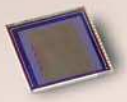
OmniVision



datasheet

PRELIMINARY SPECIFICATION

1/4" color CMOS QSXGA (5 megapixel) image sensor
with OmniBSI™ technology

OV5647

color CMOS QSXGA (5 megapixel) image sensor with OmniBSI™ technology

datasheet (COB)
PRELIMINARY SPECIFICATION

version 1.0
november 2009

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

applications

- cellular phones
- toys
- PC multimedia
- digital still cameras

ordering information

- **OV05647-G04A** (color, chip probing, 200 µm backgrinding, reconstructed wafer)

features

- 1.4 µm x 1.4 µm pixel with OmniBSI technology for high performance (high sensitivity, low crosstalk, low noise)
- optical size of 1/4"
- automatic image control functions: automatic exposure control (AEC), automatic white balance (AWB), automatic band filter (ABF), automatic 50/60 Hz luminance detection, and automatic black level calibration (ABLC)
- programmable controls for frame rate, AEC/AGC 16-zone size/position/weight control, mirror and flip, cropping, windowing, and panning
- image quality controls: lens correction, defective pixel canceling
- support for output formats: 8-/10-bit raw RGB data
- support for video or snapshot operations
- support for LED and flash strobe mode
- support for internal and external frame synchronization for frame exposure mode
- support for horizontal and vertical sub-sampling
- standard serial SCCB interface
- digital video port (DVP) parallel output interface
- MIPI interface (two lanes)
- 32 bytes of embedded one-time programmable (OTP) memory
- on-chip phase lock loop (PLL)
- embedded 1.5V regulator for core power
- programmable I/O drive capability, I/O tri-state configurability
- support for black sun cancellation

key specifications

- **active array size:** 2592 x 1944
- **power supply:**
  core: 1.5V ± 5% (with embedded 1.5V regulator)
  analog: 2.6 ~ 3.0V (2.8V typical)
  I/O: 1.7V ~ 3.0V
- **power requirements:**
  active: TBD
  standby: TBD
- **temperature range:**
  operating: -30°C to 70°C (see **table 8-2**)
  stable image: 0°C to 50°C (see **table 8-2**)
- **output formats:** 8-/10-bit RGB RAW output
- **lens size:** 1/4"
- **lens chief ray angle:** 24° (see **figure 10-2**)
- **input clock frequency:** 6~27 MHz
- **S/N ratio:** TBD
- **dynamic range:** TBD
- **maximum image transfer rate:**
  QSXGA (2592 x 1944): 15 fps
  1080p: 30 fps
  960p: 45 fps
  720p: 60 fps
  VGA (640 x 480): 90 fps
  QVGA (320 x 240): 120 fps
- **sensitivity:** TBD
- **shutter:** rolling shutter / global shutter
- **maximum exposure interval:** 1968 x t_{ROW}
- **pixel size:** 1.4 µm x 1.4 µm
- **well capacity:** TBD
- **dark current:** TBD
- **fixed pattern noise (FPN):** TBD
- **image area:** 3673.6 µm x 2738.4 µm
- **die dimensions:** 5520 µm x 4700 µm

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 OV5647 image sensor. The die information is shown in **section 9**.

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

pad number	signal name	pad type	description
1	AVDD	power	power for analog circuit, 2.8V
2	AGND	power	ground for analog circuit
3	DOGND	power	ground for digital I/O
4	SCL	input	SCCB clock input
5	SDA	I/O	SCCB data I/O
6	DVDD	power	power for digital core circuit, 1.5V (connect to 0.1uF capacitor to ground)
7	SGND	power	ground for pixel array
8	GPIO1	I/O	GPIO 1
9	GPIO0	I/O	GPIO 0
10	STROBE	I/O	strobe output
11	FREX	I/O	frame exposure control
12	DOVDD	power	power for digital I/O, 1.7 ~ 3.0V
13	VREF2	reference	reference analog circuit (connect to 0.1uF capacitor to ground)
14	VREF1	reference	reference for analog circuit (connect to 0.1uF capacitor to ground)
15	PWDN	input	power down control (active high with internal pull-down resistor)
16	DVDD	power	power for digital core circuit, 1.5V (connect to 0.1uF capacitor to ground)
17	RESETB	input	hardware reset (active low with internal pull-up resistor)
18	AVDD	power	power for analog circuit, 2.8V
19	AGND	power	ground for analog circuit
20	TM	input	test mode (active high with internal pull down resistor)
21	DOGND	power	ground for digital I/O
22	DVDD	power	power for digital core circuit, 1.5V (connect to 0.1uF capacitor to ground)

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

pad number	signal name	pad type	description
23	DVDD	power	power for digital core circuit, 1.5V (connect to 0.1uF capacitor to ground)
24	DOVDD	power	power for digital I/O, 1.7 ~ 3.0V
25	DOGND	power	ground for digital I/O
26	AVDD	power	power for analog circuit, 2.8V
27	HREF	I/O	DVP HREF output
28	PCLK	I/O	DVP PCLK output
29	VSYNC	I/O	DVP VSYNC output
30	DOVDD	power	power for digital I/O, 1.7 ~ 3.0V
31	D0	I/O	DVP data bit 0
32	D1	I/O	DVP data bit 1
33	D2	I/O	DVP data bit 2
34	D3	I/O	DVP data bit 3
35	D9/MDN0	I/O	DVP data bit 9/ MIPI data lane0 negative output
36	D8/MDP0	I/O	DVP data bit 8/ MIPI data lane0 positive output
37	EVDD	power	power for MIPI circuit, 1.5V (connect to DVDD)
38	D7/MCN	I/O	DVP data bit 7/ MIPI clock negative output
39	D6/MCP	I/O	DVP data bit 6/ MIPI clock positive output
40	EGND	power	ground for MIPI TX circuit
41	D5/MDN1	I/O	DVP data bit 5/ MIPI data lane1 negative output
42	D4/MDP1	I/O	DVP data bit 4/ MIPI data lane1 positive output
43	EGND	power	ground for MIPI TX circuit
44	PVDD	power	power for PLL circuit, 2.8V (connect to AVDD)
45	XCLK	input	system input clock
46	DOVDD	power	power for digital I/O, 1.7 ~ 3.0V
47	DVDD	power	power for digital core circuit, 1.5V (connect to 0.1uF capacitor to ground)
48	DOGND	power	ground for digital I/O
49	AVDD	power	power for analog circuit, 2.8V
50	AGND	power	ground for analog circuit

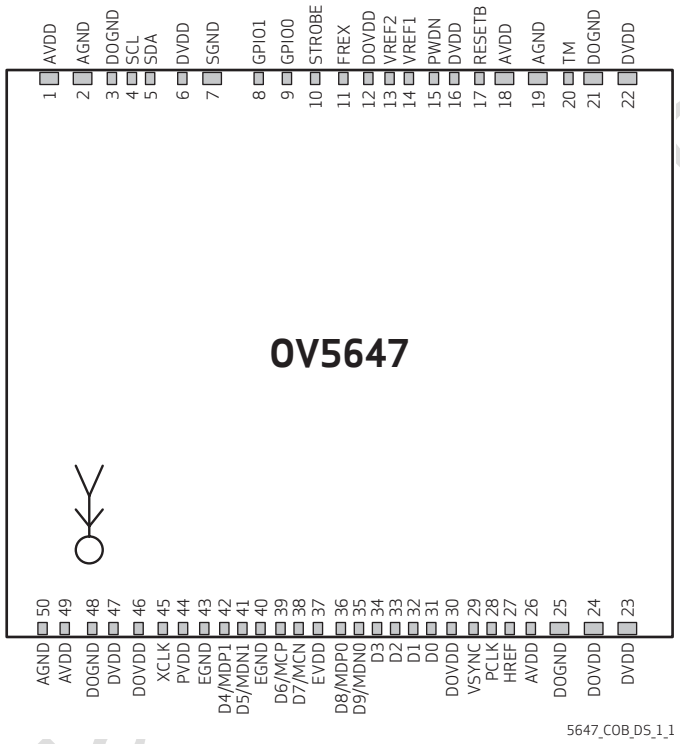
table 1-2 pad configuration under various conditions

signal	RESET[a]	RESET[b]	post-RESET	software sleep	hardware standby (power down pin = 1)
VSYNC	high-z	high-z	input by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
HREF	high-z	high-z	input by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
PCLK	high-z	high-z	input by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
D[9:0]	high-z	high-z	input by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
FREX	high-z	high-z	input by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
STROBE	high-z	high-z	input by default (configurable)	high-z by default (configurable)	high-z by default (configurable)
XCLK	high-z	input	input	input	high-z
SIOD	open drain	I/O	I/O	I/O	open drain
SIOC	high-z	input	input	input	high-z
MCP	0	output	output	0	0
MCN	0	output	output	0	0
MDP0	high-z	high-z	output	high-z	high-z
MDN0	high-z	high-z	output	high-z	high-z
MDP1	high-z	high-z	output	high-z	high-z
MDN1	high-z	high-z	output	high-z	high-z

a. some customer assume PWDN pin = 1 when chip power up

b. PWDN pin = 0 when chip power up

figure 1-1 pad diagram



2 system level description

2.1 overview

The OV5647 is a low voltage, high performance, 5 megapixel CMOS image sensor that provides 2592x1944 video output using OmniBSI™ technology. It provides multiple resolution raw images via the control of the serial camera control bus or MIPI interface.

The OV5647 has an image array capable of operating up to 15 fps in 2592x1944 resolution with user control of image quality, data transfer, camera functions through the SCCB interface. The OV5647 uses innovative OmniBSI technology to improve the sensor performance without the physical and optical trade-off.

For customized application, the OV5647 includes a one-time programmable (OTP) memory.

2.2 architecture

The OV5647 sensor core generates streaming pixel data at a constant frame rate, indicated by HREF and VSYNC. **figure 2-1** shows the functional block of the OV5647 image sensor. **figure 2-2** shows an example application of the OV5647 sensor.

figure 2-1 OV5647 block diagram

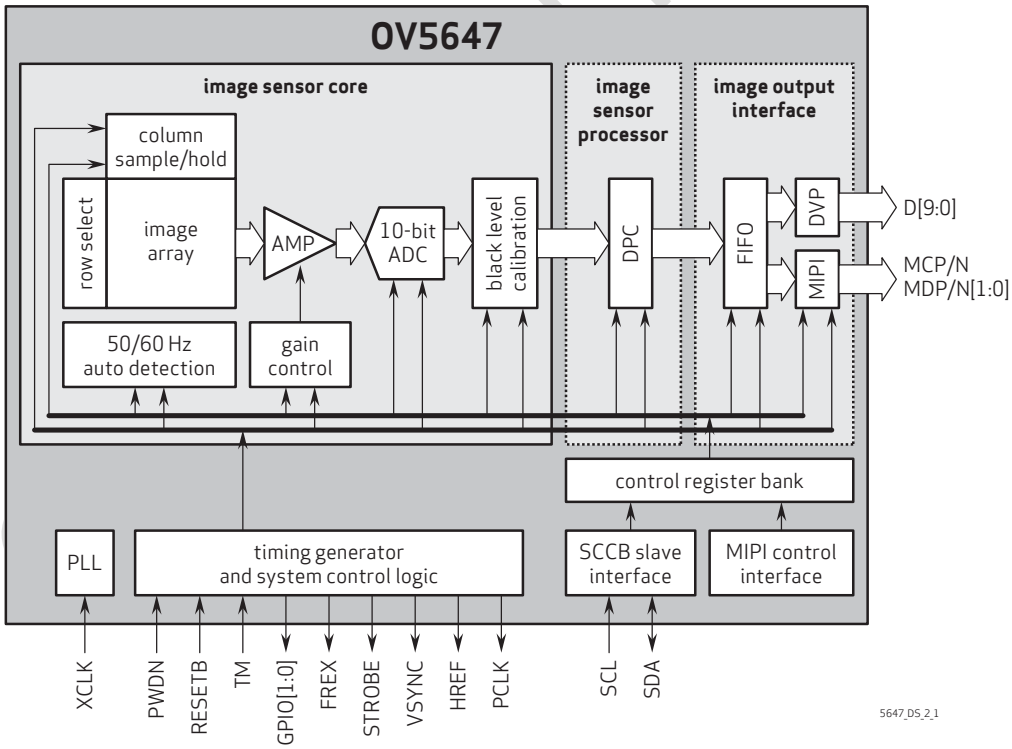
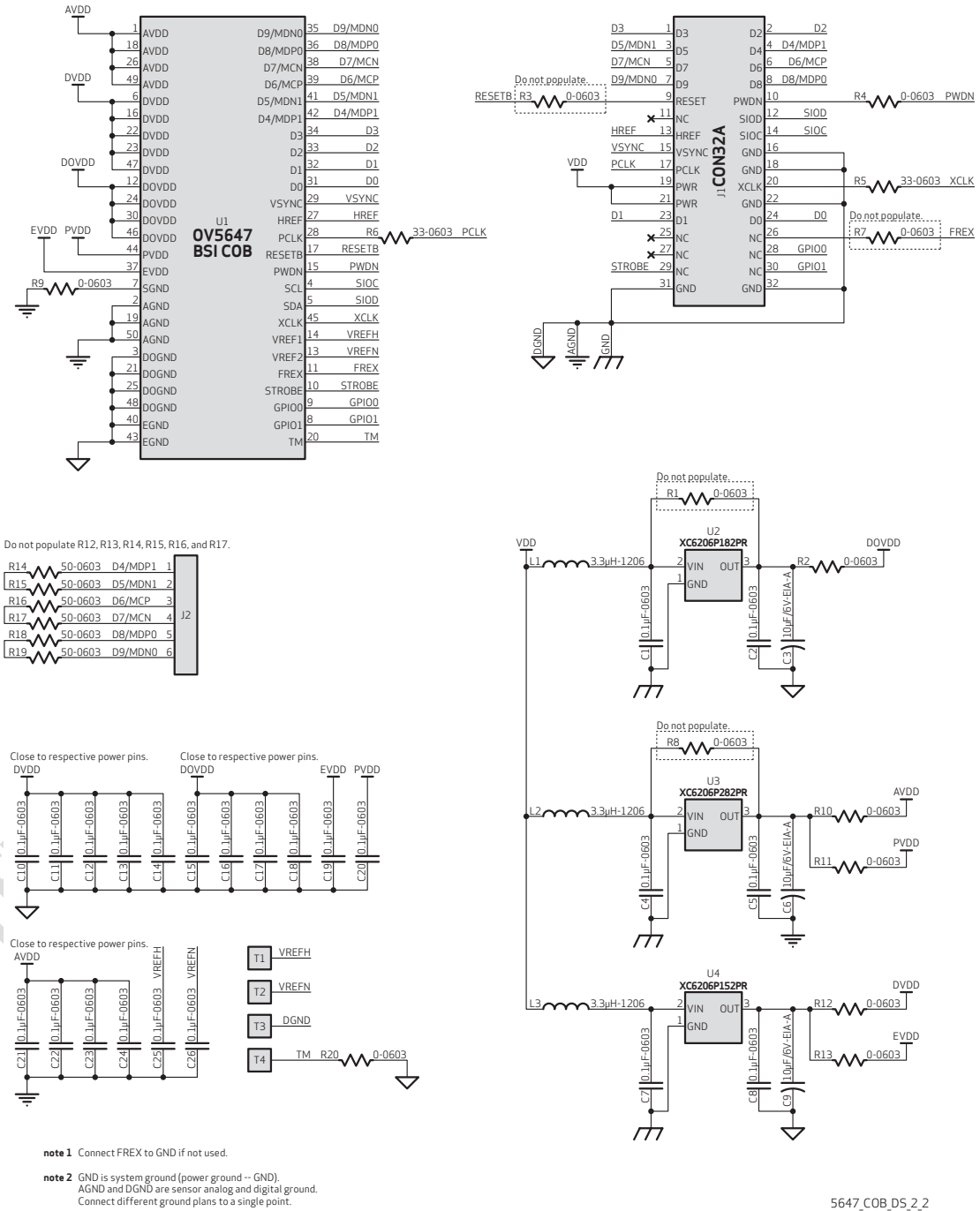


figure 2-2 reference design schematic



5647_COB_DS_2_2

2.3 format and frame rate

table 2-1 format and frame rate

format	resolution	frame rate	scaling method	pixel clock
5 Mpixel	2592x1944	15 fps	full resolution	80 MHz
1080p	1920x1080	30 fps	cropping	68 MHz
960p	1280x960	45 fps	cropping, subsampling/ binning	91.2 MHz
720p	1280x720	60 fps	cropping, subsampling/ binning	92 MHz
VGA	640x480	90 fps	cropping, subsampling/ binning	46.5 MHz
QVGA	320x240	120 fps	cropping, subsampling/ binning	32.5 MHz

2.4 I/O control

2.4.1 system clock control

The PLL is inside the chip which generates a default 96 MHz clock from 6~27 MHz input clock. An inside programmable clock divider is used to generate different frame rate timing.

2.5 power up sequence

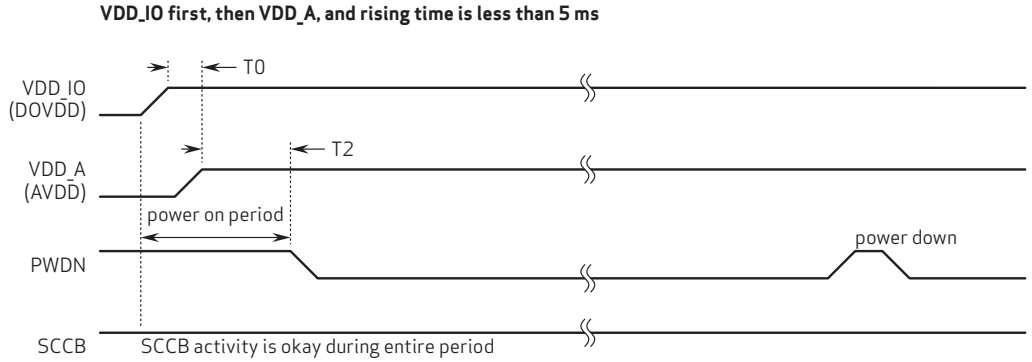
Based on the system power configuration (1.8V or 2.8V for I/O power), using external DVDD or internal DVDD, the power up sequence will differ. If 1.8V is used for I/O power, using the internal DVDD is preferred. If 2.8V is used for I/O power, due to a high voltage drop at the internal DVDD regulator, there is a potential heat issue. Hence, for a 2.8V power system, OmniVision recommends using an external DVDD source. Due to the higher power down current when using an external DVDD source, OmniVision strongly recommends cutting off all power supplies, including the external DVDD, when the sensor is not in use in the case of 2.8V I/O and external DVDD.

2.5.1 power up with internal DVDD

For powering up with the internal DVDD and SCCB access during the power ON period, the following conditions must occur:

1. if V_{DD-IO} and V_{DD-A} are turned ON at the same time, make sure V_{DD-IO} becomes stable before V_{DD-A} becomes stable
2. PWDN is active high with an asynchronized design (does not need clock)
3. PWDN must go high during the power up period
4. for PWDN to go low, power must first become stable (AVDD to PWDN $\geq$ 5 ms)
5. RESETB is active low with an asynchronized design
6. state of RESETB does not matter during power up period once DOVDD is up
7. master clock XCLK should provide at least 1 ms before host accesses sensor's SCCB
8. host can access SCCB bus (if shared) during entire period. 20 ms after PWDN goes low or 20 ms after RESETB goes high if reset is inserted after PWDN goes low, host can access sensor's SCCB to initialize sensor

figure 2-3 power up timing with internal DVDD



note T0 ≥ 0 ms: delay from VDD_IO stable to VDD_A stable
T2 ≥ 5 ms: delay from VDD_A stable to sensor power up stable

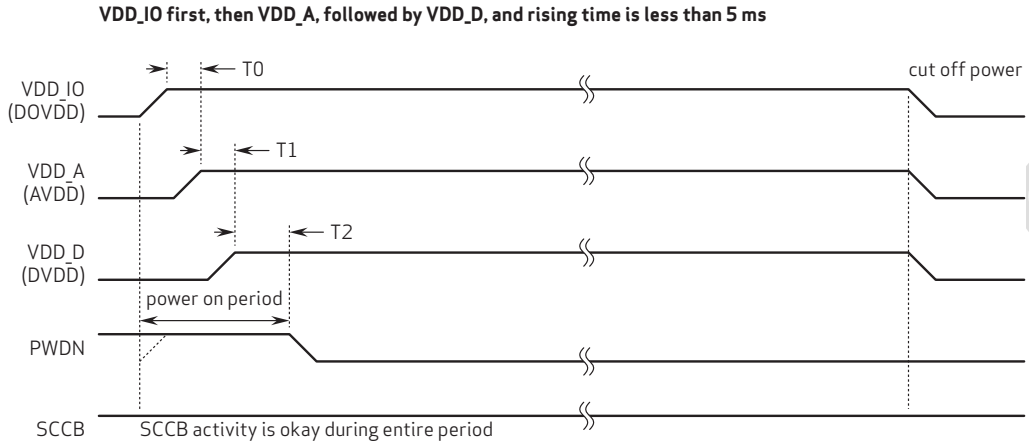
5647_DS_2_3

2.5.2 power up with external DVDD source

For powering up with an external DVDD source and SCCB access during the power ON period, the following conditions must occur:

1. if V_{DD-IO} and V_{DD-A} are turned ON at the same time, make sure V_{DD-IO} becomes stable before V_{DD-A} becomes stable
2. if V_{DD-A} and V_{DD-D} are turned ON at the same time, make sure V_{DD-A} becomes stable before V_{DD-D} becomes stable
3. PWDN is active high with an asynchronized design (does not need clock)
4. for PWDN to go low, power must first become stable (DVDD to PWDN ≥ 5 ms)
5. all powers are cut off when the camera is not in use (power down mode is not recommended
6. RESETB is active low with an asynchronized design
7. state of RESETB does not matter during power up period once DOVDD is up
8. master clock XVCLK should provide at least 1 ms before host accesses sensor's SCCB
9. host can access SCCB bus (if shared) during entire period. 20 ms after PWDN goes low or 20 ms after RESETB goes high if reset is inserted after PWDN goes high, host can access sensor's SCCB to initialize sensor

figure 2-4 power up timing with external DVDD source



note T0 ≥ 0 ms: delay from VDD_IO stable to VDD_A stable
T1 ≥ 0 ms: delay from VDD_A stable to VDD_D stable
T2 ≥ 5 ms: delay from VDD_D stable to sensor power up stable

5647_DS_2_4

2.6 reset

Two reset modes are available for the OV5647:

- hardware reset
- SCCB software reset

The OV5647 sensor includes a **RESETB** pad that forces a complete hardware reset when it is pulled low (GND). The OV5647 clears all registers and resets them to their default values when a hardware reset occurs. A reset can also be initiated through the SCCB interface by setting register 0x0103[0] to high.

The whole chip will be reset during power up. Manually applying a hard reset upon power up is recommended even though the on-chip power up reset is included. The hard reset is active low with an asynchronized design. The reset pulse width should be greater than or equal to 1 ms.

2.7 standby and sleep

Two suspend modes are available for the OV5647:

- hardware standby
- SCCB software sleep

To initiate hardware standby mode, the **PWDN** pad must be tied to high. When this occurs, the OV5647 internal device clock is halted and all internal counters are reset and registers are maintained. Executing a software sleep (0x0100[0]) through the SCCB interface suspends internal circuit activity but does not halt the device clock. All register content is maintained in both modes.

3 block level description

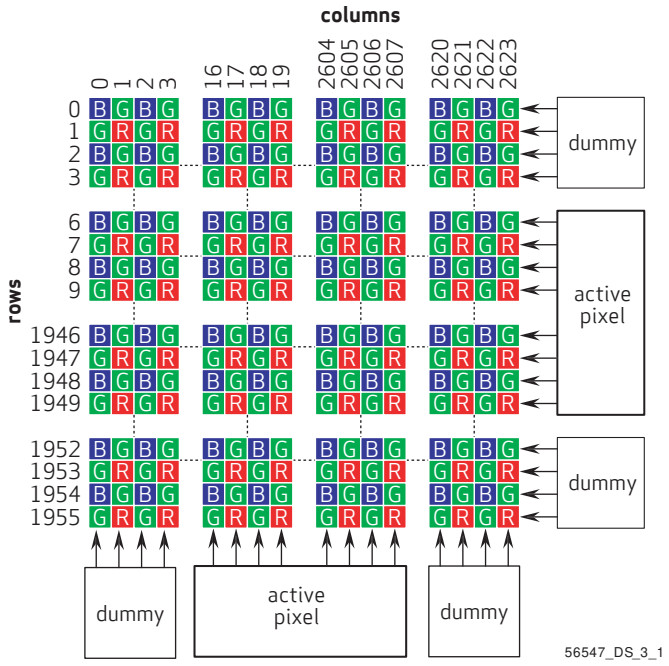
3.1 pixel array structure

The OV5647 sensor has an image array of 2624 columns by 1956 rows (5,132,544 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 5,132,544 pixels, 5,038,848 (2592x1944) are active pixels and can be output. The other pixels are used for black level calibration and interpolation. The center 2592x1944 is suggested to be output from the whole active pixel array. The backend processor can use the boundary pixels for additional processing.

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

figure 3-1 sensor array region color filter layout



3.2 binning

The OV5647 supports 2x2 binning for better SNR in low light conditions. See **table 3-1** for horizontal and vertical binning registers.

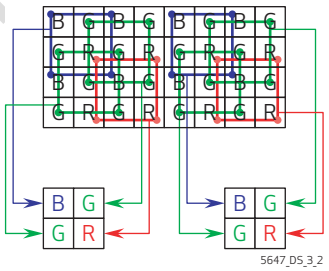
table 3-1 horizontal and vertical binning registers

address	register name	default value	R/W	description
0x3820	TIMING_TC_REG20	0x40	RW	Bit[0]: Vertical binning 0: Disable 1: Enable
0x3821	TIMING_TC_REG21	0x00	RW	Bit[0]: Horizontal binning 0: Disable 1: Enable

Sub-sampling is necessary when using binning.

Sensor timing adjustment is necessary after applying binning. Please consult your local OmniVision FAE for details.

figure 3-2 example of 2x2 binning



3.3 analog amplifier

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

3.4 10-bit A/D converters

The balanced signal is then digitized by the on-chip 10-bit ADC. It can operate at up to 27 MHz and is fully synchronous to the pixel clock. The actual conversion rate is determined by the frame rate.

4 image sensor core digital functions

4.1 mirror and flip

The OV5647 provides mirror and flip read-out modes, which respectively reverse the sensor data read-out order horizontally and vertically (see **figure 4-1**). In flip mode, the OV5647 does not need additional settings because the ISP block will auto-detect whether the pixel is in the red line or blue line and make the necessary adjustments.

figure 4-1 mirror and flip samples

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

5647_DS_4_1

table 4-1 mirror flip control registers

address	register name	default value	R/W	description
0x3820	TIMING_TC_REG20	0x40	RW	Timing Control Bit[2]: r_vflip_isp Bit[1]: r_vflip_snr
0x3821	TIMING_TC_REG20	0x00	RW	Timing Control Bit[2]: r_mirror_isp Bit[1]: r_mirror_snr

4.2 image windowing

An image windowing area is defined by four parameters, x_addr_start, x_addr_end, y_addr_start, y_addr_end. By properly setting the parameters, any portion or size within the sensor array can be defined as an visible area. This windowing is achieved by simply masking the pixels outside the defined window; thus, it will not affect the original timing.

figure 4-2 image windowing

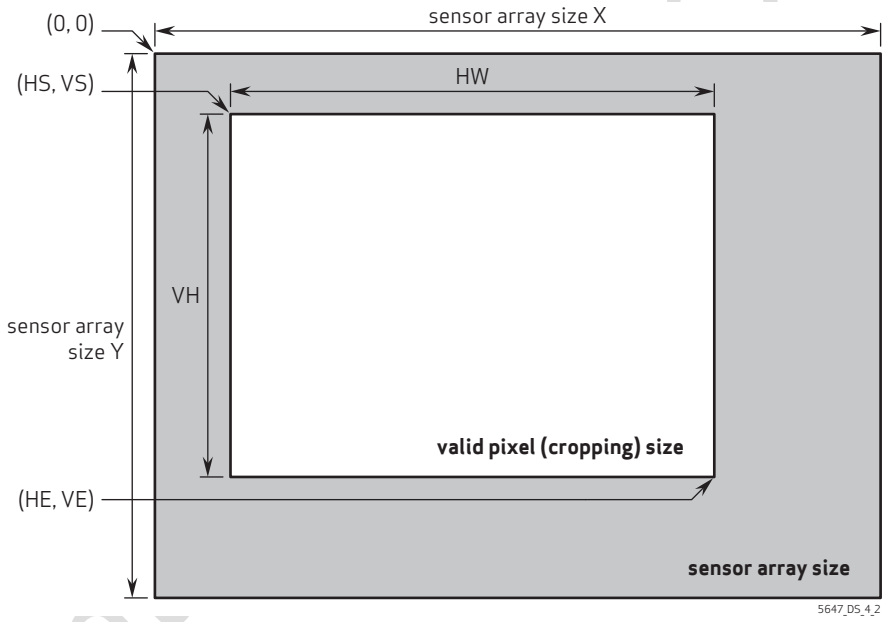


table 4-2 image windowing registers

address	register name	default value	R/W	description
0x3800	TIMING_X_ADDR_START	0x00	RW	Bit[3:0]: x_addr_start[11:8]
0x3801	TIMING_X_ADDR_START	0x0C	RW	Bit[7:0]: x_addr_start[7:0]
0x3802	TIMING_Y_ADDR_START	0x00	RW	Bit[3:0]: y_addr_start[11:8]
0x3803	TIMING_Y_ADDR_START	0x04	RW	Bit[7:0]: y_addr_start[7:0]
0x3804	TIMING_X_ADDR_END	0x0A	RW	Bit[3:0]: x_addr_end[11:8]
0x3805	TIMING_X_ADDR_END	0x33	RW	Bit[7:0]: x_addr_end[7:0]
0x3806	TIMING_Y_ADDR_END	0x07	RW	Bit[3:0]: y_addr_end[11:8]
0x3807	TIMING_Y_ADDR_END	0xA3	RW	Bit[7:0]: y_addr_end[7:0]

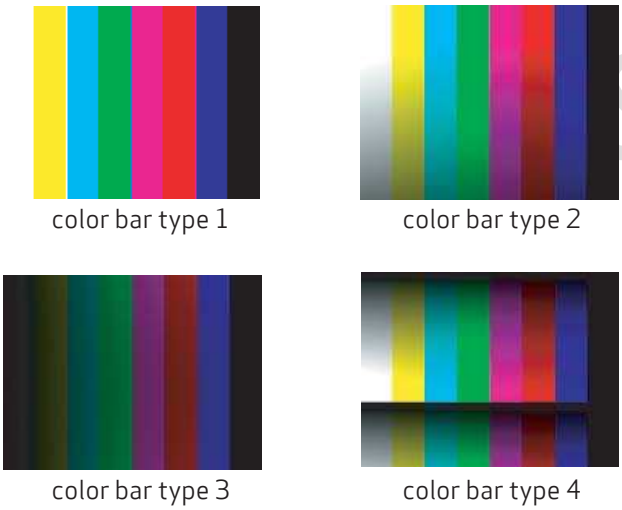
4.3 test pattern

For testing purposes, the OV5647 offers three types of test patterns, color bar, square and random data.The OV5647 also offers two effects: transparent effect and rolling bar effect. The output type of test pattern is controlled by register 0x503D[1:0] register (test_pattern_type).

4.3.1 color bar

There are four types of color bars shown in **figure 4-3**. The output type of color the color bar can be selected by bar style register 0x503D[3:2].

figure 4-3 color bar types

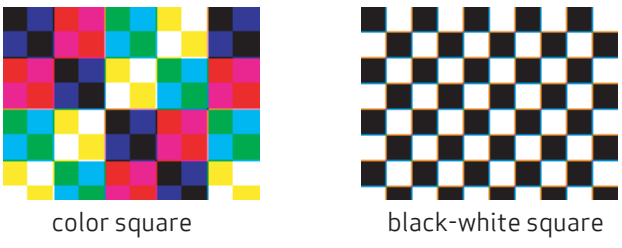


5647_DS_4_3

4.3.2 square

There are two types of square: color square and black-white square. Register 0x503D[4] (squ_bw) determines which type of square will be output.

figure 4-4 color, black and white square bars



5647_DS_4_4

4.3.3 random data

There are two types of random data test pattern: frame-changing and frame-fixed random data. The output type of random data is decided by register 0x503E[4] (rnd_same). The random seed is set by register 0x503E[3:0] (rnd_seed).

4.3.4 transparent effect

The transparent effect is enabled by register 0x503D[5] (transparent_mode). If this register is set, the transparent test pattern will be gotten. **figure 4-5** is a example which shows a transparent color bar image.

figure 4-5 transparent effect

5647_DS_4_5

4.3.5 rolling bar effect

The rolling bar is set by register 0x503D[6] (rolling_bar). If it is set, an inverted-color rolling bar will roll from up to down. **figure 4-6** is a example which shows a rolling bar on color bar image.

figure 4-6 rolling bar effect

rolling bar effect

5647_DS_4_6

table 4-3 test pattern registers

address	register name	default value	R/W	description
0x503D	ISP CTRL3D	0x00	RW	Bit[7]: test_pattern_en 0: Disable 1: Enable Bit[6]: rolling_bar 0: Disable rolling bar 1: Enable rolling bar Bit[5]: transparent_mode 0: Disable 1: Enable Bit[4]: squ_bw_mode 0: Output square is color square 1: Output square is black-white square Bit[3:2]: bar_style When set to different value, the different type color bar will be output Bit[1:0]: test_pattern_type 00: Color bar 01: Square 10: Random data 11: Input data
0x503E	ISP CTRL3E	0x00	RW	Bit[6]: win_cut_en Bit[5]: isp_test 0: Two lowest bits are 1 1: Two lowest bits are 0 Bit[4]: rnd_same 0: Frame changing random data pattern 1: Frame-fixed random data pattern Bit[3:0]: rnd_seed Initial seed for random data pattern

4.4 50/60Hz detection

When the integration time is not an integer multiple of the period of light intensity, the image will flicker. The function of the detector is to detect whether the sensor is under a 50 Hz or 60 Hz light source so that the basic step of integration time can be determined. Contact your local OmniVision FAE for auto detection settings.

table 4-4 50/60 Hz detection control registers

address	register name	default value	R/W	description
0x3C00	50/60 HZ DETECTION CTRL00	0x00	RW	Bit[5:3]: 50/60 Hz detection control Contact local OmniVision FAE for the correct settings Bit[2]: band_def Band50 default value 0: 60 Hz as default value 1: 50 Hz as default value Bit[1:0]: 50/60 Hz detection control Contact local OmniVision FAE for the correct settings
0x3C01	50/60 HZ DETECTION CTRL01	0x00	RW	Bit[7]: band_man_en Band detection manual mode 0: Manual mode disable 1: Manual mode enable Bit[6:0]: 50/60 Hz detection control Contact local OmniVision FAE for the correct settings
0x3C02~ 0x3C0B	50/60 HZ DETECTION CTRL02	0x00	RW	Bit[7:0]: 50/60 Hz detection control Contact local OmniVision FAE for the correct settings
0x3C0C	50/60 HZ DETECTION CTRL0C	–	R	Bit[0]: band50 0: Detection result is 60 Hz 1: Detection result is 50 Hz

4.5 AEC and AGC algorithms

4.5.1 overview

The Auto Exposure Control (AEC) and Auto Gain Control (AGC) allows the image sensor to adjust the image brightness to a desired range by setting the proper exposure time and gain applied to the image. Besides automatic control, exposure time and gain can be set manually from external control. The related registers are listed in **table 4-5**

table 4-5 AEC/AGC control function registers

address	register name	default value	R/W	description
0x3500	EXPOSURE	0x00	RW	Bit[3:0]: Exposure[19:16]
0x3501	EXPOSURE	0x00	RW	Bit[7:0]: Exposure[15:8]
0x3502	EXPOSURE	0x20	RW	Bit[7:0]: Exposure[7:0]
0x3503	MANUAL CTRL	0x00	RW	Bit[5:4]: Gain latch timing delay x0: Gain has no latch delay 01: Gain delay of 1 frame 11: Gain delay of 2 frames Bit[2]: VTS manual 0: Auto enable 1: Manual enable Bit[1]: AGC manual 0: Auto enable 1: Manual enable Bit[0]: AEC manual 0: Auto enable 1: Manual enable
0x350A	AGC	0x00	RW	Bit[1:0]: Gain[9:8] AGC real gain output high byte
0x350B	AGC	0x00	RW	Bit[7:0]: Gain[7:0] AGC real gain output low byte
0x350C	VTS DIFF	0x06	RW	Bit[7:0]: vts_diff[15:8] When in manual mode, set to 0x00
0x350D	VTS DIFF	0x18	RW	Bit[7:0]: vts_diff[7:0] When in manual mode, set to 0x00

4.5.2 average-based algorithm

The average-based AEC controls image luminance using registers **WPT** (0x3A0F), **BPT** (0x3A10), **WPT2** (0x3A1B), and **BPT2** (0x3A1E). In average-based mode, the value of register **WPT** (0x3A0F) indicates the high threshold value for image change from unstable to stable state, and the value of register **BPT** (0x3A10) indicates the low threshold value for image change from unstable to stable state. The value of register **WPT2** (0x3A1B) indicates the high threshold value for image change from stable state to unstable state and the value of register **BPT2** (0x3A1E) indicates the low threshold value for image change from stable state to unstable state. When the target image luminance average value AVG (0x5693) is within the range specified by registers **WPT2** (0x3A1B) and **BPT2** (0x3A1E), the AEC keeps the image exposure and gain. When register AVG (0x5693) is greater than the value in register **WPT2** (0x3A1B), the AEC will decrease the image exposure and gain until it falls into the range of {0x3A10, 0x3A0F}. When register AVG (0x5693) is less than the value in register **BPT2** (0x3A1E), the AEC will increase the image exposure and gain until it falls into the range of {0x3A10, 0x3A0F}. Accordingly, the value in register **WPT** (0x3A0F) should be greater than the value in register **BPT** (0x3A10). The value of register **WPT2** should be no less than the value of register **WPT**(0x3A0F), and the value of register **BPT2** (0x3A1E) should be no greater than the value of **BPT** (0x3A10).

The AEC function supports both manual and auto speed selections in order to bring the image exposure into the range set by the values in registers **WPT** (0x3A0F) and **BPT** (0x3A10). For manual speed mode, the step is fixed and supports both normal and fast modes. AEC set to normal mode will allow for the slowest step increment or decrement in the image exposure to maintain the specified range. AEC set to fast mode will provide for an approximate ten-step increment or decrement in the image exposure to maintain the specified range. For auto speed mode, the step will automatically be adjusted according to the difference between the target and present values. The auto ratio of steps can be set by register bits AEC CTRL05[4:0] (0x3A05).

Register HIGH VPT (0x3A11) and register LOW VPT (0x3A1F) controls the fast AEC range in manual speed mode. If the target image AVG (0x5693) is greater than HIGH VPT (0x3A11), AEC will decrease by half. If register AVG (0x5693) is less than LOW VPT (0x3A1F), AEC will double, as shown in **figure 4-7**. These registers have no effect in auto speed mode.

figure 4-7 desired convergence

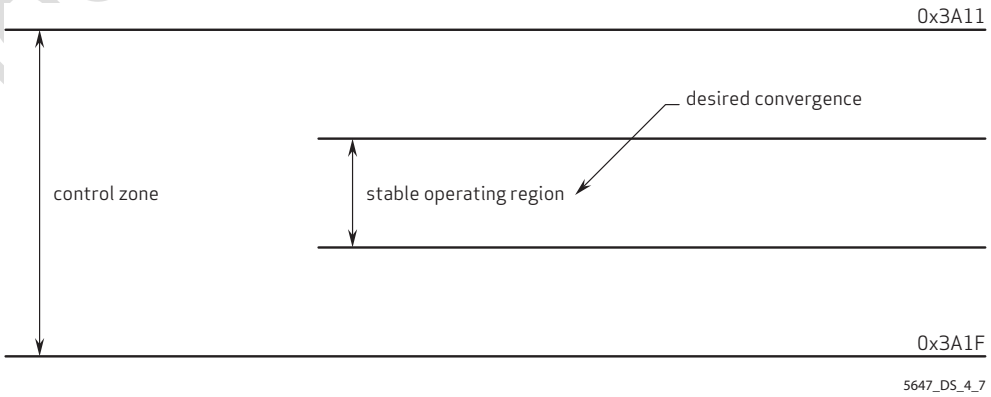


table 4-6 average based control function registers

address	register name	default value	R/W	description
0x3A0F	WPT	0x78	RW	Bit[7:0]: WPT Stable range high limit (enter)
0x3A10	BPT	0x68	RW	Bit[7:0]: BPT Stable range low limit (enter)
0x3A11	HIGH VPT	0xD0	RW	Bit[7:0]: vpt_high Fast zone high limit when step ratio auto mode is disabled
0x3A1B	WPT2	0x78	RW	Bit[7:0]: wpt2 Stable range high limit (from stable state to unstable state)
0x3A1E	BPT2	0x68	RW	Bit[7:0]: bpt2 Stable range low limit (from stable state to unstable state)
0x3A1F	LOW VPT	0x40	RW	Bit[7:0]: vpt_low Fast zone low limit when step ratio auto mode is disabled

For the average-based AEC/AGC algorithm, the measured window is horizontally and vertically adjustable and divided by sixteen (4x4) zones (see figure 4-5). Each zone (or block) is 1/16th of the image and has a 4-bit weight in calculating the average luminance (YAVG). The final YAVG is the weighted average of the sixteen zones. The 4-bit weight could be n/16 where n is from 0 to 15.

4.5.3 average luminance (YAVG)

Auto exposure time calculation is based on a frame brightness average value. By properly setting x_start, x_end, y_start, and y_end as shown in **figure 4-8**, a 4x4 grid average window is defined. It will automatically divide each zone into 4x4 zones. The average value is the weighted average of the 16 sections. **table 4-7** lists the corresponding registers.

figure 4-8 average-based window definition

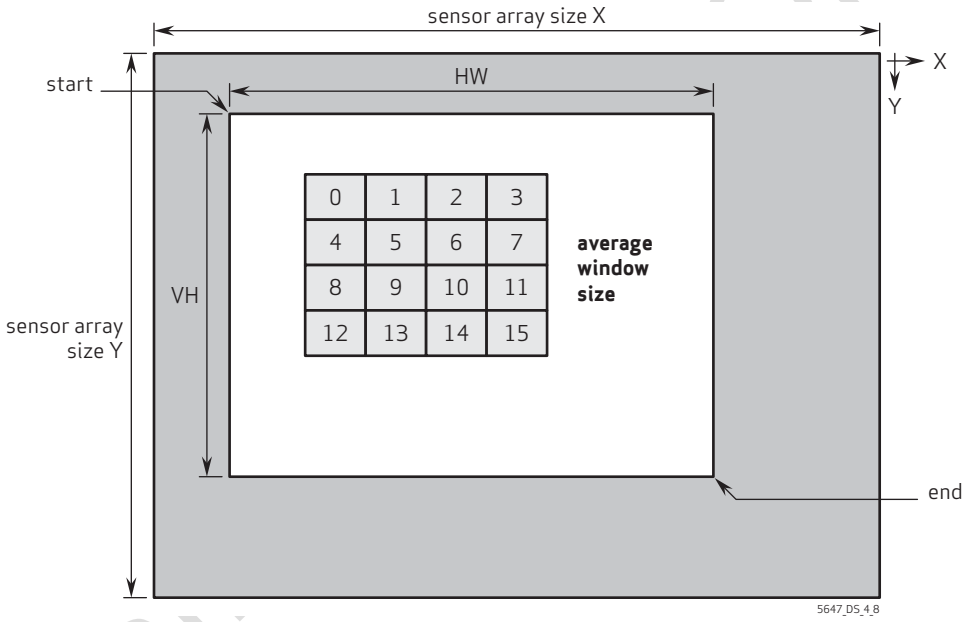


table 4-7 average luminance control function registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5680	XSTART	0x00	RW	Bit[3:0]: x_start[11:8] Horizontal start position for average window high byte
0x5681	XSTART	0x00	RW	Bit[7:0]: x_start[7:0] Horizontal start position for average window low byte
0x5682	YSTART	0x00	RW	Bit[3:0]: y_start[11:8] Vertical start position for average window low byte
0x5683	YSTART	0x00	RW	Bit[7:0]: y_start[7:0] Vertical start position for average window low byte
0x5684	X WINDOW	0x0A	RW	Bit[4:0]: Window X in manual average window mode high byte

table 4-7 average luminance control function registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5685	X WINDOW	0x20	RW	Bit[7:0]: Window X in manual average window mode low byte
0x5686	Y WINDOW	0x07	RW	Bit[3:0]: Window Y in manual average window mode high byte
0x5687	Y WINDOW	0x98	RW	Bit[7:0]: Window Y in manual average window mode low byte
0x5688	WEIGHT00	0x11	RW	Bit[7:4]: Window1 weight Bit[3:0]: Window0 weight
0x5689	WEIGHT01	0x11	RW	Bit[7:4]: Window3 weight Bit[3:0]: Window2 weight
0x568A	WEIGHT02	0x11	RW	Bit[7:4]: Window5 weight Bit[3:0]: Window4 weight
0x568B	WEIGHT03	0x11	RW	Bit[7:4]: Window7 weight Bit[3:0]: Window6 weight
0x568C	WEIGHT04	0x11	RW	Bit[7:4]: Window9 weight Bit[3:0]: Window8 weight
0x568D	WEIGHT05	0x11	RW	Bit[7:4]: Window11 weight Bit[3:0]: Window10 weight
0x568E	WEIGHT06	0x11	RW	Bit[7:4]: Window13 weight Bit[3:0]: Window12 weight
0x568F	WEIGHT07	0x11	RW	Bit[7:4]: Window15 weight Bit[3:0]: Window14 weight
0x5690	AVG CTRL10	–	R	Bit[1]: avg_opt Bit[0]: avg_man 0: Auto average window 1: Manual average window
0x5693	AVG READOUT	–	R	Bit[7:0]: avg value

4.6 AEC/AGC steps

The AEC and AGC work together to obtain adequate exposure/gain based on the current environmental illumination. In order to achieve the best signal-to-noise ratio (SNR), extending the exposure time is always preferred rather than raising the gain when the current illumination is getting brighter. Vice versa, under dark conditions, the action to decrease the gain is always taken prior to shortening the exposure time.

4.6.1 auto exposure control (AEC)

The function of the AEC is to calculate the necessary integration time of the next frame and send the information to the timing control block. Based on the statistics of previous frames, the AEC is able to determine whether the integration time should increase, decrease, fast increase, fast decrease, or remain the same.

In extremely bright situations, the LAEC activates, allowing integration time to be less than one row. In extremely dark situations, the night mode activates, allowing integration time to be larger than one frame.

To avoid image flickering under a periodic light source, the integration time can be adjusted in steps of integer multiples of the period of the light source.

4.6.2 LAEC

If the integration time is only one row period but the image is too bright, AEC will enter LAEC mode. LAEC ON/OFF can be set in register bit 0x3A00[6].

4.6.3 banding mode ON with AEC

In Banding ON mode, the exposure time will fall in steps of integer multiples of the period of light intensity.

Banding ON/OFF can be set in register 0x3A00[5].

For a given light flickering frequency, the band step can be expressed in units of row period.

The band steps for 50Hz and 60Hz light sources can be set in registers {0x3A08[1:0], 0x3A09[7:0]} and {0x3A0A[1:0], 0x3A0B[7:0]}, respectively.

- Banding mode OFF with AEC
- When banding mode is OFF, integration time increases/decreases as normal. It is not necessarily multiples of band steps.

4.6.4 night mode

The OV5647 supports long integration time such as 1 frame, 2 frames, 3 frames, 4 frames, 5 frames, 6 frames, 7 frames, and 8 frames in dark conditions. This is achieved by slowing down the original frame rate and waiting for exposure. Night mode ceiling can be set in register bits 0x3A02[15:8], 0x3A03[7:0]. Night mode can be disabled by setting register bit 0x3A00[2] to 0. Also, when in night mode, the increase and decrease step can be based on band or frames, depending on register 0x3A05[6]. The minimum increase/decrease step can be one band. The step can be based both on bands and frames.

4.6.5 auto gain control (AGC)

Unlike prolonging integration time, increasing gain will amplify both signal and noise. Thus, AGC usually starts after AEC is full. However, in cases where adjacent AEC step changes are too large (>1/16), AGC steps should be inserted in between. The AGC ceiling can be set in {0x3A18[1:0], 0x3A19[7:0]}.

4.7 black level calibration (BLC)

The pixel array contains several optically shielded (black) lines. These lines are used as reference for black level calibration. There are three main functions of the BLC:

- adjusting all normal pixel values based on the values of the black levels
- applying multiplication to all pixel values based on digital gain

table 4-8 BLC control functions

address	register name	default value	R/W	description
0x4000	BLC CTRL00	0x89	RW	BLC Control (0: disable, 1: enable) Bit[7]: blc_median_filter_enable Bit[3]: adc_11bit_mode Bit[2]: apply2blackline Bit[1]: blackline_averageframe Bit[0]: BLC enable
0x4002	BLC CTRL02	0x45	RW	Bit[7]: format_change_en format_change_i from fmt will be effect when it is enable Bit[6]: blc_auto_en Bit[5:0]: reset_frame_num
0x4005	BLC CTRL05	0x18	RW	Bit[5]: one_line_mode Bit[4]: remove_none_imagedata Bit[3]: blc_man_1_en Bit[2]: blackline_bggr_man_en 0: bgbg/grgr is decided by rblue/hswap 1: bgbg/grgr fix; Bit[1]: bgbg/grgr is decided by rblue/hswap blc_always_up_en 0: Normal freeze 1: BLC always update
0x4009	BLACK LEVEL	0x10	RW	Bit[7:0]: blc_blackleveltarget0

4.8 strobe flash and frame exposure

4.8.1 strobe flash control

The strobe signal is programmable. 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. 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 (see **table 4-9**).

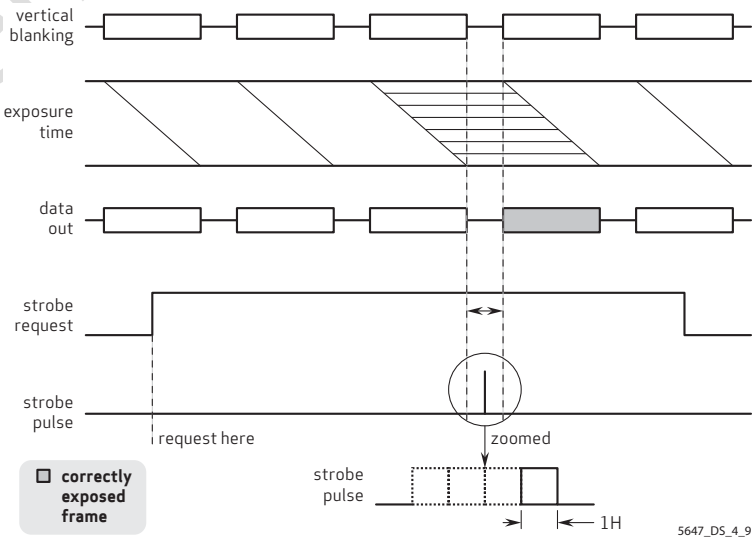
table 4-9 flashlight modes

mode	output	AEC / AGC	AWB
xenon	one-pulse	no	no
LED 1	pulse	no	no
LED 2	pulse	no	yes
LED 3	continuous	yes	yes

4.9 xenon flash control

After a strobe request is submitted, the strobe pulse will be activated at the beginning of the third frame (see **figure 4-9**). The third frame will be correctly exposed. The pulse width can be changed in Xenon mode between 1H and 4H, depending on register 0x3B00[3:2], where H is one row period.

figure 4-9 xenon flash mode



4.9.1 LED1 & 2 mode

Two frames after the strobe request is submitted, the third frame is correctly exposed. The strobe pulse will be activated only one time if the strobe end request is set correctly (see **figure 4-10**). If end request is not sent, the strobe signal is activated intermittently until the strobe end request is set (see **figure 4-11**). The number of skipped frames is programmable using registers {0x3A1C, 0x3A1D}.

figure 4-10 LED 1 & 2 mode - one pulse output

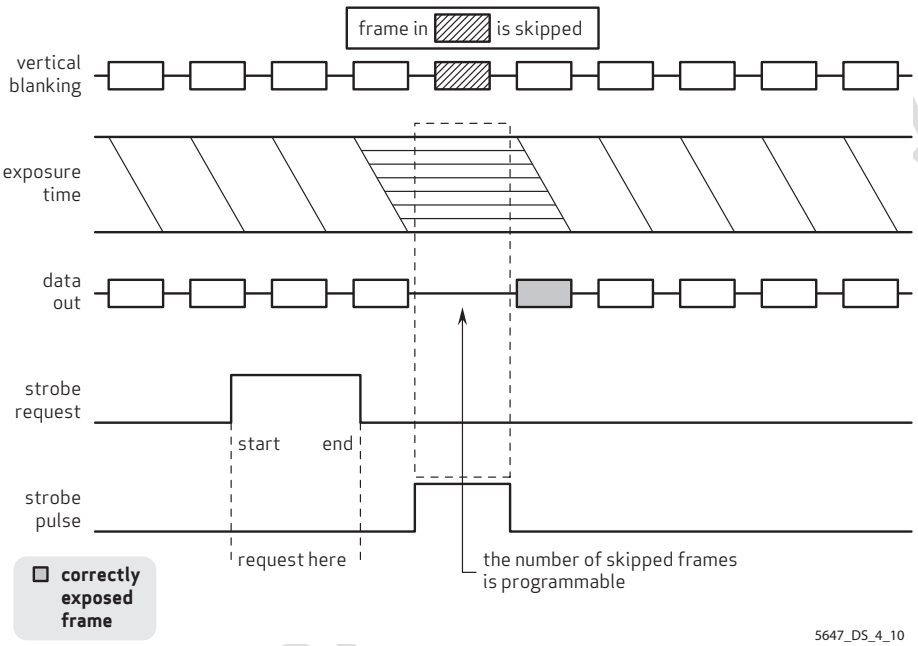
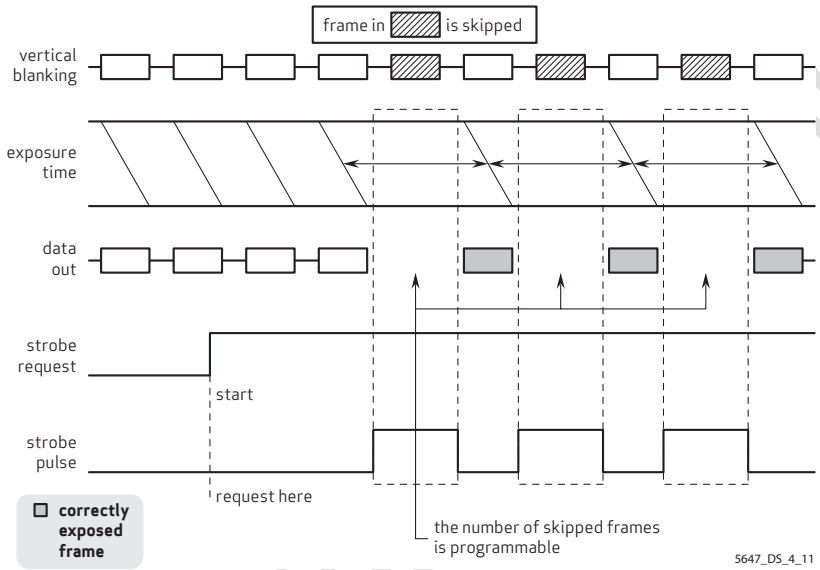


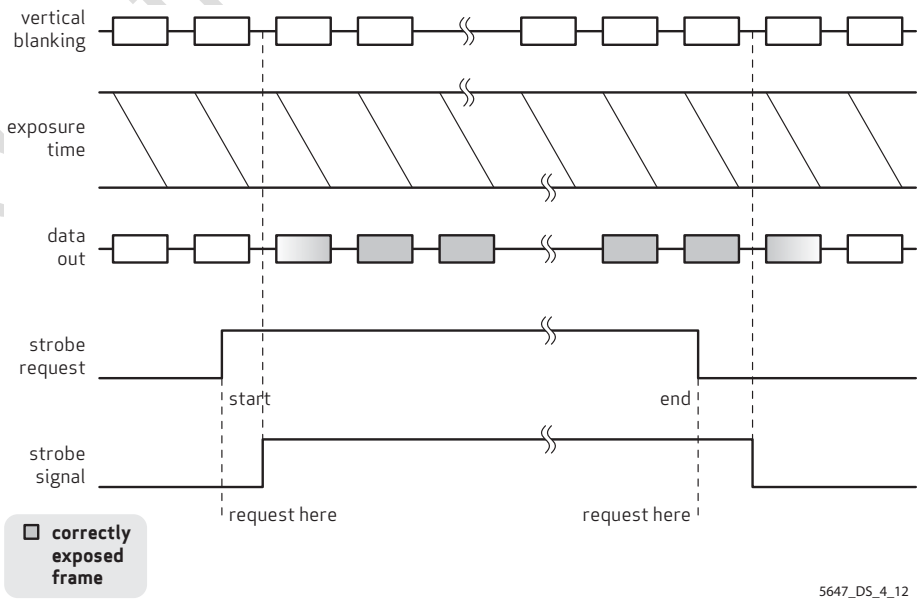
figure 4-11 LED 1 & 2 mode - multiple pulse output



4.9.2 LED 3 mode

In LED 3 mode, the strobe signal stays active until the strobe end request is sent (see figure 4-12).

figure 4-12 LED 3 mode



4.10 frame exposure (FREX) mode

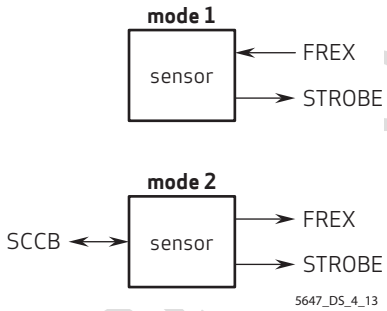
4.10.1 FREX control

In FREX mode, whole frame pixels start integration at the same time, rather than integrating row by row. After the user-defined exposure time (0x3B01, 0x3B04, 0x3B05), 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 OV5647 supports two modes of FREX (see **figure 4-13**):

- mode 1: Frame exposure and shutter control requests come from the external system via the FREX pin. The sensor will send a strobe output signal to control the flash light
- mode 2: Frame exposure request comes from the external system via the SCCB register 0x3B08[0]. The sensor will output two signals, shutter control signal through the FREX pin and strobe signal through the STROBE pin

figure 4-13 FREX modes



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.

4.11 FREX strobe flash control

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

table 4-10 FREX strobe control functions

address	register name	default value	R/W	description
0x3B00	STROBE CTRL	0x00	RW	Strobe Control Bit[7]: Strobe request ON/OFF 0: OFF/BLC 1: ON Bit[6]: Strobe pulse reverse Bit[3:2]: width_in_xenon 00: 1 row period 01: 2 row period 10: 3 row period 11: 4 row period Bit[1:0]: Strobe mode 00: xenon 01: LED 1 10: LED 2 11: LED 3
0x3B01	STROBE_FREX_EXP_H2	0x00	RW	Bit[7:0]: frex_exp[23:16]
0x3B02	STROBE_SHUTTER_DLY	0x08	RW	Bit[4:0]: shutter_dly[12:8]
0x3B03	STROBE_SHUTTER_DLY	0x00	RW	Bit[7:0]: shutter_dly[7:0]
0x3B04	STROBE_FREX_EXP_H	0x04	RW	Bit[7:0]: frex_exp[15:8]
0x3B05	STROBE_ FREX_EXP_L	0x00	RW	Bit[7:0]: frex_exp[7:0]
0x3B06	FREX CTRL	0x04	RW	FREX Control Bit[7:6]: frex_pchg_width Bit[5:4]: frex_strobe_option Bit[3:0]: frex_strobe_width[3:0]
0x3B07	STROBE_ FREX_MODE_SEL	0x08	RW	Bit[3]: fx1_fm_en Bit[2]: frex_inv Bit[1:0]: FREX mode select 00: frex_strobe mode0 01: frex_strobe mode1 1x: Rolling strobe
0x3B08	STROBE_FREX_EXP_REQ	0x00	RW	Bit[0]: frex_exp_req
0x3B09	FREX_SHUTTER_DELAY	0x00	RW	Bit[2:0]: frex end option
0x3B0A	STROBE_FREX_RST_ LENGTH	0x04	RW	Bit[2:0]: frex_rst_length[2:0]
0x3B0B	STROBE_WIDTH	0x00	RW	Bit[7:0]: frex_strobe_width[19:12]
0x3B0C	STROBE_WIDTH	0x3D	RW	Bit[7:0]: frex_strobe_width[11:4]

4.12 one-time programmable (OTP) memory

The OV5647 supports a maximum of 256 bits of one-time programmable (OTP) memory to store chip identification and manufacturing information. It can be controlled through the SCCB (see **table 4-11**).

table 4-11 OTP control function registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3D00	OTP_DATA_0	0x00	RW	OTP Buffer 0
0x3D01	OTP_DATA_1	0x00	RW	OTP Buffer 1
0x3D02	OTP_DATA_2	0x00	RW	OTP Buffer 2
0x3D03	OTP_DATA_3	0x00	RW	OTP Buffer 3
0x3D04	OTP_DATA_4	0x00	RW	OTP Buffer 4
0x3D05	OTP_DATA_5	0x00	RW	OTP Buffer 5
0x3D06	OTP_DATA_6	0x00	RW	OTP Buffer 6
0x3D07	OTP_DATA_7	0x00	RW	OTP Buffer 7
0x3D08	OTP_DATA_8	0x00	RW	OTP Buffer 8
0x3D09	OTP_DATA_9	0x00	RW	OTP Buffer 9
0x3D0A	OTP_DATA_A	0x00	RW	OTP Buffer A
0x3D0B	OTP_DATA_B	0x00	RW	OTP Buffer B
0x3D0C	OTP_DATA_C	0x00	RW	OTP Buffer C
0x3D0D	OTP_DATA_D	0x00	RW	OTP Buffer D
0x3D0E	OTP_DATA_E	0x00	RW	OTP Buffer E
0x3D0F	OTP_DATA_F	0x00	RW	OTP Buffer F
0x3D10	OTP_DATA_16	0x00	RW	OTP Buffer 10
0x3D11	OTP_DATA_17	0x00	RW	OTP Buffer 11
0x3D12	OTP_DATA_18	0x00	RW	OTP Buffer 12
0x3D13	OTP_DATA_19	0x00	RW	OTP Buffer 13
0x3D14	OTP_DATA_20	0x00	RW	OTP Buffer 14
0x3D15	OTP_DATA_21	0x00	RW	OTP Buffer 15
0x3D16	OTP_DATA_22	0x00	RW	OTP Buffer 16
0x3D17	OTP_DATA_23	0x00	RW	OTP Buffer 17

table 4-11 OTP control function registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3D18	OTP_DATA_24	0x00	RW	OTP Buffer 18
0x3D19	OTP_DATA_25	0x00	RW	OTP Buffer 19
0x3D1A	OTP_DATA_26	0x00	RW	OTP Buffer 1A
0x3D1B	OTP_DATA_27	0x00	RW	OTP Buffer 1B
0x3D1C	OTP_DATA_28	0x00	RW	OTP Buffer 1C
0x3D1D	OTP_DATA_29	0x00	RW	OTP Buffer 1D
0x3D1E	OTP_DATA_30	0x00	RW	OTP Buffer 1E
0x3D1F	OTP_DATA_31	0x00	RW	OTP Buffer 1F
0x3D20	OTP_PROGRAM_CTRL	0x00	RW	Bit[7]: OTP_wr_busy Bit[1]: OTP_program_speed 0: Fast 1: Slow Bit[0]: OTP_program_enable Changing from 0 to 1 initiates OTP programming
0x3D21	OTP_LOAD_CTRL	0x00	RW	Bit[7]: OTP_rd_busy Bit[1]: OTPspeed 0: Fast 1: Slow Bit[0]: OTP_load_enable Changing from 0 to 1 initiates OTP read

5 image sensor processor digital functions

5.1 ISP general controls

table 5-1 ISP general control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0xFF	RW	Bit[7]: lenc_en 0: Disable 1: Enable Bit[2]: bc_en 0: Disable 1: Enable Bit[1]: wc_en 0: Disable 1: Enable
0x5001	ISP CTRL01	0x01	RW	Bit[0]: awb_en 0: Disable 1: Enable
0x5002	ISP CTRL02	0x41	RW	Bit[6]: win_en 0: Disable 1: Enable Bit[1]: otp_en 0: Disable 1: Enable Bit[0]: awb_gain_en 0: Disable 1: Enable
0x5003	ISP CTRL03	0x0A	RW	Bit[3]: buf_en 0: Disable 1: Enable Bit[2]: bin_man_set 0: Manual value as 0 1: Manual value as 1 Bit[1]: bin_auto_en 0: Disable 1: Enable
0x5005	ISP CTRL05	0x14	RW	Bit[4]: awb_bias_on 0: Disable AWB bias 1: Enable AWB bias Bit[2]: lenc_bias_on 0: Disable LENC bias 1: Enable LENC bias

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

address	register name	default value	R/W	description
0x501F	ISP CTRL1F	0x03	RW	Bit[5]: enable_opt 0: Not latched by VSYNC 1: Enable latched by VSYNC Bit[4]: cal_sel 0: DPC cal_start using SOF 1: DPC cal_start using VSYNC Bit[2:0]: fmt_sel 010: ISP output data 011: ISP input data bypass
0x5025	ISP CTRL25	0x00	RW	Bit[1:0]: avg_sel 00: Inputs of AVG module are from LENC output 01: Inputs of AVG module are from AWB gain output 10: Inputs of AVG module are from DPC output 11: Inputs of AVG module are from binning output
0x503D	ISP CTRL3D	0x00	RW	Bit[7]: test_pattern_en 0: Disable 1: Enable Bit[6]: rolling_bar 0: Disable rolling bar 1: Enable rolling bar Bit[5]: transparent_mode 0: Disable 1: Enable Bit[4]: squ_bw_mode 0: Output square is color square 1: Output square is black-white square Bit[3:2]: bar_style When set to a different value, a different type of color bar is output Bit[1:0]: test_pattern_type 00: Color bar 01: Square 10: Random data 11: Input data

table 5-1 ISP general control registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x503E	ISP CTRL3E	0x00	RW	Bit[6]: win_cut_en Bit[5]: isp_test 0: Two lowest bits are 1 1: Two lowest bits are 0 Bit[4]: rnd_same 0: Frame-changing random data pattern 1: Frame-fixed random data pattern Bit[3:0]: rnd_seed Initial seed for random data pattern
0x5046	ISP CTRL46	0x09	RW	Bit[3]: awbg_en 0: Disable 1: Enable Bit[0]: isp_en 0: Disable 1: Enable
0x504B	ISP CTRL4B	0x30	RW	ISP Control (0: disable; 1: enable) Bit[5]: post_binning h_enable Bit[4]: post_binning v_enable Bit[3]: flip_man_en Bit[2]: flip_man Bit[1]: mirror_man_en Bit[0]: Mirror

5.2 lens correction (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 light distribution due to lens curvature. The LENC correcting curve automatic calculation according sensor gain is also added so that the LENC can adapt with the sensor gain. Also, the LENC supports the subsample function in both horizontal and vertical directions.

Registers 0x5888 ~ 0x588F need to change only when DSP input is not generated internally. In other words, the DSP input is from an external sensor.

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

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0x89	RW	Bit[7]: lenc_en 0: Disable 1: Enable
0x583E	MAX GAIN	0x40	RW	Bit[7:0]: max_gain
0x583F	MIN GAIN	0x20	RW	Bit[7:0]: min_gain
0x5840	MIN Q	0x18	RW	Bit[6:0]: min_q
0x5841	LENC CTRL59	0x0D	RW	Bit[3]: ADDBLC 0: Disable BLC add back function 1: Enable BLC add back function Bit[2]: blc_en 0: Disable BLC function 1: Enable BLC function Bit[1]: gain_man_en Bit[0]: autoq_en 0: Used constant Q (0x40) 1: Used calculated Q
0x5842	BR HSCALE	0x01	RW	Bit[3:0]: br_hscale[11:8] Reciprocal of horizontal step for BR channel. BR channel in whole image is divided into 5x5 blocks. The step is used to point to the border of the adjacent block
0x5843	BR HSCALE	0x2B	RW	Bit[7:0]: br_hscale[7:0] Reciprocal of horizontal step for BR channel. BR channel in whole image is divided into 5x5 blocks. The step is used to point to the border of the adjacent block

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

address	register name	default value	R/W	description
0x5844	BR VSCALE	0x01	RW	Bit[2:0]: br_vscale[10:8] Reciprocal of vertical step for BR channel. BR channel in whole image is divided into 5x5 blocks. The step is used to point to the border of the adjacent block
0x5845	BR VSCALE	0x8D	RW	Bit[7:0]: br_vscale[7:0] Reciprocal of vertical step for BR channel. BR channel in whole image is divided into 5x5 blocks. The step is used to point to the border of the adjacent block
0x5846	G HSCALE	0x01	RW	Bit[3:0]: g_hscale[11:8] Reciprocal of horizontal step for G channel. G channel in whole image is divided into 6x6 blocks. The step is used to point to the border of the adjacent block
0x5847	G HSCALE	0x8F	RW	Bit[7:0]: g_hscale[7:0] Reciprocal of horizontal step for G channel. G channel in whole image is divided into 6x6 blocks. The step is used to point to the border of the adjacent block
0x5848	G VSCALE	0x01	RW	Bit[2:0]: g_vscale[10:8] Reciprocal of vertical step for G channel. G channel in whole image is divided into 6x6 blocks. The step is used to point to the border of the adjacent block
0x5849	G VSCALE	0x09	RW	Bit[7:0]: g_vscale[7:0] Reciprocal of vertical step for G channel. G channel in whole image is divided into 6x6 blocks. The step is used to point to the border of the adjacent block

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. Also, some special functions are available for those pixels located at the image boundary. To remove the defect pixel effect correctly, the proper threshold should first be determined.

table 5-3 defect pixel cancellation registers

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0xFF	RW	Bit[2]: bc_en 0: Disable 1: Enable Bit[1]: wc_en 0: Disable 1: Enable
0x5780~ 0x5791	DPC CTRL	–	RW	Debug Control Changing these registers is not recommended

5.4 auto white balance (AWB)

The main function of Auto White Balance (AWB) is the process of removing unrealistic color casts so that objects which appear white in person are rendered white in the image or video. Thus, the AWB makes sure that the white color is always a white color in different color temperatures. It supports manual white balance and auto white balance. For auto white balance, simple AWB is supplied. For auto white balance, the adjust option is also provided for the customer.

table 5-4 AWB control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5001	ISP CTRL01	0x01	RW	Bit[1]: awb_en 0: Disable 1: Enable

table 5-4 AWB control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x5180	AWB CTRL	0x00	RW	Bit[6]: fast_awb 0: Disable fast AWB calculation function 1: Enable fast AWB calculation function Bit[5]: freeze_gain_en When it is enabled, the output AWB gains will be input AWB gains Bit[4]: freeze_sum_en When it is set, the sums and averages value will be same as previous frame Bit[3]: gain_man_en 0: Output calculated gains 1: Output manual gains set by registers Bit[2]: start_sel 0: Select the last HREF falling edge of before gain input as calculated start signal 1: Select the last HREF falling edge of after gain input as calculated start signal
0x5181	AWB DELTA	0x20	RW	Bit[7]: delta_opt Bit[6]: base_man_en Bit[5:0]: awb_delta Delta value to increase or decrease the gains
0x5182	STABLE RANGE	0x04	RW	Bit[7:0]: stable_range
0x5183	STABLE RANGEW	0x08	RW	Bit[7:0]: stable_rangew Wide stable range
0x5184	HSIZE_MAN	0x01	RW	Bit[3:0]: hsize_man[11:8]
0x5185	HSIZE_MAN	0xE0	RW	Bit[7:0]: hsize_man[7:0]
0x5186	MANUAL RED GAIN MSB	0x04	RW	Bit[3:0]: red_gain_man[11:8]
0x5187	MANUAL RED GAIN LSB	0x00	RW	Bit[7:0]: red_gain_man[7:0]
0x5188	MANUAL GREEN GAIN MSB	0x04	RW	Bit[3:0]: grn_gain_man[11:8]
0x5189	MANUAL GREEN GAIN LSB	0x00	RW	Bit[7:0]: grn_gain_man[7:0]

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

address	register name	default value	R/W	description
0x518A	MANUAL BLUE GAIN MSB	0x04	RW	Bit[3:0]: blu_gain_man[11:8]
0x518B	MANUAL BLUE GAIN LSB	0x00	RW	Bit[7:0]: blu_gain_man[7:0]
0x518C	RED GAIN LIMIT	0xF0	RW	Bit[7:4]: red_gain_up_limit Bit[3:0]: red_gain_dn_limit They are only the highest 4 bits of limitation. Max red gain is {red_gan_up_limit,FF} Min red gain is {red_gain_dn_limit,00}
0x518D	GREEN GAIN LIMIT	0xF0	RW	Bit[7:4]: green_gain_up_limit Bit[3:0]: green_gain_dn_limit They are only the highest 4 bits of limitation. Max green gain is {green_gan_up_limit,FF} Min green gain is {green_gain_dn_limit,00}
0x518E	BLUE GAIN LIMIT	0xF0	RW	Bit[7:4]: blue_gain_up_limit Bit[3:0]: blue_gain_dn_limit They are only the highest 4 bits of limitation. Max blue gain is {blue_gan_up_limit,FF} Min blue gain is {blue_gain_dn_limit,00}

5.5 post binning function

CFA image subsample will suffer zig_zag issues around slant edges and color shift for it is an non-uniform method in physical coordinate. Post binning will map these pixels to their physically correct location.

table 5-5 post binning control registers

address	register name	default value	R/W	description
0x5003	ISP CTRL3	0x0A	RW	Bit[2]: bin_en
0x504B	ISP CTRL75	0x30	RW	Bit[5]: h_en Bit[4]: v_en

6 image sensor output interface digital functions

6.1 system control

System control registers include clock, reset control, and PLL configure.

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

address	register name	default value	R/W	description
0x3000	SC_CMMN_PAD_OEN0	0x00	RW	io_y_oen[11:8]
0x3001	SC_CMMN_PAD_OEN1	0x00	RW	io_y_oen[7:0]
0x3002	SC_CMMN_PAD_OEN2	0x00	RW	Bit[7]: io_vsync_oen Bit[6]: io_href_oen Bit[5]: io_pclk_oen Bit[4]: io_frex_oen Bit[3]: io_strobe_oen Bit[2]: io_sda_oen Bit[1]: io_gpio1_oen Bit[0]: io_gpio0_oen
0x3006	SC_CMMN_PLL_CTR13	0x00	RW	Bit[5:2]: SDIV Clock divider for 50/60 Hz detection block
0x3008	SC_CMMN_PAD_OUT0	0x00	RW	Bit[3:0]: io_y_o[11:8]
0x3009	SC_CMMN_PAD_OUT1	0x00	RW	Bit[7:0]: io_y_o[7:0]
0x300A	SC_CMMN_CHIP_ID	0x56	R	Chip ID High
0x300B	SC_CMMN_CHIP_ID	0x47	R	Chip ID Low
0x300C	SC_CMMN_SCCB_ID	0x6C	RW	SCCB ID
0x300D	SC_CMMN_PAD_OUT2	0x00	RW	Bit[7]: io_vsync_o Bit[6]: io_href_o Bit[5]: io_pclk_o Bit[4]: io_frex_o Bit[3]: io_strobe_o Bit[2]: io_sda_o Bit[1]: io_gpio1_o Bit[0]: io_gpio0_o
0x300E	SC_CMMN_PAD_SEL0	0x00	RW	Bit[3:0]: io_y_sel[11:8]

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

address	register name	default value	R/W	description
0x300F	SC_CMMN_PAD_SEL1	0x00	RW	Bit[7:0]: io_y_sel[7:0]
0x3010	SC_CMMN_PAD_SEL2	0x00	RW	Bit[7]: io_vsync_sel Bit[6]: io_href_sel Bit[5]: io_pclk_sel Bit[4]: io_frex_sel Bit[3]: io_strobe_sel Bit[2]: io_sda_sel Bit[1]: io_gpio1_sel Bit[0]: io_gpio0_sel
0x3011	SC_CMMN_PAD_PK	0x02	RW	Bit[7]: pd_dato_en Bit[6:5]: iP2X3v[3:2] Bit[1]: frex_enb 0: Enable 1: Disable
0x3013	SC_CMMN_A_PWC_PK_O	0x00	RW	Bit[7:4]: Debug control Changing these registers is not recommended Bit[3]: bp_regulator 0: Enable internal regulator 1: Disable internal regulator Bit[2:0]: Debug control Changing these registers is not recommended
0x3014	SC_CMMN_A_PWC_PK_O	0x0B	RW	Bit[6:4]: apd[2:0] Bit[3:0]: DIO
0x3016	SC_CMMN_MIPI_PHY	0x00	RW	Bit[7:6]: LPH Bit[3]: mipi_pad_enable Bit[2]: pgm_bp_hs_en_lat btpass the latch of hs_enable Bit[1:0]: ictl[1:0] Bias current adjustment
0x3017	SC_CMMN_MIPI_PHY	0x10	RW	Bit[7:6]: pgm_vcm[1:0] High speed common mode voltage Bit[5:4]: pgm_lptx[1:0] 01: Driving strength of low speed transmitter Bit[3]: IHALF Bias current reduction Bit[2]: pgm_vicd CD input low voltage Bit[1]: pgm_vih CD input high voltage-dummy Bit[0]: pgm_hs_valid Valid delay-dummy

table 6-1 system control registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x3018	SC_CMMN_MIPI_ SC_CTRL	0x58	RW	Bit[7:5]: mipi_lane_mode 0: One lane mode 1: Two lane mode Bit[4]: r_phy_pd_mipi 1: Power donw PHY HS TX Bit[3]: r_phy_pd_lprx 1: Power down PHY LP RX module Bit[2]: mipi_en 0: DVP enable 1: MIPI enable Bit[1]: mipi_susp_reg MIPI system Suspend register 1: suspend Bit[0]: lane_dis_op 0: Use mipi_release1/2 and lane_disable1/2 to disable two data lane 1: Use lane_disable1/2 to disable two data lane
0x3019	SC_CMMN_MIPI_ SC_CTRL	0x10	RW	Bit[7:0]: MIPI ULPS resume mark1 detect length
0x3021	SC_CMMN_MISC_ CTRL	0x23	RW	Bit[5]: fst_stby_ctr 1: Software standby enter at l_blk 0: Software standby enter at v_blk Bit[4]: mipi_ctr_en 1: Enable MIPI remote reset and suspend control SC 0: Disable the function Bit[3]: mipi_rst_sel 0: MIPI remote reset all registers 1: MIPI remote reset all digital modules Bit[2]: gpio_pclk_en Bit[1]: frex_ef_sel Bit[0]: cen_global_o
0x3022	SC_CMMN_MIPI_ SC_CTRL	0x00	RW	Bit[3]: lptx_ck_opt Bit[2]: pull_down_clk_lane Bit[1]: pull_down_data_lane2 Bit[0]: pull_down_data_lane1
0x302A	SC_CMMN_SUB_ID	–	R	Bit[7:4]: Process Bit[3:0]: Version

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

address	register name	default value	R/W	description
0x3034	SC_CMMN_PLL_ CTRL0	0x1A	RW	Bit[6:4]: pll_charge_pump Bit[3:0]: mipi_bit_mode 0000: 8 bit mode 0001: 10 bit mode Others: Reserved to future use
0x3035	SC_CMMN_PLL_ CTRL1	0x11	RW	Bit[7:4]: system_clk_div Will slow down all clocks Bit[3:0]: scale_divider_mipi MIPI PCLK/SERCLK can be slowed down when image is scaled down
0x3036	SC_CMMN_PLL_ MULTIPLIER	0x69	RW	Bit[7:0]: PLL_multiplier (4~252) can be any integer during 4~127 and only even integer during 128~252
0x3037	SC_CMMN_PLL_ CTR13	0x03	RW	Bit[4]: pll_root_div 0: Bypass 1: /2 Bit[3:0]: pll_prediv 1, 2, 3, 4, 6, 8
0x3039	SC_CMMN_PLL_ CTRL_R	0x00	RW	Bit[7]: pll_bypass
0x303A	SC_CMMN_PLLS_ CTRL0	0x00	RW	Bit[7]: plls_bypass
0x303B	SC_CMMN_PLLS_ CTRL1	0x19	RW	Bit[4:0]: plls_multiplier
0x303C	SC_CMMN_PLLS_ CTRL2	0x11	RW	Bit[6:4]: plls_cp Bit[3:0]: plls_sys_div
0x303D	SC_CMMN_PLLS_ CTRL3	0x30	RW	Bit[5:4]: plls_pre_div 00: /1 01: /1.5 10: /2 11: /3 Bit[2]: plls_div_r 0: /1 1: /2 Bit[1:0]: plls_seld5 00: /1 01: /1 10: /2 11: /2.5

6.2 SCCB

table 6-2 system control registers

address	register name	default value	R/W	description
0x3100	SCCB CTRL	0x00	RW	Bit[3]: r_sda_dly_en Bit[2:0]: r_sda_dly
0x3101	SCCB OPT	0x12	RW	Bit[4]: en_ss_addr_inc Bit[3]: r_sda_byp_sync 0: Two clock stage SYNC for sda_i 1: No sync for sda_i Bit[2]: r_scl_byp_sync 0: Two clock stage SYNC for scl_i 1: No sync for scl_i Bit[1]: r_msk_glitch Bit[0]: r_msk_stop
0x3102	SCCB FILTER	0x00	RW	Bit[7:4]: r_sda_num Bit[3:0]: r_scl_num
0x3103	SCCB SYSREG	0x00	RW	Bit[6]: ctrl_rst_mipisc Bit[5]: ctrl_rst_srb Bit[4]: ctrl_rst_sccb_s Bit[3]: ctrl_rst_pon_sccb_s Bit[2]: ctrl_rst_clkmod Bit[1]: ctrl_rst_mipi_phy_rst_o Bit[0]: ctrl_pll_rst_o
0x3104	PWUP DIS	0x01	RW	Bit[4]: r_srb_clk_syn_en Bit[3]: pwup_dis2 Bit[2]: pwup_dis1 Bit[1]: pll_clk_sel Bit[0]: pwup_dis0
0x3105	PADCLK DIV	0x11	RW	Bit[5]: SCLK use p_clk_i Bit[4]: Sleep enable Bit[3:0]: PADCLK divider for SCCB
0x3106	SRB CTRL	0xF9	RW	Bit[3:2]: PLL clock divider 00: pll_sclk 01: pll_sclk/2 10: pll_sclk/4 11: pll_sclk Bit[1]: rst_arb 1: Reset arbiter Bit[0]: sclk_arb 1: Enable SCLK to arbiter

6.3 group register write

The OV5647 supports group register write with up to four groups. Each group could have up to 16 registers.

Example settings:

6C 0x3208 0x00; Group 0 begin

6C 0x3503 0x03; register 1

6C 0x3501 0x7A; register 2

6C 0x3502 0xA0; register 3

6C 0x3208 0x10; Group 0 end

6C 0x3208 0xA0; write register group 0

table 6-3 group hold control registers

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	0x04	RW	Group1 Start Address in SRAM, actual address is {0x3201[3:0], 4'h0}
0x3202	GROUP ADR2	0x08	RW	Group2 Start Address in SRAM, actual address is {0x3202[3:0], 4'h0}
0x3203	GROUP ADR3	0x0B	RW	Group3 sStart 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 LEN2	–	R	Length of Group2
0x3207	GROUP LEN3	–	R	Length of Group3
0x3208	GROUP ACCESS	–	W	Bit[7:4]: Group_ctrl 0000: Enter group write mode 0001: Exit group write mode 1010: Initiate group write Bit[3:0]: Group ID 0000: Group 0 0001: Group 1 0010: Group 2 0011: Group 3

6.4 timing control

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

address	register name	default value	R/W	description
0x3800	TIMING_X_ADDR_START	0x00	RW	Bit[3:0]: x_addr_start[11:8]
0x3801	TIMING_X_ADDR_START	0x0C	RW	Bit[7:0]: x_addr_start[7:0]
0x3802	TIMING_Y_ADDR_START	0x00	RW	Bit[3:0]: y_addr_start[11:8]
0x3803	TIMING_Y_ADDR_START	0x04	RW	Bit[7:0]: y_addr_start[7:0]
0x3804	TIMING_X_ADDR_END	0x0A	RW	Bit[3:0]: x_addr_end[11:8]
0x3805	TIMING_X_ADDR_END	0x33	RW	Bit[7:0]: x_addr_end[7:0]
0x3806	TIMING_Y_ADDR_END	0x07	RW	Bit[3:0]: y_addr_end[11:8]
0x3807	TIMING_Y_ADDR_END	0xA3	RW	Bit[7:0]: y_addr_end[7:0]
0x3808	TIMING_X_OUTPUT_SIZE	0x0A	RW	Bit[3:0]: DVP output horizontal width[11:8]
0x3809	TIMING_X_OUTPUT_SIZE	0x20	RW	Bit[7:0]: DVP output horizontal width[7:0]
0x380A	TIMING_Y_OUTPUT_SIZE	0x07	RW	Bit[3:0]: DVP output vertical height[11:8]
0x380B	TIMING_Y_OUTPUT_SIZE	0x98	RW	Bit[7:0]: DVP output vertical height[7:0]
0x380C	TIMING_HTS	0x0A	RW	Bit[4:0]: Total horizontal size[12:8]
0x380D	TIMING_HTS	0x8C	RW	Bit[7:0]: Total horizontal size[7:0]
0x380E	TIMING_VTS	0x07	RW	Bit[1:0]: Total vertical size[9:8]
0x380F	TIMING_VTS	0xB0	RW	Bit[7:0]: Total vertical size[7:0]
0x3810	TIMING_ISP_X_WIN	0x00	RW	Bit[3:0]: ISP horizontal offset[11:8]
0x3811	TIMING_ISP_X_WIN	0x04	RW	Bit[7:0]: ISP horizontal offset[7:0]
0x3812	TIMING_ISP_Y_WIN	0x00	RW	Bit[3:0]: ISP vertical offset[11:8]
0x3813	TIMING_ISP_Y_WIN	0x02	RW	Bit[7:0]: ISP vertical offset[7:0]
0x3814	TIMING_X_INC	0x11	RW	Bit[7:4]: h_odd_inc Horizontal subsample odd increase number Bit[3:0]: h_even_inc Horizontal subsample even increase number

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

address	register name	default value	R/W	description
0x3815	TIMING_Y_INC	0x11	RW	Bit[7:4]: v_odd_inc Vertical subsample odd increase number Bit[3:0]: v_even_inc Vertical subsample even increase number
0x3816	TIMING_HSYNCST	0x00	RW	Bit[3:0]: HSYNC start point[11:8]
0x3817	TIMING_HSYNCST	0x00	RW	Bit[7:0]: HSYNC start point[7:0]
0x3818	TIMING_HSYNCW	0x00	RW	Bit[3:0]: HSYNC window[11:8]
0x3819	TIMING_HSYNCW	0x00	RW	Bit[7:0]: HSYNC window[7:0]
0x3820	TIMING_TC_REG20	0x40	RW	Bit[2]: r_vflip_isp Bit[1]: r_vflip_snr Bit[0]: r_vbin
0x3821	TIMING_TC_REG21	0x00	RW	Bit[2]: r_mirror_isp Bit[1]: r_mirror_snr Bit[0]: r_hbin
0x3822	TIMING_TC_REG22	0x10	RW	Bit[4:0]: r_ablc

6.5 strobe

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

address	register name	default value	R/W	description
0x3B00	STROBE_RSTRB	0x00	RW	Bit[7]: Strobe ON Bit[6]: Reverse Bit[3:2]: width_in_xenon Bit[1:0]: Mode select 00: Xenon 01: LED1 10: LED2 11: LED3
0x3B01	STROBE_FREX_EXP_H2	0x00	RW	Bit[7:0]: frex_exp[23:16]
0x3B02	STROBE_SHUTTER_DLY	0x08	RW	Bit[4:0]: shutter_dly[12:8]
0x3B03	STROBE_SHUTTER_DLY	0x00	RW	Bit[7:0]: shutter_dly[7:0]
0x3B04	STROBE_FREX_EXP_H	0x04	RW	Bit[7:0]: frex_exp[15:8]
0x3B05	STROBE_ FREX_EXP_L	0x00	RW	Bit[7:0]: frex_exp[7:0]

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

address	register name	default value	R/W	description
0x3B06	STROBE_FREX_CTRL0	0x04	RW	Bit[7:6]: frex_pchg_width Bit[5:4]: frex_strobe_option Bit[3:0]: frex_strobe_width[3:0]
0x3B07	STROBE_ FREX_MODE_SEL	0x08	RW	Bit[4]: frex_sa1 Bit[3]: fx1_fm_en Bit[2]: frex_inv Bit[1:0]: Frex mode select 00: frex_strobe mode 0 01: frex_strobe mode 1 1x: Rolling strobe
0x3B08	STROBE_FREX_EXP_REQ	0x00	RW	Bit[0]: frex_exp_req
0x3B09	FREX_SHUTTER_DELAY	0x00	RW	Bit[2:0]: FREX end option
0x3B0A	STROBE_FREX_RST_LENGTH	0x04	RW	Bit[2:0]: frex_rst_length[2:0]
0x3B0B	STROBE_WIDTH	0x00	RW	Bit[7:0]: frex_strobe_width [19:12]
0x3B0C	STROBE_WIDTH	0x3D	RW	Bit[7:0]: frex_strobe_width[11:4]

6.6 frame control (FC)

Frame control (FC) is used to mask some specified frame by setting the appropriate registers.

table 6-6 frame control registers

address	register name	default value	R/W	description
0x4200	FRAME CONTROL00	0x00	RW	Bit[2]: fcnt_eof_sel Bit[1]: fcnt_mask_dis Bit[0]: Frame counter reset
0x4201	FRAME CONTROL01	0x00	RW	Control Passed Frame Number Bit[3:0]: Frame ON number When both ON and OFF numbers are set to 0x00, frame control is in bypass mode
0x4202	FRAME CONTROL02	0x00	RW	Control Masked Frame Number Bit[3:0]: Frame OFF number When both ON and OFF numbers are set to 0x00, frame control is in bypass mode
0x4203	FRAME CONTROL03	0x00	RW	Bit[6]: rblue_mask_dis Bit[5]: data_mask_dis Bit[4]: valid_mask_dis Bit[3]: href_mask_dis Bit[2]: eof_mask_dis Bit[1]: sof_mask_dis Bit[0]: all_mask_dis

6.7 digital video port (DVP)

The Digital Video Port (DVP) provides 10-bit parallel data output in all formats supported and extended features including compression mode, HSYNC mode, CCIR656 mode, and test pattern output.

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

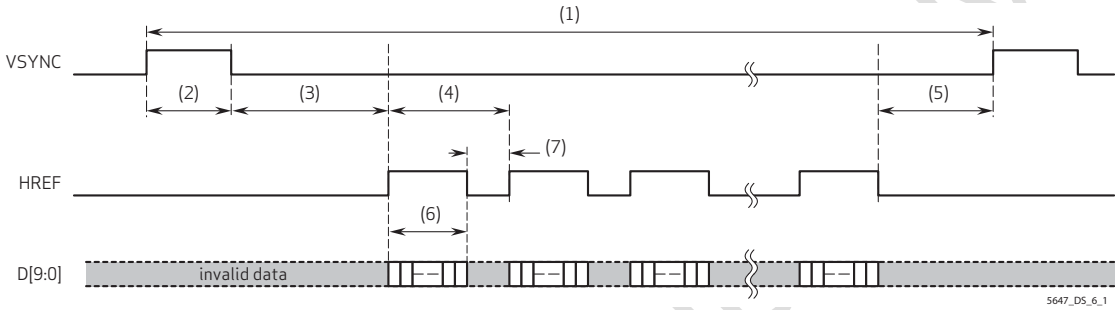
address	register name	default value	R/W	description
0x4700	DVP MODE SELECT	0x04	RW	Bit[3]: CCIR v select Bit[2]: CCIR f select Bit[1]: CCIR656 mode enable Bit[0]: HSYNC mode enable
0x4701	DVP VSYNC WIDTH CONTRL	0x01	RW	VSYNC Width (in terms of number of lines)
0x4702	DVP_HSYVSY_NEG_WIDTH	0x01	RW	Bit[7:0]: VSYNC length in terms of pixel count[15:8]

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

address	register name	default value	R/W	description
0x4703	DVP_HSYVSY_NEG_WIDTH	0x00	RW	Bit[7:0]: VSYNC length in terms of pixel count[7:0]
0x4704	DVP VSYNC MODE	0x00	RW	Bit[3:2]: r_vsyncount_sel Bit[1]: r_vsync3_mod Bit[0]: r_vsync2_mod
0x4705	DVP_EOF_VSYNC DELAY	0x00	RW	Bit[7:0]: eof_vsync_delay[23:16] SOF/EOF negative edge to VSYNC positive edge delay
0x4706	DVP_EOF_VSYNC DELAY	0x00	RW	Bit[7:0]: eof_vsync_delay[15:8] SOF/EOF negative edge to VSYNC positive edge dealy
0x4707	DVP_EOF_VSYNC DELAY	0x00	RW	Bit[7:0]: eof_vsync_delay[7:0] SOF/EOF negative edge to VSYNC positive edge delay
0x4708	DVP_POL_CTRL	0x01	RW	Bit[7]: Clock DDR mode enable Bit[5]: VSYNC gated clock enable Bit[4]: HREF gated clock enable Bit[3]: No first for FIFO Bit[2]: HREF polarity reverse Bit[1]: VSYNC polarity reverse Bit[0]: PCLK polarity reverse
0x4709	BIT_TEST_PATTERN	0x00	RW	Bit[7]: FIFO bypass mode Bit[6:4]: Data bit swap Bit[3]: Bit test mode Bit[2]: 10-bit bit test Bit[1]: 8-bit bit test Bit[0]: Bit test enable
0x470A	DVP_BYP_CTRL	0x00	RW	Bit[7:0]: bypass_ctrl[15:8]
0x470B	DVP_BYP_CTRL	0x00	RW	Bit[7:0]: bypass_ctrl[7:0]
0x470C	DVP_BYP_SEL	0x00	RW	Bit[4]: HREF select Bit[3:0]: Bypass select

6.7.1 DVP timing

figure 6-1 DVP timing diagram



note
The timing values shown in **table 6-8** may vary depending upon register settings.

table 6-8 DVP timing specifications (sheet 1 of 2)

mode	timing
5 Megapixel 2592x1944	(1) 5313600 tp(2700x1968) (2) 2956 tp (3) 29624 tp (4) 2700 tp (5) 32328 tp (6) 2592 tp (7) 108 tp
1080p 1920x1080	(1) 2260992 tp (2048x1104) (2) 2304 tp (3) 22472 tp (4) 2048 tp (5) 24504 tp (6) 1920 tp (7) 128 tp
960p 1280x960	(1) 2015232 tp (2048x984) (2) 2304 tp (3) 12872 tp (4) 2048 tp (5) 34744 tp (6) 1280 tp (7) 768 tp
720p 1280x720	(1) 1523712 tp (2048x744) (2) 2304 tp (3) 21064 tp (4) 2048 tp (5) 26552 tp (6) 1280 tp (7) 768 tp

table 6-8 DVP timing specifications (sheet 2 of 2)

mode	timing
VGA 640x480	(1) 1032192 tp (2048x504) (2) 2304 tp (3) 13512 tp (4) 2048 tp (5) 34744 tp (6) 640 tp (7) 1408 tp
QVGA 320x240	(1) 540672 tp (2048x264) (2) 2304 tp (3) 13832 tp (4) 2048 tp (5) 34744 tp (6) 320 tp (7) 1728 tp

6.8 mobile industry processor interface (MIPI)

MIPI provides a single uni-directional clock lane and two bi-directional data lane solution for communication links between components inside a mobile device. The two data lanes have full support for HS (uni-directional) and LP (bi-directional) data transfer mode.

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

address	register name	default value	R/W	description
0x4800	MIPI CTRL 00	0x04	RW	MIPI Control 00 Bit[7]: mipi_hs_only 0: MIPI can support CD and ESCAPE mode 1: MIPI always in High Speed mode Bit[6]: ck_mark1_en 1: Enable clock lane mark1 when resume 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[3]: Lane select 0: Use lane1 as default data lane 1: Use lane2 as default data lane Bit[2]: Idle status 0: MIPI bus will be LP00 when no packet to transmit 1: MIPI bus will be LP11 when no packet to transmit Bit[1]: Clock lane first bits 0: Output 0x55 1: Output 0xAA Bit[0]: Clock lane disable 1: Manually set clock lane to low power mode

table 6-9 MIPI transmitter registers (sheet 2 of 8)

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]: Short packet WORD COUNTER manual enable 0: Use frame counter or line counter 1: Select spkt_wc_reg_o (see {0x4812, 0x4813}) 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]: mark1_en1 1: After each rst release, lane 1 should send mark1 for wkup_dly_o when mipi_sys_susp =1 Bit[0]: mark1_en2 1: After each reset release, lane 2 should send mark1 for wkup_dly_o when mipi_sys_susp=1

table 6-9 MIPI transmitter registers (sheet 3 of 8)

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_o[7:0]
0x4803	MIPI CTRL 03	0x50	RW	MIPI Control 03 Bit[7:6]: lp_glitch_nu 0: Use 2d of lp_in 1: Mask one sclk cycle glitch of lp_in Bit[5:4]: cd_glitch_nu 0: Use 2d of lp_cd_in 1: Mask one SCLK cycle glitch of lp_cd_in Bit[3]: Enable CD plus of data lane1 0: Disable 1: Enable Bit[2]: Enable CD plus of data lane2 0: Disable 1: Enable Bit[1]: Enable CD of data_lane1 from PHY 0: Disable 1: Enable Bit[0]: Enable CD of data_lane2 from PHY 0: Disable 1: Enable

table 6-9 MIPI transmitter registers (sheet 4 of 8)

address	register name	default value	R/W	description
0x4804	MIPI CTRL 04	0x8D	RW	MIPI Control 04 Bit[7]: wait_pkt_end 1: Wait HS packet end when send UL command Bit[6]: tx_lsb_first 0: lp_tx and lp_rx high bit first 1: Low power transmit low bit first Bit[5]: dir_recover_sel 0: Auto change to output only when TurnAround command 1: Auto change to output when LP11 and GPIO is output Bit[4]: mipi_reg_en 0: Disable MIPI_REG_P to access registers, LP data will write to VFIFO 1: Enable MIPI_REG_P to access registers Bit[3]: Address read/write register will auto add 1 0: Disable 1: Enable Bit[2]: LP TX lane select 0: Select lane1 to transmit LP data 1: Select lane2 to transmit LP data Bit[1]: wr_first_byte 1: lp_rx will write first byte (command byte) to RAM Bit[0]: rd_ta_en 1: Send TurnAround command after sending register read data

table 6-9 MIPI transmitter registers (sheet 5 of 8)

address	register name	default value	R/W	description
0x4805	MIPI CTRL 05	0x10	RW	MIPI Control 05 Bit[7]: MIPI lane1 disable 1: Disable MIPI data lane1, lane1 will be LP00 Bit[6]: MIPI lane2 disable 1: Disable MIPI data lane2, lane2 will be LP00 Bit[5]: lpx_p_sel 0: Automatically calculate t_lpx_o in pclkex domain, unit pclk2x 1: Use lp_p_min[7:0] Bit[4]: lp_rx_intr_sel 0: Send lp_rx_intr_o at the first byte 1: Send lp_rx_intr_o at the end of receiving Bit[3]: cd_tst_sel 1: Select PHY test pins Bit[2]: mipi_reg_mask 1: Disable MIPI access SRB Bit[1]: clip enable Bit[0]: hd_sk_en 0: Disable MIPI and MCU handshake registers 1: Disable MIPI and MCU handshake registers
0x4806	MIPI REG RW CTRL	0x28	RW	Bit[7]: prbs_en Test mode Bit[6]: mipi_test Bit[5]: mipi_lp_op 0: Use new option to reduce mipi_lptx_p Bit[4]: two_lane_man_en 1: Use two_lane_man to manually control two_lane_mode Bit[3]: two_lane_man Bit[2]: rst_rtn_en 1: Change to input to allow host RW register after reset Bit[1]: frame_end_en 1: After frame end packet, change to input to allow host RW register Bit[0]: line_end_en 1: After line end packet, change to input to allow host RW register
0x480A	MIPI BIT ORDER	0x00	RW	Bit[2]: Bit order reverse Bit[1:0]: Bit position adjustment 01: {D[7:0],D[9:8]} 10: {D[1:0],D[9:2]}

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

address	register name	default value	R/W	description
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
0x4814	MIPI CTRL14	0x2A	RW	MIPI Control 14 Bit[7:6]: Virtual channel of MIPI Bit[5:0]: Data type in manual mode
0x4815	MIPI_DT_SPKT	0x00	RW	Bit[6]: pclk_div 0: Use rising edge of mipi_pclk_o to generate MIPI bus to PHY 1: Use falling edge of mipi_pclk_o to generate MIPI bus to PHY Bit[5:0]: Manual data type for short packet
0x4818	HS_ZERO_MIN	0x00	RW	High byte of the minimum value for hs_zero Unit ns
0x4819	HS_ZERO_MIN	0x96	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, 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	0x86	RW	Low byte of the minimum value for clk_zero, clk_zero_real = clk_zero_min_o + Tui*ui_clk_zero_min_o
0x481E	CLK_PREPARE_MIN	0x00	RW	High byte of the minimum value for clk_prepare, unit ns Bit[1:0]: clk_prepare_min[9:8]
0x481F	CLK_PREPARE_MIN	0x3C	RW	Low byte of the 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	0x56	RW	Low byte of the minimum value for clk_post clk_post_real = clk_post_min_o + Tui*ui_clk_post_min_o
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

table 6-9 MIPI transmitter registers (sheet 7 of 8)

address	register name	default value	R/W	description
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
0x4826	HS_PREPARE_MIN	0x00	RW	High byte of the minimum value for hs_prepare, unit ns Bit[1:0]: hs_prepare_min[9:8]
0x4827	HS_PREPARE_MIN	0x32	RW	Low byte of the minimum value for hs_prepare hs_prepare_real = hs_prepare_min_o + Tui*ui_hs_prepare_min_o
0x4827	HS_PREPARE_MIN	0x32	RW	Low byte of the minimum 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
0x482A	UI_HS_ZERO_MIN	0x05	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_MIN	0x00	RW	Minimum UI 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	0x04	RW	Minimum UI Value of hs_prepare, unit UI
0x4832	UI_HS_EXIT_MIN	0x00	RW	Minimum UI Value of hs_exit, unit UI
0x4833	MIPI_REG_MIN	0x00	RW	MIPI register address, lower bound (high byte) Address range of MIPI RW registers is from mipi_reg_min to mipi_reg_max
0x4834	MIPI_REG_MIN	0x00	RW	MIPI register address, lower bound (low byte)
0x4835	MIPI_REG_MAX	0xFF	RW	MIPI register address, upper bound (high byte)
0x4836	MIPI_REG_MAX	0xFF	RW	MIPI register address, upper bound (low byte)
0x4837	PCLK_PERIOD	0x15	RW	Period of pclk2x, pclk_div = 1, and 1-bit decimal

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

address	register name	default value	R/W	description
0x4838	WKUP_DLY	0x02	RW	Wakeup delay for MIPI
0x483A	DIR_DLY	0v08	RW	Change LP direction delay/2 after LP11
0x483C	MIPI CTRL 33	0x4F	RW	Bit[7:4]: t_lpx, unit: sclk cycles Bit[3:0]: t_clk_pre, unit: sclk cycles
0x483D	MIPI_T_TA_GO	0x10	RW	t_ta_go Unit: SCLK cycles
0x483E	MIPI_T_TA_SURE	0x06	RW	t_ta_sure Unit: SCLK cycles
0x483F	MIPI_T_TA_GET	0x14	RW	t_ta_get Unit: SCLK cycles
0x4843	SNR_PCLK_DIV	0x00	RW	Bit[0]: PCLK divider 0: PCLK/SCLK = 2 and pclk_div = 1 1: PCLK/SCLK = 1 and pclk_div = 1
0x4860	MIPI CTRL 60	–	R	MIPI Read/Write Only Bit[0]: mipi_dis_me 0: Enable MIPI read/write registers 1: Disable MIPI read/write registers
0x4861	HD_SK_REG0	–	R	MIPI Read/Write, SCCB and MCU Read Only
0x4862	HD_SK_REG1	–	R	MIPI Read/Write, SCCB and MCU Read Only
0x4863	HD_SK_REG2	–	R	MIPI Read/Write, SCCB and MCU Read Only
0x4864	HD_SK_REG3	–	R	MIPI Read/Write, SCCB and MCU Read Only
0x4865	MIPI_ST	–	R	Bit[5]: lp_rx_sel_i 1: MIPI_LP_RX receives LP data Bit[4]: tx_busy_i 1: MIPI_TX_LP_TX is busy to send LP data Bit[3]: mipi_lp_p1_i MIPI low power input for lane 1p Bit[2]: mipi_lp_n1_i MIPI low power input for lane 1n Bit[1]: mipi_lp_p2_i MIPI low power input for lane 2p Bit[0]: mipi_lp_n2_i MIPI low power input for lane 2n
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

7 register tables

The following tables provide descriptions of the device control registers contained in the OV5647. For all registers enable/disable bits, ENABLE = 1 and DISABLE = 0. The device slave addresses are 0x6C for write and 0x6D for read.

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

address	register name	default value	R/W	description
0x3000	SC_CMMN_PAD_ OEN0	0x00	RW	Bit[7:4]: io_y_oen[11:8] Bit[3:0]: Not used
0x3001	SC_CMMN_PAD_ OEN1	0x00	RW	Bit[7:0]: io_y_oen[7:0]
0x3002	SC_CMMN_PAD_ OEN2	0x00	RW	Bit[7]: io_vsync_oen Bit[6]: io_href_oen Bit[5]: io_pclk_oen Bit[4]: io_frex_oen Bit[3]: io_strobe_oen Bit[2]: io_sda_oen Bit[1]: io_gpio1_oen Bit[0]: io_gpio0_oen
0x3003~ 0x3005	DEBUG MODE	–	–	Debug Mode
0x3006	SC_CMMN_PLL_ CTR13	0x00	RW	Bit[7:6]: Debug control Changing these registers is not recommended Bit[5:2]: SDIV Clock divider for 50/60 Hz detection block Bit[1:0]: Debug control Changing these registers is not recommended
0x3007	DEBUG MODE	–	–	Debug Mode
0x3008	SC_CMMN_PAD_ OUT0	0x00	RW	Bit[7:4]: Not used Bit[3:0]: io_y_o[11:8]
0x3009	SC_CMMN_PAD_ OUT1	0x00	RW	Bit[7:0]: io_y_o[7:0]
0x300A	SC_CMMN_CHIP_ID	0x56	R	Chip ID high
0x300B	SC_CMMN_CHIP_ID	0x47	R	Chip ID low
0x300C	SC_CMMN_SCCB_ID	0x6C	RW	SCCB ID

table 7-1 system control registers (sheet 2 of 5)

address	register name	default value	R/W	description
0x300D	SC_CMMN_PAD_OUT2	0x00	RW	Bit[7]: io_vsync_o Bit[6]: io_href_o Bit[5]: io_pclk_o Bit[4]: io_frex_o Bit[3]: io_strobe_o Bit[2]: io_sda_o Bit[1]: io_gpio1_o Bit[0]: io_gpio0_o
0x300E	SC_CMMN_PAD_SEL0	0x00	RW	Bit[7:4]: Debug control Changing these registers is not recommended Bit[3:0]: io_y_sel[11:8]
0x300F	SC_CMMN_PAD_SEL1	0x00	RW	Bit[7:0]: io_y_sel[7:0]
0x3010	SC_CMMN_PAD_SEL2	0x00	RW	Bit[7]: io_vsync_sel Bit[6]: io_href_sel Bit[5]: io_pclk_sel Bit[4]: io_frex_sel Bit[3]: io_strobe_sel Bit[2]: io_sda_sel Bit[1]: io_gpio1_sel Bit[0]: io_gpio0_sel
0x3011	SC_CMMN_PAD_PK	0x02	RW	Bit[7]: pd_dato_en Bit[6:5]: iP2X3v[3:2] Bit[4:2]: Not used Bit[1]: frex_enb 0: Enable 1: Disable Bit[0]: Not used
0x3012	DEBUG MODE	–	–	Debug Mode
0x3013	SC_CMMN_A_PWC_PK_O	0x00	RW	Bit[7:4]: Debug control Changing these registers is not recommended Bit[3]: bp_regulator 0: Enable internal regulator 1: Disable internal regulator Bit[2:0]: Debug control Changing these registers is not recommended
0x3014	SC_CMMN_A_PWC_PK_O	0x0B	RW	Bit[7]: Not used Bit[6:4]: apd[2:0] Bit[3:0]: dio

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

address	register name	default value	R/W	description
0x3016	SC_CMMN_MIPI_PHY	0x00	RW	Bit[7:6]: lph Bit[5:4]: Not used Bit[3]: mipi_pad_enable Bit[2]: pgm_bp_hs_en_lat Bypass the latch of hs_enable Bit[1:0]: ictl[1:0] Bias current adjustment
0x3017	SC_CMMN_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 Bias current reduction Bit[2]: pgm_vicd CD input low voltage Bit[1]: pgm_vih CD input high voltage-dummy Bit[0]: pgm_hs_valid Valid delay-dummy
0x3018	SC_CMMN_MIPI_SC_CTRL	0x58	RW	Bit[7:5]: mipi_lane_mode 0: One lane mode 1: Two lane mode Bit[6]: r_phy_pd_mipi 0: Not used 1: Power down PHY HS TX Bit[5]: r_phy_pd_lprx 0: Not used 1: Power down PHY LP RX module Bit[6]: mipi_en 0: DVP enable 1: MIPI enable Bit[5]: mipi_susp_reg MIPI system suspend register 0: Not used 1: Suspend Bit[4]: lane_dis_op 0: Use mipi_release1/2 and lane_disable1/2 to disable two data lane 1: Use lane_disable1/2 to disable two data lane Bit[3:0]: Not used
0x3019	SC_CMMN_MIPI_SC_CTRL	0x10	RW	Bit[7:0]: MIPI ULPS resume mark1 detect length

table 7-1 system control registers (sheet 4 of 5)

address	register name	default value	R/W	description
0x301A~ 0x3020	DEBUG MODE	–	–	Debug Mode
0x3021	SC_CMMN_MISC_ CTRL	0x23	RW	Bit[7:6]: Not used Bit[5]: fst_stby_ctr 0: Software standby enter at v_blk 1: Software standby enter at l_blk Bit[4]: mipi_ctr_en 1: Enable MIPI remote reset and suspend control SC 0: Disable the function Bit[3]: mipi_rst_sel 0: MIPI remote reset all registers 1: MIPI remote reset all digital modules Bit[2]: gpio_pclk_en Bit[1]: frex_ef_sel Bit[0]: cen_global_o
0x3022	SC_CMMN_MIPI_SC_ CTRL	–	R	Bit[7:4]: Not used Bit[3]: lptx_ck_opt Bit[2]: pull_down_clk_lane Bit[1]: pull_down_data_lane2 Bit[0]: pull_down_data_lane1
0x302A	SC_CMMN_SUB_ID	–	R	Bit[7:4]: Process Bit[3:0]: Version
0x3034	SC_CMMN_PLL_ CTRL0	0x1A	RW	Bit[7]: Not used Bit[6:4]: pll_charge_pump Bit[3:0]: mipi_bit_mode 0000: 8 bit mode 0001: 10 bit mode Others: Reserved to future use
0x3035	DEBUG MODE	–	–	Debug Mode
0x3036	SC_CMMN_PLL_ MULTIPLIER	0x69	RW	Bit[7:0]: PLL_multiplier (4~252) Can be any integer during 4~127 and only even integer during 128~252
0x3037	SC_CMMN_PLL_ CTR13	0x03	RW	Bit[7:5]: Debug mode Bit[4]: pll_root_div 0: Bypass 1: /2 Bit[3:0]: pll_prediv 1, 2, 3, 4, 6, 8

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

address	register name	default value	R/W	description
0x3038	SC_CMMN_PLL_ DEBUG_OPT	0x00	RW	Bit[7]: pll_mult_debug_en Bit[1:0]: pll_mult1_debug
0x3039	SC_CMMN_PLL_ CTRL_R	0x00	RW	Bit[7]: pll_bypass Bit[6:0]: Not used
0x303A	SC_CMMN_PLLS_ CTRL0	0x00	RW	Bit[7]: plls_bypass Bit[6:0]: Not used
0x303B	SC_CMMN_PLLS_ CTRL1	0x19	RW	Bit[7:5]: Not used Bit[4:0]: plls_multiplier
0x303C	SC_CMMN_PLLS_ CTRL2	0x11	RW	Bit[6:4]: plls_cp Bit[3:0]: plls_sys_div
0x303D	SC_CMMN_PLLS_ CTRL3	0x30	RW	Bit[7:6]: Not used Bit[5:4]: plls_pre_div 00: /1 01: /1.5 10: /2 11: /3 Bit[2]: plls_div_r 0: /1 1: /2 Bit[1:0]: plls_seld5 00: /1 01: /1 10: /2 11: /2.5
0x3040~ 0x3044	DEBUG MODE	–	–	Debug Mode

table 7-2 SCCB registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3100	SCCB ID	0x6C	RW	SCCB Slave ID
0x3100	SCCB CTRL	0x0	RW	Bit[7:4]: Not used Bit[3]: r_sda_dly_en Bit[2:0]: r_sda_dly

table 7-2 SCCB registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3101	SCCB OPT	0x12	RW	Bit[7:5]: en_ss_addr_inc Bit[4]: en_ss_addr_inc Bit[3]: r_sda_byp_sync 0: Two clock stage SYNC for sda_i 1: No SYNC for sda_i Bit[2]: r_scl_byp_sync 0: Two clock stage sync for scl_i 1: No sync for scl_i Bit[1]: r_msk_glitch Bit[0]: r_msk_stop
0x3102	SCCB FILTER	0x00	RW	Bit[7:4]: r_sda_num Bit[3:0]: r_scl_num
0x3103	SCCB SYSREG	0x00	RW	Bit[7]: Not used Bit[6]: ctrl_rst_mipisc Bit[5]: ctrl_rst_srb Bit[4]: ctrl_rst_sccb_s Bit[3]: ctrl_rst_pon_sccb_s Bit[2]: ctrl_rst_clkmod Bit[1]: ctrl_rst_mipi_phy_rst_o Bit[0]: ctrl_pll_rst_o
0x3104	PWUP DIS	0x01	RW	Bit[7:5]: Not used Bit[4]: r_srb_clk_syn_en Bit[3]: pwup_dis2 Bit[2]: pwup_dis1 Bit[1]: pll_clk_sel Bit[0]: pwup_dis0
0x3105	PADCLK DIV	0x11	RW	Bit[7:6]: Not used Bit[5]: sclk use p_clk_i Bit[4]: Sleep enable Bit[3:0]: PAD CLK divider for SCCB
0x3106	SRB CTRL	0xF9	RW	Bit[7:4]: Not used Bit[3:2]: PLL clock divider 00: pll_sclk 01: pll_sclk/2 10: pll_sclk/4 11: pll_sclk Bit[1]: rst_arb 0: Not used 1: Reset arbiter Bit[0]: sclk_arb 0: Not used 1: Enable SCLK to arbiter

table 7-3 group hold control registers

address	register name	default value	R/W	description
0x3200	SRM_GRUP_ADR0	0x00	RW	srm_group_adr0
0x3200	GROUP ADR0	0x00	RW	Group0 Start Address in SRAM, actual address is {0x3200[3:0], 0x0}
0x3201	GROUP ADR1	0x04	RW	Group1 Start Address in SRAM, actual address is {0x3201[3:0], 0x0}
0x3202	GROUP ADR2	0x08	RW	Group2 Start Address in SRAM, actual address is {0x3202[3:0], 0x0}
0x3203	GROUP ADR3	0x0B	RW	Group3 Start Address in SRAM, actual address is {0x3203[3:0], 0x0}
0x3204	GROUP LEN0	–	R	Length of Group0
0x3205	GROUP LEN1	–	R	Length of Group1
0x3206	GROUP LEN2	–	R	Length of Group2
0x3207	GROUP LEN3	–	R	Length of Group3
0x3208	GROUP ACCESS	–	W	Bit[7:4]: Group_ctrl 0000: Enter group write mode 0001: Exit group write mode 1010: Initiate group write Bit[3:0]: Group ID 0000: Group 0 0001: Group 1 0010: Group 2 0011: Group 3
0x3209	DEBUG MODE	–	–	Not Used

table 7-4 AEC/AGC 1 registers

address	register name	default value	R/W	description
0x3500	EXPOSURE	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Exposure[19:16]
0x3501	EXPOSURE	0x00	RW	Bit[7:0]: Exposure[15:8]
0x3502	EXPOSURE	0x20	RW	Bit[7:0]: Exposure[7:0]
0x3503	MANUAL CTRL	0x00	RW	Bit[7:6]: Not used Bit[5:4]: Gain latch timing delay x0: Gain has no latch delay 01: Gain delay of 1 frame 11: Gain delay of 2 frames Bit[2]: VTS manual 0: Auto enable 1: Manual enable Bit[1]: AGC manual 0: Auto enable 1: Manual enable Bit[0]: AEC manual 0: Auto enable 1: Manual enable
0x350A	AGC	0x00	RW	Bit[7:2]: Not used Bit[1:0]: Gain[9:8] AGC real gain output high byte
0x350B	AGC	0x00	RW	Bit[7:0]: Gain[7:0] AGC real gain output low byte
0x350C	VTS DIFF	0x06	RW	Bit[7:0]: vts_diff[15:8] When in manual mode, set to 0x00
0x350D	VTS DIFF	0x18	RW	Bit[7:0]: vts_diff[7:0] When in manual mode, set to 0x00

table 7-5 system timing registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3800	TIMING_X_ADDR_ START	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: x_addr_start[11:8]
0x3801	TIMING_X_ADDR_ START	0x0C	RW	Bit[7:0]: x_addr_start[7:0]
0x3802	TIMING_Y_ADDR_ START	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: y_addr_start[11:8]
0x3803	TIMING_Y_ADDR_ START	0x04	RW	Bit[7:0]: y_addr_start[7:0]

table 7-5 system timing registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3804	TIMING_X_ADDR_END	0x0A	RW	Bit[7:4]: Debug mode Bit[3:0]: x_addr_end[11:8]
0x3805	TIMING_X_ADDR_END	0x33	RW	Bit[7:0]: x_addr_end[7:0]
0x3806	TIMING_Y_ADDR_END	0x07	RW	Bit[7:4]: Debug mode Bit[3:0]: y_addr_end[11:8]
0x3807	TIMING_Y_ADDR_END	0xA3	RW	Bit[7:0]: y_addr_end[7:0]
0x3808	TIMING_X_OUTPUT_SIZE	0x0A	RW	Bit[7:4]: Debug mode Bit[3:0]: DVP output horizontal width[11:8]
0x3809	TIMING_X_OUTPUT_SIZE	0x20	RW	Bit[7:0]: DVP output horizontal width[7:0]
0x380A	TIMING_Y_OUTPUT_SIZE	0x07	RW	Bit[7:4]: Debug mode Bit[3:0]: DVP output vertical height[11:8]
0x380B	TIMING_Y_OUTPUT_SIZE	0x98	RW	Bit[7:0]: DVP output vertical height[7:0]
0x380C	TIMING_HTS	0x0A	RW	Bit[7:5]: Debug mode Bit[4:0]: Total horizontal size[12:8]
0x380D	TIMING_HTS	0x8C	RW	Bit[7:0]: Total horizontal size[7:0]
0x380E	TIMING_VTS	0x07	RW	Bit[7:2]: Debug mode Bit[1:0]: Total vertical size[9:8]
0x380F	TIMING_VTS	0xB0	RW	Bit[7:0]: Total vertical size[7:0]
0x3810	TIMING_ISP_X_WIN	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: ISP horizontal offset[11:8]
0x3811	TIMING_ISP_X_WIN	0x04	RW	Bit[7:0]: ISP horizontal offset[7:0]
0x3812	TIMING_ISP_Y_WIN	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: ISP vertical offset[11:8]
0x3813	TIMING_ISP_Y_WIN	0x02	RW	Bit[7:0]: ISP vertical offset[7:0]
0x3814	TIMING_X_INC	0x11	RW	Bit[7:4]: h_odd_inc Horizontal subsample odd increase number Bit[3:0]: h_even_inc Horizontal subsample even increase number

table 7-5 system timing registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x3815	TIMING_Y_INC	0x11	RW	Bit[7:4]: v_odd_inc Vertical subsample odd increase number Bit[3:0]: v_even_inc Vertical subsample even increase number
0x3816	TIMING_HSYNCST	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: HSYNC start point[11:8]
0x3817	TIMING_HSYNCST	0x00	RW	Bit[7:0]: HSYNCstart point[7:0]
0x3818	TIMING_HSYNCW	0x00	RW	Bit[7:4]: Debug mode Bit[3:0]: HSYNC window[11:8]
0x3819	TIMING_HSYNCW	0x00	RW	Bit[7:0]: HSYNC window[7:0]
0x3820	TIMING_TC_REG20	0x40	RW	Bit[7]: Not used Bit[6:4]: For testing only Bit[3]: Not used Bit[2]: r_vflip_isp Bit[1]: r_vflip_snr Bit[0]: r_vbin
0x3821	TIMING_TC_REG21	0x00	RW	Bit[7:5]: For testing only Bit[4]: Not used Bit[3]: For testing only Bit[2]: r_mirror_isp Bit[1]: r_mirror_snr Bit[0]: r_hbin
0x3822~ 0x3834	DEBUG MODE	–	–	Debug Mode

table 7-6 AEC/AGC 2 registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3A00	AEC CTRL00	0x78	RW	Bit[7]: Not used Bit[6]: Less one line mode Bit[5]: Band function Bit[4]: Band low limit mode Bit[3]: start_sel Bit[2]: Night mode Bit[1]: Not used Bit[0]: Freeze
0x3A01	MIN EXPO	0x01	RW	Bit[7:0]: min expo
0x3A02	MAX EXPO 60	0x3D	RW	Bit[7:0]: max expo[15:8]
0x3A03	MAX EXPO 60	0x80	RW	Bit[7:0]: max expo[7:0]
0x3A05	AEC CTRL05	0x30	RW	Bit[7]: f50_reverse 0: Hold 50, 60Hz detect input 1: Switch 50, 60Hz detect input Bit[6]: frame_insert 0: In night mode, insert frame disable 1: In night mode, insert frame enable Bit[5]: step_auto_en 0: Step manual mode 1: Step auto_mode Bit[4:0]: step_auto_ratio In step auto mode, set the step ratio setting to adjust speed
0x3A06	AEC CTRL06	0x10	RW	Bit[7:5]: Not used Bit[4:0]: step_man1 Step manual Increase mode fast step
0x3A07	AEC CTRL07	0x18	RW	Bit[7:4]: step_man2 Step manual, slow step Bit[3:0]: step_man3 Step manual, decrease mode fast step
0x3A08	B50 STEP	0x01	RW	Bit[7:2]: Not used Bit[1:0]: b50_step[9:8]
0x3A09	B50 STEP	0x27	RW	Bit[7:0]: b50_step[7:0]
0x3A0A	B60 STEP	0x00	RW	Bit[7:2]: Not used Bit[1:0]: b60_step[9:8]
0x3A0B	B60 STEP	0xF6	RW	Bit[7:0]: b60_step[7:0]

table 7-6 AEC/AGC 2 registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3A0C	AEC CTRL0C	0xE4	RW	Bit[7:4]: e1_max Decimal line high limit zone Bit[3:0]: e1_min Decimal line low limit zone
0x3A0D	B60 MAX	0x08	RW	Bit[7:6]: Not used Bit[5:0]: b60_max
0x3A0E	B50 MAX	0x06	RW	Bit[7:6]: Not used Bit[5:0]: b50_max
0x3A0F	WPT	0x78	RW	Bit[7:0]: WPT Stable range high limit (enter)
0x3A10	BPT	0x68	RW	Bit[7:0]: BPT Stable range low limit (enter)
0x3A11	HIGH VPT	0xD0	RW	Bit[7:0]: vpt_high
0x3A12	MANUAL AVG	0x00	RW	Bit[7:0]: avg_man
0x3A13	PRE GAIN	0x40	RW	Bit[7]: Not used Bit[6]: pre-gain enable Bit[5:0]: pre-gain value
0x3A14	MAX EXPO 50	0x0E	RW	Bit[7:0]: Maximum expo[15:8]
0x3A15	MAX EXPO 50	0x40	RW	Bit[7:0]: Maximum expo[7:0]
0x3A17	NIGHT MODE GAIN BASE	0x01	RW	Bit[7:2]: Not used Bit[1:0]: gnight_thre 00: 0x00 01: 0x10 10: 0x30 11: 0x70
0x3A18	AEC GAIN CEILING	0x00	RW	Bit[7:2]: Not used Bit[1:0]: gain_ceiling[9:8]
0x3A19	AEC GAIN CEILING	0x7C	RW	Bit[7:0]: gain_ceiling[7:0]
0x3A1A	DIFF MAX	0x04	RW	Bit[7:0]: diff_max
0x3A1B	WPT2	0x78	RW	Bit[7:0]: wpt2 Stable range high limit (go out)
0x3A1C	LED ADD ROW	0x06	RW	Bit[7:0]: led_add_row[15:8] Exposure values added when STROBE is ON
0x3A1D	LED ADD ROW	0x18	RW	Bit[7:0]: led_add_row[7:0] Exposure values added when STROBE is ON

table 7-6 AEC/AGC 2 registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x3A1E	BPT2	0x68	RW	Bit[7:0]: bpt2 Stable range low limit (go out)
0x3A1F	LOW VPT	0x40	RW	Bit[7:0]: vpt_low Step manual mode, fast zone low limit
0x3A20	AEC CTRL20	0x00	RW	Bit[7:2]: Not used Bit[1]: man_avg_en_i 0: Disable 1: Enable Bit[0]: Not used
0x3A21	AEC CTRL21	0x70	RW	Bit[7:]: Not used Bit[6:4]: Frame insert number Bit[3:0]: Not used

table 7-7 STROBE/frame exposure control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3B00	STROBE_RSTRB	0x00	RW	Strobe Control Bit[7]: Strobe request ON/OFF 0: OFF/BLC 1: ON Bit[6]: Strobe pulse reverse Bit[3:2]: width_in_xenon 00: 1 row period 01: 2 row period 10: 3 row period 11: 4 row period Bit[1:0]: Strobe mode 00: xenon 01: LED 1 10: LED 2 11: LED 3
0x3B01	STROBE_FREX_EXP_H2	0x00	RW	Bit[7:0]: frex_exp[23:16]
0x3B02	STROBE_SHUTTER_DLY	0x08	RW	Bit[7:0]: shutter_dly[12:8]
0x3B03	STROBE_SHUTTER_DLY	0x00	RW	Bit[7:0]: shutter_dly[7:0]
0x3B04	STROBE_FREX_EXP_H	0x04	RW	Bit[7:0]: frex_exp[15:8]
0x3B05	STROBE_ FREX_EXP_L	0x00	RW	Bit[7:0]: frex_exp[7:0]

table 7-7 STROBE/frame exposure control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3B06	STROBE_FREX_CTRL0	0x04	RW	Bit[7:6]: frex_pchg_width Bit[5:4]: frex_strobe_option Bit[3:0]: frex_strobe_width[3:0]
0x3B07	STROBE_FREX_MODE_SEL	0x08	RW	Bit[4]: frex_sa1 Bit[3]: fx1_fm_en Bit[2]: frex_inv Bit[1:0]: FREX strobe 00: frex_strobe mode0 01: frex_strobe mode1 1x: Rolling strobe
0x3B08	STROBE_FREX_EXP_REQ	0x00	RW	Bit[7:1]: Not used Bit[0]: frex_exp_req
0x3B09	FREX_SHUTTER_DELAY	0x00	RW	Bit[7:3]: Not used Bit[2:0]: FREX end option
0x3B0A	STROBE_FREX_RST_LENGTH	0x04	RW	Bit[7:3]: Not used Bit[2:0]: frex_rst_length[2:0]
0x3B0B	STROBE_WIDTH	0x00	RW	Bit[7:0]: frex_strobe_width[19:12]
0x3B0C	STROBE_WIDTH	0x3D	RW	Bit[7:0]: frex_strobe_width[11:4]

table 7-8 50/60 HZ DETECTION registers

address	register name	default value	R/W	description
0x3C00	50/60 HZ DETECTION CTRL00	0x00	RW	Bit[7:6]: Debug control Changing these registers is not recommended Bit[5:3]: 50/60 Hz detection control Contact local OmniVision FAE for the correct settings Bit[2]: band_def Band50 default value 0: 60 Hz as default value 1: 50 Hz as default value Bit[1:0]: 50/60 Hz detection control register Contact local OmniVision FAE for the correct settings
0x3C01	50/60 HZ DETECTION CTRL01	0x00	RW	Bit[7]: band_man_en Band detection manual mode 0: Manual mode disable 1: Manual mode enable Bit[6:0]: 50/60 Hz detection control Contact local OmniVision FAE for the correct settings
0x3C02~ 0x3C0B	50/60 HZ DETECTION CTRL02	–	RW	50/60 Hz detection Control Contact local OmniVision FAE for the correct settings
0x3C0C	50/60 HZ DETECTION CTRL0C	–	R	Bit[7:1]: Debug control Changing these registers is not recommended Bit[0]: band50 0: Detection result is 60 Hz 1: Detection result is 50 Hz
0x3C0D~ 0x3C1E	DEBUG INFORMATION	–	RW	50/60 Hz Detection Control Contact local OmniVision FAE for the correct settings

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

address	register name	default value	R/W	description
0x3D00	OTP_DATA_0	0x00	RW	OTP Buffer 0
0x3D01	OTP_DATA_1	0x00	RW	OTP Buffer 1
0x3D02	OTP_DATA_2	0x00	RW	OTP Buffer 2
0x3D03	OTP_DATA_3	0x00	RW	OTP Buffer 3
0x3D04	OTP_DATA_4	0x00	RW	OTP Buffer 4
0x3D05	OTP_DATA_5	0x00	RW	OTP Buffer 5
0x3D06	OTP_DATA_6	0x00	RW	OTP Buffer 6
0x3D07	OTP_DATA_7	0x00	RW	OTP Buffer 7
0x3D08	OTP_DATA_8	0x00	RW	OTP Buffer 8
0x3D09	OTP_DATA_9	0x00	RW	OTP Buffer 9
0x3D0A	OTP_DATA_A	0x00	RW	OTP Buffer A
0x3D0B	OTP_DATA_B	0x00	RW	OTP Buffer B
0x3D0C	OTP_DATA_C	0x00	RW	OTP Buffer C
0x3D0D	OTP_DATA_D	0x00	RW	OTP Buffer D
0x3D0E	OTP_DATA_E	0x00	RW	OTP Buffer E
0x3D0F	OTP_DATA_F	0x00	RW	OTP Buffer F
0x3D10	OTP_DATA_16	0x00	RW	OTP Buffer 10
0x3D11	OTP_DATA_17	0x00	RW	OTP Buffer 11
0x3D12	OTP_DATA_18	0x00	RW	OTP Buffer 12
0x3D13	OTP_DATA_19	0x00	RW	OTP Buffer 13
0x3D14	OTP_DATA_20	0x00	RW	OTP Buffer 14
0x3D15	OTP_DATA_21	0x00	RW	OTP Buffer 15
0x3D16	OTP_DATA_22	0x00	RW	OTP Buffer 16
0x3D17	OTP_DATA_23	0x00	RW	OTP Buffer 17
0x3D18	OTP_DATA_24	0x00	RW	OTP Buffer 18
0x3D19	OTP_DATA_25	0x00	RW	OTP Buffer 19
0x3D1A	OTP_DATA_26	0x00	RW	OTP Buffer 1A
0x3D1B	OTP_DATA_27	0x00	RW	OTP Buffer 1B

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

address	register name	default value	R/W	description
0x3D1C	OTP_DATA_28	0x00	RW	OTP Buffer 1C
0x3D1D	OTP_DATA_29	0x00	RW	OTP Buffer 1D
0x3D1E	OTP_DATA_30	0x00	RW	OTP Buffer 1E
0x3D1F	OTP_DATA_31	0x00	RW	OTP Buffer 1F
0x3D20	OTP_PROGRAM_CTRL	0x00	RW	Bit[7]: OTP_wr_busy Bit[6:2]: Debug control Changing these registers is not recommended Bit[1]: OTP_program_speed 0: Fast 1: Slow Bit[0]: OTP_program_enable Changing from 0 to 1 initiates OTP programming
0x3D21	OTP_LOAD_CTRL	0x00	RW	Bit[7]: OTP_rd_busy Bit[1]: OTPspeed 0: Fast 1: Slow Bit[0]: OTP_load_enable Changing from 0 to 1 initiates OTP read

table 7-10 BLC registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x4000	BLC CTRL00	0x89	RW	BLC Control (0: disable ISP; 1: enable ISP) Bit[7]: blc_median_filter_enable Bit[6:4]: Not used Bit[3]: adc_11bit_mode Bit[2]: apply2blackline Bit[1]: blackline_averageframe Bit[0]: BLC enable
0x4001	BLC CTRL01	0x00	RW	Bit[7:6]: Not used Bit[5:0]: start_line
0x4002	BLC CTRL02	0x45	RW	Bit[7]: format_change_en format_change_i from fmt will be effect when it is enable Bit[6]: blc_auto_en Bit[5:0]: reset_frame_num
0x4003	BLC CTRL03	0x08	RW	Bit[7]: blc_redo_en Write 1 into it will trigger a BLC redo N frames begin Bit[6]: Freeze Bit[5:0]: manual_frame_num
0x4004	BLC CTRL04	0x08	RW	Bit[7:0]: blc_line_num
0x4005	BLC CTRL05	0x18	RW	Bit[7:6]: Not used Bit[5]: one_line_mode Bit[4]: remove_none_imagedata Bit[3]: blc_man_1_en Bit[2]: blackline_bggr_man_en 0: bgbg/grgr is decided by rblue/hswap 1: bgbg/grgr fix Bit[1]: bgbg/grgr is decided by rblue/hswap blc_always_up_en 0: Normal freeze 1: BLC always update Bit[0]: Not used
0x4006	BLC CTRL06	0x08	RW	Bit[7:6]: Not used Bit[5]: bl_num_man_en Bit[4:0]: bl_num_man

table 7-10 BLC registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x4007	BLC CTRL07	0x00	RW	Bit[7:5]: Not used Bit[4:3]: win_sel 00: Full image 01: Windows do not contain the first 16 pixels and the last 16 pixels 10: Windows do not contain the first 1/16 image and the last 1/16 image 11: Windows do not contain the first 1/8 image and the last 1/8 image Bit[2:0]: Bypass_mode 000: Bypass data_i after limit bits 001: Bypass data_i[11:0] 011: Bypass data_i[12:1] 100: Bypass debug data bbrr 101: Bypass debug data gggg 1xx: Not used
0x4008	BLC CTRL08	0x00	RW	BLC Control (0: disable ISP; 1: enable ISP) Bit[7:4]: Not used Bit[3]: flip_man_en Bit[2]: flip_man Bit[1]: bl_flip_man_en Bit[0]: bl_flip_man
0x4009	BLACK LEVEL	0x10	RW	Bit[7:0]: blc_blackleveltarget0
0x400A~ 0x400B	DEBUG MODE	–	–	Debug Mode
0x400C	BLC MAN0	0x00	RW	Bit[7:0]: blc_man0[15:8]
0x400D	BLC MAN0	0x00	RW	Bit[7:0]: blc_man0[7:0]
0x400E	BLC MAN1	0x00	RW	Bit[7:0]: blc_man1[15:8]
0x400F	BLC MAN1	0x00	RW	Bit[7:0]: blc_man1[7:0]
0x4010	BLC MAN2	0x00	RW	Bit[7:0]: blc_man2[15:8]
0x4011	BLC MAN2	0x00	RW	Bit[7:0]: blc_man2[7:0]
0x4012	BLC MAN3	0x00	RW	Bit[7:0]: blc_man3[15:8]
0x4013	BLC MAN3	0x00	RW	Bit[7:0]: blc_man3[7:0]
0x402C	BLACK_LEVEL00	–	R	Bit[7:0]: blacklevel00[15:8]
0x402D	BLACK_LEVEL00	–	R	Bit[7:0]: blacklevel00[7:0]

table 7-10 BLC registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x402E	BLACK_LEVEL01	–	R	Bit[7:0]: blacklevel01[15:8]
0x402F	BLACK_LEVEL01	–	R	Bit[7:0]: blacklevel01[7:0]
0x4030	BLACK_LEVEL10	–	R	Bit[7:0]: blacklevel10[15:8]
0x4031	BLACK_LEVEL10	–	R	Bit[7:0]: blacklevel10[7:0]
0x4032	BLACK_LEVEL11	–	R	Bit[7:0]: blacklevel11[15:8]
0x4033	BLACK_LEVEL11	–	R	Bit[7:0]: blacklevel11[7:0]
0x4050	BLC MAX	0xFF	RW	Bit[7:0]: blc max black level
0x4051	STABLE RANGE	0x7F	RW	Bit[7:0]: BLC stable range
0x4052	ONE CHANNEL	0x00	RW	Bit[7:0]: blc_one_channel
0x4060	BLC BR THRE0	0x00	RW	Bit[7:0]: blc_br_thr_0
0x4061	BLC BR THRE1	0x00	RW	Bit[7:0]: blc_br_thr_1
0x4062	BLC BR THRE2	0x00	RW	Bit[7:0]: blc_br_thr_2
0x4063	BLC BR THRE3	0x00	RW	Bit[7:0]: blc_br_thr_3
0x4064	BLC BR THRE4	0x00	RW	Bit[7:0]: blc_br_thr_4
0x4065	BLC BR THRE5	0x00	RW	Bit[7:0]: blc_br_thr_5
0x4066	BLC G THRE0	0x00	RW	Bit[7:0]: blc_g_thr_0
0x4067	BLC G THRE1	0x00	RW	Bit[7:0]: blc_g_thr_1
0x4068	BLC G THRE2	0x00	RW	Bit[7:0]: blc_g_thr_2
0x4069	BLC G THRE3	0x00	RW	Bit[7:0]: blc_g_thr_3
0x406A	BLC G THRE4	0x00	RW	Bit[7:0]: blc_g_thr_4
0x406B	BLC G THRE5	0x00	RW	Bit[7:0]: blc_g_thr_5
0x406C	BLC BRG COMP EN	0x00	RW	Bit[7:0]: blc_brg_comp_en

table 7-11 frame control registers

address	register name	default value	R/W	description
0x4200	FRAME CTRL0	0x00	RW	Bit[7:3]: Not used Bit[2]: fcnt_eof_sel Bit[1]: fcnt_mask_dis Bit[0]: fcnt_reset
0x4201	FRAME ON NUMBER	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Frame ON number
0x4202	FRAME OFF NUMBER	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Frame OFF number
0x4203	FRAME CTRL1	0x00	RW	Bit[7:6]: Not used 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

table 7-12 DVP registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4700	MODE SELECT	0x04	RW	Bit[7:4]: Not used Bit[3]: CCIR V select Bit[2]: CCIR F select Bit[1]: CCIR656 mode enable Bit[0]: HSYNC mode enable
0x4701	VSYNC WIDTH	0x01	RW	VSYNC Length in Terms of Line Count
0x4702	VSYNC NEG_WIDTH_H	0x01	RW	Bit[7:0]: VSYNC length in terms of pixel count[15:8]
0x4703	VSYNC NEG_WIDTH_L	0x00	RW	Bit[7:0]: VSYNC length in terms of pixel count[7:0]
0x4704	VSYNC MODE	0x00	RW	Bit[7:4]: Not used Bit[3:2]: r_vsyncout_sel Bit[1]: VSYNC mode3 Bit[0]: VSYNC mode2
0x4705	EOF VSYNC_DELAY_2	0x00	RW	Bit[7:0]: eof_vsync_delay[23:16] SOF/EOF negative edge to VSYNC positive edge delay
0x4706	EOF VSYNC_DELAY_1	0x00	RW	Bit[7:0]: eof_vsync_delay[15:8] SOF/EOF negative edge to VSYNC positive edge dealy

table 7-12 DVP registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4707	EOF VSYNC_DELAY_0	0x00	RW	Bit[7:0]: eof_vsync_delay[7:0] SOF/EOF negative edge to VSYNC positive edge delay
0x4708	POLARITY CTRL	0x01	RW	Bit[7]: Clock DDR mode enable Bit[6]: Not used Bit[5]: VSYNC gate clock enable Bit[4]: HREF gate clock enable Bit[3]: No frst for FIFO Bit[2]: HREF polarity reverse option Bit[1]: VSYNC polarity reverse option Bit[0]: PCLK polarity reverse option
0x4709	MOTO ORDER	0x00	RW	Bit[7]: FIFO bypass mode Bit[6:4]: Data bit swap Bit[3]: Bit test mode Bit[2]: 10-bit bit test Bit[1]: 8-bit bit test Bit[0]: Bit test enable
0x470A	BYP CTRL1	0x00	RW	Bit[7:0]: bypass_ctrl[15:8]
0x470B	BYP CTRL0	0x00	RW	Bit[7:0]: bypass_ctrl[7:0]
0x470C	BYP SEL	0x00	RW	Bit[7:5]: Not used Bit[4]: href_sel Bit[3:0]: bypass_sel

table 7-13 MIPI top registers (sheet 1 of 11)

address	register name	default value	R/W	description
0x4800	MIPI CTRL 00	0x04	RW	MIPI Control 00 Bit[7]: mipi_hs_only 0: MIPI can support CD and ESCAPE mode 1: MIPI always in high speed mode Bit[6]: ck_mark1_en 0: Not used 1: Enable clock lane mark1 when resume 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[3]: Lane select 0: Use lane1 as default data lane 1: Use lane2 as default data lane Bit[2]: Idle status 0: MIPI bus will be LP00 when no packet to transmit 1: MIPI bus will be LP11 when no packet to transmit Bit[1]: Clock lane first bits 0: Output 0x55 1: Output 0xAA Bit[0]: Clock lane disable 0: Not used 1: Manually set clock lane to low power mode

table 7-13 MIPI top registers (sheet 2 of 11)

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]: Short packet WORD COUNTER manual enable 0: Use frame counter or line counter 1: Select spkt_wc_reg_o (see {0x4812, 0x4813}) 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]: mark1_en1 0: Not used 1: After each rst release, lane 1 should send mark1 for wkup_dly_o when mipi_sys_susp =1 Bit[0]: mark1_en2 0: Not used 1: After each reset release, lane 2 should send mark1 for wkup_dly_o when mipi_sys_susp=1

table 7-13 MIPI top registers (sheet 3 of 11)

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_o[7:0]

table 7-13 MIPI top registers (sheet 4 of 11)

address	register name	default value	R/W	description
0x4803	MIPI CTRL 03	0x50	RW	MIPI Control 03 Bit[7:6]: lp_glitch_nu 0: Use 2d of lp_in 1: Mask one SCLK cycle glitch of lp_in Bit[5:4]: cd_glitch_nu 0: Use 2d of lp_cd_in 1: Mask one SCLK cycle glitch of lp_cd_in Bit[3]: Enable CD plus of data lane1 0: Disable 1: Enable Bit[2]: Enable CD plus of data lane2 0: Disable 1: Enable Bit[1]: Enable CD of data_lane1 from PHY 0: Disable 1: Enable Bit[0]: Enable CD of data_lane2 from PHY 0: Disable 1: Enable

table 7-13 MIPI top registers (sheet 5 of 11)

address	register name	default value	R/W	description
0x4804	MIPI CTRL 04	0x8D	RW	MIPI Control 04 Bit[7]: wait_pkt_end 0: Not used 1: Wait HS packet end when send UL command Bit[6]: tx_lsb_first 0: lp_tx and lp_rx high bit first 1: Low power transmit low bit first Bit[5]: dir_recover_sel 0: Auto change to output only when TurnAround command 1: Auto change to output when LP11 and GPIO is output Bit[4]: mipi_reg_en 0: Disable MIPI_REG_P to access registers, LP data will write to VFIFO 1: Enable MIPI_REG_P to access registers Bit[3]: Address read/write register will auto add 1 0: Disable 1: Enable Bit[2]: LP TX lane select 0: Select lane1 to transmit LP data 1: Select lane2 to transmit LP data Bit[1]: wr_first_byte 0: Not used 1: lp_rx will write first byte (command byte) to RAM Bit[0]: rd_ta_en 0: Not used 1: Send TurnAround command after sending register read data

table 7-13 MIPI top registers (sheet 6 of 11)

address	register name	default value	R/W	description
0x4805	MIPI CTRL 05	0x10	RW	MIPI Control 05 Bit[7]: MIPI lane1 disable 0: Not used 1: Disable MIPI data lane1, lane1 will be LP00 Bit[6]: MIPI lane2 disable 0: Not used 1: Disable MIPI data lane2, lane2 will be LP00 Bit[5]: lpx_p_sel 0: Automatically calculate t_lpx_o in pclkex domain, unit pclk2x 1: Use lp_p_min[7:0] Bit[4]: lp_rx_intr_sel 0: Send lp_rx_intr_o at the first byte 1: Send lp_rx_intr_o at the end of receiving Bit[3]: cd_tst_sel 0: Not used 1: Select PHY test pins Bit[2]: mipi_reg_mask 0: Not used 1: Disable MIPI access SRB Bit[1]: clip enable Bit[0]: hd_sk_en 0: Disable MIPI and MCU handshake registers 1: Disable MIPI and MCU handshake registers

table 7-13 MIPI top registers (sheet 7 of 11)

address	register name	default value	R/W	description
0x4806	MIPI REG RW CTRL	0x28	RW	Bit[7]: Test mode Bit[6]: mipi_test Bit[5]: mipi_lp_op 0: Use new option to reduce mipi_lptx_p 1: Not used Bit[4]: two_lane_man_en 0: Not used 1: Use two_lane_man to manually control two_lane_mode Bit[3]: two_lane_man Bit[2]: rst_rtn_en 0: Not used 1: Change to input to allow host RW register after reset Bit[1]: frame_end_en 0: Not used 1: After frame end packet, change to input to allow host RW register Bit[0]: line_end_en 0: Not used 1: After line end packet, change to input to allow host RW register
0x480A	MIPI BIT ORDER	0x00	RW	Bit[7:3]: Not used Bit[2]: Bit order reverse Bit[1:0]: Bit position adjustment 01: {D[7:0],D[9:8]} 10: {D[1:0],D[9:2]}
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
0x4814	MIPI CTRL14	0x2A	RW	MIPI Control 14 Bit[7:6]: Virtual channel of MIPI Bit[5:0]: Data type in manual mode
0x4815	MIPI_DT_SPKT	0x00	RW	Bit[7]: Not used Bit[6]: pclk_div 0: Use rising edge of mipi_pclk_o to generate MIPI bus to PHY 1: Use falling edge of mipi_pclk_o to generate MIPI bus to PHY Bit[5:0]: Manual data type for short packet
0x4818	HS_ZERO_MIN	0x00	RW	High byte of the minimum value for hs_zero Unit ns

table 7-13 MIPI top registers (sheet 8 of 11)

address	register name	default value	R/W	description
0x4819	HS_ZERO_MIN	0x96	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, 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	0x86	RW	Low byte of the minimum value for clk_zero, clk_zero_real = clk_zero_min_o + Tui*ui_clk_zero_min_o
0x481E	CLK_PREPARE_ MIN	0x00	RW	High byte of the minimum value for clk_prepare, Unit ns Bit[7:2]: Not used Bit[1:0]: clk_prepare_min[9:8]
0x481F	CLK_PREPARE_ MIN	0x3C	RW	Low byte of the 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[7:2]: Not used Bit[1:0]: clk_post_min[9:8]
0x4821	CLK_POST_MIN	0x56	RW	Low byte of the minimum value for clk_post clk_post_real = clk_post_min_o + Tui*ui_clk_post_min_o
0x4822	CLK_TRAIL_MIN	0x00	RW	High byte of the minimum value for clk_trail, unit ns Bit[7:2]: Not used 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
0x4824	LPX_P_MIN	0x00	RW	High byte of the minimum value for lpx_p, unit ns Bit[7:2]: Not used 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
0x4826	HS_PREPARE_MIN	0x00	RW	High byte of the minimum value for hs_prepare, unit ns Bit[7:2]: Not used Bit[1:0]: hs_prepare_min[9:8]

table 7-13 MIPI top registers (sheet 9 of 11)

address	register name	default value	R/W	description
0x4827	HS_PREPARE_MIN	0x32	RW	Low byte of the minimum 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[7:2]: Not used 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
0x482A	UI_HS_ZERO_MIN	0x05	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 _MIN	0x00	RW	Minimum UI 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	0x04	RW	Minimum UI Value of hs_prepare, unit UI
0x4832	UI_HS_EXIT_MIN	0x00	RW	Minimum UI Value of hs_exit, unit UI
0x4833	MIPI_REG_MIN	0x00	RW	MIPI register address, lower bound (high byte) Address range of MIPI RW registers is from mipi_reg_min to mipi_reg_max
0x4834	MIPI_REG_MIN	0x00	RW	MIPI Register Address, lower bound (low byte)
0x4835	MIPI_REG_MAX	0xFF	RW	MIPI Register Address, upper bound (high byte)
0x4836	MIPI_REG_MAX	0xFF	RW	MIPI Register Address, upper bound (low byte)
0x4837	PCLK_PERIOD	0x15	RW	Period of pclk2x, pclk_div = 1, and 1-bit decimal
0x4838	WKUP_DLY	0x02	RW	Wakeup Delay for MIPI
0x483A	DIR_DLY	0v08	RW	Change LP Direction Delay/2 after LP11

table 7-13 MIPI top registers (sheet 10 of 11)

address	register name	default value	R/W	description
0x483B	MIPI_LP_GPIO	0x33	RW	Bit[7]: lp_sel1 0: Generate mipi_lp_dir1_o automatically 1: Use lp_dir_man1 as mipi_lp_dir1_o Bit[6]: lp_dir_man1 0: Input 1: Output Bit[5]: lp_p1_o Bit[4]: lp_n1_o Bit[3]: lp_sel2 0: Generate mipi_lp_dir2_o automatically 1: Use lp_dir_man2 as mipi_lp_dir2_o Bit[2]: lp_dir_man2 0: Input 1: Output Bit[1]: lp_p2_o Bit[0]: lp_n2_o
0x483C	MIPI CTRL 33	0x4F	RW	Bit[7:4]: t_lpx, unit: sclk cycles Bit[3:0]: t_clk_pre, unit: sclk cycles
0x483D	MIPI_T_TA_GO	0x10	RW	t_ta_go Unit: SCLK cycles
0x483E	MIPI_T_TA_SURE	0x06	RW	t_ta_sure Unit: SCLK cycles
0x483F	MIPI_T_TA_GET	0x14	RW	t_ta_get Unit: SCLK cycles
0x4843	SNR_PCLK_DIV	0x00	RW	Bit[7:1]: Not used Bit[0]: PCLK divider 0: PCLK/SCLK = 2 and pclk_div = 1 1: PCLK/SCLK = 1 and pclk_div = 1
0x4860	MIPI CTRL 60	–	R	MIPI Read/Write only Bit[7:1]: Not used Bit[0]: mipi_dis_me 0: Enable MIPI read/write registers 1: Disable MIPI read/write registers
0x4861	HD_SK_REG0	–	R	MIPI Read/Write, SCCB and MCU Read Only
0x4862	HD_SK_REG1	–	R	MIPI Read/Write, SCCB and MCU Read Only
0x4863	HD_SK_REG2	–	R	MIPI Read/Write, SCCB and MCU Read Only
0x4864	HD_SK_REG3	–	R	MIPI Read/Write, SCCB and MCU Read Only

table 7-13 MIPI top registers (sheet 11 of 11)

address	register name	default value	R/W	description
0x4865	MIPI_ST	–	R	Bit[7:6]: Not used Bit[5]: lp_rx_sel_i 0: Not used 1: MIPI_LP_RX receives LP data Bit[4]: tx_busy_i 0: Not used 1: MIPI_TX_LP_TX is busy to send LP data Bit[3]: mipi_lp_p1_i MIPI low power input for lane 1p Bit[2]: mipi_lp_n1_i MIPI low power input for lane 1n Bit[1]: mipi_lp_p2_i MIPI low power input for lane 2p Bit[0]: mipi_lp_n2_i MIPI low power input for lane 2n
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

table 7-14 ISPFC registers

address	register name	default value	R/W	description
0x4900	FRAME CTRL0	0x00	RW	Bit[7:3]: Not used Bit[2]: fcnt_eof_sel Bit[1]: fcnt_mask_dis Bit[0]: fcnt_reset
0x4901	FRAME ON NUMBER	0x00	RW	Bit[7:3]: Not used Bit[3:0]: Frame ON number
0x4902	FRAME OFF NUMBER	0x00	RW	Bit[7:3]: Not used Bit[3:0]: Frame OFF number
0x4903	FRAME CTRL1	0x00	RW	Bit[7:6]: Not used 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

table 7-15 ISP TOP control registers (sheet 1 of 6)

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0xFF	RW	Bit[7]: lenc_en 0: Disable 1: Enable Bit[6:3]: Not used Bit[2]: bc_en 0: Disable 1: Enable Bit[1]: wc_en 0: Disable 1: Enable Bit[0]: Not used
0x5001	ISP CTRL01	0x01	RW	Bit[7:1]: Not used Bit[0]: awb_en 0: Disable 1: Enable
0x5002	ISP CTRL02	0x41	RW	Bit[7]: Not used Bit[6]: win_en 0: Disable 1: Enable Bit[1]: otp_en 0: Disable 1: Enable Bit[0]: awb_gain_en 0: Disable 1: Enable
0x5003	ISP CTRL03	0x0A	RW	Bit[7:4]: Not used Bit[3]: buf_en 0: Disable 1: Enable Bit[2]: bin_man_set 0: Manual value as 0 1: Manual value as 1 Bit[1]: bin_auto_en 0: Disable 1: Enable Bit[0]: Not used
0x5004	ISP CTRL04	0x00	RW	Bit[7:4]: Not used Bit[3]: size_man_en 0: Disable 1: Enable Bit[2:0]: Not used

table 7-15 ISP TOP control registers (sheet 2 of 6)

address	register name	default value	R/W	description
0x5005	ISP CTRL05	0x31	RW	Bit[7]: sof_man 0: SOF from BLC module 1: SOF from pre_isp module Bit[6]: awb_bias_man_en 0: AWB bias manual disable 1: AWB bias manual enable Bit[5]: awb_bias_on 0: Disable AWB bias 1: Enable AWB bias Bit[4:3]: Not used Bit[2]: lenc_bias_on 0: Disable LENC bias 1: Enable LENC bias Bit[1]: Disable LENC bias s2p_sw_en_o Bit[0]: Disable LENC bias avg_en 0: Disable 1: Enable
0x5006	ISP CTRL06	0x00	RW	ISP Control (0: disable ISP; 1: enable ISP) Bit[7]: x_odd_inc_man_en Bit[6]: y_even_inc_man_en Bit[5]: x_odd_inc_man_en Bit[4]: y_even_inc_man_en Bit[3]: x_offset_man_en Bit[2]: y_offset_man_en Bit[1]: x_skip_man_en Bit[0]: y_skip_man_en
0x5007	ISP CTRL07	0x00	RW	ISP Control (0: disable ISP; 1: enable ISP) Bit[7]: bin_mode_man_en Bit[6]: bin_mode_man Bit[5]: win_x_off_man_en Bit[4]: win_y_off_man_en Bit[3]: win_x_out_man_en Bit[2]: win_y_out_man_en Bit[1]: isp_input_h_man_en Bit[0]: isp_input_v_man_en
0x5008	X OFFSET MAN	0x00	RW	Bit[7:4]: Not used Bit[3:0]: x_offset_man[11:8]
0x5009	X OFFSET MAN	0x00	RW	Bit[7:0]: x_offset_man[7:0]
0x500A	Y OFFSET MAN	0x00	RW	Bit[7:3]: Not used Bit[2:0]: y_offset_man[10:8]
0x500B	Y OFFSET MAN	0x00	RW	Bit[7:0]: y_offset_man[7:0]

table 7-15 ISP TOP control registers (sheet 3 of 6)

address	register name	default value	R/W	description
0x500C	WIN X OFFSET MAN	0x00	RW	Bit[7:4]: Not used Bit[3:0]: win_x_offset_man[11:8]
0x500D	WIN X OFFSET MAN	0x00	RW	Bit[7:0]: win_x_offset_man[7:0]
0x500E	WIN Y OFFSET MAN	0x00	RW	Bit[7:3]: Not used Bit[2:0]: win_y_offset_man[10:8]
0x500F	WIN Y OFFSET MAN	0x00	RW	Bit[7:0]: win_y_offset_man[7:0]
0x5010	WIN X OUT MAN	0x00	RW	Bit[7:4]: Not used Bit[3:0]: win_x_out_man[11:8]
0x5011	WIN X OUT MAN	0x00	RW	Bit[7:0]: win_x_out_man[7:0]
0x5012	WIN Y OUT MAN	0x00	RW	Bit[7:3]: Not used Bit[2:0]: win_y_out_man[10:8]
0x5013	WIN Y OUT MAN	0x00	RW	Bit[7:0]: win_y_out_man[7:0]
0x5014	ISP INPUT X MAN	0x00	RW	Bit[7:4]: Not used Bit[3:0]: isp_x_input_man[11:8]
0x5015	ISP INPUT X MAN	0x00	RW	Bit[7:0]: isp_x_input_man[7:0]
0x5016	ISP INPUT Y MAN	0x00	RW	Bit[7:3]: Not used Bit[2:0]: isp_y_input_man[10:8]
0x5017	ISP INPUT Y MAN	0x00	RW	Bit[7:0]: isp_y_input_man[7:0]
0x5018	ISP CTRL18	0x00	RW	Bit[7:4]: x_odd_inc_man Bit[3:0]: x_even_inc_man
0x5019	ISP CTRL19	0x00	RW	Bit[7:4]: y_odd_inc_man Bit[3:0]: y_even_inc_man
0x501A	ISP CTRL1A	0x00	RW	Bit[7:4]: Not used Bit[3:2]: x_skip_man Bit[1:0]: y_skip_man
0x501B~ 0x501C	DEBUG MODE	–	–	Debug Mode
0x501D	ISP CTRL1D	0x00	RW	Bit[7]: Not used Bit[6:4]: win_y_offset_adjust Bit[3:0]: Not used

table 7-15 ISP TOP control registers (sheet 4 of 6)

address	register name	default value	R/W	description
0x501F	ISP CTRL1F	0x03	RW	Bit[7:6]: Not used Bit[5]: enable_opt 1: Enable latched by VSYNC 0: Not latched by VSYNC Bit[4]: cal_sel 0: DPC cal_start using SOF 1: DPC cal_start using VSYNC Bit[3]: Not used Bit[2:0]: fmt_sel 0: ISP output data 1: ISP input data bypass
0x5025	ISP CTRL25	0x00	RW	Bit[7:4]: Not used Bit[1:0]: avg_sel 00: Inputs of AVG module are from LENC output 01: Inputs of AVG module are from AWB gain output 10: Inputs of AVG module are from DPC output 11: Inputs of AVG module are from binning output
0x5026~ 0x503C	DEBUG MODE	–	–	Debug Mode
0x503D	ISP CTRL3D	0x00	RW	Bit[7]: test_pattern_en 0: Disable 1: Enable Bit[6]: rolling_bar 0: Disable rolling bar 1: Enable rolling bar Bit[5]: transparent_mode 0: Disable 1: Enable Bit[4]: squ_bw_mode 0: Output square is color square 1: Output square is black-white square Bit[3:2]: bar_style When set to a different value, a different type color bar will be output Bit[1:0]: test_pattern_type 00: Color bar 01: Square 10: Random data 11: Input data

table 7-15 ISP TOP control registers (sheet 5 of 6)

address	register name	default value	R/W	description
0x503E	ISP CTRL3E	0x00	RW	Bit[7]: Not used Bit[6]: win_cut_en Bit[5]: isp_test 0: Two lowest bits are 1 1: Two lowest bits are 0 Bit[4]: Two lowest bits are rnd_same 0: Frame-changing random data pattern 1: Frame-fixed random data pattern Bit[3:0]: rnd_seed Initial seed for random data pattern
0x504B	ISP CTRL4B	0x30	RW	ISP Control (0: disable ISP; 1: enable ISP) Bit[7:6]: Not used Bit[5]: post_binning h_enable Bit[4]: post_binning v_enable Bit[3]: flip_man_en Bit[2]: flip_man Bit[1]: mirror_man_en Bit[0]: Mirror
0x504C	ISP CTRL4C	0x04	RW	Bit[7:0]: bias_man
0x504D	ISP CTRL4D	0x00	RW	ISP Control (0: Disable ISP; 1: Enable ISP) Bit[7:4]: Not used Bit[3]: lenc_xoff_man_en Bit[2]: lenc_yoff_man_en Bit[1]: lenc_gain_man_en Bit[0]: lenc_bias_man_en
0x504E	ISP CTRL4E	0x04	RW	Bit[7:4]: Not used Bit[3:0]: lenc_xoff_man[11:8]
0x504F	ISP CTRL4F	0x00	RW	Bit[7:0]: lenc_xoff_man[7:0]
0x5052	ISP CTRL52	0x0A	RW	Bit[7:4]: Not used Bit[3:0]: lenc_yoff_man[11:8]
0x5053	ISP CTRL53	0x00	RW	Bit[7:0]: lenc_yoff_man[7:0]
0x5054	ISP CTRL54	0x00	RW	Bit[7:2]: Not used Bit[1:0]: lenc_gain_man[9:8]
0x5055	ISP CTRL55	0x00	RW	Bit[7:0]: lenc_gain_man[7:0]
0x5056	ISP CTRL56	0x00	RW	Bit[7:6]: Not used Bit[5]: lenc_skipx_man Bit[4]: lenc_skipy_man Bit[3:2]: lenc_skipy_man Bit[1:0]: lenc_skipx_man

table 7-15 ISP TOP control registers (sheet 6 of 6)

address	register name	default value	R/W	description
0x5057	ISP CTRL57	0x00	RW	Bit[7]: sram_test_dpc1 Bit[6]: sram_test_dpc2 Bit[5]: sram_test_dpc3 Bit[4]: sram_test_dpc4 Bit[3:0]: Not used
0x5058	ISP CTRL58	0xAA	RW	Bit[7:4]: sram_rm_dpc1 Bit[3:0]: sram_rm_dpc2
0x5059	ISP CTRL59	0xAA	RW	Bit[7:4]: sram_rm_dpc3 Bit[3:0]: sram_rm_dpc4

table 7-16 AWB registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5180	AWB CTRL	0x00	RW	Bit[7]: hsize_man_en Bit[6]: fast_awb 0: Disable fast AWB calculation function 1: Enable fast AWB calculation function Bit[5]: freeze_gain_en When it is enabled, the output AWB gains are input AWB gains Bit[4]: freeze_sum_en When it is set, the sums and averages value are the same as previous frame Bit[3]: gain_man_en 0: Output calculated gains 1: Output manual gains set by registers Bit[2]: start_sel 0: Select the last href falling edge of before gain input as cal start signal 1: Select the last href falling edge of after gain input as cal start signal Bit[1]: after_gma Bit[0]: Not used

table 7-16 AWB registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x5181	AWB DELTA	0x20	RW	Bit[7]: delta_opt Bit[6]: base_man_en Bit[5:0]: awb_delta Delta value to increase or decrease the gains
0x5182	STABLE RANGE	0x04	RW	Bit[7:0]: stable_range
0x5183	STABLE RANGEW	0x08	RW	Bit[7:0]: stable_rangew Wide stable range
0x5184	HSIZE_MAN	0x01	RW	Bit[7:4]: Not used Bit[3:0]: hsize_man[11:8]
0x5185	HSIZE_MAN	0xE0	RW	Bit[7:0]: hsize_man[7:0]
0x5186	MANUAL RED GAIN MSB	0x04	RW	Bit[7:4]: Not used Bit[3:0]: red_gain_man[11:8]
0x5187	MANUAL RED GAIN LSB	0x00	RW	Bit[7:0]: red_gain_man[7:0]
0x5188	MANUAL GREEN GAIN MSB	0x04	RW	Bit[7:4]: Not used Bit[3:0]: grn_gain_man[11:8]
0x5189	MANUAL GREEN GAIN LSB	0x00	RW	Bit[7:0]: grn_gain_man[7:0]
0x518A	MANUAL BLUE GAIN MSB	0x04	RW	Bit[7:4]: Not used Bit[3:0]: blu_gain_man[11:8]
0x518B	MANUAL BLUE GAIN LSB	0x00	RW	Bit[7:0]: blu_gain_man[7:0]
0x518C	RED GAIN LIMIT	0xF0	RW	Bit[7:4]: red_gain_up_limit Bit[3:0]: red_gain_dn_limit They are only the highest 4 bits of limitation. Maximum red gain is {red_gan_up_limit,FF} Minimum red gain is {red_gain_dn_limit,00}
0x518D	GREEN GAIN LIMIT	0xF0	RW	Bit[7:4]: green_gain_up_limit Bit[3:0]: green_gain_dn_limit They are only the highest 4 bits of limitation. Maximum green gain is {green_gan_up_limit,FF} Minimum green gain is {green_gain_dn_limit,00}

table 7-16 AWB registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x518E	BLUE GAIN LIMIT	0xF0	RW	Bit[7:4]: blue_gain_up_limit Bit[3:0]: blue_gain_dn_limit They are only the highest 4 bits of limitation. Maximum blue gain is {blue_gan_up_limit,FF} Minimum blue gain is {blue_gain_dn_limit,00}
0x518F	FRAME CNT	0x00	RW	Bit[7:4]: Not used Bit[3:0]: awb_frame_cnt
0x51DF	BASE MAN	0x10	RW	Bit[7:0]: base_man

table 7-17 average registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5680	X START	0x00	RW	Bit[7:5]: Not used Bit[4:0]: x_start[11:8] Horizontal start position for average window high byte
0x5681	X START	0x00	RW	Bit[7:0]: x_start[7:0] Horizontal start position for average window low byte
0x5682	Y START	0x00	RW	Bit[7:4]: Not used Bit[3:0]: y_start[10:8] Vertical start position for average window low byte
0x5683	Y START	0x00	RW	Bit[7:0]: y_start[7:0] Vertical start position for average window low byte
0x5684	X WINDOW	0x0A	RW	Bit[7:5]: Not used Bit[4:0]: Window X in manual average window mode high byte
0x5685	X WINDOW	0x20	RW	Bit[7:0]: Window X in manual average window mode low byte
0x5686	Y WINDOW	0x07	RW	Bit[7:4]: Not used Bit[3:0]: Window Y in manual average window mode high byte
0x5687	Y WINDOW	0x98	RW	Bit[7:0]: Window Y low byte in manual average window mode
0x5688	WEIGHT00	0x11	RW	Bit[7:4]: window1_weight Bit[3:0]: window0_weight

table 7-17 average registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5689	WEIGHT01	0x11	RW	Bit[7:4]: window3_weight Bit[3:0]: window2_weight
0x568A	WEIGHT02	0x11	RW	Bit[7:4]: window5_weight Bit[3:0]: window4_weight
0x568B	WEIGHT03	0x11	RW	Bit[7:4]: window7_weight Bit[3:0]: window6_weight
0x568C	WEIGHT04	0x11	RW	Bit[7:4]: window9_weight Bit[3:0]: window8_weight
0x568D	WEIGHT05	0x11	RW	Bit[7:4]: window11_weight Bit[3:0]: window10_weight
0x568E	WEIGHT06	0x11	RW	Bit[7:4]: window13_weight Bit[3:0]: window12_weight
0x568F	WEIGHT07	0x11	RW	Bit[7:4]: window15_weight Bit[3:0]: window14_weight
0x5690	AVG CTRL10	0x02	R	Bit[7:2]: Not used Bit[1]: avg_opt Bit[0]: avg_man 0: Auto average window 1: Manual average window
0x5691	AVG WEIGHT SUM	–	R	avg_wt_sum_o
0x5692	DEBUG MODE	–	–	Debug Mode
0x5693	AVG READOUT	–	R	Bit[7:0]: AVG value

table 7-18 DPC registers

address	register name	default value	R/W	description
0x5780~ 0x5791	DPC CTRL	–	RW	Debug Control Changing these registers is not recommended

table 7-19 LENC registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x5800	GMTRX00	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_00
0x5801	GMTRX01	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_01
0x5802	GMTRX02	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_02
0x5803	GMTRX03	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_03
0x5804	GMTRX04	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_04
0x5805	GMTRX05	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_05
0x5806	GMTRX10	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_06
0x5807	GMTRX11	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_07
0x5808	GMTRX12	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_08
0x5809	GMTRX13	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_09
0x580A	GMTRX14	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_0a
0x580B	GMTRX15	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_0b
0x580C	GMTRX20	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_0c
0x580D	GMTRX21	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_0d
0x580E	GMTRX22	0x00	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_0e
0x580F	GMTRX23	0x00	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_0f
0x5810	GMTRX24	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_10
0x5811	GMTRX25	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_11

table 7-19 LENC registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x5812	GMTRX30	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_12
0x5813	GMTRX31	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_13
0x5814	GMTRX32	0x00	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_14
0x5815	GMTRX33	0x00	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_15
0x5816	GMTRX34	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_16
0x5817	GMTRX35	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_17
0x5818	GMTRX40	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_18
0x5819	GMTRX41	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_19
0x581A	GMTRX42	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_1a
0x581B	GMTRX43	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_1b
0x581C	GMTRX44	0x08	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_1c
0x581D	GMTRX45	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_1d
0x581E	GMTRX50	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_1e
0x581F	GMTRX51	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_1f
0x5820	GMTRX52	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_20
0x5821	GMTRX53	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_21
0x5822	GMTRX54	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_22
0x5823	GMTRX55	0x10	RW	Bit[7:6]: Not used Bit[5:0]: green_matrix_23
0x5824	BRMATRX00	0xAA	RW	Bit[7:4]: blue_matrix_00 Bit[3:0]: red_matrix_00

table 7-19 LENC registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x5825	BRMATRX01	0xAA	RW	Bit[7:4]: blue_matrix_01 Bit[3:0]: red_matrix_01
0x5826	BRMATRX02	0xAA	RW	Bit[7:4]: blue_matrix_02 Bit[3:0]: red_matrix_02
0x5827	BRMATRX03	0xAA	RW	Bit[7:4]: blue_matrix_03 Bit[3:0]: red_matrix_03
0x5828	BRMATRX04	0xAA	RW	Bit[7:4]: blue_matrix_04 Bit[3:0]: red_matrix_04
0x5829	BRMATRX05	0xAA	RW	Bit[7:4]: blue_matrix_05 Bit[3:0]: red_matrix_05
0x582A	BRMATRX06	0x99	RW	Bit[7:4]: blue_matrix_06 Bit[3:0]: red_matrix_06
0x582B	BRMATRX07	0x99	RW	Bit[7:4]: blue_matrix_07 Bit[3:0]: red_matrix_07
0x582C	BRMATRX08	0x99	RW	Bit[7:4]: blue_matrix_08 Bit[3:0]: red_matrix_08
0x582D	BRMATRX09	0xAA	RW	Bit[7:4]: blue_matrix_09 Bit[3:0]: red_matrix_09
0x582E	BRMATRX20	0xAA	RW	Bit[7:4]: blue_matrix_20 Bit[3:0]: red_matrix_20
0x582F	BRMATRX21	0x99	RW	Bit[7:4]: blue_matrix_21 Bit[3:0]: red_matrix_21
0x5830	BRMATRX22	0x88	RW	Bit[7:4]: blue_matrix_22 Bit[3:0]: red_matrix_22
0x5831	BRMATRX23	0x99	RW	Bit[7:4]: blue_matrix_23 Bit[3:0]: red_matrix_23
0x5832	BRMATRX24	0xAA	RW	Bit[7:4]: blue_matrix_24 Bit[3:0]: red_matrix_24
0x5833	BRMATRX30	0xAA	RW	Bit[7:4]: blue_matrix_30 Bit[3:0]: red_matrix_30
0x5834	BRMATRX31	0x99	RW	Bit[7:4]: blue_matrix_31 Bit[3:0]: red_matrix_31
0x5835	BRMATRX32	0x99	RW	Bit[7:4]: blue_matrix_32 Bit[3:0]: red_matrix_32
0x5836	BRMATRX33	0x99	RW	Bit[7:4]: blue_matrix_33 Bit[3:0]: red_matrix_33
0x5837	BRMATRX34	0xAA	RW	Bit[7:4]: blue_matrix_34 Bit[3:0]: red_matrix_34

table 7-19 LENC registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x5838	BRMATRX40	0xAA	RW	Bit[7:4]: blue_matrix_40 Bit[3:0]: red_matrix_40
0x5839	BRMATRX41	0xAA	RW	Bit[7:4]: blue_matrix_41 Bit[3:0]: red_matrix_41
0x583A	BRMATRX42	0xAA	RW	Bit[7:4]: blue_matrix_42 Bit[3:0]: red_matrix_42
0x583B	BRMATRX43	0xAA	RW	Bit[7:4]: blue_matrix_43 Bit[3:0]: red_matrix_43
0x583C	BRMATRX44	0xAA	RW	Bit[7:4]: blue_matrix_44 Bit[3:0]: red_matrix_44

table 7-20 cluster DPC registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5900	OTP START ADDR	0x10	RW	Bit[7:6]: Not used Bit[5:0]: otp_start_addr
0x5901	OTP END ADDR	0x1F	RW	Bit[7:6]: Not used Bit[5:0]: otp_end_addr
0x5902	OTP CTRL02	0x00	RW	Bit[7:5]: Not used Bit[4]: man_inc_en Bit[3]: disable_mf Bit[2]: disable_offset Bit[1]: mirror_opt Bit[0]: disable_bin
0x5903	OTP CTRL03	0x6F	RW	Bit[7]: Not used Bit[6:5]: recov_method Bit[4]: fixed_replace Bit[3]: fixed_ptn Bit[2]: flip_opt Bit[1]: expo_en Bit[0]: gain_en
0x5904	EXPO CONS	0x00	RW	Bit[7]: Not used Bit[6:0]: otp_expo_constrain
0x5905	EXPO CONS	0x00	RW	Bit[7:0]: otp_expo_constrain
0x5906	GAIN CONS	0x07	RW	Bit[7:6]: Not used Bit[5:0]: otp_expo_constrain

table 7-20 cluster DPC registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5907	OTP CTRL07	0x38	RW	Bit[7]: Not used Bit[6:4]: remain_bit Bit[3:0]: Threshold
0x5908	OTP MAN X EVEN INC	0x01	RW	Bit[7:4]: Not used Bit[3:0]: otp_man_x_even_inc
0x5909	OTP MAN X ODD INC	0x01	RW	Bit[7:4]: Not used Bit[3:0]: otp_man_x_odd_inc
0x590A	OTP MAN Y EVEN INC	0x01	RW	Bit[7:4]: Not used Bit[3:0]: otp_man_y_even_inc
0x590B	OTP MAN Y ODD INC	0x01	RW	Bit[7:4]: Not used Bit[3:0]: otp_man_y_odd_inc
0x590C~ 0x590D	DEBUG MODE	–	–	Not Used

table 7-21 windows registers

address	register name	default value	R/W	description
0x5980	WINDOW XSTART	0x00	RW	Bit[7:5]: Not used Bit[4:0]: window_xstart[12:8]
0x5981	WINDOW XSTART	0x00	RW	Bit[7:0]: window_xstart[7:0]
0x5982	WINDOW YSTART	0x00	RW	Bit[7:4]: Not used Bit[3:0]: window_ystart[11:8]
0x5983	WINDOW YSTART	0x00	RW	Bit[7:0]: window_ystart[7:0]
0x5984	WIN X WIN	0x10	RW	Bit[7:5]: Not used Bit[4:0]: window_x_win[12:8]
0x5985	WIN X WIN	0xA0	RW	Bit[7:0]: window_x_win[7:0]
0x5986	WIN Y WIN	0x0C	RW	Bit[7:4]: Not used Bit[3:0]: window_y_win[11:8]
0x5987	WIN Y WIN	0x78	RW	Bit[7:0]: window_y_win[7:0]
0x5988	WIN MAN	0x00	RW	Bit[7:1]: Not used Bit[0]: Window manual enable 0: Auto mode 1: Manual mode

table 7-22 AEC/AGC 3 registers

address	register name	default value	R/W	description
0x5A00	DIGC CTRL0	0x00	RW	Bit[7:3]: Not used Bit[2]: dig_comp_bypass Bit[1]: man_opt Bit[0]: man_en
0x5A02	DIG COMP MAN	0x02	RW	Bit[7:2]: Not used Bit[1:0]: dig_comp_man[9:8]
0x5A03	DIG COMP MAN	0x00	RW	Bit[7:0]: dig_comp_man[7:0]
0x5A20	SNR GAIN MAN	0x00	RW	Bit[7:1]: Not used Bit[0]: gainc_sg_man[8]
0x5:A21	SNR GAIN MAN	0x00	RW	Bit[7:0]: gainc_sg_man[7:0]
0x5A22	DIG GAIN MAN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: gainc_dg_man[9:8]
0x5A23	DIG GAIN MAN	0x00	RW	Bit[7:0]: gainc_dg_man[7:0]
0x5A24	GAINC CTRL0	0x00	RW	Bit[7:3]: Not used Bit[2]: OPT Bit[1]: bypass_opt Bit[0]: gainc_man_en
0x5A25	GAINC DG RDOUT	–	R	Bit[7:2]: Not used Bit[1:0]: gainc_dig_comp[9:8]
0x5A26	GAINC DG RDOUT	–	R	Bit[7:0]: gainc_dig_comp[7:0]
0x5A27	GAINC SG RDOUT	–	R	Bit[7:1]: Not used Bit[0]: gainc_snr[8]
0x5A28	GAINC SG RDOUT	–	R	Bit[7:0]: gainc_snr[7:0]
0x5A29	GAINC SG RDOUT	–	R	Bit[7:2]: Not used Bit[1:0]: gainc_realgain[9:8]
0x5A2A	GAINC SG RDOUT	–	R	Bit[7:0]: gainc_realgain[7:0]
0x5A40	GAINF ANA NUM	0x07	RW	Bit[7:0]: gainf_ana_bit_num
0x5A41	GAINF DIG GAIN	0x00	RW	Bit[7:0]: gainf_dig_gain

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 +125°C
supply voltage (with respect to ground)	V_{DD-A}	4.5V
	V_{DD-D}	3V
	V_{DD-IO}	4.5V
electro-static discharge (ESD)	human body model	2000V
	machine model	200V
all input/output voltages (with respect to ground)		-0.3V to V_{DD-IO} + 1V
I/O current on any input or output pin		±200 mA

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

8.2 functional temperature

table 8-2 functional temperature

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

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_A < 70°C)

symbol	parameter	min	typ	max	unit
supply					
V_{DD-A}	supply voltage (analog)	2.6	2.8	3.0	V
V_{DD-DO}	supply voltage (digital I/O)	1.7	1.8	3.0	V
V_{DD-D}	supply voltage (digital core)[a]	1.425	1.5	1.575	V
V_{DD-E}	supply voltage (MIPI)	1.425	1.5	1.575	V
I_{DD-A}	active (operating) current 2592 x 1944 @ 15 fps[b]		TBD	TBD	mA
I_{DD-DO}			TBD	TBD	mA
I_{DD-A}	active (operating) current 720p @ 30fps		TBD	TBD	mA
I_{DD-DO}			TBD	TBD	mA
I_{DD-A}	active (operating) current 720p @ 60fps		TBD	TBD	mA
I_{DD-DO}			TBD	TBD	mA
I_{DD-A}	active (operating) current VGA @ 30fps		TBD	TBD	mA
I_{DD-DO}			TBD	TBD	mA
I_{DD-A}	active (operating) current VGA @ 60fps		TBD	TBD	mA
I_{DD-DO}			TBD	TBD	mA
$I_{DDS-SCCB}$[c]	standby current		TBD	TBD	µA
$I_{DDS-PWDN}$			TBD	TBD	µA
digital inputs (typical conditions: AVDD = 2.8V, DVDD = 1.5V, DOVDD = 1.8V)					
V_{IL}	input voltage LOW			0.54	V
V_{IH}	input voltage HIGH	1.26			V
C_{IN}	input capacitor			10	pF
digital outputs (standard loading 25 pF)					
V_{OH}	output voltage HIGH	1.62			V
V_{OL}	output voltage LOW			0.18	V
serial interface inputs					
V_{IL}[d]	SCL and SDA	-0.5	0	0.54	V
V_{IH}[d]	SCL and SDA	1.26	1.8	2.3	V

a. when internal regulator is bypassed

b. using internal regulator for DVDD and short DVDD with EVDD; DOVDD = 2.8V. The currents are for DVP output. MIPI output will results 5%-10% lower active current on I_{DD-DO}

c. external clock is stopped during measurement

d. based on DOVDD = 1.8V

8.4 AC characteristics

table 8-4 AC characteristics (T_A = 25°C, V_{DD-A} = 2.8V)

symbol	parameter	min	typ	max	unit
ADC parameters					
B	analog bandwidth		48		MHz
DLE	DC differential linearity error		0.5		LSB
ILE	DC integral linearity error		1		LSB
	settling time for hardware reset			<1	ms
	settling time for software reset			<1	ms
	settling time for resolution mode change			<1	ms
	settling time for register setting			<300	ms

table 8-5 timing characteristics

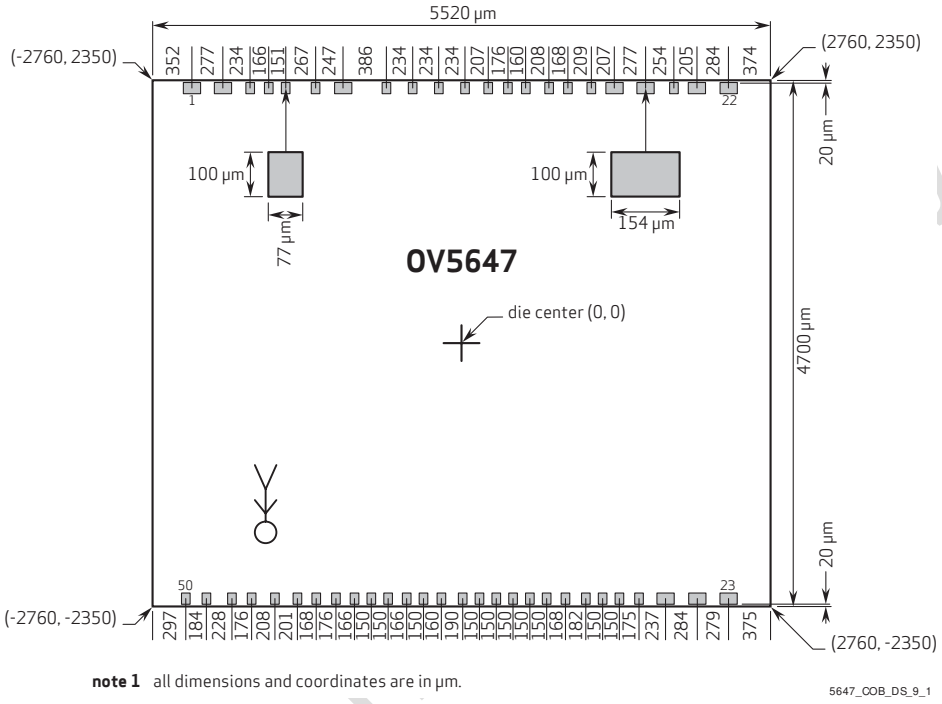
symbol	parameter	min	typ	max	unit
oscillator and clock input					
f_{OSC}	frequency (XCLK)	6	24	27	MHz
t_r, t_f	clock input rise/fall time			5 (10[a])	ns

a. if using the internal PLL

9 mechanical specifications

9.1 physical specifications

figure 9-1 die specifications



note 1 all dimensions and coordinates are in µm.

5647_COB_DS_9_1

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

pad number	pad name	x coordinate	y coordinate	x pitch	y pitch	pad size
1	AVDD	-2408	2280			154x100
2	AGND	-2131	2280	277	0	154x100
3	DOGND	-1888	2280	243	0	77x100
4	SCL	-1722	2280	166	0	77x100
5	SDA	-1571	2280	151	0	77x100
6	DVDD	-1304	2280	267	0	77x100
7	SGND	-1057	2280	247	0	154x100
8	GPIO1	-671	2280	386	0	77x100

table 9-1 pad location coordinates (sheet 2 of 3)

pad number	pad name	x coordinate	y coordinate	x pitch	y pitch	pad size
9	GPIO0	-437	2280	234	0	77x100
10	STROBE	-203	2280	234	0	77x100
11	FREX	31	2280	234	0	77x100
12	DOVDD	238	2280	207	0	77x100
13	VREF2	414	2280	176	0	77x100
14	VREF1	574	2280	160	0	77x100
15	PWDN	782	2280	208	0	77x100
16	DVDD	950	2280	168	0	77x100
17	RESETB	1159	2280	209	0	77x100
18	AVDD	1366	2280	207	0	154x100
19	AGND	1643	2280	277	0	154x100
20	TM	1897	2280	254	0	77x100
21	DOGND	2102	2280	205	0	154x100
22	DVDD	2386	2280	284	0	154x100
23	DVDD	2385	-2280	-1	-4560	154x100
24	DOVDD	2106	-2280	-279	0	154x100
25	DOGND	1822	-2280	-284	0	154x100
26	AVDD	1585	-2280	-237	0	77x100
27	HREF	1410	-2280	-175	0	77x100
28	PCLK	1260	-2280	-150	0	77x100
29	VSYNC	1110	-2280	-150	0	77x100
30	DOVDD	928	-2280	-182	0	77x100
31	D0	760	-2280	-168	0	77x100
32	D1	610	-2280	-150	0	77x100
33	D2	460	-2280	-150	0	77x100
34	D3	310	-2280	-150	0	77x100
35	D9/MDN0	160	-2280	-150	0	77x100
36	D8/MDP0	10	-2280	-150	0	77x100
37	EVDD	-180	-2280	-190	0	77x100
38	D7/MCN	-340	-2280	-160	0	77x100

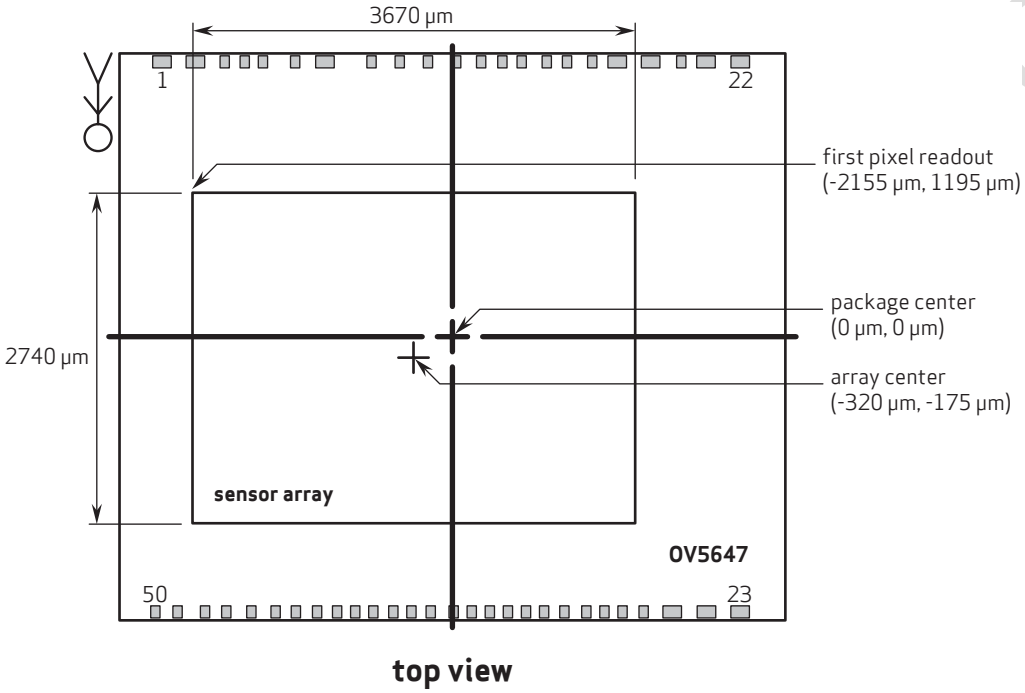
table 9-1 pad location coordinates (sheet 3 of 3)

pad number	pad name	x coordinate	y coordinate	x pitch	y pitch	pad size
39	D6/MCP	-490	-2280	-150	0	77x100
40	EGND	-656	-2280	-166	0	77x100
41	D5/MDN1	-806	-2280	-150	0	77x100
42	D4/MDP1	-956	-2280	-150	0	77x100
43	EGND	-1122	-2280	-166	0	77x100
44	PVDD	-1298	-2280	-176	0	77x100
45	XCLK	-1466	-2280	-168	0	77x100
46	DOVDD	-1667	-2280	-201	0	77x100
47	DVDD	-1875	-2280	-208	0	77x100
48	DOGND	-2051	-2280	-176	0	77x100
49	AVDD	-2279	-2280	-228	0	77x100
50	AGND	-2463	-2280	-184	0	77x100

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.

5647_COB_DS_10_1

10.2 lens chief ray angle (CRA)

figure 10-2 chief ray angle (CRA)

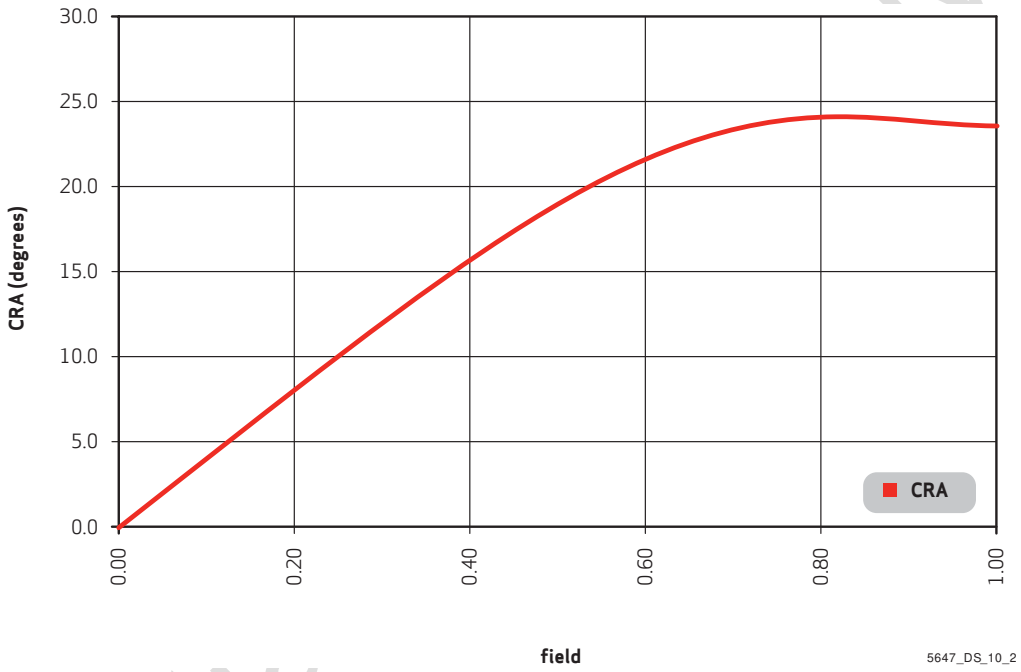


table 10-1 CRA versus image height plot (sheet 1 of 2)

field (%)	image height (mm)	CRA (degrees)
0.00	0.000	0.0
0.05	0.114	2.0
0.10	0.227	4.1
0.15	0.341	6.1
0.20	0.454	8.1
0.25	0.568	10.1
0.30	0.681	12.0
0.35	0.795	13.8
0.40	0.908	15.6
0.45	1.022	17.3
0.50	1.135	18.9

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

field (%)	image height (mm)	CRA (degrees)
0.55	1.249	20.4
0.60	1.362	21.6
0.65	1.476	22.6
0.70	1.589	23.4
0.75	1.703	23.9
0.80	1.816	24.1
0.85	1.930	24.1
0.90	2.043	23.9
0.95	2.157	23.7
1.00	2.270	23.6

revision history

version 1.0 11.03.2009

- initial release

the clear advantage™

FINLAND

Mouhijärvi + 358 3 341 1898

GERMANY

Munich +49 89 63 81 99 88

CHINA

Beijing + 86 10 6580 1690

Shanghai + 86 21 6105 5100

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**