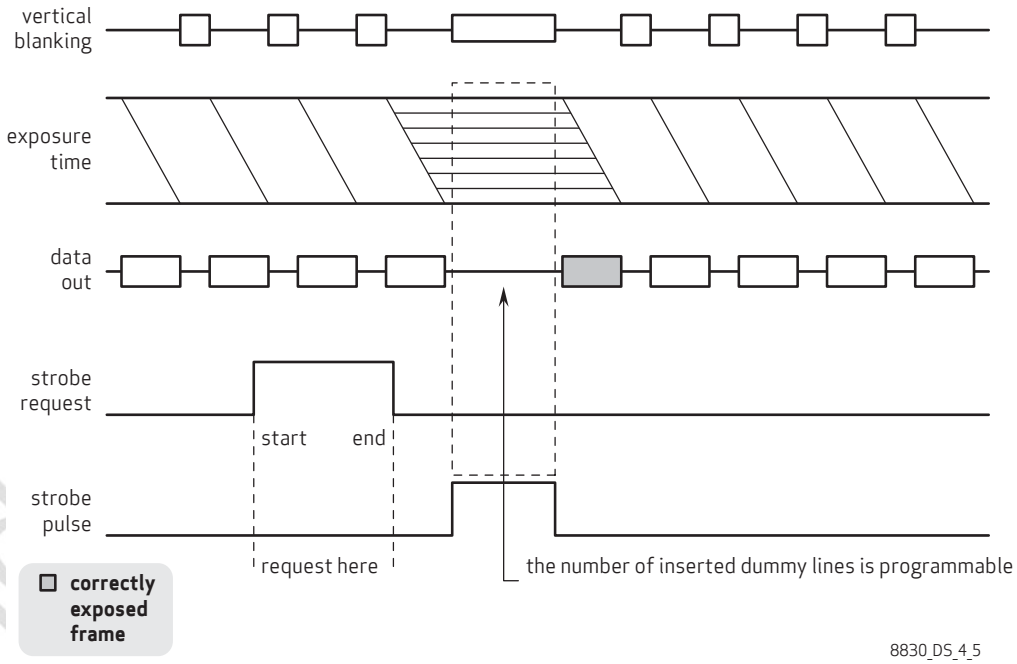
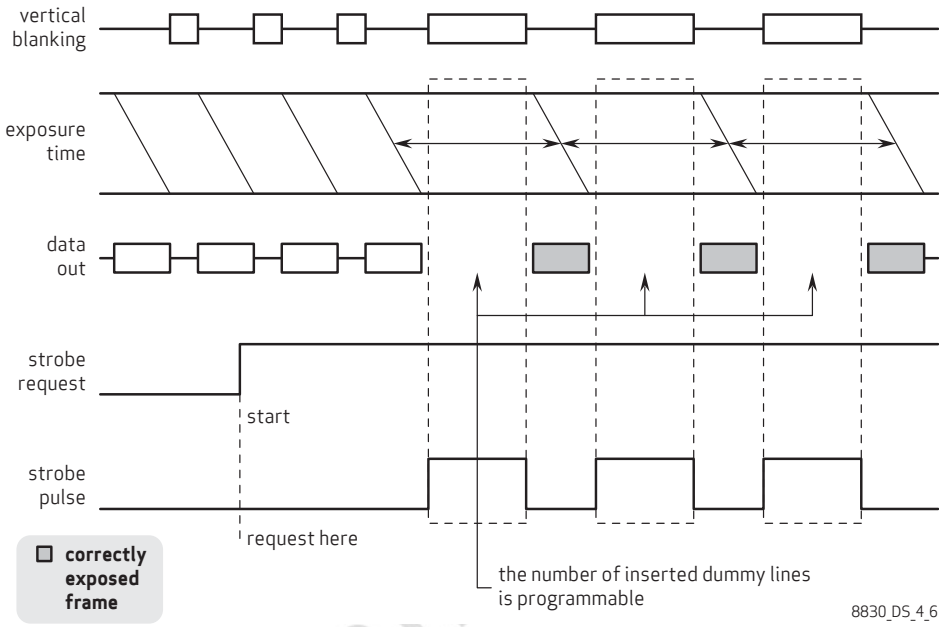


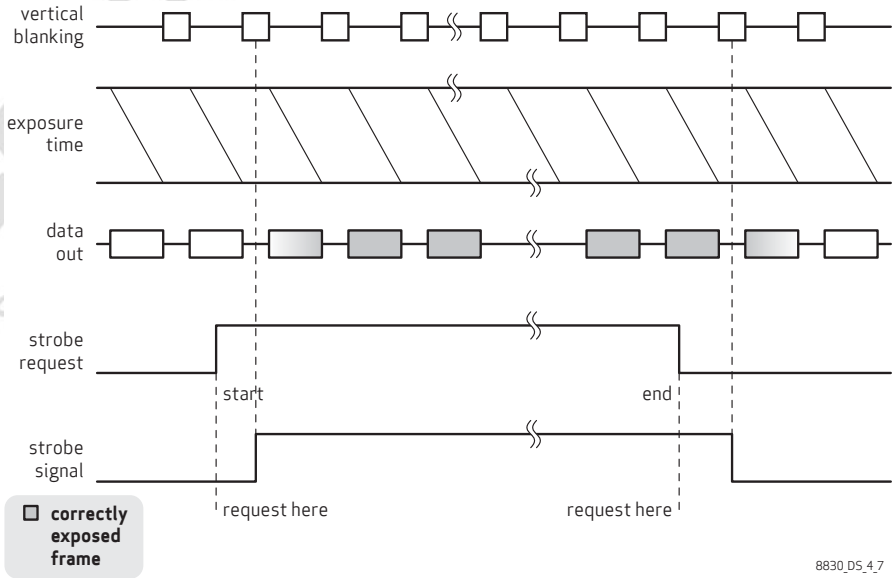
figure 4-6 LED 1 & 2 mode - multiple pulse output



4.7.1.3 LED 3 mode

In LED 3 mode, the strobe signal stays active until the strobe end request is sent (see **figure 4-7**).

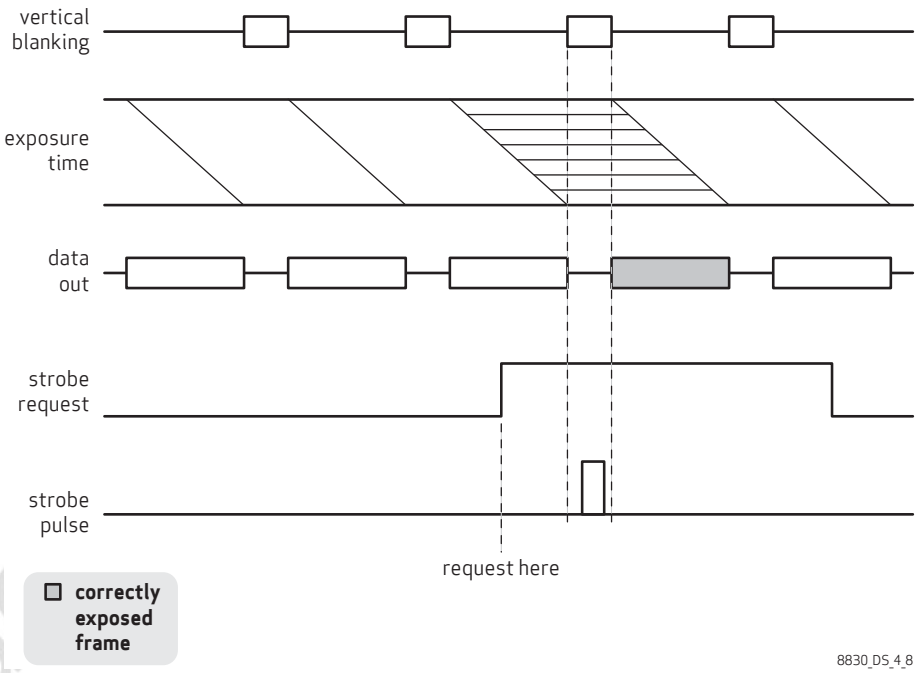
figure 4-7 LED 3 mode



4.7.1.4 LED 4 mode

In LED 4 mode, the strobe signal width is controlled by register 0x3B05 (see **figure 4-8**). Strobe width = 128 × (2^0x3B05[1:0]) × (0x3B05[7:2] + 1) × sclk_period. The maximum value of 0x3B02[7:2] is 6'b111110.

figure 4-8 LED 4 mode



4.7.2 frame exposure (FREX) mode

In FREX mode, whole frame pixels start integration at the same time, rather than integrating row by row. After the user-defined exposure time, the shutter closes, preventing further integration and the image begins to read out. After the readout finishes, the shutter opens again and the sensor resumes normal mode, waiting for the next FREX request.

The OV8830 supports two modes of FREX (see **figure 4-9** and **figure 4-10**):

figure 4-9 FREX mode 1

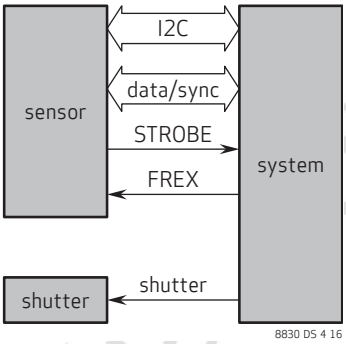
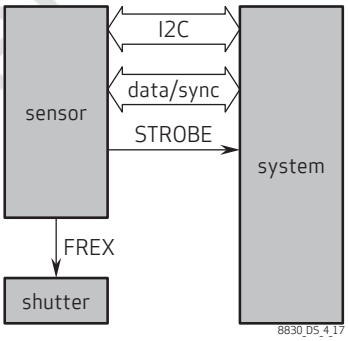


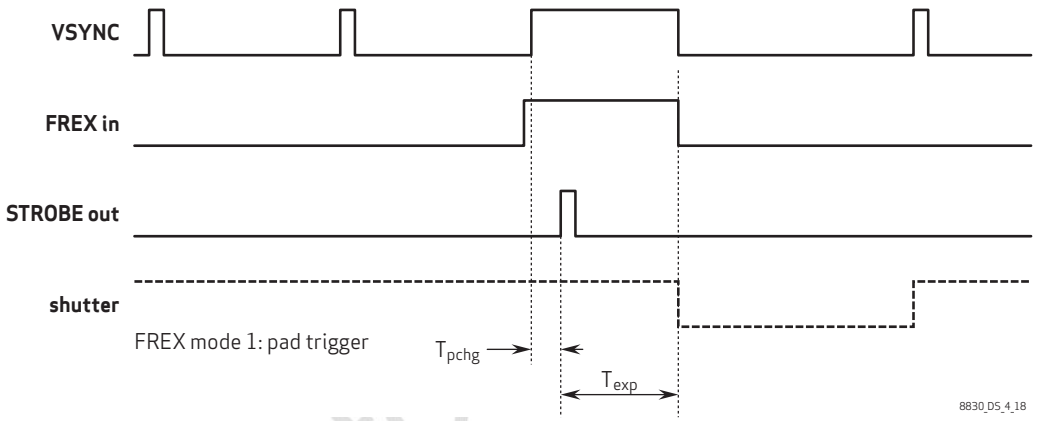
figure 4-10 FREX mode 2



In mode 1, the FREX pin is configured as an input while it is configured as an output in mode 2. In both mode 1 and mode 2, the strobe output is irrelevant with the rolling strobe function. When in rolling shutter mode, the strobe function and this FREX/shutter control function do not work at the same time.

The timing diagram for mode 1 is shown in **figure 4-11**.

figure 4-11 FREX mode 1 timing diagram



In mode 1, the host asserts FREX at any time in preview mode (mechanical shutter is open at this time). The sensor will trigger STROBE to indicate the start of exposure time. Exposure time is calculated from the STROBE rising edge to when the mechanical shutter closes. The host will control when to close the mechanical shutter (shutter delay is handled by the host). The host can re-open the shutter after receiving the entire image data or the next VSYNC signal.

The timing diagrams for mode 2 are shown in **figure 4-12** and **figure 4-13**.

figure 4-12 FREX mode 2 (shutter delay = 0) timing diagram

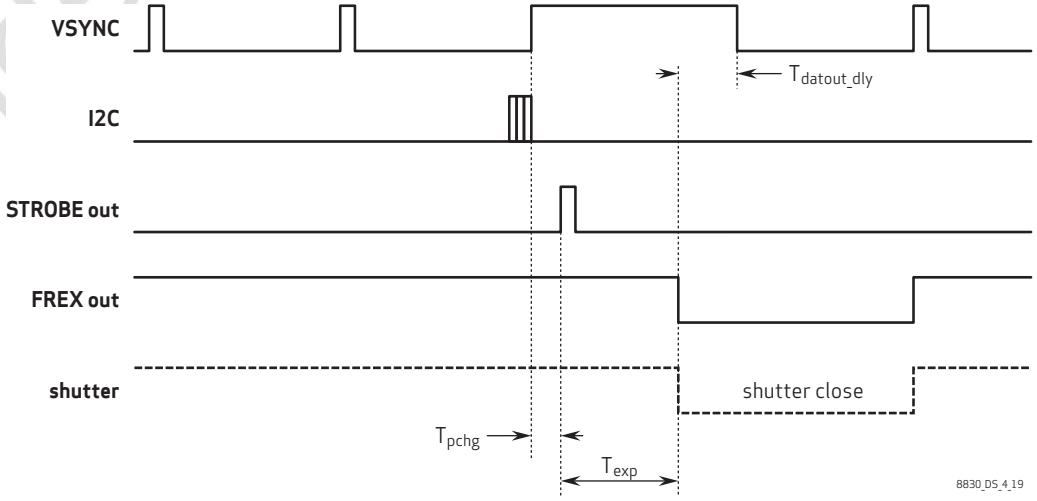


figure 4-13 FREX mode 2 (shutter delay > 0) timing diagram

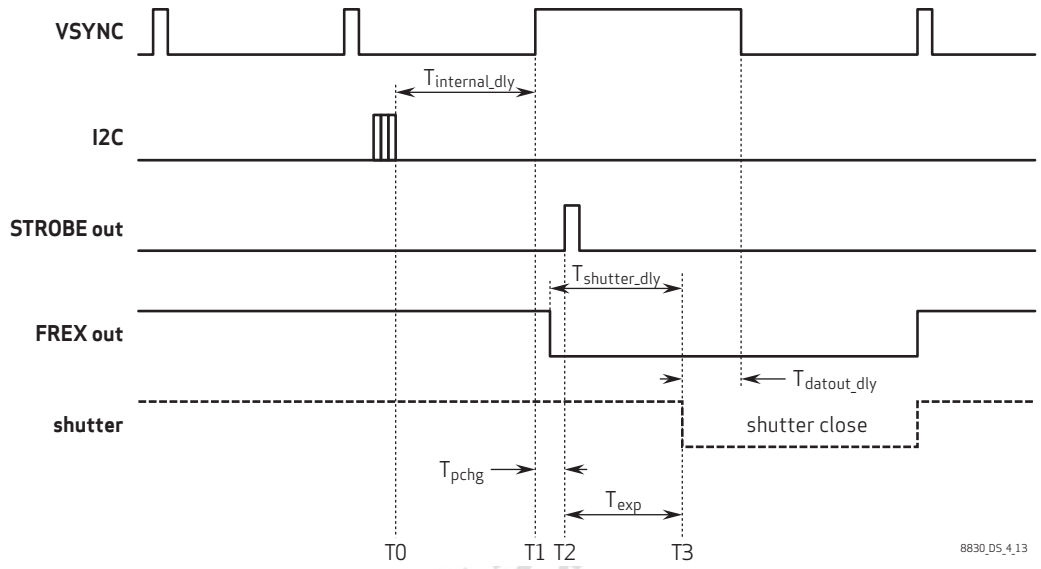
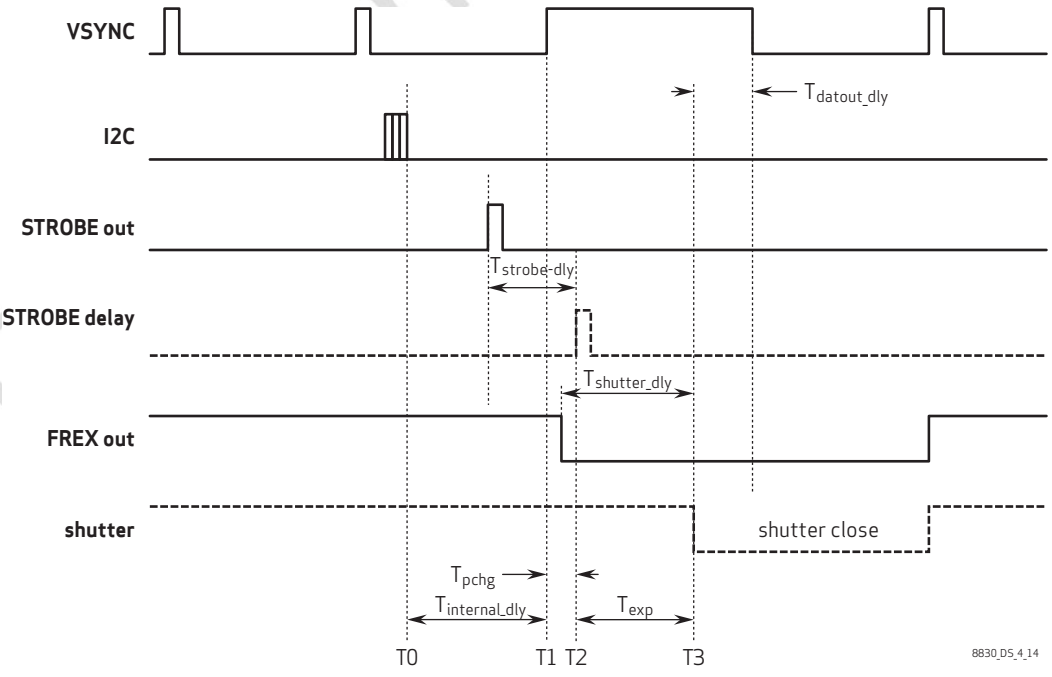


figure 4-14 STROBE delay timing diagram



Before using mode 2, the host needs to program exposure time at sensor frame mode exposure (registers 0x37C5, 0x37C6, 0x37C7)), shutter delay (registers **0x37CC**, **0x37CD**), strobe width (registers 0x37C9, 0x37CA, 0x37CB), strobe delay (registers 0x37D2, 0x37D3), and data output delay (registers 0x37D0, 0x37D1). The host triggers this mode by I2C at any time in preview mode (mechanical shutter is open at this time). The sensor can either start frame exposure right away (since the current data packet is broken, the receiver may get a packet error) or wait for the current frame to finish (controlled by register). If there is no STROBE delay, the sensor will trigger STROBE to indicate the start of exposure time. Exposure time is calculated from STROBE rising edge to when the mechanical shutter closes. Otherwise, the STROBE signal will be sent out even before the sensor begins to pre-charge. The host can control the sensor to start sending image data after a certain delay (registers **0x37D0**, **0x37D1**) after FREX goes low. The host can re-open the shutter after receiving the entire image data or the next VSYNC signal.

See **table 4-9** for LED strobe control functions and **table 4-10** for FREX strobe control functions.

table 4-9 LED strobe control registers

address	register name	default value	R/W	description
0x3B00	STROBE CTRL00	0x00	RW	Bit[7]: Strobe request ON/OFF Bit[6]: Strobe polarity 0: Active high 1: Active low Bit[5:4]: Pulse width in xenon mode Bit[2:0]: Strobe mode select 000: Xenon 001: LED1 010: LED2 011: LED3 100: LED4
0x3B04	STROBE CTRL04	0x00	RW	Bit[3]: start_point_sel Bit[2]: Strobe repeat enable Bit[0]: Strobe latency 0: Strobe generated at next frame 1: Delay one frame, strobe generated 2 frames later
0x3B05	STROBE CTRL05	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

table 4-10 FREX strobe control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x37C5	SENSOR_FREX_EXP	0x00	RW	Bit[7:0]: frex_exp[23:16] MSB of frame exposure time in mode 2. Exposure time is in units of 256 clock cycles. See **0x37C6** and **0x37C7**.
0x37C6	SENSOR_FREX_EXP	0x00	RW	Bit[7:0]: frex_exp[15:8] Middle byte of frame exposure time in mode 2. See **0x37C5** and **0x37C7**.
0x37C7	SENSOR_FREX_EXP	0x00	RW	Bit[7:0]: frex_exp[7:0] LSB of frame exposure time in mode 2. See **0x37C5** and **0x37C6**.
0x37C9	SENSOR_STROBE_ WIDTH	0x00	RW	Bit[3:0]: strobe_width[19:16] MSB of strobe width in mode 2. Strobe width is in units of 2 clock cycles. See registers **0x37CA** and **0x37CB**.
0x37CA	SENSOR_STROBE_ WIDTH	0x00	RW	Bit[7:0]: strobe_width[15:8] Middle byte of strobe width in mode 2. See registers **0x37C9** and **0x37CB**.
0x37CB	SENSOR_STROBE_ WIDTH	0x00	RW	Bit[7:0]: strobe_width[7:0] LSB of strobe width in mode 2. See registers **0x37C9** and **0x37CA**.
0x37CC	SENSOR_SHUTTER_ DLY	0x00	RW	Bit[4:0]: shutter_dly[12:8] MSB of shutter delay in mode 2. Shutter delay is in units of 256 clock cycles. See register **0x37CD**.
0x37CD	SENSOR_SHUTTER_ DLY	0x00	RW	Bit[7:0]: shutter_dly[7:0] LSB of shutter delay in mode 2. Shutter delay is in units of 256 clock cycles. See register **0x37CC**.
0x37CE	SENSOR_FREX_ PCHG_WIDTH	0x01	RW	Bit[7:0]: frex_pchg_width[15:8] MSB of sensor precharge in mode 2. Sensor precharge is in units of 2 system clock cycles (see **section 2.10.1**). See register **0x37CF**.
0x37CF	SENSOR_FREX_ PCHG_WIDTH	0x00	RW	Bit[7:0]: frex_pchg_width[7:0] LSB of sensor precharge in mode 2. Sensor precharge is in units of 2 system clock cycles (see **section 2.10.1**). See register **0x37CE**.

table 4-10 FREX strobe control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x37D0	SENSOR_DATOUT_DLY	0x00	RW	Bit[7:0]: datout_dly[15:8] LSB of readout delay time in mode 2. Readout delay time is in units of 256 clock cycles. See register **0x37D1**.
0x37D1	SENSOR_DATOUT_DLY	0x00	RW	Bit[7:0]: datout_dly[7:0] LSB of readout delay time in mode 2. Readout delay time is in units of 256 clock cycles. See register **0x37D0**.
0x37D2	SENSOR_STROBE_DLY	0x00	RW	Bit[4:0]: sensor_strobe_dly[12:8] MSB of strobe delay in mode 2. Shutter delay is in units of 256 clock cycles.
0x37D3	SENSOR_STROBE_DLY	0x00	RW	Bit[7:0]: sensor_strobe_dly[7:0] LSB of strobe delay in mode 2. Shutter delay is in units of 256 clock cycles.
0x37DE	SENSOR_FREX_1E	0x00	RW	Bit[6]: frex_i2c_req_repeat_trig_sel 0: SOF 1: EOF
0x37DF	SENSOR_STROBE_DLY	0x00	RW	Bit[7]: frex_i2c_req (self clearing) Bit[6]: frex_i2c_req_repeat (debug) Bit[5]: frex_strobe_out_sel Bit[4]: frex_nopchg Bit[3]: frex_strobe polarity Bit[2]: frex_shutter polarity Bit[1]: frex_i from pad in Bit[0]: no_latch at SOF for frex_i2c_req

4.7.2.1 exposure time control

Registers: r_frame_exp = {0x37C5, 0x37C6, 0x37C7}, 24 bits, 1 step = 256 clock cycles.

Minimum exposure time: 0x37C5 = 0x00, 0x37C6 = 0x00, 0x37C7 = 0x00.

If OV8830 works at 216 MHz, the minimum exposure time is 0 and minimum step is 1.2 µs.

Maximum exposure time: 0x37C5 = 0xFF, 0x37C6 = 0xFF, 0x37C7 = 0xFF.

If OV8830 works at 216 MHz, the maximum exposure time is 19.9 sec.

4.7.2.2 shutter delay control

Registers: r_shutter_dly = {0x37CC[4:0], 0x37CD[7:0]}, 13 bits, 1 step = 256 clock cycles.

Minimum shutter delay time: 0x37CC = 0x00, 0x37CD = 0x00.

Minimum step is 1.2 µs.

Maximum shutter delay time: 0x37CC = 0x1F, 0x37CD = 0xFF.
If OV8830 works at 216 MHz, the maximum shutter delay time is 9.7 ms.

4.7.2.3 sensor precharge control
Registers: r_frex_pchg = {0x37CE[7:0], 0x37CF[7:0]}, 16 bits, 1 step = 2 system clock cycles (refer to **section 2.10**).

These registers affect sensor performance. It is for internal use and not recommended for customer to change. Time requirement: 10 µs, for example.

4.7.2.4 strobe control
Registers: r_strobe_width = {0x37C9[3:0], 0x37CA[7:0], 0x37CB[7:0]}, 20 bits, 1 step = 2 clock cycles.

These registers control the strobe signal output width.

4.7.2.5 strobe delay control
Registers: r_shutter_dly = {0x37D2[4:0], 0x37D3[7:0]}, 13 bits, 1 step = 256 clock cycles.

Minimum strobe delay time: 0x37D2=0x00, 0x37D3=0x00.

Minimum step is 1.2 µs.

Maximum strobe delay time: 0x37D2=0x1F, 0x37D3=0xFF.

If OV8830 works at 216 MHz, the maximum strobe delay time is 9.7 ms.

4.7.2.6 data output delay control
Registers: r_dataout_dly = {0x37D0[7:0], 0x37D1[7:0]}, 16 bits, 1 step = 256 clock cycles

Minimum strobe delay time: 0x37D0=0x00, 0x37D1=0x01

Minimum step is 1.2 µs

Maximum strobe delay time: 0x37D0=0xFF, 0x37D3=0xFF

If OV8830 works at 216 MHz, the maximum data out delay time is 77.7 ms

Frame mode control timing restrictions:

1. In frame exposure mode, rolling mode exposure time must be set to be zero.
2. STROBE width must be no longer than internal frame mode exposure time.
3. Minimum value in data out delay control register is "1".
4. In external trigger frame mode, the exposure controlled by registers {0x37C5, 0x37C6, 0x37C7} must be no longer than the external FREX pulse width.

5 image sensor processor digital functions

5.1 ISP general controls

The ISP module provides lens correction, defect pixel cancellation, etc.

table 5-1 ISP general control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0x06	RW	Bit[7]: LENC correction enable 0: Disable 1: Enable Bit[2]: Black defect pixel cancellation enable 0: Disable 1: Enable Bit[1]: White defect pixel cancellation enable 0: Disable 1: Enable
0x5001	ISP CTRL01	0x01	RW	Bit[0]: Manual white balance (MWB) enable 0: Disable 1: Enable
0x5002	ISP CTRL02	0x00	RW	Bit[7]: Scale enable 0: Disable 1: Enable
0x5005	ISP CTRL05	0x1C	RW	Bit[4]: MWB bias ON This will subtract the BLC target before MWB gain and add the target back after MWB 0: Disable 1: Enable
0x501F	ISP BYPASS	0x00	RW	Bit[5]: Bypass ISP Bypasses all ISP modules except window and pre-ISP
0x5025	ISP AVG SEL	0x00	RW	Bit[1:0]: Average select 00: Use sensor raw to calculate average data 01: Use the data after LENC to calculate average data 10: Use the data after MWB gain to calculate average data

table 5-1 ISP general control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5041	ISP CTRL 41	0x04	RW	Bit[7]: Scale manually select 0: Disable, scaling is auto enabled when the output size is less than the input size 1: Enable, scaling is manually enabled or disabled, depending on register 0x5041[5] Bit[5]: Manual scale enable 0: Manual scale disable 1: Manual scale enable Bit[4]: Post binning filter enable 0: Disable 1: Enable Bit[2]: Average enable 0: Disable 1: Enable

5.2 LENC

The main purpose of the LENC is to compensate for lens imperfection. According to the area where each pixel is located, the module calculates a gain for the pixel, correcting each pixel with its gain calculated to compensate for the luminance and color distribution due to lens curvature. Also, the LENC supports the subsample function in both 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

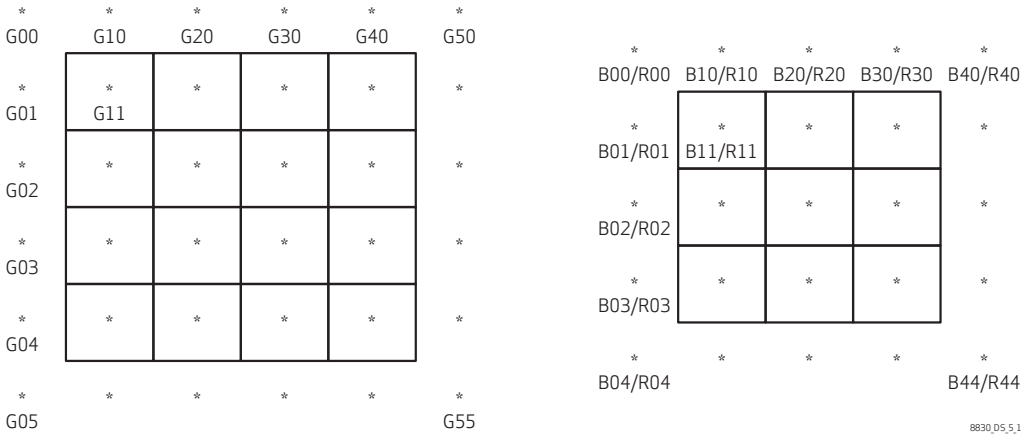


figure 5-2 luminance compensation level calculation

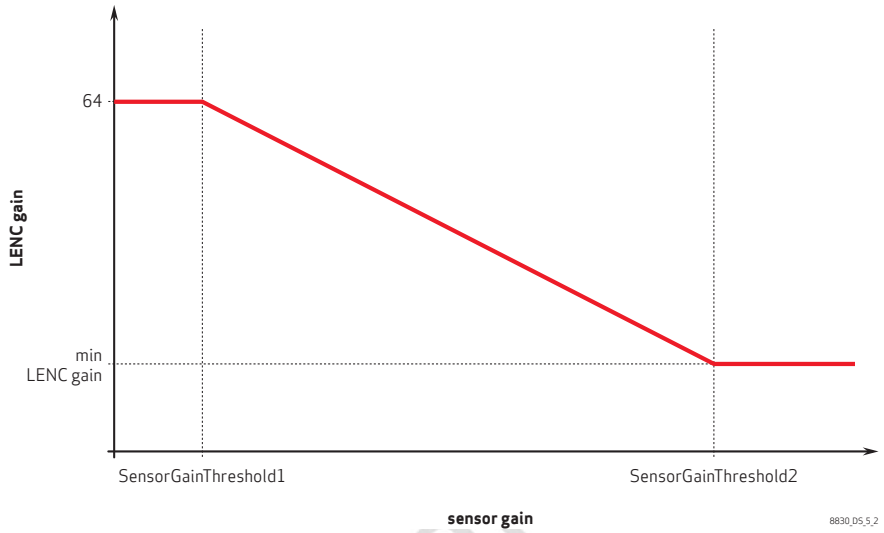


table 5-2 LENC control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0x06	RW	Bit[7]: LENC correction 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
0x5806~ 0x5823	LENC G10~ LENCG55	–	RW	Bit[5:0]: Control point G10~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

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.

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

address	register name	default value	R/W	description
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 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 SENSORGAIN THRESHOLD1	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 SENSORGAIN THRESHOLD2	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	MIN LENC GAIN	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]

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

address	register name	default value	R/W	description
0x5841	LENC CTRL	0x00	RW	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[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	For horizontal color gain calculation, this value indicates the step between two connected horizontal pixels. Bit[2:0]: br_Hscale[10:8]
0x5843	LENC BRHSCALE	0x00	RW	Bit[7:0]: br_Hscale[7:0]
0x5844	LENC BRVSCALE	0x00	RW	For vertical color gain calculation, this value indicates the step between two connected vertical pixels. Bit[2:0]: br_Vscale[10:8]
0x5845	LENC BRVSCALE	0x00	RW	Bit[7:0]: br_Vscale[7:0]
0x5846	LENC GHSCALE	0x00	RW	For horizontal luminance gain calculation, this value indicates the step between two connected horizontal pixels. Bit[2:0]: g_Hscale[10:8]
0x5847	LENC GHSCALE	0x00	RW	Bit[7:0]: g_Hscale[7:0]
0x5848	LENC GVSCALE	0x00	RW	For vertical luminance gain calculation, this value indicates the step between two connected horizontal pixels. Bit[2:0]: g_Vscale[10:8]
0x5849	LENC GVSCALE	0x00	RW	Bit[7:0]: g_Vscale[7:0]

5.3 defect pixel cancellation (DPC)

Due to processes and other reasons, pixel defects in the sensor array will occur. Thus, these bad or wounded pixels will generate wrong color values. The main purpose of Defect Pixel Cancellation (DPC) function is to remove the effect caused by these bad or wounded pixels. To remove the defect pixel effect correctly, the proper threshold should first be determined.

Most of the sensor contains defect pixels due to pixel processing, hardware design, etc. The value of the defect pixel usually has an abrupt change compared to normal pixels which is sensitive to the human eye. The DPC module is designed to recover these white or black pixels while maintaining image quality. Without registering the position of these defect pixels, the DPC algorithm will inevitably remove some of the image details, which is similar to defects in a size-limited window.

table 5-3 DPC registers

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0x06	RW	Bit[2]: Remove black defect pixel 0: Disable 1: Enable Bit[1]: Remove white defect pixel 0: Disable 1: Enable

5.4 scalar

The OV8830 includes a scalar function that allows the user to arbitrarily set an output image size (width and height) that is smaller than the designated array size. The scalar module outputs the specified image size and maintains the field-of-view as the input image to the scalar. Note that the frame rate will not change in scaling mode.

figure 5-3 scaling function

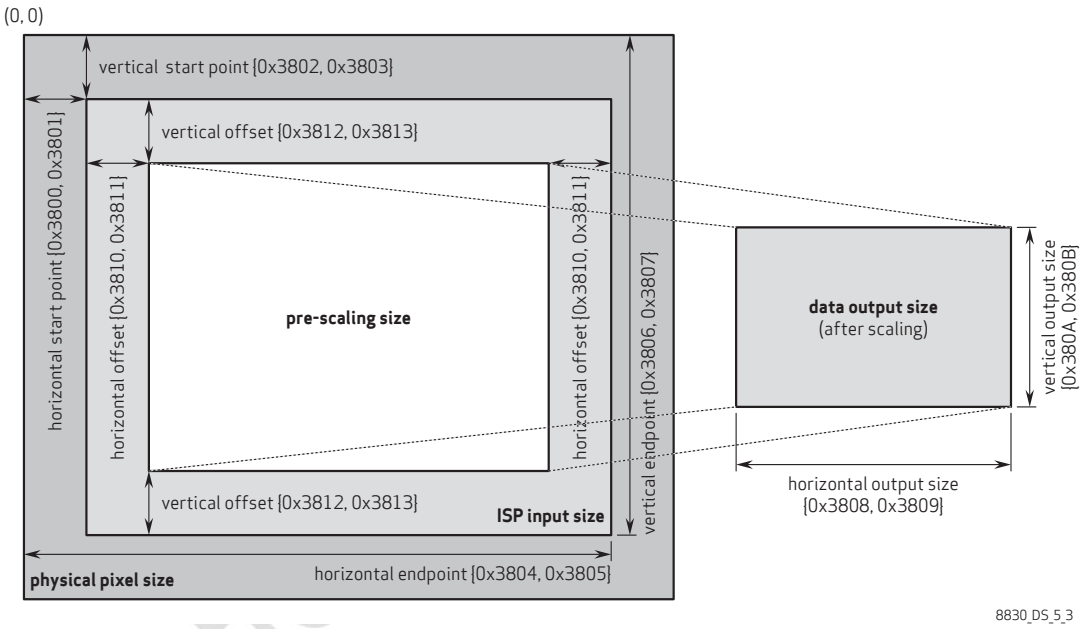


table 5-4 scalar control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5041	ISP CTRL41	0x04	RW	Bit[7]: Scale manually select 0: Disable, scaling is auto enabled when the output size is less than the input size 1: Enable, scaling is manually enabled or disabled, depending on register 0x5041[5] Bit[5]: Manual scale enable 0: Scale disable 1: Scale enable
0x5600	SCALE HFACTOR	0x00	RW	Bit[1:0]: Scale horizontal factor[9:8]
0x5601	SCALE HFACTOR	0x80	RW	Bit[7:0]: Scale horizontal factor[7:0]

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

address	register name	default value	R/W	description
0x5602	SCALE VFACTOR	0x00	RW	Bit[1:0]: Scale vertical factor[9:8]
0x5603	SCALE VFACTOR	0x80	RW	Bit[7:0]: Scale vertical factor[7:0]
0x5068	SCALE H INVT	0x00	RW	Bit[7:6]: Horizontal MSB Bit[4:0]: Horizontal inverse
0x506A	SCALE V INVT	0x00	RW	Bit[7:6]: Vertical MSB Bit[4:0]: Vertical inverse

5.5 MWB

The Manual White Balance (MWB) provides digital gain for R, G, and B channels. Each channel gain is 12-bit. 0x400 is 1x gain.

table 5-5 MWB control registers

address	register name	default value	R/W	description
0x3400	MWB GAIN00	0x04	RW	Bit[3:0]: MWB red gain[11:8] Digital gain in red channel Red gain = MWB red gain[11:0] / 0x400
0x3401	MWB GAIN01	0x00	RW	Bit[7:0]: MWB red gain[7:0] Digital gain in red channel Red gain = MWB red gain[11:0] / 0x400
0x3402	MWB GAIN02	0x04	RW	Bit[3:0]: MWB green gain[11:8] Digital gain in green channel Green gain = MWB green gain[11:0] / 0x400
0x3403	MWB GAIN03	0x00	RW	Bit[7:0]: MWB green gain[7:0] Digital gain in green channel Green gain = MWB green gain[11:0] / 0x400
0x3404	MWB GAIN04	0x04	RW	Bit[3:0]: MWB blue gain[11:8] Digital gain in blue channel Blue gain = MWB blue gain[11:0] / 0x400
0x3405	MWB GAIN05	0x00	RW	Bit[7:0]: MWB blue gain[7:0] Digital gain in blue channel Blue gain = MWB blue gain[11:0] / 0x400
0x3406	MWB CTRL	0x00	RW	Bit[0]: MWB enable

5.6 average (AVG)

The main function of AVG is to average the data channel value using special filters.

table 5-6 AVG control registers

address	register name	default value	R/W	description
0x5041	ISP CTRL41	0x04	RW	Bit[2]: Average enable 0: Disable 1: Enable
0x5680	AVG X START	0x00	RW	Bit[3:0]: X start offset[11:8]
0x5681	AVG X START	0x00	RW	Bit[7:0]: X start offset[7:0]
0x5682	AVG Y START	0x00	RW	Bit[3:0]: Y start offset[11:8]
0x5683	AVG Y START	0x00	RW	Bit[7:0]: Y start offset[7:0]
0x5684	AVG WIN WIDTH	0x10	RW	Bit[3:0]: Window width[11:8]
0x5685	AVG WIN WIDTH	0xA0	RW	Bit[7:0]: Window width[7:0]
0x5686	AVG WIN HEIGHT	0x0C	RW	Bit[3:0]: Window height[11:8]
0x5687	AVG WIN HEIGHT	0x78	RW	Bit[7:0]: Window height[7:0]
0x5688	AVG MANUAL CTRL	0x02	RW	Bit[0]: Average size manual 0: Disable, use ISP pre-scale 1: Enable, use registers 0x5680~0x5687
0x568A	AVG READOUT	–	R	Bit[7:0]: Average readout

6 system control

System control registers include clock, reset control, and PLL configuration. Individual modules can be reset or clock gated by setting the appropriate registers. For system control registers, see **table 7-1**. For PLL configuration registers, see **table 7-2**.

6.1 mobile industry processor interface (MIPI)

MIPI provides a single uni-directional clock lane and 1/2/4 bi-directional data lane solution for communication links between components inside a mobile device. Each data lane has full support for HS (uni-direction). Contact your local OmniVision FAE for more details.

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

address	register name	default value	R/W	description
0x4800	MIPI CTRL00	0x04	RW	MIPI Control 00 Bit[7]: mipi_hs_only 1: MIPI is always in high speed mode Bit[5]: Clock lane gate enable 0: Clock lane is free running 1: Gate clock lane when no packet to transmit Bit[4]: Line sync enable 0: Do not send line short packet for each line 1: Send line short packet for each line Bit[2]: Idle status 0: MIPI bus will be LP00 when there is no packet to transmit 1: MIPI bus will be LP11 when there is no packet to transmit

table 6-1 MIPI registers (sheet 2 of 9)

address	register name	default value	R/W	description
0x4801	MIPI CTRL01	0x0F	RW	MIPI Control 01 Bit[7]: Long packet data type manual enable 0: Use mipi_dt 1: Use dt_man_o as long packet data (see register **0x4814**[5:0]) Bit[6]: Short packet data type manual enable 1: Use dt_spkt as short packet data (see register **0x4815**[5:0]) Bit[5]: first_bit Change first bit of ck_lane 0: Output 0x05 1: Output 0xAA Bit[4]: PH bit order for ECC 0: (DI[7:0],WC[7:0],WC[15:8]) 1: (DI[0:7],WC[0:7],WC[8:15]) Bit[3]: PH byte order for ECC 0: (DI,WC L,WC H) 1: (DI,WC H,WC L) Bit[2]: PH byte order2 for ECC 0: (DI,WC) 1: (WC,DI) Bit[1]: LPX select for pclk domain 0: Auto calculate t_lpx_p, unit: pclk2x cycle 1: Use lpx_p_min[7:0]

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

address	register name	default value	R/W	description
0x4802	MIPI CTRL02	0x00	RW	MIPI Control 02 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 CTRL 03	0x50	RW	MIPI Control 03 Bit[3]: manu_ofset_o t_period manual offset Bit[2]: r_manual_half2one t_period half to 1
0x4804	MIPI CTRL 04	0x8D	RW	MIPI Control 04 Bit[5]: PRBS enable Bit[4]: Lane number manual enable Bit[3:0]: Manual lane number
0x4805	MIPI CTRL 05	0x10	RW	MIPI Control 05 Bit[5]: mipi_ul_tx_en Bit[4]: tx_lsb_first Bit[3:0]: sw_t_lpx

table 6-1 MIPI registers (sheet 4 of 9)

address	register name	default value	R/W	description
0x4806	MIPI REG R/W CTRL	0x28	RW	Bit[3]: lpda_retim_manu_o Bit[2]: lpda_retim_sel_o 1: Manual Bit[1]: lpck_retim_manu_o Bit[0]: lpck_retim_sel_o 1: Manual
0x4810	MIPI MAX FRAME COUNT	0xFF	RW	High Byte of Max Frame Count of Frame Sync Short Packet
0x4811	MIPI MAX FRAME COUNT	0xFF	RW	Low Byte of Max Frame Count of Frame Sync Short Packet
0x4813	MIPI CTRL13	0x00	RW	mipi_ctrl13 RW Bit[2]: vc_sel input vc or reg vc Bit[1:0]: vc Virtual channel of MIPI
0x4814	MIPI CTRL14	0x2A	RW	MIPI Control 14 Bit[5:0]: Data type in manual mode
0x4815	MIPI_DT_SPKT	0x00	RW	Bit[6]: pclk_div 0: Use falling edge of mipi_pclk_o to generate MIPI bus to PHY 1: Use rising edge of mipi_pclk_o to generate MIPI bus to PHY Bit[5:0]: Manual Data Type for short packet
0x4816	EMB_DT_SEL	0x52	RW	emb_dt_sel RW Bit[6]: emb_line_sel 1: Use emb_dt as dt in first emb_line_nu Bit[5:0]: emb_dt Manually set emb data type
0x4818	HS_ZERO_MIN	0x00	RW	High Byte of the Minimum Value for hs_zero, unit ns
0x4819	HS_ZERO_MIN	0x70	RW	Low Byte of the Minimum Value for hs_zero, unit ns hs_zero_real = hs_zero_min_o + Tui*ui_hs_zero_min_o
0x481A	HS_TRAIL_MIN	0x00	RW	High Byte of the Minimum Value for hs_trail, unit ns
0x481B	HS_TRAIL_MIN	0x3C	RW	Low Byte of the Minimum Value for hs_trail, unit ns hs_trail_real = hs_trail_min_o + Tui*ui_hs_trail_min_o
0x481C	CLK_ZERO_MIN	0x01	RW	High Byte of the Minimum Value for clk_zero, unit ns

table 6-1 MIPI registers (sheet 5 of 9)

address	register name	default value	R/W	description
0x481D	CLK_ZERO_MIN	0x06	RW	Low Byte of the Minimum Value for clk_zero, unit ns clk_zero_real = clk_zero_min_o + Tui*ui_clk_zero_min_o
0x481E	CLK_PREPARE_ MAX	0x5F	RW	clk_prepare_max RW Max value of clk_prepare, unit ns
0x481F	CLK_PREPARE_ MIN	0x26	RW	Minimum Value for clk_prepare clk_prepare_real = clk_prepare_min_o + Tui*ui_clk_prepare_min_o
0x4820	CLK_POST_MIN	0x00	RW	High Byte of the Minimum Value for clk_post, unit ns Bit[1:0]: clk_post_min[9:8]
0x4821	CLK_POST_MIN	0x3C	RW	Low Byte of the Minimum Value for clk_post clk_post_real = clk_post_min_o + Tui*ui_clk_post_min_o Bit[7:0]: clk_post_min[7:0]
0x4822	CLK_TRAIL_MIN	0x00	RW	High Byte of the Minimum Value for clk_trail, unit ns Bit[1:0]: clk_trail_min[9:8]
0x4823	CLK_TRAIL_MIN	0x3C	RW	Low Byte of the Minimum Value for clk_trail clk_trail_real = clk_trail_min_o + Tui*ui_clk_trail_min_o Bit[7:0]: clk_trail_min[7:0]
0x4824	LPX_P_MIN	0x00	RW	High Byte of the Minimum Value for lpx_p, unit ns Bit[1:0]: lpx_p_min[9:8]
0x4825	LPX_P_MIN	0x32	RW	Low Byte of the Minimum Value for lpx_p lpx_p_real = lpx_p_min_o + Tui*ui_lpx_p_min_o Bit[7:0]: lpx_p_min[7:0]
0x4826	HS_PREPARE_MIN	0x28	RW	hs_prepare_min RW minimum value of hs_prepare, unit ns
0x4827	HS_PREPARE_MAX	0x55	RW	Max Value for hs_prepare hs_prepare_real = hs_prepare_min_o + Tui*ui_hs_prepare_min_o
0x4828	HS_EXIT_MIN	0x00	RW	High Byte of the Minimum Value for hs_exit, unit ns Bit[1:0]: hs_exit_min[9:8]
0x4829	HS_EXIT_MIN	0x64	RW	Low Byte of the Minimum Value for hs_exit hs_exit_real = hs_exit_min_o + Tui*ui_hs_exit_min_o Bit[7:0]: hs_exit_min[7:0]
0x482A	UI_HS_ZERO_MIN	0x06	RW	Minimum UI Value of hs_zero, unit UI

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

address	register name	default value	R/W	description
0x482B	UI_HS_TRAIL_MIN	0x04	RW	Minimum UI Value of hs_trail, unit UI
0x482C	UI_CLK_ZERO_MIN	0x00	RW	Minimum UI Value of clk_zero, unit UI
0x482D	UI_CLK_PREPARE	0x00	RW	ui_clk_prepare_min_ctrl RW Bit[7:4]: ui_clk_prepare_max Bit[3:0]: ui_clk_prepare_min Value of clk_prepare, unit UI
0x482E	UI_CLK_POST_MIN	0x34	RW	Minimum UI Value of clk_post, unit UI
0x482F	UI_CLK_TRAIL_MIN	0x00	RW	Minimum UI Value of clk_trail, unit UI
0x4830	UI_LPX_P_MIN	0x00	RW	Minimum UI Value of lpx_p, unit UI
0x4831	UI_HS_PREPARE_MIN	0x64	RW	ui_hs_prepare RW Bit[7:4]: ui_hs_prepare_max Bit[3:0]: ui_hs_prepare_min UI value of hs_prepare, unit UI
0x4832	UI_HS_EXIT_MIN	0x00	RW	Minimum UI Value of hs_exit, unit UI
0x4836	GLB_MODE_SEL	0x00	RW	glb_mode_sel Bit[0]: smia_cal_en 0: Use period to calculate 1: Use SMIA bit rate to calculate
0x4837	PCLK_PERIOD	0x0A	RW	Period of pclk2x, pclk_div = 1, and 1-bit decimal
0x4838	MIPI_LP_GPIO0	0x00	RW	Bit[7]: lp_sel0 0: auto gen 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 gen 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

table 6-1 MIPI registers (sheet 7 of 9)

address	register name	default value	R/W	description
0x4839	MIPI_LP_GPIO1	0x00	RW	Bit[7]: lp_sel2 0: Auto gen 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 gen mipi_lp_dir3_o 1: Use lp_dir_man3 to be mipi_lp_dir3_o Bit[2]: lp_dir_man3 0: Input 1: Output Bit[1]: lp_p3_o Bit[0]: lp_n3_o
0x483A	MIPI_LP_GPIO2	0x00	RW	Bit[7]: lp_sel4 0: Auto gen mipi_lp_dir4_o 1: Use lp_dir_man4 to be mipi_lp_dir4_o Bit[6]: lp_dir_man4 0: Input 1: Output Bit[5]: lp_p4_o Bit[4]: lp_n4_o Bit[3]: lp_sel5 0: Auto gen mipi_lp_dir5_o 1: Use lp_dir_man5 to be mipi_lp_dir5_o Bit[2]: lp_dir_man5 0: Input 1: Output Bit[1]: lp_p5_o Bit[0]: lp_n5_o

table 6-1 MIPI registers (sheet 8 of 9)

address	register name	default value	R/W	description
0x483B	MIPI_LP_GPIO3	0x00	RW	Bit[7]: lp_sel6 0: Auto gen mipi_lp_dir6_o 1: Use lp_dir_man6 to be mipi_lp_dir6_o Bit[6]: lp_dir_man6 0: Input 1: Output Bit[5]: lp_p6_o Bit[4]: lp_n6_o Bit[3]: lp_sel7 0: Auto gen mipi_lp_dir7_o 1: Use lp_dir_man7 to be mipi_lp_dir7_o Bit[2]: lp_dir_man7 0: Input 1: Output Bit[1]: lp_p7_o Bit[0]: lp_n7_o
0x483C	MIPI_CTRL3C	0x42	RW	Bit[3:0]: t_clk_pre Unit: pclk2x cycle
0x483D	MIPI_LP_GPIO4	0x00	RW	Bit[7]: lp_ck_sel0 0: Auto gen 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 gen 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
0x4840	START_OFFSET	0x00	RW	Bit[4:0]: start_offset[12:8]
0x4841	START_OFFSET	0x00	RW	Bit[7:0]: start_offset[7:0]
0x4842	START_MODE	0x01	RW	Bit[1:0]: Delay mode select 00: Delay one line mode 01: Old mode, delay about 100 Tp 10: VHREF manual mode 11: Reserved

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

address	register name	default value	R/W	description
0x484A	SEL_MIPI_CTRL4A	0x07	RW	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[2]: line_st_sel_o 0: LS after HREF 1: LS after fifo_st Bit[1]: clk_start_sel_o 0: Clk starts after SOF 1: Clk starts after reset Bit[0]: sof_sel_o 0: FS after HREF occurs 1: FS after SOF
0x484C	SEL_MIPI_CTRL4C	0x02	RW	Bit[5]: SMIA fcnt_i select Bit[4]: MIPI high speed only test mode enable Bit[3]: Set frame count to inactive mode (keep 0) Bit[2]: Vsub select 0: Valid from behind 1: Valid in front Bit[1:0]: Input data valid (e.g., for YUV420) 01: Valid = 1 10: Valid = 2 11: Valid = 3
0x484D	TEST_PATTEN_DATA	0xB6	RW	Data Lane Test Pattern
0x484E	FE_DLY	0x10	RW	Last Packet to Frame End Delay / 2
0x484F	TEST_PATTEN_CK_DATA	0x55	RW	clk_test_patten_reg
0x4864	MIPI_LCNT	–	R	Bit[7:0]: mipi_lcnt[15:8]
0x4865	MIPI_LCNT	–	R	Bit[7:0]: mipi_lcnt[7:0]
0x4866	T_GLB_TIM_H	–	R	Bit[7]: VHREF ahead of flag Must delay VHREF Bit[6:0]: vhref_delay_h
0x4867	T_GLB_TIM_L	–	R	vhref_delay_l

6.2 low-voltage differential signaling (LVDS)

LVDS is a common differential signaling interface that has low power consumption, minimal EMI, and excellent noise immunity. LVDS supports 1/2/4 lane configurations. The maximum data rate for LVDS is 700 Mbps per lane. In case of multiple lanes, each data lane transmits one pixel serially from MSB to LSB using rising and falling edges of clock. Contact your local OmniVision FAE for more details.

Features include:

- supports 10-bit mode
- supports 1/2/4 lanes mode
- in 2 and 4 lanes mode, supports 4 byte sync code per lane or split in every lane
- supports SAV first or EAV first switching
- supports manual setting sync code (can set different sync code for frame start/end and line start/end)
- supports manual setting dummy data in blanking duration
- supports bit swap or not to fit for different PHY
- supports PCLK inversion

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

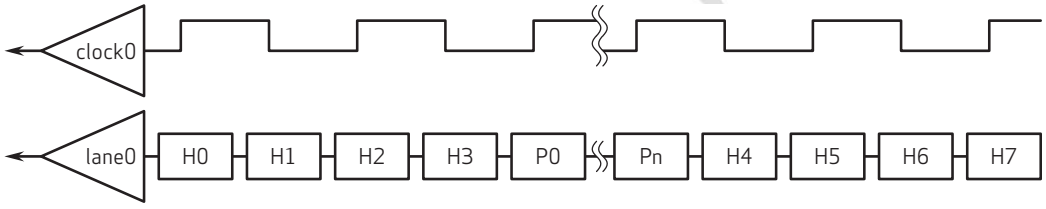
address	register name	default value	RW	description
0x4A00	LVDS_R0 2A	0xAA	RW	Bit[6]: SYNC code manual mode enable Bit[5]: SYNC code enable when only 1 lane Bit[4]: PCLK invert enable Bit[3]: Bit swap default 1: Do swap Bit[2]: F parameter in CCIR656 standard Bit[1]: SAV first enable Bit[0]: SYNC code mode 0: Split 1: Per lane
0x4A02	LVDS_R2 0	0x00	RW	Bit[7:0]: Dummy data0[15:8]
0x4A03	LVDS_R3 80	0x01	RW	Bit[7:0]: Dummy data0[7:0]
0x4A04	LVDS_R4 00	0x00	RW	Bit[7:0]: Dummy data1[15:8]
0x4A05	LVDS_R5 10	0x08	RW	Bit[7:0]: Dummy data1[7:0]
0x4A06	LVDS_R6 00	0xAA	RW	frame_start Sync Code in Manual Sync Code Mode
0x4A07	LVDS_R7 00	0x55	RW	frame_end Sync Code in Manual Sync Code Mode
0x4A08	LVDS_R8 00	0x99	RW	line_start Sync Code in Manual Sync Code Mode
0x4A09	LVDS_R9 00	0x66	RW	line_end Sync Code in Manual Sync Code Mode
0x4A0A	LVDS_RA 00	0x08	RW	Reserved
0x4A0B	LVDS_RB 00	0x00	RW	Reserved

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

address	register name	default value	RW	description
0x4A0C	LVDS_RC 00	0x00	RW	Reserved
0x4A0D	LVDS_RD 00	0x00	RW	Reserved

6.2.1 output modes

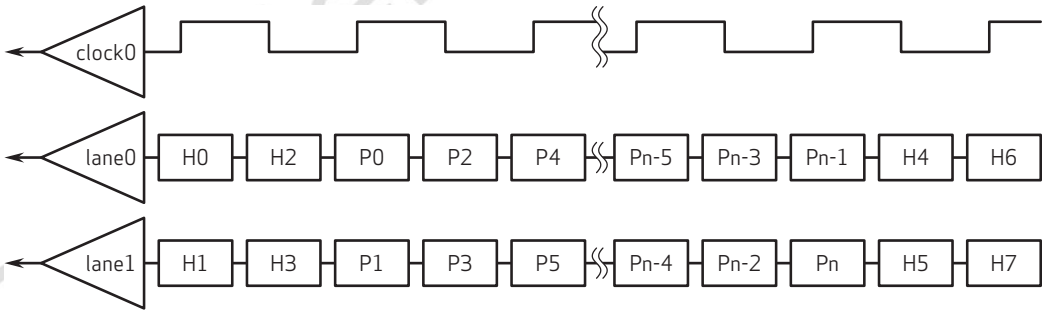
figure 6-1 LVDS 1-lane mode



note H0-H7 = header (sync code) / P0-Pn = pixel

8830_DS_6_1

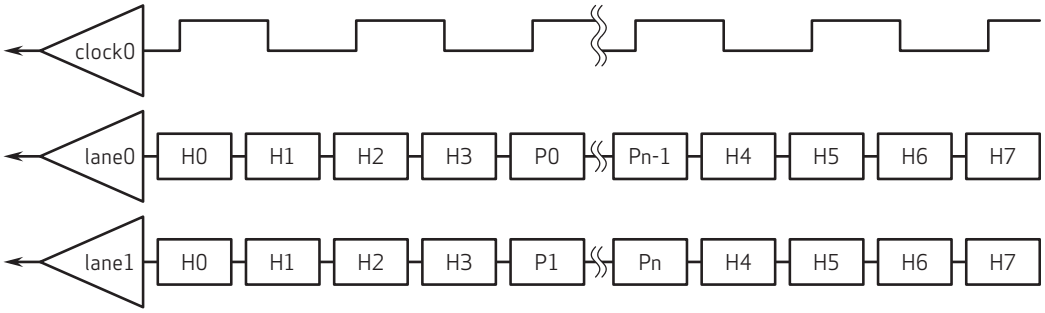
figure 6-2 LVDS 2-lane mode (sync code option 1)



H0-H7 = header (sync code) / P0-Pn = pixel

8830_DS_6_2

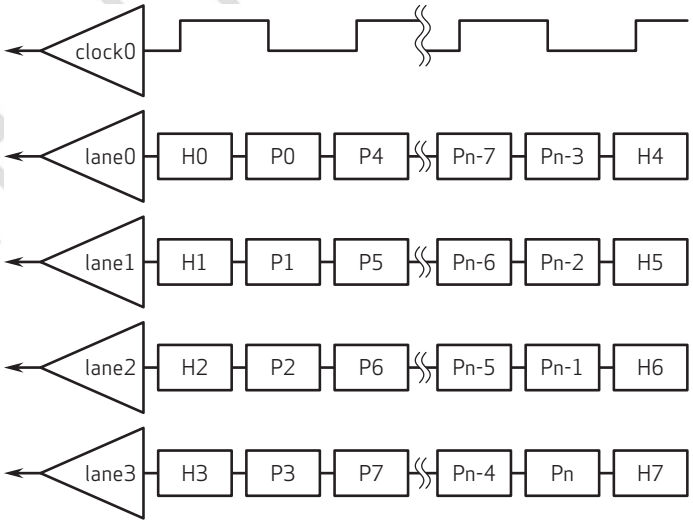
figure 6-3 LVDS 2-lane mode (sync code option 2)



note H0-H7 = header (sync code) / P0-Pn = pixel

8830_DS_6_3

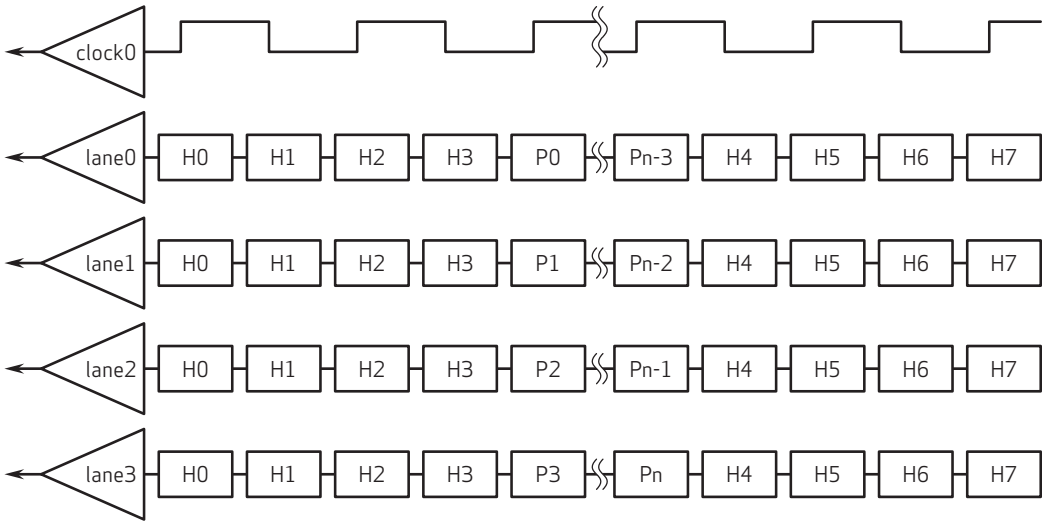
figure 6-4 LVDS 4-lane mode (sync code option 1)



H0-H7 = header (sync code) / P0-Pn = pixel

8830_DS_6_4

figure 6-5 LVDS 4-lane mode (sync code option 2)



6.2.2 PHY specification

figure 6-6 PHY specification diagram

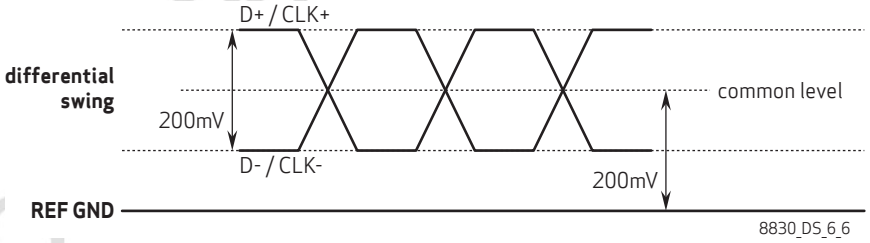


table 6-3 PHY specification table

description	minimum	typical	maximum	units
differential output	150	200	250	mV
common level output	150	200	250	mV
rise/fall time	150		0.3 UI	ps
data-clock skew	-0.15 UI		0.15 UI	ps
impedance	40	50	60	Ω

6.2.3 LVDS lane configuration and sync

figure 6-7 LVDS lane configuration and sync

one lane

lane	FFF	000	000	SAV0	P0	P1	P2	...	Pn	FFF	000	000	EAV0	...

two lanes

lane0	FFF	000	P0	P2	...	Pn-1	FFF	000	...	FFF	000	P0	P2	...
lane1	000	SAV0	P1	P3	...	Pn	000	EAV0	...	000	SAV0	P1	P3	...

four lanes

lane0	FFF	P0	P4	...	Pn-3	FFF	...	FFF	P0	P4	...	Pn-3	FFF	...
lane1	000	P1	P5	...	Pn-2	000	...	000	P1	P5	...	Pn-2	000	...
lane2	000	P2	P6	...	Pn-1	000	...	000	P2	P6	...	Pn-1	000	...
lane3	SAV0	P3	P7	...	Pn	EAV0	...	SAV0	P3	P7	...	Pn	EAV0	...

2 LANE EXAMPLE

SAV ACTIVE (FFF 000 000 800)

lane0	...	FFF	000	...										
lane1	...	000	800	...										

EAV ACTIVE (FFF 000 000 9D0)

lane0	...	FFF	000	...										
lane1	...	000	9D0	...										

SAV BLANKING (FFF 000 000 AB0)

lane0	...	FFF	000	...										
lane1	...	000	AB0	...										

EAV BLANKING (FFF 000 000 B60)

lane0	...	FFF	000	...										
lane1	...	000	B60	...										

8830_DS_6_7

7 register tables

The following tables provide descriptions of the device control registers contained in the OV8830. For all register enable/disable bits, ENABLE = 1 and DISABLE = 0. The device slave addresses are 0x20 for write and 0x21 for read (when SID=1, 0x6C for write and 0x6D for read).

7.1 system control registers [0x3001 - 0x303F]

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

address	register name	default value	R/W	description
0x0100	MODE_SELECT	0x00	RW	Bit[7:1]: Not used Bit[0]: SMIA 0: software_standby 1: Streaming
0x0103	SOFTWARE_RST	0x00	RW	Bit[7:1]: Not used Bit[0]: software_reset
0x3001	SC_PAD_CTRL	0x0A	RW	Bit[7]: pd_output when standby Bit[6:5]: pad_drivability 00: 1x 01: 2x 10: 3x 11: 4x Bit[4:3]: Debug mode Bit[2]: d0_fsin_disable Bit[1]: d1_frex_in_disable Bit[0]: Debug mode
0x3002	SC_PAD_OEN0	0x08	RW	Bit[7]: VSYNC output enable Bit[6]: HREF output enable Bit[5]: PCLK output enable Bit[4]: FREX output enable Bit[3]: Strobe output enable Bit[2]: SDA output enable Bit[1:0]: Debug mode
0x3003	NOT USED	–	–	Not Used
0x3004~ 0x3008	DEBUG	–	–	Debug Registers
0x3009	SC_PAD_OUT0	0x00	RW	Bit[7]: VSYNC output value Bit[6]: HREF output value Bit[5]: PCLK output value Bit[4]: FREX output value Bit[3]: Strobe output value Bit[2]: SDA output value Bit[1:0]: Debug mode

table 7-1 system control registers (sheet 2 of 6)

address	register name	default value	R/W	description
0x300A	SC_CHIP_ID_BK	–	R	Chip ID
0x300B	SC_CHIP_ID_BK	–	R	Chip ID
0x300C	SC_SCCB_ID	0x20	RW	SCCB ID
0x300D~ 0x300E	NOT USED	–	–	Not Used
0x300F	SC_PLL_CTRL0	0x13	RW	Bit[7]: Not used Bit[6:4]: CP for PLL1 Bit[3:0]: Debug mode
0x3010	SC_PLL_CTRL1	0x31	RW	Bit[7]: scale_div_man_en Bit[6]: mipi_div_man_en Bit[5:4]: mipi_div Bit[3:0]: pclk_scale_div
0x3011	SC_MIPI_SC_CTRL0	0x41	RW	Bit[7:4]: lane_num 0001: 1 lane 0010: 2 lane 0100: 4 lane Others:Not used Bit[3]: mipi_phy_rst_o Bit[2]: r_phy_pd_mipi 1: Power down PHY HS TX Bit[1]: r_phy_pd_lprx 1: Power down PHY LP RX module Bit[0]: mipi_en 0: DVP enable 1: MIPI enable
0x3012	SC_MIPI_PHY	0x08	RW	Bit[7:4]: Debug mode Bit[3]: mipi_pad Bit[2]: pgm_bp_hs_en_lat Bypass the latch of hs_enable Bit[1:0]: Debug mode
0x3013	SC_MIPI_PHY	0x10	RW	Bit[7:6]: pgm_vcm[1:0] High speed common mode voltage Bit[5:4]: pgm_lptx[1:0] Driving strength of low speed transmitter, 01 Bit[3]: ihalf Bit[2:1]: Debug mode Bit[0]: predrv_sw
0x3014	SC_MIPI_SC_CTRL1	0x00	RW	Bit[7:6]: mipi_d4_skew Bit[5:4]: mipi_d3_skew Bit[3:2]: mipi_d2_skew Bit[1:0]: mipi_d1_skew

table 7-1 system control registers (sheet 3 of 6)

address	register name	default value	R/W	description
0x3015	SC_MIPI_SC_CTRL2	0x00	RW	Bit[7]: disable_MIPI_lane_4 Bit[6]: disable_MIPI_lane_3 Bit[5]: disable_MIPI_lane_2 Bit[4]: disable_MIPI_lane_1 Bit[3]: phy_mode 0: LVDS 1: MIPI Bit[2]: mipi_ck_lane_dis Bit[1:0]: mipi_ck_skew_o
0x3016	SC_CLKRST0	0xF0	RW	Bit[7]: Enable array clock Bit[6]: Enable strobe control clock Bit[5]: Enable AEC clock Bit[4]: Enable timing control clock Bit[3]: Reset array Bit[2]: Reset strobe control Bit[1]: Reset AEC Bit[0]: Reset timing control
0x3017	SC_CLKRST1	0xF0	RW	Bit[7]: Enable temperature sensor clock Bit[6]: Enable ISP clock Bit[5]: Enable ARB clock Bit[4]: Enable VFIFO clock Bit[3]: Reset temperature sensor Bit[2]: Reset ISP Bit[1]: Reset ARB Bit[0]: Reset VFIFO
0x3018	SC_CLKRST2	0xF0	RW	Bit[7]: pclk_dvp Bit[6]: sclk_mipi Bit[5]: sclk_fc Bit[4]: sclk_otp Bit[3]: rst_dvp Bit[2]: rst_mipi Bit[1]: rst_fc Bit[0]: rst_otp
0x3019	SC_CLKRST3	0xF0	RW	Bit[7]: sclk_blc Bit[6]: sclk_ispfc Bit[5]: sclk_fmt Bit[4]: sclk_embline Bit[3]: rst_blc Bit[2]: rst_ispfc Bit[1]: rst_fmt Bit[0]: rst_embline

table 7-1 system control registers (sheet 4 of 6)

address	register name	default value	R/W	description
0x301A	SC_CLKRST4	0xF0	RW	Bit[7]: sclk_grp Bit[6]: padclk_mipi_sc Bit[5]: pclk_vfifo Bit[4]: pclk_mipi Bit[3]: rst_grp Bit[2]: rst_mipi_sc Bit[1:0]: Debug mode
0x301B	SC_CLKRST5	0xB4	RW	Bit[7:6]: Debug mode Bit[5]: sclk_bist20 Bit[4]: sclk_snr_sync Bit[3:2]: Debug mode Bit[1]: rst_bist20 Bit[0]: rst_snr_sync
0x301C	NOT USED	–	–	Not Used
0x301D	SC_FREX_RST_MASK0	0x02	RW	Bit[7]: frex_mask_aec Bit[6]: frex_mask_blc Bit[5]: frex_mask_isp Bit[4]: frex_mask_dvp Bit[3]: frex_mask_mipi Bit[2]: frex_mask_vfifo Bit[1]: frex_mask_arb Bit[0]: frex_mask_mipi_phy
0x301E	SC_CLOCK_SEL	0x00	RW	Bit[7:5]: sdiv Divider for sigma-delta Bit[4]: sclk_en 0: Use pll_pclk_i for sclk 1: Use pll_sclk_i for sclk Bit[3]: pclk_sel Bit[2:1]: sclk_sel Bit[0]: sclk2x_sel

table 7-1 system control registers (sheet 5 of 6)

address	register name	default value	R/W	description
0x301F	SC_MISC_CTRL	0x03	RW	Bit[7]: Debug mode Bit[6]: mipi_2lane 0: 1 lane 1: 2 lane Bit[5]: mipi_clk_lane_ctrl 0: Clk lane hold lp00 when pd_mipi 1: Clk lane is high z when pd_mipi Bit[4]: mipi_ctr_en 0: Disable the function 1: Enable MIPI remote reset and suspend control sc Bit[3]: mipi_rst_sel 0: MIPI remote reset all registers 1: MIPI remote reset all digital module Bit[2]: gpio_pclk_en Bit[1]: frex_ef_sel Bit[0]: cen_global_o
0x3020	SC_A_PWC_PK_O	0x00	RW	Bit[7:0]: Debug mode
0x3021	SC_A_PWC_PK_O	0x00	RW	Bit[7:6]: Analog debug Bit[5]: Bypass internal DVDD regulator Bit[4:0]: Analog debug
0x3022	SC_A_PWC_PK_O	0x04	RW	Bit[7]: pd_vmrst Bit[6]: pd_dac Bit[5]: pd_ana Bit[4]: pd_sram Bit[3]: Debug mode Bit[2:0]: pd_pixelvdd_sel[2:0]
0x3023	SC_LOW_PWR_CTR	0x00	RW	Bit[6]: phy_pd_mipi_pwdn_dis Bit[5]: phy_pd_lprx_pwdn_dis Bit[4]: stb_rst_dis 0: Reset all block at software standby mode 1: Temperature control, sensor control, and ISP are reset, others not Bit[3]: pd_ana_dis Bit[2]: pd_big_regulator_dis Bit[1]: phy_pd_mipi_slppd_dis Bit[0]: phy_pd_lprx_slppd_dis
0x3024~ 0x3026	DEBUG	–	–	Debug Registers

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

address	register name	default value	R/W	description
0x3027	SC_PAD_SEL0	0x00	RW	Bit[7]: VSYNC output selection Bit[6]: HREF output selection Bit[5]: PCLK output selection Bit[4]: FREX output selection Bit[3]: Strobe output selection Bit[2]: SDA output selection Bit[1:0]: Debug mode
0x3028~ 0x3029	NOT USED	–	–	Not Used
0x302A	SC_CHIP_REVISION	0x00	RW	Chip Revision
0x302B	NOT USED	–	–	Not Used
0x302C	SC_REG2C	0x00	RW	Debug Mode
0x302D~ 0x3030	NOT USED	–	–	Not Used
0x3031~ 0x303F	DEBUG	–	–	Debug Registers

7.2 PLL registers [0x3080 - 0x30B6]

table 7-2 PLL registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3080	PLL_PLL0	0x01	RW	Bit[7:1]: Debug mode Bit[0]: pll1_op_2lane_clk_div
0x3081	PLL_PLL1	0x00	RW	Debug Mode
0x3082	PLL_PLL2	0x01	RW	Bit[2:0]: assign pll1_cp[2:0]
0x3083	PLL_PLL3	0x00	RW	Debug Mode
0x3084	PLL_PLL4	0x01	RW	Debug Mode
0x3085~ 0x308F	NOT USED	–	–	Not Used
0x3090	PLL_PLL10	0x02	RW	Bit[7:3]: Debug mode Bit[2:0]: pll2_prediv[2:0]
0x3091	PLL_PLL11	0x12	RW	Bit[7:6]: Debug mode Bit[5:0]: pll2_multiplier[5:0]
0x3092	PLL_PLL12	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: pll2_divs[3:0]
0x3093	PLL_PLL13	0x00	RW	Bit[7:2]: Debug mode Bit[1:0]: pll2_seld5[1:0] 00: /1 01: /1 10: /2 11: /2.5 Others:/Not allowed
0x3094	PLL_PLL14	0x00	RW	Debug Mode
0x3095	PLL_PLL15	0x00	RW	Debug Mode
0x3096	PLL_PLL16	0x01	RW	Debug Mode
0x3097	PLL_PLL17	0x00	RW	Debug Mode
0x3098	PLL_PLL18	0x03	RW	Bit[7:3]: Debug mode Bit[2:0]: pll3_prediv[2:0] 000: /1 001: /1 010: /2 011: /3 100: /1.5 Others:/Not allowed

table 7-2 PLL registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3099	PLL_PLL19	0x19	RW	Bit[7:5]: Debug mode Bit[4:0]: pll3_multiplier2[4:0] (range from x2~x31)
0x309A	PLL_PLL1A	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: pll3_divs[3:0] Divider=/(1+pll3_divs[3:0]) (range from /1~15)
0x309B	PLL_PLL1B	0x00	RW	Bit[7:2]: Debug mode Bit[1:0]: pll3_div[1:0] 00: /1 01: /1 10: /2 11: /2.5 Others:/Not allowed
0x309C	PLL_PLL1C	0x00	RW	Bit[7:1]: Debug mode Bit[0]: pll3_multiplier1[0] Multiplier1=1+pll3_multiplier1 (range from x1~x2)
0x309D	PLL_PLL1D	0x00	RW	Debug Mode
0x309E	PLL_PLL1E	0x01	RW	Debug Mode
0x309F	PLL_PLL1F	0x00	RW	Debug Mode
0x30A0	PLL_PLL20	0xC8	RW	Debug Mode
0x30A1	PLL_PLL21	0x60	RW	Bit[7:0]: ext_clk_freq_x8[7:0]
0x30A2	DEBUG MODE	0x00	RW	Debug Mode
0x30A3~ 0x30AF	NOT USED	–	–	Not Used
0x30B0	DEBUG MODE	0x05	RW	Debug Mode
0x30B1	DEBUG MODE	0x02	RW	Debug Mode
0x30B2	DEBUG MODE	0x00	RW	Debug Mode
0x30B3	PLL_MULTIPLIER	0x54	RW	Bit[7:0]: pll1_multiplier[7:0]
0x30B4	PLL_PLL1_PRE_ PLL_DIV	0x02	RW	Bit[7:3]: Debug mode Bit[2:0]: pll_prediv[2:0] 001: /1 010: /2 011: /3 100: /4 Others: Not allowed

table 7-2 PLL registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x30B5	PLL_PLL1_OP_ PIX_CLK_DIV	0x04	RW	Bit[7:4]: Debug mode Bit[3:0]: pll1_op_pix_div[3:0] 0x4: /8 0x5: /10 Others: Not allowed
0x30B6	PLL_PLL1_OP _SYS_CLK_DIV	0x01	RW	Bit[7:4]: Debug mode Bit[3:0]: pll1_op_sys_div[3:0] 0001:/1 0010:/2 0100:/4 0110:/6 1000:/8 Others: Not allowed
0x3104	REG3104	0xA1	RW	Bit[7]: sclk_sw2pll2 Bit[6:0]: Debug mode

7.3 group hold registers [0x3200 - 0x320F]

table 7-3 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[3:0], 4'h0}
0x3201	GROUP ADR1	0x10	RW	Group1 Start Address in SRAM Actual address is {0x3201[3:0], 4'h0}
0x3202	GROUP ADR2	0x20	RW	Group2 Start Address in SRAM Actual address is {0x3202[3:0], 4'h0}
0x3203	GROUP ADR3	0x30	RW	Group3 Start Address in SRAM Actual address is {0x3203[3:0], 4'h0}
0x3204	GROUP LEN0	–	R	Length of Group0
0x3205	GROUP LEN1	–	R	Length of Group1
0x3206	GROUP LEN0	–	R	Length of Group2
0x3207	GROUP LEN1	–	R	Length of Group3

table 7-3 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: 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 0x10) 0010: Group bank 2 (default start from address 0x20) 0011: Group bank 3 (default start from address 0x30) Others: Debug mode
0x3209	GROUP0 PERIOD	0x00	RW	Number of Frames to Stay in Group0
0x320A	GROUP1 PERIOD	0x00	RW	Number of Frames to Stay in Group1
0x320B	GRP_SW_CTRL	0x01	RW	Bit[7:6]: Debug mode Bit[5]: grp0_start_opt Bit[4]: frame_cnt_trig Bit[3]: group_switch_repeat_en Enable the first group (group 0) and second group repeatable switch Bit[2]: context_en Enable the switch from second group back to first group (group 0) automatically Bit[1:0]: Second group select
0x320C	SRAM TEST	0x01	RW	Bit[7:5]: Debug mode 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	Group0 Frame Count
0x320F	FM_CNT_GRP1	–	R	Group 1 Frame Count

7.4 MWB control registers [0x3400 - 0x3406]

table 7-4 MWB control registers

address	register name	default value	R/W	description
0x3400	MWB GAIN00	0x04	RW	Bit[7:4]: Debug mode Bit[3:0]: MWB red gain[11:8] Digital gain in red channel Red gain = MWB red gain[11:0] / 0x400
0x3401	MWB GAIN01	0x00	RW	Bit[7:0]: MWB red gain[7:0] Digital gain in red channel Red gain = MWB red gain[11:0] / 0x400
0x3402	MWB GAIN02	0x04	RW	Bit[7:4]: Debug mode Bit[3:0]: MWB green gain[11:8] Digital gain in green channel Green gain = MWB green gain[11:0] / 0x400
0x3403	MWB GAIN03	0x00	RW	Bit[7:0]: MWB green gain[7:0] Digital gain in green channel Green gain = MWB green gain[11:0] / 0x400
0x3404	MWB GAIN04	0x04	RW	Bit[7:4]: Debug mode Bit[3:0]: MWB blue gain[11:8] Digital gain in blue channel Blue gain = MWB blue gain[11:0] / 0x400
0x3405	MWB GAIN05	0x00	RW	Bit[7:0]: MWB blue gain[7:0] Digital gain in blue channel Blue gain = MWB blue gain[11:0] / 0x400
0x3406	MWB CTRL	0x00	RW	Bit[7:1]: Not used Bit[0]: MWB enable

7.5 manual AEC/AGC registers [0x3500 - 0x350B, 0x5041, 0x5680 - 0x568A]

table 7-5 manual AEC/AGC registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3500	AEC LONG EXPO	–	RW	Long Exposure Bit[7:4]: Not used Bit[3:0]: Long Exposure[19:16]
0x3501	AEC LONG EXPO	0x02	RW	Long Exposure Bit[7:0]: Long Exposure[15:8]
0x3502	AEC LONG EXPO	–	RW	Long Exposure Bit[7:0]: Long Exposure[7:0] Low 4 bits are fraction bits
0x3503	AEC MANUAL	–	RW	AEC Manual Mode Control Bit[7:6]: Not used Bit[5]: Gain delay option 0: 1 frame latch 1: Delay 1 frame latch Bit[4]: Choose delay option 0: Delay disable 1: Delay enable Bit[3:2]: Not used Bit[1]: AGC manual enable There is no auto module in this device so this bit should be always be 1 0: Auto enable 1: Manual enable Bit[0]: AEC manual enable There is no auto module in this device so this bit should be always keep 1 0: Auto enable 1: Manual enable
0x3504~ 0x3505	AGC DEBUG	–	RW	Debug Mode
0x3506	AEC SHORT EXPO	–	RW	Short Exposure Bit[7:4]: Not used Bit[3:0]: Short exposure[19:16]
0x3507	AEC SHORT EXPO	0x02	RW	Short Exposure Bit[7:0]: Short exposure[15:8]
0x3508	AEC SHORT EXPO	–	RW	Short Exposure Bit[7:0]: Short exposure[7:0] Low 4 bits are fraction bits

table 7-5 manual AEC/AGC registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3509	AEC GAIN CONVERT	0x10	RW	AEC Manual Mode Control Bit[7:5]: Not used Bit[4]: Sensor gain convert enable 0: Debug mode 1: Use real gain {0x350A,0x350B} as real gain Bit[3:0]: Not used
0x350A	AEC AGC ADJ	–	RW	Gain Output to Sensor Bit[7:3]: Not used Bit[2:0]: Debug mode
0x350B	AEC AGC ADJ	–	RW	Gain Output to Sensor Bit[7:0]: Gain[7:0] This gain is real gain. The low 4 bits are fraction bits. Gain=0x350B/16
0x5041	ISP CTRL41	0x04	RW	Bit[7]: Scale manually select 0: Disable, scale is auto enabled when the output size is less than the input size 1: Enable, scale is manually enabled or disabled, depending on register 0x5041[5] Bit[6]: Not used Bit[5]: Manual scale enable 0: Scale disable 1: Scale enable Bit[4]: Post binning filter enable 0: Disable 1: Enable Bit[3]: Not used Bit[2]: Average enable 0: Disable 1: Enable Bit[1:0]: Not used
0x5680	AVG X START	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: X start offset[11:8]
0x5681	AVG X START	0x00	RW	Bit[7:0]: X start offset[7:0]
0x5682	AVG Y START	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: Y start offset[11:8]
0x5683	AVG Y START	0x00	RW	Bit[7:0]: Y start offset[7:0]
0x5684	AVG WIN WIDTH	0x0C	RW	Bit[7:4]: Debug mode Bit[3:0]: Window width[11:8]
0x5685	AVG WIN WIDTH	0xC0	RW	Bit[7:0]: Window width[7:0]

table 7-5 manual AEC/AGC registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x5686	AVG WIN HEIGHT	0x09	RW	Bit[7:4]: Debug mode Bit[3:0]: Window height[11:8]
0x5687	AVG WIN HEIGHT	0x90	RW	Bit[7:0]: Window height[7:0]
0x5688	AVG MANUAL CTRL	0x02	RW	Bit[7:2]: Debug mode Bit[1]: Average option Bit[0]: Average size manual 0: Disable 1: Enable
0x568A	AVG READOUT	–	R	Bit[7:0]: Average readout

7.6 ADC and analog registers [0x3600 - 0x3684]

table 7-6 ADC and analog registers

address	register name	default value	R/W	description
0x3600~ 0x3684	ANALOG CTRL	–	–	Analog Control Registers

7.7 sensor control registers [0x3700 - 0x377F]

table 7-7 sensor control registers

address	register name	default value	R/W	description
0x3700~ 0x377F	SENSOR TIMING CTRL	–	–	Sensor Timing Control Registers

7.8 PSRAM control registers [0x3780 - 0x37A3]

table 7-8 PSRAM control registers

address	register name	default value	R/W	description
0x3780~ 0x37A3	PSRAM_REG0~ PSRAM_REG23	–	–	PSRAM Control Registers

7.9 FREX control registers [0x37C5 - 0x37DF]

table 7-9 FREX control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x37C5	SENSOR_FREX_EXP	0x00	RW	Bit[7:0]: sensor_frex_exp[23:16] MSB of frame exposure time in mode 2. Exposure time is in units of 256 clock cycles.
0x37C6	SENSOR_FREX_EXP	0x00	RW	Bit[7:0]: sensor_frex_exp[15:8] Middle byte of frame exposure time in mode 2.
0x37C7	SENSOR_FREX_EXP	0x00	RW	Bit[7:0]: sensor_frex_exp[7:0] LSB of frame exposure time in mode 2.
0x37C9	SENSOR_STROBE_ WIDTH	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: sensor_strobe_width[19:16] MSB of strobe width in mode 2 Strobe width is in units of 2 clock cycle
0x37CA	SENSOR_STROBE_ WIDTH	0x00	RW	Bit[7:0]: sensor_strobe_width[15:8] Middle byte of strobe width in mode 2
0x37CB	SENSOR_STROBE_ WIDTH	0x00	RW	Bit[7:0]: sensor_strobe_width[7:0] LSB of strobe width in mode 2
0x37CC	SENSOR_SHUTTER_ DLY	0x00	RW	Bit[7:5]: Debug mode Bit[4:0]: sensor_shutter_dly[12:8] MSB of shutter delay in mode 2 Shutter delay is in units of 256 clock cycles
0x37CD	SENSOR_SHUTTER_ DLY	0x00	RW	Bit[7:0]: sensor_shutter_dly[7:0] LSB of shutter delay in mode 2
0x37CE	SENSOR_FREX_PCHG _WIDTH	0x01	RW	Bit[7:0]: sensor_frex_pchg_width[15:8] MSB of sensor precharge in mode 2. Sensor precharge is in units of 2 clock cycle
0x37CF	SENSOR_FREX_PCHG _WIDTH	0x00	RW	Bit[7:0]: sensor_frex_pchg_width[7:0] LSB of sensor precharge in mode 2
0x37D0	SENSOR_DATOUT_ DLY	0x00	RW	Bit[7:0]: sensor_datout_dly[15:8] MSB of readout delay time in mode 2. Readout delay time is in units of 256 clock cycles

table 7-9 FREX control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x37D1	SENSOR_DATOUT_DLY	0x00	RW	Bit[7:0]: sensor_datout_dly[7:0] LSB of readout delay time in mode 2
0x37D2	SENSOR_STROBE_DLY	0x00	RW	Bit[7:5]: Debug mode Bit[4:0]: sensor_strobe_dly[12:8] MSB of strobe delay time in mode 2. Strobe delay time is in units of 256 clock cycles
0x37D3	SENSOR_STROBE_DLY	0x00	RW	Bit[7:0]: sensor_strobe_dly[7:0] LSB of strobe delay time in mode 2
0x37DE	SENSOR_FREX_1E	0x00	RW	Bit[7:1]: Debug mode Bit[0]: frex_i2c_req_repeat_trig_sel 0: SOF 1: EOF
0x37DF	SENSOR_FREX_REQ	0x00	RW	Bit[7]: frex_exposure mode I2C trigger request, self clear Bit[6]: frex_i2c_req_repeat, debug Bit[5]: Strobe output in frame exposure mode Bit[4]: frex_nopchg Bit[3]: frexmode_strobe polarity control Bit[2]: frexmode_shutter polarity control Bit[1]: Frame exposure mode select 0: Frame exposure mode is trigger by I2C command 1: Frame exposure mode is trigger by external FREX input Bit[0]: no_latch at SOF for frex_i2c_req

7.10 timing control registers [0x3800 - 0x382F]

table 7-10 timing control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3800	TIMING_X_ADDR_START	0x00	RW	Array Horizontal Start Point High Byte
0x3801	TIMING_X_ADDR_START	0x0C	RW	Array Horizontal Start Point Low Byte
0x3802	TIMING_Y_ADDR_START	0x00	RW	Array Vertical Start Point High Byte

table 7-10 timing control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3803	TIMING_Y_ADDR_START	0x0C	RW	Array Vertical Start Point Low Byte
0x3804	TIMING_X_ADDR_END	0x0C	RW	Array Horizontal End Point High Byte
0x3805	TIMING_X_ADDR_END	0xD3	RW	Array Horizontal End Point Low Byte
0x3806	TIMING_Y_ADDR_END	0x09	RW	Array Vertical End Point High Byte
0x3807	TIMING_Y_ADDR_END	0xA3	RW	Array Vertical End Point Low Byte
0x3808	TIMING_X_OUTPUT_SIZE	0x0C	RW	ISP Horizontal Output Width High Byte
0x3809	TIMING_X_OUTPUT_SIZE	0xC0	RW	ISP Horizontal Output Width Low Byte
0x380A	TIMING_Y_OUTPUT_SIZE	0x09	RW	ISP Vertical Output Height High Byte
0x380B	TIMING_Y_OUTPUT_SIZE	0x90	RW	ISP Vertical Output Height Low Byte
0x380C	TIMING_HTS	0x0E	RW	Total Horizontal Timing Size High Byte
0x380D	TIMING_HTS	0x18	RW	Total Horizontal Timing Size Low Byte
0x380E	TIMING_VTS	0x09	RW	Total Vertical Timing Size High Byte
0x380F	TIMING_VTS	0xAC	RW	Total Vertical Timing Size Low Byte
0x3810	TIMING_ISP_X_WIN	0x00	RW	ISP Horizontal Windowing Offset High Byte
0x3811	TIMING_ISP_X_WIN	0x02	RW	ISP Horizontal Windowing Offset Low Byte
0x3812	TIMING_ISP_Y_WIN	0x00	RW	ISP Vertical Windowing Offset High Byte
0x3813	TIMING_ISP_Y_WIN	0x02	RW	ISP Vertical Windowing Offset Low Byte
0x3814	TIMING_X_INC	0x11	RW	Bit[7:4]: x_odd_inc Bit[3:0]: x_even_inc
0x3815	TIMING_Y_INC	0x11	RW	Bit[7:4]: y_odd_inc Bit[3:0]: y_even_inc
0x3816	TIMING_HSYNC_START	0x00	RW	HSYNC Start Point High Byte
0x3817	TIMING_HSYNC_START	0x00	RW	HSYNC Start Point Low Byte
0x3818	TIMING_HSYNC_END	0x00	RW	HSYNC End Point High Byte
0x3819	TIMING_HSYNC_END	0x00	RW	HSYNC End Point Low Byte
0x381A	TIMING_HSYNC_FIRST	0x00	RW	HSYNC First Active Row Start Position High Byte
0x381B	TIMING_HSYNC_FIRST	0x00	RW	HSYNC First Active Row Start Position Low Byte
0x381C~ 0x381F	NOT USED	–	–	Not Used

table 7-10 timing control registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x3820~ 0x3823	TIMING FORMAT CTRL	–	–	Timing Format Control Registers
0x3824~ 0x3827	NOT USED	–	–	Not Used
0x3828~ 0x382D	DEBUG	–	–	Debug Registers
0x382E	TIMING_LINE_CNT	–	R	
0x382F	TIMING_LINE_CNT	–	R	

7.11 strobe control registers [0x3B00 - 0x3B05]

table 7-11 strobe control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3B00	STROBE CTRL	0x00	RW	Bit[7]: Strobe ON/OFF Bit[6]: Strobe polarity 0: Active high 1: Active low Bit[5:4]: width_in_xenon Bit[3]: Debug mode Bit[2:0]: Mode 000: Xenon 001: LED1 010: LED2 011: LED3 100: LED4
0x3B02	STROBE DMY H	0x00	RW	Dummy Lines Added in LED1 and LED2 Mode, MSB
0x3B03	STROBE DMY L	0x00	RW	Dummy Lines Added in LED1 and LED2 Mode, LSB
0x3B04	STROBE CTRL	0x00	RW	Bit[7:4]: Debug mode Bit[3]: start_point_sel Bit[2]: Strobe repeat enable Bit[1]: Not used Bit[0]: Strobe latency 0: Strobe generated at next frame 1: Strobe generated 2 frames later

table 7-11 strobe control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3B05	STROBE WIDTH	0x00	RW	Bit[7:2]: Strobe pulse width step Bit[1:0]: Strobe pulse width gain Strobe_pulse_width = 128*(2*gain)*(step+1)*Tsclk

7.12 illumination PWM control registers [0x3B40 - 0x3B51]

table 7-12 illumination PWM registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3B40	PULSE1 DELAY	0x10	RW	Bit[7:0]: First pulse PWM1 delay (0~31) 0x00: -0.5 frame 0x1F: 0.5 frame
0x3B41	PULSE2 DELAY	0x10	RW	Bit[7:0]: Second PWM2 pulse delay (0~31) 0x00: -0.5 frame 0x1F: 0.5 frame
0x3B42	PULSE3 DELAY	0x10	RW	Bit[7:0]: Third pulse PWM3 delay (0~31) 0x00: -0.5 frame 0x1F: 0.5 frame
0x3B43	PULSE4 DELAY	0x10	RW	Bit[7:0]: Fourth pulse PWM4 delay (0~31) 0x00: -0.5 frame 0x1F: 0.5 frame
0x3B44	DURATION CTRL0	0x11	RW	Bit[7:4]: Second pulse PWM2 duration (0~15 frames) Bit[3:0]: First pulse PWM1 duration (0~15 frames)
0x3B45	DURATION CTRL1	0x11	RW	Bit[7:4]: Fourth pulse PWM4 duration (0~15 frames) Bit[3:0]: Third pulse PWM3 duration (0~15 frames)
0x3B46	PULSE1 DUTY	0x1F	RW	Bit[7:0]: First pulse PWM1 duty cycle (0~31)
0x3B47	PULSE2 DUTY	0x1F	RW	Bit[7:0]: Second pulse PWM2 duty cycle increase step 0~31
0x3B48	PULSE3 DUTY	0x1F	RW	Bit[7:0]: Third pulse PWM3 duty cycle (0~31)
0x3B49	PULSE4 DUTY	0x1F	RW	Bit[7:0]: Fourth pulse PWM4 duty cycle decrease step 0~31
0x3B4A	GAP1	0x00	RW	Gap B/W Pulse 1 and Pulse 2 (0~255 frames)

table 7-12 illumination PWM registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3B4B	GAP2	0x00	RW	Gap B/W Pulse 2 and Pulse 3 (0~255 frames)
0x3B4C	GAP3	0x00	RW	Gap B/W Pulse 3 and Pulse 4 (0~255 frames)
0x3B4D	GAP4	0x00	RW	Gap B/W Pulse 4 and Pulse 1 When Repeat is On (0~255 frames)
0x3B4E	PWM CTRL	0x00	RW	Bit[7]: pwm_req_r (read only) Bit[6]: Delay option Bit[5]: illum_sel Bit[4]: duty_no_map Bit[3]: no_gap Bit[2]: sel_slot_out Bit[1]: Enable illumination mode 2 Bit[0]: pwm_repeat
0x3B4F	SLOT WIDTH	0x02	RW	Slot Width
0x3B50	PWM2 DUTY HOLD	0x01	RW	Bit[7:4]: Not used Bit[3:0]: Number of frames with duty cycle not change in PWM2
0x3B51	PWM4 DUTY HOLD	0x01	RW	Bit[7:4]: Not used Bit[3:0]: Number of frames with duty cycle not change in PWM4

7.13 OTP control registers [0x3D80 - 0x3D87]

table 7-13 OTP control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3D80	OTP PROGRAM CTRL	0x00	RW	Bit[7]: otp_pgenb_o 1: Program on going Bit[6:1]: Debug mode Bit[0]: Program OTP To start program, write 1'b1 to bit[0]
0x3D81	OTP LOAD CTRL	–	R	Bit[7]: opt_load_o 1: Load on going Bit[6:1]: Debug mode Bit[0]: Load / dump OTP Writing to this register will start loading data
0x3D82	OTP PROGRAM PULSE	0x40	RW	Bit[7:0]: Control program strobe pulse by 8*Tsclk

table 7-13 OTP control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3D83	OTP LOAD PULSE	0x05	RW	Bit[3:0]: Control load strobe pulse by Tsclk
0x3D84	OPT MODE CTRL	0x00	RW	Bit[7]: program_dis 0: Enable 1: Disable Bit[6]: mode_select 0: Auto mode 1: Manual mode Bit[5:0]: Memory select
0x3D85	OTP START ADDR	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: Start address for manual mode
0x3D86	OTP END ADDR	0x0F	RW	Bit[7:4]: Debug mode Bit[3:0]: End address for manual mode
0x3D87	OTP PS2CS	0x08	RW	Bit[7:4]: Debug mode Bit[3:0]: PS to CSB time control by sclk

7.14 BLC control registers [0x4000 - 0x4057, 0x5C00 - 0x5C08]

table 7-14 BLC control registers (sheet 1 of 6)

address	register name	default value	R/W	description
0x4000	BLC BYPASS	0x10	RW	Bit[7]: BLC bypass enable 0: Disable BLC bypass (BLC enable) 1: Enable BLC bypass (no BLC) Bit[6:5]: Debug mode Bit[4]: BLC zero line window select 0: Choose full width as window 1: Window select as BLC does Bit[3:0]: Debug mode
0x4001	BLC START LN	0x06	RW	Bit[7:6]: Not used Bit[5:0]: BLC start line (0, 2, 4, 6,...)

table 7-14 BLC control registers (sheet 2 of 6)

address	register name	default value	R/W	description
0x4002	BLC AUTO	0x45	RW	Bit[7]: Format change enable 0: BLC keep same after format change 1: BLC will redo after format change Bit[6]: BLC auto enable 0: Get the black level from manual register 1: Calculate the black level from auto statistics Bit[5:0]: Reset frame number[5:0] Frames that will continue to go through BLC after reset
0x4003	BLC FREEZE	0x08	RW	Bit[7]: BLC redo enable 0: Normal 1: Force BLC to redo N frames (where N = 0x4003[5:0]) when this bit set Bit[6]: Freeze enable 0: Normal 1: BLC black level will not update priority lower than always update Bit[5:0]: Manual frame number BLC redo frame number
0x4004	BLC BLC NUM	0x04	RW	Bit[7:6]: Not used Bit[5:0]: Number of black lines used

table 7-14 BLC control registers (sheet 3 of 6)

address	register name	default value	R/W	description
0x4005	BLC MAN CTRL	0x18	RW	Bit[7:6]: Debug mode Bit[5]: One line BLC mode (only for debug) Bit[4]: Do not output black line 0: Output black line 1: No black line output Bit[3]: blc_man_1_en Apply one channel offset (register 0x400C and 0x400D) to all manual BLC channels Bit[2]: Debug mode Bit[1]: blc_always_up_en 0: BLC will keep doing several frames after reset, after that, it will keep no change until gain change (controlled by register bit 0x4005[0]) or format change (controlled by register bit 0x4002[7]). 1: BLC always update in every frame Bit[0]: agc_change_from_system 0: agc_change generated by BLC PRE 1: agc_change from system
0x4006	BLCZLINE_COEF	0x40	RW	Bit[7:0]: Debug mode
0x4007	BLC WIN	0x80	RW	Bit[7]: Black line median filter enable 0: Disable median filter 1: Enable median filter Bit[6:5]: Debug mode Bit[4:3]: Window selection 00: Full image 01: Windows do not contain the first 16 pixels and the end 16 pixels 10: Windows do not contain the first 1/16 image and the end 1/16 image 11: Windows do not contain the first 1/8 image and the end 1/8 image Bit[2:0]: Debug mode

table 7-14 BLC control registers (sheet 4 of 6)

address	register name	default value	R/W	description
0x4008	BLC FLIP REG	0x00	RW	Bit[7:4]: First part BLC calculation start line address Bit[3]: BLC zero line black level manual enable Bit[2]: BLC debug Bit[1]: Manual flip enable Bit[0]: Flip value
0x4009	BLC TARGET	0x10	RW	Bit[7:0]: Black level target[7:0]
0x400A~ 0x400B	NOT USED	–	–	Not Used
0x400C	BLC MAN LEVEL0	0x00	RW	Bit[7:4]: Not used Bit[3:0]: BLC manual level channel 0[11:8]
0x400D	BLC MAN LEVEL0	0x00	RW	Bit[7:0]: BLC manual level channel 0[7:0]
0x400E	BLC MAN LEVEL1	0x00	RW	Bit[7:4]: Not used Bit[3:0]: BLC manual level channel 1[11:8]
0x400F	BLC MAN LEVEL1	0x00	RW	Bit[7:0]: BLC manual level channel 1[7:0]
0x4010	BLC MAN LEVEL2	0x00	RW	Bit[7:4]: Not used Bit[3:0]: BLC manual level channel 2[11:8]
0x4011	BLC MAN LEVEL2	0x00	RW	Bit[7:0]: BLC manual level channel 2[7:0]
0x4012	BLC MAN LEVEL3	0x00	RW	Bit[7:4]: Not used Bit[3:0]: BLC manual level channel 3[11:8]
0x4013	BLC MAN LEVEL3	0x00	RW	Bit[7:0]: BLC manual level channel 3[7:0]
0x4014	SHORT BLC MAN LEVEL0	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Short BLC manual level channel 0[11:8]
0x4015	SHORT BLC MAN LEVEL0	0x00	RW	Bit[7:0]: Short BLC manual level channel 0[7:0]
0x4016	SHORT BLC MAN LEVEL1	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Short BLC manual level channel 1[11:8]
0x4017	SHORT BLC MAN LEVEL1	0x00	RW	Bit[7:0]: Short BLC manual level channel 1[7:0]
0x4018	SHORT BLC MAN LEVEL2	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Short BLC manual level channel 2[11:8]
0x4019	SHORT BLC MAN LEVEL2	0x00	RW	Bit[7:0]: Short BLC manual level channel 2[7:0]

table 7-14 BLC control registers (sheet 5 of 6)

address	register name	default value	R/W	description
0x401A	SHORT BLC MAN LEVEL3	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Short BLC manual level channel 3[11:8]
0x401B	SHORT BLC MAN LEVEL3	0x00	RW	Bit[7:0]: Short BLC manual level channel 3[7:0]
0x401C~ 0x404D	DEBUG	–	–	Debug Registers
0x404E	BLC BLC MAX	0xFF	RW	Bit[7:0]: BLC maximum Maximum black level can reach. If the black level calculated is larger than the maximum, it will replaced by the maximum left shift and fill with 1 in the LSB. For example, If the max is 0x80. The black level is 0x90 and will use 0x203.
0x404F	BLC STABLE RANGE	0x7F	RW	Bit[7:0]: BLC stable range BLC will keep doing until the black level difference between the previous and the current frame is within the stable range.
0x4050~ 0x4057	DEBUG	–	–	Debug Registers
0x5C00	PBLC CTRL	0x80	RW	Bit[7]: Pre BLC enable Bit[6]: Image window cut enable It will cut register 0x5C05 lines from the image Bit[5:1]: Debug mode Bit[0]: BLC window cut enable 0: No window cut, BLC will receive all the black lines. 1: Cut the black lines Provide two windows. The black lines between registers 0x5C01 and 0x5C02 will remain. The black lines between registers 0x5C03 and 0x5C04 will also remain.
0x5C01	PBLC WIN_START1	0x00	RW	Bit[7:5]: Debug mode Bit[4:0]: BLC window start1
0x5C02	PBLC WIN_END1	0x00	RW	Bit[7:5]: Debug mode Bit[4:0]: BLC window end1
0x5C03	PBLC WIN_START2	0x00	RW	Bit[7:5]: Debug mode Bit[4:0]: BLC window start2

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

address	register name	default value	R/W	description
0x5C04	PBLC WIN_END2	0x00	RW	Bit[7:5]: Debug mode Bit[4:0]: BLC window end2
0x5C05	PBLC WIN CNT	0x00	RW	Bit[7:6]: Debug mode Bit[5:0]: Image lines to cut
0x5C06~ 0x5C07	DEBUG	–	–	Debug Registers
0x5C08	PBLC BLC_NUM	0x10	RW	Bit[7]: BLC number manual enable BLC number is not from system. It is from 0x5C08[5:0] Bit[6]: Debug mode Bit[5:0]: BLC number manual value

7.15 frame control registers [0x4200 - 0x4243]

table 7-15 frame control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4200	FRAME CTRL0	0x00	RW	Bit[7:3]: Debug mode Bit[2]: fcnt_eof_sel Bit[1]: fcnt_mask_dis Bit[0]: fcnt_reset
0x4201	FRAME ON NUMBER	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: Frame ON number
0x4202	FRAME OFF NUMBER	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: Frame OFF number
0x4203	FRAME CTRL1	0x00	R/W	Bit[7:6]: Debug mode 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
0x4240	ISP FRAME CTRL0	0x00	R/W	Bit[7:3]: Debug mode Bit[2]: fcnt_eof_sel Bit[1]: fcnt_mask_dis Bit[0]: fcnt_reset
0x4241	ISP FRAME ON NUMBER	0x00	R/W	Bit[7:4]: Debug mode Bit[3:0]: Frame ON number

table 7-15 frame control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4242	ISP FRAME OFF NUMBER	0x00	R/W	Bit[7:4]: Debug mode Bit[3:0]: Frame OFF number
0x4243	ISP FRAME CTRL1	0x00	R/W	Bit[7:6]: Debug mode 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

7.16 format control registers [0x4300 - 0x4316]

table 7-16 format control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4300	DATA_MAX H	0xFF	RW	Bit[7:0]: Data max[9:2]
0x4301	DATA_MIN H	0x00	RW	Bit[7:0]: Data min[9:2]
0x4302	CLIP L	0x0C	RW	Bit[7:4]: Debug mode Bit[3:2]: Data max[1:0] Bit[1:0]: Data min[1:0]
0x4303	FORMAT CTRL3	0x00	RW	Bit[7]: Fading enable 0: Disable 1: Enable Bit[6]: Horizontal fading inverse 0: Normal 1: Inverse Bit[5]: Swap bit order Bit[4]: 16 color bar inverse 0: Normal 1: Inverse Bit[3]: 16 color bar enable 0: 16 color bar OFF 1: 16 color bar enable Bit[2]: Bit test enable (works when 0x4303[0] = 1) Bit[1]: Bit test pattern 8-bit enable Bit[0]: Bit test pattern mode
0x4304	FORMAT CTRL4	0x08	RW	Bit[7]: Debug mode Bit[6:4]: data_bit_swap Bit[3]: tst_full_win Bit[2:0]: bar_pad

table 7-16 format control registers (sheet 2 of 2)

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]: Debug mode Bit[3:2]: padff Bit[1:0]: padcc
0x4307	EMBED CTRL	0x31	RW	Bit[7:4]: embed_line_st Bit[3]: embed_start_man Bit[2]: dpc_threshold_opt 0: For white pixel 1: For black pixel Bit[1]: embed_byte_order Bit[0]: embed_en
0x4308~ 0x430F	DEBUG	–	–	Debug Registers
0x4310	DEBUG MODE	–	–	Debug Mode
0x4311	VSYNC_WIDTH_H	0x04	RW	Bit[7:0]: VSYNC width[15:8] (in term of pixel numbers)
0x4312	VSYNC_WIDTH_L/	0x00	RW	Bit[7:0]: VSYNC width[7:0] (in term of pixel numbers)
0x4313	VSYNC_CTRL	0x00	RW	Bit[7:5]: Debug mode Bit[4]: VSYNC polarity Bit[3:2]: VSYNC output select 00: VSYNC output 11: Bypass EOF to output Bit[1]: VSYNC mode 2 Bit[0]: VSYNC mode 1
0x4314	VSYNC_DELAY1	0x00	RW	Bit[7:0]: VSYNC trigger signal to VSYNC output delay[23:16]
0x4315	VSYNC_DELAY2	0x01	RW	Bit[7:0]: VSYNC trigger signal to VSYNC output delay[15:8]
0x4316	VSYNC_DELAY3	0x00	RW	Bit[7:0]: VSYNC trigger signal to VSYNC output delay[7:0]

7.17 VFIFO control registers [0x4600 - 0x460F]

table 7-17 VFIFO control registers

address	register name	default value	R/W	description
0x4600	VFIFO_R0	0x00	RW	Bit[7]: Debug mode Bit[6]: SOF rst fi enable Bit[5]: Vblanking signal sel Bit[4]: SRAM R/W mode 0: Toggle control 1: Read priority Bit[3]: Manual set line length Bit[2]: Debug mode Bit[1]: SRAM bypass enable Bit[0]: Reset VFIFO every frame
0x4601~ 0x460F	DEBUG	–	–	Debug Registers

7.18 MIPI control registers [0x4800 - 0x4867]

table 7-18 MIPI control registers (sheet 1 of 9)

address	register name	default value	R/W	description
0x4800	MIPI CTRL 00	0x04	RW	MIPI Control 00 Bit[7]: mipi_hs_only 1: MIPI always in high speed mode Bit[6]: Debug mode Bit[5]: Clock lane gate enable 0: Clock lane is free running 1: Gate clock lane when there is no packet to transmit Bit[4]: Line sync enable 0: Do not send line short packet for each line 1: Send line short packet for each line Bit[3]: Debug mode Bit[2]: Idle status 0: MIPI bus will be LP00 when there is no packet to transmit 1: MIPI bus will be LP11 when there is no packet to transmit Bit[1:0]: Debug mode

table 7-18 MIPI control registers (sheet 2 of 9)

address	register name	default value	R/W	description
0x4801	MIPI CTRL 01	0x0F	RW	MIPI Control 01 Bit[7]: Long packet data type manual enable 0: Use mipi_dt 1: Use dt_man_o as long packet data (see register **0x4814**[5:0]) Bit[6]: Short packet data type manual enable 1: Use dt_spkt as short packet data (see register **0x4815**[5:0]) Bit[5]: first_bit Change clk_lane first bit 0: Output 0x05 1: Output 0xAA Bit[4]: PH bit order for ECC 0: {DI[7:0],WC[7:0],WC[15:8]} 1: {DI[0:7],WC[0:7],WC[8:15]} Bit[3]: PH byte order for ECC 0: {DI,WC_l,WC_h} 1: {DI,WC_h,WC_l} Bit[2]: PH byte order2 for ECC 0: {DI,WC} 1: {WC,DI} 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]: Debug mode

table 7-18 MIPI control registers (sheet 3 of 9)

address	register name	default value	R/W	description
0x4802	MIPI CTRL 02	0x00	RW	MIPI Control 02 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_oBit[7:0]
0x4803	MIPI CTRL 03	0x50	RW	MIPI Control 03 Bit[7:4]: Debug mode Bit[3]: manu_ofset_o t_period manual offset Bit[2]: r_manual_half2one t_period half to 1 Bit[1:0]: Debug mode
0x4804	MIPI CTRL 04	0x8D	RW	MIPI Control 04 Bit[7:6]: Debug mode Bit[5]: PRBS enable Bit[4]: Lane number manual enable Bit[3:0]: Manual lane number

table 7-18 MIPI control registers (sheet 4 of 9)

address	register name	default value	R/W	description
0x4805	MIPI CTRL 05	0x10	RW	MIPI Control 05 Bit[7:6]: Debug mode Bit[5]: mipi_ul_tx_en Bit[4]: tx_lsb_first Bit[3:0]: sw_t_lpx
0x4806	MIPI REG R/W CTRL	0x28	RW	Bit[7:4]: Debug mode Bit[3]: lpda_retim_manu_o Bit[2]: lpda_retim_sel_o 1: Manual Bit[1]: lpck_retim_manu_o Bit[0]: lpck_retim_sel_o 1: Manual
0x4810	MIPI MAX FRAME COUNT	0xFF	RW	High Byte of Max Frame Count of Frame Sync Short Packet
0x4811	MIPI MAX FRAME COUNT	0xFF	RW	Low Byte of Max Frame Count of Frame Sync Short Packet
0x4813	MIPI CTRL13	0x00	RW	mipi_ctrl13 RW Bit[7:3]: Debug mode Bit[2]: vc_sel input vc or reg vc Bit[1:0]: vc Virtual channel of MIPI
0x4814	MIPI CTRL14	0x2A	RW	MIPI Control 14 Bit[7:6]: Debug mode Bit[5:0]: Data type in manual mode
0x4815	MIPI_DT_SPKT	0x00	RW	Bit[7]: Debug mode Bit[6]: pclk_div 0: Use falling edge of mipi_pclk_o to generate MIPI bus to PHY 1: Use rising edge of mipi_pclk_o to generate MIPI bus to PHY Bit[5:0]: Manual Data Type for short packet
0x4816	EMB_DT_SEL	0x52	RW	emb_dt_sel RW Bit[7]: Debug mode Bit[6]: emb_line_sel 1: Use emb_dt as dt in first emb_line_nu Bit[5:0]: emb_dt Manually set emb data type
0x4818	HS_ZERO_MIN	0x00	RW	High Byte of the Minimum Value for hs_zero, unit ns
0x4819	HS_ZERO_MIN	0x70	RW	Low Byte of the Minimum Value for hs_zero, unit ns hs_zero_real = hs_zero_min_o + Tui*ui_hs_zero_min_o
0x481A	HS_TRAIL_MIN	0x00	RW	High Byte of the Minimum Value for hs_trail, unit ns

table 7-18 MIPI control registers (sheet 5 of 9)

address	register name	default value	R/W	description
0x481B	HS_TRAIL_MIN	0x3C	RW	Low Byte of the Minimum Value for hs_trail, unit ns hs_trail_real = hs_trail_min_o + Tui*ui_hs_trail_min_o
0x481C	CLK_ZERO_MIN	0x01	RW	High Byte of the Minimum Value for clk_zero, unit ns
0x481D	CLK_ZERO_MIN	0x06	RW	Low Byte of the Minimum Value for clk_zero, unit ns clk_zero_real = clk_zero_min_o + Tui*ui_clk_zero_min_o
0x481E	CLK_PREPARE_MAX	0x5F	RW	clk_prepare_max RW Max value of clk_prepare, unit ns
0x481F	CLK_PREPARE_MIN	0x26	RW	Minimum Value for clk_prepare clk_prepare_real = clk_prepare_min_o + Tui*ui_clk_prepare_min_o
0x4820	CLK_POST_MIN	0x00	RW	High Byte of the Minimum Value for clk_post, unit ns Bit[1:0]: clk_post_min[9:8]
0x4821	CLK_POST_MIN	0x3C	RW	Low Byte of the Minimum Value for clk_post clk_post_real = clk_post_min_o + Tui*ui_clk_post_min_o Bit[7:0]: clk_post_min[7:0]
0x4822	CLK_TRAIL_MIN	0x00	RW	High Byte of the Minimum Value for clk_trail, unit ns Bit[1:0]: clk_trail_min[9:8]
0x4823	CLK_TRAIL_MIN	0x3C	RW	Low Byte of the Minimum Value for clk_trail clk_trail_real = clk_trail_min_o + Tui*ui_clk_trail_min_o Bit[7:0]: clk_trail_min[7:0]
0x4824	LPX_P_MIN	0x00	RW	High Byte of the Minimum Value for lpx_p, unit ns Bit[1:0]: lpx_p_min[9:8]
0x4825	LPX_P_MIN	0x32	RW	Low Byte of the Minimum Value for lpx_p lpx_p_real = lpx_p_min_o + Tui*ui_lpx_p_min_o Bit[7:0]: lpx_p_min[7:0]
0x4826	HS_PREPARE_MIN	0x28	RW	hs_prepare_min RW minimum value of hs_prepare, unit ns
0x4827	HS_PREPARE_MAX	0x55	RW	Max Value for hs_prepare hs_prepare_real = hs_prepare_min_o + Tui*ui_hs_prepare_min_o
0x4828	HS_EXIT_MIN	0x00	RW	High Byte of the Minimum Value for hs_exit, unit ns Bit[1:0]: hs_exit_min[9:8]
0x4829	HS_EXIT_MIN	0x64	RW	Low Byte of the Minimum Value for hs_exit hs_exit_real = hs_exit_min_o + Tui*ui_hs_exit_min_o Bit[7:0]: hs_exit_min[7:0]

table 7-18 MIPI control registers (sheet 6 of 9)

address	register name	default value	R/W	description
0x482A	UI_HS_ZERO_MIN	0x06	RW	Minimum UI Value of hs_zero, unit UI
0x482B	UI_HS_TRAIL_MIN	0x04	RW	Minimum UI Value of hs_trail, unit UI
0x482C	UI_CLK_ZERO_MIN	0x00	RW	Minimum UI Value of clk_zero, unit UI
0x482D	UI_CLK_PREPARE	0x00	RW	ui_clk_prepare_min_ctrl RW Bit[7:4]: ui_clk_prepare_max Bit[3:0]: ui_clk_prepare_min Value of clk_prepare, unit UI
0x482E	UI_CLK_POST_MIN	0x34	RW	Minimum UI Value of clk_post, unit UI
0x482F	UI_CLK_TRAIL_MIN	0x00	RW	Minimum UI Value of clk_trail, unit UI
0x4830	UI_LPX_P_MIN	0x00	RW	Minimum UI Value of lpx_p, unit UI
0x4831	UI_HS_PREPARE_MIN	0x64	RW	ui_hs_prepare RW Bit[7:4]: ui_hs_prepare_max Bit[3:0]: ui_hs_prepare_min UI value of hs_prepare, unit UI
0x4832	UI_HS_EXIT_MIN	0x00	RW	Minimum UI Value of hs_exit, unit UI
0x4836	GLB_MODE_SEL	0x00	RW	glb_mode_sel Bit[7:1]: Debug mode Bit[0]: smia_cal_en 0: Use period to calculate 1: Use SMIA bit rate to calculate
0x4837	PCLK_PERIOD	0x0A	RW	Period of pclk2x, pclk_div = 1, and 1-bit decimal
0x4838	MIPI_LP_GPIO0	0x00	RW	Bit[7]: lp_sel0 0: auto gen 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 gen 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

table 7-18 MIPI control registers (sheet 7 of 9)

address	register name	default value	R/W	description
0x4839	MIPI_LP_GPIO1	0x00	RW	Bit[7]: lp_sel2 0: Auto gen 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 gen mipi_lp_dir3_o 1: Use lp_dir_man3 to be mipi_lp_dir3_o Bit[2]: lp_dir_man3 0: Input 1: Output Bit[1]: lp_p3_o Bit[0]: lp_n3_o
0x483A	MIPI_LP_GPIO2	0x00	RW	Bit[7]: lp_sel4 0: Auto gen mipi_lp_dir4_o 1: Use lp_dir_man4 to be mipi_lp_dir4_o Bit[6]: lp_dir_man4 0: Input 1: Output Bit[5]: lp_p4_o Bit[4]: lp_n4_o Bit[3]: lp_sel5 0: Auto gen mipi_lp_dir5_o 1: Use lp_dir_man5 to be mipi_lp_dir5_o Bit[2]: lp_dir_man5 0: Input 1: Output Bit[1]: lp_p5_o Bit[0]: lp_n5_o

table 7-18 MIPI control registers (sheet 8 of 9)

address	register name	default value	R/W	description
0x483B	MIPI_LP_GPIO3	0x00	RW	Bit[7]: lp_sel6 0: Auto gen mipi_lp_dir6_o 1: Use lp_dir_man6 to be mipi_lp_dir6_o Bit[6]: lp_dir_man6 0: Input 1: Output Bit[5]: lp_p6_o Bit[4]: lp_n6_o Bit[3]: lp_sel7 0: Auto gen mipi_lp_dir7_o 1: Use lp_dir_man7 to be mipi_lp_dir7_o Bit[2]: lp_dir_man7 0: Input 1: Output Bit[1]: lp_p7_o Bit[0]: lp_n7_o
0x483C	MIPI_CTRL3C	0x42	RW	Bit[7:4]: Debug mode Bit[3:0]: t_clk_pre Unit: pclk2x cycle
0x483D	MIPI_LP_GPIO4	0x00	RW	Bit[7]: lp_ck_sel0 0: Auto gen 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 gen 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
0x4840	START_OFFSET[12:8]	0x00	RW	High Byte of start_offset
0x4841	START_OFFSET	0x00	RW	Low Byte of start_offset
0x4842	START_MODE	0x01	RW	Bit[7:2]: Debug mode Bit[1:0]: Delay mode select 00: Delay one line mode 01: Old mode, delay about 100 Tp 10: VHREF manual mode 11: Debug mode

table 7-18 MIPI control registers (sheet 9 of 9)

address	register name	default value	R/W	description
0x484A	SEL_MIPI_CTRL4A	0x07	RW	Bit[7:3]: Debug mode 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]: Debug mode Bit[2]: line_st_sel_o 0: LS after HREF 1: LS after fifo_st Bit[1]: clk_start_sel_o 0: Clk starts after SOF 1: Clk starts after reset Bit[0]: sof_sel_o 0: FS after HREF occurs 1: FS after SOF
0x484C	SEL_MIPI_CTRL4C	0x02	RW	Bit[7:6]: Debug mode Bit[5]: SMIA fcnt_i select Bit[4]: MIPI high speed only test mode enable Bit[3]: Set frame count to inactive mode (keep 0) Bit[2]: Vsub select 0: Valid from behind 1: Valid in front Bit[1:0]: Input data valid (e.g., for YUV420) 01: Valid = 1 10: Valid = 2 11: Valid = 3
0x484D	TEST_PATTEN_DATA	0xB6	RW	Data Lane Test Pattern
0x484E	FE_DLY	0x10	RW	Last Packet to Frame End Delay / 2
0x484F	TEST_PATTEN_CK_DATA	0x55	RW	clk_test_patten_reg
0x4864~0x4867	DEBUG	–	–	Debug Registers

7.19 LVDS registers [0x4A00 - 0x4A0D]

table 7-19 LVDS registers

address	register name	default value	RW	description
0x4A00	LVDS_R0 2A	0xAA	RW	Bit[6]: SYNC code manual mode enable Bit[5]: SYNC code enable when only 1 lane Bit[4]: PCLK invert enable Bit[3]: Bit swap default 1: Do swap Bit[2]: F parameter in CCIR656 standard Bit[1]: SAV first enable Bit[0]: SYNC code mode 0: Split 1: Per lane
0x4A02	LVDS_R2 0	0x00	RW	Bit[7:0]: Dummy data0[15:8]
0x4A03	LVDS_R3 80	0x01	RW	Bit[7:0]: Dummy data0[7:0]
0x4A04	LVDS_R4 00	0x00	RW	Bit[7:0]: Dummy data1[15:8]
0x4A05	LVDS_R5 10	0x08	RW	Bit[7:0]: Dummy data1[7:0]
0x4A06	LVDS_R6 00	0xAA	RW	frame_start Sync Code in Manual Sync Code Mode
0x4A07	LVDS_R7 00	0x55	RW	frame_end Sync Code in Manual Sync Code Mode
0x4A08	LVDS_R8 00	0x99	RW	line_start Sync Code in Manual Sync Code Mode
0x4A09	LVDS_R9 00	0x66	RW	line_end Sync Code in Manual Sync Code Mode
0x4A0A	LVDS_RA 00	0x08	RW	Debug Mode
0x4A0B	LVDS_RB 00	0x00	RW	Debug Mode
0x4A0C	LVDS_RC 00	0x00	RW	Debug Mode
0x4A0D	LVDS_RD 00	0x00	RW	Debug Mode

7.20 temperature monitor registers [0x4D00 - 0x4D21]

table 7-20 temperature monitor registers

address	register name	default value	R/W	description
0x4D00~ 0x4D05	DEBUG	–	–	Debug Registers
0x4D06	TPM_CTRL0	0x78	RW	Bit[7]: Debug mode Bit[6:4]: cnt_bit Bit[1:0]: Div
0x4D07	TPM_CTRL_OPT	0x00	RW	Bit[7:3]: Debug mode Bit[2]: otp_reg_ctrl_en Bit[1]: result_shift_en Bit[0]: Stall
0x4D08	TPM_CTRL1	0x05	RW	Bit[7]: mul_div_sel Bit[6]: div_sel Bit[5]: pd_tmp_snr Bit[4:0]: shift_bit
0x4D09~ 0x4D21	DEBUG	–	–	Debug Registers

7.21 ISP registers [0x5000 - 0x5061, 0x5E00 - 0x5E24]

table 7-21 ISP registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x5000	ISP CTRL0	0x06	RW	Bit[7]: LENC correction enable 0: Disable 1: Enable Bit[6:3]: Not used Bit[2]: Black pixel cancellation enable 0: Disable 1: Enable Bit[1]: White pixel cancellation enable 0: Disable 1: Enable Bit[0]: Not used
0x5001	ISP CTRL1	0x01	RW	Bit[7:1]: Not used Bit[0]: Manual white balance (MWB) enable 0: Disable 1: Enable

table 7-21 ISP registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x5002	ISP CTRL2	0x00	RW	Bit[7]: Scale enable 0: Disable 1: Enable Bit[6:0]: Debug mode
0x5003~ 0x5004	DEBUG MODE	–	–	Debug Mode
0x5005	ISP BIAS CTRL	0x1C	RW	Bit[7:5]: Not used Bit[4]: MWB bias on This will subtract the BLC target before MWB gain, and add the target back after MWB 0: Disable 1: Enable Bit[3:0]: Not used
0x5006~ 0x5011	DEBUG MODE	–	–	Debug Mode
0x5012	ISP CTRL 12	0x15	RW	Bit[7:6]: Debug mode Bit[5:4]: Scale SRAM0 test Bit[3:2]: Scale SRAM1 test Bit[1:0]: Scale SRAM2 test
0x5013	ISP CTRL 13	0x04	RW	Bit[7:3]: Debug mode Bit[2]: LSB enable Bit[1:0]: Debug mode
0x501F	ISP BYPASS	0x00	RW	Bit[7:6]: Not used Bit[5]: Bypass ISP Bypasses all ISP modules except window and pre-ISP Bit[4:0]: Debug mode
0x5025	ISP AVG SEL	0x00	RW	Bit[7:4]: Debug mode Bit[3:2]: Not used Bit[1:0]: Average select 00: Use sensor raw to calculate average data 01: Use the data after LENC to calculate average data 10: Use the data after MWB_gain to calculate average data
0x502A~ 0x5040	DEBUG MODE	–	–	Debug Mode

table 7-21 ISP registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x5041	ISP CTRL41	0x04	RW	Bit[7]: Scale manually select 0: Disable, scale is auto enabled when the output size is less than the input size 1: Enable, scale is manually enabled or disabled, depending on register 0x5041[5] Bit[6]: Not used Bit[5]: Debug mode Bit[4]: Post binning filter enable 0: Disable 1: Enable Bit[3]: Not used Bit[2]: Average enable 0: Disable 1: Enable Bit[1:0]: Not used
0x5042~ 0x5045	DEBUG	–	–	Debug Registers
0x5046	ISP SOF SEL	0x0A	RW	Bit[7:6]: ISP SOF select 00: Auto mode, ISP output the SOF automatic 01: VSYNC 10: TC_SOF 11: PRE SOF Bit[5:0]: Debug mode
0x5047~ 0x505F	DEBUG	–	–	Debug Registers
0x5061	DEBUG	–	–	Debug Register

table 7-21 ISP registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x5E00	PRE ISP TEST CTRL	0x00	RW	Bit[7]: test_enable Bit[6]: Rolling enable (rolling bar in test mode) Bit[5]: Transparent image + normal image enable Bit[4]: Square black white enable Bit[3:2]: Color bar style 00: Horizontal bar 01: Vertical bar 10: Horizontal fading bar 11: Vertical fading bar Bit[1:0]: Test selection 00: Color bar 01: Random data 10: Square black white 11: Black
0x5E01	PRE ISP WIN	0x41	RW	Bit[7]: Not used Bit[6]: Window cut enable Bit[5]: ISP test Low bits to 0 Bit[4]: Random Random data reset Bit[3:0]: Random seed
0x5E02~ 0x5E03	DEBUG	–	–	Debug Registers
0x5E04	PRE ISP SCALE X	0x00	RW	Scale X Input Manual Size High Byte
0x5E05	PRE ISP SCALE X	0x00	RW	Scale X Input Manual Size Low Byte
0x5E06	PRE ISP SCALE Y	0x01	RW	Scale Y Input Manual Size High Byte
0x5E07	PRE ISP SCALE Y	0x00	RW	Scale Y Input Manual Size Low Byte
0x5E08	PRE ISP X OFF	0x00	RW	X Manual Offset High Byte
0x5E09	PRE ISP X OFF	0x00	RW	X Manual Offset Low Byte
0x5E0A	PRE ISP Y OFF	0x00	RW	Y Manual Offset High Byte
0x5E0B	PRE ISP Y OFF	0x01	RW	Y Manual Offset Low Byte
0x5E10~ 0x5E24	DEBUG	–	–	Debug Registers

7.22 scale control registers [0x5600 - 0x5608]

table 7-22 scale control registers

address	register name	default value	R/W	description
0x5600	SCALE HFACTOR	0x00	RW	Bit[7:2]: Not used Bit[1:0]: Scale horizontal factor[9:8]
0x5601	SCALE HFACTOR	0x80	RW	Bit[7:0]: Scale horizontal factor[7:0]
0x5602	SCALE VFACTOR	0x00	RW	Bit[7:2]: Not used Bit[1:0]: Scale vertical factor[9:8]
0x5603	SCALE VFACTOR	0x80	RW	Bit[7:0]: Scale vertical factor[7:0]
0x5604	SCALE AUTO	0x01	RW	Bit[7:1]: Debug mode Bit[0]: Scale auto enable 0: Use manual scale factor from registers 0x5600~0x5603 1: Calculate the scale factor from the input size and the output size set in registers 0x3800~0x380B
0x5605~ 0x5608	DEBUG	–	–	Debug Registers
0x5068	SCALE H INVT	0x00	RW	Bit[7:6]: Horizontal MSB Bit[5]: Not used Bit[4:0]: Horizontal inverse
0x506A	SCALE V INVT	0x00	RW	Bit[7:6]: Vertical MSB Bit[5]: Not used Bit[4:0]: Vertical inverse

7.23 DPC control registers [0x5780 - 0x57A1]

table 7-23 DPC control registers

address	register name	default value	R/W	description
0x5000	ISP CTRL0	0x06	RW	Bit[7]: LENC correction enable 0: Disable 1: Enable Bit[6:3]: Not used Bit[2]: Black pixel cancellation enable 0: Disable 1: Enable Bit[1]: White pixel cancellation enable 0: Disable 1: Enable Bit[0]: Not used
0x5780~ 0x57A1	DPC CTRL	–	–	DPC Control Registers

7.24 LENC registers [0x5800 - 0x5849]

table 7-24 LENC registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x5000	ISP CTRL0	0x06	RW	Bit[7]: LENC correction enable 0: Disable 1: Enable Bit[6:3]: Not used Bit[2]: Black pixel cancellation enable 0: Disable 1: Enable Bit[1]: White pixel cancellation enable 0: Disable 1: Enable Bit[0]: Not used
0x5800	LENC G00	0x10	RW	Bit[7:6]: Debug mode Bit[5:0]: Control point G00 for luminance compensation
0x5801	LENC G01	0x10	RW	Bit[7:6]: Debug mode Bit[5:0]: Control point G01 for luminance compensation

table 7-24 LENC registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x5802	LENC G02	0x10	RW	Bit[7:6]: Debug mode Bit[5:0]: Control point G02 for luminance compensation
0x5803	LENC G03	0x10	RW	Bit[7:6]: Debug mode Bit[5:0]: Control point G03 for luminance compensation
0x5804	LENC G04	0x10	RW	Bit[7:6]: Debug mode Bit[5:0]: Control point G04 for luminance compensation
0x5805	LENC G05	0x10	RW	Bit[7:6]: Debug mode Bit[5:0]: Control point G05 for luminance compensation
0x5806	LENC G10	0x10	RW	Bit[7:6]: Debug mode Bit[5:0]: Control point G10 for luminance compensation
0x5807	LENC G11	0x08	RW	Bit[7:6]: Debug mode Bit[5:0]: Control point G11 for luminance compensation
0x5808	LENC G12	0x08	RW	Bit[7:6]: Debug mode Bit[5:0]: Control point G12 for luminance compensation
0x5809~ 0x5822	LENC G13~ LENC G54	–	RW	Bit[7:6]: Debug mode Bit[5:0]: Control point G13~G54 for luminance compensation
0x5823	LENC G55	0x10	RW	Bit[7:6]: Debug mode 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

table 7-24 LENC registers (sheet 3 of 4)

address	register name	default value	R/W	description
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 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 SENSORGAIN THRESHOLD1	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 SENSORGAIN THRESHOLD2	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	MIN LENC GAIN	0x18	RW	Bit[7]: Debug mode 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]

table 7-24 LENC registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x5841	LENC CTRL	0x0D	RW	Bit[7:4]: Debug mode 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]: Debug mode 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	For horizontal color gain calculation, this value indicates the step between two connected horizontal pixels. Bit[7:3]: Debug mode Bit[2:0]: br_Hscale[10:8]
0x5843	LENC BRHSCALE	0xEF	RW	Bit[7:0]: br_Hscale[7:0]
0x5844	LENC BRVSCALE	0x01	RW	For vertical color gain calculation, this value indicates the step between two connected vertical pixels. Bit[7:3]: Debug mode Bit[2:0]: br_Vscale[10:8]
0x5845	LENC BRVSCALE	0x3E	RW	Bit[7:0]: br_Vscale[7:0]
0x5846	LENC GHSCALE	0x01	RW	For horizontal luminance gain calculation, this value indicates the step between two connected horizontal pixels. Bit[7:3]: Debug mode Bit[2:0]: g_Hscale[10:8]
0x5847	LENC GHSCALE	0x3E	RW	Bit[7:0]: g_Hscale[7:0]
0x5848	LENC GVSCALE	0x00	RW	For vertical luminance gain calculation, this value indicates the step between two connected horizontal pixels. Bit[7:3]: Debug mode Bit[2:0]: g_Vscale[10:8]
0x5849	LENC GVSCALE	0xD4	RW	Bit[7:0]: g_Vscale[7:0]

7.25 window control registers [0x5A00 - 0x5A0C]

table 7-25 window control registers

address	register name	default value	R/W	description
0x5A00	WIN XSTART_OFF	0x00	RW	Bit[7:5]: Debug mode Bit[4:0]: X start offset[12:8]
0x5A01	WIN XSTART_OFF	0x00	RW	Bit[7:0]: X start offset[7:0]
0x5A02	WIN YSTART_OFF	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: Y start offset[11:8]
0x5A03	WIN YSTART_OFF	0x00	RW	Bit[7:0]: Y start offset[7:0]
0x5A04	WIN WIDTH	0x10	RW	Bit[7:5]: Debug mode Bit[4:0]: Window width[12:8]
0x5A05	WIN WIDTH	0xA0	RW	Bit[7:0]: Window width[7:0]
0x5A06	WIN HEIGHT	0x0C	RW	Bit[7:4]: Debug mode Bit[3:0]: Window height[11:8]
0x5A07	WIN HEIGHT	0x78	RW	Bit[7:0]: Window height[7:0]
0x5A08	WIN MAN	0x00	RW	Bit[7:1]: Debug mode Bit[0]: Window size manual 0: Disable 1: Enable
0x5A09	WIN PX_CNT	–	R	Bit[7:5]: Debug mode Bit[4:0]: Pixel counter[12:8]
0x5A0A	WIN PX_CNT	–	R	Bit[7:0]: Pixel counter[7:0]
0x5A0B	WIN LN_CNT	–	R	Bit[7:4]: Debug mode Bit[3:0]: Line counter[11:8]
0x5A0C	WIN LN_CNT	–	R	Bit[7:0]: Line counter[7:0]

8 operating specifications

8.1 absolute maximum ratings

table 8-1 absolute maximum ratings

parameter		absolute maximum rating[a]
ambient storage temperature		-40°C to +95°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.

8.2 functional temperature

table 8-2 functional temperature

parameter	range
operating temperature[a]	-30°C to +70°C junction temperature
stable operating temperature[b]	0°C to +50°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

8.3 DC characteristics

table 8-3 DC characteristics (-30°C < T_J < 70°C)

symbol	parameter	min	typ	max	unit
supply					
V_{DD-A}	supply voltage (analog)	2.6	2.8	3.0	V
V_{DD-D}[a]	supply voltage (digital core for 2-lane MIPI up to 1 Gbps/lane)	1.27	1.3	1.32	V
V_{DD-D}[a]	supply voltage (digital core for 4-lane MIPI up to 700 Mbps/lane)	1.14	1.2	1.32	V
V_{DD-IO}	supply voltage (digital I/O)	1.7	1.8	3.0	V
I_{DD-A}	active (operating) current[b]		60	80	mA
I_{DD-IO}			95	130	mA
$I_{DDS-SCCB}$	standby current[c]		300	3000	µA
$I_{DDS-PWDN}$			300	3000	µA
$I_{DDS-XSHUTDOWN}$			10	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. using the internal regulator is strongly recommended for minimum power down currents

b. DVDD is generated by internal regulator. DVDD and EVDD are tied together.

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

d. based on DOVDD = 1.8V

8.4 timing characteristics

table 8-4 timing characteristics

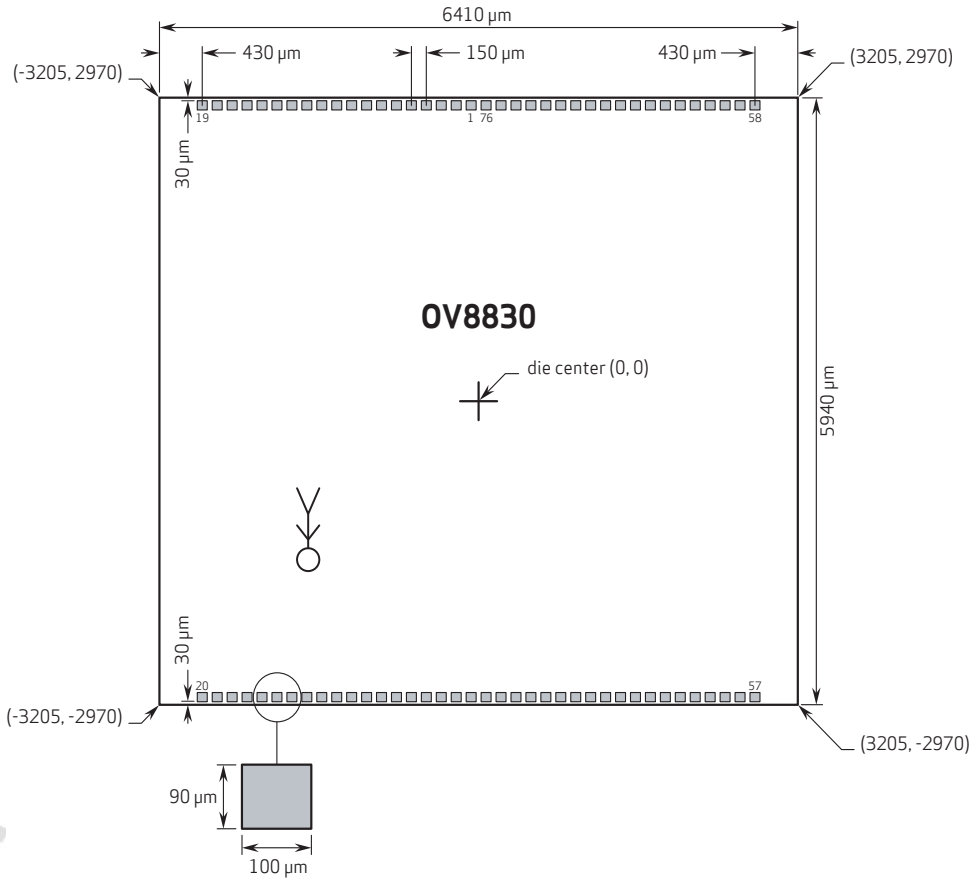
symbol	parameter	min	typ	max	unit
oscillator and clock input					
f_{OSC}	frequency (EXTCLK)	6	24	27	MHz
t_r, t_f	clock input rise/fall time			5 (10[a])	ns

a. if using internal PLL

9 mechanical specifications

9.1 physical specifications

figure 9-1 die specifications



note 1 all dimensions and coordinates are in µm.

table 9-1 pad location coordinates (sheet 1 of 4)

pad number	pad name	x coordinate	y coordinate	x pitch	y pitch	pad size
01	SIOD	-75	2895			100x90
02	NC	-225	2895	-150	0	100x90
03	NC	-375	2895	-150	0	100x90
04	FSO/VSYNC	-525	2895	-150	0	100x90

table 9-1 pad location coordinates (sheet 2 of 4)

pad number	pad name	x coordinate	y coordinate	x pitch	y pitch	pad size
05	FREX	-675	2895	-150	0	100x90
06	IL_PWM	-825	2895	-150	0	100x90
07	AVDD	-975	2895	-150	0	100x90
08	AGND	-1125	2895	-150	0	100x90
09	STROBE	-1275	2895	-150	0	100x90
10	PWDNB	-1425	2895	-150	0	100x90
11	XSHUTDOWN	-1575	2895	-150	0	100x90
12	TM	-1725	2895	-150	0	100x90
13	AGND	-1875	2895	-150	0	100x90
14	DVDD	-2025	2895	-150	0	100x90
15	DGND	-2175	2895	-150	0	100x90
16	AVDD	-2325	2895	-150	0	100x90
17	AVDD	-2475	2895	-150	0	100x90
18	AGND	-2625	2895	-150	0	100x90
19	AGND	-2775	2895	-150	0	100x90
20	VN2	-2775	-2895	0	-5790	100x90
21	VN	-2625	-2895	150	0	100x90
22	VH	-2475	-2895	150	0	100x90
23	AVDD	-2325	-2895	150	0	100x90
24	AVDD	-2175	-2895	150	0	100x90
25	AGND	-2025	-2895	150	0	100x90
26	AGND	-1875	-2895	150	0	100x90
27	DOVDD	-1725	-2895	150	0	100x90
28	DOGND	-1575	-2895	150	0	100x90
29	GPIO_2	-1425	-2895	150	0	100x90
30	NC	-1275	-2895	150	0	100x90
31	EXTCLK	-1125	-2895	150	0	100x90
32	EXTREG_EN	-975	-2895	150	0	100x90
33	VTST2	-825	-2895	150	0	100x90
34	EGND	-675	-2895	150	0	100x90

table 9-1 pad location coordinates (sheet 3 of 4)

pad number	pad name	x coordinate	y coordinate	x pitch	y pitch	pad size
35	MDP2	-525	-2895	150	0	100x90
36	MDN2	-375	-2895	150	0	100x90
37	MDP0	-225	-2895	150	0	100x90
38	MDN0	-75	-2895	150	0	100x90
39	MCP	75	-2895	150	0	100x90
40	MCN	225	-2895	150	0	100x90
41	EGND	375	-2895	150	0	100x90
42	EGND	525	-2895	150	0	100x90
43	EVDD	675	-2895	150	0	100x90
44	EVDD	825	-2895	150	0	100x90
45	MDP1	975	-2895	150	0	100x90
46	MDN1	1125	-2895	150	0	100x90
47	MDP3	1275	-2895	150	0	100x90
48	MDN3	1425	-2895	150	0	100x90
49	PVDD	1575	-2895	150	0	100x90
50	AGND	1725	-2895	150	0	100x90
51	DOGND	1875	-2895	150	0	100x90
52	DOVDD	2025	-2895	150	0	100x90
53	DOVDD	2175	-2895	150	0	100x90
54	DVDD	2325	-2895	150	0	100x90
55	DVDD	2475	-2895	150	0	100x90
56	DGND	2625	-2895	150	0	100x90
57	DGND	2775	-2895	150	0	100x90
58	DGND	2775	2895	0	5790	100x90
59	DGND	2625	2895	-150	0	100x90
60	DVDD	2475	2895	-150	0	100x90
61	DVDD	2325	2895	-150	0	100x90
62	AVDD	2175	2895	-150	0	100x90
63	AGND	2025	2895	-150	0	100x90
64	AGND	1875	2895	-150	0	100x90

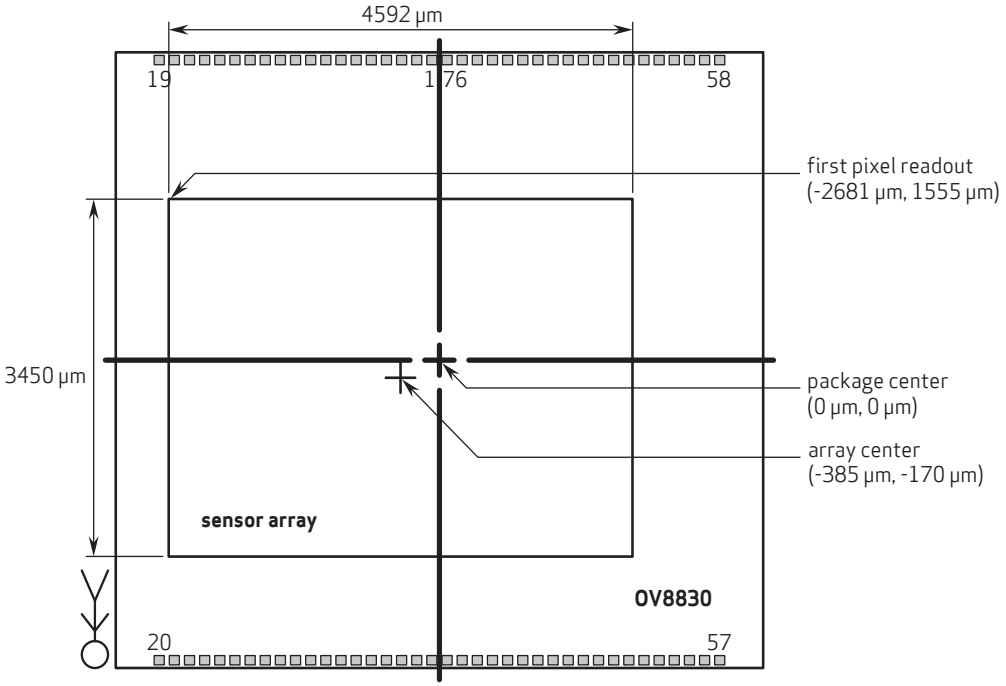
table 9-1 pad location coordinates (sheet 4 of 4)

pad number	pad name	x coordinate	y coordinate	x pitch	y pitch	pad size
65	AGND	1725	2895	-150	0	100x90
66	AVDD	1575	2895	-150	0	100x90
67	AVDD	1425	2895	-150	0	100x90
68	VTST	1275	2895	-150	0	100x90
69	DGND	1125	2895	-150	0	100x90
70	DVDD	975	2895	-150	0	100x90
71	DOVDD	825	2895	-150	0	100x90
72	GPIO_0	675	2895	-150	0	100x90
73	GPIO_2	525	2895	-150	0	100x90
74	SID	375	2895	-150	0	100x90
75	FSIN	225	2895	-150	0	100x90
76	SIOC	75	2895	-150	0	100x90

10 optical specifications

10.1 sensor array center

figure 10-1 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.

8830_COB_DS_10_1

10.2 lens chief ray angle (CRA)

figure 10-2 chief ray angle (CRA)

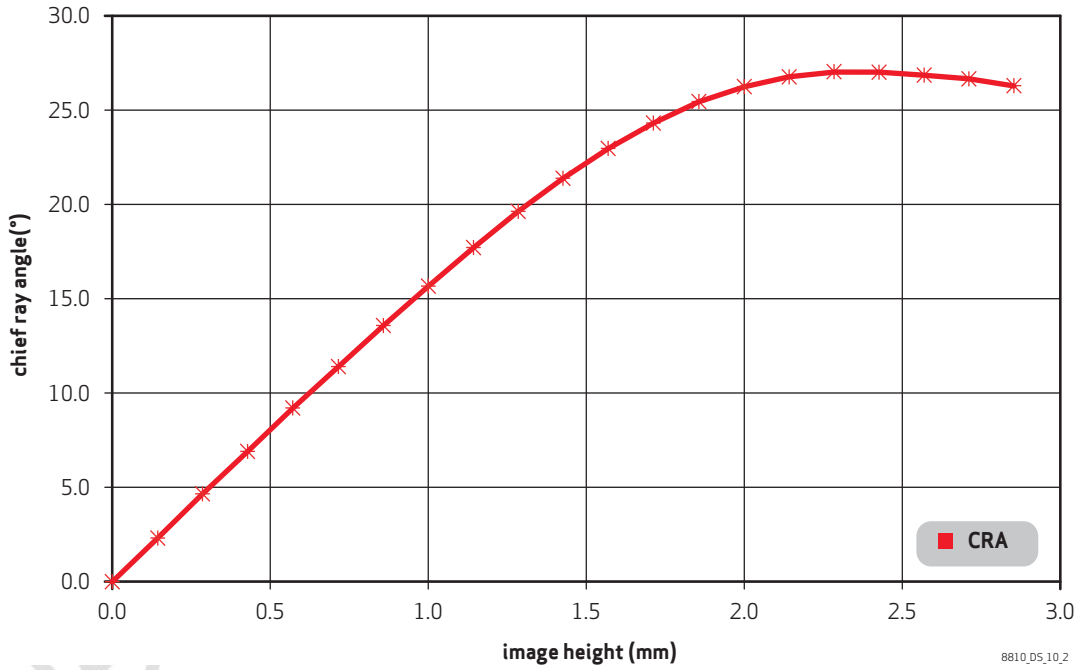


table 10-1 CRA versus image height plot (sheet 1 of 2)

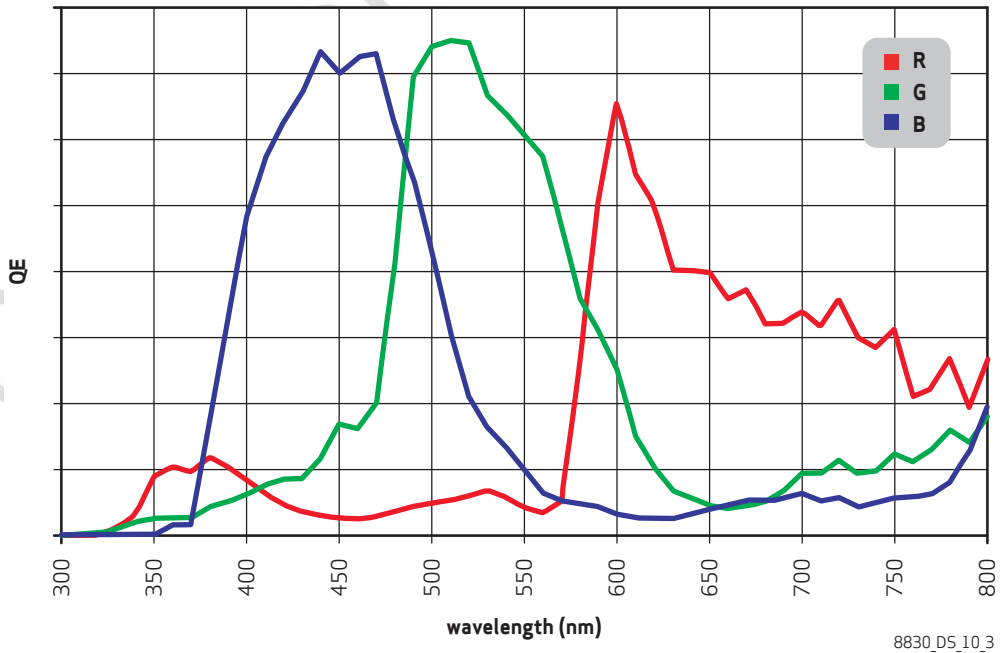
field (%)	image height (mm)	CRA (degrees)
0.00	0.000	0.0
0.05	0.143	2.3
0.10	0.286	4.6
0.15	0.428	6.9
0.20	0.571	9.2
0.25	0.714	11.4
0.30	0.857	13.6
0.35	1.000	15.7
0.40	1.142	17.7
0.45	1.285	19.6
0.50	1.428	21.4

table 10-1 CRA versus image height plot (sheet 2 of 2)

field (%)	image height (mm)	CRA (degrees)
0.55	1.571	23.0
0.6	1.714	24.4
0.65	1.856	25.5
0.70	1.999	26.3
0.75	2.142	26.8
0.80	2.285	27.0
0.85	2.428	27.0
0.90	2.570	26.9
0.95	2.713	26.7
1.00	2.856	26.3

10.3 spectrum response curve

figure 10-3 spectrum response curve diagram



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 air blowing 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 11.04.2010

- initial release

version 1.1 12.08.2010

- in the ordering information section, added product OV08830-G20A
- in the key specifications section, added "XSHUTDOWN: 10µA" to power requirements
- in the key specifications section, added pixel performance note
- in the key specifications section, added "junction temperature" to temperature range
- in chapter 2, figures 2-15 and 2-16, updated figure from XVCLK to EXTCLK
- in chapter 2, figure 2-15, updated figure from "PLL1_op_pix_div 4/5" to PLL1_op_pix_div 8/10"
- in chapter 4, table 4-14, updated 0x37C5[7:0] description from "Exposure time is in units of 128 clock cycles" to "Exposure time is in units of 256 clock cycles"
- in chapter 4, table 4-14, updated 0x37C9[3:0] description from "Strobe width is in units of 1 clock cycle" to "Strobe width is in units of 2 clock cycle"
- in chapter 4, table 4-14, updated 0x37CC[4:0], 0x37CD[7:0] descriptions from "Shutter delay is in units of 128 clock cycles" to "Shutter delay is in units of 256 clock cycles"
- in chapter 4, table 4-14, updated 0x37CE[7:0], 0x37CF[7:0] descriptions from "Sensor precharge is in units of 1 system clock cycle..." to "Sensor precharge is in units of 2 system clock cycle..."
- in chapter 4, table 4-14, updated 0x37D0[7:0], 0x37D1[7:0] descriptions from "Readout delay time is in units of 128 clock cycles" to "Readout delay time is in units of 256 clock cycles"
- in chapter 4, section 4.9.2.1, updated "...128 clock cycles" to "...256 clock cycles"
- in chapter 4, section 4.9.2.1, updated "If OV8830 works at 96 MHz, the minimum exposure time is 0 and minimum step is 1.33 s" to "If OV8830 works at 216 MHz, the minimum exposure time is 0 and minimum step is 1.2 µs"
- in chapter 4, section 4.9.2.1, updated "If OV8830 works at 96 MHz, the minimum exposure time is 0 and minimum step is 22.37 s" to "If OV8830 works at 216 MHz, the minimum exposure time is 0 and minimum step is 19.9 µs"
- in chapter 4, section 4.9.2.2, updated "...1 step = 128 clock cycles" to "...1 step = 256 clock cycles"
- in chapter 4, section 4.9.2.2, updated "Minimum step is 1.33 µs" to "Minimum step is 1.2 µs"
- in chapter 4, section 4.9.2.2, updated "If OV8830 works at 96 MHz...delay time is 10.92 ms." to "If OV8830 works at 216 MHz...delay time is 9.7 ms."
- in chapter 4, section 4.9.2.3, updated "...1 step = 1 system clock cycle..." to "...1 step = 2 system clock cycles..."
- in chapter 4, section 4.9.2.4, updated "...1 step = 1 system clock cycle..." to "...1 step = 2 system clock cycles..."
- in chapter 4, section 4.9.2.5, updated "...1 step = 128 clock cycles..." to "...1 step = 256 clock cycles..."
- in chapter 4, section 4.9.2.5, updated "Minimum step is 1.33 µs" to "Minimum step is 1.2 µs"
- in chapter 4, section 4.9.2.5, updated "If OV8830 works at 96 MHz, the maximum strobe delay time is 10.92 ms." to "If OV8830 works at 216 MHz, the maximum strobe delay time is 9.7 ms."

- in chapter 7, table 7-2, updated register 0x30B5[3:0] from "0x4: /4" to "0x4: /8" and from "0x5: /5" to "0x5: /10"
- in chapter 7, table 7-2, updated register 0x30B4[2:0]
- in chapter 7, table 7-2, updated register 0x30B5[3:0]
- in chapter 7, table 7-2, updated register 0x30B6[3:0]
- in chapter 8, added " junction temperature" to table 8-2 functional temperature
- in the key specifications section, updated active power requirements from "170mA (419mW)" to "155mA (339mW)" and standby power requirements from "30µA" to "300µA"
- in chapter 8, table 8-3, updated I_{DD-A} active (operating) current typical value from "95" to "60"
- in chapter 8, table 8-3, updated I_{DD-IO} active (operating) current typical value from "85" to "95"
- in chapter 8, table 8-3, updated $I_{DDS-SCCB}$ standby current typical value from "30" to "700"
- in chapter 8, table 8-3, updated $I_{DDS-PWDN}$ standby current typical value from "30" to "300"
- in chapter 8, table 8-3, deleted table note "b. active current is based on sensor resolution at full size and full speed, with the MIPI function, the active current needs an additional 20mA"
- in chapter 8, table 8-3, added "$I_{DDS-XSHUTDOWN}$" with a typical value "10" and maximum value "TBD"

version 1.2 01.07.2011

- on page iii, under key specifications, added sidebar note to active power requirements
- in table 1-2, added to "(XSHUTDOWN=0)" to RESET column and "(XSHUTDOWN from 0 to 1)" after RESET release column
- in table 1-3, moved "SID and EXTREG_EN" to XSHUTDOWN
- in table 2-1, changed title from "MIPI supported frame..." to "supported resolution and frame rate" and removed note b
- in sub-section 2.7.1, changed "PWDN" to "PWDNB"
- in table 2-5, added EXTREG_EN column and changed "EXTREG_EN is connected to DOVDD" to "Refer to"
- in section 2, added new table 2-6
- in table 2-7, added "or DOVDD falling..." to "XSHUTDOWN falling.." and new row for PWDNB falling - DOVDD falling
- in section 2, updated figures 2-9, 2-10, 2-11, 2-12, 2-13, moved figure 2-13 to end of section 2.7, added table 2-8 then moved table before table 2-7 and updated figures 2-13, 2-14, 2-15, 2-16
- in section 2.9, removed subsections 2.9.1 and 2.9.2 and added new table 2-8
- in table 2-9, removed "except I2C" from hardware standby with PWDNB description
- in section 2.10,updated figure 2-18, added new figure 2-20 and new table 2-10
- in sub-section 4.7.1, add registers 0x3B00[2:0] and 0x3B00[7] to paragraph and removed table 4-9
- in sub-section 4.7.1.1, add register 0x3B00[5:4] to paragraph
- in sub-section 4.7.1.2, added "For longer strobe pulse..."
- in section 4, updated figures 4-4, 4-5, 4-6, 4-7 and 4-9
- in table 7-11, updated register descriptions for 0x3B02 and 0x3B03 by changing "Strobe" to "LED1 and LED2"
- in table 8-4, changed "XVCLK" to "EXTCLK"

- in table 9-1, changed pad 29 from "PCLK" to "GPIO_2", pad 72 from "GPIO" to "GPIO_0", pad 73 from "HREF" to "GPIO_2"

version 1.21 01.19.2011

- in table 2-1, changed max frame rate with MIPI for VGA to 200 and for QVGA to 400
- replaced figure 3-2

version 1.22 03.01.2011

- in section 1, updated figure 1-1
- in section 9, updated signal names for pad numbers 10 and 11

version 1.3 04.01.2011

- on page i, under key specifications, changed power supply core from "1.2VDC ± 5%" to "1.14~1.32V", dynamic range value from "71.6" to "67" and sensitivity from "780" to "864"
- in table 1-1, added LVDS to pin descriptions for pin numbers 35, 36, 37, 38, 39 , 40, 45, 46, 47, 48
- in table 2-6, added t8, t9, t10, and t11
- in table 2-8, added t7, t8, t9 and t10
- in section 2, updated figures 2-1, 2-10 to 2-16, 2-21 and added new figure 2-20
- in table 2-10, added PLL3_prediv, PLL3_mult1, PLL3_mult2, PLL3_divs, PLL3_div and REF_CLK
- in table 2-11, added PLL3_mult2 output and REF_CLK
- in sub-section 2.11.5, added "The max size..." and "OV8830 supports four..."
- in table 2-13, updated bit descriptions for registers 0x3208[7:4], 0x320B[3:2]
- in table 4-1, updated bit description for register 0x3820 bit[6] and bit[1]
- in table 4-6, added bit descriptions for register 0x4000[4:0]
- in section 4.5, added "OmniVision reserves the last..."
- in table 4-7, updated bit descriptions for registers 0x3D80, 0x3D81,0x3D84 and changed register 0x3D1F to 0x3D0F
- in table 4-9, removed bit[1] and updated bit description for register 0x3B04[0]
- in section 4, added new sub-section 4.7.2.6
- in table 5-1, added bit[5] to register 0x5041
- in table 5-5, added register 0x3406
- in section 7, extensive changes to register descriptions
- changed default values for registers 0x3406, 0x3804, 0x3806, 0x3808, 0x380A, 0x4006, 0x4813, 0x4816, 0x4819, 0x481D, 0x4821, 0x482A, 0x483C, 0x4842, 0x484A~0x484F, 0x5000, 0x5041, 0x5800-0x5808, 0x5822~0x5828, and 0x583C throughout the document
- in table 8-3, changed max value for V_{DD-D} (4-lane MIPI) from "1.26" to "1.32"
- in table 8-3, changed table title to DC characteristics (-30°C < T_J < 70°C)
- in table 8-3, added footnote b to standby current stating "standby current is measured at room temperature"

version 1.4 05.03.2011

- in table 8-3, replaced TBDs, changed $I_{DDS-SCCB}$ standby current from "700" to "300", added new footnote b and updated new footnote c
- in section 10, added section 10.3

version 1.41 06.06.2011

- on page i, updated ordering information

the clear advantage™

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

FINLAND

Mouhijärvi + 358 3 341 1898

GERMANY

Munich +49 89 63 81 99 88

CHINA

Beijing + 86 10 6580 1690

Shanghai + 86 21 6175 9888

Shenzhen + 86 755 8384 9733

Hong Kong + 852 2403 4011

JAPAN

Tokyo + 81 3 5765 6321

KOREA

Seoul + 82 2 3478 2812

SINGAPORE + 65 6562 8250

TAIWAN

Taipei + 886 2 2657 9800 - ext.#100

website: **www.ovt.com**