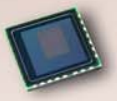
OmniVision



datasheet

PRELIMINARY SPECIFICATION

1/5" color CMOS UXGA (1600 x 1200) image sensor
with OmniPixel3-HS™ technology

OV2685

color CMOS UXGA (1600 x 1200) image sensor with OmniPixel3-HS™ technology

datasheet (CSP5)
PRELIMINARY SPECIFICATION

version 1.0
august 2013

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

applications

- cellular and picture phones
- home entertainment
- PC multimedia
- toys

ordering information

- **OV02685-H53A** (color, lead-free)
  53-pin CSP5

features

- 1.75 µm x 1.75 µm pixel with OmniPixel-3HS™ technology
- optical size of 1/5"
- 28.0° CRA
- supports images sizes: UXGA (1600H x 1200V), 1600 HD+ (1600H x 900V), SXGA(1280H x 960V), 720p (1280H x 720V), and more
- support for output formats: 10-bit RGB RAW, 8-bit YUV
- 2MP at 30 fps
- programmable controls for frame rate, mirror and flip, cropping, and windowing
- two-wire serial bus control (SCCB)
- MIPI serial output interface (2-lane MIPI)
- automatic image control functions: automatic exposure control (AEC), automatic gain control (AGC), auto white balance (AWB), and image de-noise
- on-chip phase lock loops (PLLs)
- image quality control: defect pixel correction, saturation, hue, gamma, lens correction, and automatic black level calibration
- suitable for module size of TBD

key specifications (typical)

- **active array size:** 1616 x 1216
- **power supply:**
  core: 1.7 ~ 1.9V
  analog: 2.6 ~ 3.0V
  I/O: 1.7 ~ 3.0V
- **power requirements:**
  active: TBD
  XSHUTDN: < 10 µA
- **temperature range:**
  operating: -30°C to +85°C junction temperature
  stable image: 0°C to +50°C junction temperature
- **output interfaces:** two lane MIPI serial output
- **output formats:** 10-bit RGB RAW, 8-bit YUV
- **lens size:** 1/5"
- **lens chief ray angle:** 28.0° non-linear (see figure 8-2)
- **input clock frequency:** 6~27 MHz
- **max S/N ratio:** TBD
- **dynamic range:** TBD
- **maximum image transfer rate:** 30 fps
- **sensitivity:** TBD
- **scan mode:** progressive
- **maximum exposure interval:** 1 frame – 4 t_{ROW}
- **pixel size:** 1.75 µm x 1.75 µm
- **dark current:** TBD
- **image area:** 2840 µm x 2150 µm
- **package dimensions:** 4454 µm x 4014 µm

table of contents

list of figures

list of tables

1 signal descriptions

table 1-1 lists the signal descriptions and their corresponding pinnumbers for the OV2685 image sensor. The package information is shown in section 7.

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

pin number	signal name	pin type	description
A1	NC	–	no connect
A2	AVDD	power	analog power
A3	DOVDD	power	I/O power
A4	GPIO	I/O	general purpose I/O
A5	VSYNC	I/O	video output vertical signal
A6	XSHUTDOWN	input	reset and power down (active low with pull down resister)
A7	DVDD	power	digital circuit power
A8	NC	–	no connect
B1	DOGND	ground	I/O ground
B2	AGND	ground	analog ground
B3	DVDD	power	digital circuit power
B4	SIOC	input	SCCB interface clock input
B5	DVDD	power	digital circuit power
B6	FSIN	I/O	frame sync
B7	DOVDD	power	I/O power
B8	AGND	ground	analog ground
C1	DVDD	power	digital circuit power
C2	DOGND	ground	I/O ground
C3	SIOD	I/O	SCCB interface data pin
C4	DOGND	ground	I/O ground
C5	SID	input	SCCB last bit ID input 0: SCCB ID address = 0x78 1: SCCB ID address = 0x20
C6	DOGND	ground	I/O ground
C7	DOGND	ground	I/O ground
C8	AVDD	power	analog power

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

pin number	signal name	pin type	description
D1	DVDD	power	digital circuit power
D2	DVDD	power	digital circuit power
D3	**MDN0**	output	MIPI data lane 0 negative output
D5	**MDP1**	output	MIPI data lane 1 positive output
D6	**MDN1**	output	MIPI data lane 1 negative output
D7	DOGND	ground	I/O ground
D8	NVDD	reference	analog reference
E1	DOGND	ground	I/O ground
E2	DVDD	power	digital circuit power
E3	**MDP0**	output	MIPI data lane 0 positive output
E4	**MCP**	output	MIPI clock lane positive output
E5	**MCN**	output	MIPI clock lane negative output
E6	DVDD	power	digital circuit power
E7	**XVCLK**	input	system clock input
E8	AGND	ground	analog ground
F1	DOGND	ground	I/O ground
F2	DVDD	power	digital circuit power
F3	DOGND	ground	I/O ground
F4	DOGND	ground	I/O ground
F5	DVDD	power	digital circuit power
F6	DOGND	ground	I/O ground
F7	DOGND	ground	I/O ground
F8	HVDD	reference	analog reference
G1	NC	–	no connect
G2	DOVDD	power	I/O power
G3	DOGND	ground	I/O ground
G4	DVDD	power	digital circuit power
G7	AVDD	power	analog power
G8	NC	–	no connect

table 1-2 pin configuration under various conditions

pin number	signal name	XSHUTDOWN[a]	after XSHUTDOWN release[b]	software standby[c]
A4	GPIO	high-z	input by default (configurable)	input by default (configurable)
A5	VSYNC	high-z	input by default (configurable)	input by default (configurable)
A6	XSHUTDOWN	input	input	input
B4	SIOC	high-z	input	input
B6	FSIN	high-z	input by default (configurable)	input by default (configurable)
C3	SIOD	high-z	input	input
C5	SID	high-z	input by default (configurable)	input by default (configurable)
D3	MDN0	high-z	zero	high by default (configurable)
D5	MDP1	high-z	zero	high by default (configurable)
D6	MDN1	high-z	zero	high by default (configurable)
E3	MDP0	high-z	zero	high by default (configurable)
E4	MCP	high-z	zero	high by default (configurable)
E5	MCN	high-z	zero	high by default (configurable)
E7	XVCLK	input	input	input

a. XSHUTDOWN = 0

b. XSHUTDOWN from 0 to 1

c. sensor set to sleep from streaming mode

figure 1-1 pin diagram

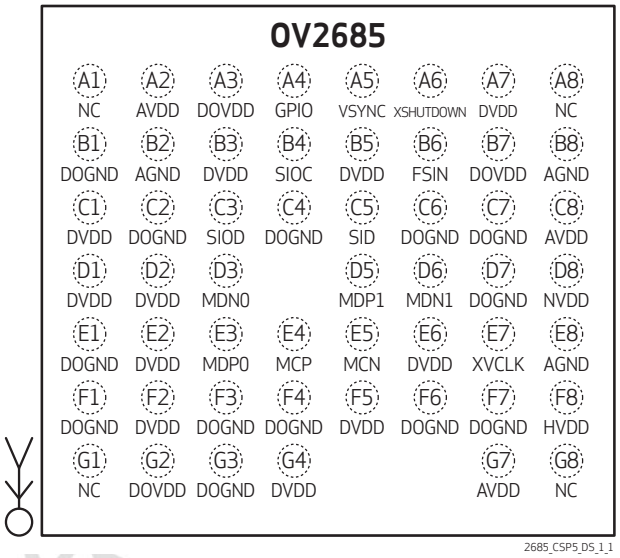


table 1-3 pad symbol and equivalent circuit (sheet 1 of 2)

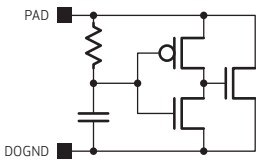
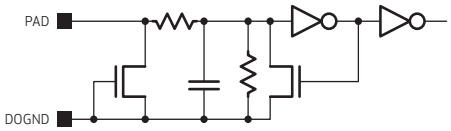
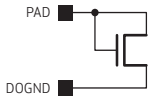
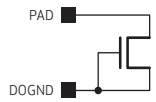
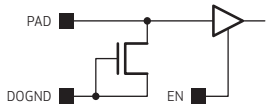
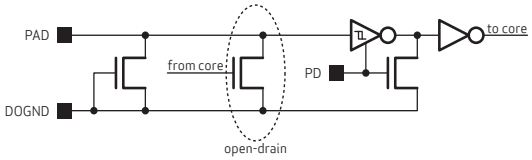
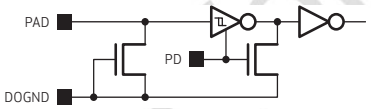
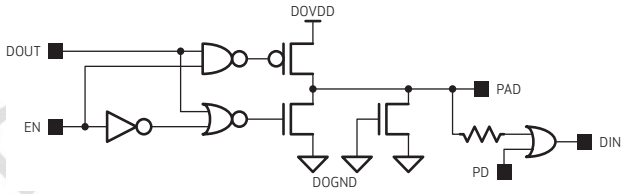
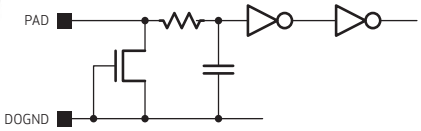
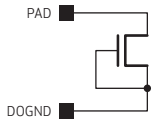
symbol	equivalent circuit
AVDD, DOVDD, DVDD	
XSHUTDOWN	
NVDD	
HVDD, AGND, DOGND	

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

symbol	equivalent circuit
XVCLK	
SIOD	
SIOC	
GPIO, FSIN, VSYNC	
SID	
MCN, MCP, MDN0, MDP0, MDN1, MDP1	

2 system level description

2.1 overview

The OV2685 SOC RGB image sensor is a high performance, 1/5-inch 2 megapixel CMOS image sensor that delivers 1600x1200 at 30 fps using OmniPixel3-HS™ pixel technology. It provides full-frame, sub-sampled, and windowed YUV MIPI images in various formats via the control of the Serial Camera Control Bus (SCCB) interface.

The low power OV2685 has a 2 megapixel image array capable of operating at up to 30 frames per second (fps) in YUV format with complete user control over image quality, formatting and output data transfer. Some image processing functions, such as defective pixel canceling, auto exposure/gain control, auto white balance, lens correction etc., are programmable through the SCCB interface.

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

The OV2685 has a 2-lane MIPI interface.

2.2 architecture

The OV2685 sensor core generates streaming pixel data at a constant frame rate to a pixel clock of 66 MHz. **figure 2-1** shows the functional block diagram of the OV2685 image sensor.

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

The exposure time is controlled by adjusting the time interval between precharging and sampling. After the data of the pixels in the row has been sampled, it is processed through an analog circuitry to correct the offset and multiply the data with corresponding gain. Following analog processing is the ADC which outputs 10-bit Raw data or 8-bit YUV422 for each pixel in the array.

figure 2-1 OV2685 block diagram

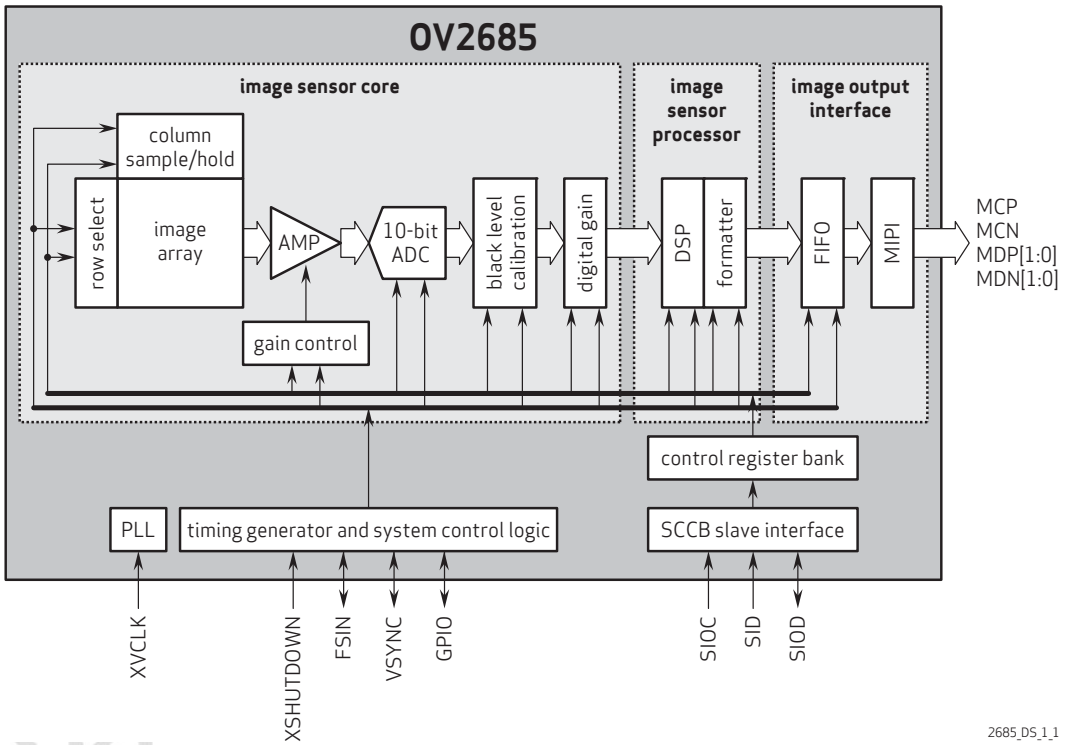
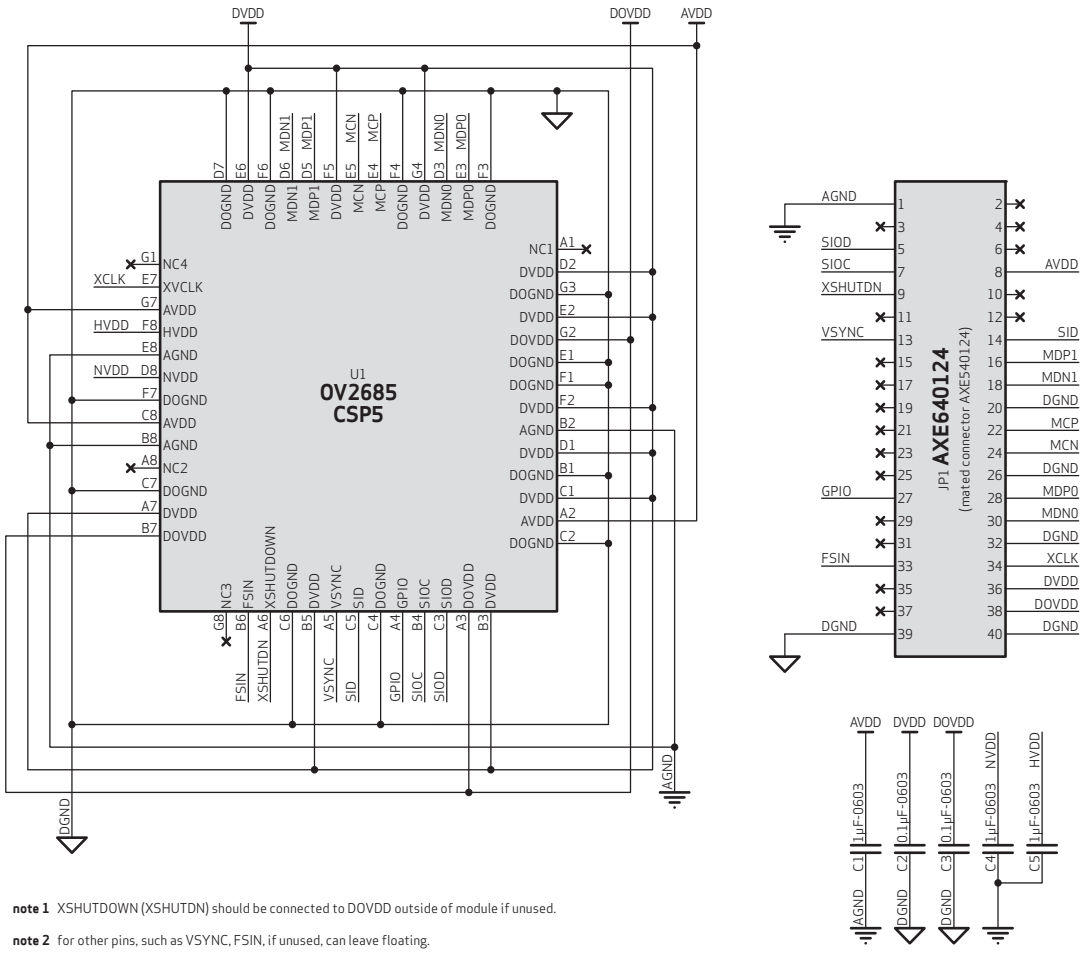


figure 2-2 OV2685 reference design schematic



note 1 XSHUTDOWN (XSHUTDN) should be connected to DOVDD outside of module if unused.

note 2 for other pins, such as VSYNC, FSIN, if unused, can leave floating.

note 3 AVDD is 2.8V of sensor analog power (clean).

note 4 DOVDD is 1.8/2.8V of sensor digital IO power (clean). 1.8V is recommended.

note 5 DVDD is 1.8V of sensor digital power.

note 6 sensor AGND and DGND should be separated and connected to a single point outside PCB (do not connect inside the module).

note 7 capacitors should be close to their related sensor pins.

note 8 If more space is available, use 1µF-0201 capacitor between DVDD and DGND.

note 9 traces of MCP, MCN, MDPx, and MDNx should have the same or similar length.
differential impedance of the clock pair and data pair transmission lines should be controlled at 100 Ohm.

note 10 SID pin should be pulled low for device address 0x78 and pulled high for device address 0x20.

note 11 all NC pins can be left floating or connected to GND if needed.

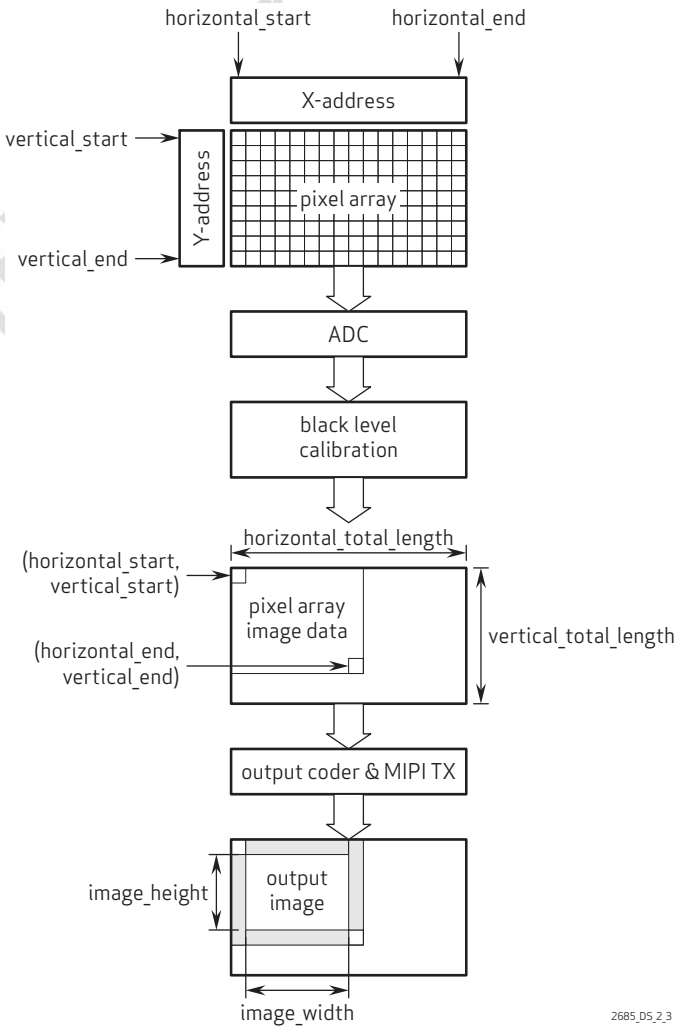
2.3 video timing overview

The sensor supports the following video timing functions.

- programmable image size
- image readout order modes – horizontal mirror and vertical flip
- variable line length and frame length

A host system can configure and control the OV2685 video timing through video timing register. For details of register control, please refer to the Register Description chapter (see **section 5**).

figure 2-3 video timing overview



2.4 pixel array addresses

The addressable pixel array of the OV2685 sensor is 1616 x 1216. The addressed region of the pixel array is controlled by the horizontal_start, vertical_start, horizontal_end and vertical_end registers. The start and end addresses are limited to even and odd numbers, respectively, to ensure that there is always an even number of pixels read out in x and y.

2.5 mirror and flip

The OV2685 provides mirror and flip readout modes, which respectively reverse the sensor data readout order horizontally and vertically (see **figure 2-4**).

figure 2-4 standard readout

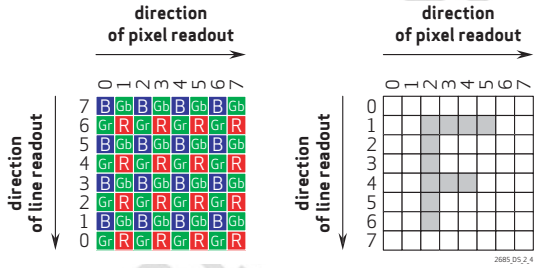


figure 2-5 horizontally mirrored readout

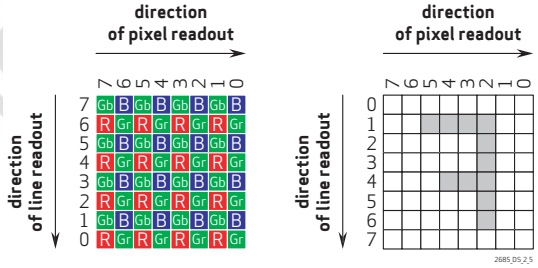


figure 2-6 vertically flipped readout

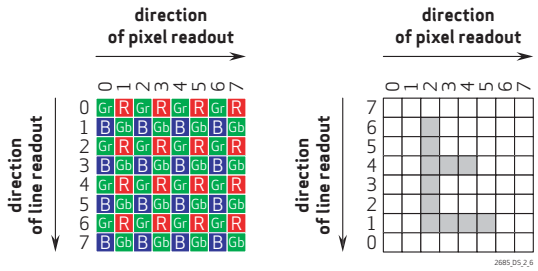


figure 2-7 horizontally mirrored vertically flipped readout

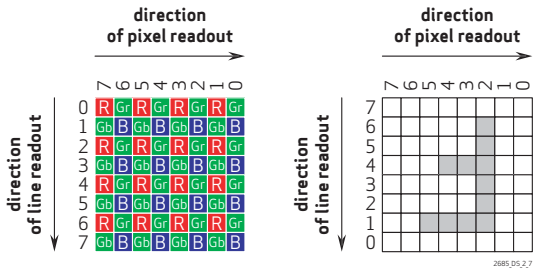


table 2-1 image orientation control registers

address	register name	default value	R/W	description
0x3820	FORMAT1	0x00	RW	Bit[2]: Flip ON/OFF select 0: Flip OFF 1: Flip ON
0x3821	FORMAT2	0x00	RW	Bit[2]: Mirror ON/OFF select 0: Mirror OFF 1: Mirror ON

2.6 format and frame rate control

The OV2685 supports frame rate control by variable line length and frame length control. It supports up to 30 frames per second in UXGA resolution. The OV2685 supports YUV422 and 10-bit RAW through a two-lane MIPI interface.

table 2-2 supported resolution and frame rate

format	resolution	max frame rate	methodology	pixel clock
UXGA	1600x1200	30 fps	full resolution	66 MHz
1600 HD+	1600x900	30 fps	full resolution (16:9) crop	66 MHz
SXGA	1280x960	30 fps	cropped 4:3	66 MHz
720p	1280x720	60 fps	cropped 16:9	66 MHz
quarter size	800x600	60 fps	2x2 binning/skip	66 MHz
VGA	640x480	60 fps	crop + 2x2 binning/skip	66 MHz

2.7 integration time control (electronic shutter control)

Throughout one image, all pixels of the OV2685 integrate light for exactly the same amount of time. This amount of time, the 'integration time', is defined using integer control parameters, integration_time_line. The integration_time_line parameter sets the number of complete sensor line periods of integration time.

table 2-3 integration time registers

address	register name	default value	R/W	description
0x3500~ 0x3502[7:4]	INTEGRATION TIME LINE	0x0020	RW	'Coarse' Integration Time in Unit of Line Period

2.8 black level calibration

The black level calibration can be applied before data is output. The black level calibration block subtracts the average signal level of optical black pixels to compensate for the dark current in the pixel output. The host can disable black level calibration.

2.9 PLL and clock generator

The OV2685 contains phase locked loop (PLL) blocks, which generate all of the necessary internal clocks from the external clock input pad. Contact your local OmniVision FAE for more details.

2.10 MIPI interface

The OV2685 supports a MIPI interface with a data transfer rate of up to 720Mbps for one-lane RAW or 576Mbps for two-lane YUV422. The OV2685 MIPI interface provides a single uni-directional clock lane and two uni-directional data lanes to communicate to components in a mobile device. The data lanes have full support for high speed (HS) data transfer mode. Contact your local OmniVision FAE for more details.

3 image sensor core digital functions

3.1 serial camera control bus (SCCB)

The host can access the registers through the SCCB interface to control the OV2685.

3.1.1 data transfer protocol

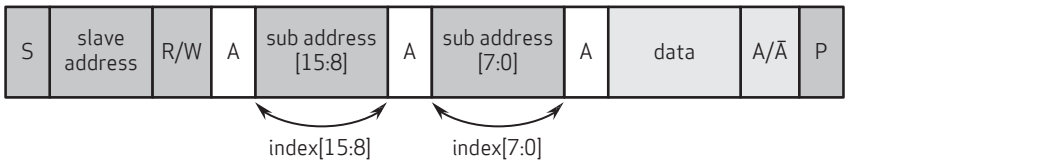
The data transfer of the OV2685 follows the SCCB protocol.

3.1.2 message format

The OV2685 supports the message format shown in **figure 3-1**. The 8-bit address of the OV2685 is 0x20 when SID pin is set to 1 or 0x78 when SID pin is set to 0. The repeated START (Sr) condition is not shown in **figure 3-2**, but is shown in **figure 3-3** and **figure 3-4**.

figure 3-1 message type

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



☐	from slave to master	S	START condition	A	acknowledge
☐	from master to slave	P	STOP condition	Ā	negative acknowledge
☐	direction depends on operation	Sr	repeated START condition		

2685_DS_3_1

3.1.3 read / write operation

The OV2685 supports four different read operations and two different write operations:

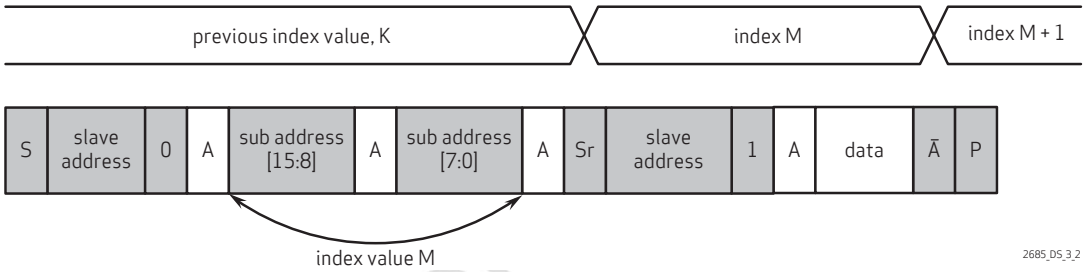
- a single read from random locations
- a sequential read from random locations
- a single read from current location
- a sequential read from current location
- single write to random locations
- sequential write starting from random location

The index in the sensor automatically increases by one after each read/write operation.

In a single read from random locations, the master does a dummy write operation to desired index, issues a repeated start condition and then addresses the camera again with a read operation. After acknowledging its slave address, the

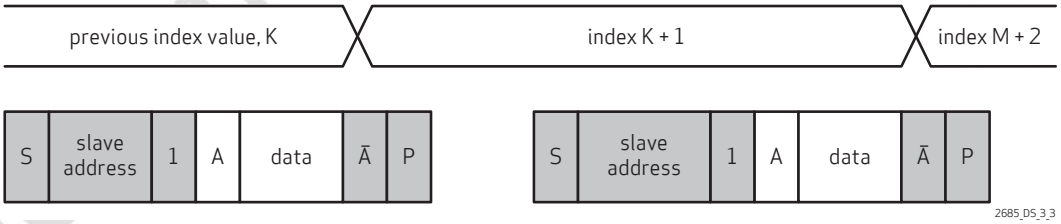
camera starts to output data onto the SIOD line as shown in **figure 3-2**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 3-2 SCCB single read from random location



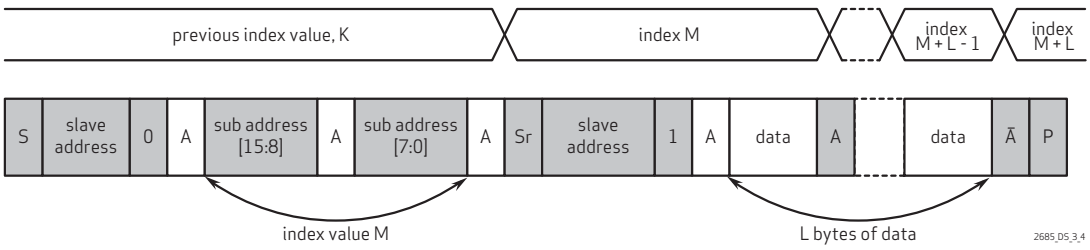
If the host addresses the camera with read operation directly without the dummy write operation, the camera responds by setting the data from last used index to the SIOD line as shown in **figure 3-3**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 3-3 SCCB single read from current location



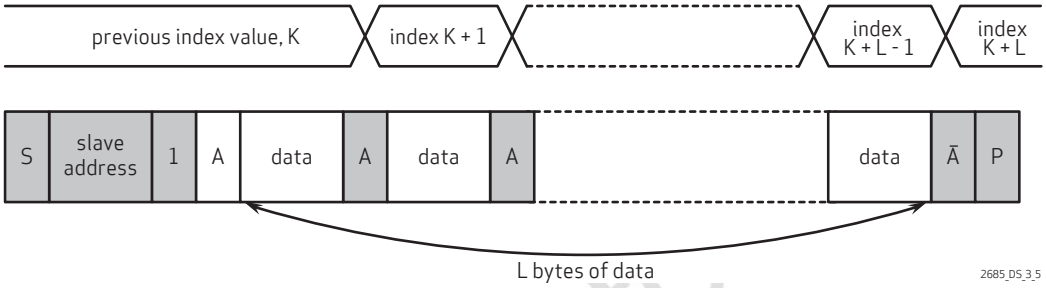
The sequential read from a random location is illustrated in **figure 3-4**. The master does a dummy write to the desired index, issues a repeated start condition after acknowledge from slave and addresses the slave again with read operation. If a master issues an acknowledge after receiving data, it acts as a signal to the slave that the read operation shall continue from the next index. When master has read the last data byte, it issues a negative acknowledge and stop condition.

figure 3-4 SCCB sequential read from random location



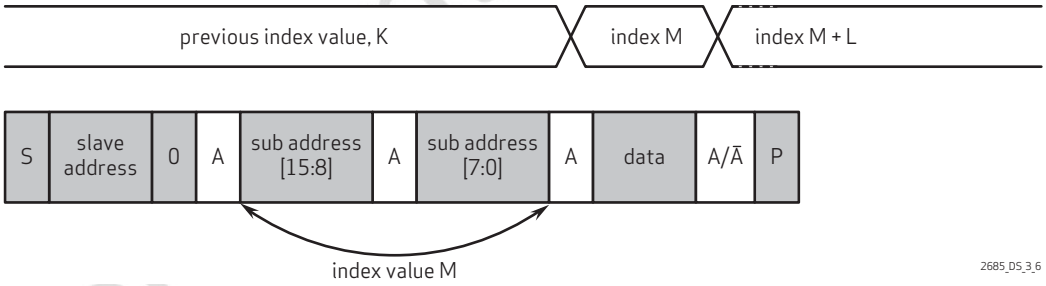
The sequential read from current location is similar to a sequential read from a random location. The only exception is that there is no dummy write operation. as shown in **figure 3-5**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 3-5 SCCB sequential read from current location



The write operation to a random location is illustrated in **figure 3-6**. The master issues a write operation to the slave, sets the index and data correspondingly after the slave has acknowledged. The write operation is terminated with a stop condition from the master.

figure 3-6 SCCB single write to random location



The sequential write is illustrated in **figure 3-7**. The slave automatically increments the index after each data byte. The sequential write operation is terminated with stop condition from the master.

figure 3-7 SCCB sequential write to random location

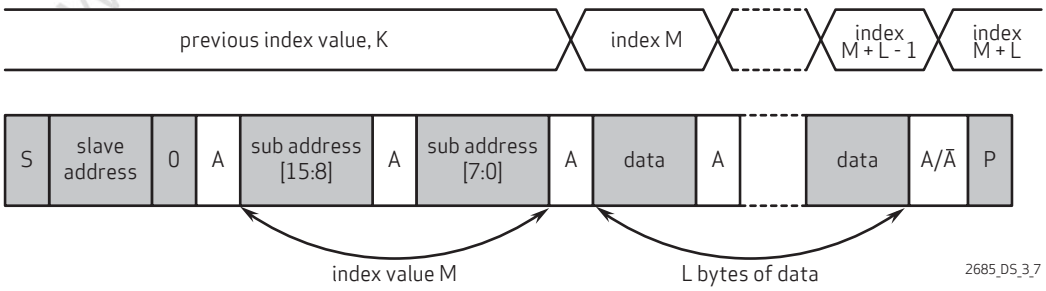


figure 3-8 SCCB interface timing

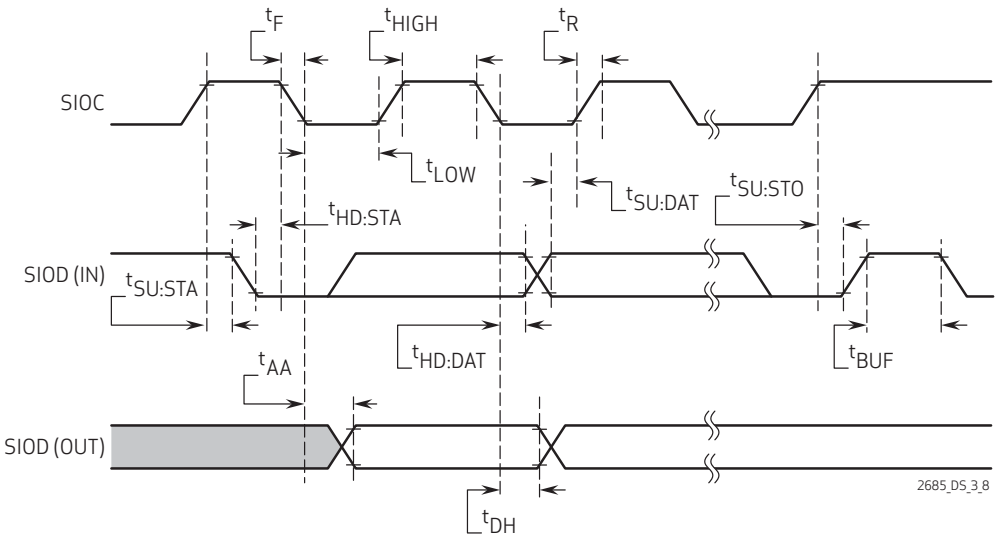


table 3-1 SCCB interface timing specifications[ab]

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

a. SCCB timing is based on 400kHz mode

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

3.2 exposure and gain control

The auto exposure control (AEC) and auto gain control (AGC) allows the OV2685 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 an external controller.

table 3-2 AEC/AGC functions

function	registers	description
AEC enable	0x3503	Bit[0]: AEC enable 0: auto 1: manual
AEC (exposure time)	0x3500~0x3502	manual exposure control[15:12] = 0x3500[3:0] manual exposure control[11:4] = 0x3501[7:0] manual exposure control[3:0] = 0x3502[7:4]
AGC enable	0x3503	Bit[1]: AGC enable 0: auto 1: manual
AGC (gain)	0x350A~0x350B	manual gain control[10:8] = 0x350A[2:0] manual gain control[7:0] = 0x350B[7:0]

The AEC controls image luminance using registers **WPT** (0x3A03) and **BPT** (0x3A04). The value of register **WPT** (0x3A03) indicates the high threshold value and the value of register **BPT** (0x3A04) indicates the low threshold value. When the target image luminance average value **YAVG** (0x3A19) is within the range specified by registers **WPT** (0x3A03) and **BPT** (0x3A04), the AEC keeps the image exposure. When register **YAVG** (0x3A19) is greater than the value in register **WPT** (0x3A03), the AEC will decrease the image exposure. When register **YAVG** (0x3A19) is less than the value in register **BPT** (0x3A04), the AEC will increase the image exposure. Accordingly, the value in register **WPT** (0x3A03) should be greater than the value in register **BPT** (0x3A04). The gap between the values of registers **WPT** (0x3A03) and **BPT** (0x3A04) controls the image stability.

table 3-3 AEC/AGC algorithms

function	register	description
WPT	0x3A03	upper limit of stable operating region in auto mode, AEC/AGC value decreases when average luminance/histogram is greater than WPT[7:0]
BPT	0x3A04	lower limit of stable operating region in auto mode, AEC/AGC value increases in auto mode when luminance average/histogram is less than BPT[7:0]
YAVG	0x3A19	luminance average - this register will auto update when average luminance is calculated from the B/Gb/Gr/R channel

3.3 general purpose input and output (GPIO)

The OV2685 has one general purpose I/O pin (GPIO) that can be configured to output logic high, output logic low, or high impedance (input mode) by writing to a register via SCCB. The default condition after initial power up is input mode.

table 3-4 GPIO and sensor working modes

working mode	GPIO
power OFF	high-z
power down (XSHUTDN = 0)	high-z
sleep (software standby, XSHUTDN = 1)	programmable (default input)
streaming	programmable

table 3-5 GPIO control functions

function	register	description
input/output control	0x3002	Bit[2]: Input/output control for GPIO 0: Input 1: Output
output logic level	0x300D	Bit[2]: Output value for GPIO
output select	0x3010	Bit[2]: Select general GPIO value through GPIO 0: HREF 1: Programmed logic level

3.4 group write

Group write is supported in order to update a group of registers at the same time. These registers are guaranteed to be written prior to the internal latch at the frame boundary (0x31xx registers will not support group write).

The OV2685 supports up to two groups. These groups share 128 bytes of memory and the size of each group is programmable by adjusting the start address.

table 3-6 group hold control

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

4 image sensor processor digital functions

4.1 ISP general controls

The ISP module provides image processor functions, creating the necessary control signals.

table 4-1 ISP top registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0x1F	RW	Bit[7]: lenc_en LENC functions enable signal 0: Disable 1: Enable Bit[6]: awb_gain_en AWB gain function enable signal 0: Disable 1: Enable Bit[5]: lcd_en LCD function enable signal 0: Disable 1: Enable Bit[4]: avg_en Average function enable signal 0: Disable 1: Enable Bit[3]: dgc_en Digital gain compensation enable 0: Disable 1: Enable Bit[2]: bc_en Black pixel cancellation function enable signal 0: Disable 1: Enable Bit[1]: wc_en White pixel cancellation function enable signal 0: Disable 1: Enable Bit[0]: blc_en BLC function enable signal 0: Disable 1: Enable
0x5003	ISP CTRL03	0x00	RW	Bit[7:0]: DGC and AWBG manual bias value

table 4-1 ISP top registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x5004	ISP CTRL04	0x18	RW	Bit[7]: uv_dns_en UV denoise functions enable signal 0: Disable 1: Enable Bit[6]: rgb_dns_en RGB denoise function enable signal 0: Disable 1: Enable Bit[5]: gamma_en Gamma function enable signal 0: Disable 1: Enable Bit[4]: cmx_en CMX function enable signal 0: Disable 1: Enable Bit[3]: cip_en CIP function enable signal 0: Disable 1: Enable Bit[2]: raw_dns_en RAW noise function enable signal 0: Disable 1: Enable Bit[1]: stretch_en Stretch function enable signal 0: Disable 1: Enable Bit[0]: awb_en AWB function enable signal 0: Disable 1: Enable

table 4-1 ISP top registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x5005	ISP CTRL05	0x12	RW	Bit[4]: sde_en SDE functions enable signal 0: Disable 1: Enable Bit[3]: lcd_out_en Output RAW format from LCD module 0: Disable 1: Enable Bit[2]: raw_en RAW output mode enable signal 0: Disable 1: Enable Bit[1]: rgb2yuv_en RGB to YUV function enable signal 0: Disable 1: Enable Bit[0]: binc_en Binning function enable signal 0: Disable 1: Enable

4.2 lens correction (LENC)

The main purpose of the LENC is to compensate for lens imperfection. According to distance of each pixel to the lens center, LENC applies the gain to each pixel to compensate to lens fall off. The LENC correcting gain is also adaptive to sensor gain. Also, the LENC supports the subsample function in both horizontal and vertical directions. Contact your local OmniVision FAE for lens correction settings.

table 4-2 LENC registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5800	RED X0	0x03	RW	Bit[7:3]: Not used Bit[2:0]: red_x0[10:8] Red x0 high byte
0x5801	RED X0	0x28	RW	Bit[7:0]: red_x0[7:0] Red x0 low byte
0x5802	RED Y0	0x02	RW	Bit[7:3]: Not used Bit[2:0]: red_y0[10:8] Red y0 high byte
0x5803	RED Y0	0x60	RW	Bit[7:0]: red_y0[7:0] Red y0 low byte

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

address	register name	default value	R/W	description
0x5804	RED A1	0x22	RW	Bit[7]: Not used Bit[6:0]: red_a1[6:0] Red a1
0x5805	RED A2	0x07	RW	Bit[7:4]: Not used Bit[3:0]: red_a2[3:0] Red a2
0x5806	RED B1	0xC2	RW	Bit[7:0]: red_b1[7:0] Red b1
0x5807	RED B2	0x08	RW	Bit[7:4]: Not used Bit[3:0]: red_b2[3:0] Red b2
0x5808	GRN X0	0x03	RW	Bit[7:3]: Not used Bit[2:0]: grn_x0[10:8] Green x0 high byte
0x5809	GRN X0	0x28	RW	Bit[7:0]: grn_x0[7:0] Green x0 low byte
0x580A	GRN Y0	0x02	RW	Bit[7:3]: Not used Bit[2:0]: grn_y0[10:8] Green y0 high byte
0x580B	GRN Y0	0x60	RW	Bit[7:0]: grn_y0[7:0] Green y0 low byte
0x580C	GRN A1	0x22	RW	Bit[7]: Not used Bit[6:0]: grn_a1[6:0] Green a1
0x580D	GRN A2	0x07	RW	Bit[7:4]: Not used Bit[3:0]: grn_a2[3:0] Green a2
0x580E	GRN B1	0xC2	RW	Bit[7:0]: grn_b1[7:0] Green b1
0x580F	GRN B2	0x08	RW	Bit[7:4]: Not used Bit[3:0]: grn_b2[3:0] Green b2
0x5810	BLU X0	0x03	RW	Bit[7:3]: Not used Bit[2:0]: blu_x0[10:8] Blue x0 high byte
0x5811	BLU X0	0x28	RW	Bit[7:0]: blu_x0[7:0] Blue x0 low byte
0x5812	BLU Y0	0x02	RW	Bit[7:3]: Not used Bit[2:0]: blu_y0[10:8] Blue y0 high byte

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

address	register name	default value	R/W	description
0x5813	BLU Y0	0x60	RW	Bit[7:0]: blu_y0[7:0] Blue y0 low byte
0x5814	BLU A1	0x22	RW	Bit[7]: Not used Bit[6:0]: blu_a1[6:0] Blue a1
0x5815	BLU A2	0x07	RW	Bit[7:4]: Not used Bit[3:0]: blu_a2[3:0] Blue a2
0x5816	BLU B1	0xC2	RW	Bit[7:0]: blu_b1[7:0] Blue b1
0x5817	BLU B2	0x08	RW	Bit[7:4]: Not used Bit[3:0]: blu_b2[3:0] Blue b2
0x5818	LENC CTRL00	0x0C	RW	Bit[3]: rst_seed Bit[2]: rnd_en Bit[1]: coef_m_en Bit[0]: gcoef_en
0x5819	LENC COEF TH	0x80	RW	Bit[7:0]: lenc_coef_th[7:0] LENC coefficient threshold
0x581A	LENC GAIN THRE1	0x06	RW	Bit[7:6]: Not used Bit[5:0]: lenc_gain_thre1[5:0]
0x581B	LENC GAIN THRE2	0xC	RW	Bit[7:6]: Not used Bit[5:0]: lenc_gain_thre2[56:0]
0x581C	COEF MAN	0x80	RW	Bit[7:0]: coef_man[7:0]

4.3 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 and advanced AWB methods are supplied. Advance AWB takes into account the *color temperature* of a light source. For advanced AWB settings, contact your local OmniVision FAE.

table 4-3 AWB registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5180	R AWB CTRL00	0x51	RW	Bit[7:6]: local_step (inside local range) Bit[5:4]: fast_step (outside local range) Bit[3]: slop_8x Bit[2]: slop_4x Bit[1]: man_en Bit[0]: avg_all
0x5181	R AWB CTRL01	0x11	RW	Bit[7:4]: max_local_cnt Bit[3:0]: max_fast_cnt
0x5182	R AWB CTRL02	0x41	RW	Bit[7]: freeze Bit[6]: fast_enable Bit[5:4]: win_idx Bit[3:1]: min_white_pixel_sel Bit[0]: bias_stat
0x5183	R AWB CTRL03	0x42	RW	Bit[7:4]: stable_range_w Stable to unstable Bit[3:0]: stable_range Unstable to stable
0x5184	R AWB CTRL04	0x20	RW	Bit[7:0]: x0 (CWF_X)
0x5185	R AWB CTRL05	0x20	RW	Bit[7:0]: y0 (CWF_Y)
0x5186	R AWB CTRL 06	0x00	RW	Bit[7:0]: kx (CWF to A) x2y
0x5187	R AWB CTRL 07	0x00	RW	Bit[7:0]: ky (CFW to day) y2x
0x5188	R AWB CTRL 08	0x10	RW	Bit[7:0]: cwf_range (square)
0x5189	R AWB CTRL 09	0x10	RW	Bit[7:0]: a_range
0x518A	R AWB CTRL 0A	0x10	RW	Bit[7:0]: day_range
0x518B	R AWB CTRL 0B	0x00	RW	Bit[7:0]: day_limit
0x518C	R AWB CTRL 0C	0x00	RW	Bit[7:0]: a_limit
0x518D	R AWB CTRL 0D	0xF0	RW	Bit[7:0]: top_limit
0x518E	R AWB CTRL 0E	0x10	RW	Bit[7:0]: bot_limit

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

address	register name	default value	R/W	description
0x518F	R AWB CTRL 0F	0xFF	RW	Bit[7:0]: red_gain_max
0x5190	R AWB CTRL 10	0x00	RW	Bit[7:0]: red_gain_min
0x5191	R AWB CTRL 11	0xFF	RW	Bit[7:0]: green_gain_max
0x5192	R AWB CTRL 12	0x00	RW	Bit[7:0]: green_gain_min
0x5193	R AWB CTRL 13	0xFF	RW	Bit[7:0]: blue_gain_max
0x5194	R AWB CTRL 14	0x00	RW	Bit[7:0]: blue_gain_min
0x5195	R AWB CTRL 15	0x04	RW	Bit[3:0]: awb_r_gain_m[11:8]
0x5196	R AWB CTRL 16	0x00	RW	Bit[7:0]: awb_r_gain_m[7:0]
0x5197	R AWB CTRL 17	0x04	RW	Bit[3:0]: awb_g_gain_m[11:8]
0x5198	R AWB CTRL 18	0x00	RW	Bit[7:0]: awb_g_gain_m[7:0]
0x5199	R AWB CTRL 19	0x04	RW	Bit[3:0]: awb_b_gain_m[11:8]
0x519A	R AWB CTRL 1A	0x00	RW	Bit[7:0]: awb_b_gain_m[7:0]
0x519B	R AWB CTRL 1B	0x10	RW	Bit[7:0]: local_limit

4.4 color interpolation (CIP)

The CIP functions include de-noising of raw images, RAW to RGBH interpolation, and edge enhancement. In sensor RAW format, each pixel will be either R, G or B. CIP will calculate the other two color values using the neighboring pixel of the same color. Thus, we can get the full RGBH information for each pixel.

figure 4-1 relationship between noise_G and camera_gain

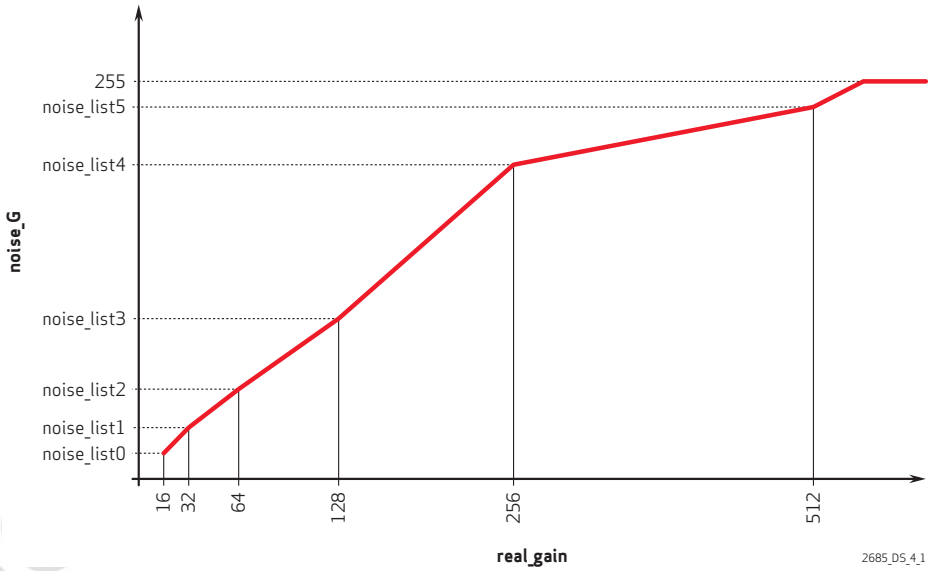


figure 4-2 relationship between CIP sharpen and camera_gain

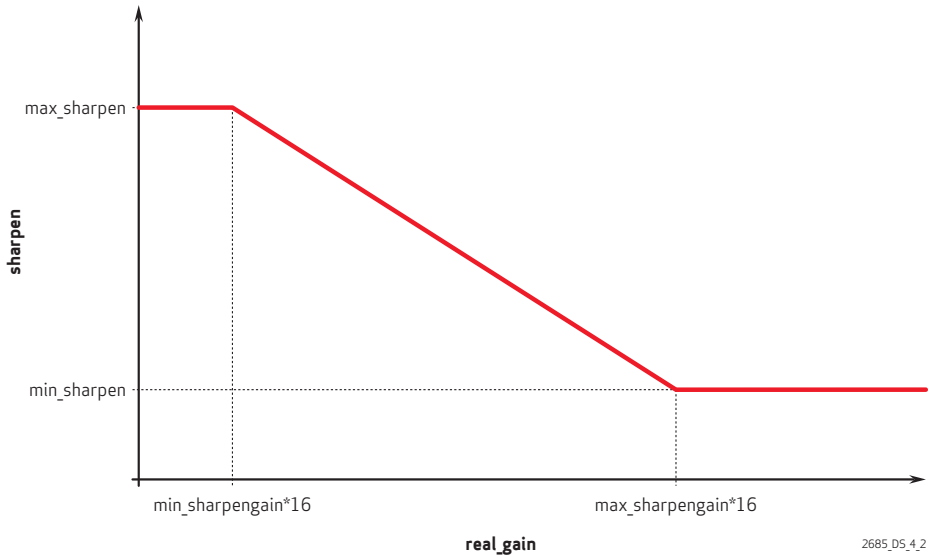


figure 4-3 relationship between CIP detail and camera_gain

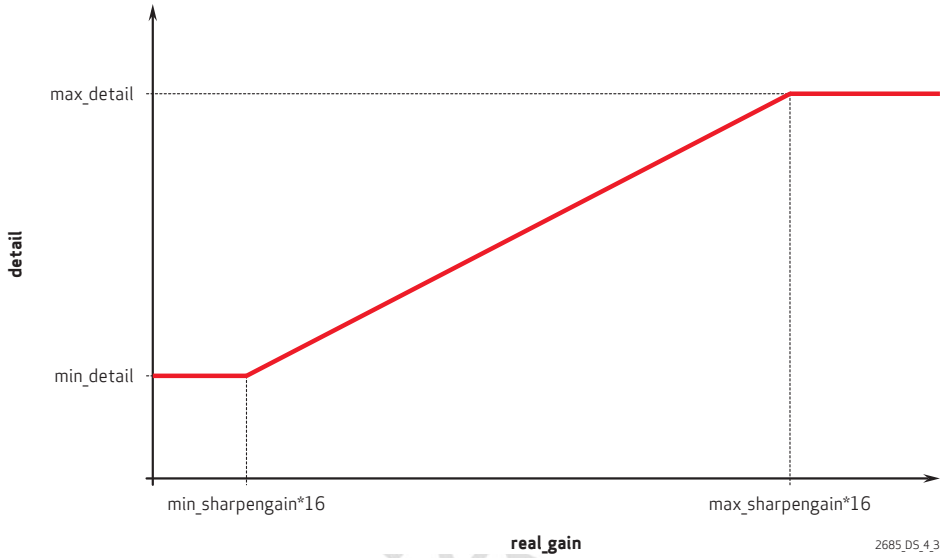


table 4-4 CIP registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5300	CIP_CTRL_00	0xE2	RW	Bit[7]: m_bColorEdgeEnable Bit[6]: m_bAntiAliasing Bit[5:3]: m_nDetailSlope Bit[2:0]: m_nNoiseYSlope
0x5301	CIP_CTRL_01	0xA0	RW	Bit[6:5]: m_nSharpenSlope Bit[4:1]: m_nGbGrShift Bit[0]: m_bGbGrRemove
0x5302	CIP_CTRL_02	0x00	RW	Bit[7:0]: m_nNoiseList[0]
0x5303	CIP_CTRL_03	0x00	RW	Bit[7:0]: m_nNoiseList[1]
0x5304	CIP_CTRL_04	0x00	RW	Bit[7:0]: m_nNoiseList[2]
0x5305	CIP_CTRL_05	0x00	RW	Bit[7:0]: m_nNoiseList[3]
0x5306	CIP_CTRL_06	0x00	RW	Bit[7:0]: m_nNoiseList[4]
0x5307	CIP_CTRL_07	0x00	RW	Bit[7:0]: m_nNoiseList[5]
0x5308	CIP_CTRL_08	0x82	RW	Bit[7:3]: m_nMaxSharpenGain Bit[2:0]: m_nMinSharpenGain
0x5309	CIP_CTRL_09	0x00	RW	Bit[5:0]: m_nMinSharpen
0x530A	CIP_CTRL_0A	0x0C	RW	Bit[5:0]: m_nMaxSharpen

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

address	register name	default value	R/W	description
0x530B	CIP_CTRL_0B	0x08	RW	Bit[7:0]: m_nMinDetail
0x530C	CIP_CTRL_0C	0x0C	RW	Bit[7:0]: m_nMaxDetail
0x530D	CIP_CTRL_0D	0x00	RW	Bit[5:0]: m_nDetailRatioList[0]
0x530E	CIP_CTRL_0E	0x0C	RW	Bit[5:0]: m_nDetailRatioList[1]
0x530F	CIP_CTRL_0F	0x14	RW	Bit[5:0]: m_nDetailRatioList[2]
0x5310	CIP_CTRL_10	0x10	RW	Bit[5:0]: m_nSharpenNegEdgeRatio
0x5311	CIP_CTRL_11	0x20	RW	Bit[7:0]: m_nClrEdgeShpT1
0x5312	CIP_CTRL_12	0x80	RW	Bit[7:0]: m_nClrEdgeShpT2
0x5313	CIP_CTRL_13	0x4B	RW	Bit[7:0]: m_nClrEdgeShpSlope
0x5314	CIP_CTRL_14	0x00	RW	Bit[6:1]: cip_sharpen_p_man Bit[0]: cip_para_man_en
0x5315	CIP_CTRL_15	0x00	RW	Bit[5:0]: cip_sharpen_m_man
0x5316	CIP_CTRL_16	0x00	RW	Bit[7:0]: cip_noise_man
0x5317	CIP_CTRL_17	0x00	RW	Bit[7:0]: cip_detail_man
0x5318	CIP_CTRL_18	–	R	Bit[7:0]: m_nNoise
0x5319	CIP_CTRL_19	–	R	Bit[7:0]: m_nDetail
0x531A	CIP_CTRL_1A	–	R	Bit[7:0]: m_nSharpenP
0x531B	CIP_CTRL_1B	–	R	Bit[7:0]: m_nSharpenM

4.5 de-noise (DNS)

Each pixel in the sensor converts the incoming light into an electrical signal which is processed into the color value of the pixel in the final image. If the same pixel is exposed several times by the same amount of light, the resulting color values would not be identical and thus, be called "noise". Even without incoming light, the electrical activity of the sensor itself will generate some signal. This additional signal is "noise". Noise in digital images is most visible in uniform surfaces. The main purpose of the DNS function is to decrease the noise.

In the OV2685, DNS works in three domains: RAW, RGB, and YUV.

4.5.1 RAW de-noise

The RAW_DNS uses a built-in recursive wavelet transform adaptive low pass filter to remove color noise in B, R, and G (option) pixels. It has thresholds for noise and edge. The thresholds are either set up through the SCCB or calculated automatically based on the AGC value.

table 4-5 RAW de-noise registers

address	register name	default value	R/W	description
0x5280	RAW_DNS_CTRL_00	0x08	RW	Bit[5:2]: m_nNoiseYSlope Bit[1]: raw_dns manual mode 0: Parameter noise and edgethre calculated by NoiseList 1: Parameter noise and edgethre configured directly by register Bit[0]: m_bGDNSEnable
0x5281	RAW_DNS_CTRL_01	0x04	RW	Bit[7:0]: m_nNoiseList[0]
0x5282	RAW_DNS_CTRL_02	0x08	RW	Bit[7:0]: m_nNoiseList[1]
0x5283	RAW_DNS_CTRL_03	0x10	RW	Bit[7:0]: m_nNoiseList[2]
0x5284	RAW_DNS_CTRL_04	0x18	RW	Bit[7:0]: m_nNoiseList[3]
0x5285	RAW_DNS_CTRL_05	0x20	RW	Bit[7:0]: m_nNoiseList[4]
0x5286	RAW_DNS_CTRL_06	0x30	RW	Bit[7:0]: m_nNoiseList[5]
0x5287	RAW_DNS_CTRL_07	0x3F	RW	Bit[7:0]: m_nMaxEdgeThre
0x5288	RAW_DNS_CTRL_08	0x08	RW	Bit[7:0]: Raw DNS noise parameter in manual mode
0x5289	RAW_DNS_CTRL_09	0x18	RW	Bit[7:0]: Raw DNS edgethre parameter in manual mode
0x528A	RAW_DNS_CTRL_10	–	R	Bit[7:0]: Noise value
0x528B	RAW_DNS_CTRL_11	–	R	Bit[7:0]: Edgethre value

4.5.2 RGB de-noise

RGB_DNS uses a one-line buffer and a built-in recursive wavelet transform adaptive low pass filter to remove color noise.

table 4-6 RGB de-noise registers

address	register name	default value	R/W	description
0x5480	RGB_DNS_CTRL00	0x19	RW	Bit[4:1]: m_nShadowExtraNoise Bit[0]: m_bSmoothYEnable
0x5481	RGB_DNS_CTRL01	0x08	RW	Bit[5:3]: m_nNoiseYList[1] Bit[2:0]: m_nNoiseYList[0]
0x5482	RGB_DNS_CTRL02	0x1A	RW	Bit[5:3]: m_nNoiseYList[3] Bit[2:0]: m_nNoiseYList[2]
0x5483	RGB_DNS_CTRL03	0x24	RW	Bit[5:3]: m_nNoiseYList[5] Bit[2:0]: m_nNoiseYList[4]
0x5484	RGB_DNS_CTRL04	0x04	RW	Bit[7:0]: m_nNoiseUVList[0]
0x5485	RGB_DNS_CTRL05	0x06	RW	Bit[7:0]: m_nNoiseUVList[1]
0x5486	RGB_DNS_CTRL06	0x08	RW	Bit[7:0]: m_nNoiseUVList[2]
0x5487	RGB_DNS_CTRL07	0x0C	RW	Bit[7:0]: m_nNoiseUVList[3]
0x5488	RGB_DNS_CTRL08	0x10	RW	Bit[7:0]: m_nNoiseUVList[4]
0x5489	RGB_DNS_CTRL09	0x18	RW	Bit[7:0]: m_nNoiseUVList[5]
0x548A	RGB_DNS_CTRL0A	0x00	RW	Bit[3:1]: rgb_dns_y_noise_man Bit[0]: rgb_dns_para_man_en
0x548B	RGB_DNS_CTRL0B	0x00	RW	Bit[7:0]: rgb_dns_uv_noise_man
0x548C	RGB_DNS_CTRL0C	–	R	Bit[7:0]: m_nNoiseY
0x548D	RGB_DNS_CTRL0D	–	R	Bit[7:0]: m_nNoiseUV

4.5.3 UV de-noise

UVDNS is designed to reduce UV channel pixel noise and increase the de-noise level in dark tones.

table 4-7 UV de-noise registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5500	UV_DNS_CTRL0	0x01	RW	Bit[7:0]: m_nNoiseList[0]
0x5501	UV_DNS_CTRL1	0x02	RW	Bit[7:0]: m_nNoiseList[1]

table 4-7 UV de-noise registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5502	UV_DNS_CTRL2	0x04	RW	Bit[7:0]: m_nNoiseList[2]
0x5503	UV_DNS_CTRL3	0x08	RW	Bit[7:0]: m_nNoiseList[3]
0x5504	UV_DNS_CTRL4	0x0C	RW	Bit[7:0]: m_nNoiseList[4]
0x5505	UV_DNS_CTRL5	0x10	RW	Bit[7:0]: m_nNoiseList[5]
0x5506	UV_DNS_CTRL6	0x06	RW	Bit[4]: uv_dns_para_man_en Bit[3:0]: m_nShadowExtraNoise
0x5507	UV_DNS_CTRL7	0x00	RW	Bit[7:0]: uv_dns_noise_man
0x5508	UV_DNS_CTRL8	–	R	Bit[7:0]: uv_dns_noise

4.6 color matrix (CMX)

The main purpose of the CMX function is to do a certain color transformation from YUVH to RGBH domain using a 2x3 matrix.

table 4-8 CMX registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5380	CMX_CTRL_00	0x00	RW	Bit[0]: Absolute value of nCCM_D[0][0] high byte
0x5381	CMX_CTRL_01	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[0][0] low byte
0x5382	CMX_CTRL_02	0x00	RW	Bit[0]: Absolute value of nCCM_D[0][1] high byte
0x5383	CMX_CTRL_03	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[0][1] low byte
0x5384	CMX_CTRL_04	0x00	RW	Bit[0]: Absolute value of nCCM_D[1][0] high byte
0x5385	CMX_CTRL_05	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[1][0] low byte
0x5386	CMX_CTRL_06	0x00	RW	Bit[0]: Absolute value of nCCM_D[1][1] high byte
0x5387	CMX_CTRL_07	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[1][1] low byte

table 4-8 CMX registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5388	CMX_CTRL_08	0x00	RW	Bit[0]: Absolute value of nCCM_D[2][0] high byte
0x5389	CMX_CTRL_09	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[2][0] low byte
0x538A	CMX_CTRL_0A	0x00	RW	Bit[0]: Absolute value of nCCM_D[2][1] high byte
0x538B	CMX_CTRL_0B	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[2][1] low byte
0x538C	CMX_CTRL_0C	0x00	RW	Bit[5]: Sign bit of nCCM_D[2][1] Bit[4]: Sign bit of nCCM_D[2][0] Bit[3]: Sign bit of nCCM_D[1][1] Bit[2]: Sign bit of nCCM_D[1][0] Bit[1]: Sign bit of nCCM_D[0][1] Bit[0]: Sign bit of nCCM_D[0][0]

4.7 line stretch

The auto contrast module is designed to linearly stretch the image to adjust the contrast according to statistics results. According to the parameters set by the user, the function can automatically calculate the maximum and minimum levels. Then, this range is stretched to the entire data range.

figure 4-4 relationship between stretch y_o and stretch y_i

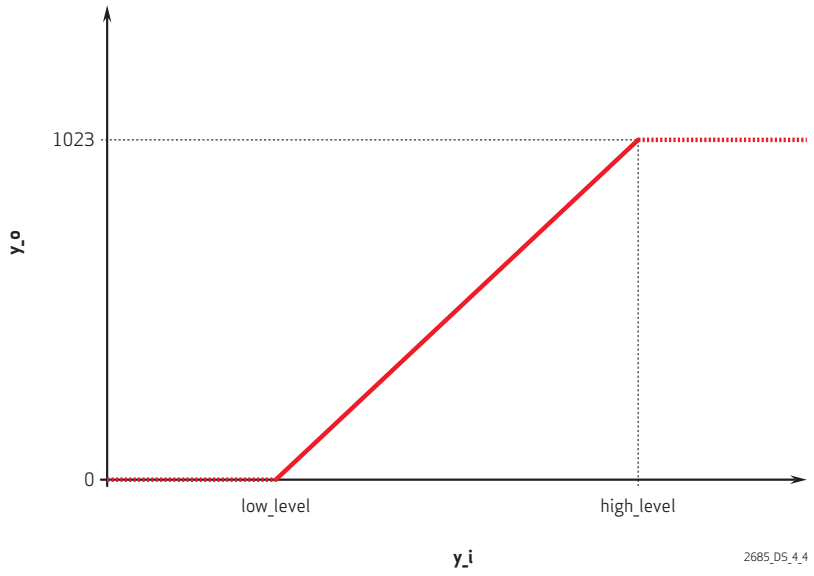


table 4-9 line stretch registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5200	STRETCH_CTRL0	0x09	RW	Bit[3:2]: Stretch minimum Real stretch minimum value is: 00: 1024 01: 2048 10: 3096 11: 8192 Bit[1]: Stretch manual enable 0: Manual disable 1: Manual enable Bit[0]: Stretch auto enable 0: Auto disable 1: Auto enable
0x5201	STRETCH_CTRL1	0x00	RW	Bit[5:0]: Stretch min low level
0x5202	STRETCH_CTRL2	0x06	RW	Bit[5:0]: Stretch max low level
0x5203	STRETCH_CTRL3	0x10	RW	Bit[4:0]: Stretch min high level
0x5204	STRETCH_CTRL4	0x41	RW	Bit[7:4]: Stretch step2 Bit[3:0]: Stretch step1
0x5205	STRETCH_CTRL5	0x10	RW	Bit[7:0]: Stretch current low level If minimum low level is 0, current low level must be greater or equal to step1
0x5206	STRETCH_CTRL6	0xFF	RW	Bit[7:0]: Stretch current high level[7:0]
0x5207	STRETCH_CTRL7	0x03	RW	Bit[3:0]: Stretch current high level[11:8]
0x5208	STRETCH_CTRL8	0x10	RW	Bit[7:0]: Stretch current low level manual value
0x5209	STRETCH_CTRL9	0xFF	RW	Bit[7:0]: Stretch current high level manual value[7:0]
0x520a	STRETCH_CTRL10	0x03	RW	Bit[3:0]: Stretch current high level manual value[11:8]
0x520b	STRETCH_CTRL11	0xC4	RW	Bit[7:0]: stretch_thres1[7:0]
0x520c	STRETCH_CTRL12	0x3B	RW	Bit[7:0]: stretch_thres1[15:8]
0x520D	STRETCH_CTRL13	0x00	RW	Bit[7:0]: stretch_thres1[23:16] stretch_thre1 = width × height × m_nPsThres1 >> 10
0x520E	STRETCH_CTRL14	0xC4	RW	Bit[7:0]: stretch_thres2[7:0]
0x520F	STRETCH_CTRL15	0x3B	RW	Bit[7:0]: stretch_thres2[15:8]
0x5210	STRETCH_CTRL16	0x00	RW	Bit[7:0]: stretch_thres2[23:16] stretch_thre2 = width × height × m_nPsThres2>>10
0x5211	STRETCH_CTRL17	–	R	Stretch Current Low Level

table 4-9 line stretch registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5212	STRETCH_CTRL18	–	R	Bit[7:0]: Stretch current high level[7:0]
0x5213	STRETCH_CTRL19	–	R	Bit[3:0]: Stretch current high level[11:8]
0x5214	STRETCH_CTRL20	–	R	Bit[7:0]: Stretch gain[7:0]
0x5215	STRETCH_CTRL21	–	R	Bit[3:0]: Stretch gain[11:8]

4.8 gamma

Gamma compensates for the non-linear characteristics of human visual perception in the RGB domain.

figure 4-5 relationship between gamma y_i and gamma y_o

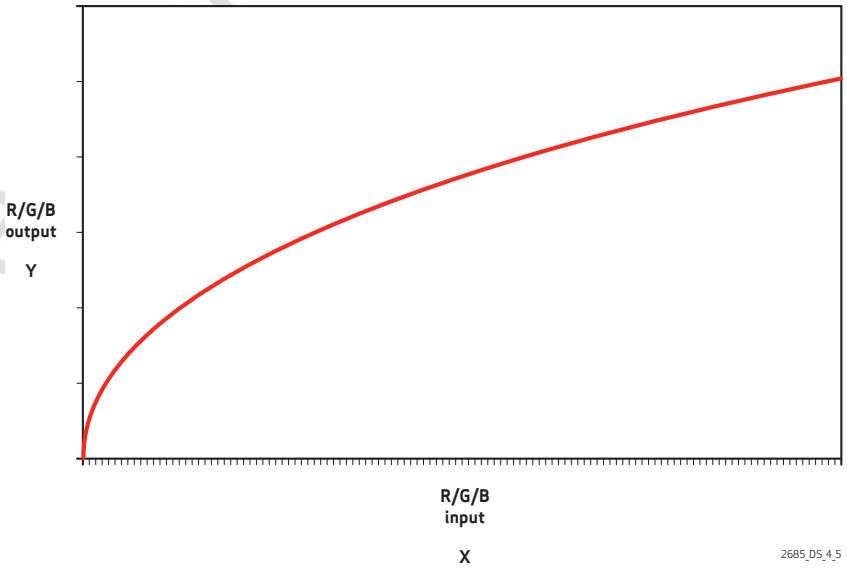


figure 4-6 relationship between nGain for H and gamma max component for R/G/B

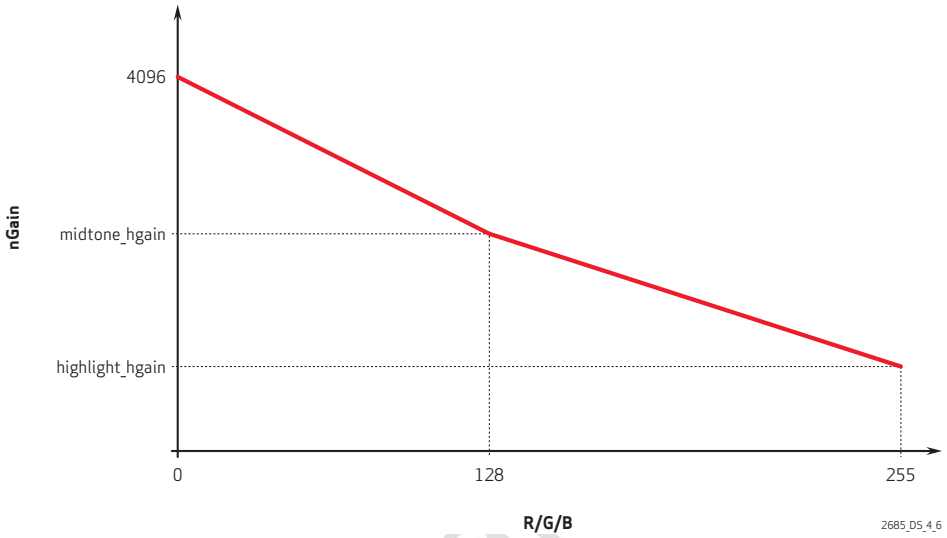


table 4-10 gamma registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5400	GAMMA_00	0x04	RW	Bit[7:0]: m_pCurveYList[0]
0x5401	GAMMA_01	0x08	RW	Bit[7:0]: m_pCurveYList[1]
0x5402	GAMMA_02	0x10	RW	Bit[7:0]: m_pCurveYList[2]
0x5403	GAMMA_03	0x20	RW	Bit[7:0]: m_pCurveYList[3]
0x5404	GAMMA_04	0x28	RW	Bit[7:0]: m_pCurveYList[4]
0x5405	GAMMA_05	0x30	RW	Bit[7:0]: m_pCurveYList[5]
0x5406	GAMMA_06	0x38	RW	Bit[7:0]: m_pCurveYList[6]
0x5407	GAMMA_07	0x40	RW	Bit[7:0]: m_pCurveYList[7]
0x5408	GAMMA_08	0x48	RW	Bit[7:0]: m_pCurveYList[8]
0x5409	GAMMA_09	0x50	RW	Bit[7:0]: m_pCurveYList[9]
0x540A	GAMMA_0A	0x60	RW	Bit[7:0]: m_pCurveYList[10]
0x540B	GAMMA_0B	0x70	RW	Bit[7:0]: m_pCurveYList[11]
0x540C	GAMMA_0C	0x90	RW	Bit[7:0]: m_pCurveYList[12]
0x540D	GAMMA_0D	0xB0	RW	Bit[7:0]: m_pCurveYList[13]
0x540E	GAMMA_0E	0xD0	RW	Bit[7:0]: m_pCurveYList[14]

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

address	register name	default value	R/W	description
0x540F	GAMMA_0F	0xFF	RW	Bit[7:0]: m_nMaxShadowHGain
0x5410	GAMMA_10	0x6E	RW	Bit[7:0]: m_nMidToneHGain
0x5411	GAMMA_11	0x10	RW	Bit[7:0]: m_nHighLightHGain

4.9 special digital effects (SDE)

The special digital effects (SDE) functions include brightness, contrast, etc. SDE also supports negative, black/white, sepia, greenish, blueish, reddish, and other image effects.

table 4-11 SDE registers

address	register name	default value	R/W	description
0x5600	R SDE CTRL0	0x00	RW	Bit[7]: fixy_en Bit[6]: neg_en Bit[5]: gray_en Bit[4]: fix_v_en Bit[3]: fix_u_en Bit[2]: contrast_en Bit[1]: satuation_en
0x5603	R SDE CTRL3	0x40	RW	Bit[7:0]: sat_u or ureg
0x5604	R SDE CTRL4	0x00	RW	Bit[7:0]: sat_v or vreg
0x5605	R SDE CTRL5	0x00	RW	Bit[7:0]: Yoffset
0x5606	R SDE CTRL6	0x20	RW	Bit[7:0]: Ygain
0x5607	R SDE CTRL7	0x00	RW	Bit[7:0]: Ybright
0x5608	R SDE CTRL8	0x00	RW	Bit[7:6]: Not used Bit[5:0]: Signset
0x5609	R SDE CTRL9	0x08	RW	Bit[7:0]: uvadj_th1
0x560A	R SDE CTRLA	0x80	RW	Bit[7:0]: uvadj_th2
0x560B	R SDE CTRLB	0x00	RW	Bit[7:1]: Not used Bit[0]: uvadj_man_en
0x560C	R SDE CTRLC	–	R	Bit[7:0]: uvadj_man_i

4.10 white/black pixel cancellation (WBC)

The main purpose of the WBC function is to remove white/black pixel defects.

WBC algorithms only do white/black pixel cancellation in the horizontal and vertical directions. The WBC function uses filters to determine which pixel is white/black and does this based on the channel.

table 4-12 WBC registers

address	register name	default value	R/W	description
0x5000	ISP CTRL0	0x1F	RW	Bit[2]: Black pixel cancellation enable signal Bit[1]: White pixel cancellation enable signal

4.11 black level calibration (BLC)

The OV2685 black level calibration function compensates for dark current to ensure constant output of black level regardless of changes in exposure time, gain, and temperature.

table 4-13 BLC control functions

function	register	description
auto/manual mode	0x4001	Bit[4]: BLC manual mode 0: Auto mode 1: Manual mode
target	0x4002 ~ 0x4003	Black target to be achieved {0x4002[1:0], 0x4003[7:0]}
blacklevel B manual offset	0x4030 ~ 0x4031	BLCBMOffs[10:0] = {0x4030[2:0], 0x4031[7:0]}
blacklevel Gb manual offset	0x4032 ~ 0x4033	BLCGBMOffs[10:0] = {0x4032[2:0], 0x4033[7:0]}
blacklevel Gr manual offset	0x4034 ~ 0x4035	BLCGRMOffs[10:0] = {0x4034[2:0], 0x4035[7:0]}
blacklevel R manual offset	0x4036 ~ 0x4037	BLCRMOffs[10:0] = {0x4036[2:0], 0x4037[7:0]}

4.12 average algorithms

The OV2685 average module uses raw data as input data for calculation. Based on the start address and the horizontal and vertical window size, the selected area will be used as 16 zones to calculate the average value of the image.

table 4-14 AVG output information

function	register	description
average	0x5913	Bit[7:0]: average[7:0]

4.13 test pattern

For testing purposes, the OV2685 offers three types of test patterns: color bar, square, and random data.

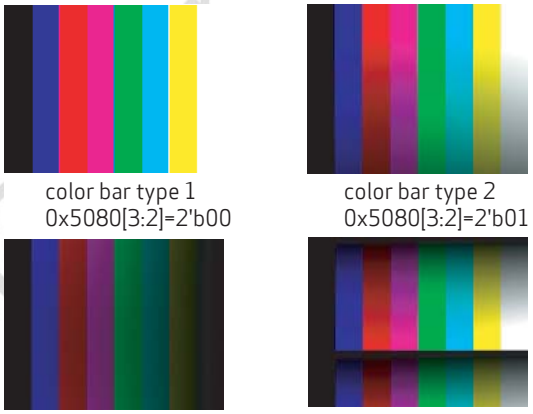
Also, the OV2685 offers two effects: transparent effect and rolling bar effect.

The output type of test pattern is controlled by the test_pattern_type register (0x5080[1:0]).

4.13.1 color bar

There are four types of color bars (see **figure 4-7**), which can be switched using the bar_style register (0x5080[3:2]).

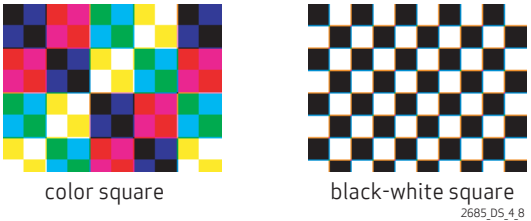
figure 4-7 color bar test patterns



4.13.2 square

There are two types of square patterns (see **figure 4-8**): color square and black-white square. The square_mode register (0x5080[4]) determines which type of square pattern will be output. This works only when register 0x5080[1:0] = 2'b10.

figure 4-8 square test patterns



4.13.3 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 determined by the same_seed_en register (0x5081[4]). The random seed is set by the seed register (0x5081[3:0]).

4.13.4 transparent effect

The transparent effect is enabled by the transparent_en register (0x5080[5]). If this register is set, the transparent test pattern will be output. **figure 4-9** shows an example of a transparent color bar image.

figure 4-9 transparent effect

4.13.5 rolling bar effect

The rolling bar is set by the rolling_bar_en register (0x5080[6]). If this register bit is set, an inverted-color rolling bar will roll from top to bottom of the color bar pattern. **figure 4-10** shows an example of a rolling bar on a color bar image.

figure 4-10 rolling bar effect

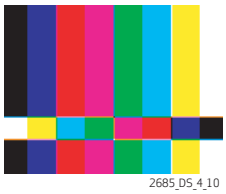


table 4-15 general color bar selection control

address	register name	default value	R/W	description
0x5080	PRE ISP CTRL00	0x00	RW	Bit[7]: test_en Test enable 0: Disable test function 1: Enable test function Bit[6]: rolling_bar_en 0: Disable rolling bar function 1: Enable rolling bar function Bit[5]: transparent_en 0: Disable transparent effect function 1: Enable transparent effect function Bit[4]: square_mode 0: Color square 1: Black-white square Bit[3:2]: color_bar_style 00: Standard color bar 01: Top-bottom darker color bar 10: Right-left darker color bar 11: Bottom-top darker color bar Bit[1:0]: test_mode 00: Color bar 01: Random data 10: Square 11: Black image
0x5081	PRE ISP CTRL01	0x41	RW	Bit[4]: same_seed_en When this bit is set, the seed used to generate the random data is the same as what is set in the seed register Bit[3:0]: seed This is the seed used in generating the random data

5 register tables

The following tables provide descriptions of the device control registers contained in the OV2685. For all register enable/disable bits, enable = 1 and DISABLE = 0. The 7-bit device slave address is 0x20 when SID pin is set to 1 or 0x78 when SID pin is set to 0.

5.1 system control [0x0100, 0x0103, 0x3002, 0x300A - 0x3034]

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

address	register name	default value	R/W	description
0x0100	SMIA R0100	0x00	RW	Bit[7:1]: Not used Bit[0]: Mode select 0: software_standby 1: Streaming
0x0103	SMIA R0103	–	W	Bit[7:1]: Not used Bit[0]: software_reset
0x3002	PAD OEN	0x6C	RW	Bit[3]: Reserved Bit[2]: io_gpio0_oen Bit[1]: io_vsync_oen Bit[0]: io_fsin_oen
0x300A	CHIP ID	0x26	R	Bit[7:0]: chip_id[23:16]
0x300B	CHIP ID	0x85	R	Bit[7:0]: chip_id[15:8]
0x300C	CHIP ID	0x00	R	Bit[7:0]: chip_id[7:0]
0x300D	PAD OUT2	0x00	RW	Bit[3]: Not used Bit[2]: io_gpio0_o Bit[1]: io_vsync_o Bit[0]: io_fsin_o
0x3010	PAD SEL2	0x00	RW	Bit[3]: Not used Bit[2]: io_gpio0_sel Bit[1]: io_vsync_sel Bit[0]: io_fsin_sel
0x3011	PAD	0x02	RW	Bit[7]: pd_ana Bit[6]: pd_pwc Bit[5:3]: Not used Bit[1:0]: ip2x3v[3:0]

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

address	register name	default value	R/W	description
0x3016	MIPI PHY	0x10	RW	Bit[7:6]: pgm_lph Bit[5:4]: pgm_lptx[1:0] Driving strength of low speed transmitter Bit[3]: bp_c_hs_en_lat Bit[2]: bp_d_hs_en_lat Bypass latch of hs_enable Bit[1:0]: ictl[1:0] Bias current adjustment
0x3017	MIPI PHY	0x00	RW	Bit[7:6]: pgm_vcm[1:0] High speed common mode voltage Bit[5:4]: Not used Bit[3:2]: d0_skew Bit[1:0]: ck_skew
0x3018	MIPI SC CTRL	0x84	RW	Bit[7:6]: lane_num 00: 0 lane 01: 1 lane 10: 2 lanes Bit[5]: Not used Bit[4]: r_phy_pd_mipi 0: MIPI enable 1: Power down PHY HS TX Bit[3:2]: mipi_bit_sel 00: 8-bit mode 01: 10-bit mode 10: 12-bit mode 11: Reserved Bit[1]: mipi_lane_dis Bit[0]: mipi_pad_o
0x3019	MIPI SC CTRL	0x40	RW	Bit[7:0]: mipi_sc_ctrl[7:0] MIPI ULPS resume mark1 detect length
0x301A	CLKRST0	0xF0	RW	Bit[7]: Not used Bit[6]: sclk_psram Bit[5]: sclk_aec Bit[4]: sclk_tc Bit[3]: mipi_phy_rst_o Bit[2]: rst_psram Bit[1]: rst_aec Bit[0]: rst_tc

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

address	register name	default value	R/W	description
0x301B	CLKRST1	0xF0	RW	Bit[7]: sclk_blc Bit[6]: sclk_isp Bit[5]: sclk_avg Bit[4]: sclk_vfifo Bit[3]: rst_blc Bit[2]: rst_isp Bit[1]: rst_avg Bit[0]: rst_vfifo
0x301C	CLKRST2	0xF0	RW	Bit[7]: pclk_dvp Bit[6]: sclk_mipi Bit[5]: sclk_sync Bit[4]: sclk_asram_tst Bit[3]: rst_dvp Bit[2]: rst_mipi Bit[1]: rst_sync Bit[0]: rst_asram_tst
0x301D	CLKRST3	0xF0	RW	Bit[7]: sclk_fc Bit[6]: sclk_grp Bit[5]: sclk_bist Bit[4]: daclk_sel0 Bit[3]: rst_fc Bit[2]: rst_grp Bit[1]: rst_bist Bit[0]: rst_ac
0x301E	CLKRST4	0xF0	RW	Bit[7]: sdclk_sd Bit[6]: padclk_mipi_sc Bit[5]: pclk_vfifo Bit[4]: pclk_mipi Bit[3]: rst_sd Bit[2]: rst_mipi_sc Bit[1:0]: Not used
0x301F	FREX RST MASK0	0x00	RW	Bit[7]: frex_mask_aec Bit[6]: frex_mask_blc Bit[5]: frex_mask_isp Bit[4]: frex_mask_dvp Bit[3]: frex_mask_mipi Bit[2]: frex_mask_vfifo Bit[1]: frex_mask_avg Bit[0]: frex_mask_mipi_phy
0x3020	CLOCK SEL	0x00	RW	Bit[5]: yuv_out_en 0: Output RAW data 1: Output YUV data Bit[3]: pclk_sel Bit[2]: auto_rst_mipiphy_disable Bit[1]: auto_pwd_mipiphy_disable Bit[0]: sclk2x_sel

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

address	register name	default value	R/W	description
0x3021	MISC CTRL	0x03	RW	Bit[5]: fst_stby_ctr 0: Software standby enter at v_blk 1: Software standby enter at l_blk Bit[4]: mipi_ctr_en 0: Disable function 1: Enable MIPI remote reset and suspend control SC Bit[3]: mipi_rst_sel 0: MIPI remote reset all registers 1: MIPI remote reset all digital modules Bit[2]: gpio_pclk_en Bit[1]: frex_ef_sel Bit[0]: cen_global_o
0x302A	SUB ID	–	R	Bit[7:4]: Process Bit[3:0]: Version
0x302B	CTRL43	0x78	RW	Bit[7:0]: Ctrl43
0x302C	TM REG	–	W	Bit[7]: tm_enable
0x3030	REG30	0x88	RW	Bit[7]: daclk_sel Bit[6]: daclk Bit[5:4]: Not used Bit[3]: mask_daclk Bit[2:0]: Not used
0x3031	REG31	0x55	RW	Bit[7]: sccb_id2_nack Bit[6:4]: p_pump_div Bit[3]: Not used Bit[2:0]: n_pump_div
0x3032	REG32	0x90	RW	Bit[7:0]: sccb_id2
0x3033	REG33 CTRL	0x20	RW	Bit[7:0]: Reg33
0x3034	IO PAD	–	R	Bit[2]: p_fsin_i Bit[1]: p_vsync_i Bit[0]: p_gpio0_i

5.2 PLL control [0x3080 - 0x3088]

table 5-2 PLL control registers

address	register name	default value	R/W	description
0x3080	PLL PREDIV	0x02	RW	Bit[7:3]: Not used Bit[2:0]: pll_prediv[2:0]
0x3081	PLL MULTIPLIER	0x00	RW	Bit[7:1]: Not used Bit[0]: pll_multiplier[8]
0x3082	PLL MULTIPLIER	0x37	RW	Bit[7:0]: pll_multiplier[7:0]
0x3083	PLL MIPI DIV	0x03	RW	Bit[7:2]: Not used Bit[1:0]: pll_mipi_div[1:0]
0x3084	PLL SYS CLK DIV	0x09	RW	Bit[7:4]: Not used Bit[3:0]: pll_sys_clk_div[3:0]
0x3085	PLL DAC DIV	0x04	RW	Bit[7:4]: Not used Bit[3:0]: pll_dac_div[3:0]
0x3086	PLL SP DIV	0x00	RW	Bit[7:4]: Not used Bit[3:0]: pll_sp_div[3:0]
0x3087	PLL LANE DIV	0x00	RW	Bit[7:2]: Not used Bit[1:0]: pll_lane_div[1:0]
0x3088	PLL CTRL	0x01	RW	Bit[7]: pll_rst Bit[6]: pll_bypass Bit[5]: pll_freq_sel Bit[4]: pll_pre_div0 Bit[2:0]: pll_cp

5.3 SCCB control [0x3100 - 0x3106]

table 5-3 SCCB 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
0x3105	CTRL5	0x10	RW	Bit[5]: sclk use p_clk_i Bit[4]: sleep_en
0x3106	CTRL6	0x01	RW	Bit[3:2]: sclk_div_opt Bit[1]: rst_arb Bit[0]: byp_arb

5.4 group hold [0x3200 - 0x320F]

table 5-4 group hold registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3200	GROUP ADR0	0x00	RW	Bit[7:3]: Not used Bit[2:0]: group_adr0[2:0]
0x3201	GROUP ADR1	0x00	RW	Bit[7:3]: Not used Bit[2:0]: group_adr1[2:0]
0x3202	GROUP ADR2	0x00	RW	Bit[7:3]: Not used Bit[2:0]: group_adr2[2:0]
0x3203	GROUP ADR3	0x00	RW	Bit[7:3]: Not used Bit[2:0]: group_adr3[2:0]

table 5-4 group hold registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3204	CTRL4	–	W	Bit[7:0]: Ctrl4
0x3205	CTRL5	–	W	Bit[7:0]: Ctrl5
0x3206	CTRL6	–	W	Bit[7:0]: Ctrl6
0x3207	CTRL7	–	W	Bit[7:0]: Ctrl7
0x3208	GROUP ACCESS	0x00	RW	Bit[7:4]: group_ctrl 0000: Group hold start 0001: Group hold end 1010: Group launch Others: Reserved Bit[3:0]: group_id 0000: Group bank 0, default start from address 0x00 0001: Group bank 1, default start from address 0x00 Others: Reserved
0x3209	GRP0	0x00	RW	Bit[7:0]: grp0 Frames for staying in grp0
0x320A	GRP1	0x00	RW	Bit[7:0]: grp1 Frames for staying in grp1
0x320B	GRP SWCTRL	0x01	RW	Bit[7]: auto_sw Bit[4]: Not used Bit[3]: group_switch_repeat Bit[2]: context_en Bit[1:0]: Second group selection
0x320D	GRP ACT	–	R	Bit[7:0]: grp_act Indicates which group is active
0x320E	CTRL14	–	R	Bit[7:0]: frm_cnt_grp0
0x320F	CTRL15	–	R	Bit[7:0]: frm_cnt_grp1

5.5 AEC/AGC [0x3500 - 0x3503, 0x350A - 0x350B]

table 5-5 AEC/AGC registers

address	register name	default value	R/W	description
0x3500	EXPO PK	0x00	RW	Bit[7:4]: Not used Bit[3:0]: expo_pk[15:12]
0x3501	EXPO PK	0x02	RW	Bit[7:0]: expo_pk[11:4]
0x3502	EXPO PK	0x00	RW	Bit[7:4]: expo_pk[3:0] Bit[3:0]: Not used
0x3503	R MANUAL	0x00	RW	Bit[7:6]: Not used Bit[5]: Gain delay option 0: 1 frame latch 1: Delay 1 frame latch Bit[4]: Choose delay option 0: Delay disable 1: Delay enable Bit[3]: gain_change_delay Bit[2]: vts_manual Bit[1]: agc_manual Bit[0]: aec_manual
0x350A	GAIN PK	0x00	RW	Bit[7:3]: Not used Bit[2:0]: gain_pk[10:8]
0x350B	GAIN PK	0x10	RW	Bit[7:0]: gain_pk[7:0]

5.6 analog control [0x3600 - 0x362A]

table 5-6 analog control registers

address	register name	default value	R/W	description
0x3600~ 0x3604	ANALOG CONTROL	–	–	Analog Control Registers
0x3605	ADC	0x00	RW	Bit[7:4]: vrbglp Bit[3]: bp_regulator Bit[2]: bp_p_pump Bit[1:0]: ckde
0x3606~ 0x362A	ANALOG CONTROL	–	–	Analog Control Registers

5.7 sensor control [0x3700 - 0x3741]

table 5-7 sensor control registers

address	register name	default value	R/W	description
0x3700~ 0x3741	SENSOR CONTROL	–	–	Sensor Control Registers

5.8 PSRAM [0x3780 - 0x3798]

table 5-8 PSRAM registers

address	register name	default value	R/W	description
0x3780~ 0x3798	PSRAM CONTROL	–	–	PSRAM Control Registers

5.9 timing control [0x3800 - 0x383E]

table 5-9 timing control registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x3800	X ADDR START	0x00	RW	Bit[7:0]: x_addr_start[15:8] Array horizontal start point high byte
0x3801	X ADDR START	0x00	RW	Bit[7:0]: x_addr_start[7:0] Array horizontal start point low byte
0x3802	Y ADDR START	0x00	RW	Bit[7:0]: y_addr_start[15:8] Array vertical start point high byte
0x3803	Y ADDR START	0x00	RW	Bit[7:0]: y_addr_start[7:0] Array vertical start point low byte
0x3804	X ADDR END	0x06	RW	Bit[7:0]: x_addr_end[15:8] Array horizontal end point high byte
0x3805	X ADDR END	0x4F	RW	Bit[7:0]: x_addr_end[7:0] Array horizontal end point low byte
0x3806	Y ADDR END	0x04	RW	Bit[7:0]: y_addr_end[15:8] Array vertical end point high byte

table 5-9 timing control registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x3807	Y ADDR END	0xBF	RW	Bit[7:0]: y_addr_end[7:0] Array vertical end point low byte
0x3808	X OUTPUT SIZE	0x06	RW	Bit[7:0]: x_output_size[15:8] ISP horizontal output width high byte
0x3809	X OUTPUT SIZE	0x40	RW	Bit[7:0]: x_output_size[7:0] ISP horizontal output width low byte
0x380A	Y OUTPUT SIZE	0x04	RW	Bit[7:0]: y_output_size[15:8] ISP vertical output height high byte
0x380B	Y OUTPUT SIZE	0xB0	RW	Bit[7:0]: y_output_size[7:0] ISP vertical output height low byte
0x380C	HTS	0x06	RW	Bit[7:0]: HTS[15:8] Total pixels per line high byte
0x380D	HTS	0xA4	RW	Bit[7:0]: HTS[7:0] Total pixels per line low byte
0x380E	VTS	0x05	RW	Bit[7:0]: VTS[15:8] Total lines per frame high byte
0x380F	VTS	0x0E	RW	Bit[7:0]: VTS[7:0] Total lines per frame low byte
0x3810	ISP X WIN	0x00	RW	Bit[7:0]: isp_x_win[15:8] ISP horizontal windowing offset high byte
0x3811	ISP X WIN	0x08	RW	Bit[7:0]: isp_x_win[7:0] ISP horizontal windowing offset low byte
0x3812	ISP Y WIN	0x00	RW	Bit[7:0]: isp_y_win[15:8] ISP vertical windowing offset high byte
0x3813	ISP Y WIN	0x08	RW	Bit[7:0]: isp_y_win[7:0] ISP vertical windowing offset low byte
0x3814	X INC	0x11	RW	Bit[7:4]: x_odd_inc Bit[3:0]: x_even_inc
0x3815	Y INC	0x11	RW	Bit[7:4]: y_odd_inc Bit[3:0]: y_even_inc
0x3816	VSYNC START ROW	0x00	RW	Bit[7:0]: vsync_start_row[15:8]
0x3817	VSYNC START ROW	0x00	RW	Bit[7:0]: vsync_start_row[7:0]
0x3818	VSYNC END ROW	0x00	RW	Bit[7:0]: vsync_end_row[15:8]
0x3819	VSYNC END ROW	0x04	RW	Bit[7:0]: vsync_end_row[7:0]

table 5-9 timing control registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x3820	FORMAT1	0xC0	RW	Bit[7]: vsub48_blc Bit[6]: vflip_blc Bit[5:3]: Not used Bit[2]: vflip Bit[1]: vbinf Bit[0]: Not used
0x3821	FORMAT2	0x00	RW	Bit[7:4]: Not used Bit[3]: manual format configuration Bit[2]: mirror Bit[1]: ASRAM readout pixel order Bit[0]: hbin
0x3822	REG22	0x46	RW	Bit[7:5]: addr0_num[3:1] Bit[4:0]: ablc_num[5:1]
0x3823	REG23	0x00	RW	Bit[7]: r_disable_vref_rst Bit[6]: ext_vs_re Bit[5]: ext_vs_en Bit[4]: r_init_man Bit[3]: r_fix_cnt_en Bit[2:0]: ablc_adj
0x3824	CS RST FSIN	0x00	RW	Bit[7:0]: cs_rst_fsin[15:8] CS reset value high byte at vs_ext
0x3825	CS RST FSIN	0x10	RW	Bit[7:0]: cs_rst_fsin[7:0] CS reset value low byte at vs_ext
0x3826	R RST FSIN	0x00	RW	Bit[7:0]: r_rst_fsin[15:8] R reset value high byte at vs_ext
0x3827	R RST FSIN	0x00	RW	Bit[7:0]: r_rst_fsin[7:0] R reset value low byte at vs_ext
0x3828	FVTS	0x00	RW	Bit[7:0]: fvts[15:8] Fractional vertical timing size high byte
0x3829	FVTS	0x00	RW	Bit[7:0]: fvts[7:0] Fractional vertical timing size low byte
0x382A	REG2A	0x00	RW	Bit[7:5]: Not used Bit[4]: frame_insert Bit[3]: vts_auto_en Bit[1:0]: href_w
0x382B	REG2B	0x02	RW	Bit[7:5]: Not used Bit[4:0]: grp_wr_start Set value at least larger than zline_number + 1

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

address	register name	default value	R/W	description
0x382C	REG2C	0xC5	RW	Bit[7]: isp_x_win_auto Bit[6]: isp_y_win_auto Bit[5]: Not used Bit[4:0]: tc_r_int_adj
0x382D	REG2D	0x00	RW	Bit[7:2]: Not used Bit[1]: vsync_polarity Bit[0]: first_frame_begin_dis
0x382F	REG2F	0x0D	RW	emb_start_adj
0x3831	GAIN ADJ	0x00	RW	Bit[7]: pregain_man_en Bit[6]: pregain_ctrl_8x Bit[5]: pregain_ctrl_4x Bit[4]: pregain_ctrl_2x Bit[3]: gain_use_aecpk 0: Use mapping gain 1: Use aec_pk_gain Bit[2:0]: Not used
0x383C	VSYNC CS POINT	0x00	RW	Bit[7:0]: VSYNC fine start point[15:8]
0x383D	VSYNC CS POINT	0x01	RW	Bit[7:0]: VSYNC fine start point[7:0]
0x383E	VSYNC CS POINT	0x00	RW	Bit[7]: vts_adj_trigger Bit[6]: vts_adj_step

5.10 BLC [0x4000 - 0x404F]

table 5-10 BLC registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x4000	BLC CTRL00	0x01	RW	Bit[7:4]: avg_weight Bit[3]: target_adj_dis Bit[2]: cmp_en Bit[1]: dither_en Bit[0]: mf_en
0x4001	BLC CTRL01	0x40	RW	Bit[7]: gain_trig_beh Bit[6]: format_trig_beh Bit[5]: Not used Bit[4]: off_man_en Bit[3]: Not used Bit[2]: blk_ln_out_en Bit[1:0]: byp_mode

table 5-10 BLC registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x4002	BLK LVL TARGET	0x00	RW	Bit[7:2]: Not used Bit[1:0]: blk_lvl_target[9:8] blk_lvl_target high 2 bits
0x4003	BLK LVL TARGET	0x10	RW	Bit[7:0]: blk_lvl_target[7:0] blk_lvl_target low 8 bits
0x4004	HWIN OFF	0x00	RW	Bit[7:3]: Not used Bit[2:0]: hwin_off[10:8] hwin_off high 3 bits
0x4005	HWIN OFF	0x02	RW	Bit[7:0]: hwin_off[7:0] hwin_off low 8 bits
0x4006	HWIN PAD	0x00	RW	Bit[7:3]: Not used Bit[2:0]: hwin_pad[10:8] hwin_pad high 3 bits
0x4007	HWIN PAD	0x02	RW	Bit[7:0]: hwin_pad[7:0] hwin_pad low 8 bits
0x4008	BLC CTRL08	0x00	RW	Bit[7:0]: bl_start
0x4009	BLC CTRL09	0x0B	RW	Bit[7:0]: bl_end
0x400A	OFF LIM TH	0x02	RW	Bit[7:4]: Not used Bit[3:0]: off_lim_th[11:8] off_lim_th high 4 bits
0x400B	OFF LIM TH	0x00	RW	Bit[7:0]: off_lim_th[7:0] off_lim_th low 8 bits
0x4010	BLC CTRL10	0xF0	RW	Bit[7]: off_trig_en Bit[6]: gain_chg_trig_en Bit[5]: fmt_chg_trig_en Bit[4]: rst_trig_en Bit[3]: man_avg_en Bit[2]: man_trig Bit[1]: off_frz_en Bit[0]: off_always_up
0x4011	BLC CTRL11	0xFF	RW	Bit[7]: r_qoption_en Bit[6]: off_chg_mf_en Bit[5]: fmt_chg_mf_en Bit[4]: gain_chg_mf_en Bit[3]: rst_mf_mode Bit[2]: off_chg_mf_mode Bit[1]: fmt_chg_mf_mode Bit[0]: gain_chg_mf_mode
0x4012	BLC CTRL12	0x08	RW	Bit[5:0]: rst_trig_fn
0x4013	BLC CTRL13	0x02	RW	Bit[5:0]: fmt_trig_fn

table 5-10 BLC registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x4014	BLC CTRL14	0x02	RW	Bit[5:0]: gain_trig_fn
0x4015	BLC CTRL15	0x02	RW	Bit[5:0]: off_trig_fn
0x4016	OFF TRIG TH	0x00	RW	Bit[7:2]: Not used Bit[1:0]: off_trig_th[9:8] off_trig_th high 2 bits
0x4017	OFF TRIG TH	0x04	RW	Bit[7:0]: off_trig_th[7:0] off_trig_th low 8 bits
0x4020	BLC CTRL20	0x00	RW	Bit[5:0]: off_cmp_th000
0x4021	BLC CTRL21	0x00	RW	Bit[5:0]: off_cmp_k000
0x4022	BLC CTRL22	0x00	RW	Bit[5:0]: off_cmp_th001
0x4023	BLC CTRL23	0x00	RW	Bit[5:0]: off_cmp_k001
0x4024	BLC CTRL24	0x00	RW	Bit[5:0]: off_cmp_th010
0x4025	BLC CTRL25	0x00	RW	Bit[5:0]: off_cmp_k010
0x4026	BLC CTRL26	0x00	RW	Bit[5:0]: off_cmp_th011
0x4027	BLC CTRL27	0x00	RW	Bit[5:0]: off_cmp_k011
0x4030	OFF MAN000	0x00	RW	Bit[7:3]: Not used Bit[2:0]: off_man000[10:8] off_man000 high 3 bits
0x4031	OFF MAN000	0x00	RW	Bit[7:0]: off_man000[7:0] off_man000 low 8 bits
0x4032	OFF MAN001	0x00	RW	Bit[7:3]: Not used Bit[2:0]: off_man001[10:8] off_man001 high 3 bits
0x4033	OFF MAN001	0x00	RW	Bit[7:0]: off_man001[7:0] off_man001 low 8 bits
0x4034	OFF MAN010	0x00	RW	Bit[7:3]: Not used Bit[2:0]: off_man010[10:8] off_man010 high 3 bits
0x4035	OFF MAN010	0x00	RW	Bit[7:0]: off_man010[7:0] off_man010 low 8 bits
0x4036	OFF MAN011	0x00	RW	Bit[7:3]: Not used Bit[2:0]: off_man011[10:8] off_man011 high 3 bits
0x4037	OFF MAN011	0x00	RW	Bit[7:0]: off_man011[7:0] off_man011 low 8 bits

table 5-10 BLC registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x4040	BLC OFFSET000	–	R	Bit[7:3]: Not used Bit[2:0]: blc_offset000[10:8] blc_offset000 high 3 bits
0x4041	BLC OFFSET000	–	R	Bit[7:0]: blc_offset000[7:0] blc_offset000 low 8 bits
0x4042	BLC OFFSET001	–	R	Bit[7:3]: Not used Bit[2:0]: blc_offset001[10:8] blc_offset001 high 3 bits
0x4043	BLC OFFSET001	–	R	Bit[7:0]: blc_offset001[7:0] blc_offset001 low 8 bits
0x4044	BLC OFFSET010	–	R	Bit[7:3]: Not used Bit[2:0]: blc_offset010[10:8] blc_offset010 high 3 bits
0x4045	BLC OFFSET010	–	R	Bit[7:0]: blc_offset010[7:0] blc_offset010 low 8 bits
0x4046	BLC OFFSET011	–	R	Bit[7:3]: Not used Bit[2:0]: blc_offset011[10:8] blc_offset011 high 3 bits
0x4047	BLC OFFSET011	–	R	Bit[7:0]: blc_offset011[7:0] blc_offset011 low 8 bits

5.11 frame control [0x4200 - 0x4207]

table 5-11 frame control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4200	R0	0x00	RW	Bit[2]: fcnt_eof_sel Bit[1]: fcnt_mask_dis Bit[0]: fcnt_reset
0x4201	R1	0x00	RW	Bit[3:0]: frame_on_number
0x4202	R2	0x00	RW	Bit[3:0]: frame_off_number

table 5-11 frame control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4203	R3	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
0x4207	R7	–	R	Bit[7:0]: frame_counter

5.12 format [0x4300 - 0x430D]

table 5-12 format registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4300	CTRL00	0x30	RW	Bit[7:4]: output_format 0x0: raw_en 0x1: y8_en 0x2: yuv444_en/rgb888_en 0x3: yuv422_en 0x4: yuv420_en 0x5: yuv420_legacy_en 0xF: byp_fmt Bit[3:0]: pix_order_ctrl
0x4301	CTRL01	0x40	RW	Bit[1:0]: yuv422_uv_ctrl 00: U/V generate from average 01: U/V generate from first pixel 11: U/V generate from second pixel
0x4302	YMAX	0x03	RW	Bit[7:2]: Not used Bit[1:0]: ymax[9:8]
0x4303	YMAX	0xFF	RW	Bit[7:0]: ymax[7:0]
0x4304	YMIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: ymin[9:8]
0x4305	YMIN	0x00	RW	Bit[7:0]: ymin[7:0]
0x4306	UMAX	0x03	RW	Bit[7:2]: Not used Bit[1:0]: umax[9:8]
0x4307	UMAX	0xFF	RW	Bit[7:0]: umax[7:0]
0x4308	UMIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: umin[9:8]
0x4309	UMIN	0x00	RW	Bit[7:0]: umin[7:0]

table 5-12 format registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x430A	VMAX	0x03	RW	Bit[7:2]: Not used Bit[1:0]: vmax[9:8]
0x430B	VMAX	0xFF	RW	Bit[7:0]: vmax[7:0]
0x430C	VMIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: vmin[9:8]
0x430D	VMIN	0x00	RW	Bit[7:0]: vmin[7:0]

5.13 CADC sync [0x4500 - 0x4504]

table 5-13 CADC sync registers

address	register name	default value	R/W	description
0x4500~ 0x4504	CADC SYNC	–	–	CADC Sync Registers

5.14 VFIFO [0x4600 - 0x4604]

table 5-14 VFIFO registers

address	register name	default value	R/W	description
0x4600	R VFIFO READ START	0x00	RW	Bit[7:0]: r_vfifo_read_start[15:8] read_start size high byte
0x4601	R VFIFO READ START	0x10	RW	Bit[7:0]: r_vfifo_read_start[7:0] read_start size low byte
0x4602	R2	0x00	RW	Bit[7:4]: r_rm Bit[3]: r_test1 Bit[2]: Not used Bit[1]: Frame reset enable Bit[0]: RAM bypass enable
0x4603	R3	0x01	RW	Bit[4]: man_start_mode Bit[1:0]: start_offseet
0x4604	R4	–	R	Bit[3]: ram_full Bit[2]: ram_empty Bit[1]: fo_full Bit[0]: fo_empty

5.15 MIPI control [0x4800 - 0x484F]

table 5-15 MIPI control registers (sheet 1 of 8)

address	register name	default value	R/W	description
0x4800	MIPI CTRL00	0x04	RW	Bit[7:6]: Not used Bit[5]: gate_sc_en 0: Clock lane is free running 1: Gate clock lane when there is no packet to transmit Bit[4]: line_sync_en 0: Do not send line short packet for each line 1: Send line short packet for each line Bit[2:0]: Not used
0x4801	MIPI CTRL01	0x00	RW	Bit[6]: spkt_dt_sel 1: Use dt_spkt as short packet data Bit[5]: first_bit Change clk_lane first bit 0: Output 8'h05 1: Output 8'hAA Bit[4]: Not used Bit[1]: LPX_select for pclk domain 0: Auto calculate t_lpx_p, unit pclk2x cycle 1: Use lpx_p_min[7:0] Bit[0]: Not used

table 5-15 MIPI control registers (sheet 2 of 8)

address	register name	default value	R/W	description
0x4802	MIPI CTRL02	0x00	RW	Bit[7]: hs_prepare_sel 0: Auto calculate T_hs_prepare, unit pclk2x 1: Use hs_prepare_min_o[7:0] Bit[6]: clk_prepare_sel 0: Auto calculate T_clk_prepare, unit pclk2x 1: Use clk_prepare_min_o[7:0] Bit[5]: clk_post_sel 0: Auto calculate T_clk_post, unit pclk2x 1: Use clk_post_min_o[7:0] Bit[4]: clk_trail_sel 0: Auto calculate T_clk_trail, unit pclk2x 1: Use clk_trail_min_o[7:0] Bit[3]: hs_exit_sel 0: Auto calculate T_hs_exit, unit pclk2x 1: Use hs_exit_min_o[7:0] Bit[2]: hs_zero_sel 0: Auto calculate T_hs_zero, unit pclk2x 1: Use hs_zero_min_o[7:0] Bit[1]: hs_trail_sel 0: Auto calculate T_hs_trail, unit pclk2x 1: Use hs_trail_min_o[7:0] Bit[0]: clk_zero_sel 0: Auto calculate T_clk_zero, unit pclk2x 1: Use clk_zero_min_o[7:0]
0x4803	MIPI CTRL03	0x00	RW	Bit[3]: manu_ofset_o t_perio manual offset Bit[2]: r_manu_half2one t_period half to 1 Bit[1:0]: Not used
0x4804	MIPI CTRL04	0x04	RW	Bit[7:4]: man_lane_num Bit[3]: lane_num_manual_enable Bit[2]: lane4_6b_en 1: Support 4, 7, 8 lane 6-bit Bit[1:0]: Not used
0x4805	MIPI CTRL05	0x00	RW	Bit[3]: lpda_retim_manu_o Bit[2]: lpda_retim_sel_o 1: Manual Bit[1]: lpck_retim_manu_o Bit[0]: lpck_retim_sel_o 1: Manual

table 5-15 MIPI control registers (sheet 3 of 8)

address	register name	default value	R/W	description
0x4806	MIPI CTRL06	0x10	RW	Bit[4]: pu_mark_en_o Power up mark1 enable Bit[3]: mipi_remot_rst Bit[2]: mipi_susp Bit[1]: smia_lane_ch_en Bit[0]: tx_lsb_first 0: Transmit high bit first 1: Low power transmit low bit first
0x4807	MIPI CTRL07	0x03	RW	Bit[3:0]: sw_t_lpx ul_tx T_lpx
0x4808	MIPI CTRL08	0x18	RW	Bit[7:0]: wkup_dly Mark1 wakeup delay/2^10
0x4810	FCNT MAX	0xFF	RW	Bit[7:0]: fcnt_max[15:8] High byte of max frame counter of frame sync short packet
0x4811	FCNT MAX	0xFF	RW	Bit[7:0]: fcnt_max[7:0] Low byte of max frame counter of frame sync short packet
0x4813	MIPI CTRL13	0x00	RW	Bit[2]: vc_sel Input VC or reg VC Bit[1:0]: VC Virtual channel of MIPI
0x4814	MIPI CTRL14	0x2A	RW	Bit[6]: lpkt_dt_sel 0: Use mipi_dt 1: Use dt_man_o as long packet data Bit[5:0]: dt_man Manual data type
0x4815	MIPI CTRL15	0x00	RW	Bit[6]: pclk_inv 0: Using falling edge of mipi_pclk_o to generate MIPI bus to PHY 1: Using rising edge of mipi_pclk_o to generate MIPI bus to PHY Bit[5:0]: manu_dt_short Manual type for short packet
0x4816	EMB DT CTRL	0x52	RW	Bit[6]: Not used Bit[5:0]: emb_dt Manually set embedded data type
0x4817	YUV420 FUN	0x00	RW	Bit[6]: yuv420_2x YUV420 2x in odd line, lcnt[0]=1 1: Use emb_dt as data in first emb_line_nu Bit[0]: yuv420_en

table 5-15 MIPI control registers (sheet 4 of 8)

address	register name	default value	R/W	description
0x4818	HS ZERO MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: hs_zero_min[9:8] High byte of minimum value of hs_zero, unit ns
0x4819	HS ZERO MIN	0x8B	RW	Bit[7:0]: hs_zero_min[7:0] Low byte of minimum value of hs_zero hs_zero_real = hs_zero_min_o + Tui*ui_hs_zero_min_o
0x481A	HS TRAIL MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: hs_trail_min[9:8] High byte of minimum value of hs_trail, unit ns
0x481B	HS TRAIL MIN	0x3C	RW	Bit[7:0]: hs_trail_min[7:0] Low byte of minimum value of hs_trail hs_trail_real = hs_trail_min_o + Tui*ui_hs_trail_min_o
0x481C	CLK ZERO MIN	0x01	RW	Bit[7:2]: Not used Bit[1:0]: clk_zero_min[9:8] High byte of minimum value of clk_zero, unit ns
0x481D	CLK ZERO MIN	0x2C	RW	Bit[7:0]: clk_zero_min[7:0] Low byte of minimum value of clk_zero clk_zero_real = clk_zero_min_o + Tui*ui_clk_zero_min_o
0x481E	CLK PREPARE MAX	0x5F	RW	Bit[7:0]: clk_prepare_max[7:0] Maximum value of clk_prepare, unit ns
0x481F	CLK PREPARE MIN	0x26	RW	Bit[7:0]: clk_prepare_min[7:0] Minimum value of clk_prepare clk_prepare_real = clk_prepare_min_o + Tui*ui_clk_prepare_min_o
0x4820	CLK POST MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: clk_post_min[9:8] High byte of minimum value of clk_post, unit ns
0x4821	CLK POST MIN	0x3C	RW	Bit[7:0]: clk_post_min[7:0] Low byte of minimum value of clk_post clk_post_real = clk_post_min_o + Tui*ui_clk_post_min_o
0x4822	CLK TRAIL MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: clk_trail_min[9:8] High byte of minimum value of clk_trail, unit ns

table 5-15 MIPI control registers (sheet 5 of 8)

address	register name	default value	R/W	description
0x4823	CLK TRAIL MIN	0x3C	RW	Bit[7:0]: clk_trail_min[7:0] Low byte of minimum value of clk_trail clk_trail_real = clk_trail_min_o + Tui*ui_clk_trail_min_o
0x4824	LPX P MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: lpx_p_min[9:8] High byte of minimum value of lpx_p, unit ns
0x4825	LPX P MIN	0x32	RW	Bit[7:0]: lpx_p_min[7:0] Low byte of minimum value of lpx_p lpx_p_real = lpx_p_min_o + Tui*ui_lpx_p_min_o
0x4826	HS PREPARE MIN	0x36	RW	Bit[7:0]: hs_prepare_min[7:0] Minimum value of hs_prepare, unit ns
0x4827	HS PREPARE MAX	0x55	RW	Bit[7:0]: hs_prepare_max[7:0] Maximum value of hs_prepare hs_prepare_real = hs_prepare_max_o + Tui*ui_hs_prepare_max_o
0x4828	HS EXIT MIN	0x00	RW	Bit[7:2]: Not used Bit[1:0]: hs_exit_min[9:8] High byte of minimum value of hs_exit, unit ns
0x4829	HS EXIT MIN	0x64	RW	Bit[7:0]: hs_exit_min[7:0] Low byte of minimum value of hs_exit hs_exit_real = hs_exit_min_o + Tui*ui_hs_exit_min_o
0x482A	UI HS ZERO MIN	0x06	RW	Bit[7:6]: Not used Bit[5:0]: ui_hs_zero_min[5:0] Minimum UI value of hs_zero, unit UI
0x482B	UI HS TRAIL MIN	0x04	RW	Bit[7:6]: Not used Bit[5:0]: ui_hs_trail_min[5:0] Minimum UI value of hs_trail, unit UI
0x482C	UI CLK ZERO MIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: ui_clk_zero_min[5:0] Minimum UI value of clk_zero, unit UI
0x482D	UI CLK PREPARE	0x00	RW	Bit[7:4]: ui_clk_prepare_max Maximum UI value of clk_prepare, unit UI Bit[3:0]: ui_clk_prepare_min Minimum UI value of clk_prepare, unit UI
0x482E	UI CLK POST MIN	0x34	RW	Bit[7:6]: Not used Bit[5:0]: ui_clk_post_min[5:0] Minimum UI value of clk_post, unit UI

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

address	register name	default value	R/W	description
0x482F	UI CLK TRAIL MIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: ui_clk_trail_min[5:0] Minimum UI value of clk_trail, unit UI
0x4830	UI LPX P MIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: ui_lpx_p_min[5:0] Minimum UI value of lpx_p(pclk2x domain), unit UI
0x4831	UI HS PREPARE	0x64	RW	Bit[7:4]: ui_hs_prepare_max Maximum UI value of hs_prepare, unit UI Bit[3:0]: ui_hs_prepare_min Minimum UI value of hs_prepare, unit UI
0x4832	UI HS EXIT MIN	0x00	RW	Bit[7:6]: Not used Bit[5:0]: ui_hs_exit_min[5:0] Minimum UI value of hs_exit, unit UI
0x4833	CTRL51	0x18	RW	Bit[7:0]: Ctrl51
0x4836	GLB MODE SEL	0x00	RW	Bit[0]: smia_cal_en 0: Use period to calculate 1: Use SMIA bit rate to calculate
0x4837	PCLK PERIOD	0x18	RW	Bit[7:0]: pclk_period[7:0] Period of pclk2x, pclk_div=1, and 1-bit decimal
0x4838	MIPI LP GPIO0	0x00	RW	Bit[7]: lp_sel0 0: Auto generate mipi_lp_dir0_o 1: Use lp_dir_man0 to be mipi_lp_dir0_o Bit[6]: lp_dir_man0 0: Input 1: Output Bit[5]: lp_p0_o Bit[4]: lp_n0_o Bit[3:0]: Not used

table 5-15 MIPI control registers (sheet 7 of 8)

address	register name	default value	R/W	description
0x4839	MIPI LP GPIO1	0x00	RW	Bit[7]: lp_sel2 0: Auto generate mipi_lp_dir2_o 1: Use lp_dir_man2 to be mipi_lp_dir2_o Bit[6]: lp_dir_man2 0: Input 1: Output Bit[5]: lp_p2_o Bit[4]: lp_n2_o Bit[3]: lp_sel3 0: Auto generate mipi_lp_dir3_o 1: Use lp_dir_man3 to be mipi_lp_dir3_o Bit[2]: lp_dir_man3 0: Input 1: Output Bit[1]: lp_p3_o Bit[0]: lp_n3_o
0x483A	MIPI LP GPIO2	0x00	RW	Bit[7]: lp_sel4 0: Auto generate mipi_lp_dir4_o 1: Use lp_dir_man4 to be mipi_lp_dir4_o Bit[6]: lp_dir_man4 0: Input 1: Output Bit[5]: lp_p4_o Bit[4]: lp_n4_o Bit[3]: lp_sel5 0: Auto generate mipi_lp_dir5_o 1: Use lp_dir_man5 to be mipi_lp_dir5_o Bit[2]: lp_dir_man5 0: Input 1: Output Bit[1]: lp_p5_o Bit[0]: lp_n5_o
0x483B	MIPI LP GPIO3	0x00	RW	Bit[7]: lp_sel6 0: Auto generate mipi_lp_dir6_o 1: Use lp_dir_man6 to be mipi_lp_dir6_o Bit[6]: lp_dir_man6 0: Input 1: Output Bit[5]: lp_p6_o Bit[4]: lp_n6_o Bit[3]: lp_sel7 0: Auto generate mipi_lp_dir7_o 1: Use lp_dir_man7 to be mipi_lp_dir7_o Bit[2]: lp_dir_man7 0: Input 1: Output Bit[1]: lp_p7_o Bit[0]: lp_n7_o

table 5-15 MIPI control registers (sheet 8 of 8)

address	register name	default value	R/W	description
0x483C	MIPI CTRL3C	0x02	RW	Bit[3:0]: t_clk_pre Unit: pclk2x cycle
0x483D	MIPI LP GPIO4	0x00	RW	Bit[7]: lp_ck_sel0 0: Auto generate mipi_ck_lp_dir0_o 1: Use lp_ck_dir_man0 to be mipi_ck_lp_dir0_o Bit[6]: lp_ck_dir_man0 0: Input 1: Output Bit[5]: lp_ck_p0_o Bit[4]: lp_ck_n0_o Bit[3:0]: Not used
0x484A	SEL MIPI CTRL4A	0x27	RW	Bit[5]: slp_lp_pon_man_o Set for power up Bit[4]: slp_lp_pon_da Bit[3]: slp_lp_pon_ck Bit[2]: mipi_slp_man_st MIPI bus status manual control enable in sleep mode Bit[1]: clk_lane_state Bit[0]: data_lane_state
0x484B	SMIA OPTION	0x07	RW	Bit[2]: line_st_sel_o 0: Line starts after HREF 1: Line starts after fifo_st Bit[1]: clk_start_sel_o 0: Clock starts after SOF 1: Clock starts after reset Bit[0]: sof_sel_o 0: Frame starts after HREF 1: Frame starts after SOF
0x484C	SEL MIPI CTRL4C	0x03	RW	Bit[6]: smia_fcnt_i select Bit[5]: prbs_enable Bit[4]: hs_test_only MIPI high speed only test mode enable Bit[3]: set_frame_cnt_0 Set frame count to inactive mode (keep 0)
0x484D	TEST PATTEN DATA	0xB6	RW	Bit[7:0]: test_patten_data[7:0] Data lane test pattern
0x484E	FE DLY	0x10	RW	Bit[7:0]: r_fe_dly_o Last packet to frame end delay / 2
0x484F	TEST PATTEN CK DATA	0x55	RW	Bit[7:0]: clk_test_patten_reg

5.16 ISP [0x5000 - 0x5015]

table 5-16 ISP registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5000	R ISP CTRL0	0x1F	RW	Bit[7]: lenc_en Bit[6]: awb_gain_en Bit[5]: lcd_en Bit[4]: avg_en Bit[3]: dgc_en Bit[2]: bc_en Bit[1]: wc_en Bit[0]: blc_en
0x5001	R ISP CTRL1	0x05	RW	Bit[7]: latch_en Bit[6:4]: win_y_offset_adjust Bit[3]: bias_man_en Bit[2:0]: avg_sel 0x0: Before BLC 0x1: After BLC 0x2: After DGC 0x3: After LENC 0x4: After AWB 0x5: After LCD 0x6: Data 0 0x7: Data 0
0x5002	R ISP CTRL2	0x20	RW	Bit[7]: aec_eof_sel Bit[6]: avg_man_en Bit[5]: dpc_href_s Bit[4]: sof_sel Bit[3:2]: eof_sel Bit[1]: bias_plus Bit[0]: offset_man_en
0x5003	BIAS MAN	0x00	RW	Bit[7:0]: bias_man[7:0]
0x5004	R ISP CTRL4	0x18	RW	Bit[7]: uv_dns_en Bit[6]: rgb_dns_en Bit[5]: gamma_en Bit[4]: cmx_en Bit[3]: cip_en Bit[2]: raw_dns_en Bit[1]: stretch_en Bit[0]: awb_en
0x5005	R ISP CTRL5	0x12	RW	Bit[4]: sde_en Bit[3]: lcd_out_en Bit[2]: raw_en Bit[1]: rgb2yuv_en Bit[0]: binc_en
0x5006	R ISP CTRL6	0x04	RW	Bit[7:0]: avg_man

table 5-16 ISP registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5007	R ISP CTRL7	0x00	RW	Bit[6]: y_swap Bit[5]: uv_swap
0x5008	R ISP CTRL8	0x50	RW	Bit[7:4]: sde_drop_sel Bit[3:0]: drop_sel
0x500A	LCD R GAIN	0x04	RW	Bit[7:4]: Not used Bit[3:0]: lcd_r_gain[11:8]
0x500B	LCD R GAIN	0x00	RW	Bit[7:0]: lcd_r_gain[7:0]
0x500C	LCD G GAIN	0x04	RW	Bit[7:4]: Not used Bit[3:0]: lcd_g_gain[11:8]
0x500D	LCD G GAIN	0x00	RW	Bit[7:0]: lcd_g_gain[7:0]
0x500E	LCD B GAIN	0x04	RW	Bit[7:4]: Not used Bit[3:0]: lcd_b_gain[11:8]
0x500F	LCD B GAIN	0x00	RW	Bit[7:0]: lcd_b_gain[7:0]
0x5010	ANA REAL GAIN	–	R	Bit[7:2]: Not used Bit[1:0]: ana_real_gain[9:8]
0x5011	ANA REAL GAIN	–	R	Bit[7:0]: ana_real_gain[7:0]
0x5012	DGC	–	R	Bit[7:2]: Not used Bit[1:0]: dgc[9:8]
0x5013	DGC	–	R	Bit[7:0]: dgc[7:0]
0x5014	CTRLE	–	R	Bit[7]: Not used Bit[6]: blk_ln_rblue_i Bit[5:0]: blk_ln_num_i
0x5015	CTRLF	–	R	Bit[7:6]: Not used Bit[5:4]: dig_gain_i Bit[3]: gain_chg_i Bit[2]: fmt_chg_i Bit[1]: mirror_i Bit[0]: flip_i

5.17 pre-ISP [0x5080 - 0x5081]

table 5-17 pre-ISP registers

address	register name	default value	R/W	description
0x5080	PRE ISP CTRL00	0x00	RW	Bit[7]: test_en Test enable Bit[6]: rolling_en Bit[5]: trans_test_mode Bit[4]: squ_en Bit[3:2]: bar_style Bit[1:0]: test_mode_sel
0x5081	PRE ISP CTRL01	0x41	RW	Bit[7]: Not used Bit[6]: cut_en Window cut enable Bit[5]: low_bits ISP test, low bits to 0 Bit[4]: random_reset Test mode, random data reset Bit[3:0]: random_seed Random seed

5.18 AWB [0x5180 - 0x519B]

table 5-18 AWB registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5180	R AWB CTRL00	0x51	RW	Bit[7:6]: local_step (inside local range) Bit[5:4]: fast_step (outside local range) Bit[3]: slop_8x Bit[2]: slop_4x Bit[1]: man_en Bit[0]: avg_all
0x5181	R AWB CTRL01	0x11	RW	Bit[7:4]: max_local_cnt Bit[3:0]: max_fast_cnt
0x5182	R AWB CTRL02	0x41	RW	Bit[7]: freeze Bit[6]: fast_enable Bit[5:4]: win_idx Bit[3:1]: min_white_pixel_sel Bit[0]: bias_stat

table 5-18 AWB registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5183	R AWB CTRL03	0x42	RW	Bit[7:4]: stable_range_w Stable to unstable Bit[3:0]: stable_range Unstable to stable
0x5184	R AWB CTRL04	0x20	RW	Bit[7:0]: x0 (CWF_X)
0x5185	R AWB CTRL05	0x20	RW	Bit[7:0]: y0 (CWF_Y)
0x5186	R AWB CTRL 06	0x00	RW	Bit[7:0]: kx (CWF to A) x2y
0x5187	R AWB CTRL 07	0x00	RW	Bit[7:0]: ky (CFW to day) y2x
0x5188	R AWB CTRL 08	0x10	RW	Bit[7:0]: cwf_range (square)
0x5189	R AWB CTRL 09	0x10	RW	Bit[7:0]: a_range
0x518A	R AWB CTRL 0A	0x10	RW	Bit[7:0]: day_range
0x518B	R AWB CTRL 0B	0x00	RW	Bit[7:0]: day_limit
0x518C	R AWB CTRL 0C	0x00	RW	Bit[7:0]: a_limit
0x518D	R AWB CTRL 0D	0xF0	RW	Bit[7:0]: top_limit
0x518E	R AWB CTRL 0E	0x10	RW	Bit[7:0]: bot_limit
0x518F	R AWB CTRL 0F	0xFF	RW	Bit[7:0]: red_gain_max
0x5190	R AWB CTRL 10	0x00	RW	Bit[7:0]: red_gain_min
0x5191	R AWB CTRL 11	0xFF	RW	Bit[7:0]: green_gain_max
0x5192	R AWB CTRL 12	0x00	RW	Bit[7:0]: green_gain_min
0x5193	R AWB CTRL 13	0xFF	RW	Bit[7:0]: blue_gain_max
0x5194	R AWB CTRL 14	0x00	RW	Bit[7:0]: blue_gain_min
0x5195	R AWB CTRL 15	0x04	RW	Bit[3:0]: awb_r_gain_m[11:8]
0x5196	R AWB CTRL 16	0x00	RW	Bit[7:0]: awb_r_gain_m[7:0]
0x5197	R AWB CTRL 17	0x04	RW	Bit[3:0]: awb_g_gain_m[11:8]
0x5198	R AWB CTRL 18	0x00	RW	Bit[7:0]: awb_g_gain_m[7:0]
0x5199	R AWB CTRL 19	0x04	RW	Bit[3:0]: awb_b_gain_m[11:8]
0x519A	R AWB CTRL 1A	0x00	RW	Bit[7:0]: awb_b_gain_m[7:0]
0x519B	R AWB CTRL 1B	0x10	RW	Bit[7:0]: local_limit

5.19 stretch [0x5200 - 0x5215]

table 5-19 stretch registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5200	STRETCH_CTRL0	0x09	RW	Bit[3:2]: Stretch minimum Real stretch minimum value is: 00: 1024 01: 2048 10: 3096 11: 8192 Bit[1]: Stretch manual enable 0: Manual disable 1: Manual enable Bit[0]: Stretch auto enable 0: Auto disable 1: Auto enable
0x5201	STRETCH_CTRL1	0x00	RW	Bit[5:0]: Stretch min low level
0x5202	STRETCH_CTRL2	0x06	RW	Bit[5:0]: Stretch max low level
0x5203	STRETCH_CTRL3	0x10	RW	Bit[4:0]: Stretch min high level
0x5204	STRETCH_CTRL4	0x41	RW	Bit[7:4]: Stretch step2 Bit[3:0]: Stretch step1
0x5205	STRETCH_CTRL5	0x10	RW	Bit[7:0]: Stretch current low level If minimum low level is 0, current low level must be greater or equal to step1
0x5206	STRETCH_CTRL6	0xFF	RW	Bit[7:0]: Stretch current high level[7:0]
0x5207	STRETCH_CTRL7	0x03	RW	Bit[3:0]: Stretch current high level[11:8]
0x5208	STRETCH_CTRL8	0x10	RW	Bit[7:0]: Stretch current low level manual value
0x5209	STRETCH_CTRL9	0xFF	RW	Bit[7:0]: Stretch current high level manual value[7:0]
0x520a	STRETCH_CTRL10	0x03	RW	Bit[3:0]: Stretch current high level manual value[11:8]
0x520b	STRETCH_CTRL11	0xC4	RW	Bit[7:0]: stretch_thres1[7:0]
0x520c	STRETCH_CTRL12	0x3B	RW	Bit[7:0]: stretch_thres1[15:8]
0x520D	STRETCH_CTRL13	0x00	RW	Bit[7:0]: stretch_thres1[23:16] stretch_thre1 = width × height × m_nPsThres1 >> 10
0x520E	STRETCH_CTRL14	0xC4	RW	Bit[7:0]: stretch_thres2[7:0]
0x520F	STRETCH_CTRL15	0x3B	RW	Bit[7:0]: stretch_thres2[15:8]

table 5-19 stretch registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5210	STRETCH_CTRL16	0x00	RW	Bit[7:0]: stretch_thres2[23:16] stretch_thre2 = width × height × m_nPsThres2>>10
0x5211	STRETCH_CTRL17	–	R	Stretch Current Low Level
0x5212	STRETCH_CTRL18	–	R	Bit[7:0]: Stretch current high level[7:0]
0x5213	STRETCH_CTRL19	–	R	Bit[3:0]: Stretch current high level[11:8]
0x5214	STRETCH_CTRL20	–	R	Bit[7:0]: Stretch gain[7:0]
0x5215	STRETCH_CTRL21	–	R	Bit[3:0]: Stretch gain[11:8]

5.20 binning_correction [0x5580]

table 5-20 binning_correction register

address	register name	default value	R/W	description
0x5580	BINC_CTRL0	0x01	RW	Bit[2]: m_nBinningYFlag Bit[1]: m_nBinningXFlag Bit[0]: m_bHBinEnable

5.21 raw_DNS [0x5280 - 0x528B]

table 5-21 raw_DNS registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5280	RAW_DNS_CTRL_00	0x08	RW	Bit[5:2]: m_nNoiseYSlope Bit[1]: raw_dns manual mode 0: Parameter noise and edgethre calculated by NoiseList 1: Parameter noise and edgethre configured directly by register Bit[0]: m_bGDNSEnable
0x5281	RAW_DNS_CTRL_01	0x04	RW	Bit[7:0]: m_nNoiseList[0]
0x5282	RAW_DNS_CTRL_02	0x08	RW	Bit[7:0]: m_nNoiseList[1]

table 5-21 raw_DNS registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5283	RAW_DNS_CTRL_03	0x10	RW	Bit[7:0]: m_nNoiseList[2]
0x5284	RAW_DNS_CTRL_04	0x18	RW	Bit[7:0]: m_nNoiseList[3]
0x5285	RAW_DNS_CTRL_05	0x20	RW	Bit[7:0]: m_nNoiseList[4]
0x5286	RAW_DNS_CTRL_06	0x30	RW	Bit[7:0]: m_nNoiseList[5]
0x5287	RAW_DNS_CTRL_07	0x3F	RW	Bit[7:0]: m_nMaxEdgeThre
0x5288	RAW_DNS_CTRL_08	0x08	RW	Bit[7:0]: Raw DNS noise parameter in manual mode
0x5289	RAW_DNS_CTRL_09	0x18	RW	Bit[7:0]: Raw DNS edgethre parameter in manual mode
0x528A	RAW_DNS_CTRL_10	–	R	Bit[7:0]: Noise value
0x528B	RAW_DNS_CTRL_11	–	R	Bit[7:0]: Edgethre value

5.22 CIP [0x5300 - 0x531B]

table 5-22 CIP registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5300	CIP_CTRL_00	0xE2	RW	Bit[7]: m_bColorEdgeEnable Bit[6]: m_bAntiAliasing Bit[5:3]: m_nDetailSlope Bit[2:0]: m_nNoiseYSlope
0x5301	CIP_CTRL_01	0xA0	RW	Bit[6:5]: m_nSharpenSlope Bit[4:1]: m_nGbGrShift Bit[0]: m_bGbGrRemove
0x5302	CIP_CTRL_02	0x00	RW	Bit[7:0]: m_nNoiseList[0]
0x5303	CIP_CTRL_03	0x00	RW	Bit[7:0]: m_nNoiseList[1]
0x5304	CIP_CTRL_04	0x00	RW	Bit[7:0]: m_nNoiseList[2]
0x5305	CIP_CTRL_05	0x00	RW	Bit[7:0]: m_nNoiseList[3]
0x5306	CIP_CTRL_06	0x00	RW	Bit[7:0]: m_nNoiseList[4]
0x5307	CIP_CTRL_07	0x00	RW	Bit[7:0]: m_nNoiseList[5]
0x5308	CIP_CTRL_08	0x82	RW	Bit[7:3]: m_nMaxSharpenGain Bit[2:0]: m_nMinSharpenGain

table 5-22 CIP registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5309	CIP_CTRL_09	0x00	RW	Bit[5:0]: m_nMinSharpen
0x530A	CIP_CTRL_0A	0x0C	RW	Bit[5:0]: m_nMaxSharpen
0x530B	CIP_CTRL_0B	0x08	RW	Bit[7:0]: m_nMinDetail
0x530C	CIP_CTRL_0C	0x0C	RW	Bit[7:0]: m_nMaxDetail
0x530D	CIP_CTRL_0D	0x00	RW	Bit[5:0]: m_nDetailRatioList[0]
0x530E	CIP_CTRL_0E	0x0C	RW	Bit[5:0]: m_nDetailRatioList[1]
0x530F	CIP_CTRL_0F	0x14	RW	Bit[5:0]: m_nDetailRatioList[2]
0x5310	CIP_CTRL_10	0x10	RW	Bit[5:0]: m_nSharpenNegEdgeRatio
0x5311	CIP_CTRL_11	0x20	RW	Bit[7:0]: m_nClrEdgeShpT1
0x5312	CIP_CTRL_12	0x80	RW	Bit[7:0]: m_nClrEdgeShpT2
0x5313	CIP_CTRL_13	0x4B	RW	Bit[7:0]: m_nClrEdgeShpSlope
0x5314	CIP_CTRL_14	0x00	RW	Bit[6:1]: cip_sharpen_p_man Bit[0]: cip_para_man_en
0x5315	CIP_CTRL_15	0x00	RW	Bit[5:0]: cip_sharpen_m_man
0x5316	CIP_CTRL_16	0x00	RW	Bit[7:0]: cip_noise_man
0x5317	CIP_CTRL_17	0x00	RW	Bit[7:0]: cip_detail_man
0x5318	CIP_CTRL_18	–	R	Bit[7:0]: m_nNoise
0x5319	CIP_CTRL_19	–	R	Bit[7:0]: m_nDetail
0x531A	CIP_CTRL_1A	–	R	Bit[7:0]: m_nSharpenP
0x531B	CIP_CTRL_1B	–	R	Bit[7:0]: m_nSharpenM

5.23 CMX [0x5380 - 0x538C]

table 5-23 CMX registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5380	CMX_CTRL_00	0x00	RW	Bit[0]: Absolute value of nCCM_D[0][0] high byte
0x5381	CMX_CTRL_01	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[0][0] low byte

table 5-23 CMX registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5382	CMX_CTRL_02	0x00	RW	Bit[0]: Absolute value of nCCM_D[0][1] high byte
0x5383	CMX_CTRL_03	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[0][1] low byte
0x5384	CMX_CTRL_04	0x00	RW	Bit[0]: Absolute value of nCCM_D[1][0] high byte
0x5385	CMX_CTRL_05	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[1][0] low byte
0x5386	CMX_CTRL_06	0x00	RW	Bit[0]: Absolute value of nCCM_D[1][1] high byte
0x5387	CMX_CTRL_07	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[1][1] low byte
0x5388	CMX_CTRL_08	0x00	RW	Bit[0]: Absolute value of nCCM_D[2][0] high byte
0x5389	CMX_CTRL_09	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[2][0] low byte
0x538A	CMX_CTRL_0A	0x00	RW	Bit[0]: Absolute value of nCCM_D[2][1] high byte
0x538B	CMX_CTRL_0B	0x00	RW	Bit[7:0]: Absolute value of nCCM_D[2][1] low byte
0x538C	CMX_CTRL_0C	0x00	RW	Bit[5]: Sign bit of nCCM_D[2][1] Bit[4]: Sign bit of nCCM_D[2][0] Bit[3]: Sign bit of nCCM_D[1][1] Bit[2]: Sign bit of nCCM_D[1][0] Bit[1]: Sign bit of nCCM_D[0][1] Bit[0]: Sign bit of nCCM_D[0][0]

5.24 gamma [0x5400 - 0x5411]

table 5-24 gamma registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5400	GAMMA_00	0x04	RW	Bit[7:0]: m_pCurveYList[0]
0x5401	GAMMA_01	0x08	RW	Bit[7:0]: m_pCurveYList[1]
0x5402	GAMMA_02	0x10	RW	Bit[7:0]: m_pCurveYList[2]

table 5-24 gamma registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5403	GAMMA_03	0x20	RW	Bit[7:0]: m_pCurveYList[3]
0x5404	GAMMA_04	0x28	RW	Bit[7:0]: m_pCurveYList[4]
0x5405	GAMMA_05	0x30	RW	Bit[7:0]: m_pCurveYList[5]
0x5406	GAMMA_06	0x38	RW	Bit[7:0]: m_pCurveYList[6]
0x5407	GAMMA_07	0x40	RW	Bit[7:0]: m_pCurveYList[7]
0x5408	GAMMA_08	0x48	RW	Bit[7:0]: m_pCurveYList[8]
0x5409	GAMMA_09	0x50	RW	Bit[7:0]: m_pCurveYList[9]
0x540A	GAMMA_0A	0x60	RW	Bit[7:0]: m_pCurveYList[10]
0x540B	GAMMA_0B	0x70	RW	Bit[7:0]: m_pCurveYList[11]
0x540C	GAMMA_0C	0x90	RW	Bit[7:0]: m_pCurveYList[12]
0x540D	GAMMA_0D	0xB0	RW	Bit[7:0]: m_pCurveYList[13]
0x540E	GAMMA_0E	0xD0	RW	Bit[7:0]: m_pCurveYList[14]
0x540F	GAMMA_0F	0xFF	RW	Bit[7:0]: m_nMaxShadowHGain
0x5410	GAMMA_10	0x6E	RW	Bit[7:0]: m_nMidToneHGain
0x5411	GAMMA_11	0x10	RW	Bit[7:0]: m_nHighLightHGain

5.25 RGB_DNS [0x5480 - 0x548D]

table 5-25 RGB_DNS registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5480	RGB_DNS_CTRL00	0x19	RW	Bit[4:1]: m_nShadowExtraNoise Bit[0]: m_bSmoothYEnable
0x5481	RGB_DNS_CTRL01	0x08	RW	Bit[5:3]: m_nNoiseYList[1] Bit[2:0]: m_nNoiseYList[0]
0x5482	RGB_DNS_CTRL02	0x1A	RW	Bit[5:3]: m_nNoiseYList[3] Bit[2:0]: m_nNoiseYList[2]
0x5483	RGB_DNS_CTRL03	0x24	RW	Bit[5:3]: m_nNoiseYList[5] Bit[2:0]: m_nNoiseYList[4]
0x5484	RGB_DNS_CTRL04	0x04	RW	Bit[7:0]: m_nNoiseUVList[0]
0x5485	RGB_DNS_CTRL05	0x06	RW	Bit[7:0]: m_nNoiseUVList[1]

table 5-25 RGB_DNS registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5486	RGB_DNS_CTRL06	0x08	RW	Bit[7:0]: m_nNoiseUVList[2]
0x5487	RGB_DNS_CTRL07	0x0C	RW	Bit[7:0]: m_nNoiseUVList[3]
0x5488	RGB_DNS_CTRL08	0x10	RW	Bit[7:0]: m_nNoiseUVList[4]
0x5489	RGB_DNS_CTRL09	0x18	RW	Bit[7:0]: m_nNoiseUVList[5]
0x548A	RGB_DNS_CTRL0A	0x00	RW	Bit[3:1]: rgb_dns_y_noise_man Bit[0]: rgb_dns_para_man_en
0x548B	RGB_DNS_CTRL0B	0x00	RW	Bit[7:0]: rgb_dns_uv_noise_man
0x548C	RGB_DNS_CTRL0C	–	R	Bit[7:0]: m_nNoiseY
0x548D	RGB_DNS_CTRL0D	–	R	Bit[7:0]: m_nNoiseUV

5.26 UV_DNS [0x5500 - 0x5508]

table 5-26 UV_DNS registers

address	register name	default value	R/W	description
0x5500	UV_DNS_CTRL0	0x01	RW	Bit[7:0]: m_nNoiseList[0]
0x5501	UV_DNS_CTRL1	0x02	RW	Bit[7:0]: m_nNoiseList[1]
0x5502	UV_DNS_CTRL2	0x04	RW	Bit[7:0]: m_nNoiseList[2]
0x5503	UV_DNS_CTRL3	0x08	RW	Bit[7:0]: m_nNoiseList[3]
0x5504	UV_DNS_CTRL4	0x0C	RW	Bit[7:0]: m_nNoiseList[4]
0x5505	UV_DNS_CTRL5	0x10	RW	Bit[7:0]: m_nNoiseList[5]
0x5506	UV_DNS_CTRL6	0x06	RW	Bit[4]: uv_dns_para_man_en Bit[3:0]: m_nShadowExtraNoise
0x5507	UV_DNS_CTRL7	0x00	RW	Bit[7:0]: uv_dns_noise_man
0x5508	UV_DNS_CTRL8	–	R	Bit[7:0]: uv_dns_noise

5.27 SDE [0x5600 - 0x560C]

table 5-27 SDE registers

address	register name	default value	R/W	description
0x5600	R SDE CTRL0	0x00	RW	Bit[7]: fixy_en Bit[6]: neg_en Bit[5]: gray_en Bit[4]: fix_v_en Bit[3]: fix_u_en Bit[2]: contrast_en Bit[1]: satuation_en
0x5603	R SDE CTRL3	0x40	RW	Bit[7:0]: sat_u or ureg
0x5604	R SDE CTRL4	0x00	RW	Bit[7:0]: sat_v or vreg
0x5605	R SDE CTRL5	0x00	RW	Bit[7:0]: Yoffset
0x5606	R SDE CTRL6	0x20	RW	Bit[7:0]: Ygain
0x5607	R SDE CTRL7	0x00	RW	Bit[7:0]: Ybright
0x5608	R SDE CTRL8	0x00	RW	Bit[7:6]: Not used Bit[5:0]: Signset
0x5609	R SDE CTRL9	0x08	RW	Bit[7:0]: uvadj_th1
0x560A	R SDE CTRLA	0x80	RW	Bit[7:0]: uvadj_th2
0x560B	R SDE CTRLB	0x00	RW	Bit[7:1]: Not used Bit[0]: uvadj_man_en
0x560C	R SDE CTRLC	–	R	Bit[7:0]: uvadj_man_i

5.28 window [0x5700 - 0x570C]

table 5-28 window registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5700	XSTART	0x00	RW	Bit[7:5]: Not used Bit[4:0]: xstart[12:8]
0x5701	XSTART	0x00	RW	Bit[7:0]: xstart[7:0]
0x5702	YSTART	0x00	RW	Bit[7:4]: Not used Bit[3:0]: ystart[11:8]
0x5703	YSTART	0x00	RW	Bit[7:0]: ystart[7:0]

table 5-28 window registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5704	X WIN	0x10	RW	Bit[7:5]: Not used Bit[4:0]: x_win[12:8]
0x5705	X WIN	0xA0	RW	Bit[7:0]: x_win[7:0]
0x5706	Y WIN	0x0C	RW	Bit[7:4]: Not used Bit[3:0]: y_win[11:8]
0x5707	Y WIN	0x78	RW	Bit[7:0]: y_win[7:0]
0x5708	CONTROL	0x00	RW	Bit[2]: flip_offset_en Bit[1]: mirror_offset_en Bit[0]: win_man_en
0x5709	PX CNT	–	R	Bit[7:5]: Not used Bit[4:0]: px_cnt[12:8]
0x570A	PX CNT	–	R	Bit[7:0]: px_cnt[7:0]
0x570B	LN CNT	–	R	Bit[7:4]: Not used Bit[3:0]: ln_cnt[11:8]
0x570C	LN CNT	–	R	Bit[7:0]: ln_cnt[7:0]

5.29 DPC [0x5780 - 0x579D]

table 5-29 DPC registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5780	DPC CTRL0	0x14	RW	Bit[5]: enable_tail Bit[4]: enable_saturate_crosscluster Bit[3]: enable_3x3_cluster Bit[2]: enable_crosscluster Bit[1]: enable_general_tail Bit[0]: manual_mode_en
0x5781	DPC CTRL1	0x0F	RW	Bit[7:4]: Saturate Bit[3]: enable_diffchannel_wpconn Bit[2]: enable_diffchannel_bpconn Bit[1]: enable_samechannel_wpconn Bit[0]: enable_samechannel_bpconn
0x5782	WTHRE LIST0	0x04	RW	Bit[7:4]: Not used Bit[3:0]: wthre_list0[3:0]
0x5783	WTHRE LIST1	0x02	RW	Bit[7:4]: Not used Bit[3:0]: wthre_list1[3:0]

table 5-29 DPC registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x5784	WTHRE LIST2	0x01	RW	Bit[7:4]: Not used Bit[3:0]: wthre_list2[3:0]
0x5785	WTHRE LIST3	0x01	RW	Bit[7:4]: Not used Bit[3:0]: wthre_list3[3:0]
0x5786	ADPTIVE PATTERN THRE	0x00	RW	Bit[7:4]: Not used Bit[3:0]: adptive_pattern_thre[3:0]
0x5787	ADPTIVE PATTERN STEP	0x04	RW	Bit[7:4]: Not used Bit[3:0]: adptive_pattern_step[3:0]
0x5788	MORE CONNECTION CASE THRE	0x0C	RW	Bit[7:4]: Not used Bit[3:0]: more_connection_case_thre[3:0]
0x5789	DPC LEVEL LIST0	0x00	RW	Bit[7:2]: Not used Bit[1:0]: dpc_level_list0[1:0]
0x578A	DPC LEVEL LIST1	0x01	RW	Bit[7:2]: Not used Bit[1:0]: dpc_level_list1[1:0]
0x578B	DPC LEVEL LIST2	0x02	RW	Bit[7:2]: Not used Bit[1:0]: dpc_level_list2[1:0]
0x578C	DPC LEVEL LIST3	0x03	RW	Bit[7:2]: Not used Bit[1:0]: dpc_level_list3[1:0]
0x578D	GAIN LIST0	0x03	RW	Bit[7]: Not used Bit[6:0]: gain_list0[6:0]
0x578E	GAIN LIST1	0x0F	RW	Bit[7]: Not used Bit[6:0]: gain_list1[6:0]
0x578F	GAIN LIST2	0x3F	RW	Bit[7]: Not used Bit[6:0]: gain_list2[6:0]
0x5790	MATCHING THRE	0x08	RW	Bit[7:4]: Not used Bit[3:0]: matching_thre[3:0]
0x5791	STATUS THRE	0x04	RW	Bit[7:4]: Not used Bit[3:0]: status_thre[3:0]
0x5792	THRE RATIO	0x04	RW	Bit[7:4]: Not used Bit[3:0]: thre_ratio[3:0]
0x5793	CTRL19	0x00	RW	Bit[7:0]: Ctrl19
0x5794	CTRL20	0x03	RW	Bit[7:0]: Ctrl20
0x5797	BTHRE	–	R	Bit[7]: Not used Bit[6:0]: bthre[6:0]
0x5798	WTHRE	–	R	Bit[7:5]: Not used Bit[4:0]: wthre[4:0]

table 5-29 DPC registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x5799	THRE1	–	R	Bit[7:5]: Not used Bit[4:0]: thre1[4:0]
0x579A	THRE2	–	R	Bit[7:6]: Not used Bit[5:0]: thre2[5:0]
0x579B	THRE3	–	R	Bit[7]: Not used Bit[6:0]: thre3[6:0]
0x579C	THRE4	–	R	Bit[7:5]: Not used Bit[4:0]: thre4[4:0]
0x579D	LEVEL	–	R	Bit[7:4]: Not used Bit[3:0]: level[3:0]

5.30 LENC [0x5800 - 0x581C]

table 5-30 LENC registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5800	RED X0	0x03	RW	Bit[7:3]: Not used Bit[2:0]: red_x0[10:8] Red x0 high byte
0x5801	RED X0	0x28	RW	Bit[7:0]: red_x0[7:0] Red x0 low byte
0x5802	RED Y0	0x02	RW	Bit[7:3]: Not used Bit[2:0]: red_y0[10:8] Red y0 high byte
0x5803	RED Y0	0x60	RW	Bit[7:0]: red_y0[7:0] Red y0 low byte
0x5804	RED A1	0x22	RW	Bit[7]: Not used Bit[6:0]: red_a1[6:0] Red a1
0x5805	RED A2	0x07	RW	Bit[7:4]: Not used Bit[3:0]: red_a2[3:0] Red a2
0x5806	RED B1	0xC2	RW	Bit[7:0]: red_b1[7:0] Red b1
0x5807	RED B2	0x08	RW	Bit[7:4]: Not used Bit[3:0]: red_b2[3:0] Red b2

table 5-30 LENC registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x5808	GRN X0	0x03	RW	Bit[7:3]: Not used Bit[2:0]: grn_x0[10:8] Green x0 high byte
0x5809	GRN X0	0x28	RW	Bit[7:0]: grn_x0[7:0] Green x0 low byte
0x580A	GRN Y0	0x02	RW	Bit[7:3]: Not used Bit[2:0]: grn_y0[10:8] Green y0 high byte
0x580B	GRN Y0	0x60	RW	Bit[7:0]: grn_y0[7:0] Green y0 low byte
0x580C	GRN A1	0x22	RW	Bit[7]: Not used Bit[6:0]: grn_a1[6:0] Green a1
0x580D	GRN A2	0x07	RW	Bit[7:4]: Not used Bit[3:0]: grn_a2[3:0] Green a2
0x580E	GRN B1	0xC2	RW	Bit[7:0]: grn_b1[7:0] Green b1
0x580F	GRN B2	0x08	RW	Bit[7:4]: Not used Bit[3:0]: grn_b2[3:0] Green b2
0x5810	BLU X0	0x03	RW	Bit[7:3]: Not used Bit[2:0]: blu_x0[10:8] Blue x0 high byte
0x5811	BLU X0	0x28	RW	Bit[7:0]: blu_x0[7:0] Blue x0 low byte
0x5812	BLU Y0	0x02	RW	Bit[7:3]: Not used Bit[2:0]: blu_y0[10:8] Blue y0 high byte
0x5813	BLU Y0	0x60	RW	Bit[7:0]: blu_y0[7:0] Blue y0 low byte
0x5814	BLU A1	0x22	RW	Bit[7]: Not used Bit[6:0]: blu_a1[6:0] Blue a1
0x5815	BLU A2	0x07	RW	Bit[7:4]: Not used Bit[3:0]: blu_a2[3:0] Blue a2
0x5816	BLU B1	0xC2	RW	Bit[7:0]: blu_b1[7:0] Blue b1

table 5-30 LENC registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x5817	BLU B2	0x08	RW	Bit[7:4]: Not used Bit[3:0]: blu_b2[3:0] Blue b2
0x5818	LENC CTRL00	0x0C	RW	Bit[3]: rst_seed Bit[2]: rnd_en Bit[1]: coef_m_en Bit[0]: gcoef_en
0x5819	LENC COEF TH	0x80	RW	Bit[7:0]: lenc_coef_th[7:0] LENC coefficient threshold
0x581A	LENC GAIN THRE1	0x06	RW	Bit[7:6]: Not used Bit[5:0]: lenc_gain_thre1[5:0]
0x581B	LENC GAIN THRE2	0xC	RW	Bit[7:6]: Not used Bit[5:0]: lenc_gain_thre2[5:0]
0x581C	COEF MAN	0x80	RW	Bit[7:0]: coef_man[7:0]

5.31 average [0x5900 - 0x5913]

table 5-31 average registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x5900	XSTART	0x00	RW	Bit[7:5]: Not used Bit[4:0]: Xstart[12:8]
0x5901	XSTART	0x00	RW	Bit[7:0]: Xstart[7:0]
0x5902	YSTART	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Ystart[11:8]
0x5903	YSTART	0x00	RW	Bit[7:0]: Ystart[7:0]
0x5904	X WIN	0x10	RW	Bit[7:5]: Not used Bit[4:0]: x_win[12:8]
0x5905	X WIN	0xA0	RW	Bit[7:0]: x_win[7:0]
0x5906	Y WIN	0x0C	RW	Bit[7:4]: Not used Bit[3:0]: y_win[11:8]
0x5907	Y WIN	0x78	RW	Bit[7:0]: y_win[7:0]
0x5908	R AVG CTRL08	0x11	RW	Bit[7:4]: wt1 Bit[3:0]: wt0

table 5-31 average registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x5909	R AVG CTRL09	0x11	RW	Bit[7:4]: wt3 Bit[3:0]: wt2
0x590A	R AVG CTRL0A	0x11	RW	Bit[7:4]: wt5 Bit[3:0]: wt4
0x590B	R AVG CTRL0B	0x11	RW	Bit[7:4]: wt7 Bit[3:0]: wt6
0x590C	R AVG CTRL0C	0x11	RW	Bit[7:4]: wt9 Bit[3:0]: wt8
0x590D	R AVG CTRL0D	0x11	RW	Bit[7:4]: wt11 Bit[3:0]: wt10
0x590E	R AVG CTRL0E	0x11	RW	Bit[7:4]: wt13 Bit[3:0]: wt12
0x590F	R AVG CTRL0F	0x11	RW	Bit[7:4]: wt15 Bit[3:0]: wt14
0x5910	R AVG CTRL10	0x02	RW	Bit[7:2]: Not used Bit[1]: avg_opt Bit[0]: avg_man
0x5911	WT SUM O	–	R	Bit[7:0]: wt_sum_o[7:0]
0x5912	AVG SCCB DONE	–	R	Bit[7:1]: Not used Bit[0]: avg_sccb_done[0]
0x5913	AVG	–	R	Bit[7:0]: avg[7:0]

6 operating specifications

6.1 absolute maximum ratings

table 6-1 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
peak solder temperature (10 second dwell time)		245°C

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

6.2 functional temperature

table 6-2 functional temperature

parameter	range
operating temperature[a]	-30°C to +85°C junction temperature
stable image temperature[b]	0°C to +50°C junction temperature

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

b. image quality remains stable throughout this temperature range

6.3 DC characteristics

note The OV2685 uses tan external power supply to provide digital core 1.8V DVDD.

table 6-3 DC characteristics (30°C < T_J < 85°C)

symbol	parameter	min	typ	max	unit
supply					
V_{DD-A}	supply voltage (analog)	2.6	2.8	3.0	V
V_{DD-D}	supply voltage (digital core for 2-lane MIPI up to 576 Mbps/lane)	1.7	1.8	1.9	V
V_{DD-IO}	supply voltage (digital I/O)	1.7	1.8	3.0	V
I_{DD-A}	active (operating) current		TBD	TBD	mA
I_{DD-IO}			TBD	TBD	mA
$I_{DDS-SCCB}$	standby current[a]		TBD	TBD	µA
$I_{DDS-XSHUTDOWN}$[b]			TBD	TBD	µA
digital inputs (typical conditions: AVDD = 2.8V, DVDD = 1.8V, DOVDD = 1.8V, EVDD = 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}[c]	SIOC and SIOD	-0.5	0	0.54	V
V_{IH}	SIOC and SIOD	1.28	1.8	3.0	V

a. standby current is measured at room temperature

b. it is necessary to cut off external DVDD outside the sensor to eliminate leakage current

c. based on DOVDD = 1.8V

6.4 timing characteristics

table 6-4 timing characteristics

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

a. if using internal PLL

6.5 power up sequence

The sensor includes an on-chip initial power-up reset feature that will reset the whole chip during power up.

6.6 hardware and software standby

The following suspend modes are available for the OV2685:

- hardware standby
  To initiate hardware standby mode, the XSHUTDOWN pin must be tied to low. When this occurs, the OV2685 internal device clock is halted and all internal counters and registers are reset.
- SCCB software standby
  Executing a software standby through the SCCB interface suspends internal circuit activity, but does not halt the device clock. All register content is maintained in standby mode.

6.7 power up and power down sequence

6.7.1 power up sequence

The digital and analog supply voltages can be powered up in any order, for example (DOVDD then AVDD or AVDD then DOVDD).

6.7.2 on-chip power up

- if XSHUTDOWN is low when the power supplies are brought up, sensor will go into hardware standby mode
- if XSHUTDOWN is high when the power supplies are brought up, sensor will go into software standby mode

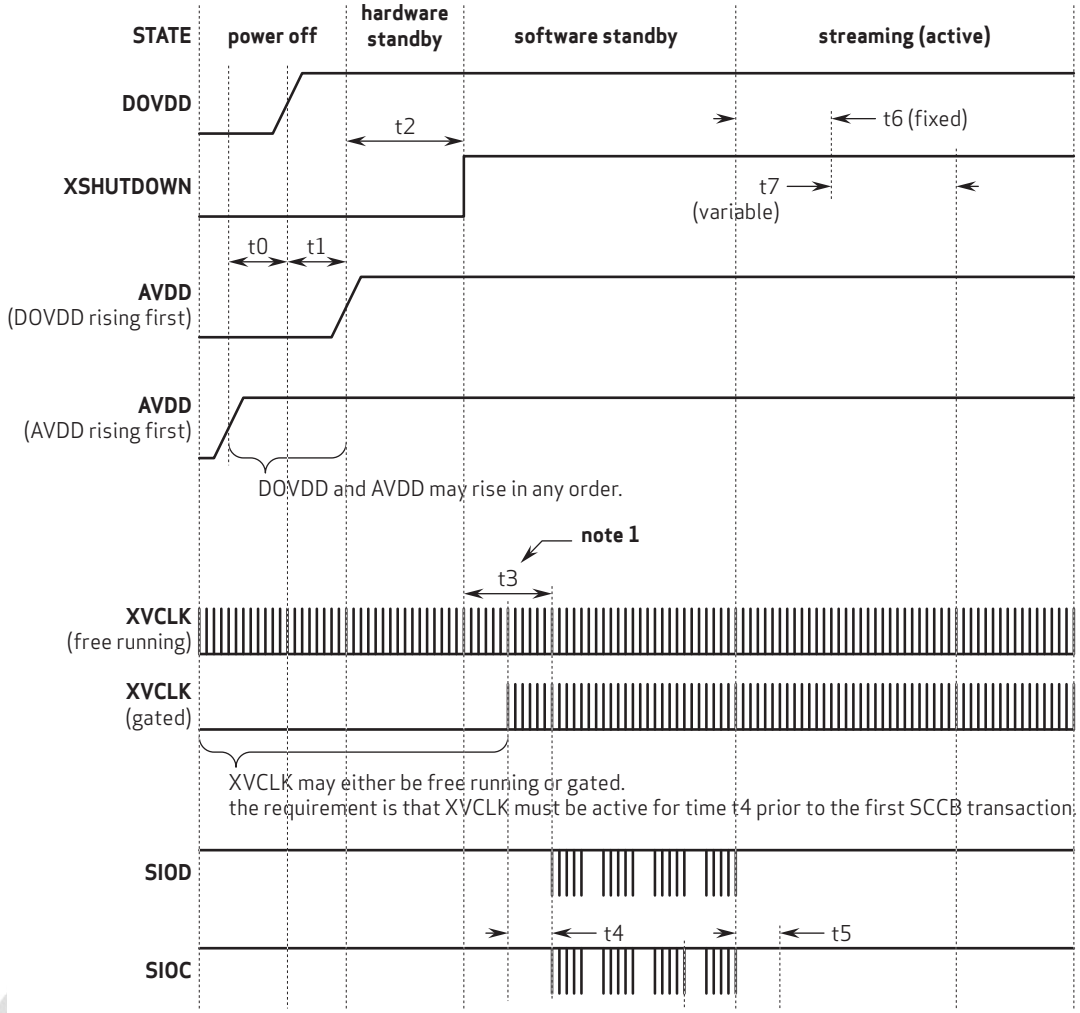
The XVCLK clock can either be initially low and then enabled during software standby mode or XVCLK can be a free running clock.

table 6-5 power up sequence timing constraints power up

constraint	label	min	max	unit
AVDD rising – DOVDD rising	t0	AVDD and DOVDD may rise in any order. rising separation can vary from 0 ns to infinity.		ns
DOVDD rising – AVDD rising	t1			ns
AVDD rising – XSHUTDOWN rising	t2	0.0		ns
XSHUTDOWN rising – first SCCB transaction	t3[a]	8192		XVCLK cycles
minimum number of XVCLK cycles prior to the first SCCB transaction	t4	8192		XVCLK cycles
PLL start up/lock time	t5		0.2	ms
entering streaming mode – first frame start sequence (fixed part)	t6		10	ms
entering streaming mode – first frame start sequence (variable part)	t7	delay is the integration time value		lines

a. when using the internal DVDD, an additional 1ms must be added to t3 to wait for SCCB to become stable

figure 6-1 power up sequence 1



note 1 when using the internal DVDD, an additional 1ms must be added to t3 to wait for SCCB to become stable

2685_DS_6_1

figure 6-2 power up sequence 2

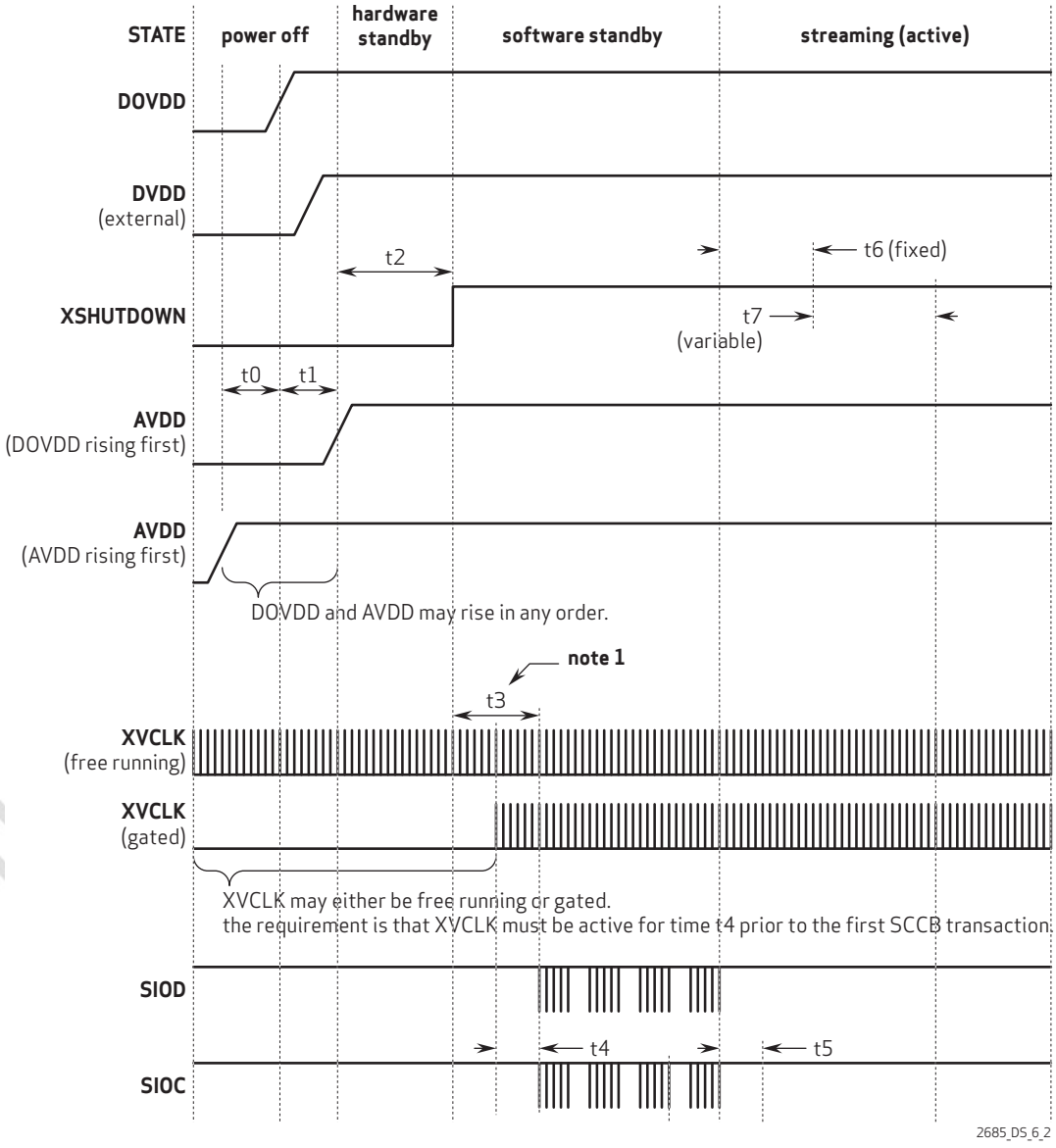


figure 6-3 standby sequence 1

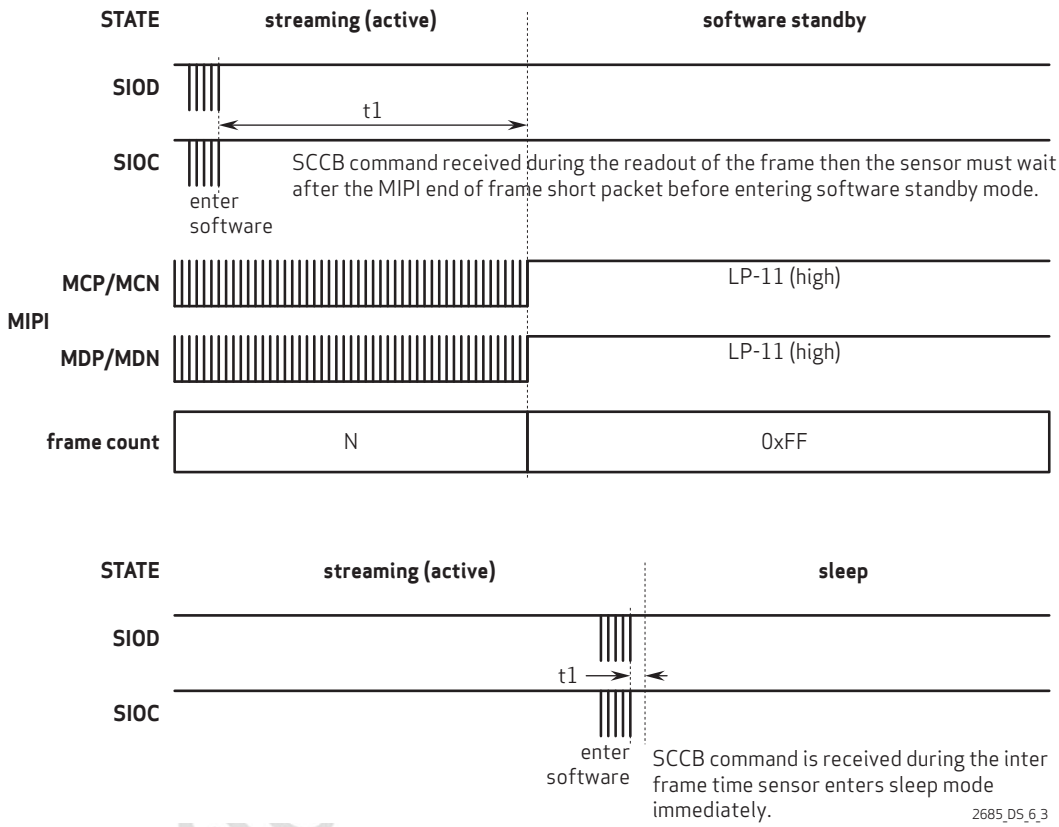
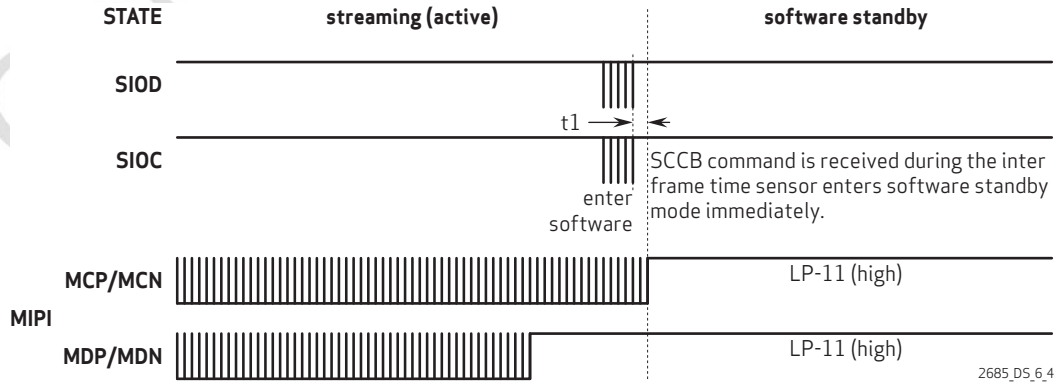


figure 6-4 standby sequence 2



6.8 power down sequence

The digital and analog supply voltages can be powered down in any order (e.g. DOVDD, then AVDD or AVDD, then DOVDD). Similar to the power-up sequence, the XVCLK input clock may be either gated or continuous. If the SCCB command to exit streaming is received while a frame of MIPI data is being output, then the sensor must wait to the MIPI frame end code before entering software standby mode.

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

table 6-6 power down sequence timing constraints

constraint	label	min	max	unit
enter software standby SCCB command device in software standby mode	t0	when a frame of MIPI data is output, wait for the MIPI end code before entering the software for standby; otherwise, enter the software standby mode immediately		
minimum of XVCLK cycles after the last SCCB transaction or MIPI frame end	t1	512		XVCLK cycles
last SCCB transaction or MIPI frame end, XSHUTDOWN falling	t2	512		XVCLK cycles
XSHUTDOWN falling - AVDD falling	t3	0.0		ns
AVDD falling - DOVDD falling	t4	AVDD and DOVDD may fall in any order, the falling separation can vary from 0 ns to infinity		ns
DOVDD falling - AVDD falling	t5			ns

figure 6-5 power down sequence 1

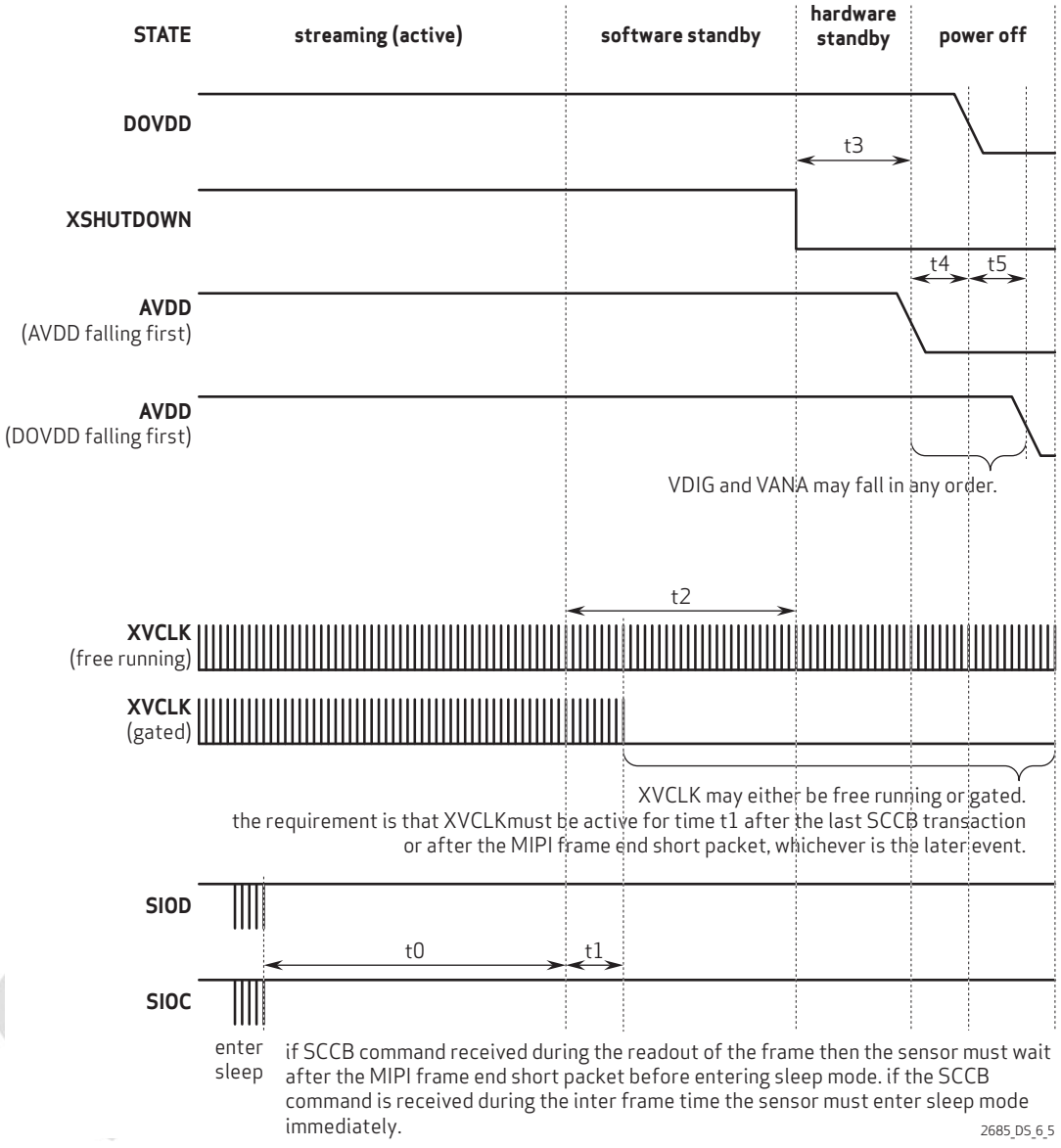
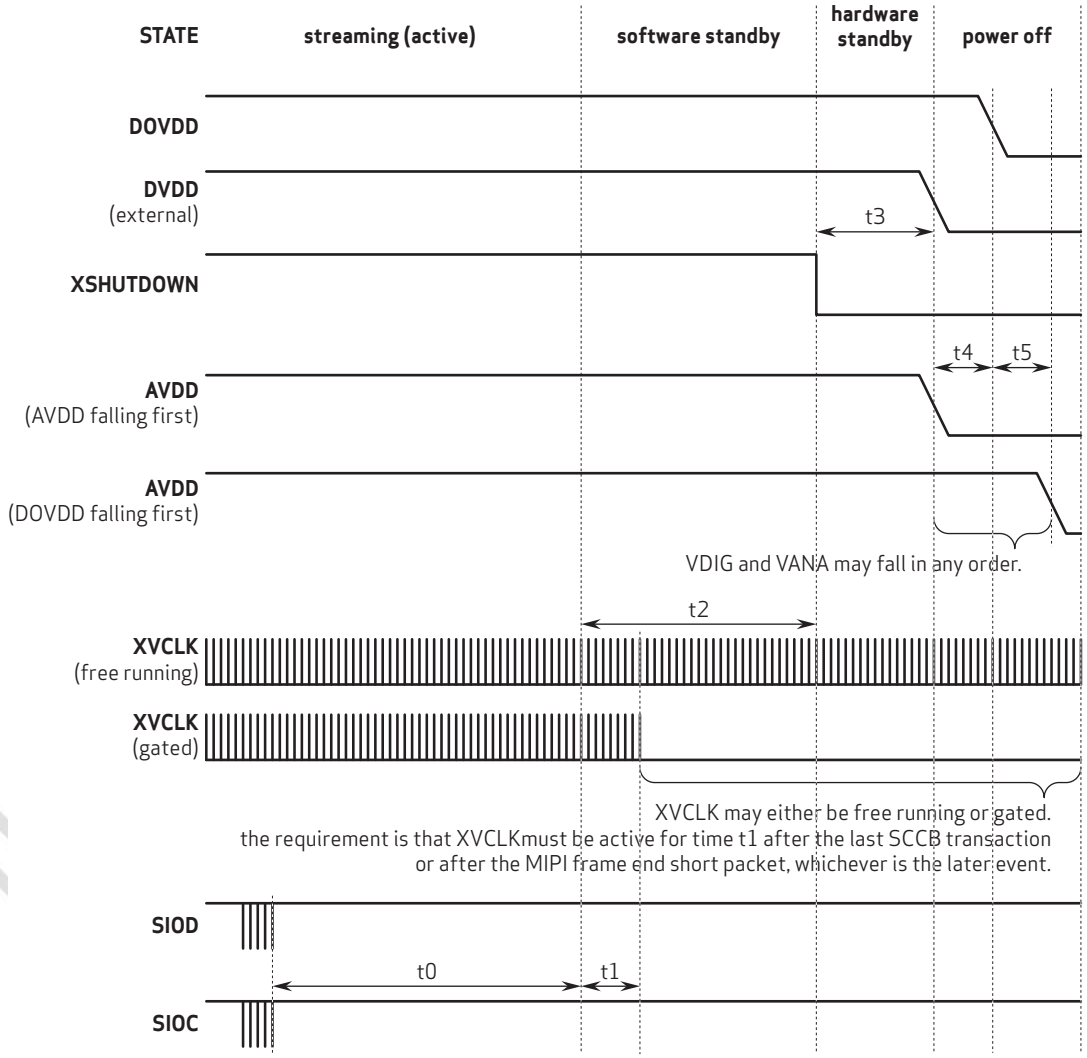


figure 6-6 power down sequence 2



7 mechanical specifications

7.1 physical specifications

figure 7-1 package specifications

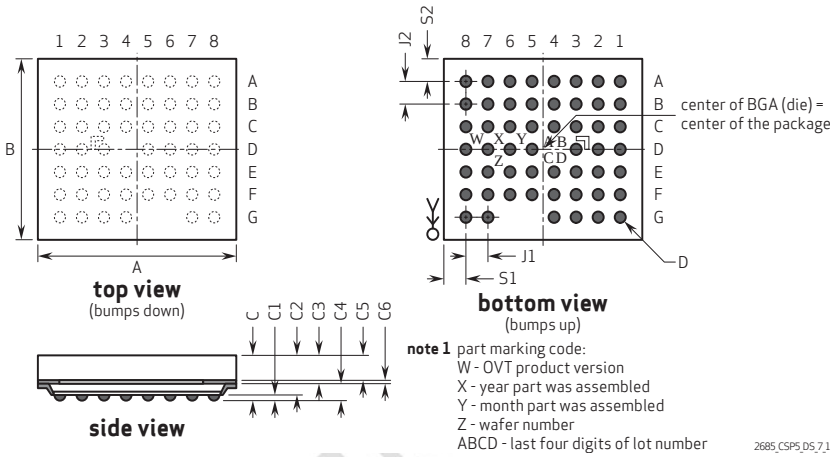


table 7-1 package dimensions

parameter	symbol	min	typ	max	unit
package body dimension x	A	4429	4454	4479	µm
package body dimension y	B	3989	4014	4039	µm
package height	C	680	740	800	µm
ball height	C1	100	130	160	µm
package body thickness	C2	575	610	645	µm
thickness from top glass surface to wafer	C3	425	445	465	µm
image plane height	C4	250	295	340	µm
glass thickness	C5	385	400	415	µm
air gap between sensor and glass	C6	41	45	49	µm
ball diameter	D	220	250	280	µm
total pin count	N		53 (4 NC)		
pins pitch x-axis	J1		530		µm
pins pitch y-axis	J2		480		µm
edge-to-pin center distance along x	S1	342	372	402	µm
edge-to-pin center distance along y	S2	537	567	597	µm

7.2 IR reflow specifications

note
The OV2685 uses a lead free package.

figure 7-2 IR reflow ramp rate requirements

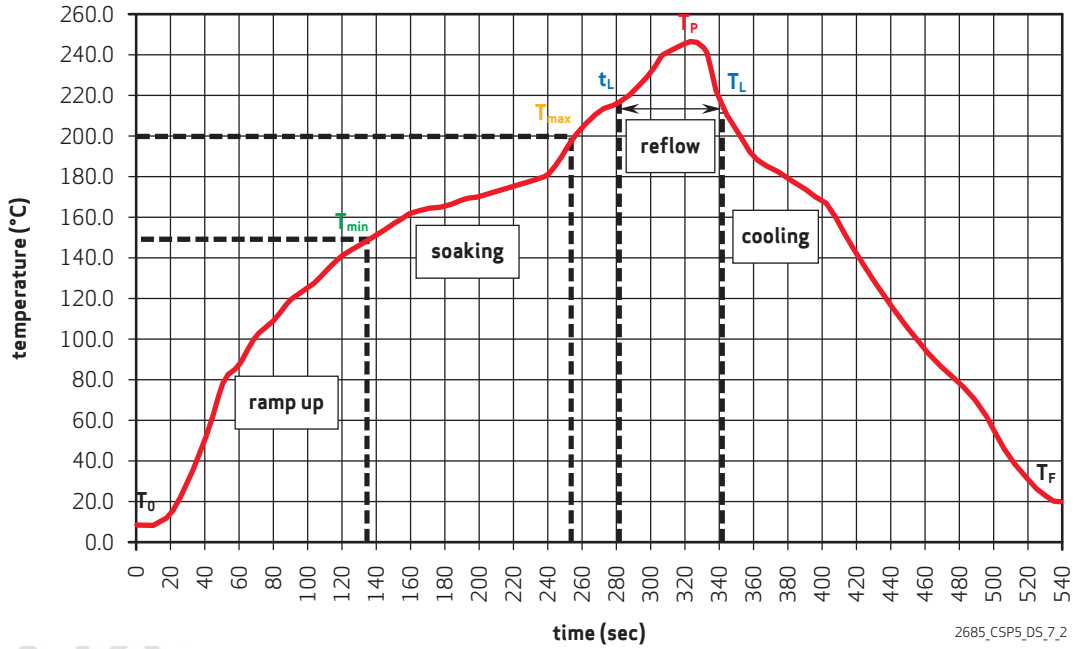


table 7-2 reflow conditions[ab]

zone	description	exposure
ramp up A (T_0 to T_{min})	heating from room temperature to 150°C	temperature slope ≤ 3°C per second
soaking	heating from 150°C to 200°C	90 ~ 150 seconds
ramp up B (t_L to T_p)	heating from 217°C to 245°C	temperature slope ≤ 3°C per second
peak temperature	maximum temperature in SMT	245°C +0/-5°C (duration max 30sec)
reflow (t_L to T_L)	temperature higher than 217°C	30 ~ 120 seconds
ramp down A (T_p to T_L)	cooling down from 245°C to 217°C	temperature slope ≤ 3°C per second
ramp down B (T_L to T_f)	cooling down from 217°C to room temperature	temperature slope ≤ 2°C per second
T_0 to T_p	room temperature to peak temperature	≤ 8 minutes

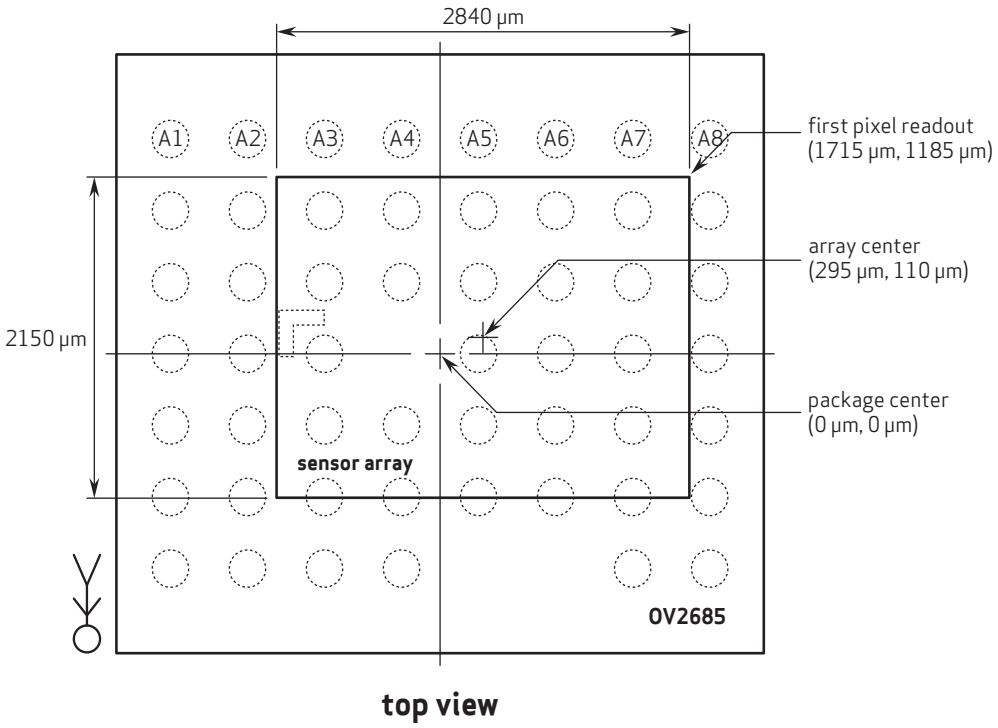
a. maximum number of reflow cycles =3

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

8 optical specifications

8.1 sensor array center

figure 8-1 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 pins A1 to A8 oriented down on the PCB.

2685_CSP5_DS_8_1

8.2 lens chief ray angle (CRA)

figure 8-2 chief ray angle (CRA)

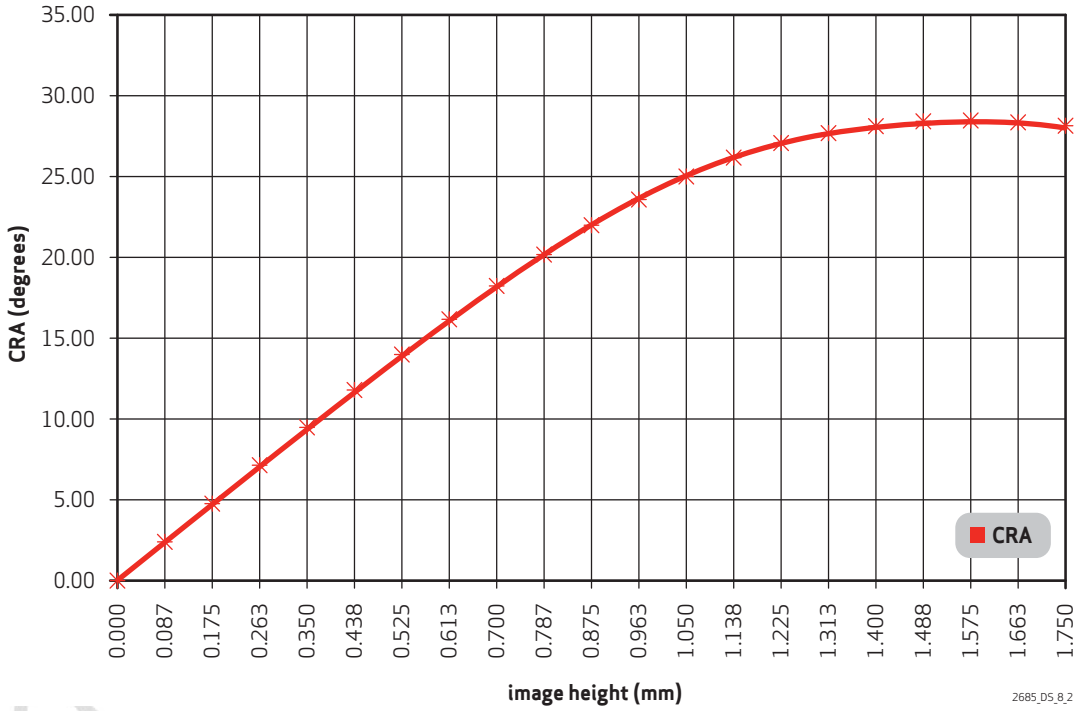


table 8-1 CRA versus image height plot (sheet 1 of 2)

field (%)	image height (mm)	CRA (degrees)
0.00	0	0
0.05	0.087	2.42
0.10	0.175	4.83
0.15	0.263	7.23
0.20	0.35	9.57
0.25	0.438	11.86
0.30	0.525	14.09
0.35	0.613	16.23
0.40	0.7	18.29
0.45	0.787	20.23

table 8-1 CRA versus image height plot (sheet 2 of 2)

field (%)	image height (mm)	CRA (degrees)
0.50	0.875	22.05
0.55	0.963	23.68
0.60	1.05	25.08
0.65	1.138	26.24
0.70	1.225	27.11
0.75	1.313	27.75
0.80	1.4	28.19
0.85	1.488	28.45
0.90	1.575	28.53
0.95	1.663	28.41
1.00	1.75	28.19

revision history

version 1.0 08.29.2013

- initial release

defining the future of digital imaging™

OmniVision Technologies, Inc.

UNITED STATES

4275 Burton Drive
Santa Clara, CA 95054

tel: + 1 408 567 3000
fax: + 1 408 567 3001
email: salesamerican@ovt.com

UNITED KINGDOM

Hampshire + 44 1256 744 610

GERMANY

Munich +49 89 63 81 99 88

INDIA

Bangalore +91 988 008 0140

CHINA

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

JAPAN

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

KOREA

Seoul + 82 2 3478 2812

SINGAPORE +65 6220 1335

TAIWAN

Taipei +886 2 2657 9800
Hsinchu +886 3 6110933

website: **www.ovt.com**