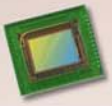


datasheet

PRELIMINARY SPECIFICATION

1/7.5" b&w CMOS VGA (640 x 480) image sensor
with OmniPixel3-GS™ technology

OV7251 (rev 1J)

b&w CMOS VGA (640 x 480) image sensor with OmniPixel3-GS™ technology

datasheet (CSP)
PRELIMINARY SPECIFICATION

version 1.1
december 2017

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

applications

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

ordering information

- **OV07251-A35A-1J** (b&w, lead-free)
  35-pin CSP

features

- 3 µm x 3 µm pixel with OmniPixel3-GS™ technology
- automatic black level calibration (ABLC)
- programmable controls for frame rate, mirror and flip, cropping and windowing
- support output formats: 8/10-bit RAW
- support for image sizes: 640x480, 320x240, 160x120
- fast mode switching
- supports horizontal and vertical 2:1 and 4:1 monochrome subsampling
- supports 2x2 monochrome binning
- one-lane MIPI serial output interface
- one-lane LVDS serial output interface
- embedded 256 bits of one-time programmable (OTP) memory for part identification
- two on-chip phase lock loops (PLLs)
- built-in 1.5V regulator for core
- PWM
- built-in strobe control

key specifications (typical)

- **active array size:** 640 x 480
- **power supply:**
  analog: 2.8V (nominal)
  core: 1.5V (optional)
  I/O: 1.8V (nominal)
- **power requirements:**
  active: 119 mW @ 120fps, VGA output
  standby: 15 µA for AVDD, 40 µA for DOVDD without input clock, 700 µA for DOVDD with input clock
  XSHUTDOWN: 5 µA for AVDD, 5 µA for DOVDD
- **temperature range:**
  operating: -30°C to 70°C junction temperature
  stable image: 0°C to 50°C junction temperature
- **output interface:** 1-lane MIPI/LVDS serial output
- **output formats:** 10-bit RAW BW
- **lens size:** 1/7.5"
- **input clock frequency:** 6~27 MHz
- **lens chief ray angle:** 29° non-linear
- **max S/N ratio:** 38 dB
- **dynamic range:** 69.6 dB @ 8x gain
- **maximum image transfer rate:**
  640 x 480: 120 fps (see **table 2-1**)
- **sensitivity:** 10,800mV/(µW.cm^{-2}.sec) @ 850nm
- **scan mode:** progressive
- **maximum exposure interval:** 502 x t_{ROW}
- **pixel size:** 3 µm x 3 µm
- **image area:** 1968 µm x 1488 µm
- **package dimensions:** 3910 µm x 3410 µm

table of contents

list of figures

list of tables

1 signal descriptions

table 1-1 lists the signal descriptions and their corresponding pin numbers for the OV7251 image sensor. The package information is shown in **section 9**.

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

pin number	signal name	pin type	description
A1	NC	–	no connect
A3	VH	reference	internal analog reference
A4	VN2	reference	internal analog reference
A5	AGND	ground	ground for analog circuit
A6	NC	–	no connect
A7	NC	–	no connect
B2	VM	reference	internal analog reference
B3	**MDP**	I/O	MIPI/LVDS data lane positive output
B4	VN1	reference	internal analog reference
B5	AVDD	power	power for analog circuit
B6	**SIOD**	I/O	SCCB data
C2	EVDD	power	power for MIPI circuit (connect to DVDD outside of sensor)
C3	**MDN**	I/O	MIPI/LVDS data lane negative output
C4	**SIOC**	input	SCCB input clock
C5	**FSIN/VSYNC**	I/O	FSIN input/ VSYNC output
C6	**TMO**	output	test mode output
D1	NC	–	no connect
D2	EGND	ground	ground for MIPI circuit
D3	**MCP**	output	MIPI/LVDS clock lane positive output
D4	DGND	ground	ground for I/O and digital circuit
D5	**PWM**	I/O	PWM output
D6	**XSHUTDOWN**	input	reset (active low with internal pull down resistor)
D7	NC	–	no connect
E2	**TM**	input	test mode (active high with internal pull down resistor)
E3	**MCN**	output	MIPI/LVDS clock lane negative output
E4	DGND	ground	ground for I/O and digital circuit

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

pin number	signal name	pin type	description
E5	EXTCLK	input	system input clock/scan clock input
E6	DOVDD	power	power for I/O circuit
F1	NC	–	no connect
F2	NC	–	no connect
F3	DVDD	reference	power for digital circuit
F4	STROBE	I/O	strobe output
F5	DVDD	reference	power for digital circuit
F6	NC	–	no connect
F7	NC	–	no connect

table 1-2 configuration under various conditions

pin number	signal name	XSHUTDOWN[a]	after XSHUTDOWN release[b]	software standby[c]
B3	MDP	high-z	high-z by default	high-z by default (configurable)
B6	SIOD	high-z	input/open drain	input/open drain
C3	MDN	high-z	high-z by default	high-z by default (configurable)
C4	SIOC	input	input	input
C5	FSIN/VSYNC	high-z	high-z by default	high-z by default (configurable)
C6	TMO	high-z	input/open drain	input/open drain
D3	MCP	high-z	high-z by default	high-z by default (configurable)
D5	PWM	high-z	high-z by default	high-z by default (configurable)
D6	XSHUTDOWN	input	input	input
E2	TM	input	input	input
E3	MCN	high-z	high-z by default	high-z by default (configurable)
E5	EXTCLK	input	input	input
F4	STROBE	high-z	high-z by default	high-z by default (configurable)

a. XSHUTDOWN = 0

b. XSHUTDOWN = 1

c. XSHUTDOWN = 1
standby initiated by register

figure 1-1 pin diagram

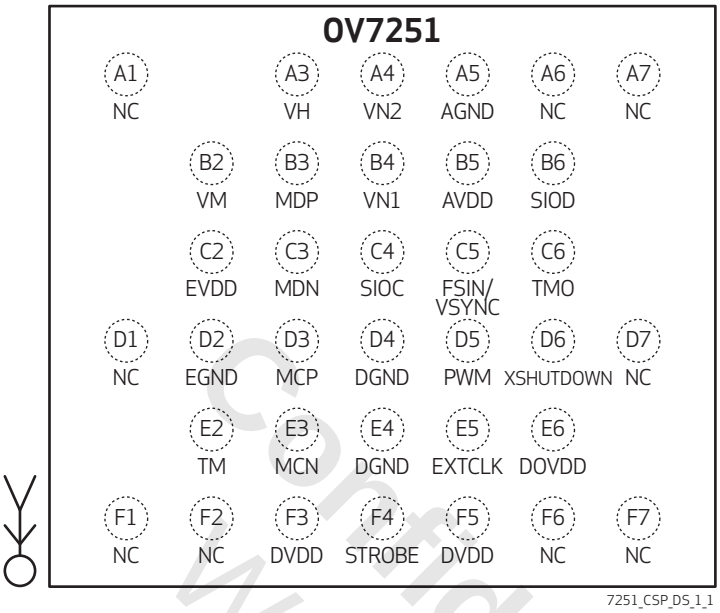


table 1-3 pad symbol and equivalent circuit (sheet 1 of 2)

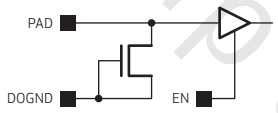
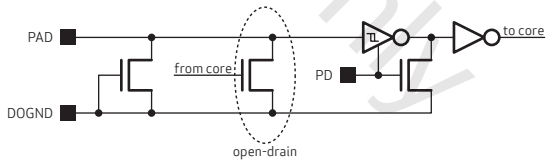
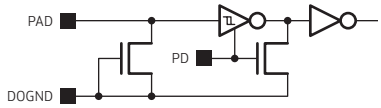
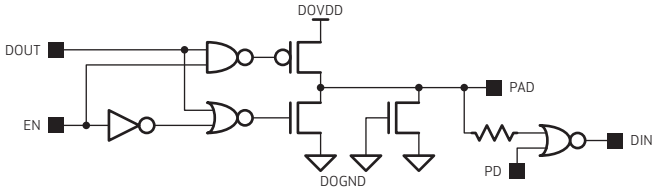
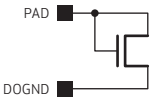
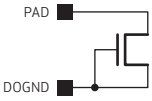
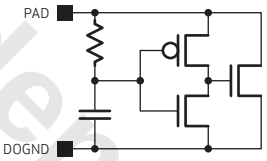
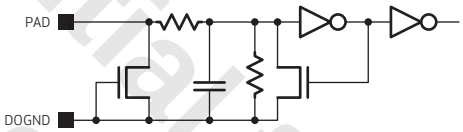
symbol	equivalent circuit
EXTCLK	
SIOD, TMO	
SIOC	

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

symbol	equivalent circuit
FSIN/VSYNC, STROBE, PWM	
VN1, VN2	
MDP, MDN, MCP, MCN, EGND, AGND, DGND, VH, VM	
AVDD, EVDD, DVDD, DOVDD	
TM, XSHUTDOWN	

2 system level description

2.1 overview

The OV7251 b&w image sensors are low voltage, high performance, 1/7.5 inch, VGA, CMOS, image sensors that provide the functionality of a single VGA (640x480) camera using OmniPixel3-GS™ technology. They provide full-frame, sub-sampled, and windowed 8/10-bit MIPI images via the control of the Serial Camera Control Bus (SCCB) interface.

The OV7251 has an image array capable of operating at up to 120 frames per second (fps) in 10-bit, VGA resolution with complete user control over image quality, formatting and output data transfer.

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 image.

For customized information purposes, the OV7251 includes 256 bits of one-time programmable (OTP) memory. The OV7251 has one lane MIPI interface.

2.2 architecture

The OV7251 sensor core generates streaming pixel data at a constant frame rate. **figure 2-1** shows the functional block diagram.

The timing generator outputs signals to access the image array. The entire pixel array is reset at the same point of time. After the exposure time has elapsed, the pixels stop gathering light and store the collected charge in a storage node. The charge then reads out row by row.

The exposure time is controlled by adjusting the time interval between reset and transferring the charge to storage node. After the data of the pixels in the row has been sampled, it is processed through analog circuitry to correct the offset and multiply the data with corresponding gain. Following analog processing is the ADC which outputs 10-bit data for each pixel in the array.

figure 2-1 OV7251 block diagram

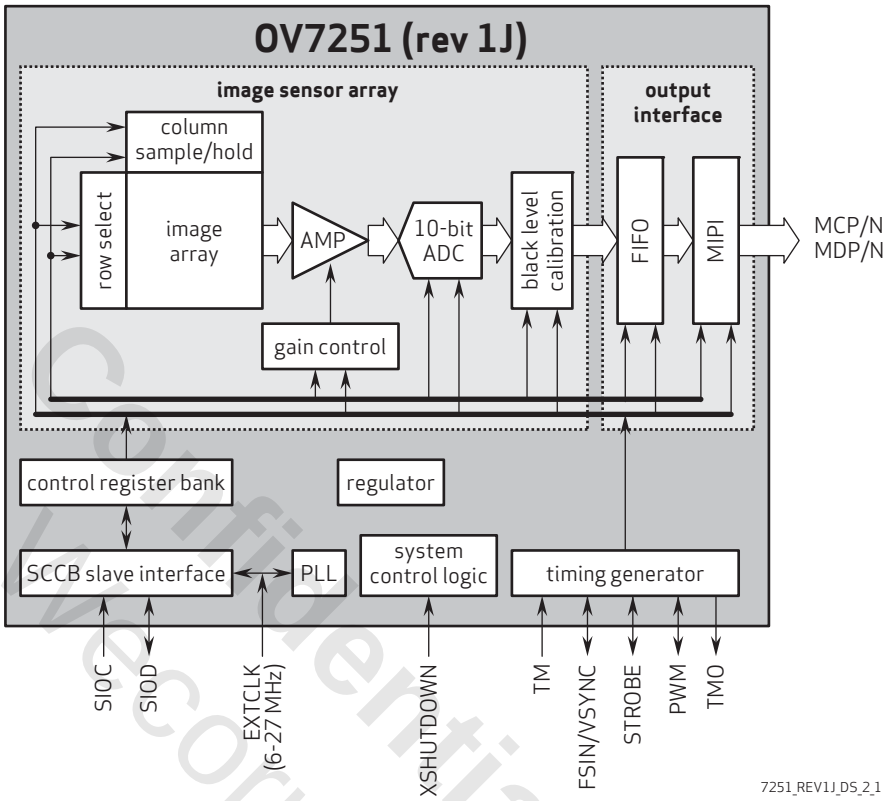
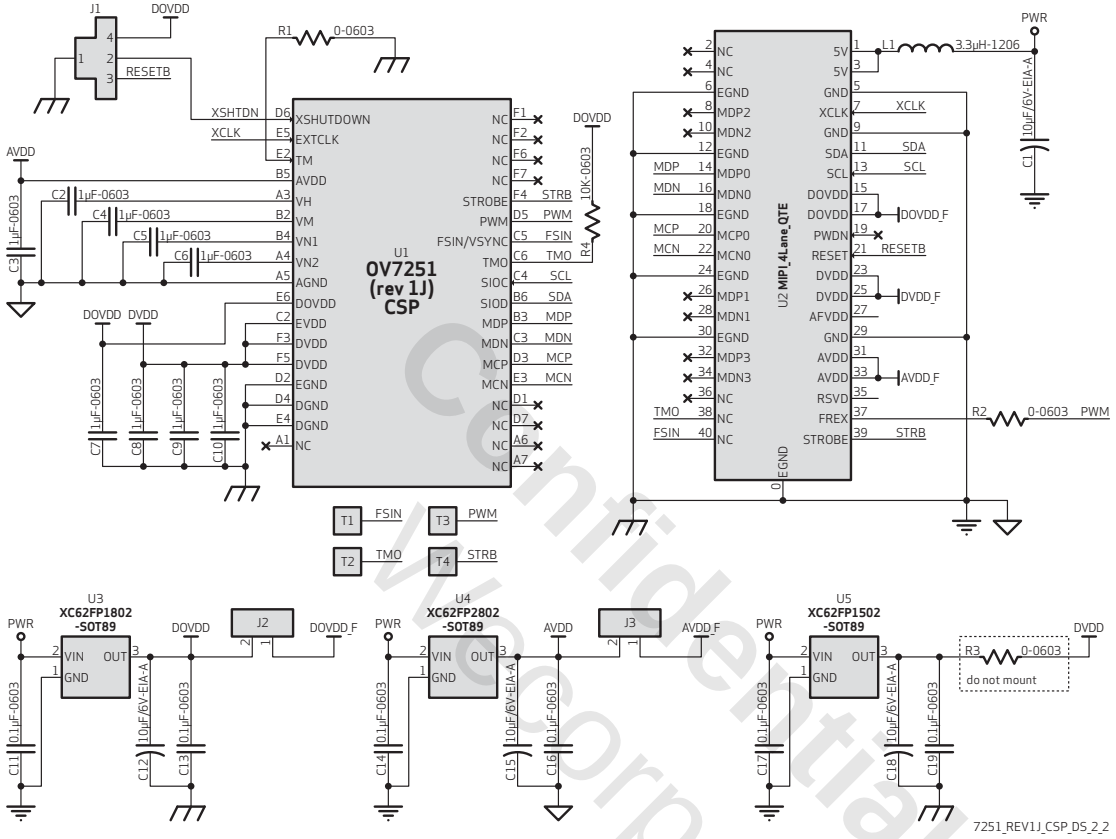


figure 2-2 reference design schematic



2.3 format and frame

The OV7251 supports RAW BW output with a 1-lane MIPI or LVDS interface.

table 2-1 supported resolution and frame rate

format[a]	resolution	max frame rate	methodology	typical MIPI data rate
full resolution	640x480	120 fps	full	1-lane @ 800Mbps
320x240	320x240	180 fps	2x2 binning, 2:1 sub-sampling	1-lane @ 800Mbps
160x120	160x120	360 fps	4:1 sub-sampling	1-lane @ 800Mbps

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

2.3.1 MIPI interface

The OV7251 supports a single lane MIPI transmitter interface with a data transfer rate of up to 800 Mbps.

figure 2-3 MIPI timing

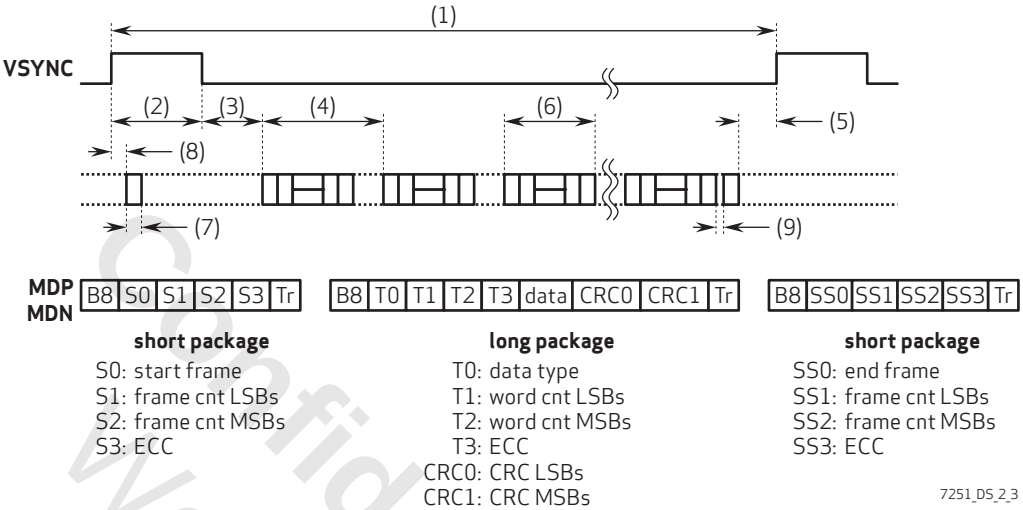


table 2-2 MIPI timing specifications

mode	timing
full resolution 640x480	(1) 478,848 tp (2) 1,024 tp (3) 11,486 tp (4) 928 tp (5) 21,251 tp (6) 387 tp (7) 2 tp (8) -178 tp (9) 29 tp where tp = Tsclk
320x240	(1) 234,688 tp (2) 1,024 tp (3) 6,496 tp (4) 772 tp (5) 42,422 tp (6) 216 tp (7) 22 tp (8) -194 tp (9) 9 tp where tp = Tsclk

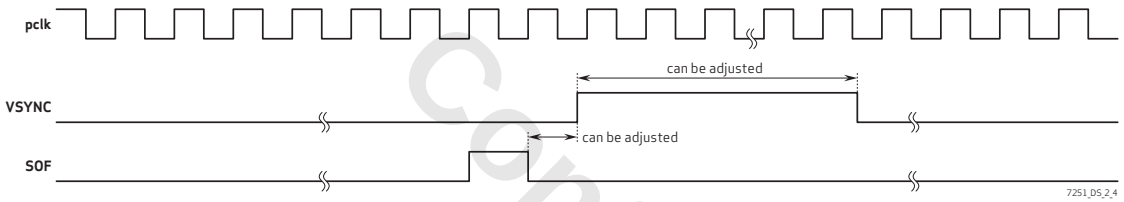
2.3.2 VSYNC timing in MIPI mode

For the MIPI output interface, the VSYNC leading edge can be triggered by either the end of frame (EOF) of the last frame or the start of frame (SOF) of the current frame. In both cases, the delay from EOF/SOF to VSYNC leading edge is controlled by registers {0x4314, 0x4315, 0x4316}, and the VSYNC pulse width is controlled by registers {0x4311, 0x4312} in units of system clock periods.

2.3.2.1 VSYNC mode 1

In mode 1, VSYNC is generated by the internal start of frame (SOF) signal (see **figure 2-4**).

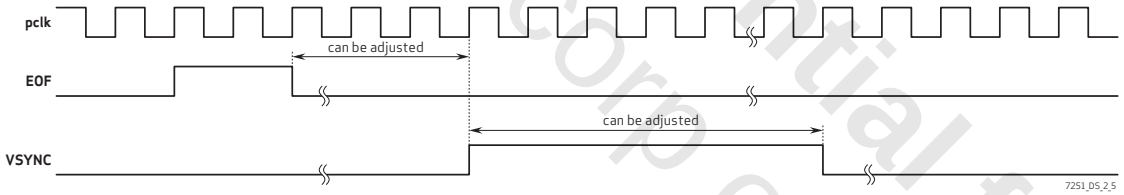
figure 2-4 VSYNC timing in mode 1



2.3.2.2 VSYNC mode 2

In mode 2, VSYNC is generated by the internal end of frame (EOF) signal (see **figure 2-5**).

figure 2-5 VSYNC timing in mode 2



2.4 I/O control

table 2-3 I/O control registers

function	register	description
output drive capability control	0x3001	Bit[6:5]: I/O pin drive capability 00: 1x 01: 2x 10: 3x 11: 4x
STROBE I/O control	0x3005	Bit[3]: input/output control for STROBE pin 0: input 1: output
STROBE output select	0x3027	Bit[3]: output selection for STROBE pin 0: normal data path 1: register control value
STROBE output value	0x3009	Bit[3]: STROBE output value
PWM I/O control	0x3005	Bit[2]: input/output control for PWM pin 0: input 1: output
PWM output select	0x3027	Bit[2]: output selection for PWM pin 0: normal data path 1: register control value
PWM output value	0x3009	Bit[2]: PWM output value
FSIN/VSYNC I/O control	0x3005	Bit[1]: input/output control for FSIN pin 0: input 1: output
FSIN/VSYNC output select	0x3027	Bit[1]: output selection for FSIN pin 0: normal data path 1: register control value
FSIN/VSYNC output value	0x3009	Bit[1]: FSIN output value

2.5 power management

2.5.1 power up sequence

The OV7251 can use an internal regulator to provide digital core 1.5V DVDD for the digital core.

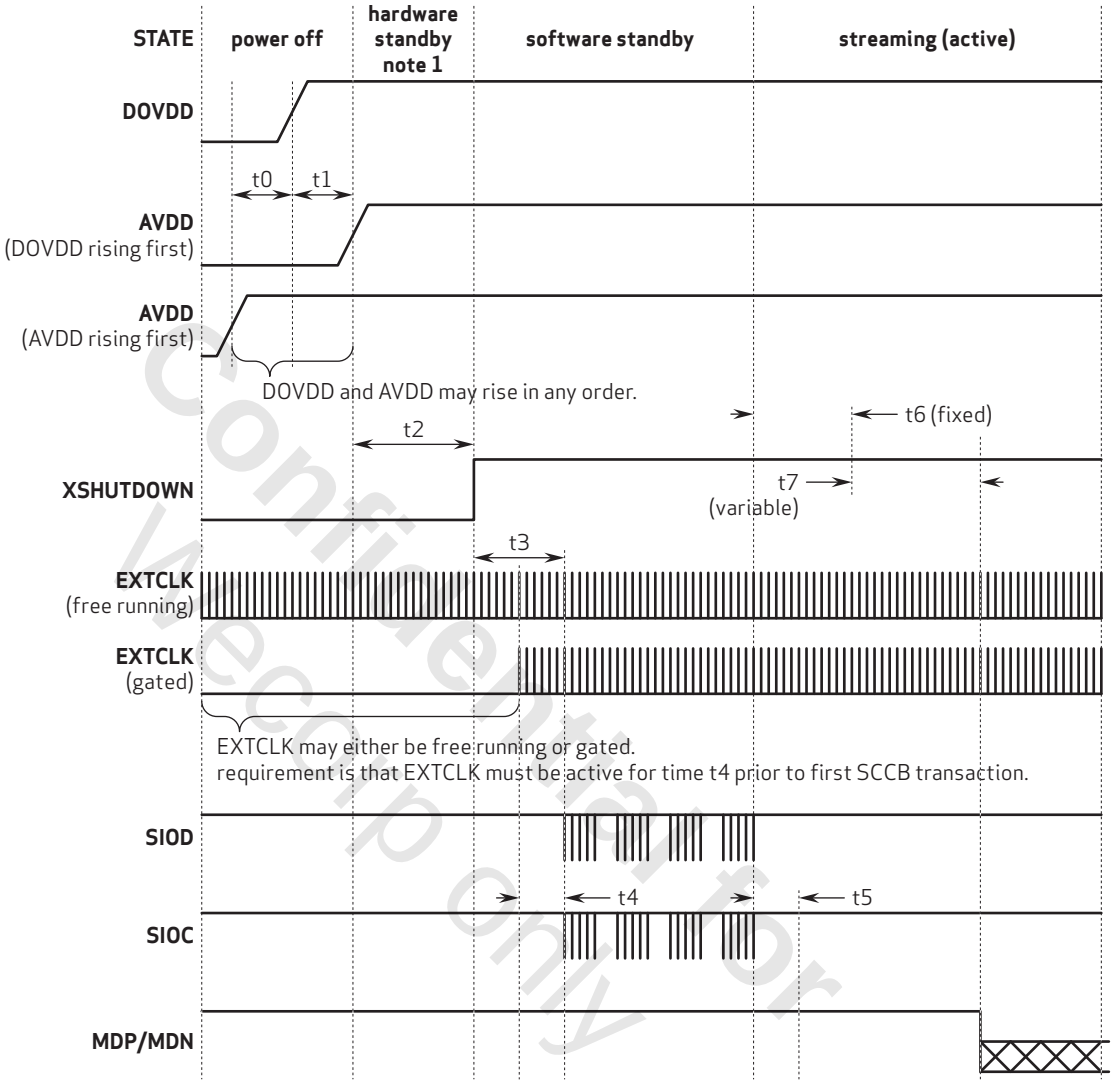
table 2-4 power up sequence

case	DVDD	XSHUTDOWN	power up sequence requirement
1	internal	GPIO	Refer to **figure 2-6** 1. AVDD rising can occur before or after DOVDD rising as long as they are rising before XSHUTDOWN rising 2. XSHUTDOWN is pulled up after AVDD and DOVDD are stable
2	external	GPIO	Refer to **figure 2-7** 1. AVDD rising can occur before or after DOVDD rising as long as they are rising before XSHUTDOWN rising 2. XSHUTDOWN is pulled up after AVDD and DOVDD are stable 3. DVDD rises after DOVDD, but before XSHUTDOWN is pulled high

table 2-5 power up sequence timing constraints

constraint	label	min	max	unit
AVDD rising – DOVDD rising	t0	0	∞	ms
DOVDD rising – AVDD rising	t1			ms
AVDD or DOVDD rising, whichever is last – XSHUTDOWN rising	t2	1		ms
XSHUTDOWN rising – first SCCB transaction	t3	65536		EXTCLK cycles
minimum number of EXTCLK cycles prior to first SCCB transaction	t4	65536		EXTCLK cycles
PLL start up/lock time	t5		0.2	ms
entering streaming mode – first frame start sequence (fixed part)	t6		10	ms
entering streaming mode – first frame start sequence (variable part)	t7	delay is exposure time value		lines
DOVDD to external DVD rising	t8	0		ms
DOVDD rising to XSHUTDOWN rising	t9	0		ms

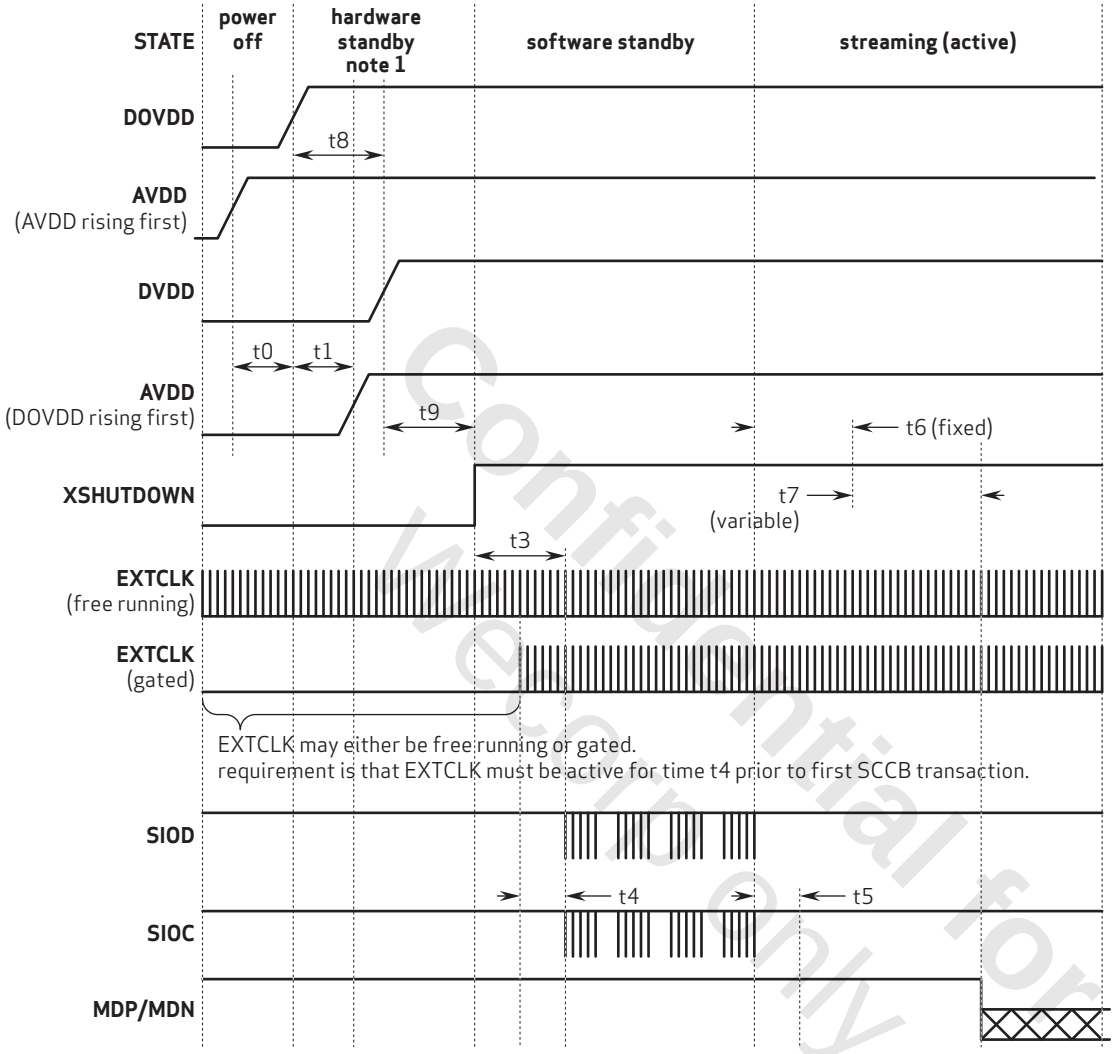
figure 2-6 power up sequence (case 1)



note 1 with minimum power consumption

7251_DS_2_6

figure 2-7 power up sequence (case 2)



note 1 with low power consumption

7251_DS_2_7

2.5.2 power down sequence

The digital and analog supply voltages can be powered down in any order (e.g., DOVDD, then AVDD or AVDD, then DOVDD). Similar to the power-up sequence, the EXTCLK input clock may be either gated or continuous. 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 will enter software standby mode immediately.

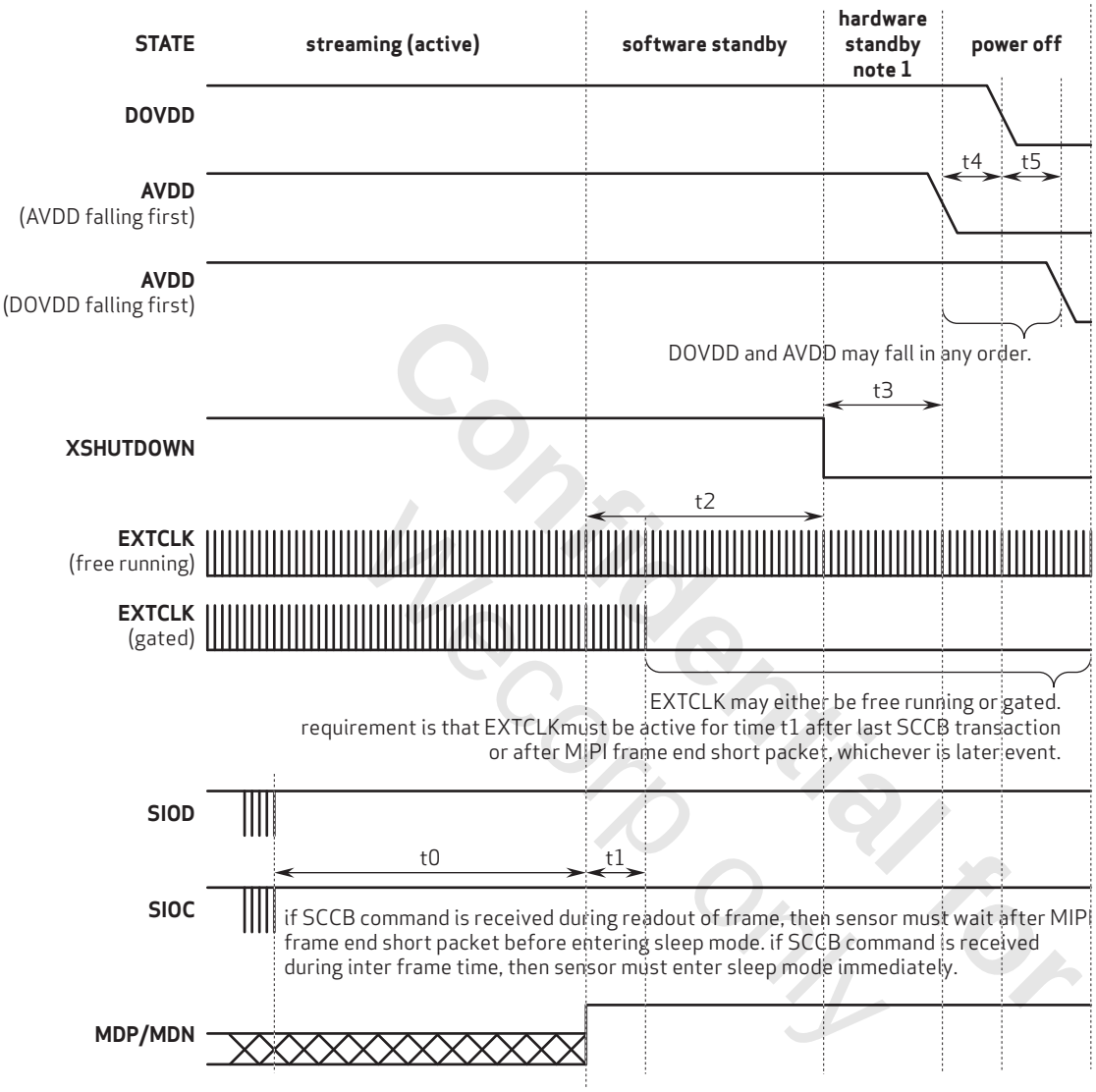
table 2-6 power down sequence

case	DVDD	XSHUTDOWN	power down sequence requirement
1	internal	GPIO	Refer to **figure 2-8** 1. software standby recommended 2. pull XSHUTDOWN low for minimum power consumption 3. AVDD and DOVDD may fall in any order
2	external	GPIO	Refer to **figure 2-9** 1. software standby recommended 2. pull XSHUTDOWN low for minimum power consumption 3. pull DVDD low 4. AVDD and DOVDD may fall in any order

table 2-7 power down sequence timing constraints

constraint	label	min	max	unit
enter software standby SCCB command device in software standby mode	t0	when a frame of MIPI data is output, wait for MIPI end code before entering software for standby; otherwise, enter software standby mode immediately		
minimum of EXTCLK cycles after last SCCB transaction or MIPI frame end	t1	512		EXTCLK cycles
last SCCB transaction or MIPI frame end, XSHUTDOWN falling	t2	512		EXTCLK cycles
XSHUTDOWN falling - AVDD falling or DOVDD falling whichever is first	t3	0.0		ms
AVDD falling - DOVDD falling	t4	AVDD and DOVDD may fall in any order, falling separation can vary from 0 ns to infinity		ms
DOVDD falling - AVDD falling	t5			ms
XSHUTDOWN falling - DVDD falling	t6	0		ms
DVDD falling to DOVDD falling	t7	0		ms

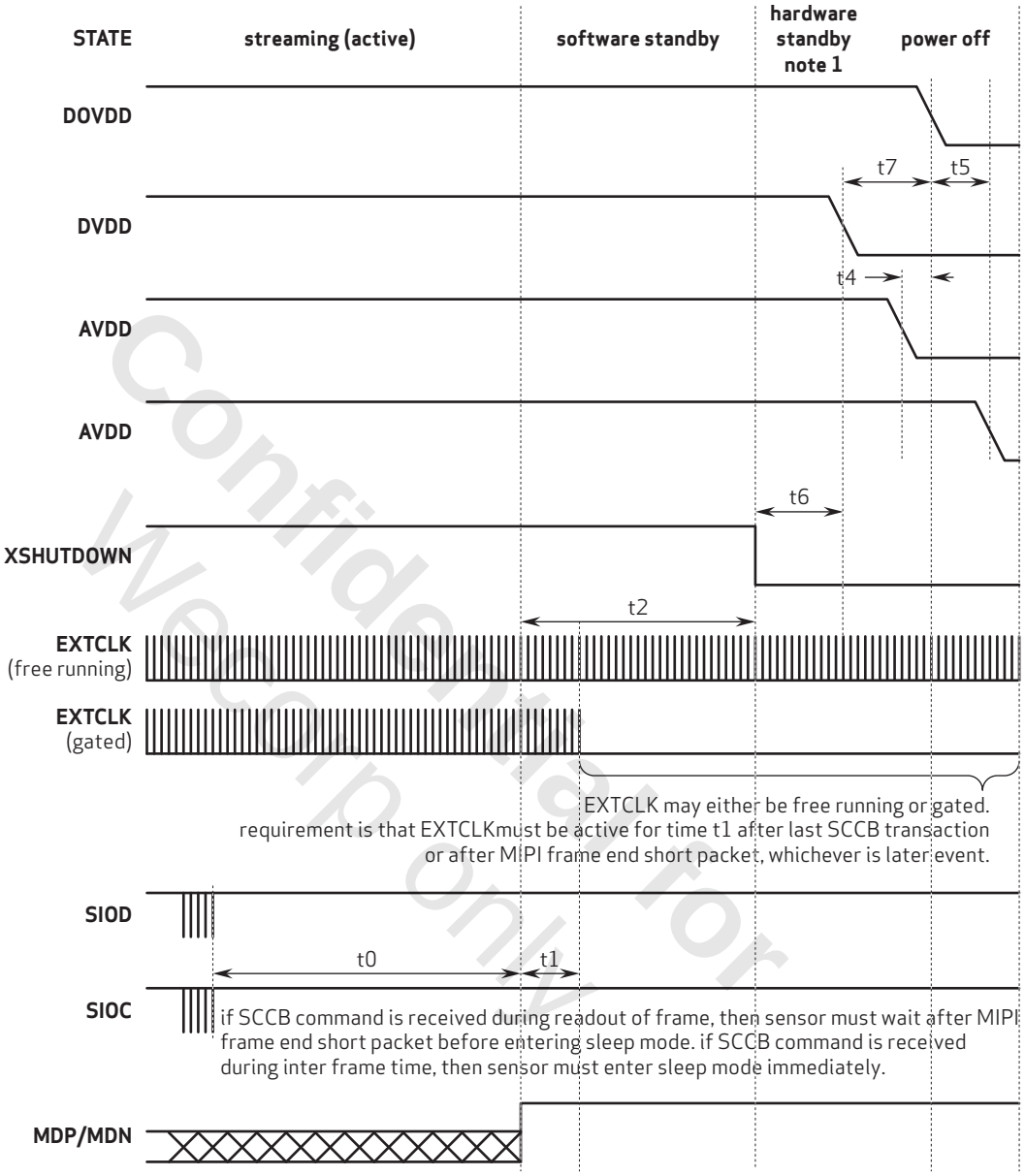
figure 2-8 power down sequence (case 1)



note 1 with minimum power consumption

7251_DS_2_8

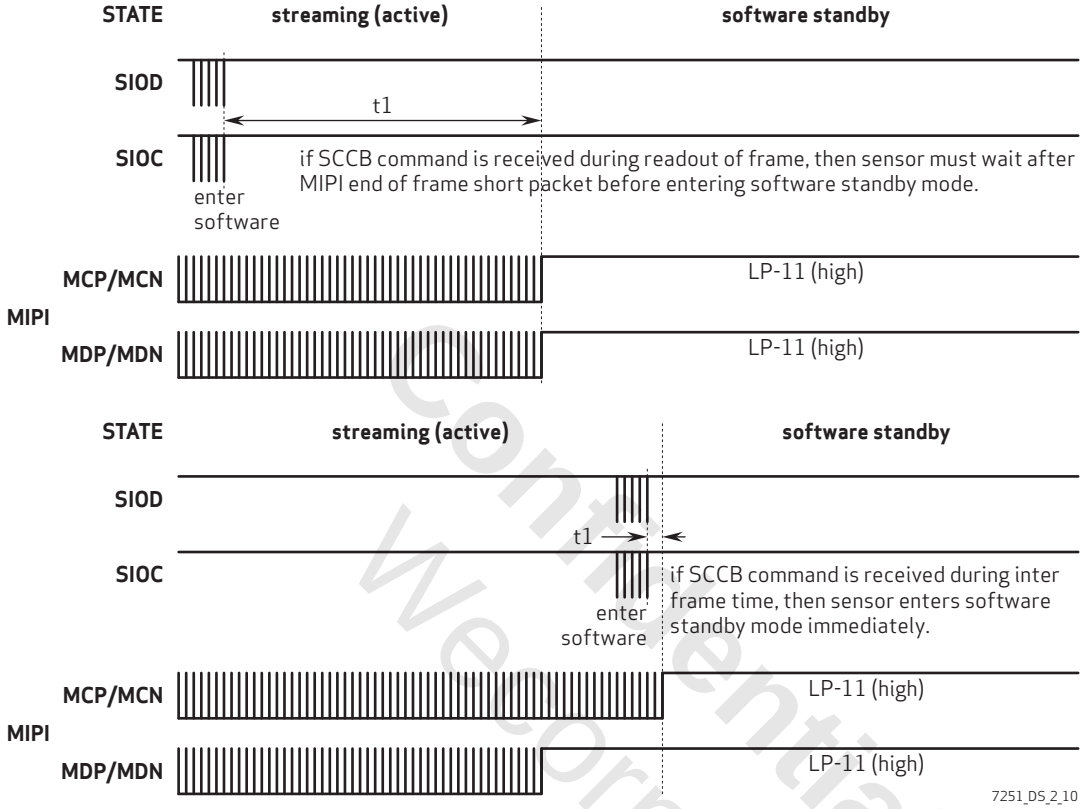
figure 2-9 power down sequence (case 2)



note 1 with low power consumption

7251_DS_2_9

figure 2-10 standby sequence



2.6 reset

The OV7251 sensor includes a **XSHUTDOWN** pin (pin **D6**) that forces a complete hardware reset when it is pulled low (GND). The OV7251 clears all registers and resets them to their default values when a hardware reset occurs.

2.6.1 power ON reset generation

The OV7251 has a power on reset that is generated after the core power becomes stable.

2.7 hardware and software standby

Two suspend modes are available for the OV7251:

- hardware standby
- software standby

table 2-8 hardware and software standby description

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

2.8 system clock control

The OV7251 has two on-chip PLLs which generate the system clock from a 6~27 MHz input clock. A programmable clock divider is provided to generate different frequencies for the system.

2.8.1 PLL configuration

figure 2-11 OV7251 PLL1 clock diagram

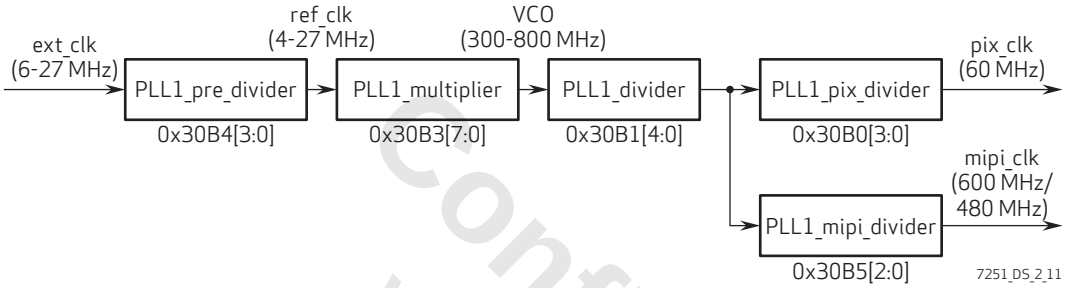
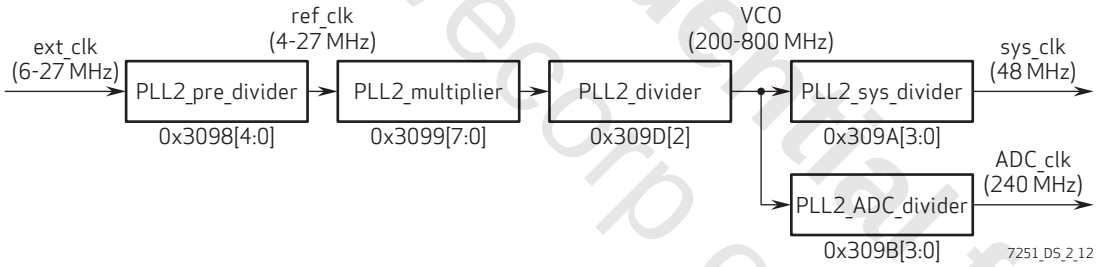


figure 2-12 OV7251 PLL2 clock diagram



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

table 2-9 PLL control registers (sheet 1 of 3)

function	address	description
PLL2_pre_divider	0x3098	Bit[4:0]: PLL2 pre-divider 0x2: /1 0x3: /1.5 0x4: /2 0x5: /2.5 0x6: /3 0x8: /4 0xC: /6 0x10: /8 Others: /1
PLL2_multiplier	0x3099	Bit[7:0]: PLL2 multiplier Multiplier = 0x3099[7:0]

table 2-9 PLL control registers (sheet 2 of 3)

function	address	description
PLL2_divider	0x309D	Bit[2]: PLL2 divider 0: /1 1: /1.5
PLL2_sys_divider	0x309A	Bit[3:0]: System clock divider 0x2: /4 0x3: /6 0x4: /8 0x5: /10 0x6: /12 0x7: /14 0x8: /16 0x9: /18 Others: Not allowed
PLL2_ADC_divider	0x309B	Bit[3:0]: PLL2 ADC clock divider 0x2: /1 0x3: /1.5 0x4: /2 0x5: /2.5 0x6: /3 0x7: /3.5 0x8: /4 0x9: /4.5 Others: Not allowed
PLL1_multiplier	0x30B3	Bit[7:0]: PLL1 multiplier Multiplier = 0x30B3[7:0]
PLL1_pre_divider	0x30B4	Bit[3:0]: PLL1 pre-divider 0x0: /1 0x1: /1 0x2: /2 0x3: /3 0x4: /4 0x5: /1.5 0x6: /6 0x7: /2.5 0x8: /8 Others: Not allowed
PLL1_pix_divider	0x30B0	Bit[3:0]: PLL1 pixel divider 0x8: /8 0xA: /10 Others: Not allowed
PLL1_divider	0x30B1	Bit[4:0]: PLL1 divider Divider = 0x30B1[4:0], when 1 ≤ 0x30B1[4:0] ≤ 16 Others: Not allowed

table 2-9 PLL control registers (sheet 3 of 3)

function	address	description
PLL1_MIPI_divider	0x30B5	Bit[2:0]: PLL1 MIPI divider 0x2: /2 0x4: /4 Others: /1

table 2-10 sample PLL configuration[a]

name	address	value
PLL2_pre_divider	0x3098[4:0]	0x04
PLL2_multiplier	0x3099[7:0]	0x28
PLL2_sys_divider	0x309A[3:0]	0x05
PLL2_ADC_divider	0x309B[3:0]	0x04
PLL_PLL1D	0x309D[2]	0x00
PLL1_multiplier	0x30B3[7:0]	0x32
PLL1_pre_divider	0x30B4[3:0]	0x02
PLL1_pix_divider	0x30B0[3:0]	0x0A
PLL1_divider	0x30B1[4:0]	0x01
SYS_CLK		48 MHz
MIPI_CLK		600 Mbps
EXTCLK		24 MHz

a. PLL control for VGA @ 100 fps with 1 lane, 10-bit output

figure 2-13 clock connection diagram

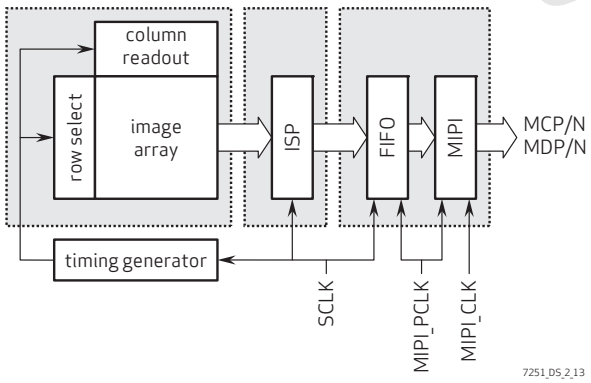


table 2-11 PLL speed limitation

parameter	value
PLL1_multiplier input	4~27 MHz
PLL1_multiplier output	300~800 MHz
PLL2_multiplier input	4~27 MHz
PLL2_multiplier output	200~800 MHz
SYS_CLK	up to 51 MHz

2.9 serial camera control bus (SCCB) interface

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

The OV7251 responds to two SCCB ID set by register SC_SCCB_ID1 (default 0xC0) and SC_SCCB_ID2 (default 0xE0). One of them can be used as a broadcasting ID and the other one can be programmed to a unique ID.

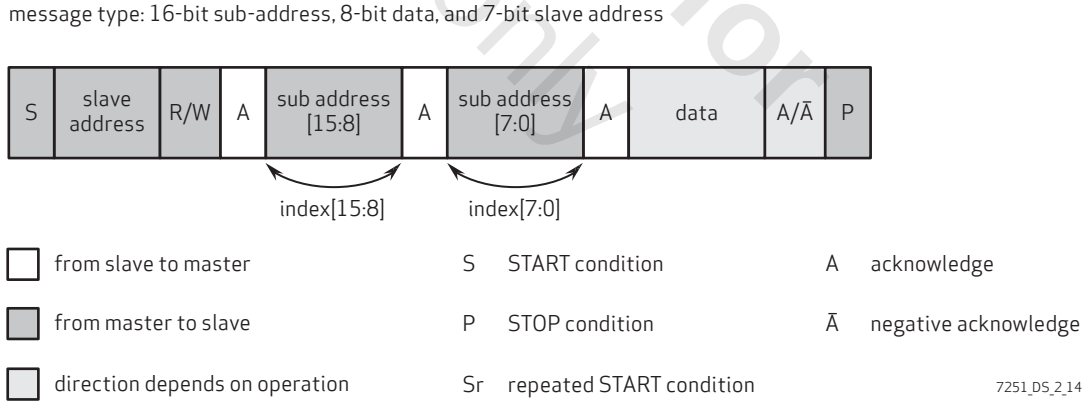
2.9.1 data transfer protocol

The data transfer of the OV7251 follows the SCCB protocol.

2.9.2 message format

The OV7251 supports the message format shown in **figure 2-14**. The repeated START (Sr) condition is not shown in SCCB single read from random location, but is shown in SCCB single read from current location and SCCB sequential read from random location.

figure 2-14 message type



2.9.3 read / write operation

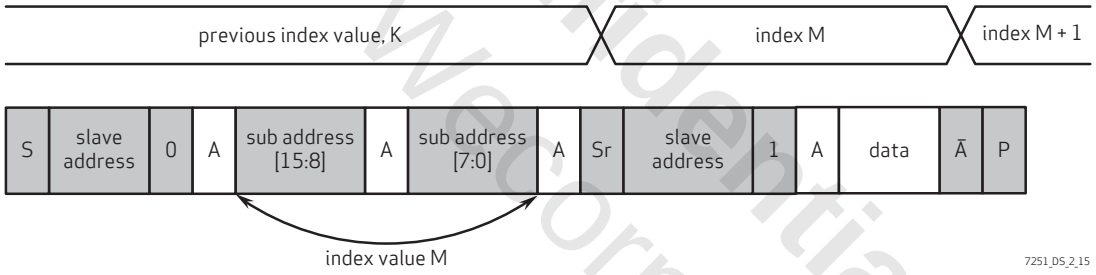
The OV7251 supports four different read operations and two different write operations:

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

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

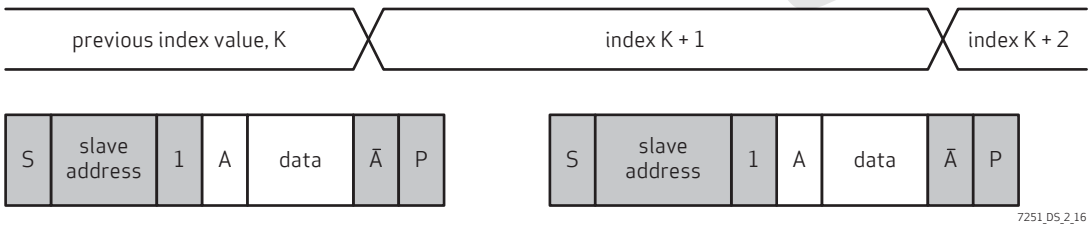
In a single read from random locations, the master does a dummy write operation to desired sub-address, issues a repeated start condition and then addresses the camera again with a read operation. After acknowledging its slave address, the camera starts to output data onto the SIOD line as shown in **figure 2-15**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-15 SCCB single read from random location



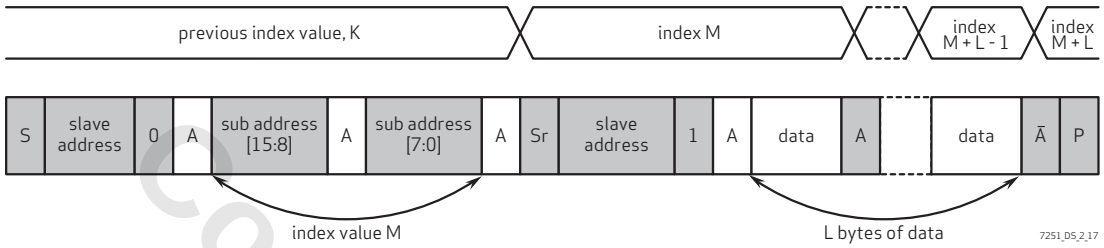
If the host addresses the camera with read operation directly without the dummy write operation, the camera responds by setting the data from last used sub-address to the SIOD line as shown in **figure 2-16**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-16 SCCB single read from current location



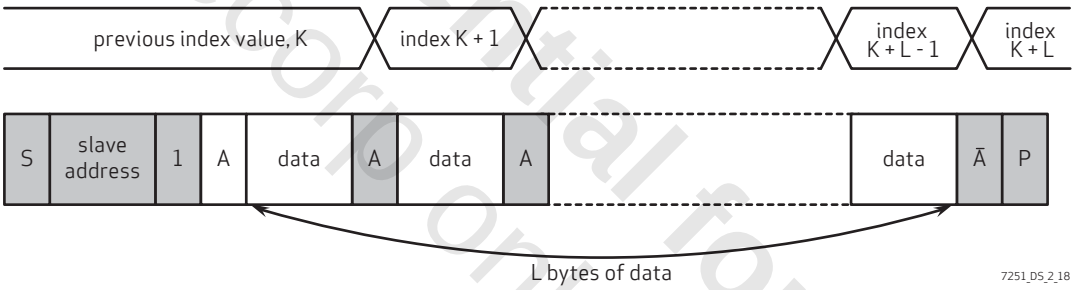
The sequential read from a random location is illustrated in **figure 2-17**. The master does a dummy write to the desired sub-address, issues a repeated start condition after acknowledge from slave and addresses the slave again with read operation. If a master issues an acknowledge after receiving data, it acts as a signal to the slave that the read operation shall continue from the next sub-address. When master has read the last data byte, it issues a negative acknowledge and stop condition.

figure 2-17 SCCB sequential read from random location



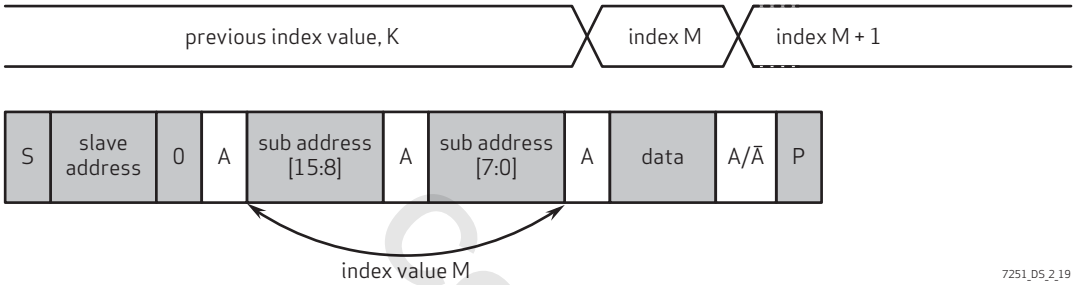
The sequential read from current location is similar to a sequential read from a random location. The only exception is that there is no dummy write operation as shown in **figure 2-18**. The master terminates the read operation by setting a negative acknowledge and stop condition.

figure 2-18 SCCB sequential read from current location



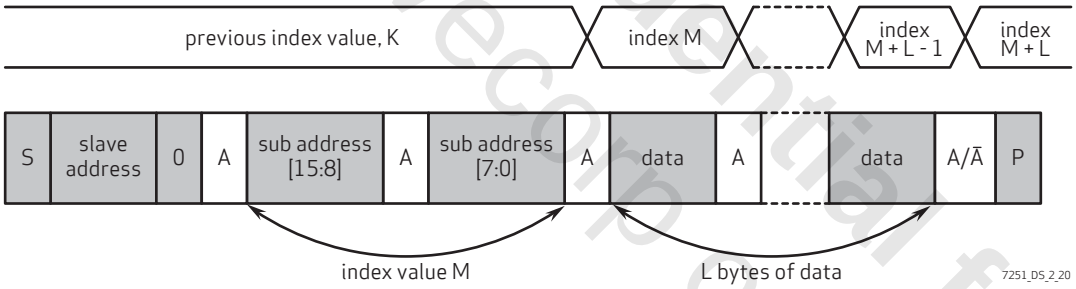
The write operation to a random location is illustrated in **figure 2-19**. The master issues a write operation to the slave, sets the sub-address and data correspondingly after the slave has acknowledged. The write operation is terminated with a stop condition from the master.

figure 2-19 SCCB single write to random location



The sequential write is illustrated in **figure 2-20**. The slave automatically increments the sub-address after each data byte. The sequential write operation is terminated with stop condition from the master.

figure 2-20 SCCB sequential write to random location



2.9.4 SCCB timing

figure 2-21 SCCB interface timing

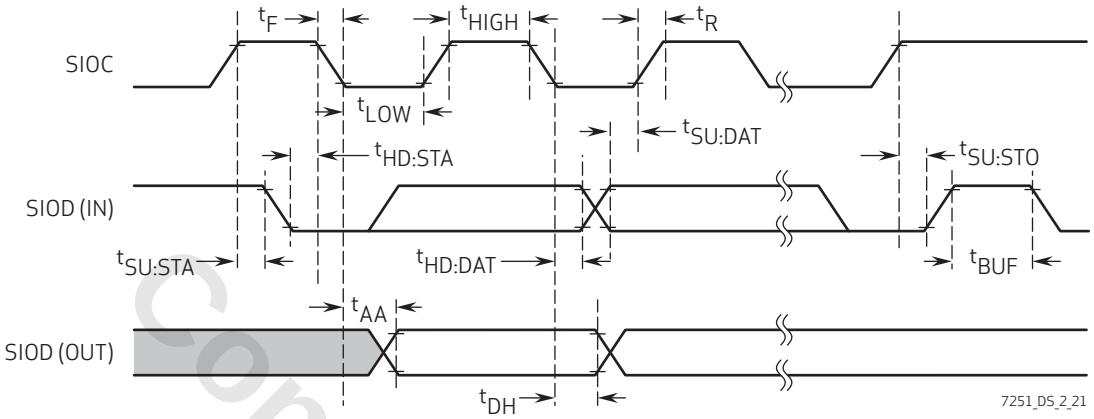


table 2-12 SCCB interface timing specifications[ab]

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

a. SCCB timing is based on 400kHz mode

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

2.9.5 group write and fast mode switching

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

The sensor can store up to two groups of registers in its buffer. The total buffer size is 128 bytes, which is divided into two for the two groups respectively. By default, the buffer for each group is 64 bytes. The buffer size for each group is programmable. The start address of the first group is set by register {0x3280[3:0], 4'h0} and the start address of the second group is set by register {0x3281[3:0], 4'h0}.

The sensor shuts down the clock of some functional blocks, including the group write function, during a long vertical blanking period. To always enable the clock for group write function, register bit 0x301C[5] should be set to 1.

Group write usually consists of register recording and launching. The register recording is started by writing 4'h0 to register bits 0x3208[7:4] and the group ID to be recorded is set by register bits 0x3208[3:0], which should be either 4'h0 or 4'h1. After recording all registers to be programmed, write 4'h1 to register bits 0x3208[7:4] with the group ID set in register bits 0x3208[3:0]. The following is an example sequence to program the gain and exposure time using group 0.

```
c0 3208 00
c0 350b 10
c0 3501 1f
c0 3502 80
c0 3208 10
```

To program the register setting stored in the buffer into the registers, write 4'hA (other values are only for debug purposes) to register 0x3208[7:4] with the group ID set in 0x3208[3:0].

```
c0 3208 a0
```

Group write can be used to switch two groups of settings repeatedly. In this case, the two groups of registers are recorded first and then the number of frames for each group of register setting are set by registers 0x3209 and 0x320A, respectively. Set register 0x320B to 0x0D to let the group write function program the two groups of registers repeatedly once register 0x3208 is set to 0xA0. To stop this repeated register programming, set register bit 0x301C[1] to 1 and then back to 0.

table 2-13 context switching control

address	register name	default value	R/W	description
0x3208	GROUP ACCESS	–	W	Group Access Bit[7:4]: group_ctrl 0000: Group hold start 0001: Group hold end 1010: Group launch 1110: Immediate launch others: Reserved Bit[3:0]: group_id 0000: Group bank 0, default start from address 0x00 0001: Group bank 1, default start from address 0x40 others: Reserved
0x3209	GRP0_PERIOD	0x00	RW	Frames For Staying in Group 0
0x320A	GRP1_PERIOD	0x00	RW	Frames For Staying in Group 1
0x320B	GRP_SWCTRL	0x01	RW	Bit[3]: group_switch_repeat Bit[2]: Group switch enable 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

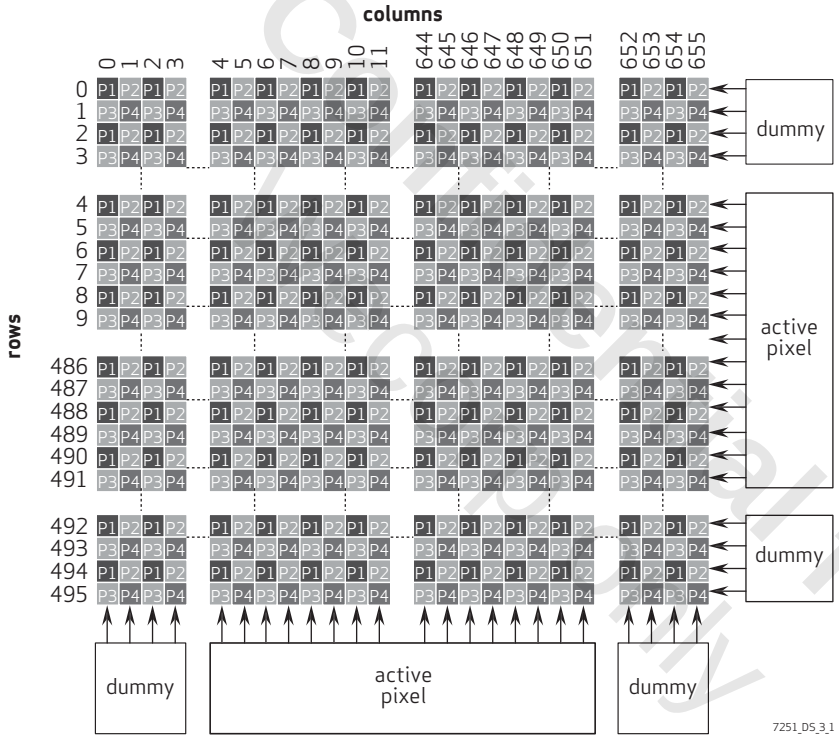
3 block level description

3.1 pixel array structure

The OV7251 sensor has an image array of 656 columns by 496 rows (325,376 pixels). **figure 3-1** shows a cross-section of the image sensor array.

Of the 325, 376 pixels, 316,224 (648x488) are active pixels and can be output. The other pixels are used for black level calibration and interpolation.

figure 3-1 sensor array layout



3.2 subsampling

There are two subsampling modes in the OV7251: binning and skipping. Both are acceptable methods of reducing output resolution while maintaining the field of view. Binning is usually preferred as it increases the pixel's signal-to-noise ratio. When the binning function is ON, voltage levels of a pair of the same pixels (e.g., P1 and P1 pixels, or P2 and P2 pixels) are averaged. In skipping mode (binning function is OFF), alternate pixels, which are not output, are merely skipped. The OV7251 supports 2x2 binning. **figure 3-2** illustrates 2x2 binning, where the voltage levels of two horizontal (2x1) pixels, which are the same, such as P3 and P3 pixels are averaged.

figure 3-2 example of 2x2 binning

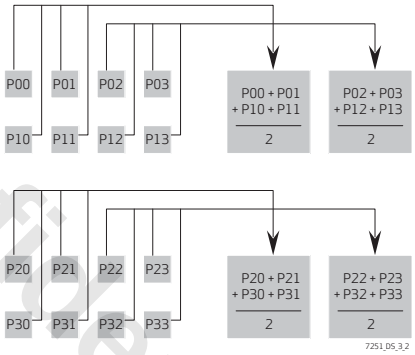


figure 3-3 example of 2:1 subsampling

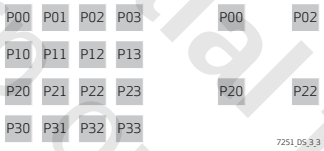


figure 3-4 example of 4:1 subsampling

table 3-1 binning-related registers

address	register name	default value	R/W	description
0x3820	TIMING_FORMAT1	0x00	RW	Bit[1]: Vertical binning
0x3821	TIMING_FORMAT2	0x00	RW	Bit[0]: Horizontal binning

4 image sensor core digital functions

4.1 mirror and flip

The OV7251 provides mirror and flip readout modes, which respectively reverse the sensor data readout order horizontally and vertically (see **figure 4-1**).

figure 4-1 mirror and flip samples

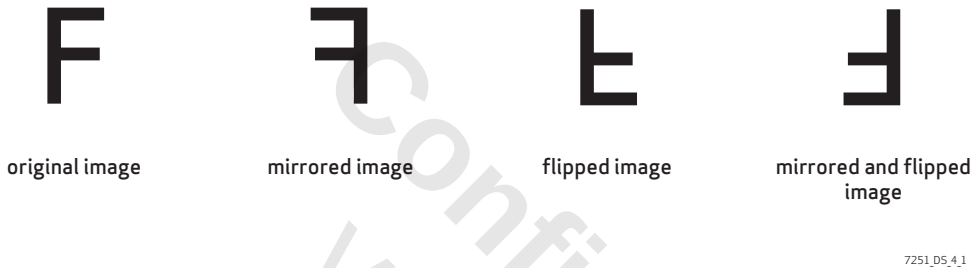


table 4-1 mirror and flip registers

address	register name	default value	R/W	description
0x3820	IMAGE_ ORIENTATION	0x00	RW	Timing Control Register Bit[2]: Vertical flip enable 0: Normal 1: Vertical flip
0x3821	IMAGE_ ORIENTATION	0x00	RW	Timing Control Register Bit[2]: Horizontal mirror enable 0: Normal 1: Horizontal mirror

4.2 image windowing

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

figure 4-2 image windowing

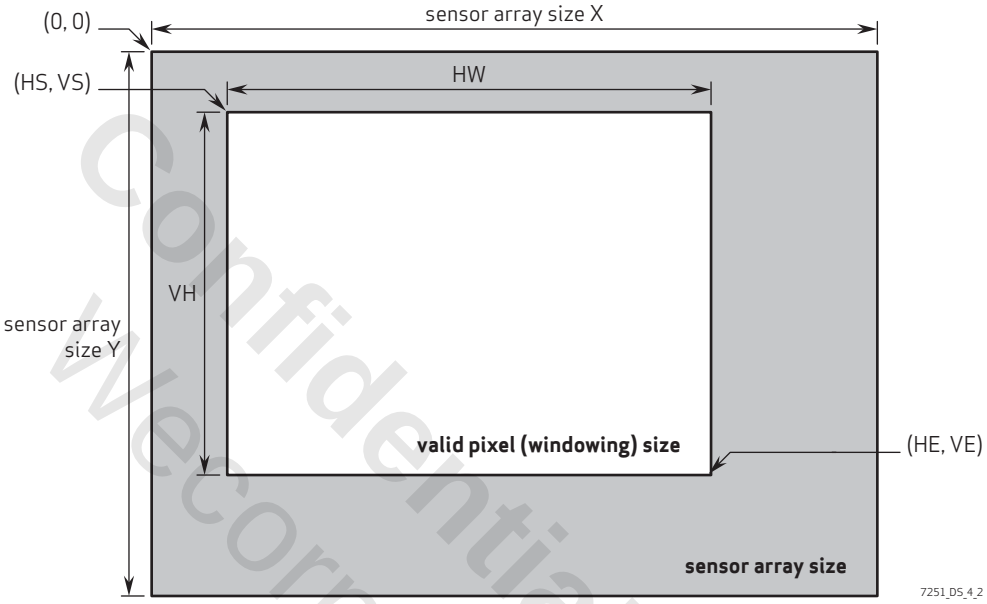


table 4-2 image windowing control functions

function	register	R/W	description
horizontal start	{0x3800, 0x3801}	RW	HS[9:8] = 0x3800 HS[7:0] = 0x3801
vertical start	{0x3802, 0x3803}	RW	VS[9:8] = 0x3802 VS[7:0] = 0x3803
horizontal end	{0x3804, 0x3805}	RW	HE[9:8] = 0x3804 HE[7:0] = 0x3805
vertical end	{0x3806, 0x3807}	RW	VE[9:8] = 0x3806 VE[7:0] = 0x3807

4.3 test pattern

For testing purposes, the OV7251 offers three test patterns:

4.3.1 general test pattern bar

figure 4-3 test pattern

table 4-3 general test pattern bar selection control

function	register	default value	R/W	description
general test pattern bar	0x5E00	0x0C	RW	Bit[7]: Test pattern bar enable

4.3.2 solid test pattern

table 4-4 solid test pattern control (sheet 1 of 2)

function	register	default value	R/W	description
solid test pattern	0x4320	0x80	RW	Bit[7:6]: Pixel order 00: P3P4/P1P2 01: P4P3/P2P1 10: P1P2/P3P4 11: P2P1/P4P3 Bit[1]: Solid test pattern enable 0: Solid test pattern OFF 1: Solid test pattern enable Bit[0]: Debug control
solid pattern P1	0x4322	0x00	RW	Bit[1:0]: solid_testpattern_P1[9:8]
solid pattern P1	0x4323	0x00	RW	Bit[7:0]: solid_testpattern_P1[7:0]
solid pattern P2	0x4324	0x00	RW	Bit[1:0]: solid_testpattern_P2[9:8]
solid pattern P2	0x4325	0x00	RW	Bit[7:0]: solid_testpattern_P2[7:0]

table 4-4 solid test pattern control (sheet 2 of 2)

function	register	default value	R/W	description
solid pattern P4	0x4326	0x00	RW	Bit[1:0]: solid_testpattern_P4[9:8]
solid pattern P4	0x4327	0x00	RW	Bit[7:0]: solid_testpattern_P4[7:0]
solid pattern P3	0x4328	0x00	RW	Bit[1:0]: solid_testpattern_P3[9:8]
solid pattern P3	0x4329	0x00	RW	Bit[7:0]: solid_testpattern_P3[7:0]

4.4 black level calibration (BLC)

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

Black level adjustments can be made with registers 0x4000, 0x4001, 0x4002, 0x4003, 0x4004 and 0x4009.

table 4-5 BLC control functions (sheet 1 of 2)

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0x85	RW	Bit[0]: BLC enable 0: Disable 1: Enable
0x4001	BLC CTRL 01	0xC2	RW	Bit[7]: Slope apply enable 0: Disable 1: Enable Bit[5:0]: BLC start line number
0x4002	BLC_AUTO	0x45	RW	Bit[7]: Format change enable 0: BLC will remain the same after format change 1: BLC will redo after format change Bit[6]: BLC auto enable 0: Get black level manually from register 1: Calculate black level from auto statistics Bit[5:0]: Reset frame number Frames that will continue to go through BLC after reset

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

address	register name	default value	R/W	description
0x4003	BLC_FREEZE	0x08	RW	Bit[7]: BLC redo enable 0: Normal 1: Force BLC to redo N frames (where N=0x4003[5:0]) when this bit is set Bit[6]: Freeze enable 0: Normal 1: BLC black level will not update. Priority lower than always update. Bit[5:0]: Manual frame number BLC redo frame number
0x4004	BLC NUM	0x04	RW	Bit[7:6]: Reserved Bit[5:0]: Number of black lines used
0x4009	BLC_TARGET	0x10	RW	Bit[7:0]: Black target level[7:0]

4.5 one time programmable (OTP) memory

The OV7251 has 256-bit embedded one time programmable (OTP) memory. The OTP memory can be programmed and read back via SCCB bus. This document provides general guidelines for programming and accessing the OTP memory.

4.5.1 OTP memory structure

128 bits of OTP memory are reserved for OmniVision internal use. These bits are usually used to store the production information or used by the some internal functions. The remaining 128 bits are fully user programmable. The user can store the production tracking information, the camera module calibration data, etc. to these bits.

table 4-6 OTP memory structure

OTP bits	function
[127:0]	reserved by OmniVision for internal use
[255:128]	user programmable

4.5.2 accessing the OTP memory

The OTP memory cannot be accessed directly. Instead, it is accessed through its register buffer 0x3D00~0x3D1F as shown in **figure 4-4**. The clock of the OTP memory controller is enabled by register bit 0x3018[4], which should be set to 1 to enable OTP access; otherwise, the sensor may automatically turn off the clock during long blanking periods to save power.

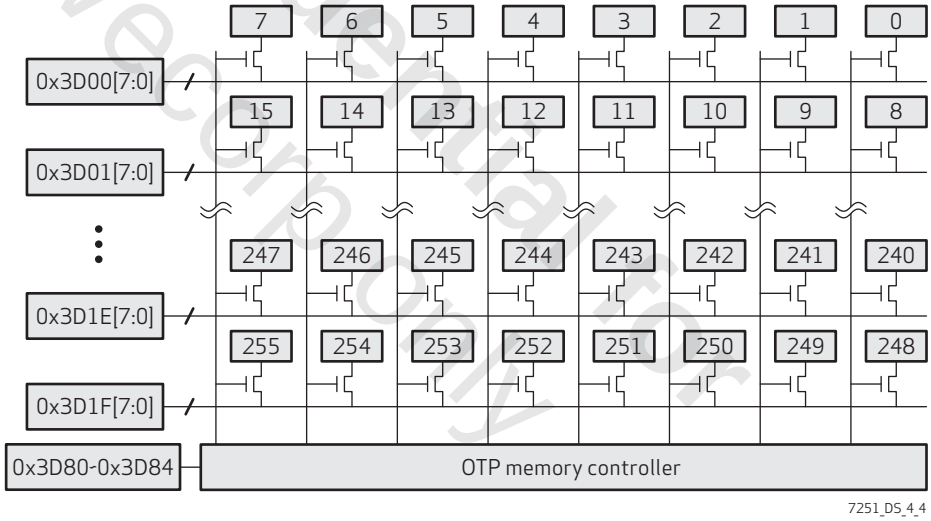
When 0x01 is programmed to register 0x3D81, the OTP memory controller will load the content of all OTP memory bits to its corresponding register buffer. After that, the user can read the OTP content from its register buffer. It is recommended to clear the register buffer to zero before loading the OTP.

When value 0x01 is programmed to register 0x3D80, the OTP memory controller will program the data of register 0x3D00~0x3D1F to its corresponding OTP memory bits. Keep in mind, the memory is one time programmable. It cannot be programmed back to 0 once it is programmed to 1. In fact, the OTP memory controller only programs those bits with value 1 in its corresponding memory buffer when programming command is issued. Multi-pass programming is allowed. However, programming 1 to an OTP bit already programmed to 1in previous pass is prohibited. The user should always program a bit from 0 to 1 only in any programming pass.

OTP access is in system clock domain, so register 0x100 has to be set to 1 to enable system clock PLL in order to access OTP. The OTP programming pulse width is controlled by register 0x3D82 and the unit is 8 system clock periods. The default value of 0x65 is for 48MHz system clock and the programming pulse width is 16.8µs. The OTP read pulse width is set by register 0x3D83 and the default value of 0x05 gives 104ns at 48MHz system clock. When the system clock frequency is different, the programming pulse should set to the closest value to 10µs and should be greater than 9µs. The system clock frequency is dependent on the input clock and PLL configuration. Refer to **section 2.8** for details.

When the OTP memory controller is programming data to OTP memory or reading data from OTP memory, the sensor will not respond to any SCCB access. Because OTP programming current is quite high, accessing sensor register is prohibited in order to prevent any glitch on the power supply. It is recommended to wait 15ms after issuing the OTP read and program command. This delay should scaled with the system clock period.

figure 4-4 OTP access



7251_DS_4_4

4.5.3 procedure for accessing OTP memory

Since the OTP memory can only be programmed once, the user should be very careful when accessing the OTP. Here is a detailed procedure for OTP access.

4.5.4 procedure to read OTP content

1. Clear software buffer which is to receive the OTP content.
2. Configure PLL, set register 0x100 to 1 if not yet set, and set register 0x3018 to 0x10.
3. Clear register buffer 0x3D00~0x3D1F to 0x00.
4. Set register 0x3D81 to 0x01.
5. Wait 15ms.
6. Read register 0x3D00~0x3D1F and set to the software buffer.

The OTP read operation is performed to verify the OTP memory is blank before program data to it, or to verify OTP contents after programming data to it.

Verifying the OTP content at the last step of camera module testing is highly recommended in case the OTP content is accidently overwritten during the module testing.

4.5.5 procedure to program OTP content

1. Follow **procedure to read OTP content** to make sure the OTP to be programmed is blank.
2. Program the intended OTP content to its corresponding register buffer, and clear unused register buffer to 0.
   a. For customer - registers 0x3D00~0x3D0F must be cleared to 0x00 before initiating the OTP programming command
3. Read back registers 0x3D00~0x3D1F to make sure they are the correct data to program to OTP memory or 0 for all other bits.
4. Write 0x01 to register 0x3D80 to initiate OTP programming.
5. Wait 15ms, any register access during this period is prohibited.
6. Follow **procedure to read OTP content** to read back the OTP content.
7. Compare the OTP content read back to the intended OTP content.

4.5.6 power supply requirement for OTP memory programming

The OTP memory is programmed using the analog power. The AVDD voltage for OTP programming must be 2.6V ~ 3.0V. The power supply should be able to provide extra 50mA for OTP programming.

4.6 pulse width modulation (PWM)

The PWM uses the pad clock input of 6~27MHz and generates a waveform with programmable frequency and duty cycle.

Frequency: frequency of pad clock divided by (1~65535)

Duty cycle: 0~100%

The PWM output is calculated as follows:

frequency = input_pad_clk_freq / pwm_freq_div_cycle_reg

duty cycle = (pwm_duty_cycle_reg /pwm_freq_div_cycle_reg) × 100%

figure 4-5 PWM output timing

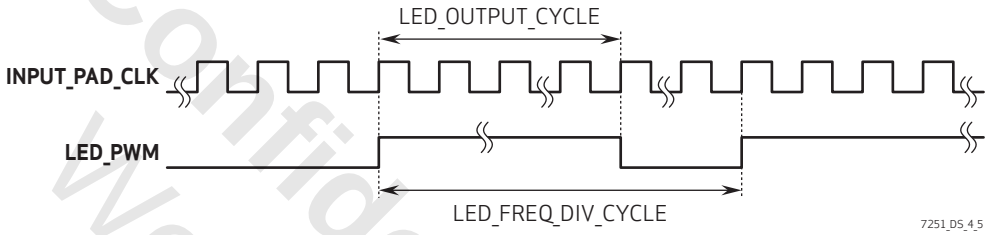


table 4-7 PWM registers

address	register name	default value	R/W	description
0x3B82	LED_PWM_REG02	0x10	RW	Bit[7:0]: pwm_freq_div_cycle_reg[15:8]
0x3B83	LED_PWM_REG03	0x00	RW	Bit[7:0]: pwm_freq_div_cycle_reg[7:0]
0x3B84	LED_PWM_REG04	0x08	RW	Bit[7:0]: pwm_duty_cycle_reg[15:8]
0x3B85	LED_PWM_REG05	0x00	RW	Bit[7:0]: pwm_duty_cycle_reg[7:0]

4.7 strobe

Strobe facilitates implementation of a flashlight. Strobe generates a pulse with a reference starting point at the time when the pixel array starts integration. Following a delay after the reference starting point, which is controlled by strobe_frame_shift_direction, strobe_frame_shift[30:0], a pulse with a width of strobe_frame_span[31:0] is generated. The step width of shift and span is programmable under system clock domain.

table 4-8 strobe control registers

address	register name	default value	R/W	description
0x3B88	LED_PWM_REG08	0x00	RW	Bit[7]: Shift direction Bit[6:0]: strobe_frame_shift[30:24]
0x3B89	LED_PWM_REG09	0x00	RW	Bit[7:0]: strobe_frame_shift[23:16]
0x3B8A	LED_PWM_REG0A	0x00	RW	Bit[7:0]: strobe_frame_shift[15:8]
0x3B8B	LED_PWM_REG0B	0x05	RW	Bit[7:0]: strobe_frame_shift[7:0]
0x3B8C	LED_PWM_REG0C	0x00	RW	Bit[7:0]: strobe_frame_span[31:24]
0x3B8D	LED_PWM_REG0D	0x00	RW	Bit[7:0]: strobe_frame_span[23:16]
0x3B8E	LED_PWM_REG0E	0x00	RW	Bit[7:0]: strobe_frame_span[15:8]
0x3B8F	LED_PWM_REG0F	0x1A	RW	Bit[7:0]: strobe_frame_span[7:0]
0x3B90	LED_PWM_REG10	0x01	RW	Bit[7:0]: r_strobe_row_st[15:8]
0x3B91	LED_PWM_REG11	0xB4	RW	Bit[7:0]: r_strobe_row_st[7:0]
0x3B92	LED_PWM_REG12	0x00	RW	Bit[7:0]: r_strobe_cs_st[15:8]
0x3B93	LED_PWM_REG13	0x10	RW	Bit[7:0]: r_strobe_cs_st[7:0]
0x3B94	LED_PWM_REG14	0x05	RW	Bit[7:0]: step_onerow_man[15:8]
0x3B95	LED_PWM_REG15	0xF2	RW	Bit[7:0]: step_onerow_man[7:0]
0x3B96	LED_PWM_REG16	0x40	RW	Bit[7]: r_strobe_frm_pwen Bit[6]: r_strobe_frm_pwst Bit[5]: r_strobe_pol Bit[4]: r_strobe_step_pix Bit[3]: r_step_onerow_precision_man Bit[2:0]: r_strobe_st_opt

4.8 low power modes

The OV7251 sensor supports three low power modes:

- low frame rate streaming mode
- internal trigger snapshot mode
- external trigger snapshot mode

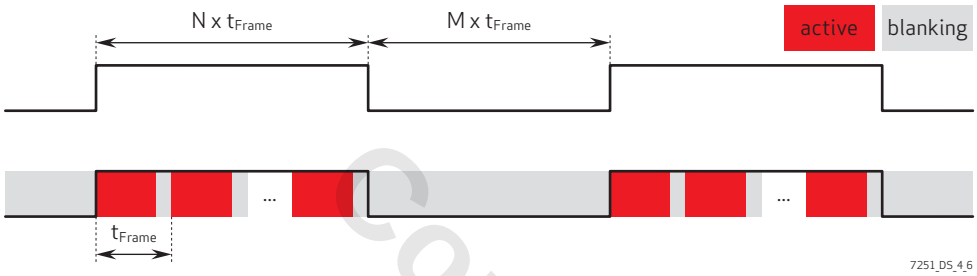
table 4-9 low power mode control registers

register	description
0x3C00	debug control for low power mode, maintains default value of 0x89 all the time.
0x3C01	Power Control Options 0x63: low power mode 0xAB: normal mode
0x3C02	Bit[1]: idle phase enable Bit[0]: streaming phase enable 00: not allowed 01: normal streaming mode 10: not allowed 11: enable low power streaming mode
0x3C03	Low Power Mode Control Bit[7]: Trigger for internal trigger snapshot mode A rising edge on 0x3C03[7] wakes the sensor up and streams out {0x3404, 0x3405} frames Bit[6:0]: Mode control 0x00: Low frame rate streaming mode (i.e., repeating the sequence of streaming {0x3404, 0x3405} frames and then sleeping {0x3C06, 0x3C07} frame) 0x17: External trigger snapshot mode A rising edge on FSIN pin wakes the sensor up and streams out {0x3404, 0x3405} frames 0x03: Internal trigger snapshot mode A rising edge on 0x3C03[7] wakes the sensor up and streams out {0x3404, 0x3405} frames Others: For debug only
{0x3C04, 0x3C05}	number of active frames
{0x3C06, 0x3C07}	number of idle frames
0x3C08~0x3C0B	not used
{0x3C0C, 0x3C0D}	row period in units of input clock period
{0x3C0E, 0x3C0F}	number of rows per base frame, usually set to the same value as {0x380E, 0x380F}
0x3023	Bit[1}: MIPI power down enable during sleep period 0: disable for low power streaming mode

4.8.1 low frame rate mode

In low frame rate mode, the OV7251 sensor streams N frames, idles for M frames, and then repeats. The power consumption of the OV7251 sensor is close to N/(N+M) of the current in full speed streaming mode but the maximum integration time is limited to about t_{Frame} - $40t_{Row}$.

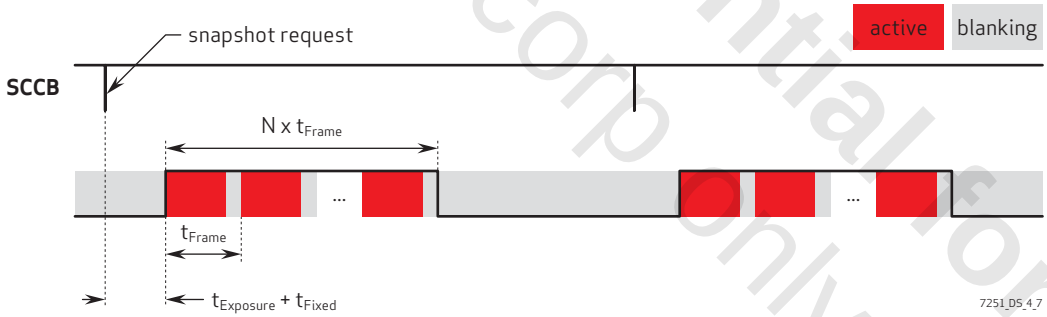
figure 4-6 low frame rate mode timing



4.8.2 snapshot mode

In snapshot mode, the OV7251 streams N frames upon request through the SCCB and then stays idle until the next request (see **figure 4-7**).

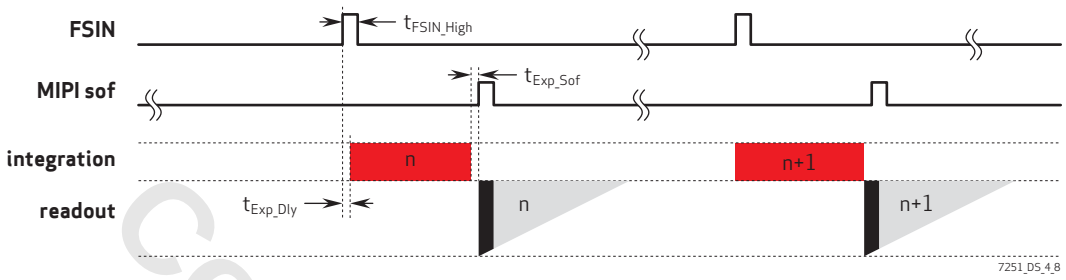
figure 4-7 snapshot mode timing



4.8.3 external trigger snapshot mode

Upon the rising edge of FSIN pulse, the sensor wakes up from sleep mode, starts integration, reads out and sends out number (set by 0x3C04, 0x3C05) of frames. The sensor then returns back to sleep mode (see **figure 4-8**).

figure 4-8 external snapshot mode timing



FSIN pulse width, t_{FSIN_High}, should be no shorter than 5 input clock cycles. The wake up sequence takes 16384 input clock cycles, and then the pixel array is reset. The integration starts when the pixel reset finish, the interval from FSIN rising to integration, t_{Exp_Dly}, equals to $16388 \times t_{XVCLK} + 11t_{Row}$. The frame start short packet is sent out about 8 row periods after integration finishes.

The reference voltage of VN2 is critical for image quality in this mode. It is recommended to have a 1µF cap on this pin to keep the voltage after the sensor goes to sleep mode between two adjacent triggers in a burst. If the sleep period is too long between bursts (e.g., more than 100ms), please discard the first triggered frame (e.g., the frame triggered by the red pulse) as shown in **figure 4-9**. The second frame can be triggered as early as the first frame finishes. It is recommended to keep the frame rate within the burst no less than 10 fps to prevent VN2 discharging too much.

figure 4-9 frame triggered by red pulse diagram

5 image sensor processor digital functions

5.1 ISP general controls

table 5-1 ISP top registers

address	register name	default value	R/W	description
0x5000	ISP CTRL00	0x85	RW	Bit[2]: window_enable 0: Disable 1: Enable Bit[1]: awb_gain_en 0: Disable 1: Enable Bit[0]: blc_enable 0: Disable 1: Enable
0x5001	ISP CTRL 01	0x00	RW	Bit[1]: Bypass ISP option 1 1: When bypass ISP option 0 is disabled, will output data after awb_gain Bit[0]: Bypass ISP option 0 1: Output data directly from ISP input

5.2 manual white balance (MWB)

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

table 5-2 manual AWB_gain registers

address	register name	default value	R/W	description
0x3400	AWB RED GAIN	0x04	RW	Bit[3:0]: AWB red gain[11:8]
0x3401	AWB RED GAIN	0x00	RW	Bit[7:0]: AWB red gain[7:0]
0x3402	AWB GRN GAIN	0x04	RW	Bit[3:0]: AWB green gain[11:8]
0x3403	AWB GRN GAIN	0x00	RW	Bit[7:0]: AWB green gain[7:0]
0x3404	AWB BLU GAIN	0x04	RW	Bit[3:0]: AWB blue gain[11:8]
0x3405	AWB BLU GAIN	0x00	RW	Bit[7:0]: AWB blue gain[7:0]
0x3406	AWB MAN CTRL	0x01	RW	Bit[0]: AWB manual control

5.3 manual exposure and gain control

table 5-3 manual exposure and gain control registers

address	register name	default value	R/W	description
0x3500	AEC EXPO	0x00	RW	Exposure Bit[7:4]: Not used Bit[3:0]: Exposure[15:12]
0x3501	AEC EXPO	0x00	RW	Exposure Bit[7:0]: Exposure[11:4] Minimum exposure time is 1 row period. Maximum exposure time is frame length - 20 row periods, where frame length is set by registers {0x380E, 0x380F}.
0x3502	AEC EXPO	0x00	RW	Exposure Bit[7:4]: Exposure[3:0] Minimum exposure time is 1 row period. Maximum exposure time is frame length - 20 row periods, where frame length is set by registers {0x380E, 0x380F}. Bit[3:0]: Debug control
0x3503	MANUAL CONTROL	0x00	RW	Bit[1]: Gain manual enable Bit[0]: Exposure manual enable
0x350B	GAIN	0x10	RW	Bit[7:0]: Gain[7:0]

6 system control

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

6.1 mobile industry processor interface (MIPI)

The OV7251 MIPI interface supports a single uni-directional clock lane and a single uni-directional data lane. The data lane has full support for high speed (HS) data transfer. Contact your local OmniVision FAE for more details.

table 6-1 MIPI top control registers (sheet 1 of 8)

address	register name	default value	R/W	description
0x4800	MIPI CTRL 00	0x44	RW	MIPI Control 00 Bit[6]: ck_mark1_en 1: Enable clock lane mark1 when resume Bit[5]: Clock lane gate enable 0: Clock lane is free running 1: Gate clock lane when no packet to transmit Bit[4]: Line sync enable 0: Do not send line short packet for each line 1: Send line short packet for each line Bit[2]: Idle status 0: MIPI bus will be LP00 when no packet to transmit 1: MIPI bus will be LP11 when no packet to transmit Bit[0]: clk_lane_dis 1: Set clock lane to LP mode manually

table 6-1 MIPI top control registers (sheet 2 of 8)

address	register name	default value	R/W	description
0x4801	MIPI CTRL 01	0x03	RW	MIPI Control 01 Bit[7]: Long packet data type manual enable 0: Use mipi_dt 1: Use dt_man_o as long packet data (see register 0x4814[5:0]) Bit[6]: Short packet data type manual enable 1: Use dt_spkt as short packet data (see register 0x4815[5:0]) Bit[5]: Short packet wc select 0: Use frame counter or line counter 1: Select spkt_wc_reg_o ({0x4812, 0x4813}) Bit[4]: PH bit order for ECC 0: {DI[7:0], WC[7:0], WC[15:8]} 1: {DI[0:7], WC[0:7], WC[8:15]} Bit[3]: PH byte order for ECC 0: {DI, WC_l, WC_h} 1: {DI, WC_h, WC_l} Bit[2]: PH byte order2 for ECC 0: {DI, WC} 1: {WC, DI} Bit[1]: mark1_en 1: When mipi_sys_susp = 1, lane1 sends mark1 from wkup_dly_o after each reset release Bit[0]: Debug control

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

address	register name	default value	R/W	description
0x4802	MIPI CTRL 02	0x00	RW	MIPI Control 02 Bit[7]: hs_prepare_sel 0: Auto calculate T_hs_prepare, unit: pclk2x 1: Use hs_prepare_min_o[7:0] Bit[6]: clk_prepare_sel 0: Auto calculate T_clk_prepare, unit: pclk2x 1: Use clk_prepare_min_o[7:0] Bit[5]: clk_post_sel 0: Auto calculate T_clk_post, unit: pclk2x 1: Use clk_post_min_o[7:0] Bit[4]: clk_trail_sel 0: Auto calculate T_clk_trail, unit: pclk2x 1: Use clk_trail_min_o[7:0] Bit[3]: hs_exit_sel 0: Auto calculate T_hs_exit, unit: pclk2x 1: Use hs_exit_min_o[7:0] Bit[2]: hs_zero_sel 0: Auto calculate T_hs_zero, unit: pclk2x 1: Use hs_zero_min_o[7:0] Bit[1]: hs_trail_sel 0: Auto calculate T_hs_trail, unit: pclk2x 1: Use hs_trail.min_o[7:0] Bit[0]: clk_zero_sel 0: Auto calculate T_clk_zero, unit: pclk2x 1: Use clk_zero_min_o[7:0]
0x4803	MIPI CTRL 03	0x50	RW	MIPI Control 03 Bit[7:6]: lp_glitch_nu 0: Use 2d of lp_in 1: Mask one SCLK cycle glitch of lp_in Bit[5:4]: cd_glitch_nu 0: Use 2d of lp_cd_in 1: Mask one SCLK cycle glitch of lp_cd_in Bit[3]: cd1_int_en 0: Disable cd plus of data lane1 1: Enable cd plus of data lane1 Bit[2]: Debug control Bit[1]: lp_cd1_en 0: Disable cd of data_lane1 from PHY 1: Enable cd of data_lane1from PHY Bit[0]: Debug control

table 6-1 MIPI top control registers (sheet 4 of 8)

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

table 6-1 MIPI top control registers (sheet 5 of 8)

address	register name	default value	R/W	description
0x4805	MIPI CTRL 05	0x10	RW	MIPI Control 05 Bit[7]: Debug control Bit[6]: lane_disable1 1: Disable MIPI data lane1, lane1 will be LP00 Bit[5]: lpx_p_sel 0: Auto calculate t_lpx_o in pclk2x 1: Use lp_p_min[7:0] Bit[4]: lp_rx_intr_sel 0: Send lp_rx_intr_o as the first byte 1: Send lp_rx_intr_o at the end of receiving Bit[3]: cd_tst_sel 1: Select PHY test pins Bit[2]: mipi_reg_mask 1: Disable MIPI access SRB Bit[1]: clip enable Bit[0]: hd_sk_en 0: Disable MIPI and MCU hand shake registers 1: Enable MIPI and MCU hand shake registers
0x4806	MIPI CTRL 06	0x0F	RW	Bit[7]: mipi_test Bit[6]: prbs_en test mode Bit[3]: mipi_slp_man_st MIPI bus status manual control enable in sleep mode Bit[2]: clk_lane_state Bit[1]: Debug control Bit[0]: data_lane1_state
0x4810	MIPI MAX FRAME COUNT	0xFF	RW	High Byte of Maximum Frame Count of Frame Sync Short Packet
0x4811	MIPI MAX FRAME COUNT	0xFF	RW	Low Byte of Maximum Frame Count of Frame Sync Short Packet
0x4812	MIPI SHORT PKT COUNTER	0x00	RW	High Byte of Manual Short Packet Word Counter
0x4813	MIPI SHORT PKT COUNTER	0x00	RW	Low Byte of Manual Short Packet Word Counter
0x4814	MIPI CTRL14	0x2A	RW	MIPI Control 14 Bit[7:6]: Virtual channel of MIPI Bit[5:0]: Data type in manual mode
0x4815	MIPI_DT_SPKT	0x40	RW	Bit[6]: pclk_inv 0: Use mipi_pclk_o rising edge 1: Use mipi_pclk_o falling edge

table 6-1 MIPI top control registers (sheet 6 of 8)

address	register name	default value	R/W	description
0x4818	HS_ZERO_MIN	0x00	RW	High Byte of Minimum Value for hs_zero, unit: ns
0x4819	HS_ZERO_MIN	0x9A	RW	Low Byte of Minimum Value for hs_zero, unit: ns hs_zero_real = hs_zero_min_o + Tui*ui_hs_zero_min_o
0x481A	HS_TRAIL_MIN	0x00	RW	High Byte of Minimum Value for hs_trail, unit: ns
0x481B	HS_TRAIL_MIN	0x3C	RW	Low Byte of Minimum Value for hs_trail, unit: ns hs_trail_real = hs_trail_min_o + Tui*ui_hs_trail_min_o
0x481C	CLK_ZERO_MIN	0x01	RW	High Byte of Minimum Value for clk_zero, unit: ns
0x481D	CLK_ZERO_MIN	0x86	RW	Low Byte of Minimum Value for clk_zero, unit: ns clk_zero_real = clk_zero_min_o + Tui*ui_clk_zero_min_o
0x481E	CLK_PREPARE_MIN	0x00	RW	High Byte of Minimum Value for clk_prepare, unit: ns
0x481F	CLK_PREPARE_MIN	0x3C	RW	Low Byte of Minimum Value for clk_prepare, unit: ns clk_prepare_real = clk_prepare_min_o + Tui*ui_clk_prepare_min_o
0x4820	CLK_POST_MIN	0x00	RW	High Byte of Minimum Value for clk_post, unit: ns Bit[1:0]: clk_post_min[9:8]
0x4821	CLK_POST_MIN	0x56	RW	Low Byte of Minimum Value for clk_post, unit: ns clk_post_real = clk_post_min_o + Tui*ui_clk_post_min_o
0x4822	CLK_TRAIL_MIN	0x00	RW	High Byte of Minimum Value for clk_trail, unit: ns Bit[1:0]: clk_trail_min[9:8]
0x4823	CLK_TRAIL_MIN	0x3C	RW	Low Byte of Minimum Value for clk_trail clk_trail_real = clk_trail_min_o + Tui*ui_clk_trail_min_o
0x4824	LPX_P_MIN	0x00	RW	High Byte of Minimum Value for lpx_p, unit: ns Bit[1:0]: lpx_p_min[9:8]
0x4825	LPX_P_MIN	0x32	RW	Low Byte of Minimum Value for lpx_p, unit: ns lpx_p_real = lpx_p_min_o + Tui*ui_lpx_p_min_o
0x4826	HS_PREPARE_MIN	0x00	RW	High Byte of Minimum Value of hs_prepare, unit: ns
0x4827	HS_PREPARE_MIN	0x32	RW	Low Byte of Minimum Value for hs_prepare, unit: ns hs_prepare_real = hs_prepare_min_o + Tui*ui_hs_prepare_min_o
0x4828	HS_EXIT_MIN	0x00	RW	High Byte of Minimum Value for hs_exit, unit: ns Bit[1:0]: hs_exit_min[9:8]
0x4829	HS_EXIT_MIN	0x64	RW	Low Byte of Minimum Value for hs_exit, unit: ns hs_exit_real = hs_exit_min_o + Tui*ui_hs_exit_min_o
0x482A	UI_HS_ZERO_MIN	0x05	RW	Minimum UI Value of hs_zero, unit: UI

table 6-1 MIPI top control registers (sheet 7 of 8)

address	register name	default value	R/W	description
0x482B	UI_HS_TRAIL_MIN	0x04	RW	Minimum UI Value of hs_trail, unit: UI
0x482C	UI_CLK_ZERO_MIN	0x00	RW	Minimum UI Value of clk_zero, unit: UI
0x482D	UI_CLK_PREPARE_MIN	0x00	RW	Bit[5:4]: 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	Minimum UI Value of clk_post, unit: UI
0x482F	UI_CLK_TRAIL_MIN	0x00	RW	Minimum UI Value of clk_trail, unit: UI
0x4830	UI_LPX_P_MIN	0x00	RW	Minimum UI Value of lpx_p, unit: UI
0x4831	UI_HS_PREPARE_MIN	0x04	RW	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	Minimum UI Value of hs_exit, unit: UI
0x4833	MIPI_REG_MIN_H	0x00	RW	MIPI RW Register Address Lower Boundary High Byte
0x4834	MIPI_REG_MIN_L	0x00	RW	MIPI RW Register Address Lower Boundary Low Byte
0x4835	MIPI_REG_MAX_H	0xFF	RW	MIPI RW Register Address Top Boundary High Byte
0x4836	MIPI_REG_MAX_L	0xFF	RW	MIPI RW Register Address Top Boundary Low Byte
0x4837	PCLK_PERIOD	0x19	RW	Period of Pclk2x, pclk_div = 1, and 1-bit Decimal
0x4838	WKUP_DLY	0x02	RW	Wakeup Delay for MIPI (Mark1 state)/2^12
0x483A	DIR_DLY	0x08	RW	Change LP Direction Delay/2 After LP11
0x483B	MIPI_LP_GPIO	0x33	RW	Bit[7]: lp_sel1 0: Auto generate mipi_lp_dir1_o 1: Use lp_dir_man1 to be mipi_lp_dir1_o Bit[6]: lp_dir_man1 0: Input 1: Output Bit[5]: lp_p1_o Bit[4]: lp_n1_o Bit[3]: lp_sel2 0: Auto generate mipi_lp_dir2_o 1: Use lp_dir_man2 to be mipi_lp_dir2_o Bit[2]: lp_dir_man2 0: Input 1: Output Bit[1]: lp_p2_o Bit[0]: lp_n2_o

table 6-1 MIPI top control registers (sheet 8 of 8)

address	register name	default value	R/W	description
0x483C	MIPI_CTRL3C	0x4F	RW	Bit[7:4]: t_lpx Unit: SCLK cycle Bit[3:0]: t_clk_pre Unit: pclk2x cycle
0x483D	T_TA_GO	0x10	RW	Unit: sclk cycle
0x483E	T_TA_SURE	0x06	RW	Unit: sclk cycle
0x483F	T_TA_GET	0x14	RW	Unit: sclk cycle
0x4846	MIPI_CLIP_MAX	0x0F	RW	Bit[3:0]: MIPI output data max value[11:8]
0x4847	MIPI_CLIP_MAX	0xFF	RW	Bit[7:0]: MIPI output data max value[7:0]
0x4848	MIPI_CLIP_MIN	0x0F	RW	Bit[3:0]: MIPI output data min value[11:8]
0x4849	MIPI_CLIP_MIN	0xFF	RW	Bit[7:0]: MIPI output data min value[7:0]
0x4850	REG_INTR_MAN	–	W	Generate 1 SCLK Cycle Pulse for MCU Interrupt
0x4851	REG_TX_WR	–	W	Generate 1 SCLK Cycle Pulse to MIPI_TX_LP_TX, and reg_wdata Will Be Sent Out Through MIPI Escape Mode
0x4852	REG_TX_STOP	–	W	Generate 1 SCLK Cycle Pulse to MIPI_TX_LP_TX, and MIPI_TX_LP_TX Will Go Back to LP11
0x4853	REG_TA_ACK	–	W	Generate 1 SCLK Cycle Pulse to MIPI_TX_LP_TX to Receive TurnAround Command
0x4854	REG_TA_REQ	–	W	Generate 1 SCLK Cycle Pulse to MIPI_TX_LP_TX to Send TurnAround Command
0x4861	HD_SK_REG0	–	R	MIPI RW, SCCB and MCU Read Only
0x4862	HD_SK_REG1	–	R	MIPI RW, SCCB and MCU Read Only
0x4863	HD_SK_REG2	–	R	MIPI RW, SCCB and MCU Read Only
0x4864	HD_SK_REG3	–	R	MIPI RW, SCCB and MCU Read Only
0x4865	MIPI_ST	–	R	Bit[5]: lp_rx_sel_i 1: MIPI_LP_RX receiving LP data Bit[4]: tx_busy_i 1: MIPI_TX_LP_TX is busy sending LP data Bit[3]: mipi_lp_p1_i MIPI low power input for lane1 P Bit[2]: mipi_lp_n1_i MIPI low power input for lane1 N Bit[1:0]: Debug control

6.2 low-voltage differential signaling (LVDS)

LVDS is a common differential signaling interface that has low power consumption, minimal EMI, and excellent noise immunity. The maximum data rate for LVDS is 800 Mbps per lane.

Features include:

- supports 10-bit mode
- supports one lane mode
- supports manual setting sync code (can set different sync code for frame start/end and line start/end)
- supports manual setting dummy data in blanking duration
- supports PCLK inversion

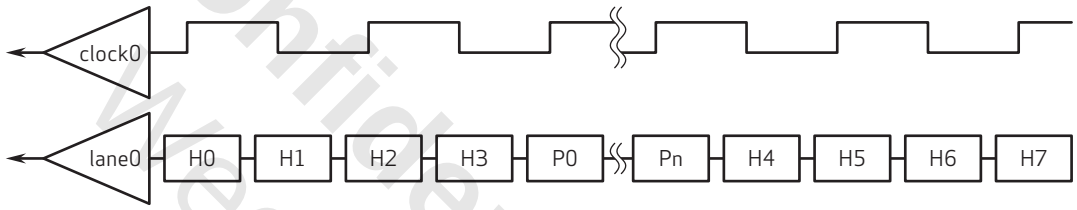
table 6-2 LVDS registers

address	register name	default value	RW	description
0x4A00	LVDS_R0	0x2A	RW	Bit[6]: SYNC code manual mode enable Bit[5]: SYNC code enable when only 1 lane Bit[4]: PCLK invert enable Bit[3]: Channel ID enable in sync per lane mode Bit[2]: F parameter in CCIR656 standard Bit[1]: SAV first enable Bit[0]: SYNC code mode 0: Split 1: Per lane
0x4A02	LVDS_R2	0x00	RW	Bit[7:0]: Dummy data0[15:8]
0x4A03	LVDS_R3	0x80	RW	Bit[7:0]: Dummy data0[7:0]
0x4A04	LVDS_R4	0x00	RW	Bit[7:0]: Dummy data1[15:8]
0x4A05	LVDS_R5	0x10	RW	Bit[7:0]: Dummy data1[7:0]
0x4A06	LVDS_R6	0xAA	RW	Blanking line_start Sync Code in Manual Sync Code Mode
0x4A07	LVDS_R7	0x55	RW	Blanking line_end Sync Code in Manual Sync Code Mode
0x4A08	LVDS_R8	0x99	RW	Video line_start Sync Code in Manual Sync Code Mode
0x4A09	LVDS_R9	0x66	RW	Video line_end Sync Code in Manual Sync Code Mode
0x4A0D	LVDS_R0D	0x00	RW	Bit[7:1]: Debug control Bit[0]: LVDS enable

The address of LVDS control registers are actually 0x4Ax0~0x4AxF respectively, where x means “don’t care”. However, there are also low power control registers at address 0x4A47, 0x4A49 and 0x4A4B, which the sensor does decode all address bits. When all of these six registers have to be programmed, low power control registers must be programmed first (which also program LVDS control registers) and then LVDS control registers (which does not program low power control register) using address 0x4Ax7, 0x4Ax9 and 0x4AxB, respectively, where x is any value other than 4. To read LVDS control registers, please use register address 0x4Ax7, 0x4Ax9 and 0x4AxB, respectively, where x is any value other than 4. However, the value returned from address 0x4A47, 0x4A49 and 0x4A4B are the bit-wise OR’ed value of register 0x4A47, 0x4A49, 0x4A4B and 0x4Ax7, 0x4Ax9 and 0x4AxB, respectively. To read back the true value of low power control registers, LVDS control registers value must be set to 0 first following the aforementioned sequence. Since register 0x4A07 and 0x4A09 are the manual sync code, it is recommended to select the default sync code by register 0x4A00[6] to avoid using them.

6.2.1 output modes

figure 6-1 LVDS 1-lane mode

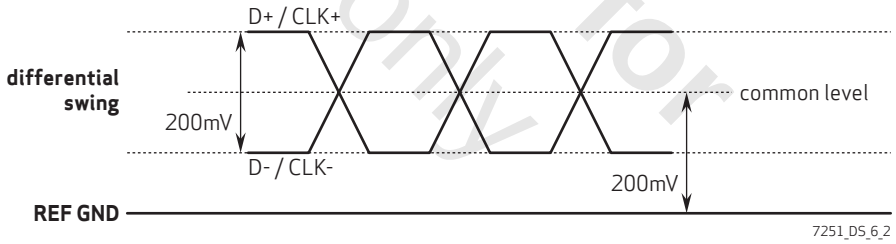


note H0-H7 = header (sync code) / P0-Pn = pixel

7251_DS_6_1

6.2.2 PHY specification

figure 6-2 PHY specification diagram



7251_DS_6_2

table 6-3 PHY specifications (sheet 1 of 2)

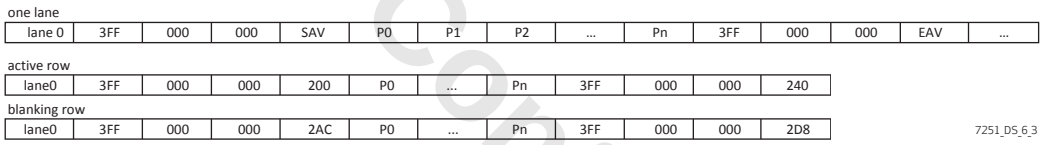
description	min	typ	max	unit
differential output	150	200	250	mV
common level output	150	200	250	mV

table 6-3 PHY specifications (sheet 2 of 2)

description	min	typ	max	unit
rise/fall time	150		0.3 UI	ps
data-clock skew	-0.15 UI		0.15 UI	ps
impedance	40	50	60	Ω

6.2.3 LVDS lane configuration and sync

figure 6-3 LVDS lane configuration and sync



7 register tables

The following tables provide descriptions of the device control registers contained in the OV7251. For all register enable/disable bits, ENABLE = 1 and DISABLE = 0. The device slave addresses are 0xC0 for write and 0xC1 for read.

7.1 system control [0x0100 - 0x010A, 0x3001 - 0x301F, 0x3023 - 0x303B]

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

address	register name	default value	R/W	description
0x0100	SC_MODE_SELECT	0x00	RW	Bit[7:1]: Not used Bit[0]: Mode select 0: software_standby 1: Streaming
0x0101~ 0x0102	RSVD	–	–	Reserved
0x0103	SC_SOFTWARE_ RESET	0x00	RW	Bit[7:1]: Not used Bit[0]: software_reset 0: Off 1: On
0x0104~ 0x0105	RSVD	–	–	Reserved
0x0106	SC_FAST_STANDBY _CTRL	0x01	RW	Bit[7:1]: Not used Bit[0]: fast_standby 0: Frame completes before mode entry 1: Frame may be truncated before mode entry
0x0107~ 0x0108	NOT USED	–	–	Not Used
0x0109	SC_SCCB_ID1	0xC0	RW	SCCB ID 0
0x010A	NOT USED	–	–	Not Used
0x3001	SC_REG1	0x02	RW	Bit[7]: pd_dato_en Bit[6:5]: Output pad drive strength 00: 1x 01: 2x 10: 3x 11: 4x Bit[4:0]: Debug control
0x3002~ 0x3003	RSVD	–	–	Reserved
0x3004	SC_REG4	0x00	RW	Bit[7:0]: Debug control

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

address	register name	default value	R/W	description
0x3005	SC_REG5	0x00	RW	Bit[7:4]: Debug control Bit[3]: Strobe output enable 0: Disable 1: Enable Bit[2]: PWM output enable 0: Disable 1: Enable Bit[1]: VSYNC output enable 0: Disable 1: Enable Bit[0]: SIOD output enable 0: Disable 1: Enable
0x3006~ 0x3007	RSVD	–	–	Reserved
0x3008	SC_REG8	0x00	RW	Bit[7:0]: Debug control
0x3009	SC_REG9	0x00	RW	Bit[7:4]: Debug control Bit[3]: Strobe output value, valid only when 0x3027[3] is 1 Bit[2]: PWM output value, valid only when 0x3027[2] is 1 Bit[1]: VSYNC output value, valid only when 0x3027[1] is 1 Bit[0]: SIOD output value, valid only when 0x3027[0] is 1
0x300A	SC_CHIP_ID_HIGH	0x77	R	Chip ID High Byte
0x300B	SC_CHIP_ID_LOW	0x50	R	Chip ID Low Byte
0x300C	SC_REG0C	–	R	Revision ID
0x300D~ 0x300E	RSVD	–	–	Reserved
0x300F	SC_REG0F	0xF0	RW	Bit[7]: Debug control Bit[6]: daclk_en Bit[5:4]: Debug control Bit[3]: DACLK same as SCLK 0: DACLK 1: SCLK Bit[2:0]: Debug control

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

address	register name	default value	R/W	description
0x3010	SC_REG10	0xE1	RW	Bit[7]: scale_div_man_en Bit[6]: daclk_en Bit[5:4]: daclk_o divider 00: /1 01: /2 10: /4 11: /1 Bit[3]: DACLK same as SCLK Bit[2:0]: pll_scale_div
0x3011	RSVD	–	–	Reserved
0x3012	SC_MIPI_PHY	0x00	RW	Bit[7:2]: Debug control Bit[1:0]: mipi_ictl[1:0]
0x3013	SC_MIPI_PHY	0x10	RW	Bit[7:6]: Common mode voltage control for MIPI high speed transmitter Bit[5:4]: Driving strength control of MIPI low power transmitter Bit[3]: bp_d_hs_en_lat Bit[2]: bp_c_hs_en_lat Bit[1]: mipi_pad Bit[0]: Debug control
0x3014	SC_MIPI_SC_CTRL0	0x19	RW	Bit[7:6]: mipi_ck_skew_o Bit[5]: Debug control Bit[4]: r_phy_pd_mipi 0: Not used 1: Power down PHY HS TX Bit[3]: Debug control Bit[2]: MIPI enable 0: Disable 1: Enable Bit[1]: mipi_susp_reg MIPI system suspend register 0: Not used 1: Suspend Bit[0]: Data lane disable option 0: Use mipi_release1/2 and lane_disable1/2 to disable two data lanes 1: Use lane_disable1/2 to disable two data lanes
0x3015	SC_MIPI_SC_CTRL1	0x10	RW	Bit[7:0]: MIPI ULPS resume mark1 detect length

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

address	register name	default value	R/W	description
0x3016	SC_CLKRST0	0xF0	RW	Bit[7]: sclk_fc Bit[6]: Debug control Bit[5]: sclk_aec Bit[4]: sclk_tc Bit[3]: rst_fc Bit[2]: Debug control Bit[1]: rst_aec Bit[0]: rst_tc
0x3017	SC_CLKRST1	0xF0	RW	Bit[7]: sclk_blc Bit[6]: sclk_isp Bit[5]: Debug control Bit[4]: sclk_vfifo Bit[3]: rst_blc Bit[2]: rst_isp Bit[1]: Debug control Bit[0]: rst_vfifo
0x3018	SC_CLKRST2	0xF0	RW	Bit[7]: Debug control Bit[6]: sclk_mipi Bit[5]: sclk_ac Bit[4]: sclk_otp Bit[3]: Debug control Bit[2]: rst_mipi Bit[1]: rst_ac Bit[0]: rst_otp
0x3019	SC_CLKRST3	0xF0	RW	Bit[7:0]: Debug control
0x301A	SC_CLKRST4	0xF0	RW	Bit[7:6]: Debug control Bit[5]: pclk_vfifo Bit[4]: pclk_mipi Bit[3]: Debug control Bit[2]: rst_mipi_sc Bit[1:0]: Debug control
0x301B	SC_CLKRST5	0xF0	RW	Bit[7]: sclk_src Bit[6]: Debug control Bit[5]: sclk_asram_tst Bit[4]: sclk_snr_sync Bit[3]: rst_src Bit[2]: Debug control Bit[1]: rst_asram_tst Bit[0]: rst_snr_sync

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

address	register name	default value	R/W	description
0x301C	SC_CLKRST6	0xF2	RW	Bit[7]: sclk_bist Bit[6]: sclk_srb Bit[5]: sclk_grp Bit[4]: Debug control Bit[3]: rst_bist Bit[2]: Debug control Bit[1]: rst_grp Bit[0]: Debug control
0x301D	SC_FREX_RST_MASK0	0x00	RW	Bit[7:0]: Debug control
0x301E	SC_CLOCK_SEL	0x0B	RW	Bit[7:4]: Debug control Bit[3]: pclk_source_sel 0: /1 1: /2 Bit[2:1]: Debug control Bit[0]: sclk2x_source_sel 0: /1 1: /2
0x301F	SC_MISC_CTRL	0x03	RW	Bit[7:6]: mipi_data_skew_o Bit[5]: mipi_clk_lane_ctrl 0: Clock lane hold lp00 when pd_mipi 1: Clock lane is high-z when pd_mipi 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[2:1]: Debug control Bit[0]: cen_global_o
0x3023	SC_LOW_PWR_CTRL	0x07	RW	Bit[7:5]: Debug control Bit[4]: stb_rst_dis 0: Reset all blocks at software standby mode 1: TC, sensor_control, ISP are reset, others are not Bit[3:2]: Debug control Bit[1]: phy_pd_mipi_slppd_dis Bit[0]: phy_pd_lprx_slppd_dis
0x3024~0x3025	RSVD	–	–	Reserved

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

address	register name	default value	R/W	description
0x3026	SC_REG26	0x00	RW	Bit[7:2]: Not used Bit[1:0]: Debug control
0x3027	SC_REG27	0x00	RW	Bit[7]: Debug control Bit[6:5]: io_gpio_sel Bit[3]: io_strobe_sel Bit[2]: io_pwm_sel Bit[1]: io_fsin_sel Bit[0]: io_sda_sel
0x3028	SC_GP_IO_IN0	–	R	Bit[7:0]: Debug control
0x3029	SC_GP_IO_IN1	–	R	Bit[7:4]: Revision ID 0x40: Rev 1A and rev 1B 0x50: Rev 1C and rev 1D 0x60: Rev 1E 0x70: Rev 1F Bit[3:2]: Debug control Bit[1:0]: GPIO[1:0]
0x302A	SC_GP_IO_IN2	–	R	Bit[7:6]: Not used Bit[5]: Strobe input value Bit[3]: FSIN input value Bit[2]: PWM input value Bit[1:0]: Debug control
0x302B	SC_SCCB_ID2	0xE0	RW	SCCB ID 2
0x302C	SC_AUTO_SLEEP_PERIOD	0x01	RW	Bit[7:0]: Auto sleep period[31:24]
0x302D	SC_AUTO_SLEEP_PERIOD	0x00	RW	Bit[7:0]: Auto sleep period[23:16]
0x302E	SC_AUTO_SLEEP_PERIOD	0x00	RW	Bit[7:0]: Auto sleep period[15:8]
0x302F	SC_AUTO_SLEEP_PERIOD	0x00	RW	Bit[7:0]: Auto sleep period[7:0]
0x3030	SC_LP_CTRL0	0x03	RW	Bit[7]: auto_sleep_en Bit[6]: gpio_sel 0: Not used 1: Sleep can be read by GPIO (Y9) Bit[5:4]: Debug control Bit[3:0]: frame_on_num
0x3031	RSVD	–	–	Reserved
0x3032	SC_IO_OEN_SLEEP	0x03	RW	Output Control in Sleep Mode

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

address	register name	default value	R/W	description
0x3033	SC_IO_OEN_SLEEP	0xFF	RW	Bit[7]: Debug control Bit[6:5]: io_gpio_oen_sleep[1:0] Bit[3]: io_strobe_oen_sleep Bit[2]: io_pwm_oen_sleep Bit[1]: io_fsin_oen_sleep Bit[0]: io_sda_oen_sleep
0x3034	RSVD	–	–	Reserved
0x3035	SC_IO_Y_OEN_ PWDN	0x03	RW	Bit[7:2]: Not used Bit[1:0]: Debug control
0x3036	SC_IO_Y_OEN_ PWDN	0xFF	RW	Bit[7:2]: Not used Bit[1:0]: Debug control
0x3037	SC_R37	0xF0	RW	Bit[7]: Debug control Bit[6]: sclk_strobe Bit[5]: sclk_fmt Bit[4]: sclk_pwm Bit[3]: Debug control Bit[2]: rst_strobe Bit[1]: rst_fmt Bit[0]: rst_pwm
0x3038	SC_REG38	0x50	RW	Bit[7]: Debug control Bit[6]: grp_clk_rst sleep en Bit[5]: clk_grp_sel 0: PADCLK 1: SCLK Bit[4]: pwr_switch_pad_clk_en Bit[3]: clk_src_11x_02x 0: 2x 1: 1x Bit[2:0]: Debug control
0x3039~ 0x303A	RSVD	–	–	Reserved
0x303B	SC_REG3B	0x00	RW	Bit[7:2]: Debug control Bit[1]: sccb_pgm_id_en Bit[0]: sccb_id2_nack_en

7.2 PLL control [0x3080 - 0x3083, 0x3098 - 0x309F, 0x30B0 - 0x30B6]

table 7-2 PLL control registers

address	register name	default value	R/W	description
0x3080	PLL_PLL0	0x01	RW	Bit[7:1]: Not used Bit[0]: pll1_vt_2lane_clk_div_o
0x3081	PLL_PLL1	0x00	RW	Bit[7:2]: Not used Bit[1]: pll1_mapping_dis Bit[0]: PLL1_byp
0x3082	PLL_PLL2	0x01	RW	Bit[7:3]: Not used Bit[2:0]: PLL1_cp
0x3083	DEBUG CTRL	–	–	Debug Control
0x3098	PLL_PLL18	0x04	RW	Bit[7:5]: Not used Bit[4:0]: PLL2_pre_divider
0x3099	PLL_PLL19	0x28	RW	Bit[7:0]: PLL2_multiplier
0x309A	PLL_PLL1A	0x05	RW	Bit[7:4]: Not used Bit[3:0]: PLL2_sys_divider
0x309B	PLL_PLL1B	0x04	RW	Bit[7:4]: Not used Bit[3:0]: PLL2_ADC_divider
0x309C	DEBUG CTRL	–	–	Debug Control
0x309D	PLL_PLL1D	0x00	RW	Bit[7:3]: Not used Bit[2]: PLL2_divider Bit[1]: pll_dac_mapping_dis Bit[0]: PLL2_byp
0x309E	PLL_PLL1E	0x01	RW	Bit[7:3]: Not used Bit[2:0]: PLL2_cp
0x309F	DEBUG CTRL	–	–	Debug Control
0x30B0	PLL_VT_PIX_CLK_DIV	0x0A	RW	Bit[7:4]: Not used Bit[3:0]: PLL1_pix_divider
0x30B1	PLL_VT_SYS_CLK_DIV	0x01	RW	Bit[7:5]: Not used Bit[4:0]: PLL1_divider
0x30B2	DEBUG CTRL	–	–	Debug Control
0x30B3	PLL_MULTIPLIER	0x32	RW	Bit[7:0]: PLL1_multiplier
0x30B4	PLL_PLL1_PRE_PLL_DIV	0x02	RW	Bit[7:3]: Not used Bit[2:0]: PLL1_pre_divider
0x30B5	PLL_PLL1_OP_PIX_CLK_DIV	0x01	RW	Bit[7:3]: Not used Bit[2:0]: PLL1_MIPI_divider
0x30B6	DEBUG CTRL	–	–	Debug Control

7.3 SCCB and group hold control [0x3100 - 0x3106, 0x31FF - 0x320F]

table 7-3 SCCB and group hold registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3100~ 0x3105	DEBUG CTRL	–	–	Debug Control
0x3106	SB_SRB_CTRL	0x12	RW	Bit[7]: Enable XVCLK (must be 1 for video streaming) Bit[6:0]: Debug control
0x31FF	SB_SWITCH	0x01	RW	Bit[7:1]: Debug control Bit[0]: SCCB slave select 0: Select SCCB slave which requires EXTCLK 1: Select SCCB slave which does not require EXTCLK
0x3200	GROUP ADR0	0x00	RW	Start Address of Group 0 Buffer Actual Start address is {0x3200[3:0], 4'h0}
0x3201	GROUP ADR1	0x04	RW	Start Address of Group 1 Buffer Actual Start Address is {0x3201[3:0], 4'h0}
0x3202~ 0x3203	RSVD	–	–	Reserved
0x3204	GROUP LEN0	–	R	Length of Group0
0x3205	GROUP LEN1	–	R	Length of Group1
0x3206~ 0x3207	RSVD	–	–	Reserved
0x3208	GROUP ACCESS	–	W	Bit[7:4]: group_ctrl 0000: Group hold start 0001: Group hold end 0110: Group launch at line blanking 1010: Group launch at vertical blanking 1110: Group launch immediately Others: Debug control Bit[3:0]: Group ID 0000: Group bank 0 0001: Group bank 1 Others: Debug control
0x3209	GROUP0 PERIOD	0x00	RW	Number of Frames to Stay in Group0
0x320A	GROUP1 PERIOD	0x00	RW	Number of Frames to Stay in Group1

table 7-3 SCCB and group hold registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x320B	GRP_SW_CTRL	0x01	RW	Bit[7:6]: Debug control Bit[5]: grp0_start_opt Bit[4]: frame_cnt_trig Bit[3]: group_switch_repeat Bit[2]: Group switch enable Bit[1:0]: Second group select
0x320C	SRAM TEST	0x0F	RW	Bit[7:5]: Debug control Bit[4]: Group hold SRAM test enable Bit[3:0]: Group hold SRAM RM[3:0]
0x320D	GRP_ACT	–	R	Active Group Indicator
0x320E	FM_CNT_GRP0	–	R	Group0 Frame Count
0x320F	FM_CNT_GRP1	–	R	Group 1 Frame Count

7.4 manual AWB_gain control [0x3400 - 0x3406]

table 7-4 manual AWB_gain registers

address	register name	default value	R/W	description
0x3400	AWB RED GAIN	0x04	RW	Bit[7:4]: Debug control Bit[3:0]: AWB red gain[11:8]
0x3401	AWB RED GAIN	0x00	RW	Bit[7:0]: AWB red gain[7:0]
0x3402	AWB GRN GAIN	0x04	RW	Bit[7:4]: Debug control Bit[3:0]: AWB green gain[11:8]
0x3403	AWB GRN GAIN	0x00	RW	Bit[7:0]: AWB green gain[7:0]
0x3404	AWB BLU GAIN	0x04	RW	Bit[7:4]: Debug control Bit[3:0]: AWB blue gain[11:8]
0x3405	AWB BLU GAIN	0x00	RW	Bit[7:0]: AWB blue gain[7:0]
0x3406	AWB MAN CTRL	0x01	RW	Bit[7:1]: Debug control Bit[0]: AWB manual control

7.5 manual AEC/AGC [0x3500 - 0x350B, 0x5D00 - 0x5D01, 0x5F00 - 0x5F05]

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

address	register name	default value	R/W	description
0x3500	AEC EXPO	0x00	RW	Exposure Bit[7:4]: Not used Bit[3:0]: Exposure[15:12]
0x3501	AEC EXPO	0x00	RW	Exposure Bit[7:0]: Exposure[11:4] Minimum exposure time is 1 row period. Maximum exposure time is frame length - 20 row periods, where frame length is set by registers {0x380E, 0x380F}.
0x3502	AEC EXPO	0x00	RW	Exposure Bit[7:4]: Exposure[3:0] Minimum exposure time is 1 row period. Maximum exposure time is frame length - 20 row periods, where frame length is set by registers {0x380E, 0x380F}. Bit[3:0]: Debug control
0x3503	AEC MANUAL	0x00	RW	AEC Manual Mode Control Bit[7:6]: Debug control Bit[5]: Gain delay option, select using 0x3503[4] 0: Latched at beginning of frame (n+1) 1: Latched at beginning of next frame (n) Bit[4]: Choose delay option 0: Select using 0x3503[5] as gain delay option 1: Choose exp_no_chg as gain delay option Bit[3:2]: Debug control Bit[1]: AGC manual enable No auto module in this chip so this bit should be always be kept 1 0: Auto enable 1: Manual enable Bit[0]: AEC manual enable No auto module in this chip so this bit should be always be kept 1 0: Auto enable 1: Manual enable
0x3504	MAN SNR GAIN	0x00	RW	Manual Sensor Gain Bit[7:2]: Debug control Bit[1:0]: Manual sensor gain[9:8]
0x3505	MAN SNR GAIN	0x00	RW	Manual Sensor Gain Bit[7:0]: Manual sensor gain[7:0]

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

address	register name	default value	R/W	description
0x3506~ 0x3508	DEBUG CTRL	–	–	Debug Control
0x3509	AEC GAIN CONVERT	0x10	RW	AEC Manual Mode Control Bit[7:5]: Debug control Bit[4]: Gain convert enable 0: Use sensor gain, {0x350A, 0x350B}, as sensor gain 1: Use real gain, {0x350A, 0x350B}, as linear gain Bit[3]: Sensor gain manual enable (BLC cannot be triggered by these gain registers) 0: Disable 1: Manual control for {0x3504, 0x3505} Bit[2]: Debug control Bit[1]: Gain change delay option 0: gain_change delay 1 frame 1: gain_change no delay Bit[0]: Debug control
0x350A	AEC AGC ADJ	0x00	RW	Gain Output to Sensor Bit[7:2]: Not used Bit[1:0]: Gain[9:8]
0x350B	AEC AGC ADJ	0x10	RW	Gain Output to Sensor Bit[7:0]: Gain[7:0] Gain = register 0x350B / 0x10 for linear gain, or Gain = (register 0x350B[7]+1)×(register 0x350B[6]+1)×(register 0x350B[5]+1) ×(register 0x350B[4]+1)×(register 0x350B[3:0]/16+1) for sensor gain. Rev 1A and Rev 1B must choose sensor gain by setting register 0x3509[4] to 0, and Rev 1C and Rev 1D must choose linear gain by setting register 0x3509[4] to 1.
0x5D00	GAIN FORMAT 00	0x07	RW	Bit[7:4]: Debug control Bit[3:0]: Analog gain bit control
0x5D01	GAIN FORMAT 01	0x00	RW	Bit[7:4]: Debug control Bit[3:0]: Digital gain bit control

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

address	register name	default value	R/W	description
0x5F00	DIG_COMP CTRL 00	0x18	RW	Bit[7:5]: Debug control Bit[4]: dither_en 0: Disable 1: Enable Bit[3]: dig_comp BLC on 0: Disable 1: Enable Bit[2]: dig_comp bypass 0: Disable 1: Enable Bit[1]: Manual option, controlled by 0x5F00[0] 0: Manual value fixed at D512 1: Manual value set by {0x5F02, 0x5F03} Bit[0]: Manual gain compensation enable 0: Disable 1: Enable
0x5F01	RSVD	–	–	Reserved
0x5F02	DIG_COMP CTRL 02	0x02	RW	Bit[7:2]: Debug control Bit[1:0]: dig_comp_man[9:8]
0x5F03	DIG_COMP CTRL 03	0x00	RW	Bit[7:0]: dig_comp_man[7:0]
0x5F04	DIG_COMP CTRL 04	–	R	Bit[7:2]: Not used Bit[1:0]: dig_comp_auto_i[9:8]
0x5F05	DIG_COMP CTRL 05	–	R	Bit[7:0]: dig_comp_auto_i[7:0]

7.6 analog control [0x3600 - 0x3684]

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

address	register name	default value	R/W	description
0x3600~ 0x3635	ANALOG REGISTERS	–	–	Analog Control Registers
0x3636	ANA_CTRL_36	0x00	RW	Bit[7:4]: Analog control Bit[3]: Internal regulator disable 0: Enable internal regulator 1: Disable internal regulator Bit[2:0]: Analog control

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

address	register name	default value	R/W	description
0x3637~ 0x3661	ANALOG REGISTERS	–	–	Analog Control Registers
0x3662	ANA_CORE_2	0x01	RW	Bit[7:2]: Debug control Bit[1]: RAW8 enable 0: RAW10 1: RAW8 Bit[0]: Debug control
0x3663~ 0x3665	ANALOG REGISTERS	–	–	Analog Control Registers
0x3666	ANA_CORE_6	0x0A	RW	FSIN/VSYNC Input and Output select Bit[7:4]: Output select 0x0: VSYNC Others: For debug only Bit[3:0]: Internal frame sync input select 0x0: From FSIN pin, used for both frame sync and frame trigger function 0xA: Fixed value 0 Others: For debug purposes
0x3667~ 0x3684	ANALOG REGISTERS	–	–	Analog Control Registers

7.7 sensor control [0x3700 - 0x37AF]

table 7-7 sensor control registers

address	register name	default value	R/W	description
0x3700~ 0x37A7	SENSOR CONTROL REGISTERS	–	–	Sensor Control Registers
0x37A8	FIFO_CTRL0_H	0x00	RW	Internal FIFO Control
0x37A9	FIFO_CTRL0_L	0x60	RW	Internal FIFO Control
0x37AA~ 0x37AF	SENSOR CONTROL REGISTERS	–	–	Sensor Control Registers

7.8 timing control [0x3800 - 0x3835, 0x3837, 0x3880 - 0x38B5]

table 7-8 timing control registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x3800	TIMING_X_ADDR_START	0x00	RW	Array Horizontal Start Point High Byte
0x3801	TIMING_X_ADDR_START	0x04	RW	Array Horizontal Start Point Low Byte
0x3802	TIMING_Y_ADDR_START	0x00	RW	Array Vertical Start Point High Byte
0x3803	TIMING_Y_ADDR_START	0x04	RW	Array Vertical Start Point Low Byte
0x3804	TIMING_X_ADDR_END	0x02	RW	Array Horizontal End Point High Byte
0x3805	TIMING_X_ADDR_END	0x8B	RW	Array Horizontal End Point Low Byte
0x3806	TIMING_Y_ADDR_END	0x01	RW	Array Vertical End Point High Byte
0x3807	TIMING_Y_ADDR_END	0xEB	RW	Array Vertical End Point Low Byte
0x3808	TIMING_X_OUTPUT_SIZE	0x02	RW	ISP Horizontal Output Width High Byte
0x3809	TIMING_X_OUTPUT_SIZE	0x80	RW	ISP Horizontal Output Width Low Byte
0x380A	TIMING_Y_OUTPUT_SIZE	0x01	RW	ISP Vertical Output Height High Byte
0x380B	TIMING_Y_OUTPUT_SIZE	0xE0	RW	ISP Vertical Output Height Low Byte
0x380C	TIMING_HTS	0x03	RW	Total Horizontal Timing Size High Byte
0x380D	TIMING_HTS	0x04	RW	Total Horizontal Timing Size Low Byte
0x380E	TIMING_VTS	0x02	RW	Total Vertical Timing Size High Byte
0x380F	TIMING_VTS	0x04	RW	Total Vertical Timing Size Low Byte
0x3810	TIMING_ISP_X_WIN	0x00	RW	ISP Horizontal Windowing Offset High Byte
0x3811	TIMING_ISP_X_WIN	0x00	RW	ISP Horizontal Windowing Offset Low Byte
0x3812	TIMING_ISP_Y_WIN	0x00	RW	ISP Vertical Windowing Offset High Byte
0x3813	TIMING_ISP_Y_WIN	0x00	RW	ISP Vertical Windowing Offset Low Byte
0x3814	TIMING_X_INC	0x11	RW	Bit[7:4]: x_odd_inc Bit[3:0]: x_even_inc
0x3815	TIMING_Y_INC	0x11	RW	Bit[7:4]: y_odd_inc Bit[3:0]: y_even_inc
0x3816~ 0x381F	DEBUG CTRL	–	–	Debug Control

table 7-8 timing control registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x3820	TIMING_FORMAT1	0x00	RW	Bit[7]: vsub48_blc_dis Bit[6]: vflip_blc Bit[5:3]: Debug control Bit[2]: Vflip Vertical image flip Bit[1]: Vbinf Vertical binning Bit[0]: Debug control
0x3821	TIMING_FORMAT2	0x00	RW	Bit[7:4]: Debug control Bit[3]: Fman Bit[2]: Mirr Horizontal image mirror Bit[1]: Debug control Bit[0]: Hbin Horizontal binning
0x3822	TIMING_REG22	0x44	RW	Bit[7:5]: addr0_num[3:1] Bit[4:0]: ablc_num[5:1]
0x3823	TIMING_REG23	0x00	RW	Bit[7]: fmt_chg_min_dly (write only) Bit[6]: ext_vs_re Bit[5]: ext_vs_en Bit[4]: r_init_man Bit[3]: vts_no_latch Bit[2:0]: ablc_adj
0x3824	TIMING_CS_RST_FSIN	0x00	RW	CS Reset Value at vs_ext High Byte
0x3825	TIMING_CS_RST_FSIN	0x00	RW	CS Reset Value at vs_ext Low Byte
0x3826	TIMING_RST_FSIN	0x00	RW	R Reset Value at vs_ext High Byte
0x3827	TIMING_RST_FSIN	0x00	RW	R Reset Value at vs_ext Low Byte
0x3828	TIMING_FVTS	0x00	RW	Fractional Vertical Timing Size High Byte
0x3829	TIMING_FVTS	0x00	RW	Fractional Vertical Timing Size Low Byte
0x382A	TIMING_REG2A	0x00	RW	Bit[7:4]: Debug control Bit[3]: vts_auto_en Bit[2]: Debug control Bit[1:0]: href_w
0x382B	TIMING_REG2B	0xFA	RW	Bit[7:4]: grp_wr_start Bit[3:0]: tc_r_int_adj
0x382C	TIMING_REG2C	0x03	RW	Bit[7:0]: hts_global_tx[15:8]
0x382D	TIMING_REG2D	0x10	RW	Bit[7:0]: hts_global_tx[7:0]

table 7-8 timing control registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x382E	TIMING_REG2E	0x07	RW	Bit[7:3]: Debug control Bit[2]: r_tc_hts_blank 0: Uniform HTS 1: Separate HTS for global transfer Bit[1]: r_blc_lines_sync_tc 0: tc_href 1: tc_href synced to image line start Bit[0]: r_blc_lines_sync 0: tc_href 1: blc_href synced to image line start
0x382F	TIMING_REG2F	0x04	RW	Bit[7]: r_pd_row_st_opt 0: auto_vbk_st 1: manu_vbk_st Bit[6]: r_pd_row_ed_opt 0: auto_vbk_ed 1: manu_vbk_ed Bit[5]: r_pd_ana_to_sys 0: ana_vbk cover sys_vbk 1: ana_vbk same as sys_vbk Bit[4]: r_pd_sys_to_ana 0: sys_vbk same as ana_vbk 1: sys_vbk covered by ana_vbk 1 Bit[3:0]: r_pd_row_ofst[3:0] Auto offset after vbk_st and before vbk_ed
0x3830	TIMING_TC_R	–	R	Bit[7:0]: System row counter[15:8]
0x3831	TIMING_TC_R	–	R	Bit[7:0] System row counter[7:0]
0x3832	TIMING_REG32	0x00	RW	Bit[7:0]: pd_vbk_st[15:8] Start of power saving in vblank Upper byte reference to row count
0x3833	TIMING_REG33	0x05	RW	Bit[7:0]: pd_vbk_st[7:0] Start of power saving in vblank Lower byte reference to row count
0x3834	TIMING_REG34	0x00	RW	Bit[7:0]: pd_vbk_ed[15:8] End of power saving in vblank Upper byte reference to row count

table 7-8 timing control registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x3835	TIMING_REG35	0x05	RW	Bit[7:0]: pd_vbk_ed[7:0] End of power saving in vblank Lower byte reference to row count
0x3837	DIGITAL BINNING CTRL	0x00	RW	Digital Binning Control Bit[7:5]: Not used Bit[4]: Debug control Bit[3:2]: Horizontal binning control Set to 2'b00 when binning is disabled and 2'b11 when binning is enabled Bit[1]: Horizontal binning summation enable 0: Average 1: Summation Bit[0]: Horizontal digital binning enable
0x3880~ 0x38B5	DEBUG CTRL	–	–	Debug Control

7.9 PWM and strobe control [0x3B80 - 0x3B97]

table 7-9 PWM and strobe control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x3B80	LED_PWM_REG00	0x10	RW	Bit[7:6]: Debug control Bit[5]: pwm_frame_trigger_always_on_opt 0: sof_vsync trigger only 1: pwm_frame always on Bit[4]: sof_vsync_trigger_en 0: Not used 1: sof_vsync trigger enable Bit[3]: pwm_frame_trigger_opt 0: Sof as trigger of pwm_frame 1: VSYNC Bit[2]: strobe_frame_trigger_opt 0: VSYNC as trigger of strobe_frame 1: SOF Bit[1]: led_pwm_frame_pol Bit[0]: led_pwm_free_pol

table 7-9 PWM and strobe control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x3B81	LED_PWM_REG01	0xA5	RW	Bit[7:0]: strobe_frame pattern for sequential 8 frames 0: Off for 8 sequential frames 1: On
0x3B82	LED_PWM_REG02	0x10	RW	Bit[7:0]: pwm_freq_div_cycle_reg[15:8]
0x3B83	LED_PWM_REG03	0x00	RW	Bit[7:0]: pwm_freq_div_cycle_reg[7:0]
0x3B84	LED_PWM_REG04	0x08	RW	Bit[7:0]: pwm_duty_cycle_reg[15:8]
0x3B85	LED_PWM_REG05	0x00	RW	Bit[7:0]: pwm_duty_cycle_reg[7:0]
0x3B86	LED_PWM_REG06	0x01	RW	Bit[7:0]: low_limit[15:8]
0x3B87	LED_PWM_REG07	0x00	RW	Bit[7:0]: low_limit[7:0]
0x3B88	LED_PWM_REG08	0x00	RW	Bit[7]: strobe_frame_sign_bit 0: Positive delay 1: Negative delay Bit[6:0]: strobe_frame_shift[31:24]
0x3B89	LED_PWM_REG09	0x00	RW	Bit[7:0]: strobe_frame_shift[23:16]
0x3B8A	LED_PWM_REG0A	0x00	RW	Bit[7:0]: strobe_frame_shift[15:8]
0x3B8B	LED_PWM_REG0B	0x05	RW	Bit[7:0]: strobe_frame_shift[7:0]
0x3B8C	LED_PWM_REG0C	0x00	RW	Bit[7:0]: strobe_frame_span[31:24]
0x3B8D	LED_PWM_REG0D	0x00	RW	Bit[7:0]: strobe_frame_span[23:16]
0x3B8E	LED_PWM_REG0E	0x00	RW	Bit[7:0]: strobe_frame_span[15:8]
0x3B8F	LED_PWM_REG0F	0x1A	RW	Bit[7:0]: strobe_frame_span[7:0]
0x3B90	LED_PWM_REG10	0x01	RW	Bit[7:0]: r_strobe_row_st[15:8]
0x3B91	LED_PWM_REG11	0xB4	RW	Bit[7:0]: r_strobe_row_st[7:0]
0x3B92	LED_PWM_REG12	0x00	RW	Bit[7:0]: r_strobe_cs_st[15:8]
0x3B93	LED_PWM_REG13	0x10	RW	Bit[7:0]: r_strobe_cs_st[7:0]
0x3B94	LED_PWM_REG14	0x05	RW	Bit[7:0]: step_onerow_man[15:8]
0x3B95	LED_PWM_REG15	0xF2	RW	Bit[7:0]: step_onerow_man[7:0]
0x3B96	LED_PWM_REG16	0x40	RW	Bit[7]: r_strobe_frm_pwen Bit[6]: r_strobe_frm_pwst Bit[5]: r_strobe_pol Bit[4]: r_strobe_step_pix Bit[3]: r_step_onerow_precision_man Bit[2:0]: r_strobe_st_opt
0x3B97	LED_PWM_REG17	0x00	RW	Bit[7:0]: Debug control

7.10 low power mode control [0x3C00 - 0x3C0F, 0x4A47 - 0x4A49, 0x4E00 - 0x4E30]

table 7-10 low power mode control registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x3C00	LOWPWR00	0x89	RW	Bit[7]: Manual frame timing control enable during low power mode (must set to 1 in low power mode) 0: Frame timing controlled by registers 0x380C~0x380F 1: Frame timing controlled by registers 0x3C0C~0x3C0F Bit[6]: Debug control Bit[5:4]: ON period timing control option 00: Normal EOF 01: Low power block self generated EOF 1x: Other debug options Bit[3]: Frame timing control option in low power mode (must set to 1) 0: Based on input clock cycle 1: Based on row and frame period defined by registers 0x3C0C~0x3C0F Bit[2]: Debug control Bit[1:0]: OFF period control option (must be set to 2'b01) 00: Normal EOF 01: Low power block self generated EOF 1x: Other debug options

table 7-10 low power mode control registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x3C01	LOWPWR01	0xAB	RW	Bit[7}: Group write function enable during off period in low power mode when register bit 0x3C01[6] = 1 Bit[6}: Turn OFF/ON sensor during off period in low power mode to save power, override register bits 0x3C01[5:4] 0: Turn on 1: Turn off Bit[5}: Turn OFF/ON analog circuit during off period in low power mode to save power 0: Turn on 1: Turn off Bit[4}: Turn OFF/ON system control circuit during off period in low power mode 0: Turn on 1: Turn off Bit[3}: Auto power ON/OFF control (must set to auto in low power mode) 0: Auto 1: Manual Bit[2}: Power control in manual mode (register bit 0x3C01[3] = 1) (must be set to 0 in normal streaming mode) Bit[1:0}: Timing control option for ON and OFF period (must be set to 2'b11) 0x63: Low power mode 0xAB: Normal mode
0x3C02	LOWPWR02	0x01	RW	Bit[7:2]: Debug control Bit[1}: Idle phase enable Bit[0}: Streaming phase enable x0: Not allowed 01: Normal streaming mode 11: Enable low power streaming mode

table 7-10 low power mode control registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x3C03	LOWPWR03	0x00	RW	Low Power Mode Control Bit[7]: Trigger for internal trigger snapshot mode A rising edge on 0x3C03[7] wakes sensor up and streams out {0x3404, 0x3405} frames Bit[6:0]: Mode control 0x00: Low frame rate streaming mode (i.e., repeating sequence of streaming {0x3404, 0x3405} frames and then sleeping {0x3C06, 0x3C07} frame) 0x17: External trigger snapshot mode A rising edge on FSIN pin wakes sensor up and streams out {0x3404, 0x3405} frames 0x03: Internal trigger snapshot mode A rising edge on 0x3C03[7] wakes sensor up and streams out {0x3404, 0x3405} frames Others: For debug only
0x3C04	LOWPWR04	0x00	RW	Bit[7:0]: Number of active frames[15:8]
0x3C05	LOWPWR05	0x03	RW	Bit[7:0]: Number of active frames[7:0]
0x3C06	LOWPWR06	0x00	RW	Bit[7:0]: Number of idle frames[15:8]
0x3C07	LOWPWR07	0x05	RW	Bit[7:0]: Number of idle frames[7:0]
0x3C08~ 0x3C0B	NOT USED	–	–	Not Used
0x3C0C	LOWPWR0C	0x00	RW	Bit[7:0]: Row period[15:8] in units of input clock period
0x3C0D	LOWPWR0D	0x00	RW	Bit[7:0]: Row period[7:0] in units of input clock period
0x3C0E	LOWPWR0E	0x00	RW	Bit[7:0]: Number of rows per base frame[15:8] Usually set to same value as {0x380E, 0x380F}
0x3C0F	LOWPWR0F	0x00	RW	Bit[7:0]: Number of rows per base frame[7:0] Usually set to same value as {0x380E, 0x380F}
0x4A47~ 0x4A49	LOWPWR CTRL REGISTERS	–	RW	Low Power Control Registers
0x4E00~ 0x4E30	DEBUG CTRL	–	–	Debug Control

7.11 OTP control [0x3D80 - 0x3D87, 0x3D8B]

table 7-11 OTP control registers

address	register name	default value	R/W	description
0x3D80	OTP PROGRAM CTRL	0x00	RW	Bit[7]: otp_pgenb_o 0: Not used 1: Program on going Bit[6:1]: Debug control Bit[0]: otp_pgm To start program Write bit 0 to 1
0x3D81	OTP LOAD CTRL	–	R	Bit[7]: pt_load_o 0: Not used 1: Load on going Bit[6:1]: Debug control Bit[0]: otp_rd Writing to this register will start loading data
0x3D82	OTP PROGRAM PULSE	0x65	RW	Bit[7:0]: Control program strobe pulse, by 8 × Tsclk
0x3D83	OTP LOAD PULSE	0x05	RW	Bit[7:4]: Not used Bit[3:0]: Control load strobe pulse, by Tsclk
0x3D84	OPT MODE CTRL	0x00	RW	Bit[7]: program_dis 0: Enable 1: Disable Bit[6]: mode_select 0: Auto mode 1: Manual mode Bit[5:0]: Memory select
0x3D85	OTP START ADDR	0x06	RW	Bit[7:4]: Not used Bit[3:0]: Start address for manual mode
0x3D86	OTP END ADDR	0x06	RW	Bit[7:4]: Not used Bit[3:0]: End address for manual mode
0x3D87	OTP PS2CS	0x00	RW	Bit[7:4]: Not used Bit[3:0]: PS to CSB time control by sclk
0x3D8B	DEBUG CTRL	–	–	Debug Control

7.12 BLC control [0x4000 - 0x4051]

table 7-12 BLC control registers (sheet 1 of 4)

address	register name	default value	R/W	description
0x4000	BLC CTRL 00	0x30	RW	Bit[7:6]: BLC unstable option control Bit[5]: BLC median filter enable 0: Disable 1: Enable Bit[4:0]: Slope average weight
0x4001	BLC CTRL 01	0xC2	RW	Bit[7]: Slope apply enable 0: Disable 1: Enable Bit[6]: Dither enable 0: Disable 1: Enable Bit[5:0]: BLC start line number
0x4002	BLC AUTO	0xC5	RW	Bit[7]: Format change enable 0: BLC keep same after format change 1: BLC will redo after format change Bit[6]: BLC auto enable 0: Get black level from manual register 1: Calculate black level from auto statistics Bit[5:0]: Reset frame number Frames BLC continue after reset
0x4003	BLC FREEZE	0x01	RW	Bit[7]: Manual redo enable Bit[6]: Freeze enable 0: Normal 1: BLC black level will not update; Priority lower than always update Bit[5:0]: Manual frame number BLC redo frame number
0x4004	BLC NUM	0x04	RW	Bit[7:6]: Debug control Bit[5:0]: Number of black lines used

table 7-12 BLC control registers (sheet 2 of 4)

address	register name	default value	R/W	description
0x4005	BLC MAN CTRL	0x00	RW	Bit[7:6]: Debug control Bit[5]: One line slope mode Bit[4]: Median filter option 0: Image will not pass median filter 1: Image will pass median filter Bit[3]: blc_man_1_en Apply one channel offset {0x400C, 0x400D} to all manual BLC channels Bit[2]: Release black line enable 0: Disable 1: Enable Bit[1]: blc_always_up_en 0: BLC will continue several frames after reset; after that, it will no longer change until gain changes (controlled by bit[0]), or format changes (controlled by register 0x4002[7]). 1: BLC always updates in every frame Bit[0]: Debug control 0: agc_change generated by BLC pre 1: agc_change from system
0x4006	DEBUG CTRL	–	–	Debug Control
0x4007	BLC WIN	0x20	RW	Bit[7:6]: Debug control Bit[5]: r_gain_change_enable 0: Disable 1: Enable Bit[4:3]: Window selection 00: Full image 01: Windows do not contain first 16 pixels and last 16 pixels 10: Windows do not contain first 1/16 image and last 1/16 image 11: Windows do not contain first 1/8 image and last 1/8 image Bit[2:0]: Bypass mode 000: Bypass data_i after limit bits 001: Bypass data_i[9:0]: 010: Bypass data_i[10:1]: 011: Bypass debug data bbrr 100: Bypass debug data gggg 101~111: Not used
0x4008	BLC STABLE RANGE	0xFF	RW	Bit[7]: BLC stable range option enable Bit[6]: BLC stable range value Bit[5:0]: BLC stable range value
0x4009	BLC TARGET	0x10	RW	Black Level Target

table 7-12 BLC control registers (sheet 3 of 4)

address	register name	default value	R/W	description
0x400A~ 0x400B	DEBUG CTRL	–	–	Debug Control
0x400C	BLC MAN LEVEL0	0x00	RW	Bit[7:4]: Debug control Bit[3:0]: BLC manual level channel 0[11:8]
0x400D	BLC MAN LEVEL0	0x00	RW	Bit[7:0]: BLC manual level channel 0[7:0]
0x400E	BLC MAN LEVEL1	0x00	RW	Bit[7:4]: Debug control Bit[3:0]: BLC manual level channel 1[11:8]
0x400F	BLC MAN LEVEL1	0x00	RW	Bit[7:0]: BLC manual level channel 1[7:0]
0x4010	BLC MAN LEVEL2	0x00	RW	Bit[7:4]: Debug control Bit[3:0]: BLC manual level channel 2[11:8]
0x4011	BLC MAN LEVEL2	0x00	RW	Bit[7:0]: BLC manual level channel 2 [7:0]
0x4012	BLC MAN LEVEL3	0x00	RW	Bit[7:4]: Debug control Bit[3:0]: BLC manual level channel 3[11:8]
0x4013	BLC MAN LEVEL3	0x00	RW	Bit[7:0]: BLC manual level channel 3[7:0]
0x4014~ 0x402B	NOT USED	–	–	Not Used
0x402C	BLC LEVEL 0	–	R	Bit[7:6: Debug control Bit[5:0]: Black level channel 0[13:8]
0x402D	BLC LEVEL 0	–	R	Bit[7:0]: Black level channel 0[7:0]
0x402E	BLC LEVEL 1	–	R	Bit[7:6]: Debug control Bit[5:0]: Black level channel 1[13:8]
0x402F	BLC LEVEL 1	–	R	Bit[7:0]: Black level channel 1[7:0]
0x4030	BLC LEVEL 2	–	R	Bit[7:6]: Debug control Bit[5:0]: Black level channel 2[13:8]
0x4031	BLC LEVEL 2	–	R	Bit[7:0]: Black level channel 2[7:0]
0x4032	BLC LEVEL 3	–	R	Bit[7:6]: Debug control Bit[5:0]: Black level channel 3[13:8]
0x4033	BLC LEVEL 3	–	R	Bit[7:0]: Black level channel 3[7:0]
0x4034	BLC SLOPE LEVEL 0	–	R	Bit[7:6]: Debug control Bit[5:0]: Slope level channel 0[13:8]
0x4035	BLC SLOPE LEVEL 0	–	R	Bit[7:0]: Slope level channel 0[7:0]
0x4036	BLC SLOPE LEVEL 1	–	R	Bit[7:6]: Debug control Bit[5:0]: Slope level channel 1[13:8]
0x4037	BLC SLOPE LEVEL 1	–	R	Bit[7:0]: Slope level channel 1[7:0]

table 7-12 BLC control registers (sheet 4 of 4)

address	register name	default value	R/W	description
0x4038	BLC SLOPE LEVEL 2	–	R	Bit[7:6]: Debug control Bit[5:0]: Slope level channel 2[13:8]
0x4039	BLC SLOPE LEVEL 2	–	R	Bit[7:0]: Slope level channel 2[7:0]
0x403A	BLC SLOPE LEVEL 3	–	R	Bit[7:6]: Debug control Bit[5:0]: Slope level channel 3[13:8]
0x403B	BLC SLOPE LEVEL 3	–	R	Bit[7:0]: Slope level channel 3[7:0]
0x403C~ 0x404D	NOT USED	–	–	Not Used
0x404E	BLC AVG	0xC4	RW	Bit[7]: BLC slope average enable Bit[6]: BLC slope average reset Bit[5:0]: BLC slope average frame numbers
0x404F	BLC CTRL 4F	0x01	RW	Bit[7:3]: Debug control Bit[2]: swap_gain_option Bit[1]: vsize_man_en 0: Disable 1: Enable Bit[0]: Manual v size[8]
0x4050	BLC CTRL 50	0x98	RW	Bit[7:0]: Manual v size[7:0]
0x4051	BLC CTRL 51	0x01	RW	Bit[7:1]: Debug control Bit[0]: r_vsync_latch_en

7.13 frame control [0x4240 - 0x4243]

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

address	register name	default value	R/W	description
0x4240	FC CTRL0	0x00	RW	Bit[7:4]: Not used Bit[3]: sof_sel Bit[2]: fcnt_eof_sel Bit[1]: fcnt_mask_dis Bit[0]: fcnt_reset
0x4241	FRAME ON NUM	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Frame on number
0x4242	FRAME OFF NUM	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Frame off number

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

address	register name	default value	R/W	description
0x4243	FC CTRL3	0x00	RW	Bit[7]: Not used 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

7.14 format control [0x4300 - 0x4307, 0x4310 - 0x4316, 0x4320 - 0x4329]

table 7-14 format control registers (sheet 1 of 2)

address	register name	default value	R/W	description
0x4300	DATA_MAX H	0xFF	RW	Bit[7:0]: Data max[9:2]
0x4301	DATA_MIN H	0x00	RW	Bit[7:0]: Data min[9:2]
0x4302	CLIP L	0x0C	RW	Bit[7:4]: Not used Bit[3:2]: Data max[1:0] Bit[1:0]: Data min[1:0]
0x4303	FORMAT CTRL3	0x00	RW	Bit[7]: r_inc_en Bit[6]: r_inc_pattern Bit[5]: r_pad_lsb Bit[4]: r_bar_mux Bit[3]: r_bar_en Bit[2]: r_bit_shift_tst_en Bit[1]: r_tst_bit8 Bit[0]: r_bit_shift_tst_md
0x4304	FORMAT CTRL4	0x08	RW	Bit[7]: Not used Bit[6:4]: data_bit_swap Bit[3]: tst_full_win Bit[2:0]: bar_pad
0x4305~ 0x4306	RSVD	–	–	Reserved
0x4307	EMBED CTRL	0x30	RW	Bit[7:0]: Debug control
0x4310	DEBUG CTRL	–	–	Debug Control
0x4311	VSYNC_WIDTH_H	0x04	RW	Bit[7:0]: VSYNC width[15:8] (in terms of pixel numbers)

table 7-14 format control registers (sheet 2 of 2)

address	register name	default value	R/W	description
0x4312	VSYNC_WIDTH_L	0x00	RW	Bit[7:0]: VSYNC width[7:0] (in terms of pixel numbers)
0x4313	VSYNC_CTRL	0x00	RW	Bit[7:5]: Not used Bit[4]: VSYNC polarity Bit[3:2]: VSYNC output select Bit[1]: VSYNC mode 3 Bit[0]: VSYNC mode 2
0x4314	VSYNC_DELAY1	0x00	RW	Bit[7:0]: VSYNC trigger to VSYNC delay[23:16]
0x4315	VSYNC_DELAY2	0x01	RW	Bit[7:0]: VSYNC trigger to VSYNC delay[15:8]
0x4316	VSYNC_DELAY3	0x00	RW	Bit[7:0]: VSYNC trigger to VSYNC delay[7:0]
0x4320	TST PATTERN CTRL	0x80	RW	Bit[7:6]: Pixel order 00: P3P4/P1P2 01: P4P3/P2P1 10: P1P2/P3P4 11: P2P1/P4P3 Bit[5]: byte_swap Bit[4:2]: Debug control Bit[1]: Solid test pattern enable 0: Solid test pattern OFF 1: Solid test pattern enable Bit[0]: Debug control
0x4321	RSVD	–	–	Reserved
0x4322	SOLID_P1_H	0x00	RW	P1 Value for Solid Test Pattern MSB
0x4323	SOLID_P1_L	0x00	RW	P1 Value for Solid Test Pattern LSB
0x4324	SOLID_P2_H	0x00	RW	P2 Value for Solid Test Pattern MSB
0x4325	SOLID_P2_L	0x00	RW	P2 Value for Solid Test Pattern LSB
0x4326	SOLID_P4_H	0x00	RW	P4 Value for Solid Test Pattern MSB
0x4327	SOLID_P4_L	0x00	RW	P4 Value for Solid Test Pattern LSB
0x4328	SOLID_P3_H	0x00	RW	P3 Value for Solid Test Pattern MSB
0x4329	SOLID_P3_L	0x00	RW	P3 Value for Solid Test Pattern LSB

7.15 VFIFO control [0x4500 - 0x4502, 0x4600 - 0x4604]

table 7-15 VFIFO control registers

address	register name	default value	R/W	description
0x4500	DEBUG CTRL	–	–	Debug Control
0x4501	SC_REG1501	0x08	RW	Bit[7]: Debug control Bit[6]: Column read out circuit timing control (must be set to 1 in low power mode, it has no effect in normal streaming mode) Bit[5:4]: Debug control Bit[3:2]: Debug control (must be set to 2'bx0) Bit[1:0]: Output sequence control of column read out circuit
0x4502	DEBUG CTRL	–	–	Debug Control
0x4600	READ START H	0x00	RW	VFIFO Read Start Point High Byte
0x4601	READ START L	0x28	RW	VFIFO Read Start Point Low Byte
0x4602	VFIFO CTRL2	0xF0	RW	Bit[7:4]: r_rm Bit[3]: r_test1 Bit[2]: Debug control Bit[1]: Frame reset enable Bit[0]: RAM bypass enable
0x4603	VFIFO CTRL3	0x11	RW	Bit[7:5]: Debug control Bit[4]: man_start_mode Bit[3:2]: Debug control Bit[1:0]: start_offset
0x4604	VFIFO STATUS	–	R	Bit[7:4]: Debug control Bit[3]: ram_full Bit[2]: ram_empty Bit[1]: fo_full Bit[1:0]: fo_empty

7.16 MIPI top [0x4800 - 0x4806, 0x4810 - 0x4849, 0x4850 - 0x4854, 0x4860 - 0x4865]

table 7-16 MIPI top control registers (sheet 1 of 9)

address	register name	default value	R/W	description
0x4800	MIPI CTRL 00	0x44	RW	MIPI Control 00 Bit[7]: Debug control Bit[6]: ck_mark1_en 0: Not used 1: Enable clock lane mark1 when resume Bit[5]: Clock lane gate enable 0: Clock lane is free running 1: Gate clock lane when there is no packet to transmit Bit[4]: Line sync enable 0: Do not send line short packet for each line 1: Send line short packet for each line Bit[3]: Debug control Bit[2]: Idle status 0: MIPI bus will be LP00 when there is no packet to transmit 1: MIPI bus will be LP11 when there is no packet to transmit Bit[1]: Debug control Bit[0]: clk_lane_dis 0: Not used 1: Set clock lane to LP mode manually

table 7-16 MIPI top control registers (sheet 2 of 9)

address	register name	default value	R/W	description
0x4801	MIPI CTRL 01	0x03	RW	MIPI Control 01 Bit[7]: Long packet data type manual enable 0: Use mipi_dt 1: Use dt_man_o as long packet data (see register 0x4814[5:0]) Bit[6]: Short packet data type manual enable 0: Not used 1: Use dt_spkt as short packet data (see register 0x4815[5:0]) Bit[5]: Short packet wc sel 0: Use frame counter or line counter 1: Select spkt_wc_reg_o ({0x4812, 0x4813}) Bit[4]: PH bit order for ECC 0: {DI[7:0], WC[7:0], WC[15:8]} 1: {DI[0:7], WC[0:7], WC[8:15]} Bit[3]: PH byte order for ECC 0: {DI, WC_l, WC_h} 1: {DI, WC_h, WC_l} Bit[2]: PH byte order2 for ECC 0: {DI, WC} 1: {WC, DI} Bit[1]: mark1_en 0: Not used 1: When mipi_sys_susp = 1, lane1 sends mark1 from wkup_dly_o after each reset release Bit[0]: Debug control

table 7-16 MIPI top control registers (sheet 3 of 9)

address	register name	default value	R/W	description
0x4802	MIPI CTRL 02	0x00	RW	MIPI Control 02 Bit[7]: hs_prepare_sel 0: Auto calculate T_hs_prepare, unit: pclk2x 1: Use hs_prepare_min_o[7:0] Bit[6]: clk_prepare_sel 0: Auto calculate T_clk_prepare, unit: pclk2x 1: Use clk_prepare_min_o[7:0] Bit[5]: clk_post_sel 0: Auto calculate T_clk_post, unit: pclk2x 1: Use clk_post_min_o[7:0] Bit[4]: clk_trail_sel 0: Auto calculate T_clk_trail, unit: pclk2x 1: Use clk_trail_min_o[7:0] Bit[3]: hs_exit_sel 0: Auto calculate T_hs_exit, unit: pclk2x 1: Use hs_exit_min_o[7:0] Bit[2]: hs_zero_sel 0: Auto calculate T_hs_zero, unit: pclk2x 1: Use hs_zero_min_o[7:0] Bit[1]: hs_trail_sel 0: Auto calculate T_hs_trail, unit: pclk2x 1: Use hs_trail.min_o[7:0] Bit[0]: clk_zero_sel 0: Auto calculate T_clk_zero, unit: pclk2x 1: Use clk_zero_min_o[7:0]
0x4803	MIPI CTRL 03	0x50	RW	MIPI Control 03 Bit[7:6]: lp_glitch_nu 0: Use 2d of lp_in 1: Mask one SCLK cycle glitch of lp_in Bit[5:4]: cd_glitch_nu 0: Use 2d of lp_cd_in 1: Mask one SCLK cycle glitch of lp_cd_in Bit[3]: cd1_int_en 0: Disable cd plus of data lane1 1: Enable cd plus of data lane1 Bit[2]: Debug control Bit[1]: lp_cd1_en 0: Disable cd of data_lane1 from PHY 1: Enable cd of data_lane1 from PHY Bit[0]: Debug control

table 7-16 MIPI top control registers (sheet 4 of 9)

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

table 7-16 MIPI top control registers (sheet 5 of 9)

address	register name	default value	R/W	description
0x4805	MIPI CTRL 05	0x10	RW	MIPI Control 05 Bit[7]: Debug control Bit[6]: lane_disable1 0: Not used 1: Disable MIPI data lane1, lane1 will be LP00 Bit[5]: lpx_p_sel 0: Auto calculate t_lpx_o in pclk2x 1: Use lp_p_min[7:0] Bit[4]: lp_rx_intr_sel 0: Send lp_rx_intr_o as first byte 1: Send lp_rx_intr_o at end of receiving Bit[3]: cd_tst_sel 0: Not used 1: Select PHY test pins Bit[2]: mipi_reg_mask 0: Not used 1: Disable MIPI access SRB Bit[1]: clip enable Bit[0]: hd_sk_en 0: Disable MIPI and MCU hand shake registers 1: Enable MIPI and MCU hand shake registers
0x4806	MIPI CTRL 06	0x0F	RW	Bit[7]: mipi_test Bit[6]: prbs_en Test mode Bit[5:4]: Debug control Bit[3]: mipi_slp_man_st MIPI bus status manual control enable in sleep mode Bit[2]: clk_lane_state Bit[1]: Debug control Bit[0]: data_lane1_state
0x4810	MIPI MAX FRAME COUNT	0xFF	RW	High Byte of Maximum Frame Count of Frame Sync Short Packet
0x4811	MIPI MAX FRAME COUNT	0xFF	RW	Low Byte of Maximum Frame Count of Frame Sync Short Packet
0x4812	MIPI SHORT PKT COUNTER	0x00	RW	High Byte of Manual Short Packet Word Counter
0x4813	MIPI SHORT PKT COUNTER	0x00	RW	Low Byte of Manual Short Packet Word Counter
0x4814	MIPI CTRL14	0x2A	RW	MIPI Control 14 Bit[7:6]: Virtual channel of MIPI Bit[5:0]: Data type in manual mode

table 7-16 MIPI top control registers (sheet 6 of 9)

address	register name	default value	R/W	description
0x4815	MIPI_DT_SPKT	0x40	RW	Bit[7]: Not used Bit[6]: pclk_inv 0: Use mipi_pclk_o rising edge 1: Use mipi_pclk_o falling edge Bit[5:0]: Not used
0x4816~ 0x4817	RSVD	–	–	Reserved
0x4818	HS_ZERO_MIN	0x00	RW	High Byte of Minimum Value for hs_zero, unit: ns
0x4819	HS_ZERO_MIN	0x9A	RW	Low Byte of Minimum Value for hs_zero, unit: ns hs_zero_real = hs_zero_min_o + Tui*ui_hs_zero_min_o
0x481A	HS_TRAIL_MIN	0x00	RW	High Byte of Minimum Value for hs_trail, unit: ns
0x481B	HS_TRAIL_MIN	0x3C	RW	Low Byte of Minimum Value for hs_trail, unit: ns hs_trail_real = hs_trail_min_o + Tui*ui_hs_trail_min_o
0x481C	CLK_ZERO_MIN	0x01	RW	High Byte of Minimum Value for clk_zero, unit: ns
0x481D	CLK_ZERO_MIN	0x86	RW	Low Byte of Minimum Value for clk_zero, unit: ns clk_zero_real = clk_zero_min_o + Tui*ui_clk_zero_min_o
0x481E	CLK_PREPARE_ MIN	0x00	RW	High Byte of Minimum Value for clk_prepare, unit: ns
0x481F	CLK_PREPARE_ MIN	0x3C	RW	Low Byte of Minimum Value for clk_prepare, unit: ns clk_prepare_real = clk_prepare_min_o + Tui*ui_clk_prepare_min_o
0x4820	CLK_POST_MIN	0x00	RW	High Byte of Minimum Value for clk_post, unit: ns Bit[7:2]: Not used Bit[1:0]: clk_post_min[9:8]
0x4821	CLK_POST_MIN	0x56	RW	Low Byte of Minimum Value for clk_post, unit: ns clk_post_real = clk_post_min_o + Tui*ui_clk_post_min_o
0x4822	CLK_TRAIL_MIN	0x00	RW	High Byte of Minimum Value for clk_trail, unit: ns Bit[7:2]: Not used Bit[1:0]: clk_trail_min[9:8]
0x4823	CLK_TRAIL_MIN	0x3C	RW	Low Byte of Minimum Value for clk_trail, unit: ns clk_trail_real = clk_trail_min_o + Tui*ui_clk_trail_min_o
0x4824	LPX_P_MIN	0x00	RW	High Byte of Minimum Value for lpx_p, unit: ns Bit[7:2]: Not used Bit[1:0]: lpx_p_min[9:8]
0x4825	LPX_P_MIN	0x32	RW	Low Byte of Minimum Value for lpx_p, unit: ns lpx_p_real = lpx_p_min_o + Tui*ui_lpx_p_min_o

table 7-16 MIPI top control registers (sheet 7 of 9)

address	register name	default value	R/W	description
0x4826	HS_PREPARE_ MIN	0x00	RW	High Byte of Minimum Value of hs_prepare, unit: ns
0x4827	HS_PREPARE_ MIN	0x32	RW	Low Byte of Minimum Value for hs_prepare, unit: ns hs_prepare_real = hs_prepare_min_o + Tui*ui_hs_prepare_min_o
0x4828	HS_EXIT_MIN	0x00	RW	High Byte of Minimum Value for hs_exit, unit: ns Bit[7:2]: Not used Bit[1:0]: hs_exit_min[9:8]
0x4829	HS_EXIT_MIN	0x64	RW	Low Byte of Minimum Value for hs_exit, unit: ns hs_exit_real = hs_exit_min_o + Tui*ui_hs_exit_min_o
0x482A	UI_HS_ZERO_ MIN	0x05	RW	Minimum UI Value of hs_zero, unit: UI
0x482B	UI_HS_TRAIL_ MIN	0x04	RW	Minimum UI Value of hs_trail, unit: UI
0x482C	UI_CLK_ZERO_ MIN	0x00	RW	Minimum UI Value of clk_zero, unit: UI
0x482D	UI_CLK_PREPARE _MIN	0x00	RW	Bit[7:6]: Not used Bit[5:4]: 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	Minimum UI Value of clk_post, unit: UI
0x482F	UI_CLK_TRAIL_ MIN	0x00	RW	Minimum UI Value of clk_trail, unit: UI
0x4830	UI_LPX_P_MIN	0x00	RW	Minimum UI Value of lpx_p, unit: UI
0x4831	UI_HS_PREPARE_ MIN	0x04	RW	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	Minimum UI Value of hs_exit, unit: UI
0x4833	MIPI_REG_MIN_H	0x00	RW	MIPI RW Register Address Lower Boundary High Byte
0x4834	MIPI_REG_MIN_L	0x00	RW	MIPI RW Register Address Lower Boundary Low Byte
0x4835	MIPI_REG_MAX_H	0xFF	RW	MIPI RW Register Address Top Boundary High Byte
0x4836	MIPI_REG_MAX_L	0xFF	RW	MIPI RW Register Address Top Boundary Low Byte
0x4837	PCLK_PERIOD	0x19	RW	Period of Pclk2x, pclk_div = 1, and 1-bit Decimal
0x4838	WKUP_DLY	0x02	RW	Wakeup Delay for MIPI (Mark1 state)/2^12

table 7-16 MIPI top control registers (sheet 8 of 9)

address	register name	default value	R/W	description
0x4839	RSVD	–	–	Reserved
0x483A	DIR_DLY	0x08	RW	Change LP Direction Delay/2 After LP11
0x483B	MIPI_LP_GPIO	0x33	RW	Bit[7]: lp_sel1 0: Auto generate mipi_lp_dir1_o 1: Use lp_dir_man1 to be mipi_lp_dir1_o Bit[6]: lp_dir_man1 0: Input 1: Output Bit[5]: lp_p1_o Bit[4]: lp_n1_o Bit[3]: lp_sel2 0: Auto generate mipi_lp_dir2_o 1: Use lp_dir_man2 to be mipi_lp_dir2_o Bit[2]: lp_dir_man2 0: Input 1: Output Bit[1]: lp_p2_o Bit[0]: lp_n2_o
0x483C	MIPI_CTRL3C	0x4F	RW	Bit[7:4]: t_lpx Unit: SCLK cycle Bit[3:0]: t_clk_pre Unit: pclk2x cycle
0x483D	T_TA_GO	0x10	RW	Unit: sclk cycle
0x483E	T_TA_SURE	0x06	RW	Unit: sclk cycle
0x483F	T_TA_GET	0x14	RW	Unit: sclk cycle
0x4840~ 0x4845	RSVD	–	–	Reserved
0x4846	MIPI_CLIP_MAX	0x0F	RW	Bit[7:4]: Not used Bit[3:0]: MIPI output data maximum value[11:8]
0x4847	MIPI_CLIP_MAX	0xFF	RW	Bit[7:0]: MIPI output data maximum value[7:0]
0x4848	MIPI_CLIP_MIN	0x0F	RW	Bit[7:4]: Not used Bit[3:0]: MIPI output data minimum value[11:8]
0x4849	MIPI_CLIP_MIN	0xFF	RW	Bit[7:0]: MIPI output data minimum value[7:0]
0x4850	REG_INTR_MAN	–	W	Generate 1 SCLK Cycle Pulse for MCU Interrupt
0x4851	REG_TX_WR	–	W	Generate 1 SCLK Cycle Pulse to MIPI_TX_LP_TX, and reg_wdata Will Be Sent Out Through MIPI Escape Mode
0x4852	REG_TX_STOP	–	W	Generate 1 SCLK Cycle Pulse to MIPI_TX_LP_TX, and MIPI_TX_LP_TX Will Go Back to LP11
0x4853	REG_TA_ACK	–	W	Generate 1 SCLK Cycle Pulse to MIPI_TX_LP_TX to Receive TurnAround Command

table 7-16 MIPI top control registers (sheet 9 of 9)

address	register name	default value	R/W	description
0x4854	REG_TA_REQ	–	W	Generate 1 SCLK Cycle Pulse to MIPI_TX_LP_TX to Send TurnAround Command
0x4860	DEBUG CTRL	–	–	Debug Control
0x4861	HD_SK_REG0	–	R	MIPI RW, SCCB and MCU Read Only
0x4862	HD_SK_REG1	–	R	MIPI RW, SCCB and MCU Read Only
0x4863	HD_SK_REG2	–	R	MIPI RW, SCCB and MCU Read Only
0x4864	HD_SK_REG3	–	R	MIPI RW, SCCB and MCU Read Only
0x4865	MIPI_ST	–	R	Bit[7:6]: Not used Bit[5]: lp_rx_sel_i 0: Not used 1: MIPI_LP_RX receiving LP data Bit[4]: tx_busy_i 0: Not used 1: MIPI_TX_LP_TX is busy sending LP data Bit[3]: mipi_lp_p1_i MIPI low power input for lane1 P Bit[2]: mipi_lp_n1_i MIPI low power input for lane1 N Bit[1:0]: Debug control

7.17 LVDS control [0x4A00, 0x4A02 - 0x4A0F]

table 7-17 LVDS registers

address	register name	default value	RW	description
0x4A00	LVDS_R0	0x2A	RW	Bit[7]: Debug control Bit[6]: SYNC code manual mode enable Bit[5]: SYNC code enable when only 1 lane Bit[4]: PCLK invert enable Bit[3]: Channel ID enable in sync per lane mode Bit[2]: F parameter in CCIR656 standard Bit[1]: SAV first enable Bit[0]: SYNC code mode 0: Split 1: Per lane
0x4A02	LVDS_R2	0x00	RW	Bit[7:0]: Dummy data0[15:8]
0x4A03	LVDS_R3	0x80	RW	Bit[7:0]: Dummy data0[7:0]
0x4A04	LVDS_R4	0x00	RW	Bit[7:0]: Dummy data1[15:8]
0x4A05	LVDS_R5	0x10	RW	Bit[7:0]: Dummy data1[7:0]
0x4A06	LVDS_R6	0xAA	RW	Blanking line_start Sync Code in Manual Sync Code Mode
0x4A07	LVDS_R7	0x55	RW	Blanking line_end Sync Code in Manual Sync Code Mode
0x4A08	LVDS_R8	0x99	RW	Video line_start Sync Code in Manual Sync Code Mode
0x4A09	LVDS_R9	0x66	RW	Video line_end Sync Code in Manual Sync Code Mode
0x4A0A~ 0x4A0C	DEBUG CTRL	–	–	Debug Control
0x4A0D	LVDS_R0D	0x00	RW	Bit[7:1]: Debug control Bit[0]: LVDS enable
0x4A0E~ 0x4A0F	DEBUG CTRL	–	–	Debug Control

7.18 ISP top [0x5000 - 0x5017, 0x5E00 - 0x5E08]

table 7-18 ISP top registers (sheet 1 of 3)

address	register name	default value	R/W	description
0x5000	ISP CTRL 00	0x85	RW	Bit[7:6]: isp_sof_sel 00: sof_auto (default pre_isp SOF) 01: ISP VSYNC input 10: tc_sof_i 11: pre_isp_sof Bit[5]: isp_eof_sel 0: ISP window EOF 1: tc_eof_i Bit[4:3]: Debug control Bit[2]: window_enable 0: Disable 1: Enable Bit[1]: awb_gain_en 0: Disable 1: Enable Bit[0]: blc_enable 0: Disable 1: Enable
0x5001	ISP CTRL 01	0x00	RW	Bit[7]: Digital compensation enable 0: Disable 1: Enable Bit[6:5]: Debug control Bit[4]: ISP latch enable 0: Disable 1: Enable Bit[3]: ISP size manual enable 0: Disable 1: Enable Bit[2]: AWB bias enable 0: Disable 1: Enable Bit[1]: Bypass ISP option 1 1: When bypass ISP option 0 is disabled, will output data after awb_gain Bit[0]: Bypass ISP option 0 1: Output data directly from ISP input
0x5002	ISP CTRL 02	0x00	RW	Bit[7:1]: Debug control Bit[0]: manual_x_addr_st[8]
0x5003	ISP CTRL 03	0x00	RW	Bit[7:0]: manual_x_addr_st[7:0]
0x5004	ISP CTRL 04	0x00	RW	Bit[7:1]: Debug control Bit[0]: manual_y_addr_st[8]

table 7-18 ISP top registers (sheet 2 of 3)

address	register name	default value	R/W	description
0x5005	DEBUG CTRL	–	–	Debug Control
0x5006	ISP CTRL 06	0x00	RW	Bit[7:1]: Debug control Bit[0]: manual_x_addr_end[8]
0x5007	ISP CTRL 07	0x00	RW	Bit[7:0]: manual_x_addr_end[7:0]
0x5008	ISP CTRL 08	0x00	RW	Bit[7:1]: Debug control Bit[0]: manual_y_addr_end[8]
0x5009	ISP CTRL 09	0x00	RW	Bit[7:0]: manual_y_addr_end [7:0]
0x500A~ 0x500F	DEBUG CTRL	–	–	Debug Control
0x5010	SENSOR BIAS	–	R	Sensor Bias Debug
0x5011	LINEAR GAIN	–	R	AEC Linear Gain Debug
0x5012	AWB_GAIN_R	–	R	Bit[7:4]: Debug control Bit[3:0]: AWB gain R channel[11:8]
0x5013	AWB_GAIN_R	–	R	Bit[7:0]: AWB gain R channel[7:0]
0x5014	AWB_GAIN_G	–	R	Bit[7:4]: Debug control Bit[3:0]: AWB gain G channel[11:8]
0x5015	AWB_GAIN_G	–	R	Bit[7:0]: AWB gain G channel[7:0]
0x5016	AWB_GAIN_B	–	R	Bit[7:4]: Debug control Bit[3:0]: AWB gain B channel[11:8]
0x5017	AWB_GAIN_B	–	R	Bit[7:0]: AWB gain B channel[7:0]
0x5E00	PRE_ISP 00	0x0C	RW	Bit[7]: Test pattern bar enable 0: Disable 1: Enable Bit[6]: Debug control Bit[5]: Mirror option for X offset Bit[4]: Flip option for Y offset Bit[3]: Mirror order (P1P2 or P2P1) Bit[2]: Flip order (P1P4 or P4P1) Bit[1:0]: Debug control
0x5E01	PRE_ISP 01	–	R	Window X Offset High Byte
0x5E02	PRE_ISP 02	–	R	Window X Offset Low byte
0x5E03	PRE_ISP 03	–	R	Window Y Offset High Byte
0x5E04	PRE_ISP 04	–	R	Window X Offset Low byte
0x5E05	PRE_ISP 05	–	R	Window X Output Size High Byte
0x5E06	PRE_ISP 06	–	R	Window X Output Size Low Byte

table 7-18 ISP top registers (sheet 3 of 3)

address	register name	default value	R/W	description
0x5E07	PRE_ISP 07	–	R	Window Y Output Size High Byte
0x5E08	PRE_ISP 08	–	R	Window Y Output Size Low Byte

7.19 window control [0x5A00 - 0x5A0C]

table 7-19 window control registers

address	register name	default value	R/W	description
0x5A00	MAN_XSTART_OFF	0x00	RW	Bit[7:5]: Not used Bit[4:0]: X start offset[12:8]
0x5A01	MAN_XSTART_OFF	0x00	RW	Bit[7:0]: X start offset[7:0]
0x5A02	MAN_YSTART_OFF	0x00	RW	Bit[7:4]: Not used Bit[3:0]: Y start offset[11:8]
0x5A03	MAN_YSTART_OFF	0x00	RW	Bit[7:0]: Y start offset[7:0]
0x5A04	MAN_WIN_WIDTH	0x10	RW	Bit[7:5]: Not used Bit[4:0]: Window width[12:8]
0x5A05	MAN_WIN_WIDTH	0xA0	RW	Bit[7:0]: Window width[7:0]
0x5A06	MAN_WIN_HEIGHT	0x0C	RW	Bit[7:4]: Not used Bit[3:0]: Window height[11:8]
0x5A07	MAN_WIN_HEIGHT	0x78	RW	Bit[7:0]: Window height[7:0]
0x5A08	WIN_MAN	0x00	RW	Bit[7:1]: Not used Bit[0]: Window size manual 0: Disable 1: Enable
0x5A09	WIN_PX_CNT	–	R	Bit[7:5]: Not used Bit[4:0]: Pixel counter[12:8]
0x5A0A	WIN_PX_CNT	–	R	Bit[7:0]: Pixel counter[7:0]
0x5A0B	WIN_LN_CNT	–	R	Bit[7:4]: Not used Bit[3:0]: Line counter[11:8]
0x5A0C	WIN_LN_CNT	–	R	Bit[7:0]: Line counter[7:0]

8 operating specifications

8.1 absolute maximum ratings

table 8-1 absolute maximum ratings

parameter		absolute maximum rating[a]
ambient storage temperature		-40°C to +125°C
supply voltage (with respect to ground)	V_{DD-A}	4.5V
	V_{DD-D}	3V
	V_{DD-IO}	4.5V
electro-static discharge (ESD)	human body model	2000V
	machine model	200V
all input/output voltages (with respect to ground)		-0.3V to V_{DD-IO} + 1V
I/O current on any input or output pin		± 200 mA

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

8.2 functional temperature

table 8-2 functional temperature

parameter	range
operating temperature[a]	-30°C to +70°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

8.3 DC characteristics

table 8-3 DC characteristics (T_A = 23°C ± 2°C)

symbol	parameter	min	typ	max	unit
supply					
V_{DD-A}	supply voltage (analog)	2.6	2.8	3.0	V
V_{DD-IO}	supply voltage (digital I/O)	1.7	1.8	3.0	V
I_{DD-A}	active (operating) current[a]	12	16	20	mA
I_{DD-IO}		28	41	55	mA
$I_{DDS-SCCB}$[b]	standby current	0.50	1.2	1.5	mA
$I_{DDS-SCCB}$		20	50	100	µA
$I_{DDS-XSHUTDOWN}$		2	10	30	µA
digital inputs (typical conditions: AVDD = 2.8V, DVDD = 1.5V, DOVDD = 1.8V)					
V_{IL}	input voltage LOW			0.54	V
V_{IH}	input voltage HIGH	1.26			V
C_{IN}	input capacitor			10	pF
digital outputs (standard loading 25 pF)					
V_{OH}	output voltage HIGH	1.62			V
V_{OL}	output voltage LOW			0.18	V
serial interface inputs					
V_{IL}[c]	SIOC and SIOD	-0.5	0	0.54	V
V_{IH}	SIOC and SIOD	1.28	1.8	3.0	V

a. 640x480 @ 120 fps

b. with EXTCLK

c. based on DOVDD = 1.8V

8.4 timing characteristics

table 8-4 timing characteristics

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

a. if using internal PLL

9 mechanical specifications

9.1 physical specifications

figure 9-1 package specifications

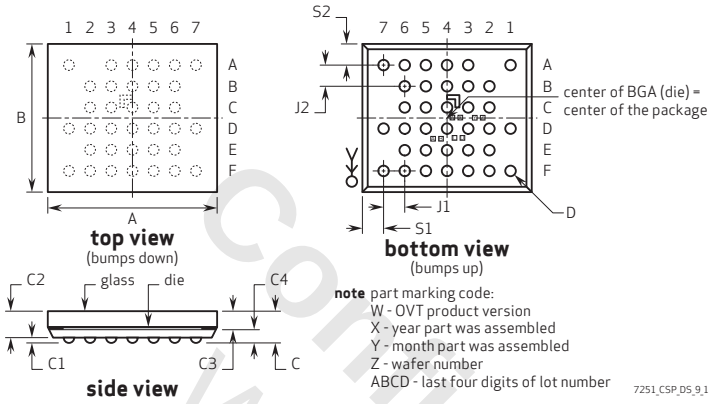


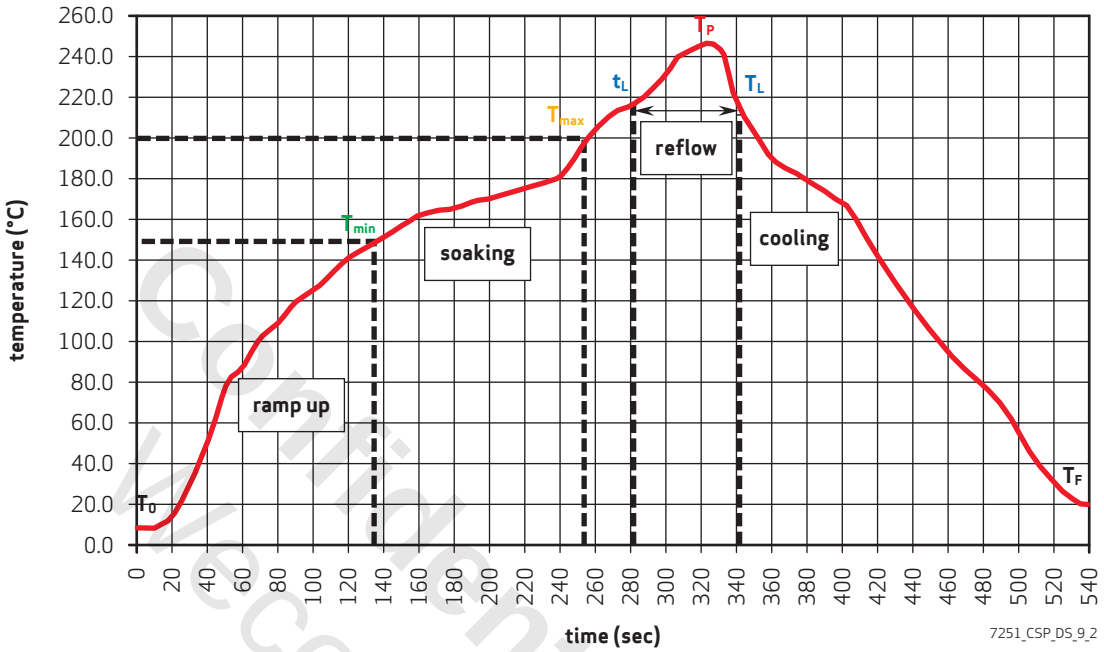
table 9-1 package dimensions

parameter	symbol	min	typ	max	unit
package body dimension x	A	3885	3910	3935	µm
package body dimension y	B	3385	3410	3435	µm
package height	C	700	760	820	µm
ball height	C1	100	130	160	µm
package body thickness	C2	585	630	675	µm
thickness of glass surface to wafer	C3	425	445	465	µm
image plane height	C4	260	315	370	µm
ball diameter	D	220	250	280	µm
total pin count	N		35 (9 NC)		
pin count x-axis	N1		7		
pin count y-axis	N2		6		
pins pitch x-axis	J1		500		µm
pins pitch y-axis	J2		520		µm
edge-to-pin center distance along x	S1	425	455	485	µm
edge-to-pin center distance along y	S2	375	405	435	µm
air gap between die and glass		40	45	50	µm

9.2 IR reflow specifications

note
The OV7251 uses a lead free package.

figure 9-2 IR reflow ramp rate requirements



note OmniVision recommends CSP packages use underfill as a part of camera assembly process

table 9-2 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 30 sec)
reflow (t_L to T_L)	temperature higher than 217°C	30 ~ 120 seconds
ramp down A (T_p to T_L)	cooling from 245°C to 217°C	temperature slope ≤ 3°C per second
ramp down B (T_L to T_f)	cooling from 217°C to room temperature	temperature slope ≤ 2°C per second
T_0 to T_p	room temperature to peak temperature	≤ 8 minutes

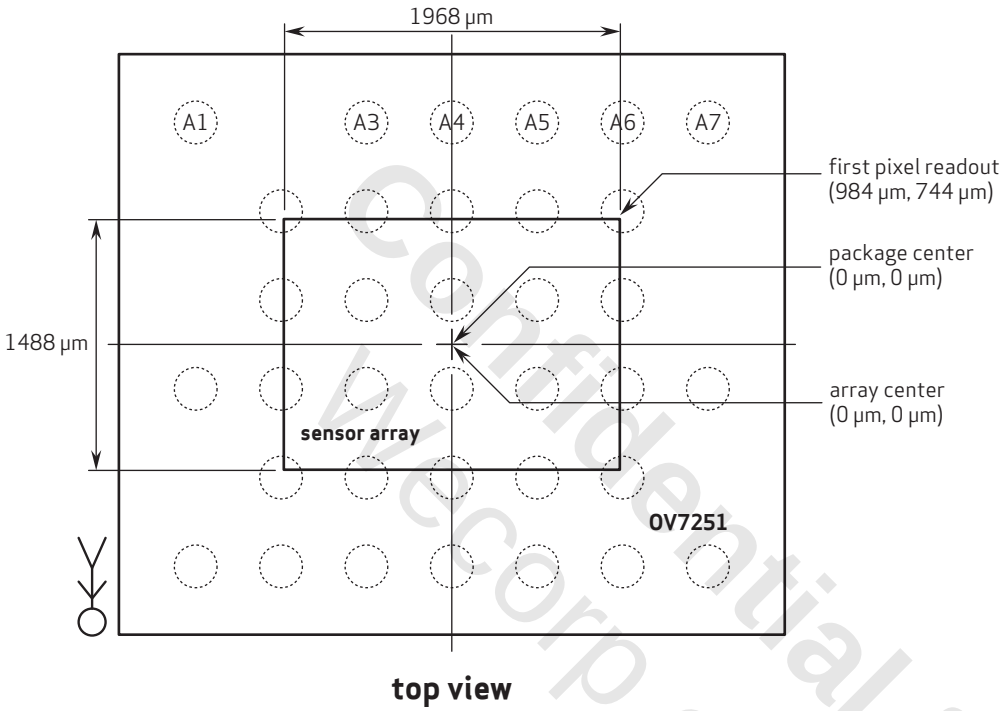
a. maximum number of reflow cycles = 3

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

10 optical specifications

10.1 sensor array center

figure 10-1 sensor array center



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

note 2 as most optical assemblies invert and mirror the image, the chip is typically mounted with pins A1 to A7 oriented down on the PCB.

7251_CSP_DS_10_1

10.2 lens chief ray angle (CRA)

figure 10-2 chief ray angle (CRA)

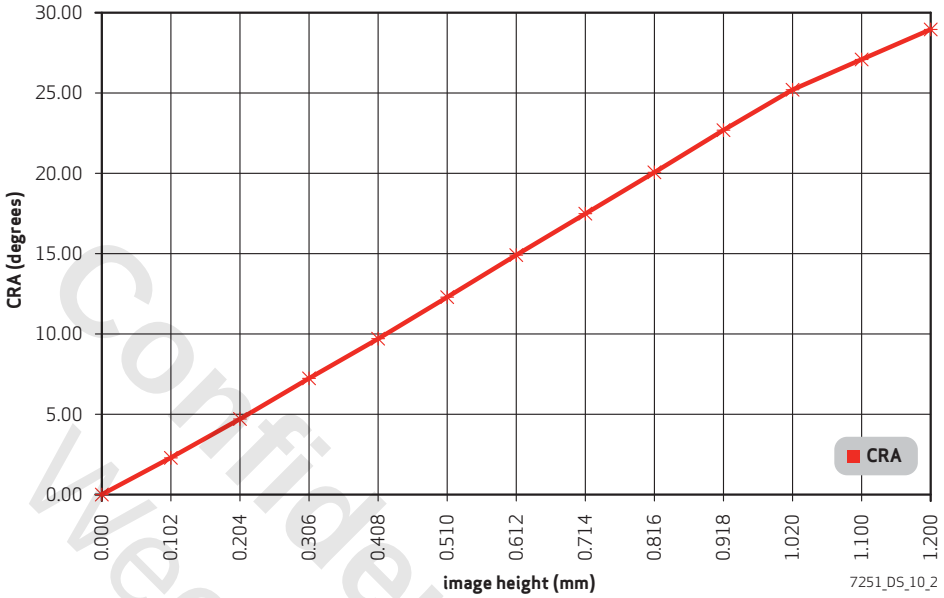


table 10-1 CRA versus image height plot

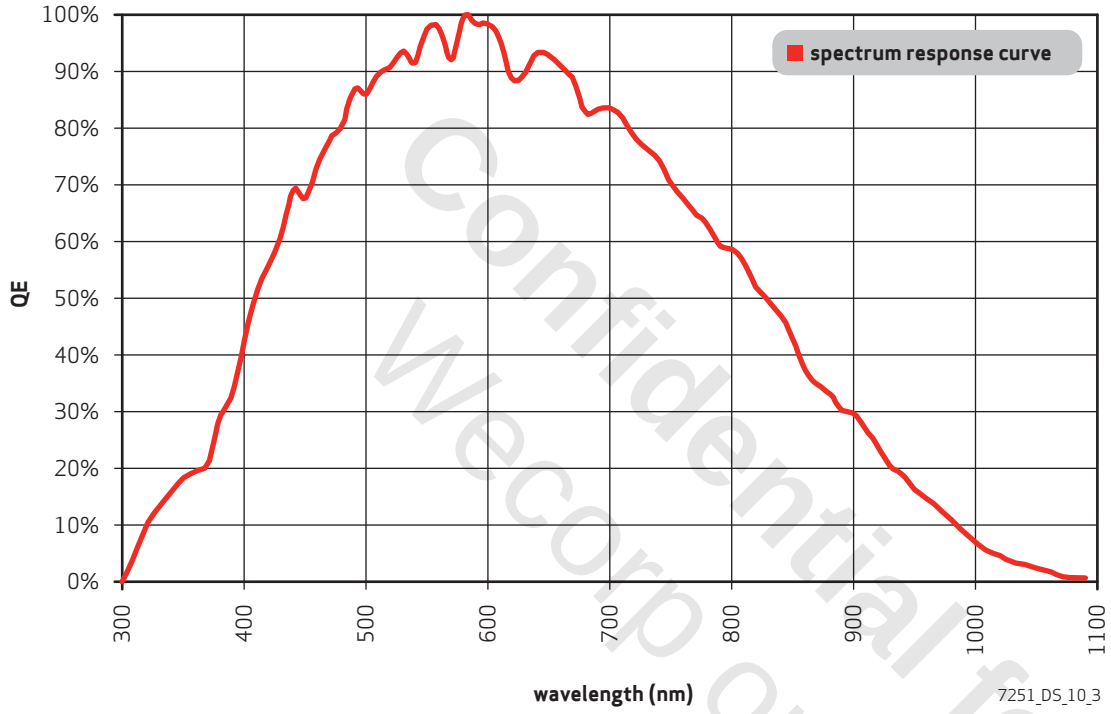
field (%)	image height (mm)	CRA (degrees)
0	0	0
0.085	0.102	2.3
0.17	0.204	4.7
0.255	0.306	7.2
0.34	0.408	9.7
0.425	0.51	12.3
0.51	0.612	14.9
0.595	0.714	17.5
0.68	0.816	20.1
0.765	0.918	22.7
0.85	1.02	25.2
0.916666667	1.1	27.1
1	1.2	29

10.3 IR cut off wavelength

Wavelength above 975 nm must be cut off to avoid package structure ghosting.

10.4 spectrum response

figure 10-3 spectrum response curve



revision history

version 1.0 08.28.2017

- initial release

version 1.1 12.22.2017

- in section 4.4, removed table 4-6
- in table 6-2, added row for register 0x4A0D
- in section 6.2, added paragraph after table 6-2
- in chapter 7, updated description of register 0x4501 and moved it from table 7-1 to table 7-15
- in section 7.8, changed title to "timing control [0x3800 ~ 0x3835, 0x3837, 0x3880 ~ 0x38B5]"
- in table 7-8, added a row for registers 0x3880~0x38B5
- in table 7-10, updated description of registers 0x3C00 and 0x3C01 and added a row for registers 0x4E00~0x4E30
- in table 7-13, removed register 0x4244
- in section 7.15, changed title to "VFIFO control [0x4500 ~ 0x4502, 0x4600 ~ 0x4604]"
- in table 7-15, added a row for register 0x4500 and added a row for register 0x4502
- in section 7.17, changed title to "LVDS control [0x4A00, 0x4A02 ~ 0x4A0F]", added description for register 0x4A0D, and removed registers 0x4A40~0x4A4D

defining the future of digital imaging™

OmniVision Technologies, Inc.

UNITED STATES

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

UNITED KINGDOM

Hook, Hampshire +44 1256 744 610

GERMANY

Munich +49 89 63 81 99 88

INDIA

Bangalore +91 80 4112 8966

CHINA

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

JAPAN

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

KOREA

Seoul + 82 2 3472 3769

SINGAPORE +65 6933 1933

TAIWAN

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

website: **www.ovt.com**