



1/5-Inch System-On-A-Chip (SOC) CMOS Digital Image Sensor

MT9D113

For the latest data sheet, refer to Aptina's Web site: www.aplina.com

Features

- 2Mp resolution (1600H x 1200V)
- 1/5-inch optical format
- Same or better image quality compared to MT9D112
- Individual module ID support through one-time programmable (OTP) memory
- Surface fit lens correction (LC) to compensate for lens/small pixel vignetting and corner color variations
- Automatic functions: Exposure, white balance, black level offset correction, flicker detection and avoidance, color saturation control, defect identification and correction, aperture correction, and GPIO
- Programmable controls: Exposure, white balance, horizontal and vertical blanking, color, sharpness, gamma, lens shading correction, horizontal and vertical image flip, zoom, windowing, sampling rates, and GPIO
- 15 frames per second (fps) at 1600H x 1200V with moderate pixel clock frequency (≤ 65 MHz) to minimize baseband reception interference and 30 fps at 800H x 600V
- 2 x 2 pixel binning to improve low-light image quality
- Support for external LED or xenon flash
- On-chip phase-locked loop (PLL) to minimize the number of system clocks
- Low power modes to prolong battery life of portable devices
- Fail-safe I/Os with programmable output slew rate
- Industry standard two-wire serial interface for controls
- 10-bit parallel or MIPI serial interfaces for image data

Applications

- Cellular phones
- PC cameras
- PDAs

Table 1: Key Performance Parameters

Parameter	Value
Pixel size	Symmetric 1.75 μ m pixel
Optical format	1/5-inch
Targeted module size (estimated)	6.5mm x 6.5mm x 5mm
Targeted die size	5.06mm x 5.24mm
Array format (active)	1600H x 1200V (4:3 aspect ratio)
Imaging area	2.8mm x 2.10mm: 3.50mm diagonal (4:3 aspect ratio)
CRA	25° (MAX at 90% image height)
Color filter array	RGB color filters
Frame rate	15 fps (full resolution), 30 fps (video mode)
Scan mode	Progressive
Shutter	Electronic rolling shutter (ERS)
Windowing	Programmable
Output Interfaces	MIPI single-lane or parallel 10-bit
Data rate	MIPI: Up to 640 Mb/s and up to 85 MHz parallel data Parallel: 80 MB/s (parallel no FIFO), 63 MB/s (parallel with FIFO)
Responsivity	0.2 V/Lux-sec (550nm)
Signal-to-noise ratio	>37dB (MAX)
Dynamic range	>60dB (pixel)
Full well	>6900e- (linear)
Read noise	4e- (8x Gain)
Supply voltage	Analog: 2.8V (nominal) Digital: 1.8V (nominal) I/O: 1.8V or 2.8V (nominal) OTP: 8V (nominal)
Operating temperature range	-30°C to 70°C (at junction)
Package	Bare die (in wafer form)

Ordering Information

Table 2: Available Part Numbers

Part Number	Description
MT9D113D00STCK25AC1	Bare die



Table of Contents

Features	1
Applications	1
Ordering Information	1
Overview	6
Typical Connections	6
Signal Descriptions	8
Architecture	9
Firmware	10
External Host Interface (Two-Wire Slave-Only Interface)	11
Always-On Power Domain	13
Sensor Core	13
Pixel Array	14
Analog Processing	15
PLL-Generated Master Clock	16
PLL Setup	16
Digital Processing	17
Readout Modes	17
Input Interface to Image Flow Processor	24
Image Flow Processor	25
Pixel Reconstruction (Lens Shading Correction)	26
Color Rendering/Statistics Collection (Color Correction)	27
Digital Scaling/Output Format	28
YUV-to-RGB/YUV Conversion and Output Formatting	29
Output Interface from IFP	30
Parallel and MIPI Output	30
Output Format and Timing	30
Control Functions	34
Sequencer: Camera Operating System	34
Mode: Context Information	35
Camera Functions	37
Simple Rule-based Auto Exposure	37
AE Driver	37
Evaluative Algorithm	38
Accelerated Settling During Overexposure	38
Exposure Control	38
Flicker Detection and Avoidance	38
Auto White Balance	39
Flash	39
Histogram: Dark Level Adjustments, Low Light and Tonal Controls	39
Optics	40
Power Modes	41
Power Application Sequence	41
Power-On Reset	42
Hard Reset	43
Soft Reset	44
Power Removal Sequence	45
Standby Modes	46
Soft Standby	46
Hard Standby	46
Hard Standby Mode	47
Entering Standby Mode	47



MT9D113: 1/5-Inch SOC Digital Image Sensor

Table of Contents

Exiting Standby Mode	47
Soft Standby Mode	48
Entering Standby Mode	48
Exiting Standby Mode	48
Other Features	49
One-time Programmable (OTP) Memory	49
Programming the OTP Memory	49
Reading the OTP Memory	50
GPIO and Output Enable Controls	51
General Purpose I/Os	51
Output Enable Control	51
GPIO Control	52
Overview of GPIO Signals	52
Electrical Specifications	53
Absolute Maximum Rating	53
I/O Parameters	54
AC Electricals	54
DC Electricals	55
Appendix A - Registers	57
SYSCTL Registers	57
XDMA Registers	61
GPIO Registers	63
Core Registers	65
SOC1 Registers	67
SOC2 Registers	73
Appendix B - Driver Variables	85
Sequencer Variables (ID = 1)	85
Auto Exposure Variables (ID = 2)	93
Auto White Balance Variables (ID = 3)	96
Anti-Flicker Variables (ID = 4)	101
Mode Variables (ID = 7)	103
Histogram Variables (ID = 11)	113
Revision History	118



List of Figures

Figure 1:	MT9D113 Typical Connections ¹ (Single Camera)	6
Figure 2:	SOC Block Diagram	9
Figure 3:	Firmware Architecture.	10
Figure 4:	External Host Register Map	11
Figure 5:	Two-Wire Serial Control Bus Timing	12
Figure 6:	Sensor Core Block Diagram	14
Figure 7:	Pixel Color Pattern Detail (Top Right Corner)	14
Figure 8:	Imaging a Scene	15
Figure 9:	6 Pixels in Normal and Column Mirror Readout Modes	17
Figure 10:	6 Rows in Normal and Row Mirror Readout Modes	18
Figure 11:	8 Pixels in Normal and Column Skip 2X Readout Modes.	18
Figure 12:	Pixel Readout (no skipping)	19
Figure 13:	Pixel Readout (x_odd_inc = 3, y_odd_inc = 1)	19
Figure 14:	Pixel Readout (x_odd_inc = 1, y_odd_inc = 3)	20
Figure 15:	Pixel Readout (x_odd_inc = 3, y_odd_inc = 3)	20
Figure 16:	Pixel Readout (x_odd_inc = 3, y_odd_inc = 1, x_bin = 1)	21
Figure 17:	Pixel Readout (x_odd_inc = 3, y_odd_inc = 3, x_ybin = 1)	22
Figure 18:	Valid Image Data.	23
Figure 19:	Pixel Data Timing Example.	23
Figure 20:	Available Test Patterns	24
Figure 21:	IFP Block Diagram	25
Figure 22:	Gamma Correction Curve.	28
Figure 23:	Timing of Full Frame Data or Scaled Data Passing Through the FIFO	31
Figure 24:	Output Interface Timing Waveforms	32
Figure 25:	Sequencer Finite State Machine	35
Figure 26:	Chief Ray Angle (CRA) vs. Image Height	40
Figure 27:	Power Application Sequence Timing	41
Figure 28:	Internal Power-On Reset.	42
Figure 29:	Hard Reset Operation	43
Figure 30:	Soft Reset Operation	44
Figure 31:	Hard Standby Mode Operation	47
Figure 32:	Soft Standby Mode Operation	48
Figure 33:	Sequence of Signals for OTP Memory Operation	50



List of Tables

Table 1:	Key Performance Parameters	1
Table 2:	Available Part Numbers	1
Table 3:	Signal Description and Direction	8
Table 4:	List of Drivers	10
Table 5:	Two-Wire Serial Control Bus Timing Parameters	12
Table 6:	Row Address Sequencing (Sampling)	21
Table 7:	Row Address Sequencing (Binning)	22
Table 8:	Data Formats Supported by MIPI Interface	30
Table 9:	YCrCb Output Data Ordering	31
Table 10:	RGB Ordering in Default Mode	31
Table 11:	2-Byte RGB Format	32
Table 12:	Parallel Image Bus Timing Parameters	32
Table 13:	Output Interface Timing	32
Table 14:	POR Parameters	42
Table 15:	Hard Reset	43
Table 16:	Soft Reset Signal Timing	44
Table 17:	Hard Standby Signal Timing	47
Table 18:	Soft Standby Signal Timing	48
Table 19:	GPIO Related Registers and Variables	52
Table 20:	Maximum Rating	53
Table 21:	I/O Parameters	54
Table 22:	AC Electricals	54
Table 23:	DC Electricals	55
Table 24:	SYSCTL Registers Sorted by Name	57
Table 25:	Description of SYSCTL Registers	57
Table 26:	XDMA Registers Sorted by Name	61
Table 27:	Description of XDMA Registers	62
Table 28:	GPIO Registers Sorted by Name	63
Table 29:	Description of GPIO Registers	63
Table 30:	Core Registers Sorted by Name	65
Table 31:	Description of Core Registers	65
Table 32:	SOC1 Registers Sorted by Name	67
Table 33:	Description of SOC1 Registers	68
Table 34:	SOC2 Registers Sorted by Name	73
Table 35:	Description of SOC2 Registers	76
Table 36:	Sequencer Variables Sorted by Name	85
Table 37:	Description of Sequencer Variables	86
Table 38:	Auto Exposure Variables Sorted by Name	93
Table 39:	Description of Auto Exposure Variables	93
Table 40:	Auto White Balance Variables Sorted by Name	96
Table 41:	Description of Auto White Balance Variables	97
Table 42:	Anti-Flicker Variables Sorted by Name	101
Table 43:	Description of Anti-Flicker Variables	102
Table 44:	Mode Variables Sorted by Name	103
Table 45:	Description of Mode Variables	104
Table 46:	Histogram Variables Sorted by Name	113
Table 47:	Description of Histogram Variables	114



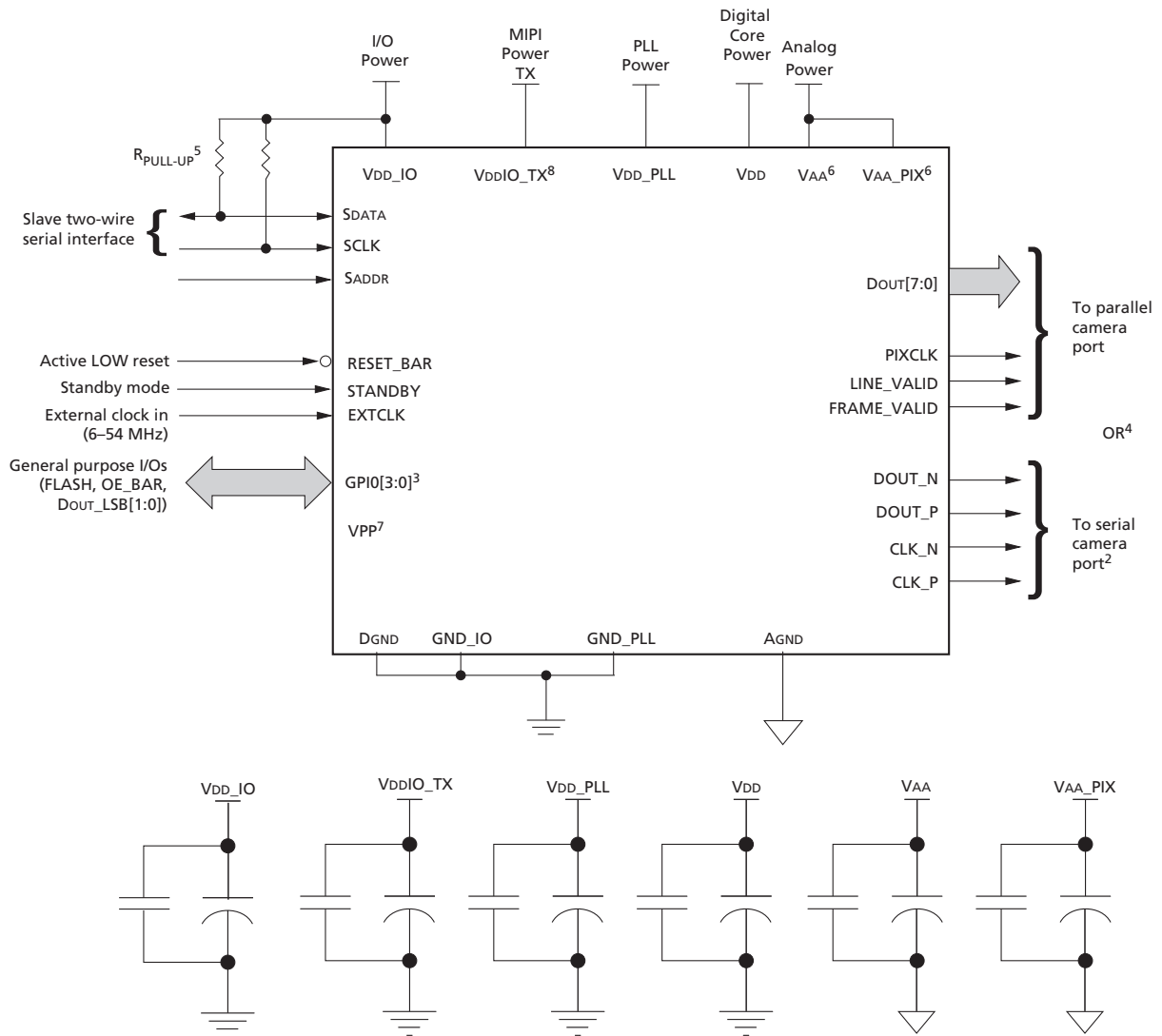
MT9D113: 1/5-Inch SOC Digital Image Sensor Overview

Overview

Aptina's MT9D113 is a complete camera system-on-a-chip (SOC) image sensor that outputs sharp, clear, progressive scan images while extending an application's battery life. It provides near-CCD image quality while maintaining a small form factor.

Typical Connections

Figure 1: MT9D113 Typical Connections¹ (Single Camera)



- Notes:
1. This typical configuration shows only one scenario out of multiple possible variations for this sensor.
 2. If a MIPI Interface is not required, the following pads must be left floating: DOUT_P, DOUT_N, CLK_P, and CLK_N.
 3. The general purpose input/output (GPIO) pads can serve multiple features that can be reconfigured. The function and direction will vary by applications.
 4. Only one of the output modes (serial or parallel) can be used at any time.



MT9D113: 1/5-Inch SOC Digital Image Sensor Typical Connections

5. Aptina recommends a resistor value of $1.5K\Omega$ to V_{DD_IO} for the two-wire serial interface $R_{PULL-UP}$; however, greater values may be used for slower transmission speed.
6. V_{AA} and V_{AA_PIX} may be tied together. Although separate decoupling capacitors are recommended for V_{AA} and V_{AA_PIX} , decoupling capacitors can be shared if one would like to reduce module size.
7. V_{PP} is the one-time programmable (OTP) memory programming voltage and should be left floating during normal operation.
8. 1.8V supply shared by MIPI interface and V_{DD} to reduce number of decoupling caps, hence, module size. V_{DDIO_TX} must be connected to a 1.8v power supply source even though Mipi interface is not used.
9. Aptina recommends that 0.1 μ F and 1 μ F decoupling capacitors for each power supply are mounted as close as possible to the pad. Actual values and results may vary depending on layout and design considerations.
10. V_{DD_PLL} and V_{AA} can share the same power source in which case GND_PLL must be connected to GND.
11. Internal pull-up in $RESET_BAR$ pin and its value to be determined.



Signal Descriptions

Table 3: Signal Description and Direction

Name	Type	Description	Note
STANDBY	Input	Hardware standby	3
EXTCLK	Input	External clock input	
SADDR	Input	Two-wire interface device select address	
SCLK	Input	Two-wire interface serial clock	
RESET_BAR	Input	Hardware reset	4
CLK_N	Output	MIPI differential clock N	5
CLK_P	Output	MIPI differential clock P	5
DOUT_N	Output	MIPI differential data N	5
DOUT_P	Output	MIPI differential data P	5
DOUT[7:0]	Output	Parallel image data	1
FRAME_VALID	Output	Parallel pixel bus frame valid	1
LINE_VALID	Output	Parallel pixel bus line valid	1
PIXCLK	Output	Parallel pixel bus pixel clock	1
SDATA	Bidirectional	Two-wire interface serial data	
GPIO_0/DOOUT_LSB[0]	Bidirectional/Output	General-purpose I/O or LSB for Raw 10 data output during SOC Bypass	2
GPIO_1/DOOUT_LSB[1]	Bidirectional/Output	General-purpose I/O or LSB for Raw 10 data output during SOC Bypass	2
GPIO_2/OE_BAR	Bidirectional/Output	General-purpose I/O or output enable	2
GPIO_3/FLASH	Bidirectional/Output	General-purpose I/O or flash control	2
VAA	Supply	Analog core power source 2.8V nominal	
VAA_PIX	Supply	Analog core power source 2.8V nominal	
VDD	Supply	Digital core power source 1.8V nominal	
VDD_IO	Supply	Digital IO power source 1.8V or 2.8V nominal	
VDD_PLL	Supply	Digital PLL power source 2.8V nominal	
VDDIO_TX	Supply	Digital MIPI IO power source 1.8V nominal.	6

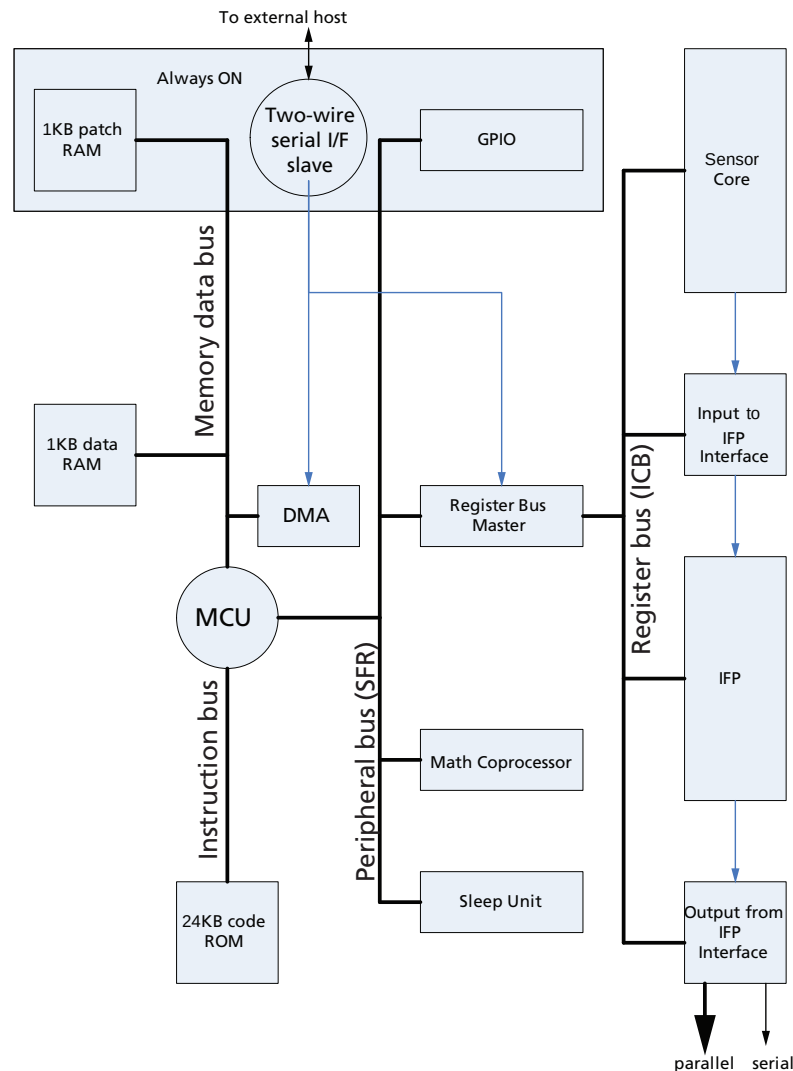
- Notes:
1. In serial only mode, DOUT[7:0], PIXCLK, and GPIO[3:0] can be left floating by setting R0x0026[1] = 1. If GPIO signals are required, DOUT[7:0] and PIXCLK should be tied to DGND and OE_BAR should be tied to VDD_IO. GPIO_2 should be configured as an input for OE_BAR function and set R0x001A[8] = 1.
 2. GPIO can be left floating if not used and must be programmed as outputs.
 3. Must be connected to DGND if not used.
 4. Must be connected to VDD_IO, internal pull-up resistor used.
 5. Can be left floating if not used.
 6. Must be connected to VDD, even in designs where the MIPI interface is not used.



Architecture

MT9D113 is the third-generation, two-megapixel camera system on a chip—a microprocessor-based camera system combining a sensor core with an image flow processor (IFP) to form a standalone solution that includes both image acquisition and processing. Both the sensor core and IFP have internal registers that can be controlled by an external host. In normal operation, the integrated system microprocessor controls most aspects of operation autonomously. The image data is transmitted to the external host system either through a parallel bus or a serial MIPI interface.

Figure 2: SOC Block Diagram



As shown in Figure 2, the external host and the integrated microprocessor (MCU) can both access all the internal resources (RAM and registers). The external host always has higher priority. The following sections briefly describe the functionality of each key component of the system.



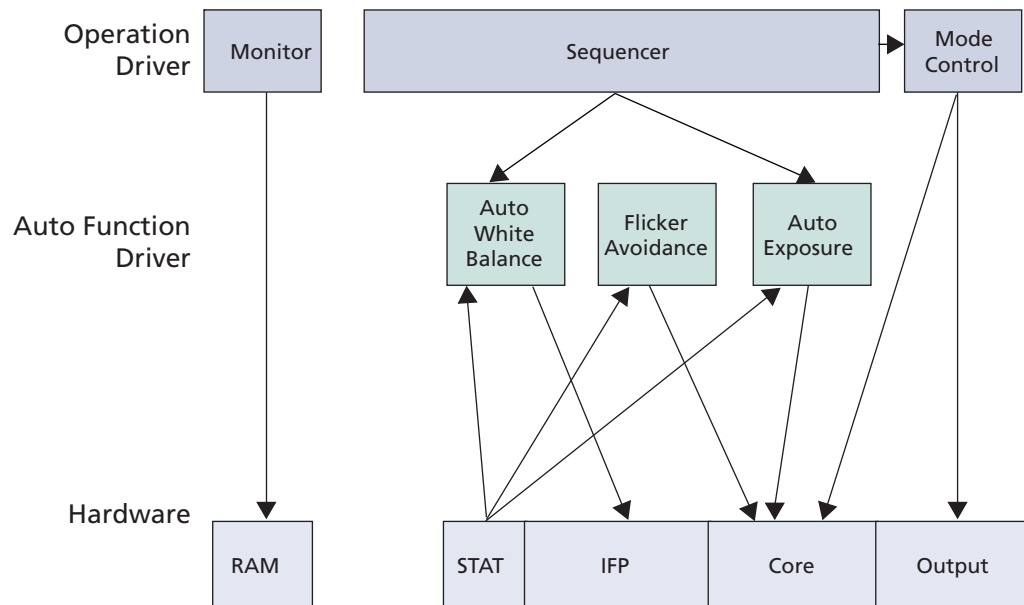
Firmware

The firmware implements all automatic camera functions such as autoexposure (AE), auto white balance (AWB), and flicker detection/avoidance (FD), as well as control functions such as sequencer, mode/context, and histogram. The firmware consists of drivers, generally one driver for each major automatic or control function.

Table 4: List of Drivers

ID	Driver Name	Description
ID = 1	Sequencer	Controls of camera main function
ID = 2	AE	Auto exposure
ID = 3	AWB	Auto white balance
ID = 4	Flicker Detection	Flicker detection and avoidance
ID = 7	Mode/Context	Context variables
ID = 11	Histogram	Reduce image flare and analyze image histogram

Figure 3: Firmware Architecture

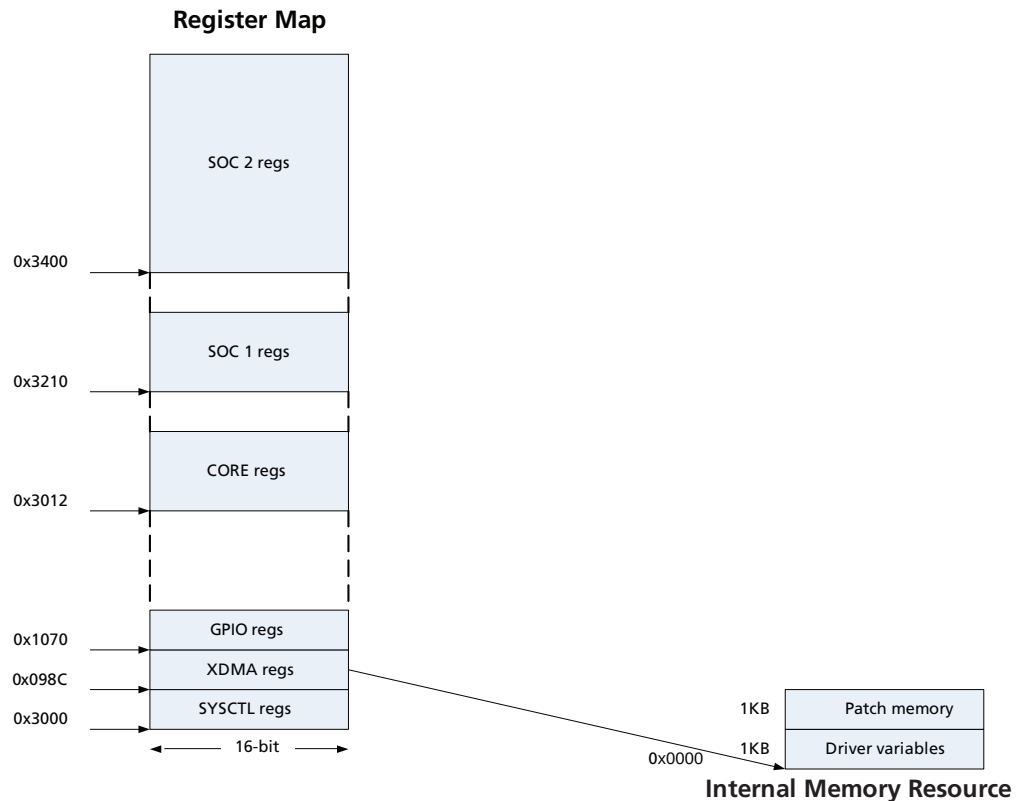




External Host Interface (Two-Wire Slave-Only Interface)

The MT9D113 will appear as a two-wire serial interface slave to the external host and its base address is selectable by the external SADDR pin input (when SADDR = 0 then base address = 0x78; when SADDR = 1 then base address = 0x7A). There are 32K addressable 16-bit registers (that is, the starting address of each register always falls into even addresses) within the MT9D113 but not all of them are being used (see Figure 4).

Figure 4: External Host Register Map



The detailed explanation of the registers are listed in “Appendix A - Registers” on page 57. Most of the registers are self-explanatory; however, XDMA registers are worth explaining here. The XDMA registers allow the external host to indirectly access the MT9D113’s internal memory resources, which include the firmware driver variables. To access the variables in “Appendix B - Driver Variables” on page 85, use logical accesses provided by the XDMA registers.

The external host interface is implemented through a two-wire interface that enables direct read/write access to hardware registers and indirect access to firmware variables within MT9D113. The interface is designed to be compatible with the MIPI alliance standard for camera serial interface 2 (CSI-2) 1.0, which uses the electrical characteristics and transfer protocols of the two-wire serial interface specifications.

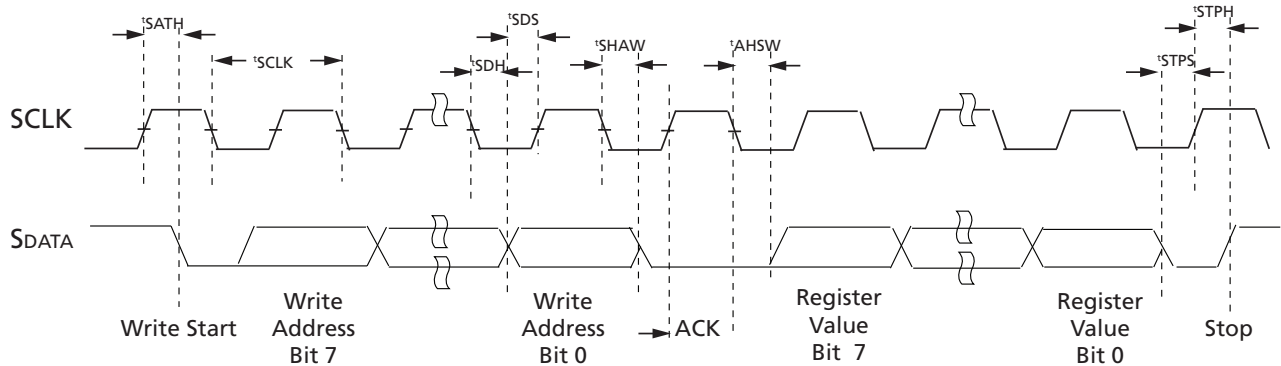
The interface protocol uses a master/slave model in which a master controls one or more slave devices. The sensor acts as a slave device while the external host acts as a master device. The master generates a clock (SCLK) that is an input to the sensor and used to synchronize the transactions at the interface.



Data is transferred between the master and the slave on a bidirectional serial data bus (SDATA). Both SCLK and SDATA are pulled up to VDD_IO off-chip by a 1.5KΩ resistor. Either the slave or master device can drive SDATA to LOW—the interface determines which device is allowed to drive SDATA at any given time.

Figure 5: Two-Wire Serial Control Bus Timing

WRITE Sequence



READ Sequence

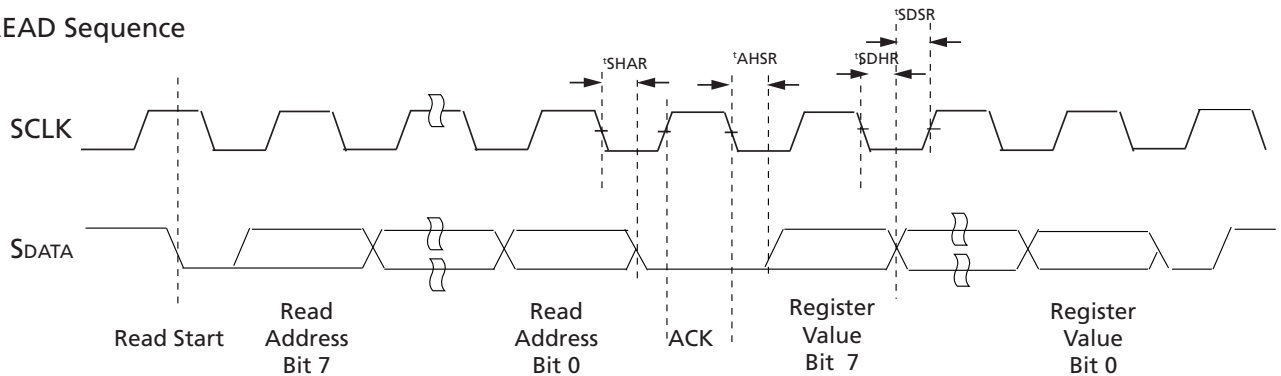


Table 5: Two-Wire Serial Control Bus Timing Parameters

Symbol	Definition	Condition	Min	Typ	Max	Unit
f_{SCLK}	Serial clock frequency		100	—	400	KHz
t_{SCLK}	Serial clock period		2.5	—	10	ms
DCSCLK	Serial clock duty cycle		45	50	55	%
t_{SRTH}	Start hold time	Write/Read	300	—	—	ns
t_{STPH}	Stop hold time	Write/Read	600	—	—	ns
t_{SDH}	SDATA hold	Write	300	—	—	ns
t_{SDS}	SDATA setup	Write	300	—	—	ns
t_{SHAW}	SDATA hold to Ack	Write	300	—	—	ns
t_{AHSW}	ACK hold to SDATA	Write	400	—	—	ns
t_{SDHR}	SDATA hold	Read	300	—	—	ns
t_{SFSR}	SDATA setup	Read	300	—	—	ns
t_{SHAR}	SDATA hold to ACK	Read	300	—	—	ns
t_{AHSR}	ACK hold to SDATA	Read	300	—	—	ns



Always-On Power Domain

The Always-On Power Domain (AOPD) will provide an area of functionality that will always be active while power is applied to the MT9D113. The external host interface is located in the AOPD. The domain also includes miscellaneous clock and reset controls as well as configuration registers for the sensor core, processor core, clock configuration, and clock reset control. The user-loadable patch memory is also included in this domain. This memory will remain powered when the main core power is shut down using the standby command.

Sensor Core

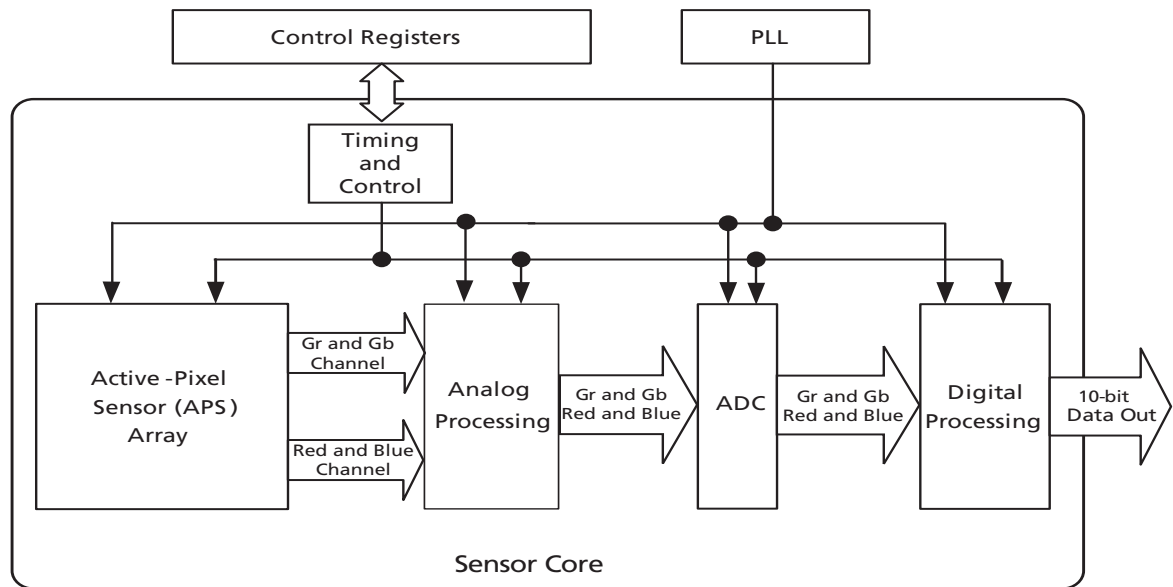
The sensor core of the MT9D113 is a progressive-scan sensor that generates a stream of pixel data at a constant frame rate, qualified by LINE_VALID (LV) and FRAME_VALID (FV). The maximum pixel rate is 30 Mp/s, corresponding to a pixel clock rate of 63.25 MHz. Figure 6 on page 14 shows a block diagram of the sensor core. It includes a 2.0Mp active-pixel array. The timing and control circuitry sequences through the rows of the array, resetting and then reading each row in turn. In the time interval between resetting a row and reading that row, the pixels in the row integrate incident light. The exposure is controlled by varying the time interval between reset and readout. Once a row has been read, the data from the columns are sequenced through an analog signal chain (providing offset correction and gain), and then through an ADC. The output from the ADC is a 10-bit value for each pixel in the array.

The pixel array contains optically active and light-shielded (dark) pixels. The dark pixels are used to provide data for the offset-correction algorithms (black level control).

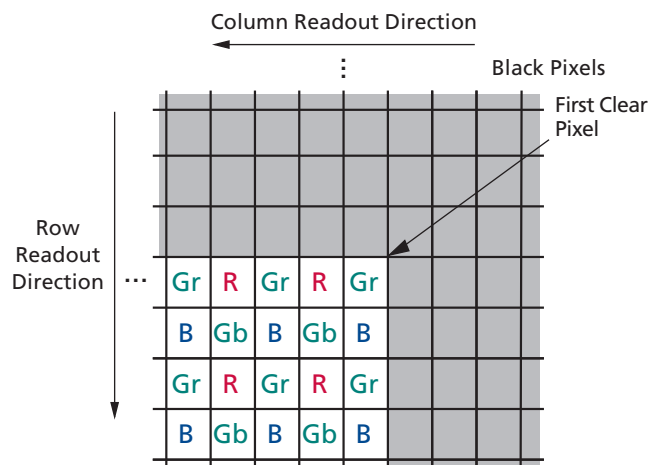
The sensor core contains a set of control and status registers that can be used to control many aspects of the sensor behavior including the frame size, exposure, and gain settings. These registers are controlled by the firmware and can be accessed through a two-wire serial interface. Register values written to the sensor core may be overwritten by firmware.

The output from the core is a Bayer pattern; alternate rows are a sequence of either green and red pixels or blue and green pixels. The offset and gain stages of the analog signal chain provide per-color control of the pixel data.

A flash strobe output signal is provided to allow an external Xenon or LED light source to synchronize with the sensor exposure time.

Figure 6: Sensor Core Block Diagram**Pixel Array**

The sensor core uses a Bayer color pattern, as shown in Figure 7. The even-numbered rows contain green and red pixels; odd-numbered rows contain blue and green pixels. Even-numbered columns contain green and blue pixels; odd-numbered columns contain red and green pixels.

Figure 7: Pixel Color Pattern Detail (Top Right Corner)

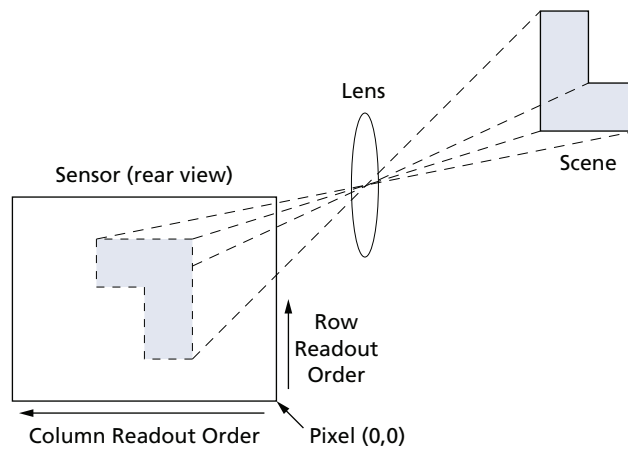


Default Readout Order

By convention, the sensor core pixel array is shown with pixel (0,0) in the top right corner. This reflects the actual layout of the array on the die. When the sensor is operating in a system, the active surface of the sensor faces the scene as shown in Figure 8.

When the image is read out of the sensor, it is read one row at a time, with the rows and columns sequenced. By convention, data from the sensor is shown with the first pixel read out in the case of the sensor core in the top left corner.

Figure 8: Imaging a Scene



Analog Processing

Analog Readout Channel

The sensor core features an analog readout channel, as shown in Figure 6 on page 14. The readout channel consists of a gain stage, a sample-and-hold stage with black level calibration capability, and a 10-bit ADC.

Gain Options

The MT9D113 provides per-color gain control as well as the option of global gain control. The per-color and global gain control can be used interchangeably. A WRITE to a global gain register is aliased as a WRITE of the same data to the four associated color-dependent gain registers.

Integer digital gains in the range 0–7 can be programmed. A digital gain of 0 sets all pixel values to 0 (the pixel data will simply represent the value applied by the pedestal block). Gain settings are updated in every frame by the MCU auto functions such as AWB, AE, and FD. To make manual adjustments to gain settings, the MCU automatic exposure and automatic white balance adjustment features must be disabled.



Integration Time

The integration time (exposure) of the MT9D113 is controlled by variables. While coarse integration time controls the integration duration in terms of row times, fine integration time allows for sub-row times accuracy in terms of pixel clocks. Integration time is updated in every frame by the MCU auto feature; to make manual adjustments to integration time, the MCU auto features must be disabled.

Because of the basic operation of the ERS, it is not advisable to set an integration time that is greater than the frame time.

On the MT9D113, it is not necessary to reprogram the frame time in order to make longer integration times available, as the frame time adjusts automatically. However, long integration times increase the likelihood of image degradation due to increased accumulation of dark current.

If the integration time is changed while FV is asserted for frame n , the first frame output using the new integration time is frame $(n + 2)$. The sequence is as follows:

1. During frame n , the new integration time is held in the pending register.
2. At the start of frame $(n + 1)$, the new integration time is transferred to the live register. Integration for each row of frame $(n + 1)$ has been completed using the old integration time.
3. The earliest time that a row can start integrating using the new integration time is immediately after that row has been read for frame $(n + 1)$.
4. When frame $(n + 2)$ is read out, it will have been integrated using the new integration time.

If the integration time is changed on successive frames, each value written will be applied for a single frame; the latency between writing a value and it affecting the frame readout remains at two frames.

When the integration time and the gain are changed at the same time, the gain update is held off by one frame so that the first frame output with the new integration time also has the new gain applied.

PLL-Generated Master Clock

The PLL can generate a master clock signal whose frequency is up to 85 MHz (input clock from 6 MHz through 54 MHz).

PLL Setup

Because the input clock frequency is unknown, the sensor starts up with the PLL disabled. The PLL takes time to power up. During this time, the behavior of its output clock signal is not guaranteed. The PLL output frequency is determined by two constants, M and N, and the input clock frequency.

$$VCO = \frac{Fin \times 2 \times M}{N + 1}$$

$$PLL \text{ output frequency} = \frac{VCO}{P1 + 1} \quad (EQ 1)$$



Digital Processing

Readout Options

The sensor core supports different readout options to modify the image before it is sent to the IFP. The readout can be limited to a specific window of the original pixel array.

For preview modes, the sensor core supports both skipping and pixel averaging in x and y directions.

By changing the readout direction the image can be flipped in the vertical direction and/or mirrored in the horizontal direction.

Window Size

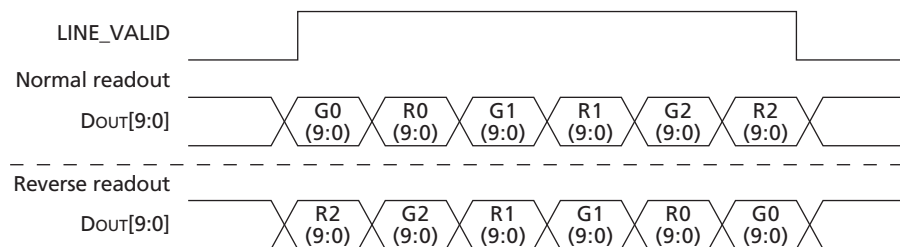
The image output size is set with registers `x_addr_start`, `x_addr_end`, `y_addr_start`, and `y_addr_end`. The edge pixels in the 1600 x 1200 array are present to avoid edge defects and should not be included in the visible window. Binning will change the image output size.

Readout Modes

Horizontal Mirror

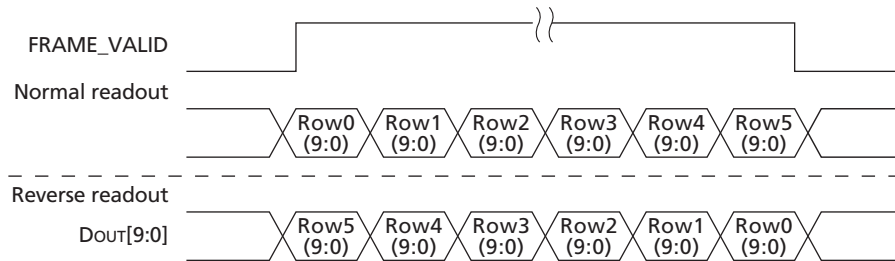
When the sensor is configured to mirror the image horizontally, the order of pixel readout within a row is reversed, so that readout starts from `x_addr_end` and ends at `x_addr_start`. Figure 9 shows a sequence of 6 pixels being read out with normal readout and reverse readout. The SOC corrects for this change in sensor core output.

Figure 9: 6 Pixels in Normal and Column Mirror Readout Modes



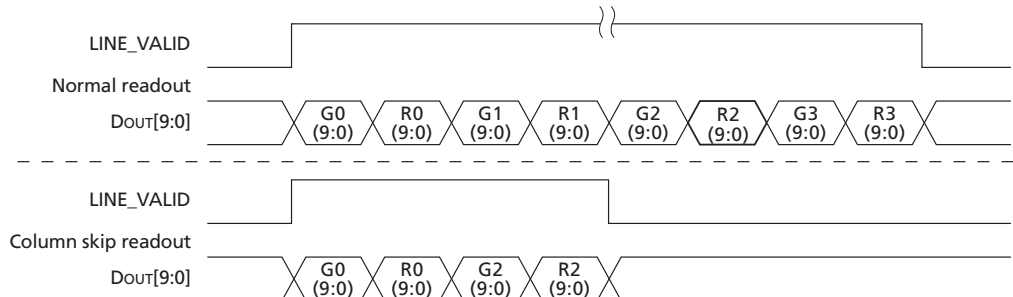
Vertical Flip

When the sensor is configured to flip the image vertically, the order in which pixel rows are read out is reversed, so that row readout starts from `y_addr_end` and ends at `y_addr_start`. Figure 10 shows a sequence of 6 rows being read out with normal readout and reverse readout. The SOC corrects for this change in sensor core output.

**Figure 10: 6 Rows in Normal and Row Mirror Readout Modes**

Column and Row Skip

The sensor core supports subsampling. Subsampling reduces the amount of data processed by the analog signal chain in the sensor and thereby allows the frame rate to be increased. This reduces the amount of row and column data processed and is equivalent to the skip2 readout mode provided by earlier Aptina image sensors. Set the proper image output and crop sizes before enabling subsampling.

Figure 11: 8 Pixels in Normal and Column Skip 2X Readout Modes

Pixel Readouts

Figures 12 through Figure 15 on page 20 show a sequence of data being read out with no skipping, with $x_odd_inc = 3$ and $y_odd_inc = 1$, with $x_odd_inc = 1$ and $y_odd_inc = 3$, and with $x_odd_inc = 3$ and $y_odd_inc = 3$.

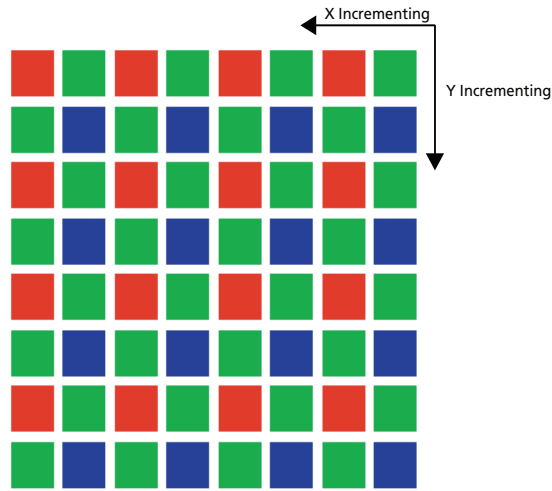
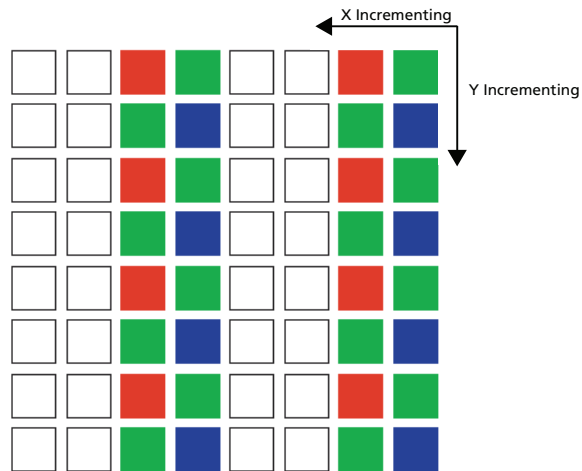
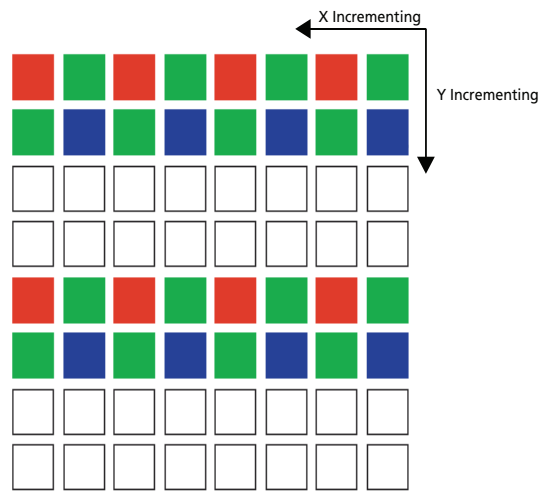
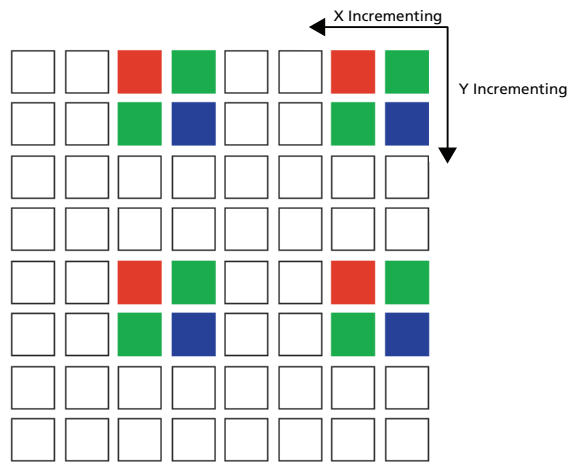
**Figure 12: Pixel Readout (no skipping)****Figure 13: Pixel Readout ($x_odd_inc = 3, y_odd_inc = 1$)**

Figure 14: Pixel Readout ($x_odd_inc = 1, y_odd_inc = 3$)Figure 15: Pixel Readout ($x_odd_inc = 3, y_odd_inc = 3$)

Programming Restrictions when Skipping

When skipping is enabled as a viewfinder mode, and the sensor is switched back and forth between full resolution and skipping, keep *line_length_pck* constant. This allows the same integration times to be used in each mode.

When subsampling is enabled, it may be necessary to adjust the *x_addr_end* and *y_addr_end* settings. The values for these registers must correspond with rows/ columns that form part of the subsampling sequence. To make the adjustment, use the following rules:

$$remainder = (addr_end - addr_start + 1) \text{ AND } 4 \quad (EQ\ 2)$$

$$\text{if } (remainder == 0) \text{ } addr_end = addr_end - 2 \quad (EQ\ 3)$$



Table 6 shows the row address sequencing for normal and subsampled (with $y_odd_inc = 3$) readout. The same sequencing applies to column addresses for subsampled readout. There are two possible subsampling sequences (because the subsampling sequence only reads half of the rows and columns) depending upon the alignment of the start address.

Table 6: Row Address Sequencing (Sampling)

Normal	Subsampled Sequence 1	Subsampled Sequence 2
0	0	No data
1	1	No data
2	No data	2
3	No data	3
4	4	No data
5	5	No data
6	No data	6
7	No data	7

Binning

The MT9D113 sensor core supports 2 x 1 and 2 x 2 analog binning (column binning, also called x-binning and row/column binning, also called xy-binning). Binning has many of the same characteristics as subsampling but because it gathers image data from all pixels in the active window (rather than a subset of them), it achieves superior image quality and avoids the aliasing artifacts that can be a characteristic side effect of subsampling.

Binning is enabled by selecting the appropriate subsampling settings ($x_odd_inc = 3$ and $y_odd_inc = 1$ for x-binning, $x_odd_inc = 3$ and $y_odd_inc = 3$ for xy-binning) and setting the appropriate binning bit in read_mode register. As for subsampling, x_addr_end and y_addr_end may require adjustment when binning is enabled.

The effect of the different subsampling settings is shown in Figure 16 and in Figure 17 on page 22.

Figure 16: Pixel Readout ($x_odd_inc = 3$, $y_odd_inc = 1$, $x_bin = 1$)

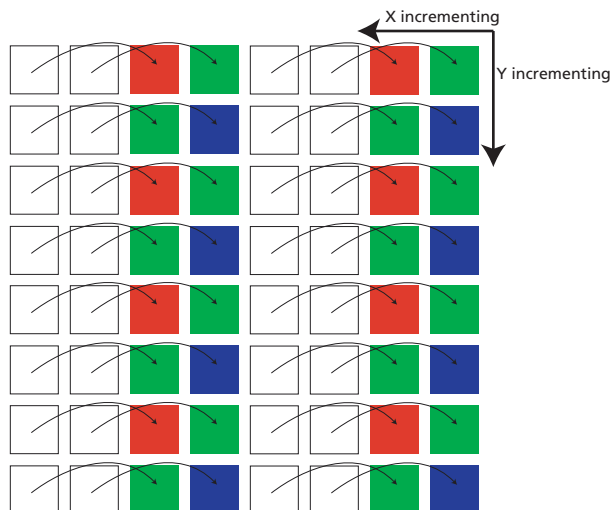
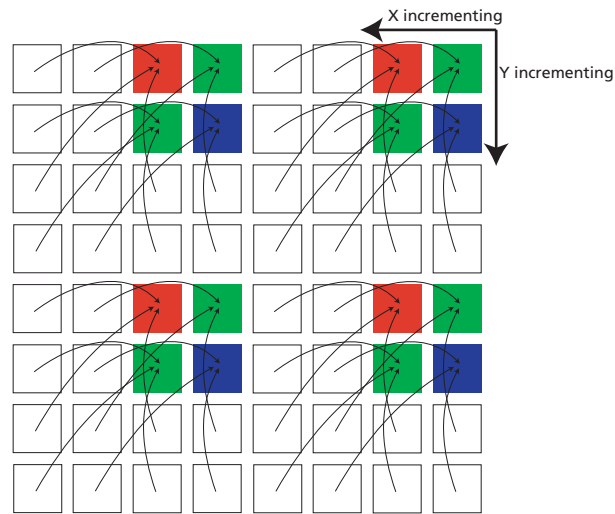


Figure 17: Pixel Readout ($x_odd_inc = 3$, $y_odd_inc = 3$, $x_ybin = 1$)

Binning Limitations

Binning requires a different sequencing of the pixel array and imposes different timing limits on the operation of the sensor. In particular, xy-binning requires two READ operations from the pixel array for each line of output data, which has the effect of increasing the minimum line blanking time.

As a result, when xy-binning is enabled, some of the programming limits declared in the parameter limit registers are no longer valid. In addition, the default values for some of the manufacturer-specific registers need to be reprogrammed. None of these adjustments are required for x-binning. The sensor must be taken out of streaming mode before switching between binned and non-binned operation. The row addresses for various binning modes are shown in Table 7.

Table 7: Row Address Sequencing (Binning)

Normal	Binning Sequence 1	Binning Sequence 2
0	0, 2	No data
1	1, 3	No data
2	No data	2, 4
3	No data	3, 5
4	4, 6	No data
5	5, 7	No data
6	No data	6, 8
7	No data	7, 9

Raw Data Format

The sensor core image data is read out in a progressive scan. Valid image data is surrounded by horizontal blanking and vertical blanking, as shown in Figure 18. The amount of horizontal blanking and vertical blanking is programmable. LV is HIGH during the shaded region of the figure.



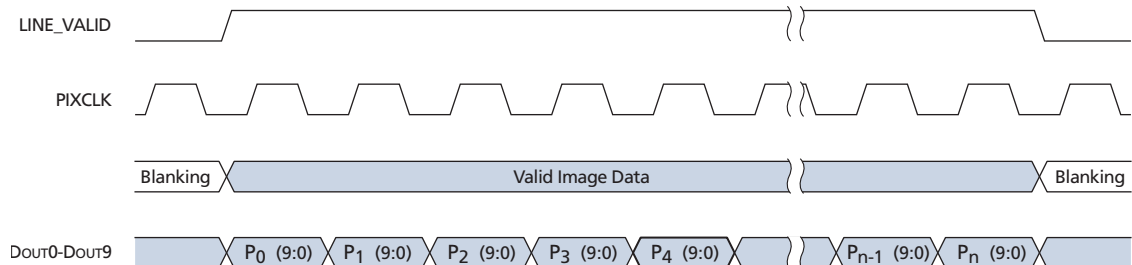
Figure 18: Valid Image Data

$P_{0,0} P_{0,1} P_{0,2} \dots P_{0,n-1} P_{0,n}$ $P_{1,0} P_{1,1} P_{1,2} \dots P_{1,n-1} P_{1,n}$	$00\ 00\ 00 \dots 00\ 00\ 00$ $00\ 00\ 00 \dots 00\ 00\ 00$
VALID IMAGE	HORIZONTAL BLANKING
$P_{m-1,0} P_{m-1,1} \dots P_{m-1,n-1} P_{m-1,n}$ $P_{m,0} P_{m,1} \dots P_{m,n-1} P_{m,n}$	$00\ 00\ 00 \dots 00\ 00\ 00$ $00\ 00\ 00 \dots 00\ 00\ 00$
$00\ 00\ 00 \dots 00\ 00\ 00$ $00\ 00\ 00 \dots 00\ 00\ 00$	$00\ 00\ 00 \dots 00\ 00\ 00$ $00\ 00\ 00 \dots 00\ 00\ 00$
VERTICAL BLANKING	VERTICAL/HORIZONTAL BLANKING
$00\ 00\ 00 \dots 00\ 00\ 00$ $00\ 00\ 00 \dots 00\ 00\ 00$	$00\ 00\ 00 \dots 00\ 00\ 00$ $00\ 00\ 00 \dots 00\ 00\ 00$

Raw Data Timing

The sensor core output data is synchronized with the PIXCLK output. When LV is HIGH, one pixel's data is output on the 10-bit DOUT output bus every PIXCLK period. By default, the PIXCLK signal runs at the same frequency as the master clock, and its falling edges occur one-half of a master clock period after transitions on LV, FV, and DOUT[9:0] (see Figure 19). This allows PIXCLK to be used as a clock to sample the data. PIXCLK is continuously enabled, even during the blanking period.

Figure 19: Pixel Data Timing Example





Input Interface to Image Flow Processor

The input interface to the image flow processor (IFP) is the front-end of the IFP in which it will choose between the sensor core output or a test pattern generator output. During normal operation, a stream of raw Bayer image data from the sensor is continuously fed into the IFP. For testing purposes, the test generator output is selected. The generator provides a selection of test patterns sufficient for basic testing of the IFP. Test patterns (shown in Figure 20) are accessible by programming a register.

Figure 20: Available Test Patterns

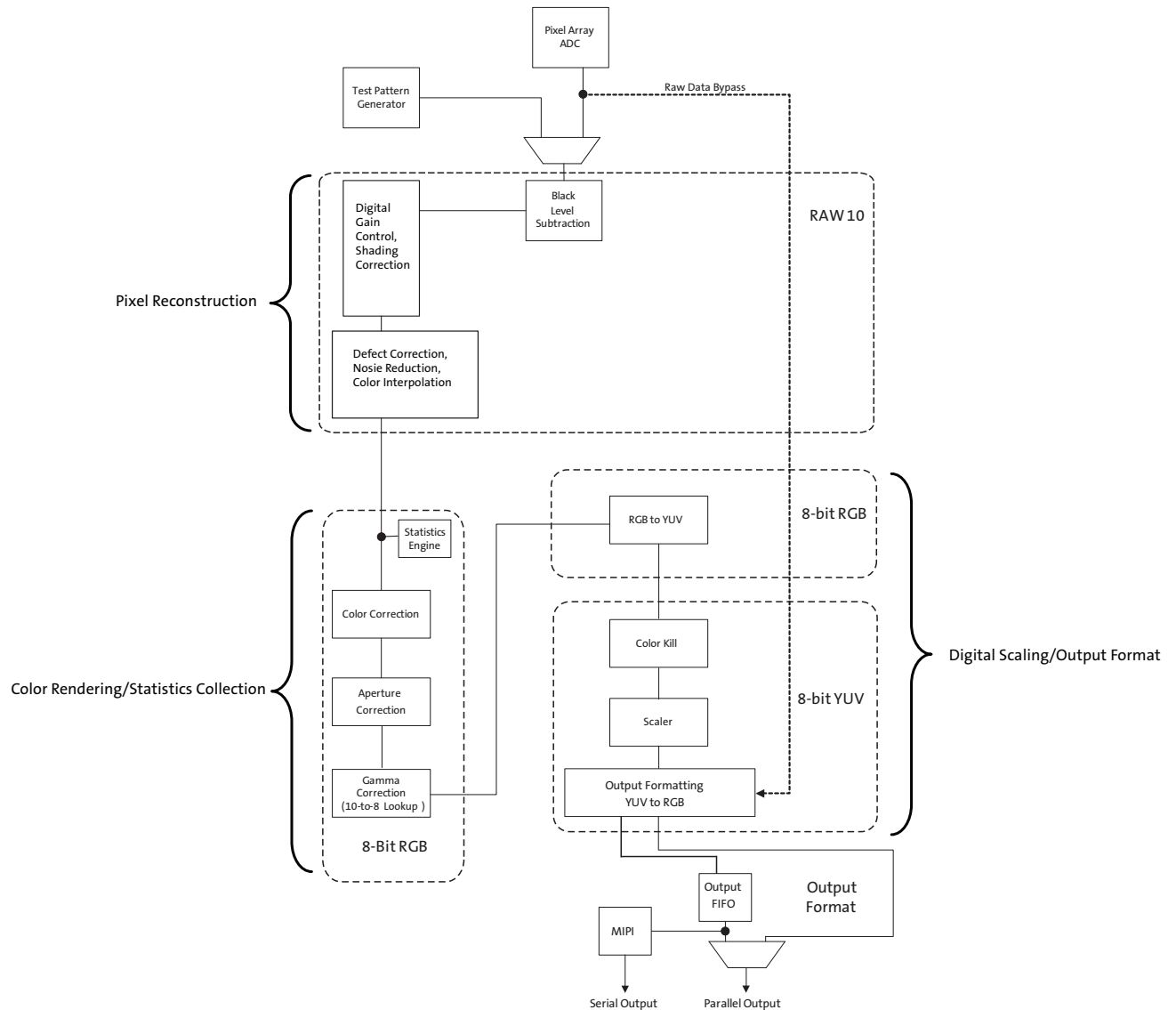
Test Pattern	Register Value	Example
Flat Field	0x100[3:2] = 1	
Vertical Ramp	0x100[3:2] = 2	
Color Bar	0x100[3:2] = 3	
Vertical Stripes	0x100[3:2] = 4	
Pseudo-Random	0x100[3:2] = 5	
Horizontal Stripes	0x100[3:2] = 6	



Image Flow Processor

Most of the IFP functions can be controlled directly by the MCU; the external host will control it indirectly through the driver variables. The IFP processing can be broken into three different phases: pixel reconstruction, color rendering/statistics collection, and digital scaling/output format. Figure 21 shows a simplified IFP block diagram and the operating color space at each processing phase.

Figure 21: IFP Block Diagram





Pixel Reconstruction (Lens Shading Correction)

First Black Level Subtraction and Digital Gain

Image stream processing starts with black level subtraction and multiplication of all pixel values by a programmable digital gain. Both operations can be independently set to separate values for each color channel (R, Gr, Gb, B). Independent color channel digital gain can be adjusted with registers. Independent color channel black level adjustments can also be made. If the black level subtraction produces a negative result for a particular pixel, the value of this pixel is set to “0.”

Shading Correction (SC)

Lenses tend to produce images whose brightness is significantly attenuated near the edges. There are also other factors causing fixed pattern signal gradients in images captured by image sensors. The cumulative result of all these factors is known as image shading. The MT9D113 has an embedded shading correction module that can be programmed to counter the shading effects on each individual R, Gb, Gr, and B color signal.

The Correction Function

Color-dependent solutions are calibrated using the sensor, lens system, and an image of an evenly illuminated, featureless grey calibration field. From the resulting image, the color correction functions can be derived.

The correction functions can then be applied to each pixel value to equalize the response across the image as follows:

$$P_{corrected}(row,col) = P_{sensor}(row,col) * f(row,col) \quad (EQ\ 4)$$

where P are the pixel values and f is the color dependent correction functions for each color channel.

Defect Correction

The IFP performs continuous defect correction that can mask pixel array defects such as high dark-current (hot) pixels and pixels that are darker or brighter than their neighbors due to photoresponse nonuniformity. The module is edge-aware with exposure that is based on configurable thresholds. The thresholds are changed continuously based on the brightness of the current scene.

Noise Reduction

Noise reduction can be enabled or disabled. Thresholds can be set through register settings.

Color Interpolation and Edge Detection

In the raw data stream fed by the sensor core to the IFP, each pixel is represented by a 10-bit integer number, which can be considered proportional to the pixel's response to a one-color light stimulus, red, green, or blue, depending on the pixel's position under the color filter array. Initial data processing steps—up to and including the defect correction—preserve the one-color-per-pixel nature of the data stream. After the defect correction, the data stream must be converted to a three-colors-per-pixel stream appropriate for standard color processing. The conversion is done by an edge-sensitive color inter-



polation module. The module pads the incomplete color information available for each pixel with information extracted from an appropriate set of neighboring pixels. The algorithm used to select this set and extract the information seeks the best compromise between preserving edges and filtering out high frequency noise in flat field areas. The edge threshold can be set through register settings.

Aperture Correction

To increase image sharpness, a programmable 1D or 2D aperture correction (sharpening filter) is applied to color-corrected image data. The gain and threshold for 1D and 2D correction can be defined through register settings.

Color Rendering/Statistics Collection (Color Correction)

Color Correction

To achieve good color fidelity of the IFP output, interpolated RGB values of all pixels are subjected to color correction. The IFP multiplies each vector of three pixel colors by a 3 x 3 color correction matrix. The three components of the resulting color vector are all sums of three 10-bit numbers. Since such sums can have up to 12 significant bits, the bit width of the image data stream is widened to 12 bits per color (36 bits per pixel). The color correction matrix can be programmed by the user or automatically selected by the auto white balance (AWB) algorithm implemented in the IFP. Ideally, color correction should produce output colors that are independent of the spectral sensitivity and color crosstalk characteristics of the image sensor. The optimal values of the color correction matrix elements depend on those sensor characteristics and on the spectrum of light incident on the sensor. The color correction variables can be adjusted through register settings.

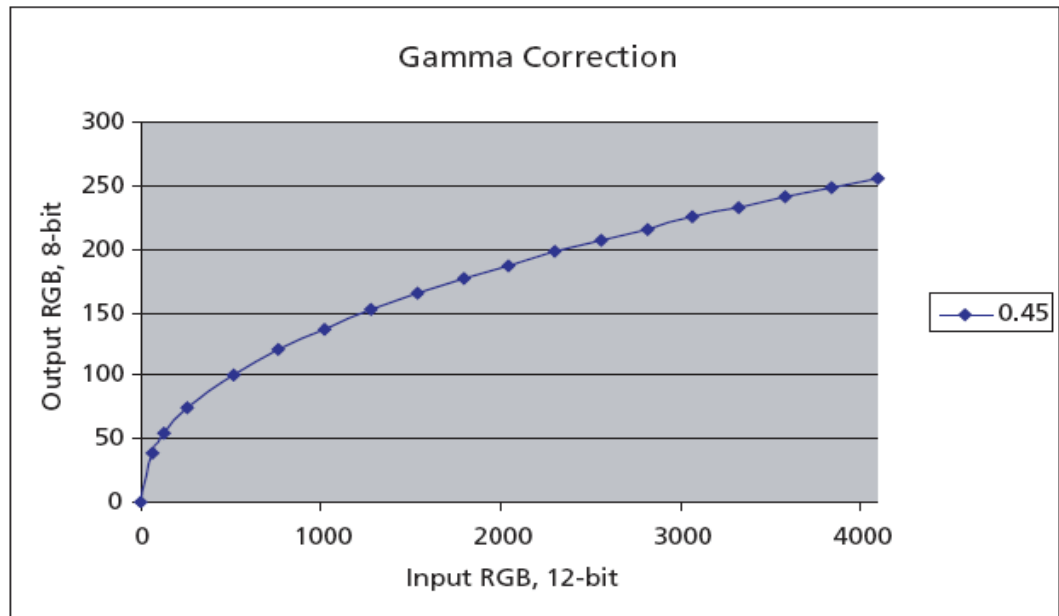
Image Cropping

By configuring the cropped and output windows to various sizes, different zooming levels (for example 4X, 2X, and 1X) can be achieved. The location of the cropped window is also configurable so that panning is also supported. A separate cropped window is defined for context A and context B. In both contexts, the height and width definitions for the output window must be equal to or smaller than the cropped image.

Gamma Correction

The gamma correction curve (as shown in Figure 22 on page 28) is implemented as a piecewise linear function with 19 knee points, taking 12-bit arguments and mapping them to 8-bit output. The abscissas of the knee points are fixed at 0, 64, 128, 256, 512, 768, 1024, 1280, 1536, 1792, 2048, 2304, 2560, 2816, 3072, 3328, 3584, 3840, and 4096. The 8-bit ordinates are programmable through IFP registers.

The MT9D113 IFP includes a block for gamma correction that can adjust its shape based on brightness to enhance the performance under certain lighting conditions. Two custom gamma correction tables may be uploaded, one corresponding to a brighter lighting condition, the other corresponding to a darker lighting condition. At power-up, the IFP loads the two tables with default values. The final gamma correction table used depends on the brightness of the scene and can take the form of either uploaded tables or an interpolated version of the two tables. A single (non-adjusting) table for all conditions can also be used.

**Figure 22: Gamma Correction Curve****Color Kill**

To remove high or low light color artifacts, a color kill circuit is included. It affects only pixels whose luminance exceeds a certain preprogrammed threshold. The U and V values of those pixels are attenuated proportionally to the difference between their luminance and the threshold.

Digital Scaling/Output Format**Special Effects**

Special effects like negative image, sepia, or black and white can be applied to the data stream at this point. These effects can be enabled and selected by registers.

RGB to YUV Conversion

For further processing, the data is converted from RGB color space to YUV color space.

YUV Color Filter

As an optional processing step, noise suppression by one-dimensional low-pass filtering of Y and/or UV signals is possible. A 3- or 5-tap filter can be selected for each signal.

Image Scaling

To ensure that the size of images output by the MT9D113 can be tailored to the needs of all users, the IFP includes a scaler module. When enabled, this module performs rescaling of incoming images—shrinks them to arbitrarily selected width and height without reducing the field of view and without discarding any pixel values.

The scaler performs pixel binning—divides each input image into rectangular bins corresponding to individual pixels of the desired output image, averages pixel values in these bins, and assembles the output image from the bin averages. Pixels lying on bin



boundaries contribute to more than one bin average; their values are added to bin-wide sums of pixel values with fractional weights. The entire procedure preserves all image information that can be included in the downsized output image and filters out high frequency features that could cause aliasing.

The image cropping and scaler module can be used together to implement a digital zoom and pan. If the scaler is programmed to output images smaller than images coming from the sensor core, zoom effect can be produced by cropping the latter from their maximum size down to the size of the output images. The ratio of these two sizes determines the maximum attainable zoom factor. For example, a 1600 x 1200 image rendered on a 250 x 150 display can be zoomed up to eight times, since $1280/160 = 1024/128 = 8$. A panning effect can be achieved by fixing the size of the cropping window and moving it around the pixel array.

Due to the loss of sharpness caused by pixel binning during image scaling, 1D aperture correction may be applied to increase image sharpness.

YUV-to-RGB/YUV Conversion and Output Formatting

The YUV data stream emerging from the scaling module can either exit the color pipeline as-is or be converted before exit to an alternative YUV or RGB data format.

Color Conversion Formulas

Y'U'V'

This conversion is ITU-R BT .601 scaled to make YUV range from 0 through 255. This setting is recommended for JPEG encoding and is the most popular.

$$Y' = 0.299 \times R' + 0.587 \times G' + 0.114 \times B' \quad (\text{EQ 5})$$

$$U' = 0.564 \times (B' - Y') + 128 \quad (\text{EQ 6})$$

$$V' = 0.713 \times (R' - Y') + 128 \quad (\text{EQ 7})$$

There is an option where 128 is not added to U'V'.

Y'Cb'Cr' Using sRGB Formulas

The MT9D113 implements the sRGB standard. This option provides YCbCr coefficients for a correct 4:2:2 transmission.

Note: $16 < Y' < 235$; $16 < Cb < 240$; $16 < Cr < 240$; and $0 \leq RGB \leq 255$

$$Y' = (0.2126 \times R' + 0.7152 \times G' + 0.0722 \times B') \times (219/256) + 16 \quad (\text{EQ 8})$$

$$Cb' = 0.5389 \times (B' - Y') \times (224/256) + 128 \quad (\text{EQ 9})$$

$$Cr' = 0.635 \times (R' - Y') \times (224/256) + 128 \quad (\text{EQ 10})$$

Y'U'V' Using sRGB Formulas

Similar to the previous set of formulas, but has YUV spanning a range of 0 through 255.

$$Y' = 0.2126 \times R' + 0.7152 \times G' + 0.0722 \times B' \quad (\text{EQ 11})$$

$$U' = 0.5389 \times (B' - Y') + 128 = -0.1146 \times R - 0.3854 \times G + 0.5 \times B \quad (\text{EQ 12})$$

$$V' = 0.635 \times (R' - Y') + 128 = 0.5 \times R - 0.4542 \times G - 0.0458 \times B \quad (\text{EQ 13})$$

There is an option to disable adding 128 to U'V'. The reverse transform is as follows:



$$R' = Y + 1.5748 \times V \quad (\text{EQ 14})$$

$$G' = Y - 0.1873 \times (U - 128) - 0.4681 \times (V - 128) \quad (\text{EQ 15})$$

$$B' = Y + 1.8556 \times (U - 128) \quad (\text{EQ 16})$$

Output Interface from IFP

This interface contains parallel image bus, MIPI serial bus interfaces, and walking 1s connectivity test generator.

Parallel and MIPI Output

The user can select to use either the serial MIPI output or the 10-bit parallel output to transmit the data. Only one of the output modes can be used at any time.

The parallel output can be used with an output FIFO whose memory is shared with the MIPI output FIFO to retain a constant pixel output clock independent from the scaling factor.

When scaling the image or skipping lines, the data would be generated in bursts and the pixel clock would turn on and off in intervals, which might lead to EMI problems. The output FIFO will group all active pixel data together so the pixel clock can be run at a constant speed.

The MIPI output transmitter implements a serial differential sub-LVDS transmitter capable of up to 640 Mb/s. It supports multiple formats, error checking, and custom short packets.

Table 8: Data Formats Supported by MIPI Interface

Data Format	Data Type
YUV 422 8-bit	0x1E
565RGB	0x22
555RGB	0x21
444RGB	0x20
RAW8	0x2A
RAW10	0x2B

Notes: 1. Data will be packed as RAW8 if the data type specified does not match any of the above data types.

Output Format and Timing

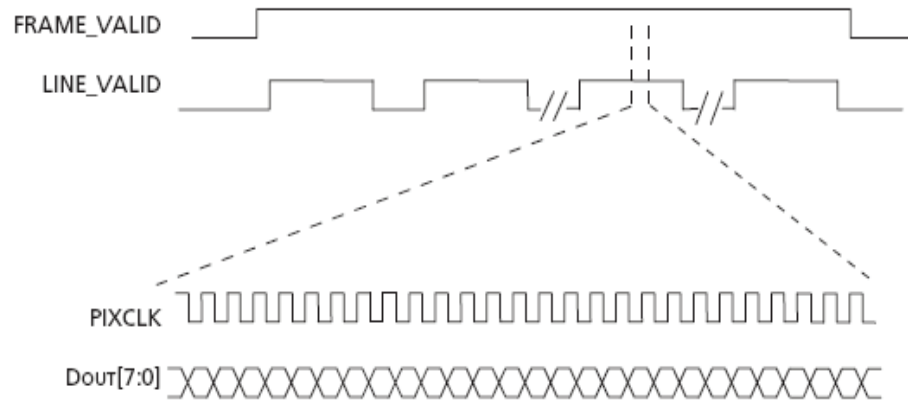
YUV/RGB Output

YUV or RGB data can be output either directly from the output formatting block or through a FIFO buffer with a capacity of 1024 bytes. This size is large enough to hold one-fourth of a scan line at full resolution. Buffering of data is a way to equalize the data output rate when image scaling is used. Scaling produces an intermittent data stream consisting of short high-rate bursts separated by idle periods. High pixel clock frequency during bursts may be undesirable due to EMI concerns.

Figure 23 depicts the output timing of a YUV/RGB scan line when a scaled data stream is equalized by buffering or when no scaling takes place. The pixel clock frequency remains constant during each LV high period. Scaled data is output at a lower frequency than full size frames, which helps to reduce EMI.



Figure 23: Timing of Full Frame Data or Scaled Data Passing Through the FIFO



YUV/RGB Data Ordering

The MT9D113 supports swapping YCrCb mode, as illustrated in Table 9.

Table 9: YCrCb Output Data Ordering

Mode	Data Sequence			
Default (no swap)	Cb _i	Y _i	Cr _i	Y _{i+1}
Swapped CrCb	Cr _i	Y _i	Cb _i	Y _{i+1}
Swapped YC	Y _i	Cb _i	Y _{i+1}	Cr _i
Swapped CrCb, YC	Y _i	Cr _i	Y _{i+1}	Cb _i

The RGB output data ordering in default mode is shown in Table 10. The odd and even bytes are swapped when luma/chroma swap is enabled. R and B channels are bitwise swapped when chroma swap is enabled.

Table 10: RGB Ordering in Default Mode

Mode (Swap Disabled)	Byte	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀
565RGB	Odd	R ₇ R ₆ R ₅ R ₄ R ₃ G ₇ G ₆ G ₅
	Even	G ₄ G ₃ G ₂ B ₇ B ₆ B ₅ B ₄ B ₃
555RGB	Odd	0 R ₇ R ₆ R ₅ R ₄ R ₃ G ₇ G ₆
	Even	G ₄ G ₃ G ₂ B ₇ B ₆ B ₅ B ₄ B ₃
444xRGB	Odd	R ₇ R ₆ R ₅ R ₄ G ₇ G ₆ G ₅ G ₄
	Even	B ₇ B ₆ B ₅ B ₄ 0 0 0 0
x444RGB	Odd	0 0 0 0 R ₇ R ₆ R ₅ R ₄
	Even	G ₇ G ₆ G ₅ G ₄ B ₇ B ₆ B ₅ B ₄



10-Bit Bypass Output

Raw 10-bit Bayer data from the sensor core can be output in bypass mode in two ways:

1. Using eight data output signals (DOUT[7:0]) and GPIO[1:0]. The GPIO signals are the lowest two bits of data.
2. Using only eight signals (DOUT[7:0]) and a special 8 + 2 data format, shown in Table 11.

Table 11: 2-Byte RGB Format

Byte	Bits Used	Bit Sequence
Odd bytes	8 data bits	D ₉ D ₈ D ₇ D ₆ D ₅ D ₄ D ₃ D ₂
Even bytes	2 data bits + 6 unused bits	0 0 0 0 0 0 D ₁ D ₀

FIFO

During normal pipeline operation, the output data rate is determined by a number of factors: input image size, degree of scaling, and sensor operation mode. As these parameters change during normal sensor operation, output frequency changes. This output frequency may generate RF noise, interfering with the mobile device. By using an output FIFO to maintain a constant output clock frequency, noise is easily filtered out.

The FIFO accumulates data and after a certain number of bytes are stored, it will output them in a single burst, making sure that the data rate within the burst remains constant. This approach utilizes a free-running clock, thus making possible minimal RF interference.

Table 12: Parallel Image Bus Timing Parameters

Definition	Conditions	Min	Typ	Max	Unit
Output pixel clock (PIXCLK) freq.	PLL bypass	6	—	54	MHz
	PLL	40	60	85	MHz
Output pixel clock (PIXCLK) period	PLL bypass	166.7	—	18.5	ns
	PLL	25	16.7	11.7	ns

Figure 24: Output Interface Timing Waveforms

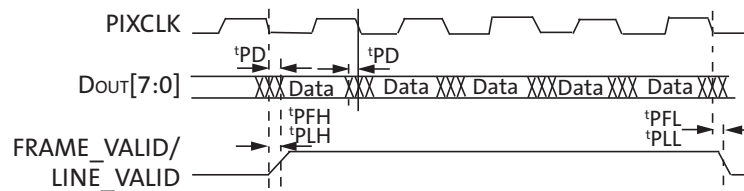


Table 13: Output Interface Timing

$f_{EXTCLK} = 6-54 \text{ MHz}$, $V_{DD} = V_{DDIO_TX} = V_{DD_IO} = 1.8V$; $V_{AA} = V_{AA_PIX} = V_{DD_PLL} = 2.8V$; $T_j = 25^\circ C$

Symbol	Definition	Condition	Min	Typ	Max	Unit
f_{PIXCLK}	PIXCLK frequency	PLL disabled	6	—	54	MHz
		PLL enabled	6	—	85	

**Table 13: Output Interface Timing**
 $f_{EXTCLK} = 6-54 \text{ MHz}$, $V_{DD} = V_{DDIO_TX} = V_{DD_IO} = 1.8V$; $V_{AA} = V_{AA_PIX} = V_{DD_PLL} = 2.8V$; $T_j = 25^\circ C$

Symbol	Definition	Condition	Min	Typ	Max	Unit
t_{PD}^1	PIXCLK to (DOUT, FV, LV) valid		-5	-	5	ns
t_{PFH}^1	PIXCLK to FV HIGH		-5	-	5	
t_{PFL}^1	PIXCLK to FV LOW		-5	-	5	
t_{PLH}^1	PIXCLK to LV HIGH		-5	-	5	
t_{PLL}^1	PIXCLK to LV LOW		-5	-	5	

Notes: 1. Valid for $C_{LOAD} < 20pF$ on PIXCLK, DOUT, LINE_VALID, and FRAME_VALID pads. Loads must be matched as closely as possible.



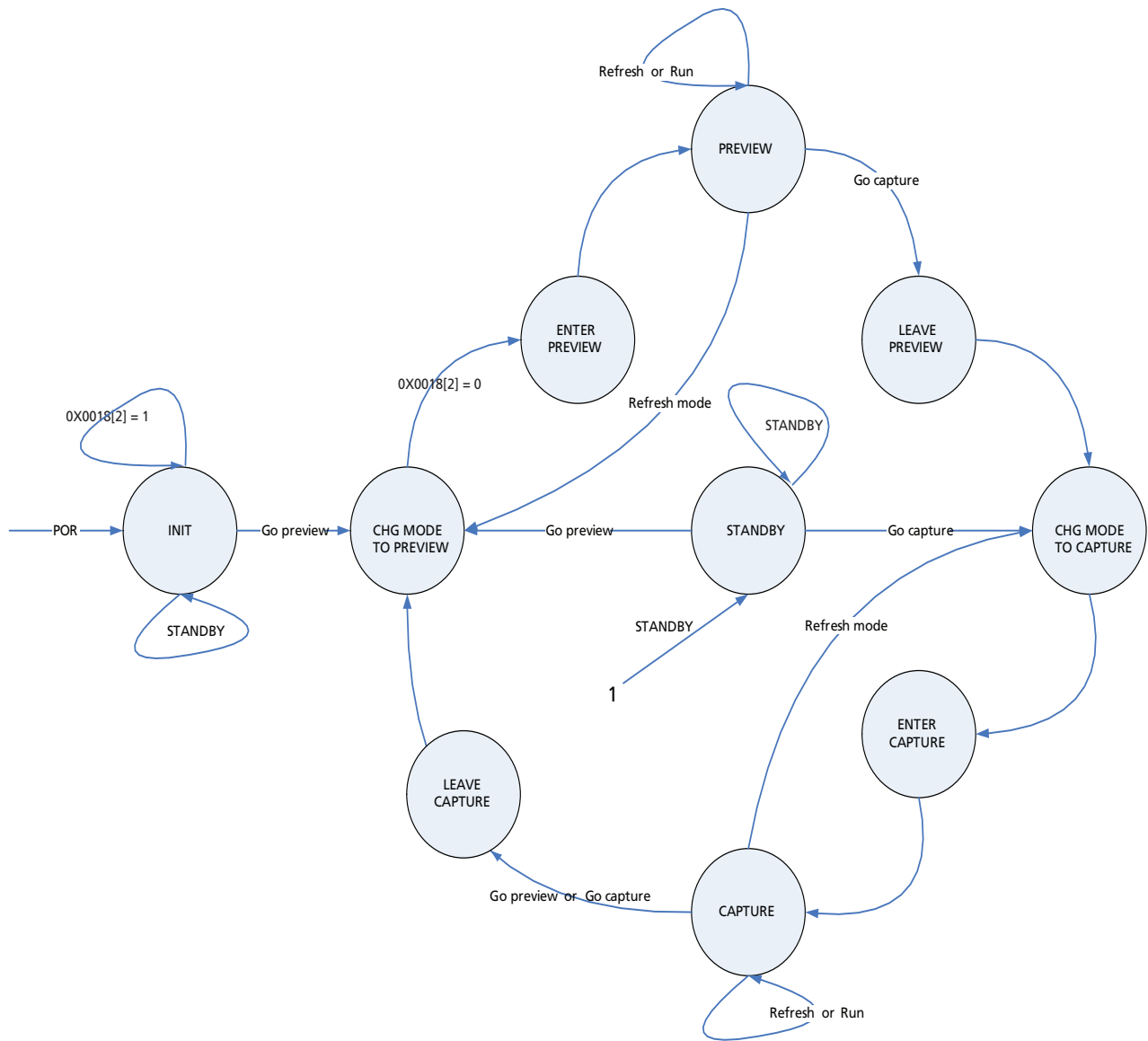
Control Functions

Sequencer: Camera Operating System

The sequencer is a finite state machine that controls the general operations of the camera and switching between operating modes. Camera operation is organized in states such as enter preview, preview, leave preview, enter capture, and capture. The sequencer carries out a number of commands such as run, go preview, go capture, refresh, and refresh mode. The current state of the sequencer is indicated in a variable *seqr.state*. To execute a command, the user has to set a particular command number in the variable *seqr.cmd*. The external host should monitor *seqr.state* to know when to change resolution or capture frames.

Each state has its configuration; therefore, the user should set up the state configuration to customize the camera configuration before executing the corresponding program such as GO TO CAPTURE. A typical camera operating scenario is:

1. Configure mode variables after hardware reset.
2. Configure preview mode.
3. Execute sensor REFRESH program.
4. Run in preview until shutter button is depressed.
5. Capture a frame.

**Figure 25: Sequencer Finite State Machine**

Notes: 1. Any state except INIT can transition to Standby state by either hard standby or soft standby.

Mode: Context Information

The mode driver reduces integration efforts by managing most aspects of switching the two contexts. It remembers vital register values for each image acquisition context and loads these values to the appropriate registers in the IFP upon context switching.

For the mode driver variables to take effect, the user changes the variable values in the mode driver (ID = 7). Upon the next mode change or sequencer's REFRESH command, these driver variable values will be loaded to the appropriate physical sensor core and IFP registers. The image processing will take the new values at the beginning of the next frame acquired.



MT9D113: 1/5-Inch SOC Digital Image Sensor Control Functions

To control the output image size, the user can modify the mode driver variables such as *output_width_A*, *output_height_A*, *output_width_B*, and *output_height_B*. The mode driver will automatically apply any appropriate downscaling filter to achieve this output image size as well as updating the watermark of the output FIFO. It is important to set up the sensor core to output an image equal to or larger than the crop window size, which in turn is equal to or larger than the desired output image size.



Camera Functions

Simple Rule-based Auto Exposure

The auto exposure (AE) algorithm performs automatic adjustments of the image brightness by controlling exposure time and analog gains of the sensor core as well as digital gains applied to the image.

Auto exposure is implemented by a firmware driver that analyzes image statistics collected by the exposure measurement engine, makes a decision, and programs the sensor core and color pipeline to achieve the desired exposure. The measurement engine subdivides the image into 16 windows organized as a 4 x 4 grid.

Two auto exposure algorithm modes are available:

1. Average brightness tracking (ABT)
2. Dynamic range tracking (DRT)

The average brightness tracking AE uses a constant average tracking algorithm where a target brightness value is compared to a current brightness value, and the gain and integration time are adjusted accordingly to meet the target requirement.

The DRT AE examines the data from the statistics engine and produces a corrected brightness target so that overexposed or underexposed scenes can be avoided. This also allows manual control of the image brightness. Both algorithms are available in preview and capture modes.

AE Driver

DRT exposure mode is activated during preview. This mode can also be enabled during video capture mode. The DRT exposure mode relies on the statistics engine to track speed and amplitude of the change of the overall luminance in the selected windows of the image.

Backlight compensation is achieved by weighting the luminance in the center of the image higher than the luminance on the periphery. Other algorithm features include the rejection of fast fluctuations in illumination (time averaging), control of speed of response, and control of the sensitivity to the small changes. While the default settings are adequate in most situations, the user can program target brightness, measurement window, and other parameters.

The driver calculates image brightness based on average luma values received from 16 programmable equal-size rectangular windows forming a 4 x 4 grid. In preview mode, 16 windows are combined in two segments: central and peripheral. The central segment includes four central windows. All remaining windows belong to the peripheral segment. Scene brightness is calculated as average luma in each segment taken with certain weights.

The driver changes AE parameters (integration time, gains, and so on) to drive brightness to the programmable target. The value of the single step approach to the target value can be controlled.

To avoid unwanted reaction of AE on small fluctuations of scene brightness or momentary scene changes, the AE driver uses a temporal filter for luma and a threshold around the AE luma target. The driver changes AE parameters only if the filtered luma is larger than the AE target step and pushes the luma beyond the threshold.



Evaluative Algorithm

A scene-evaluative AE algorithm is available for use in snapshot mode. The algorithm performs scene analysis and classification with respect to its brightness, contrast, and composure and then decides to increase, decrease, or keep the original exposure target. It makes the most difference for backlight and bright outdoor conditions.

Accelerated Settling During Overexposure

The AE speed is direction-dependent. Transitioning from oversaturation to target can take more time than transitioning from undersaturation. The AE driver has a mode that speeds up AE for overexposed scenes.

The AE driver counts the number of AE windows whose average brightness is equal to or greater than some value, 250 by default. For a scene having saturated regions, the average luma is underestimated due to signal clipping. The driver compensates underestimation by a factor that can be defined.

Exposure Control

To achieve the required amount of exposure, the AE driver adjusts the sensor integration time, gains, ADC reference, and IFP digital gains. In addition, a variable is available for the user to adjust the overall brightness of the scene.

Flicker Detection and Avoidance

Flicker occurs when the integration time is not an integer multiple of the period of the light intensity. The automatic flicker detection block does not compensate for the flicker, but rather avoids it by detecting the flicker frequency and adjusting the integration time.

To reject flicker, integration time is typically adjusted in increments of steps. The incremental step specifies the duration in row times equal to one flicker period. Thus, flicker is rejected if integration time is kept to an integral factor of the flicker period.

For integration times below the light intensity period (10ms for 50Hz environment), flicker cannot be avoided.

Flicker shows up in the image as horizontal bars rolling up or down. The MCU looks for these rolling bars using a thin horizontal window, which outputs luma average and is applied to 48 points in the upper half of the image. The MCU repeats the same sampling on the next frame; 48 samples from the previous frame are subtracted from corresponding samples from the current frame. Skipping more frames between subtraction can be set by a variable. The MCU then smooths the 48 sampling points, applies an amplitude threshold to avoid false detection, and looks at the resulting waveform. If flicker is present, the waveform should have a frequency within the search range. Assuming the flicker power is a sine wave, subtracting two frames results in:

$$\sin(wt) - \sin(wt + a) = 2 \times \sin(a/2) \times \cos(wt + (a/2)) \quad (\text{EQ 17})$$

which is a cosine wave of the original frequency.



Auto White Balance

The MT9D113 has a built-in AWB algorithm designed to compensate for the effects of changing spectra of the scene illumination on the quality of the color rendition. The algorithm consists of two major parts: a measurement engine performing statistical analysis of the image and a driver performing the selection of the optimal color correction matrix, digital gains, and sensor core analog gains. While default settings of these algorithms are adequate in most situations, the user can reprogram the base color correction matrices, place limits on color channel gains, and control the speed of both matrix and gain adjustments.

Flash

To take a snapshot, the user must send a command that changes the context from preview to snapshot. The typical sequence of events after this command is:

1. The camera may turn on its LED flash, if it has one and is required to use it. With the flash on, the camera exposure and white balance are automatically adjusted to the changed illumination of the scene.
2. The camera captures one or more frames of desired size. A camera equipped with a xenon flash strobes while capturing images. When the images are captured, the camera automatically returns to context A and resumes running in preview mode.

Note: This sequence of events can take up to 10 frames.

Histogram: Dark Level Adjustments, Low Light and Tonal Controls

The histogram driver continually works to reduce image flare and continually analyzes input image histogram and dynamically adjusts the black level. When flare is present (hence, the histogram does not contain dark tones), it causes the driver to subtract a higher black level, thus regaining the lost contrast. In certain situations, the scene may contain no dark tones without flare. The histogram driver cannot distinguish this situation and alters the black level just the same, causing the image to have more contrast, which looks acceptable in many situations.

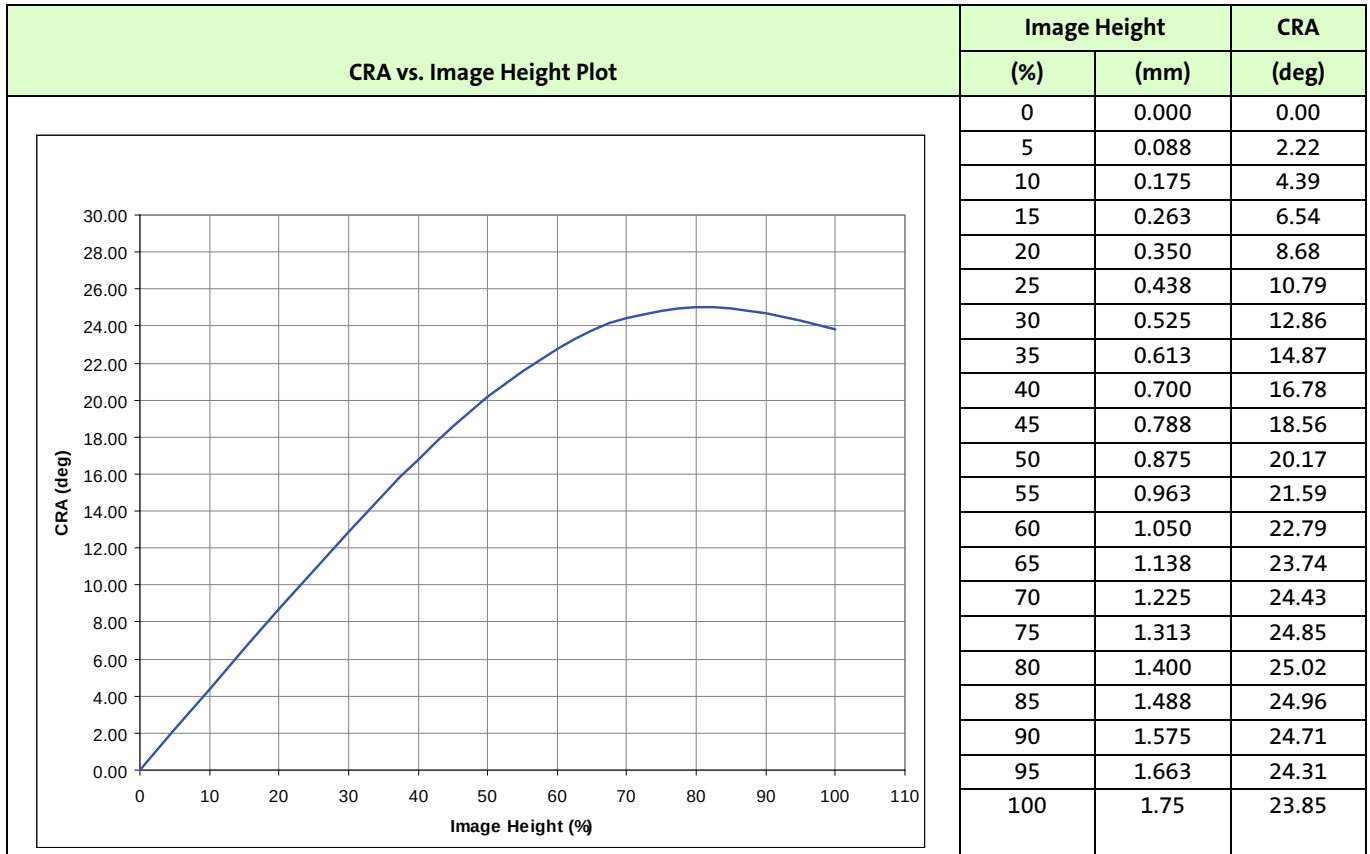
Besides black level adjustments and low light scene parameters, this driver also contains variables for the preloaded and custom gamma tables.



Optics

Figure 26 shows the CRA versus image height.

Figure 26: Chief Ray Angle (CRA) vs. Image Height





Power Modes

Power Application Sequence

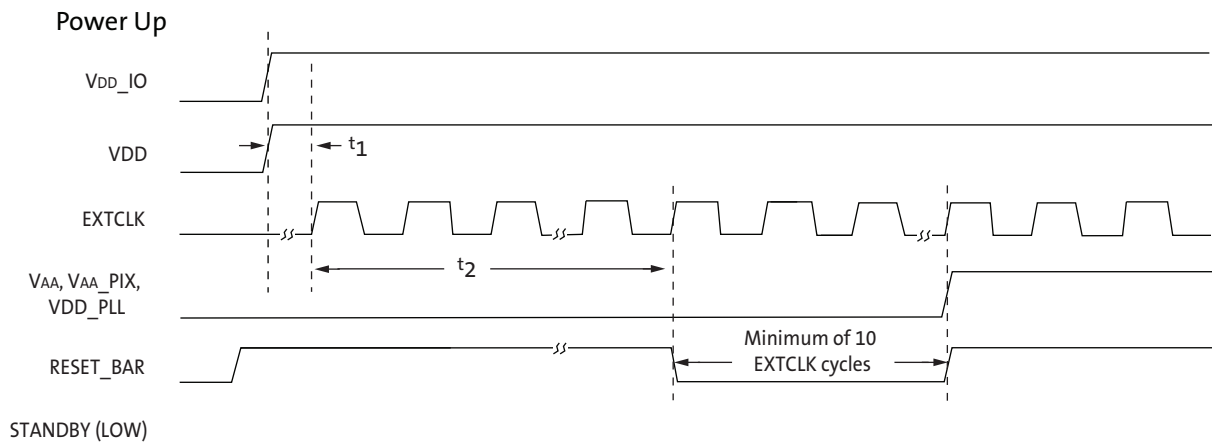
In order to maintain low power consumption during this process, the following sequence is recommended:

1. Ensure that STANDBY is de-asserted and RESET_BAR is asserted.
2. Apply IO and digital supplies (VDD_IO, VDD) to the image sensor.
3. Wait for the digital power supplies to be stable.
4. Enable EXTCLK.
5. Wait for EXTCLK to stabilize and de-assert RESET_BAR for a minimum of 10 EXTCLK cycles.
6. After asserting the RESET_BAR, apply analog supplies (VAA, VAA_PIX, and VDD_PLL) to the image sensor.
7. After 6000 EXTCLK cycles from the end of step 6, the image sensor will be in soft standby state.

In cases where the above mentioned procedure can not be followed, the following conditions would affect the sensor's power consumption during the power application sequence:

- When analog supplies are applied prior to the digital and IO supplies, high current consumption on the analog supplies may be present.

Figure 27: Power Application Sequence Timing



- Notes:
1. t_1 : The time for VDD_IO and VDD to stabilize is highly dependent on the system's overall design. Users of the MT9D113 should verify the DC voltage stabilization time on their system.
 2. t_2 : The time for EXTCLK to stabilize is also highly dependent on the system's overall design. Users of the MT9D113 should verify the EXTCLK's stabilization time on their system.



Power-On Reset

The sensor includes a power-on reset feature that initiates a reset upon power-up. Even though this feature is included on the device, it is advised that the user still manually assert a hard reset upon power-up.

The MT9D113's POR circuit generates internal reset only and it will not generate the external reset signal through RESET_BAR.

The POR circuit requires VDD ramp time to be less than 10 μ s. If the ramp time is longer than 10 μ s, the POR operation is not guaranteed and external reset must be used.

The POR circuit will generate an internal reset when VDD falls below 1.25V (typical) for 1 μ s (typical) period, as shown in Figure 28 and Table 14.

The POR circuit reset and external RESET_BAR signals are gated together to generate an internal reset for MT9D113.

Figure 28: Internal Power-On Reset

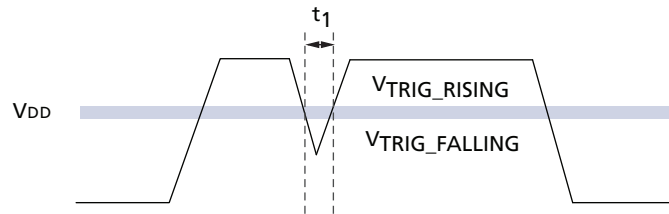


Table 14: POR Parameters

Symbol	Definition	Min	Typ	Max	Unit
t_1	Minimum VDD spike width below VTRIG_FALLING considered to be a reset	–	1	–	μ s
VTRIG_RISING	VDD rising trigger voltage	1.15	1.4	1.55	V
VTRIG_FALLING	VDD falling trigger voltage	1	1.25	1.45	V



Hard Reset

The MT9D113 enters the reset state when the external RESET_BAR signal is asserted LOW, as shown in Figure 29. All the output signals will be in High-Z state except the MIPI outputs, which will be driven LOW.

Figure 29: Hard Reset Operation

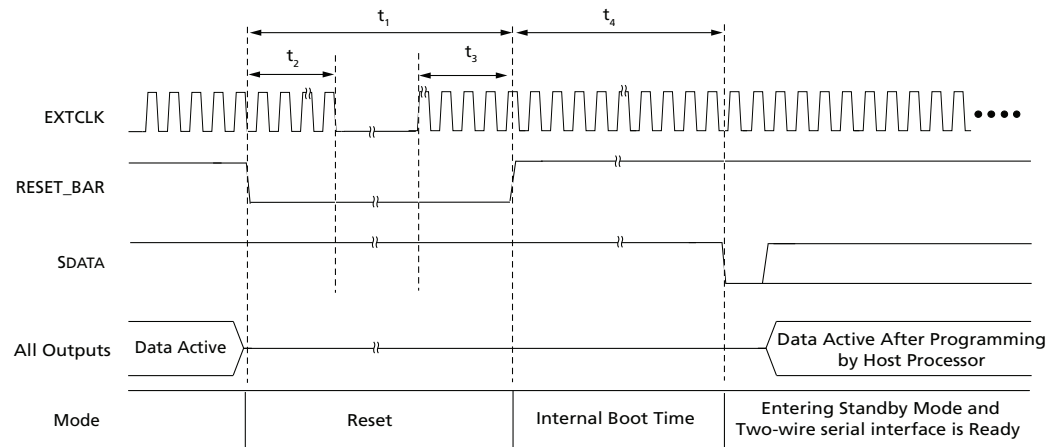


Table 15: Hard Reset

Symbol	Definition	Min	Typ	Max	Unit
t_1	RESET_BAR pulse width	100	—	—	EXTCLK cycles
t_2	Active EXTCLK required after RESET_BAR asserted	10	—	—	
t_3	Active EXTCLK required before RESET_BAR de-asserted	10	—	—	
t_4	Maximum internal boot time	—	—	6000	



Soft Reset

The host processor can reset the MT9D113 using the two-wire serial interface by writing to SYSCCTL 0x001A. Two types of soft reset are available. SYSCCTL 0x001A[0] is used to reset the MT9D113 which is similar to external RESET_BAR signal. SYSCCTL 0x001A[1] is used to reset the MIPI block within the MT9D113. Both bits are active HIGH. Even if MIPI is not used (parallel mode), reset both bits to "0" for normal operation.

1. Set SYSCCTL 0x001A[1:0] to 0x3 to initiate internal reset cycle.
2. Wait 6000 EXTCLK cycles.
3. Reset SYSCCTL 0x001A[1:0] to 0x0 for normal operation.

Figure 30: Soft Reset Operation

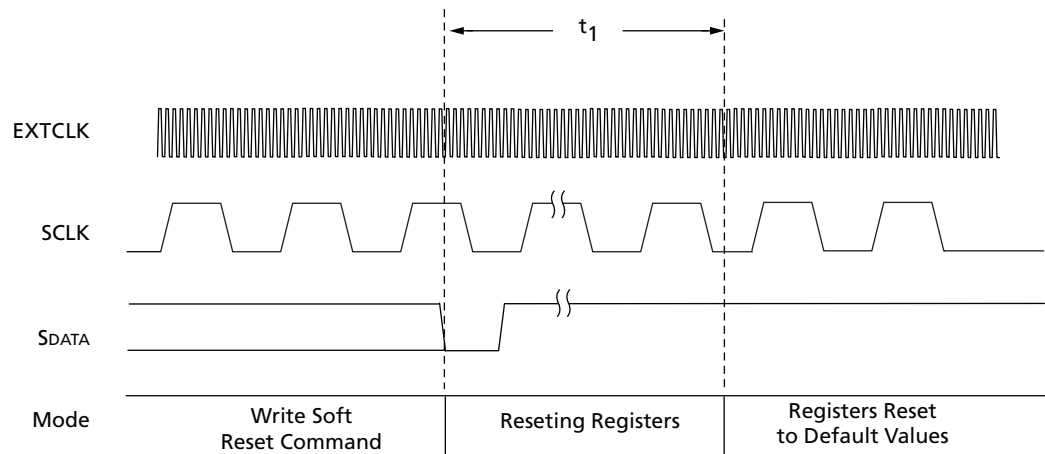


Table 16: Soft Reset Signal Timing

Symbol	Parameter	Min	Typ	Max	Unit
t_1	Maximum soft reset time	—	—	6000	EXTCLKs



Power Removal Sequence

In order to remove the power supplies from the image sensor, the following method is recommended:

1. Put the image sensor into either hard standby or soft standby mode as described in “Standby Modes” on page 46.
2. Remove all digital and analog supplies.

During step 2, the digital and analog supplies can be removed safely either one by one or at the same time, provided that step 1 is executed previously. In order to achieve a known power down state in the sensor, it is necessary to execute step 1 prior to removing any power supplies during the power removal process.



Standby Modes

The MT9D113 supports two different standby modes:

- Soft standby mode
- Hard standby mode

For each mode, entry can be overridden by programming the *standby_control* register. Patch codes, PLL configurations, parallel and MIPI IO selections are retained in all standby modes.

Soft Standby

Soft standby can be enabled by register access; it disables the sensor core and most of the digital logic. The two-wire serial interface is still active and the MT9D113 can be programmed through register commands. All register settings and RAM content will be preserved. Soft standby can be performed in any sequencer state.

Hard Standby

Hard standby mode uses the STANDBY signal to shut down digital power (VDD) and ensure the lowest power consumption. All the two-wire serial interface settings and firmware variables except patch codes, PLL configuration, parallel/MIPI IO and context selections will be lost in this mode. Starting up from this mode is equivalent to power up and current context selections. The two-wire serial interface will be inactive and the sensor must be started up by de-asserting the STANDBY signal.

Hard Standby Without Loss of Variable Data

Hard standby mode without the loss of variable data can also be achieved. This mode stores the variables and state of the sensor before entering standby (similar to soft standby). The power consumption is lower than that of soft standby, as internal clocks are turned off, and the two-wire serial interface will be inactive.

Since the hard standby without the loss of variable data mode is also activated by STANDBY, the *en_vdd_dis_soft* register needs to be programmed to indicate the selection of this mode before STANDBY is asserted. De-asserting STANDBY will cause the sensor to come out of the standby mode. This also causes the sensor to resume operation from the state before the STANDBY signal was asserted. By default, asserting the STANDBY signal causes the hard standby mode described above.



Hard Standby Mode

The MT9D113 can enter hard standby mode by using external STANDBY signal, as shown in Figure 31. EXTCLK can be stopped to reduce the power consumption during hard standby mode. Two-wire serial interface and IFP block shut down even when EXTCLK is running during hard standby mode.

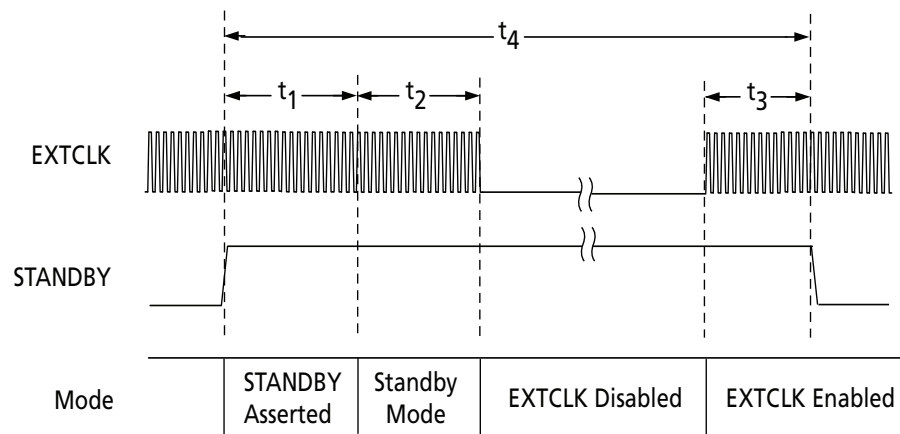
Entering Standby Mode

1. Assert STANDBY signal (HIGH).

Exiting Standby Mode

1. De-assert STANDBY signal (LOW).
2. Follow “Soft Reset” on page 44.

Figure 31: Hard Standby Mode Operation



Notes: 1. In hard standby mode, EXTCLK is automatically gated-off, and the two-wire serial interface is not active.

Table 17: Hard Standby Signal Timing

Symbol	Parameter	Min	Typ	Max	Unit
t_1	Standby entry complete (0x301A[4] = 1)	1 Frame + 20	–	1 Frame + 40	EXTCLKs
t_2	Active EXTCLK required after STANDBY asserted	10	–	–	EXTCLKs
t_3	Active EXTCLK required before STANDBY de-asserted	10	–	–	EXTCLKs
t_4	STANDBY pulse width	1 Frame + 120	–	1 Frame + 120	EXTCLKs



Soft Standby Mode

The MT9D113 can enter soft standby mode by writing to a SYSTL register through the two-wire serial interface, as shown in Figure 32. EXTCLK can be stopped to reduce the power consumption during soft standby mode. However, since two-wire serial interface requires EXTCLK to operate, Micron recommends that EXTCLK run continuously. If EXTCLK needs to be stopped, Micron recommends using hardware standby mode.

Entering Standby Mode

1. SYSTL 0x0018[3] must be set to "1" to activate standby mode. This will generate an interrupt to the MCU when SYSTL 0x0018[0] is set to "1."
2. Set SYSTL 0x0018[0] to "1" to initiate standby mode.
3. Check until SYSTL 0x0018[14] changes to "1" to indicate MT9D113 is in standby mode.

Exiting Standby Mode

1. Reset SYSTL 0x0018[0] to "0."
2. Check until SYSTL 0x0018[14] changes to "0" to indicate the MT9D113 is out of standby mode.

Figure 32: Soft Standby Mode Operation

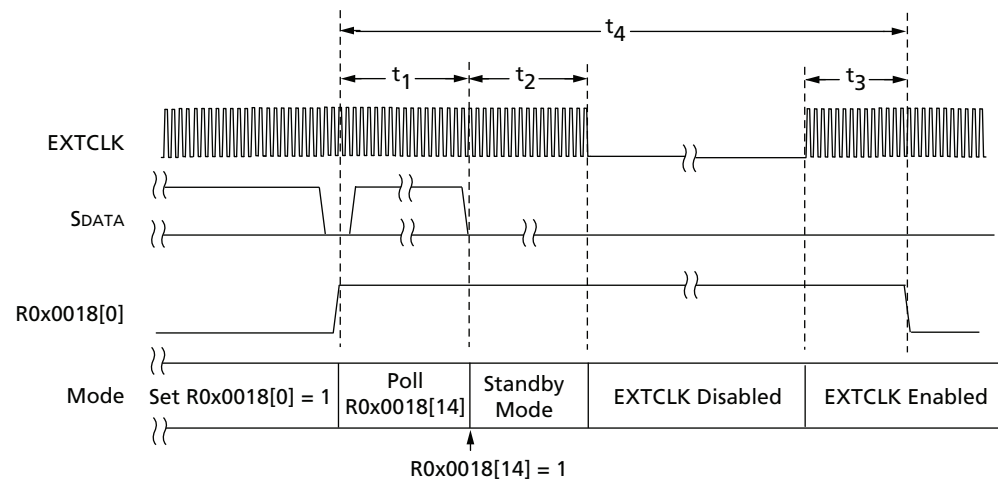


Table 18: Soft Standby Signal Timing

Symbol	Parameter	Min	Typ	Max	Unit
t_1	Standby entry complete (0x301A[4] = 1)	1 Frame + 20	—	1 Frame + 40	EXTCLKs
t_2	Active EXTCLK required after soft standby activates	10	—	—	EXTCLKs
t_3	Active EXTCLK required before soft standby de-activates	10	—	—	EXTCLKs
t_4	Minimum standby time	1 Frame + 100	—	1 Frame + 120	EXTCLKs



Other Features

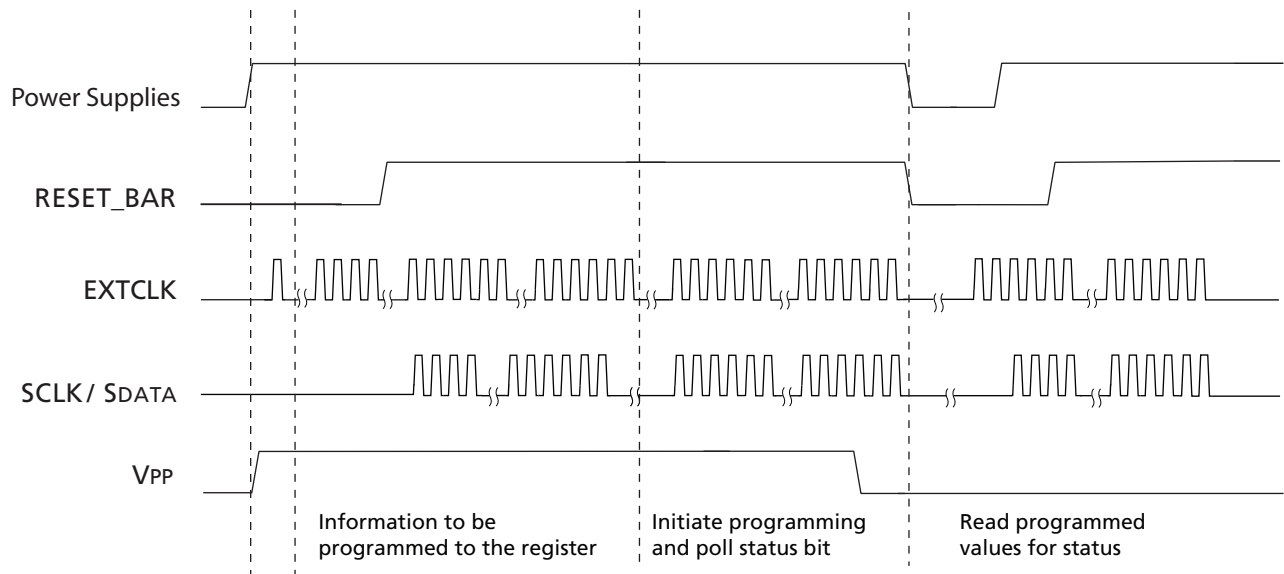
One-time Programmable (OTP) Memory

The MT9D113 contains two bytes of OTP memory, suitable for storing module identification that can be programmed during the module manufacturing process. Programming the OTP memory requires the use of a high voltage (8.0V) at the VPP pin. During normal operation, the VPP pin should be left floating. The OTP memory can be accessed through the two-wire serial interface.

Programming the OTP Memory

The programming of the OTP memory requires the sensor to be fully powered and remain in software standby with its clock input applied. The steps below describe the programming process:

1. Apply power to all the power rails of the sensor (VDD, VDD_IO, VAA, VAA_PIX, and so on).
2. Supply a high voltage (8.0V) to the VPP pin.
3. Provide the EXTCLK clock input.
4. Perform the proper reset sequence to the sensor.
5. Place the sensor in software standby (sensor default state upon power-up).
6. Write the information to be programmed to the program value register through the two-wire serial interface.
7. Write to the program value register to indicate the high voltage is applied at the VPP pin in order to start the programming process.
8. Poll the status bit from the program value register for the completion of the programming process.
9. Remove the high voltage at VPP pin.
10. Verify the data at the OTP memory by performing a normal read procedure.


Figure 33: Sequence of Signals for OTP Memory Operation


Reading the OTP Memory

Reading the OTP memory data requires the sensor to be fully powered and operational with its clock input applied. The data can be read through a register from the two-wire serial interface.



GPIO and Output Enable Controls

General Purpose I/Os

The four GPIOs of the MT9D113 can be configured in multiple ways. Each of the I/Os can be used as a simple input/output that can be programmed from the host. The status of the GPIO is read at power up and can be used as a module ID to identify different module suppliers. In addition, module ID can be stored in the OTP memory of the sensor. Information on the OTP memory can be found in “One-time Programmable (OTP) Memory” on page 49.

If 10-bit RAW output is required, GPIO[1:0] can be configured as bit 1 and bit 0 (the LSBs) of a 10-bit data bus.

GPIO[3] can be configured to output a flash pulse to trigger an external xenon or LED flash or a shutter pulse to control an external shutter.

GPIO[2] can also be configured as an input to be used as an OE_BAR signal for the data bus.

The general purpose inputs are enabled or disabled through register settings. Once enabled, all four inputs must be driven to valid logic levels by external signals. The state of the general purpose inputs can be read from a register.

Output Enable Control

When the parallel pixel data interface is enabled, its signals can be switched asynchronously between being driven and High-Z under signal or register control.



GPIO Control

The MT9D113 has four general purpose I/O (GPIO) signals and that can be individually programmed to perform various functions.

Overview of GPIO Signals

- Extension of two lower data bits during 10-bit output mode
- External flash control output
- External OE_BAR control input for parallel port signals

Table 19: GPIO Related Registers and Variables

Map	Address	Bits	Descriptions
SYSCTL	0x001A	[8]	GPIO[2] Enable. GPIO[2] can be configured as an Output Enable pad. 0: GPIO[2] does not function as OE. 1: GPIO[2] functions as OE.
SYSCTL	0x001E	[6:4]	Controls the output slew rate of GPIO signals. "111" programs the fastest slew rate. Refer to AC/DC specification in the datasheet for the numbers. Default value is "000" for slowest slew rate.
SYSCTL	0x0024	[3:0]	The state of GPIO signals during power on. Read-only. GPIO_0 is always operated as /OE during input modes. External pull-up and pull-down resistors are required.
GPIO	0x1070	[12:9]	GPIO data port. State of current GPIO signals can be read through this register.
GPIO	0x1074	[12:9]	GPIO Set command. When GPIO is configured as output, setting this register to "1" will write "1" to GPIO port. Use GPIO Clear command to clear.
GPIO	0x1076	[12:9]	GPIO Clear command. When GPIO is configured as output, setting this register to "1" will write "0" to GPIO port. Use GPIO Set command to set.
GPIO	0x1078	[12:9]	GPIO direction control for each GPIO signals. 0 = Output 1 = Input After power-on, GPIO ports are in input mode.



Electrical Specifications

Absolute Maximum Rating

Caution Table 20 shows stress ratings only, and functional operation of the device at these or any other conditions above those indicated in the product specification is not implied. Stresses above those listed may cause permanent damage to the device. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Table 20: Maximum Rating

Symbol	Parameter	Rating		Unit
		Min	Max	
VDD_MAX	Core digital voltage	−0.3	2.4	V
VDD_IO_MAX	I/O digital voltage	−0.3	4.0	V
VAA_MAX	Analog voltage	−0.3	4.0	V
VAA_PIX_MAX	Pixel supply voltage	−0.3	4.0	V
VDD_PLL_MAX	PLL supply voltage	−0.3	4.0	V
VIH_MAX	Input HIGH voltage	−0.3	VDD_IO + 0.3	V
VIL_MAX	Input LOW voltage	−0.3		V
T_OP	Operating temperature (measured at junction)	−30	75	°C
T_ST	Storage temperature	−40	85	°C



I/O Parameters

Table 21: I/O Parameters
 $f_{EXTCLK} = 6\text{--}54\text{ MHz}$, $V_{DD} = V_{DDIO_TX} = V_{DD_IO} = 1.8\text{V}$; $V_{AA} = V_{AA_PIX} = V_{DD_PLL} = 2.8\text{V}$; $T_j = 25^\circ\text{C}$

Symbol	Parameter	Min	Typ	Max	Unit	Note
V_{IH}	Input logic HIGH threshold	$V_{DD_IO} - 0.3$	—	$V_{DD_IO} + 0.3$	V	
V_{IL}	Input logic LOW threshold	— 0.3	—	0.6	V	$V_{DD_IO} = 2.8\text{V}$
		— 0.3	—	0.3	V	$V_{DD_IO} = 1.8\text{V}$
V_{OH}	Output logic HIGH threshold	$V_{DD_IO} - 0.3$	—	—	V	At specified at I_{OH}
V_{OL}	Output logic LOW threshold	—	—	0.3	V	
C_{IN}	Input pins - (SCLK, SDATA) capacitance	—	3.5	—	pF	
C_{LOAD}	Output pins - (SDATA) load capacitance		—	30	pF	
PUP_SDATA	SDATA pull-up resistor		1.5		k Ω	

AC Electricals

Table 22: AC Electricals
 $f_{EXTCLK} = 6\text{--}54\text{ MHz}$, $V_{DD} = V_{DDIO_TX} = V_{DD_IO} = 1.8\text{V}$; $V_{AA} = V_{AA_PIX} = V_{DD_PLL} = 2.8\text{V}$; $T_j = 25^\circ\text{C}$

Symbol	Parameter	Min	Typ	Max	Unit	Note
f_{EXTCLK}	External clock input frequency	6	—	54	MHz	
t_r	External clock input rise time	—	2	5	ns	1
t_f	External clock input fall time	—	2	5	ns	1
DC_{EXTCLK}	External clock input duty cycle	40	50	60	%	
t_{JITTER}	External clock input jitter	—	—	500	ps	Peak-to-peak

Notes: 1. Measured when the PLL is off. Specification is not applicable when the PLL is on.



MT9D113: 1/5-Inch SOC Digital Image Sensor Electrical Specifications

DC Electricals

Table 23: DC Electricals

^fEXTCLK = 6–54 MHz; VDD = VDDIO_TX = VDD_IO = 1.8V; VAA = VAA_PIX = VDD_PLL = 2.8V; T_J = 25°C unless stated otherwise

Symbol	Parameter	Condition		Min	Typ	Max	Unit	Notes
VDD	Digital core supply voltage			1.7	1.8	1.95	V	
VDD_PLL	PLL supply voltage			2.5	2.8	3.1	V	
VAA	Analog supply voltage			2.5	2.8	3.1	V	
VAA_PIX	Pixel supply voltage			2.5	2.8	3.1	V	
VDD_IO	Digital IO supply voltage	For VDD_IO = 1.8V		1.7	1.8	1.95	V	
		For VDD_IO = 2.8		2.5	2.8	3.1	v	
VDDIO_TX	MIPI supply voltage			1.7	1.8	1.95	V	
VPP	OTP memory program voltage			–	8	–	V	
IDD_IO	Digital IO supply current	Context A	VDD_IO = 1.8V	–	8	–	mA	1
			VDD_IO = 2.8V	–	15	–	mA	
		Context B	VDD_IO = 1.8V	–	15	–	mA	
			VDD_IO = 2.8V	–	33	–	mA	
IDD_PLL	PLL supply current	PLL is OFF		–	N/A	–	mA	2
		PLL is ON		–	13	–	mA	
IDD	Digital core supply current	Operating in parallel mode	Context A	–	15	–	mA	3
IAA	Analog supply current			–	40	–	mA	
IAA_PIX	Pixel supply current			–	1.5	–	mA	
IDDIO_TX	MIPI supply current			–	N/A	–	mA	
IDD	Digital core supply current		Context B	–	25	–	mA	3
IAA	Analog supply current			–	40	–	mA	
IAA_PIX	Pixel supply current			–	0.8	–	mA	
IDDIO_TX	MIPI supply current			–	N/A	–	mA	
IDD	Digital core supply current	Operating in serial mode	Context A	–	23	–	mA	4
IAA	Analog supply current			–	40	–	mA	
IAA_PIX	Pixel supply current			–	1.5	–	mA	
IDDIO_TX	MIPI supply current			–	5	–	mA	
IDD	Digital core supply current		Context B	–	35	–	mA	
IAA	Analog supply current			–	40	–	mA	
IAA_PIX	Pixel supply current			–	0.8	–	mA	
IDDIO_TX	MIPI supply current			–	8	–	mA	
I _{Hardstandby}	Total standby current	STANDBY signal asserted		–	20	70	μA	5
I _{Softstandby}		R0x0018[0] = 1		–	20	–	μA	6
I _{Softstandby}		R0x0018[0] = 1, R0x0028 = 1		–	800	–	μA	7
I _{Softstandby}		R0x0018[0] = 1 R0x0028 = 0		–	3	–	mA	8

- Notes:
1. VDD_IO current is dependent on the output data rate.
 2. PLL is on with 85 MHz output frequency setting.
 3. VDDIO_TX current is only applicable in serial output mode.
 4. PLL is on with 480 MHz VCO frequency settings.
 5. Either with EXTCLK running or EXTCLK stopped and EXTCLK signal is either pulled up or pulled down. Measured at T_J = 70°C.
 6. EXTCLK is stopped and EXCLK pin is either pulled up or pulled down.
 7. EXTCLK running at 27 MHz and VDD_DIS_SOFT enabled.



MT9D113: 1/5-Inch SOC Digital Image Sensor Electrical Specifications

8. EXTCLK running at 27 MHz and VDD_DIS_SOFT disabled.



Appendix A - Registers

SYSCTL Registers

Table 24: SYSCTL Registers Sorted by Name

Register Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
R22(R0x0016)	clocks_control	dddd dddd dddd dddd	0 (0x0000)
R40(R0x0028)	en_vdd_dis_soft	0000 0000 0000 000d	1 (0x0001)
R36(R0x0024)	gpi_status	0000 0000 0000 0000	0 (0x0000)
R0(R0x0000)	chip_id	0000 0000 0000 0000	9600 (0x2580)
R28(R0x001C)	mcu_boot_mode	dddd dddd dddd dddd	0 (0x0000)
R30(R0x001E)	pad_slew	0000 0ddd 0ddd 0ddd	1024 (0x0400)
R20(R0x0014)	pll_control	d0dd dddd dddd dddd	8697 (0x21F9)
R16(R0x0010)	pll_dividers	00dd dddd dddd dddd	870 (0x0366)
R18(R0x0012)	pll_p_dividers	00dd dddd dddd dddd	245 (0x00F5)
R26(R0x001A)	reset_and_misc_control	0000 d0dd dddd d0dd	80 (0x0050)
R24(R0x0018)	standby_control_and_status	dd00 0000 00dd dddd	16425 (0x4029)
R34(R0x0022)	vdd_dis_counter	dddd dddd dddd dddd	1080 (0x0438)
R32(R0x0020)	vdd_dis_soft	0000 0000 0000 000d	0 (0x0000)

Table 25: Description of SYSCTL Registers

Register Dec(Hex)	Bits	Default	Name
0 R0x0000	15:0	0x2580	chip_id (RO, AOPD)
	Model ID. Read-only.		
16 R0x0010	15:0	0x0366	pll_dividers (R/W, AOPD)
	15:14	X	Reserved.
	13:8	0x0003	n PLL N-Divider value.
	7:0	0x0066	m PLL M-Divider value.
	PLL Dividers.		
18 R0x0012	15:0	0x00F5	pll_p_dividers (R/W, AOPD)
	15:14	X	Reserved.
	13:12	0x0000	word_clock_divider Division ratio of word_clock/word_clockn from bit_clock.
	11:8	0x0000	p3 PLL System Clock Divider Ratio P3. Legal Values: 0, 1, 3, 5, 7, 9, 11, 13, 15.
	7:4	0x000F	p2 PLL System Clock Divider Ratio P2. Legal Values: 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15.
	3:0	0x0005	p1 PLL System Clock Divider Ratio P1. Legal values: 0, 1, 3, 5, 7, 9, 11, 13, 15.
	PLL P Dividers.		


Table 25: Description of SYSTL Registers (continued)

Register Dec(Hex)	Bits	Default	Name
20 R0x0014	15:0	0x21F9	pll_control (R/W, AOPD)
	15	RO	pll_lock PLL Lock Status. This bit indicates when the PLL has acquired lock. After the PLL is enabled and the internal counters are reset, this bit becomes high when PLL lock is detected. 0: PLL lock is not detected. 1: PLL lock is detected.
	14	X	Reserved.
	13	0x0001	clockin_hyst_en Hysteresis on EXTCLK Enable. 0: Hysteresis on EXTCLK is not enabled. 1: Hysteresis on EXTCLK is enabled.
	12	0x0000	Reserved.
	11	0x0000	Reserved.
	10	0x0000	Reserved.
	9	0x0000	Reserved.
	8	0x0001	reset_cntr Counters Reset. This bit is used to reset the counters internally for the PLL to run after initialization. After the PLL is enabled (0x14[1]), reset the counters so the PLL will achieve lock before it is used. 0: Do not reset internal counters. 1: Reset internal counters.
	7:4	0x000F	Reserved.
	3:2	0x0002	Reserved.
	1	0x0000	pll_enable PLL Enable. This bit enables the PLL, and should be enabled after the proper PLL dividers (0x10) and PLL System Clock Divider Ratio (0x12) have been initialized. 0: PLL is not enabled. 1: PLL is enabled.
	0	0x0001	pll_bypass Asserted to bypass the PLL. This should be asserted during PLL initialization. 0: PLL is not bypassed. 1: PLL is bypassed.
	PLL Control.		



Table 25: Description of SYSTL Registers (continued)

Register Dec(Hex)	Bits	Default	Name
22 R0x0016	15:0	0x0000	clocks_control (R/W, AOPD)
	15	0x0000	Reserved.
	14	X	Reserved.
	13	0x0000	Reserved.
	12:11	0x0000	Reserved.
	10	X	Reserved.
	9	0x0000	clk_clkin_en Enable external clock pad. 0: EXTCLK input is disabled. 1: EXTCLK input is enabled.
	8	0x0000	Reserved.
	7	0x0000	Reserved.
	6	0x0000	Reserved.
	5	0x0000	Reserved.
	4	0x0000	Reserved.
	3	0x0000	Reserved.
	2	0x0000	Reserved.
	1	0x0000	Reserved.
	0	0x0000	Reserved.
	Clocks Control.		
24 R0x0018	15:0	0x4029	standby_control_and_status (R/W, AOPD)
	15	RO	Reserved.
	14	0x0001	standby_done Standby State Indicator. When soft standby state is initiated, this bit is asserted when standby state is achieved. 0: SOC not in standby state. 1: SOC in standby state.
	13:6	X	Reserved.
	5	0x0001	Reserved.
	4	0x0000	Reserved.
	3	0x0001	en_xirq Standby Control Enable. This bit is used to disable the standby function (both STANDBY pin and the two-wire serial interface) of the sensor. 0: Standby control not enabled. 1: Standby control enabled.
	2	0x0000	Reserved.
	1	0x0000	Reserved.
	0	0x0001	standby_i2c Soft Standby Bit. This bit is set by default upon power-up, so the sensor starts up in soft standby mode. 0: SOC not in soft standby. 1: SOC in soft standby.
	Standby Control and Status		



Table 25: Description of SYSTL Registers (continued)

Register Dec(Hex)	Bits	Default	Name
26 R0x001A	15:0	0x0050	reset_and_misc_control (R/W, AOPD)
	15:12	X	Reserved.
	11	0x0000	Reserved.
	10	X	Reserved.
	9	0x0000	parallel_enable Parallel Output Enable. 0: Parallel output is disabled. 1: Parallel output is disabled. This disables the I/O buffers for LV, FV, PIXCLK and Dout[7:0]
	8	0x0000	oe_n_enable GPIO[2] Enable. GPIO[2] can be configured as an Output Enable pad. 0: GPIO[2] does not function as OE. 1: GPIO[2] functions as OE.
	7	X	Reserved.
	6	0x0001	Reserved.
	5	0x0000	Reserved.
	4	0x0001	ip_pd_en GPIO Input Pad Power-down Enable. When any GPIO pads are configured as input, this controls if the input pads remains on during standby. 0: GPIO pads remains power-up during standby. 1: GPIO pads are turned off during standby.
	3	0x0000	mipi_tx_en MIPI Transmitter Enable. 0: MIPI transmitter is not enabled. 1: MIPI transmitter is enabled.
	2	X	Reserved.
	1	0x0000	mipi_tx_reset MIPI Transmitter Reset. 0: MIPI transmitter not in reset. 1: MIPI transmitter in reset.
	0	0x0000	reset_soc_i2c Soft Reset. This reset is achieved by the serial two-wire interface, and its effect is the same as the RESET_BAR. 0: SOC not in reset. 1: SOC in reset.
Reset and Control.			
28 R0x001C	15:0	0x0000	mcu_boot_mode (R/W, AOPD)
	15:8	RO	Reserved.
	7:4	0x0000	Reserved.
	3	0x0000	Reserved.
	2	0x0000	Reserved.
	1	0x0000	Reserved.
	0	0x0000	mcu_soft_rst_i2c This bit causes a soft reset. All sensor core registers and variables will be reset to their default states, programmed automatically by firmware. Some of the SOC registers will also be programmed by firmware during the soft reset. 0: MCU is not in reset. 1: MCU is in reset.
	MCU Boot Mode		

**Table 25: Description of SYSCTL Registers (continued)**

Register Dec(Hex)	Bits	Default	Name
30 R0x001E	15:0	0x0400	pad_slew (R/W, AOPD)
	15:11	X	Reserved.
	10:8	0x0004	slew_pxlclk PCLK Slew Rate Control.ce.
	7	X	Reserved.
	6:4	0x0000	slew_gpio GPIO Slew Rate Control.
	3	X	Reserved.
	2:0	0x0000	slew_io Parallel Data Output Slew Rate Control.
Pad Slew Control			
32 R0x0020	15:0	0x0000	vdd_dis_soft (R/W, AOPD)
	Standby without Registers Save. This bit triggers a loss of register states internally to achieve the lowest power consumption during standby. This should be controlled by firmware. 0: Standby without registers save is not enabled. 1: Standby without registers save is enabled.		
34 R0x0022	15:0	0x0438	vdd_dis_counter (R/W, AOPD)
	Internal Standby Delay. This bit controls the delay of when STANDBY assertion takes effect. Delay = $20 * 10^{** - 6} * \text{freq_of_extclk}$. Default is based on a EXTCLK of 54 MHz.		
36 R0x0024	15:0	0x0000	gpi_status (RO, AOPD)
	The status of the GPIO signals are sampled at power up and the states of the pins are stored at this register. This is an alternate way for storing module ID.		
40 R0x0028	15:0	0x0001	en_vdd_dis_soft (R/W, AOPD)
	When the STANDBY pin is asserted, the sensor can enter two standby modes. One mode maintains the registers and current sensor configuration while the other mode clears all the state information, restoring the chip to its default state upon resume. The latter state (no registers and configuration saved) results in the lowest power consumption. This pin determine which state the sensor enters upon a STANDBY assertion. 0: The registers and state variables are retained when STANDBY is asserted. 1: The registers and state variables are not retained when STANDBY is asserted; resulting in the lowest power consumption.		

XDMA Registers

Table 26: XDMA Registers Sorted by Name

Register Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
R2444(R0x098C)	mcu_variable_address	dddd dddd dddd dddd	0 (0x0000)
R2448(R0x0990)	mcu_variable_data0	dddd dddd dddd dddd	0 (0x0000)
R2450(R0x0992)	mcu_variable_data1	dddd dddd dddd dddd	0 (0x0000)
R2452(R0x0994)	mcu_variable_data2	dddd dddd dddd dddd	0 (0x0000)
R2454(R0x0996)	mcu_variable_data3	dddd dddd dddd dddd	0 (0x0000)
R2456(R0x0998)	mcu_variable_data4	dddd dddd dddd dddd	0 (0x0000)
R2458(R0x099A)	mcu_variable_data5	dddd dddd dddd dddd	0 (0x0000)
R2460(R0x099C)	mcu_variable_data6	dddd dddd dddd dddd	0 (0x0000)
R2462(R0x099E)	mcu_variable_data7	dddd dddd dddd dddd	0 (0x0000)



Table 27: Description of XDMA Registers

Register Dec(Hex)	Bits	Default	Name
2444 R0x098C	15:0	0x0000	mcu_variable_address (R/W)
	15	0x0000	access_8bit_mode 8-bit /16-bit Access Select. 0: 16-bit access 1: 8-bit access
	14:13	0x0000	logical_select Select Logical Access Bits 14:13 of address for physical access; R0x98C[14:13]=01 select logical access.
	12:8	0x0000	logical_id Driver ID Bits 12:8 of address for physical access; driver ID for logical access logical_offset.
	7:0	0x0000	Driver variable Bits 7:0 of address for physical access; driver variable offset for logical access.
2448 R0x0990	15:0	0x0000	mcu_variable_data0 (R/W)
	To read a variable from MCU memory, set address in R0x98C and read data from R0x990. To write to a variable, set address in R0x98C and write to R0x990. 16-bit and 8-bit variables can be accessed, see R0x98C. When reading an 8-bit variable (R0x98C[15]=1) R0x990 [15:8] are set to "0." When writing to an 8-bit variable, R0x990 [15:8] are ignored.		
2450 R0x0992	15:0	0x0000	mcu_variable_data1 (R/W)
	Use these registers to read or write up to 16 bytes of variable data using the burst two-wire serial interface access mode. The variables must have consecutive addresses.		
2452 R0x0994	15:0	0x0000	mcu_variable_data2 (R/W)
	Use these registers to read or write up to 16 bytes of variable data using the burst two-wire serial interface access mode. The variables must have consecutive addresses.		
2454 R0x0996	15:0	0x0000	mcu_variable_data3 (R/W)
	Use these registers to read or write up to 16 bytes of variable data using the burst two-wire serial interface access mode. The variables must have consecutive addresses.		
2456 R0x0998	15:0	0x0000	mcu_variable_data4 (R/W)
	Use these registers to read or write up to 16 bytes of variable data using the burst two-wire serial interface access mode. The variables must have consecutive addresses.		
2458 R0x099A	15:0	0x0000	mcu_variable_data5 (R/W)
	Use these registers to read or write up to 16 bytes of variable data using the burst two-wire serial interface access mode. The variables must have consecutive addresses.		
2460 R0x099C	15:0	0x0000	mcu_variable_data6 (R/W)
2462 R0x099E	15:0	0x0000	mcu_variable_data7 (R/W)
	Use these registers to read or write up to 16 bytes of variable data using the burst two-wire serial interface access mode. The variables must have consecutive addresses.		



GPIO Registers

Table 28: GPIO Registers Sorted by Name

Register Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
R4208(R0x1070)	gpio_data	000d ddd0 0000 0000	0 (0x0000)
R4216(R0x1078)	gpio_dir	000d ddd0 0000 0000	7680 (0x1E00)
R4214(R0x1076)	gpio_output_clear	000d ddd0 0000 0000	0 (0x0000)
R4212(R0x1074)	gpio_output_set	000d ddd0 0000 0000	0 (0x0000)

Table 29: Description of GPIO Registers

Register Dec(Hex)	Bits	Default	Name
4208 R0x1070	15:0	0x0000	gpio_data (R/W)
	15:13	X	Reserved.
	12:9	0x0000	gpio_3_0_data Shows the state of GPIO pads and controls some of those pads when configured as outputs. Bits 12 to 9 show/control the state of the pad GPIO3 to GPIO0, respectively. If certain GPIO pads are floating during STANDBY, the corresponding bits should be tied to "0" converting the pads to output to reduce standby current.
	8:0	X	Reserved.
	Shows the state of the GPIO pads and controls those configured as outputs. The state of all GPIO pads are sampled at power-up (see also gpi_status at 0x0024).		
4212 R0x1074	15:0	0x0000	gpio_output_set (R/W)
	15:13	X	Reserved.
	12:9	0x0000	gpio_3_0_output_set Output high when the GPIO is configured as an output. All the bits are read only, and the bits reset back to 0 after they are written. 0: Corresponding GPIO pad unchanged. 1: Corresponding GPIO pad becomes 1.
	8:0	X	Reserved.
	This register sets the desired output GPIO to HIGH. Only GPIO3 and GPIO2 can be controlled using this register.		
4214 R0x1076	15:0	0x0000	gpio_output_clear (R/W)
	15:13	X	Reserved.
	12:9	0x0000	gpio_3_0_output_clear Output low when the GPIO is configured as an output. All the bits are read only, and the bits reset back to 0 after they are written. 0: Corresponding GPIO pad unchanged. 1: Corresponding GPIO pad becomes 0.
	8:0	X	Reserved.
	This register sets the desired output GPIO to LOW. Only GPIO3 and GPIO2 can be controlled using this register.		


Table 29: Description of GPIO Registers (continued)

Register Dec(Hex)	Bits	Default	Name
4216 R0x1078	15:0	0x1E00	gpio_dir (R/W)
	15:13	X	Reserved.
	12	0x0001	gpio_3_dir When configured as a flash output, this pin is controlled by the flash configuration variables in the Sequencer variables. 0: General purpose output. 1: OE_N.
	11	0x0001	gpio_2_dir GPIO2 can be set as input or output. 0: FLASH. 1: General-purpose input.
	10	0x0001	gpio_1_dir For 10-bit output. 0: LSB1. 1: General purpose input.
	9	0x0001	gpio_0_dir To display 10-bit output, this bit needs to be set to output as DOUT_LSB0. 0: DOUT_LSB0. 1: General-purpose input.
	8:0	X	Reserved.
	Controls the direction of five GPIO pads. Each GPIO can be configured independently to be an input or output. Setting any bit to "0" configures the corresponding GPIO pad as an output, and setting any bit to "1" configures the corresponding GPIO pad as an input.		



Core Registers

Table 30: Core Registers Sorted by Name

Register Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
R12334(R0x302E)	analogue_gain_code_blue	0000 0000 0ddd dddd	11 (0x000B)
R12328(R0x3028)	analogue_gain_code_global	0000 0000 0ddd dddd	11 (0x000B)
R12336(R0x3030)	analogue_gain_code_greenb	0000 0000 0ddd dddd	11 (0x000B)
R12330(R0x302A)	analogue_gain_code_greenr	0000 0000 0ddd dddd	11 (0x000B)
R12332(R0x302C)	analogue_gain_code_red	0000 0000 0ddd dddd	11 (0x000B)
R12306(R0x3012)	coarse_integration_time	dddd dddd dddd dddd	16 (0x0010)
R12342(R0x3036)	digital_gain_blue	0000 0ddd 0000 0000	256 (0x0100)
R12344(R0x3038)	digital_gain_greenb	0000 0ddd 0000 0000	256 (0x0100)
R12338(R0x3032)	digital_gain_greenr	0000 0ddd 0000 0000	256 (0x0100)
R12340(R0x3034)	digital_gain_red	0000 0ddd 0000 0000	256 (0x0100)
R12308(R0x3014)	fine_integration_time	dddd dddd dddd dddd	246 (0x00F6)
R12362(R0x304A)	otpm_control	0000 0000 0ddd dddd	0 (0x0000)

Table 31: Description of Core Registers

Register Dec(Hex)	Bits	Default	Name	Frame Sync'd	Bad Frame
12306 R0x3012	15:0	0x0010	coarse_integration_time (R/W)	Y	N
Integration time specified in multiples of line_length_pck_.					
12308 R0x3014	15:0	0x00F6	fine_integration_time (R/W)	Y	N
Integration time specified as a number of pixel clocks.					
12328 R0x3028	15:0	0x000B	analogue_gain_code_global (R/W)	Y	N
Writing a gain code to this register is equivalent to writing that code to each of the 4 color-specific gain code registers. Reading from this register returns the value most recently written to the analogue_gain_code_greenR register.					
12330 R0x302A	15:0	0x000B	analogue_gain_code_greenr (R/W)	Y	N
The gain code written to this register sets the gain for green pixels on red/green rows of the pixel array.					
12332 R0x302C	15:0	0x000B	analogue_gain_code_red (R/W)	Y	N
The gain code written to this register sets the gain for red pixels.					
12334 R0x302E	15:0	0x000B	analogue_gain_code_blue (R/W)	Y	N
The gain code written to this register sets the gain for blue pixels.					
12336 R0x3030	15:0	0x000B	analogue_gain_code_greenb (R/W)	Y	N
The gain code written to this register sets the gain for green pixels on blue/green rows of the pixel array.					
12338 R0x3032	15:0	0x0100	digital_gain_greenr (R/W)	Y	N
Digital gain applied to green pixels on red/green rows of the pixel array. The value is an unsigned 8.8 fixed-point format.					
12340 R0x3034	15:0	0x0100	digital_gain_red (R/W)	Y	N
Digital gain applied to red pixels of the pixel array. The value is an unsigned 8.8 fixed-point format.					
12342 R0x3036	15:0	0x0100	digital_gain_blue (R/W)	Y	N
Digital gain applied to blue pixels of the pixel array. The value is an unsigned 8.8 fixed-point format.					

**Table 31: Description of Core Registers**

Register Dec(Hex)	Bits	Default	Name	Frame Sync'd	Bad Frame
12344 R0x3038	15:0	0x0100	digital_gain_greenb (R/W)	Y	N
Digital gain applied to green pixels on blue/green rows of the pixel array. The value is an unsigned 8.8 fixed-point format.					
12362 R0x304A	15:0	0x0000	otpm_control (R/W)		
	15:7	X	Reserved.		
	6	RO	otpm_control_auto_rd_success Indicates whether the automatic read sequence was successful.	N	N
	5	RO	otpm_control_auto_rd_end Indicates whether the automatic read sequence is finished.	N	N
	4	0x0000	otpm_control_auto_rd_start Triggers automatic OTPM read sequence. The 16-bit value stored in the one-time programmable memory will be read and available in otpm_read_value.	N	N
	3	X	Reserved.		
	2	RO	otpm_control_auto_wr_success Indicates whether the automatic write sequence was successful.	N	N
	1	RO	otpm_control_auto_wr_end Indicates whether the automatic write sequence is finished.	N	N
	0	0x0000	otpm_control_auto_wr_start Triggers automatic OTPM write sequence. The high voltage must be available on the high voltage pad before this sequence is triggered. The 16-bit value in otpm_program_value will be written to the one-time programmable memory.	N	N



SOC1 Registers

Table 32: SOC1 Registers Sorted by Name

Register Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
R12908(R0x326C)	ap2d_aperture_parameters	0ddd dddd dddd dddd	2568 (0x0A08)
R12928(R0x3280)	blue_offset	0000 00dd dddd dddd	42 (0x002A)
R12816(R0x3210)	color_pipeline_control	0000 0d0d d0dd d000	432 (0x01B0)
R12942(R0x328E)	dm_edge_th	0000 0000 dddd dddd	12 (0x000C)
R12924(R0x327C)	green1_offset	0000 00dd dddd dddd	42 (0x002A)
R12926(R0x327E)	green2_offset	0000 00dd dddd dddd	42 (0x002A)
R13300(R0x33F4)	kernel_config	0000 000d dddd dddd	3 (0x0003)
R12828(R0x321C)	output_control_status	dddd dddd d000 dddd	3 (0x0003)
R12830(R0x321E)	output_control_status_2	dddd dddd dddd dddd	16 (0x0010)
R13104(R0x3330)	output_format_test	0000 dddd dddd dddd	0 (0x0000)
R12922(R0x327A)	red_offset	0000 00dd dddd dddd	42 (0x002A)
R12944(R0x3290)	test_pattern	0000 0000 0dd0 0000	0 (0x0000)
R13180(R0x337C)	yuv_ycbcr_control	0000 0000 0000 dddd	6 (0x0006)


Table 33: Description of SOC1 Registers

Register Dec (Hex)	Bits	Default	Name
12816 R0x3210	15:0	0x01B0	color_pipeline_control (R/W)
	15:11	X	Reserved.
	10	0x0000	Reserved.
	9	X	Reserved.
	8	0x0001	bypass_scaler Enable Scaler. 0: Scaler bypassed. 1: Scaler enabled.
	7	0x0001	gamma_en Enable Gamma correction. 0: Gamma correction disabled. 1: Gamma correction enabled.
	6	X	Reserved.
	5	0x0001	en_ccm Enable color correction. 0: Color correction disabled. 1: Color correction enabled.
	4	0x0001	en_ap Enable 2D aperture correction. 0: 2D aperture correction disabled. 1: 2D aperture correction enabled.
	3	0x0000	shading_correction_enable Enable pixel shading correction. All coefficients and other configuration settings (including other fields in this register) must be set up before enabling shading correction. 0: Pixel shading correction disabled. 1: Pixel shading correction enabled.
	2:0	X	Reserved.



Table 33: Description of SOC1 Registers (continued)

Register Dec (Hex)	Bits	Default	Name
12828 R0x321C	15:0	0x0003	output_control_status (R/W)
	15:12	RO	Reserved.
	11:8	0x0000	Reserved.
	7	0x0000	sensor_bypass TX FIFO Input Control This bit determines the source for the output FIFO source data stream; it is only used when sensor_bypass (0x321C[3:0] = 0). 0: Cpipe 1: Sensor
	6:4	X	Reserved.
	3:0	0x0003	txfifo_bypass Output Selection 0: tx_fifo (requires 0x1A[2] to be set). 1: Sensor. 2: Sam observe. 3: Cpipe format. 4: Test walking 1s cpipe frequency. 5: Test walking 1s sensor frequency. 6: Reserved. 7: Test PIXCLK
	Output Control and Status		
12830 R0x321E	15:0	0x0010	output_control_status_2 (R/W)
	15:11	0x0000	Reserved.
	10	0x0000	gated_pixel_clock_output Gated Pixel Clock Output. This bit is used to disable the pixel clock during blanking intervals. 0: Pixel clock is not gated during blanking. 1: Pixel clock is gated during blanking.
	9:0	0x0010	Reserved.
	Output Control and Status 2		
12908 R0x326C	15:0	0x0A08	ap2d_aperture_parameters (R/W)
	15	X	Reserved.
	14	0x0000	ap_abs Positive Gain Enable. 0: Aperture gain is not forced to be positive. 1: Aperture gain is forced to be positive.
	13:11	0x0001	ap_gain_exp 2D aperture gain's exponent
	10:8	0x0002	ap_gain_mant 2D aperture gain
	7:0	0x0008	ap_knee 2D aperture threshold
12922 R0x327A	15:0	0x002A	red_offset (R/W)
	Offset subtracted from red pixels. This register contains the value for Red pixels subtracted by IFP from raw pixel values before applying lens shading correction and digital gains.		

**Table 33: Description of SOC1 Registers (continued)**

Register Dec (Hex)	Bits	Default	Name
12924 R0x327C	15:0	0x002A	green1_offset (R/W) Offset subtracted from green1 pixels. This register contains the value for Green1 pixels subtracted by IFP from raw pixel values before applying lens shading correction and digital gains.
12926 R0x327E	15:0	0x002A	green2_offset (R/W) Offset subtracted from green2 pixels. This register contains the value for Green2 pixels subtracted by IFP from raw pixel values before applying lens shading correction and digital gains.
12928 R0x3280	15:0	0x002A	blue_offset (R/W) Offset subtracted from blue pixels. This register contains the value for Blue pixels subtracted by IFP from raw pixel values before applying lens shading correction and digital gains.
12942 R0x328E	15:0	0x000C	dm_edge_th (R/W) Demosaic Edge Threshold. This is the value used in demosaic to determine if the current pixel is on an edge.
12944 R0x3290	15:0	0x0000	test_pattern (R/W)
	15:7	X	Reserved.
	6	0x0000	walk_ones_10 Walking 1s Test Pattern Width. This bit determines if the walking 1s test pattern is 8-bit or 10-bit wide. 0: 8-bit test pattern. 1: 10-bit test pattern.
	5	0x0000	en_walk_ones_tp Walking 1s Test Pattern Enable. This bit outputs the walking 1s test pattern on the parallel data bus. It can be 8-bit or 10-bit. To configure output stream for the walking 1s test pattern, see also the Output Control and Status register (0x321C). 0: Walking 1s test pattern is disabled. 1: Walking 1s test pattern is enabled.
	4:0	X	Reserved.



Table 33: Description of SOC1 Registers (continued)

Register Dec (Hex)	Bits	Default	Name
13104 R0x3330	15:0	0x0000	output_format_test (R/W)
	15:12	X	Reserved.
	11:9	0x0000	Reserved.
	8	0x0000	bypass_lc Enable Shading Correction Bypass. Bypass 8+2, after shading correction. Used for calibrating shading correction. 8+2 bypass (0x3330[6] = 1) needs to be enabled to use this mode. 0: Shading correction bypass disabled. 1: Shading correction bypass enabled.
	7	0x0000	Reserved.
	6	0x0000	bypass_8_2 Enable 8+2 bypass output. 0: Normal output. 1: 8+2 bypass output.
	5:3	0x0000	test_ramp These bits enable a test ramp output. The test ramp is generated based on the pixel position, Y (or G) = Cr (or R) = Cb (or B) = {pixel_position_count[4:0], 3'b0}. The output format is determined by 0x332E. 00: Test ramp disabled. 01: Increment by column. 10: Increment by row. 11: Increment by frame.
	2	0x0000	cb_dis This bit disables Cb or B channel. 0: Normal. 1: Disable Cb channel in YUV mode (B in RGB mode).
	1	0x0000	y_dis This bit disables Y or G channel. 0: Normal. 1: Disable Y channel in YUV mode (G in RGB mode).
	0	0x0000	cr_dis This bit disables Cr or R channel. 0: Normal. 1: Disable Cr channel in YUV mode (R in RGB mode).
13180 R0x337C	15:0	0x0006	yuv_ycbcr_control (R/W)
	15:4	X	Reserved.
	3	0x0000	clip This bit is used to adjust the range of YUV output for different applications. 0: Y, U and V are not clipped. Y in 0-255. 1: Clip Y in 16-235; U and V in 16-240.
	2	0x0001	add_uv_offset Add 128 to U and V 0: No offset added to U and V. 1: Add 128 to U and V.
	1	0x0001	Reserved.
	0	0x0000	normalize This bit is used to normalize the range of YUV output for different applications. 0: Y, U and V are not normalized. Y in 0-255. 1: Normalize Y in 16-235; U and V in 16-240.


Table 33: Description of SOC1 Registers (continued)

Register Dec (Hex)	Bits	Default	Name
13300 R0x33F4	15:0	0x0003	kernel_config (R/W)
	15:9	X	Reserved.
	8:4	0x0000	Reserved.
	3	0x0000	nr_enable Noise reduction enable. 0: Noise reduction algorithm disabled. 1: Noise reduction algorithm enabled.
	2	0x0000	Reserved.
	1	0x0001	Reserved.
	0	0x0001	dc_enable Defect correction enable. 0: 2D single pixel defect correction disabled. 1: 2D single pixel defect correction enabled.



SOC2 Registers

Table 34: SOC2 Registers Sorted by Name

Register Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
R13702(R0x3586)	ae_window_size_hi	0000 0000 0000 000d	0 (0x0000)
R13700(R0x3584)	ae_window_size_lo	dddd dddd dddd dddd	19200 (0x4B00)
R13696(R0x3580)	ae_zone_x	dddd dddd dddd dddd	4864 (0x1300)
R13698(R0x3582)	ae_zone_y	dddd dddd dddd dddd	3584 (0x0E00)
R13316(R0x3404)	custom_short_pkt	00dd dddd dd00 0000	0 (0x0000)
R13324(R0x340C)	custom_short_pkt_wc	dddd dddd dddd dddd	0 (0x0000)
R13320(R0x3408)	line_byte_cnt	dddd dddd dddd dddd	3200 (0x0C80)
R13312(R0x3400)	mipi_control	dddd dddd dddd dddd	30766 (0x782E)
R13314(R0x3402)	mipi_status	000d ddd0 00dd 000d	17 (0x0011)
R13890(R0x3642)	pga_1	0000 0ddd dddd dddd	604 (0x025C)
R13916(R0x365C)	pga_10	dddd dddd dddd dddd	0 (0x0000)
R14096(R0x3710)	pga_100	dddd dddd dddd dddd	0 (0x0000)
R14098(R0x3712)	pga_101	dddd dddd dddd dddd	0 (0x0000)
R14100(R0x3714)	pga_102	dddd dddd dddd dddd	0 (0x0000)
R13918(R0x365E)	pga_11	dddd dddd dddd dddd	0 (0x0000)
R13920(R0x3660)	pga_12	dddd dddd dddd dddd	0 (0x0000)
R13922(R0x3662)	pga_13	dddd dddd dddd dddd	16 (0x0010)
R13924(R0x3664)	pga_14	dddd dddd dddd dddd	0 (0x0000)
R13926(R0x3666)	pga_15	dddd dddd dddd dddd	0 (0x0000)
R13928(R0x3668)	pga_16	dddd dddd dddd dddd	0 (0x0000)
R13930(R0x366A)	pga_17	dddd dddd dddd dddd	0 (0x0000)
R13932(R0x366C)	pga_18	dddd dddd dddd dddd	16 (0x0010)
R13934(R0x366E)	pga_19	dddd dddd dddd dddd	0 (0x0000)
R13892(R0x3644)	pga_2	0000 0ddd dddd dddd	804 (0x0324)
R13936(R0x3670)	pga_20	dddd dddd dddd dddd	0 (0x0000)
R13938(R0x3672)	pga_21	dddd dddd dddd dddd	0 (0x0000)
R13940(R0x3674)	pga_22	dddd dddd dddd dddd	0 (0x0000)
R13942(R0x3676)	pga_23	dddd dddd dddd dddd	0 (0x0000)
R13944(R0x3678)	pga_24	dddd dddd dddd dddd	0 (0x0000)
R13946(R0x367A)	pga_25	dddd dddd dddd dddd	0 (0x0000)
R13948(R0x367C)	pga_26	dddd dddd dddd dddd	0 (0x0000)
R13950(R0x367E)	pga_27	dddd dddd dddd dddd	0 (0x0000)
R13952(R0x3680)	pga_28	dddd dddd dddd dddd	0 (0x0000)
R13954(R0x3682)	pga_29	dddd dddd dddd dddd	0 (0x0000)
R13902(R0x364E)	pga_3	dddd dddd dddd dddd	16 (0x0010)
R13956(R0x3684)	pga_30	dddd dddd dddd dddd	0 (0x0000)
R13958(R0x3686)	pga_31	dddd dddd dddd dddd	0 (0x0000)
R13960(R0x3688)	pga_32	dddd dddd dddd dddd	0 (0x0000)
R13962(R0x368A)	pga_33	dddd dddd dddd dddd	0 (0x0000)
R13964(R0x368C)	pga_34	dddd dddd dddd dddd	0 (0x0000)
R13966(R0x368E)	pga_35	dddd dddd dddd dddd	0 (0x0000)
R13968(R0x3690)	pga_36	dddd dddd dddd dddd	0 (0x0000)
R13970(R0x3692)	pga_37	dddd dddd dddd dddd	0 (0x0000)



MT9D113: 1/5-Inch SOC Digital Image Sensor

Appendix A - Registers

Table 34: SOC2 Registers Sorted by Name (continued)

Register Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
R13972(R0x3694)	pga_38	dddd dddd dddd dddd	0 (0x0000)
R13974(R0x3696)	pga_39	dddd dddd dddd dddd	0 (0x0000)
R13904(R0x3650)	pga_4	dddd dddd dddd dddd	0 (0x0000)
R13976(R0x3698)	pga_40	dddd dddd dddd dddd	0 (0x0000)
R13978(R0x369A)	pga_41	dddd dddd dddd dddd	0 (0x0000)
R13980(R0x369C)	pga_42	dddd dddd dddd dddd	0 (0x0000)
R13982(R0x369E)	pga_43	dddd dddd dddd dddd	0 (0x0000)
R13984(R0x36A0)	pga_44	dddd dddd dddd dddd	0 (0x0000)
R13986(R0x36A2)	pga_45	dddd dddd dddd dddd	0 (0x0000)
R13988(R0x36A4)	pga_46	dddd dddd dddd dddd	0 (0x0000)
R13990(R0x36A6)	pga_47	dddd dddd dddd dddd	0 (0x0000)
R13992(R0x36A8)	pga_48	dddd dddd dddd dddd	0 (0x0000)
R13994(R0x36AA)	pga_49	dddd dddd dddd dddd	0 (0x0000)
R13906(R0x3652)	pga_5	dddd dddd dddd dddd	0 (0x0000)
R13996(R0x36AC)	pga_50	dddd dddd dddd dddd	0 (0x0000)
R13998(R0x36AE)	pga_51	dddd dddd dddd dddd	0 (0x0000)
R14000(R0x36B0)	pga_52	dddd dddd dddd dddd	0 (0x0000)
R14002(R0x36B2)	pga_53	dddd dddd dddd dddd	0 (0x0000)
R14004(R0x36B4)	pga_54	dddd dddd dddd dddd	0 (0x0000)
R14006(R0x36B6)	pga_55	dddd dddd dddd dddd	0 (0x0000)
R14008(R0x36B8)	pga_56	dddd dddd dddd dddd	0 (0x0000)
R14010(R0x36BA)	pga_57	dddd dddd dddd dddd	0 (0x0000)
R14012(R0x36BC)	pga_58	dddd dddd dddd dddd	0 (0x0000)
R14014(R0x36BE)	pga_59	dddd dddd dddd dddd	0 (0x0000)
R13908(R0x3654)	pga_6	dddd dddd dddd dddd	0 (0x0000)
R14016(R0x36C0)	pga_60	dddd dddd dddd dddd	0 (0x0000)
R14018(R0x36C2)	pga_61	dddd dddd dddd dddd	0 (0x0000)
R14020(R0x36C4)	pga_62	dddd dddd dddd dddd	0 (0x0000)
R14022(R0x36C6)	pga_63	dddd dddd dddd dddd	0 (0x0000)
R14024(R0x36C8)	pga_64	dddd dddd dddd dddd	0 (0x0000)
R14026(R0x36CA)	pga_65	dddd dddd dddd dddd	0 (0x0000)
R14028(R0x36CC)	pga_66	dddd dddd dddd dddd	0 (0x0000)
R14030(R0x36CE)	pga_67	dddd dddd dddd dddd	0 (0x0000)
R14032(R0x36D0)	pga_68	dddd dddd dddd dddd	0 (0x0000)
R14034(R0x36D2)	pga_69	dddd dddd dddd dddd	0 (0x0000)
R13910(R0x3656)	pga_7	dddd dddd dddd dddd	0 (0x0000)
R14036(R0x36D4)	pga_70	dddd dddd dddd dddd	0 (0x0000)
R14038(R0x36D6)	pga_71	dddd dddd dddd dddd	0 (0x0000)
R14040(R0x36D8)	pga_72	dddd dddd dddd dddd	0 (0x0000)
R14042(R0x36DA)	pga_73	dddd dddd dddd dddd	0 (0x0000)
R14044(R0x36DC)	pga_74	dddd dddd dddd dddd	0 (0x0000)
R14046(R0x36DE)	pga_75	dddd dddd dddd dddd	0 (0x0000)
R14048(R0x36E0)	pga_76	dddd dddd dddd dddd	0 (0x0000)
R14050(R0x36E2)	pga_77	dddd dddd dddd dddd	0 (0x0000)
R14052(R0x36E4)	pga_78	dddd dddd dddd dddd	0 (0x0000)



MT9D113: 1/5-Inch SOC Digital Image Sensor

Appendix A - Registers

Table 34: SOC2 Registers Sorted by Name (continued)

Register Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
R14054(R0x36E6)	pga_79	dddd dddd dddd dddd	0 (0x0000)
R13912(R0x3658)	pga_8	dddd dddd dddd dddd	16 (0x0010)
R14056(R0x36E8)	pga_80	dddd dddd dddd dddd	0 (0x0000)
R14058(R0x36EA)	pga_81	dddd dddd dddd dddd	0 (0x0000)
R14060(R0x36EC)	pga_82	dddd dddd dddd dddd	0 (0x0000)
R14062(R0x36EE)	pga_83	dddd dddd dddd dddd	0 (0x0000)
R14064(R0x36F0)	pga_84	dddd dddd dddd dddd	0 (0x0000)
R14066(R0x36F2)	pga_85	dddd dddd dddd dddd	0 (0x0000)
R14068(R0x36F4)	pga_86	dddd dddd dddd dddd	0 (0x0000)
R14070(R0x36F6)	pga_87	dddd dddd dddd dddd	0 (0x0000)
R14072(R0x36F8)	pga_88	dddd dddd dddd dddd	0 (0x0000)
R14074(R0x36FA)	pga_89	dddd dddd dddd dddd	0 (0x0000)
R13914(R0x365A)	pga_9	dddd dddd dddd dddd	0 (0x0000)
R14076(R0x36FC)	pga_90	dddd dddd dddd dddd	0 (0x0000)
R14078(R0x36FE)	pga_91	dddd dddd dddd dddd	0 (0x0000)
R14080(R0x3700)	pga_92	dddd dddd dddd dddd	0 (0x0000)
R14082(R0x3702)	pga_93	dddd dddd dddd dddd	0 (0x0000)
R14084(R0x3704)	pga_94	dddd dddd dddd dddd	0 (0x0000)
R14086(R0x3706)	pga_95	dddd dddd dddd dddd	0 (0x0000)
R14088(R0x3708)	pga_96	dddd dddd dddd dddd	0 (0x0000)
R14090(R0x370A)	pga_97	dddd dddd dddd dddd	0 (0x0000)
R14092(R0x370C)	pga_98	dddd dddd dddd dddd	0 (0x0000)
R14094(R0x370E)	pga_99	dddd dddd dddd dddd	0 (0x0000)


Table 35: Description of SOC2 Registers

Register Dec(Hex)	Bits	Default	Name
13312 R0x3400	15:0	0x782E	mipi_control (R/W)
	15:10	0x001E	data_type Data Format 0x1E: YUV422 8-bit 0x22: 565RGB 0x21: 555RGB 0x20: 444RGB 0x2A: RAW8 0x2B: RAW10
	9	0x0000	Reserved.
	8:6	0x0000	chan_num MIPI channel number. Virtual channel number to be inserted in MIPI packet header (b1:0 only).
	5	0x0001	Reserved.
	4	0x0000	standby_eof Wait until EOF to react to standby. 0: Respond to standby_en at end of current packet/line. 1: Respond to standby_en at end of frame.
	3	RO	reg_frame_sync Frame boundary sync bit. Asserted when it is safe to update frame-synchronized inputs (resets to 1).
	2	0x0001	cont_mipi_clk Continuous MIPI clock Asserted to transmit a continuous MIPI output clock; else clock is only active while data is being transmitted.
	1	0x0001	standby_en MIPI standby Asserted when the chip is to go into standby. The interface completes the transmission of the current packet/frame (as determined by standby_eof), and (for MIPI) sends an end-of-frame packet and goes into ultralow-power mode. Clock must continue to be supplied until mipi_stdby (0x3402[0]) is asserted.
	0	0x0000	restart_en MIPI restart enable. When this bit is asserted, the interface completes the transmission of the current packet/frame and sends an end-of-frame packet.
MIPI_Control.			


Table 35: Description of SOC2 Registers (continued)

Register Dec(Hex)	Bits	Default	Name
13314 R0x3402	15:0	0x0011	mipi_status (R/W)
	15:13	X	Reserved.
	12:11	0x0000	Reserved.
	10:9	RO	Reserved.
	8:6	X	Reserved.
	5	RO	mipi_rdy_for_data MIPI ready to receive data. Asserted once the MIPI interface is enabled and has completed its wake-up and initialization cycles.
	4	RO	mipiccp_idle MIPI idle. Asserted if MIPI interface is not currently transmitting packet data.
	3:1	X	Reserved.
	0	RO	mipi_stdby MIPI in standby. Asserted when the MIPI interface has completed the transition to ultralow-power state in response to the assertion of standby_en and remains asserted until standby_en is deasserted.
	MIPI_Status.		



Table 35: Description of SOC2 Registers (continued)

Register Dec(Hex)	Bits	Default	Name
13316 R0x3404	15:0	0x0000	custom_short_pkt (R/W)
	15:14	X	Reserved.
	13	RO	custom_short_pkt_rst Reset for custom short packet Asserted once the custom short packet has been transmitted to reset custom_short_pkt_req. Since the request may be asserted from an asynchronous, non-continuous clock domain (wrt clk) this signal will remain asserted until it detects that custom_short_pkt_req has transitioned to "0."
	12	0x0000	custom_short_pkt_frame_sync Frame synch for custom short packet When asserted any requested custom short packet will not be transmitted until after the completion of the current frame.
	11	0x0000	custom_short_pkt_req Custom short packet request. Asserted to request the insertion of a short packet in the output datastream. Should be left asserted until reset by custom_short_pkt_rst; must go to "0" before it can be reasserted to make a subsequent request.
	10:8	0x0000	custom_short_pkt_data_type Custom short packet data type The three LSBs of the data type field of a custom short packet.
	7	0x0000	frame_cnt_en Insert frame counter value in WC field Asserted if the frame counter value (8-bit) is to be inserted in the WC field of a start/end of frame short packet (else WC field = 0). Signal should be static for the duration of the frame and should only toggle between frames; that is, when frame_reg_sync is asserted. NOTE: This signal masks the frame counter in the short packet WC field but does not stop the counter internally. To ensure the counter starts at 1, frame_cnt_rst must be asserted and released with frame_cnt_en.
	6	0x0000	frame_cnt_rst Reset the frame count sent in the frame start/end short packets Asserted to reset the frame count transmitted in the frame start/end short packets. Signal must be asserted for at least one clock cycle and should only toggle between frames, that is, when frame_reg_sync is asserted.
	5:0	X	Reserved.
MIPI_Custom_Short_Packet.			
13320 R0x3408	15:0	0x0C80	line_byte_cnt (R/W)
	MIPI line byte count. The number of data bytes (not including the packet header/footer) in the line NOTE: line_byte_cnt must be programmed with a valid value for the dataformat, as specified by the MIPI/CCP specs. For 10-bit data this means this value should be a multiple of 5; for 12-bit data, a multiple of 3; and for YUV/RGB, a multiple of 2.		
13324 R0x340C	15:0	0x0000	custom_short_pkt_wc (R/W)
	The contents of the WC field of a custom short packet Value to insert in the WC field of a custom short packet specified by R0x3404.		



Table 35: Description of SOC2 Registers (continued)

Register Dec(Hex)	Bits	Default	Name
13696 R0x3580	15:0	0x1300	ae_zone_x (R/W)
	15:8	0x0013	ae_zone_x_width Width of each AE zone in x direction divided by 8, minus 1 $w = (ae_zone_x_width + 1) * 8$
	7:0	0x0000	ae_zone_x_start Start of AE window divided by 8
13698 R0x3582	15:0	0x0E00	ae_zone_y (R/W)
	15:8	0x000E	ae_zone_y_width Width of each AE zone in y direction divided by 8, minus 1. $w = (ae_zone_y_width + 1) * 8$
	7:0	0x0000	ae_zone_y_start Start of AE window divided by 8
13700 R0x3584	15:0	0x4B00	ae_window_size_lo (R/W)
	Least significant bits of the size of each AE zone in terms of number of pixels.		
13702 R0x3586	15:0	0x0000	ae_window_size_hi (R/W)
	Most significant bit of the size of each AE zone in terms of number of pixels.		
13890 R0x3642	15:0	0x025C	pga_1 (R/W)
	PGA coefficients 1.		
13892 R0x3644	15:0	0x0324	pga_2 (R/W)
	PGA coefficients 2.		
13902 R0x364E	15:0	0x0010	pga_3 (R/W)
	PGA coefficients 3.		
13904 R0x3650	15:0	0x0000	pga_4 (R/W)
	PGA coefficients 4.		
13906 R0x3652	15:0	0x0000	pga_5 (R/W)
	PGA coefficients 5.		
13908 R0x3654	15:0	0x0000	pga_6 (R/W)
	PGA coefficients 6.		
13910 R0x3656	15:0	0x0000	pga_7 (R/W)
	PGA coefficients 7.		
13912 R0x3658	15:0	0x0010	pga_8 (R/W)
	PGA coefficients 8.		
13914 R0x365A	15:0	0x0000	pga_9 (R/W)
	PGA coefficients 9.		
13916 R0x365C	15:0	0x0000	pga_10 (R/W)
	PGA coefficients 10.		
13918 R0x365E	15:0	0x0000	pga_11 (R/W)
	PGA coefficients 11.		
13920 R0x3660	15:0	0x0000	pga_12 (R/W)
	PGA coefficients 12.		
13922 R0x3662	15:0	0x0010	pga_13 (R/W)
	PGA coefficients 13.		



Table 35: Description of SOC2 Registers (continued)

Register Dec(Hex)	Bits	Default	Name
13924 R0x3664	15:0	0x0000	pga_14 (R/W)
	PGA coefficients 14.		
13926 R0x3666	15:0	0x0000	pga_15 (R/W)
	PGA coefficients 15.		
13928 R0x3668	15:0	0x0000	pga_16 (R/W)
	PGA coefficients 16.		
13930 R0x366A	15:0	0x0000	pga_17 (R/W)
	PGA coefficients 17.		
13932 R0x366C	15:0	0x0010	pga_18 (R/W)
	PGA coefficients 18.		
13934 R0x366E	15:0	0x0000	pga_19 (R/W)
	PGA coefficients 19.		
13936 R0x3670	15:0	0x0000	pga_20 (R/W)
	PGA coefficients 20.		
13938 R0x3672	15:0	0x0000	pga_21 (R/W)
	PGA coefficients 21.		
13940 R0x3674	15:0	0x0000	pga_22 (R/W)
	PGA coefficients 22.		
13942 R0x3676	15:0	0x0000	pga_23 (R/W)
	PGA coefficients 23.		
13944 R0x3678	15:0	0x0000	pga_24 (R/W)
	PGA coefficients 24.		
13946 R0x367A	15:0	0x0000	pga_25 (R/W)
	PGA coefficients 25.		
13948 R0x367C	15:0	0x0000	pga_26 (R/W)
	PGA coefficients 26.		
13950 R0x367E	15:0	0x0000	pga_27 (R/W)
	PGA coefficients 27.		
13952 R0x3680	15:0	0x0000	pga_28 (R/W)
	PGA coefficients 28.		
13954 R0x3682	15:0	0x0000	pga_29 (R/W)
	PGA coefficients 29.		
13956 R0x3684	15:0	0x0000	pga_30 (R/W)
	PGA coefficients 30.		
13958 R0x3686	15:0	0x0000	pga_31 (R/W)
	PGA coefficients 31.		
13960 R0x3688	15:0	0x0000	pga_32 (R/W)
	PGA coefficients 32.		

**Table 35: Description of SOC2 Registers (continued)**

Register Dec(Hex)	Bits	Default	Name
13962 R0x368A	15:0	0x0000	pga_33 (R/W)
	PGA coefficients 33.		
13964 R0x368C	15:0	0x0000	pga_34 (R/W)
	PGA coefficients 34.		
13966 R0x368E	15:0	0x0000	pga_35 (R/W)
	PGA coefficients 35.		
13968 R0x3690	15:0	0x0000	pga_36 (R/W)
	PGA coefficients 36.		
13970 R0x3692	15:0	0x0000	pga_37 (R/W)
	PGA coefficients 37.		
13972 R0x3694	15:0	0x0000	pga_38 (R/W)
	PGA coefficients 38.		
13974 R0x3696	15:0	0x0000	pga_39 (R/W)
	PGA coefficients 39.		
13976 R0x3698	15:0	0x0000	pga_40 (R/W)
	PGA coefficients 40.		
13978 R0x369A	15:0	0x0000	pga_41 (R/W)
	PGA coefficients 41.		
13980 R0x369C	15:0	0x0000	pga_42 (R/W)
	PGA coefficients 42.		
13982 R0x369E	15:0	0x0000	pga_43 (R/W)
	PGA coefficients 43.		
13984 R0x36A0	15:0	0x0000	pga_44 (R/W)
	PGA coefficients 44.		
13986 R0x36A2	15:0	0x0000	pga_45 (R/W)
	PGA coefficients 45.		
13988 R0x36A4	15:0	0x0000	pga_46 (R/W)
	PGA coefficients 46.		
13990 R0x36A6	15:0	0x0000	pga_47 (R/W)
	PGA coefficients 47.		
13992 R0x36A8	15:0	0x0000	pga_48 (R/W)
	PGA coefficients 48.		
13994 R0x36AA	15:0	0x0000	pga_49 (R/W)
	PGA coefficients 49.		
13996 R0x36AC	15:0	0x0000	pga_50 (R/W)
	PGA coefficients 50.		
13998 R0x36AE	15:0	0x0000	pga_51 (R/W)
	PGA coefficients 51.		

**Table 35: Description of SOC2 Registers (continued)**

Register Dec(Hex)	Bits	Default	Name
14000 R0x36B0	15:0	0x0000	pga_52 (R/W)
	PGA coefficients 52.		
14002 R0x36B2	15:0	0x0000	pga_53 (R/W)
	PGA coefficients 53.		
14004 R0x36B4	15:0	0x0000	pga_54 (R/W)
	PGA coefficients 54.		
14006 R0x36B6	15:0	0x0000	pga_55 (R/W)
	PGA coefficients 55.		
14008 R0x36B8	15:0	0x0000	pga_56 (R/W)
	PGA coefficients 56.		
14010 R0x36BA	15:0	0x0000	pga_57 (R/W)
	PGA coefficients 57.		
14012 R0x36BC	15:0	0x0000	pga_58 (R/W)
	PGA coefficients 58.		
14014 R0x36BE	15:0	0x0000	pga_59 (R/W)
	PGA coefficients 59.		
14016 R0x36C0	15:0	0x0000	pga_60 (R/W)
	PGA coefficients 60.		
14018 R0x36C2	15:0	0x0000	pga_61 (R/W)
	PGA coefficients 61.		
14020 R0x36C4	15:0	0x0000	pga_62 (R/W)
	PGA coefficients 62.		
14022 R0x36C6	15:0	0x0000	pga_63 (R/W)
	PGA coefficients 63.		
14024 R0x36C8	15:0	0x0000	pga_64 (R/W)
	PGA coefficients 64.		
14026 R0x36CA	15:0	0x0000	pga_65 (R/W)
	PGA coefficients 65.		
14028 R0x36CC	15:0	0x0000	pga_66 (R/W)
	PGA coefficients 66.		
14030 R0x36CE	15:0	0x0000	pga_67 (R/W)
	PGA coefficients 67.		
14032 R0x36D0	15:0	0x0000	pga_68 (R/W)
	PGA coefficients 68.		
14034 R0x36D2	15:0	0x0000	pga_69 (R/W)
	PGA coefficients 69.		
14036 R0x36D4	15:0	0x0000	pga_70 (R/W)
	PGA coefficients 70.		

**Table 35: Description of SOC2 Registers (continued)**

Register Dec(Hex)	Bits	Default	Name
14038 R0x36D6	15:0	0x0000	pga_71 (R/W)
	PGA coefficients 71.		
14040 R0x36D8	15:0	0x0000	pga_72 (R/W)
	PGA coefficients 72.		
14042 R0x36DA	15:0	0x0000	pga_73 (R/W)
	PGA coefficients 73.		
14044 R0x36DC	15:0	0x0000	pga_74 (R/W)
	PGA coefficients 74.		
14046 R0x36DE	15:0	0x0000	pga_75 (R/W)
	PGA coefficients 75.		
14048 R0x36E0	15:0	0x0000	pga_76 (R/W)
	PGA coefficients 76.		
14050 R0x36E2	15:0	0x0000	pga_77 (R/W)
	PGA coefficients 77.		
14052 R0x36E4	15:0	0x0000	pga_78 (R/W)
	PGA coefficients 78.		
14054 R0x36E6	15:0	0x0000	pga_79 (R/W)
	PGA coefficients 79.		
14056 R0x36E8	15:0	0x0000	pga_80 (R/W)
	PGA coefficients 80.		
14058 R0x36EA	15:0	0x0000	pga_81 (R/W)
	PGA coefficients 81.		
14060 R0x36EC	15:0	0x0000	pga_82 (R/W)
	PGA coefficients 82.		
14062 R0x36EE	15:0	0x0000	pga_83 (R/W)
	PGA coefficients 83.		
14064 R0x36F0	15:0	0x0000	pga_84 (R/W)
	PGA coefficients 84.		
14066 R0x36F2	15:0	0x0000	pga_85 (R/W)
	PGA coefficients 85.		
14068 R0x36F4	15:0	0x0000	pga_86 (R/W)
	PGA coefficients 86.		
14070 R0x36F6	15:0	0x0000	pga_87 (R/W)
	PGA coefficients 87.		
14072 R0x36F8	15:0	0x0000	pga_88 (R/W)
	PGA coefficients 88.		
14074 R0x36FA	15:0	0x0000	pga_89 (R/W)
	PGA coefficients 89.		

**Table 35: Description of SOC2 Registers (continued)**

Register Dec(Hex)	Bits	Default	Name
14076 R0x36FC	15:0	0x0000	pga_90 (R/W)
	PGA coefficients 90.		
14078 R0x36FE	15:0	0x0000	pga_91 (R/W)
	PGA coefficients 91.		
14080 R0x3700	15:0	0x0000	pga_92 (R/W)
	PGA coefficients 92.		
14082 R0x3702	15:0	0x0000	pga_93 (R/W)
	PGA coefficients 93.		
14084 R0x3704	15:0	0x0000	pga_94 (R/W)
	PGA coefficients 94.		
14086 R0x3706	15:0	0x0000	pga_95 (R/W)
	PGA coefficients 95.		
14088 R0x3708	15:0	0x0000	pga_96 (R/W)
	PGA coefficients 96.		
14090 R0x370A	15:0	0x0000	pga_97 (R/W)
	PGA coefficients 97.		
14092 R0x370C	15:0	0x0000	pga_98 (R/W)
	PGA coefficients 98.		
14094 R0x370E	15:0	0x0000	pga_99 (R/W)
	PGA coefficients 99.		
14096 R0x3710	15:0	0x0000	pga_100 (R/W)
	PGA coefficients 100.		
14098 R0x3712	15:0	0x0000	pga_101 (R/W)
	PGA coefficients 101.		
14100 R0x3714	15:0	0x0000	pga_102 (R/W)
	PGA coefficients 102.		



Appendix B - Driver Variables

Sequencer Variables (ID = 1)

Table 36: Sequencer Variables Sorted by Name

Variable Dec (Hex)	Name	Data Format (Binary)	Default Value Dec (Hex)
9 (0x0009)	seq_ae_fast_buff	0000 0000 dddd dddd	16 (0x0010)
10 (0x000A)	seq_ae_fast_step	0000 0000 dddd dddd	2 (0x0002)
11 (0x000B)	seq_awb_cont_buff	0000 0000 dddd dddd	8 (0x0008)
12 (0x000C)	seq_awb_cont_step	0000 0000 dddd dddd	2 (0x0002)
21 (0x0015)	seq_cap_mode	0000 0000 dddd dddd	0 (0x0000)
22 (0x0016)	seq_cap_num_frames	0000 0000 dddd dddd	3 (0x0003)
3 (0x0003)	seq_cmd	0000 0000 dddd dddd	1 (0x0001)
19 (0x0013)	seq_flash_th	dddd dddd dddd dddd	0 (0x0000)
6 (0x0006)	seq_flash_type	0000 0000 dddd dddd	0 (0x0000)
2 (0x0002)	seq_mode	0000 0000 dddd dddd	15 (0x000F)
17 (0x0011)	seq_options	0000 0000 dddd dddd	8 (0x0008)
23 (0x0017)	seq_preview_0_ae	0000 0000 dddd dddd	0 (0x0000)
25 (0x0019)	seq_preview_0_awb	0000 0000 dddd dddd	0 (0x0000)
24 (0x0018)	seq_preview_0_fd	0000 0000 dddd dddd	0 (0x0000)
27 (0x001B)	seq_preview_0_flash	0000 0000 dddd dddd	0 (0x0000)
26 (0x001A)	seq_preview_0_hg	0000 0000 dddd dddd	0 (0x0000)
28 (0x001C)	seq_preview_0_skipframe	0000 0000 dddd dddd	64 (0x0040)
29 (0x001D)	seq_preview_1_ae	0000 0000 dddd dddd	1 (0x0001)
31 (0x001F)	seq_preview_1_awb	0000 0000 dddd dddd	1 (0x0001)
30 (0x001E)	seq_preview_1_fd	0000 0000 dddd dddd	1 (0x0001)
33 (0x0021)	seq_preview_1_flash	0000 0000 dddd dddd	0 (0x0000)
32 (0x0020)	seq_preview_1_hg	0000 0000 dddd dddd	1 (0x0001)
34 (0x0022)	seq_preview_1_skipframe	0000 0000 dddd dddd	0 (0x0000)
35 (0x0023)	seq_preview_2_ae	0000 0000 dddd dddd	0 (0x0000)
37 (0x0025)	seq_preview_2_awb	0000 0000 dddd dddd	0 (0x0000)
36 (0x0024)	seq_preview_2_fd	0000 0000 dddd dddd	0 (0x0000)
39 (0x0027)	seq_preview_2_flash	0000 0000 dddd dddd	0 (0x0000)
38 (0x0026)	seq_preview_2_hg	0000 0000 dddd dddd	0 (0x0000)
40 (0x0028)	seq_preview_2_skipframe	0000 0000 dddd dddd	64 (0x0040)
41 (0x0029)	seq_preview_3_ae	0000 0000 dddd dddd	0 (0x0000)
43 (0x002B)	seq_preview_3_awb	0000 0000 dddd dddd	0 (0x0000)
42 (0x002A)	seq_preview_3_fd	0000 0000 dddd dddd	0 (0x0000)
45 (0x002D)	seq_preview_3_flash	0000 0000 dddd dddd	0 (0x0000)
44 (0x002C)	seq_preview_3_hg	0000 0000 dddd dddd	0 (0x0000)
46 (0x002E)	seq_preview_3_skipframe	0000 0000 dddd dddd	64 (0x0040)
4 (0x0004)	seq_state	0000 0000 dddd dddd	0 (0x0000)

**Table 37: Description of Sequencer Variables**

Variable Dec(Hex)	Bits	Default	Name
2 0x0002	15:0	0x000F	seq_mode (R/W)
	15:4	X	Reserved.
	3	0x0001	seq_mode_hg Statistics Engine. 0: Statistics engine disabled. 1: Statistics engine enabled.
	2	0x0001	seq_mode_awb Auto White Balance Driver. 0: Auto white balance disabled. 1: Auto white balance enabled.
	1	0x0001	seq_mode_fd Flicker Detection Driver. 0: Flicker detection disabled. 1: Flicker detection enabled.
	0	0x0001	seq_mode_ae Auto Exposure Driver. 0: Auto exposure disabled. 1: Auto exposure enabled.
	Set of 1-bit switches enabling various drivers and white balance option. Writing 1 to a particular switch enables the corresponding driver or option. Changes take effect immediately.		
3 0x0003	15:0	0x0001	seq_cmd (R/W)
	Command or program to execute: 0: Run. 1: Go to Preview mode. 2: Go to Capture mode . 3: Reserved. 4: Reserved. 5: Refresh. 6: Refresh mode. Writing a positive value to this variable commands the sequencer to execute the corresponding program. The sequencer resets cmd to 0, executes the program, and then resumes running in its current state.Changes take effect immediately. It is recommended that Refresh and Refresh Mode commands always be run together, and that Refresh Mode should be issued before the Refresh command.		
4 0x0004	15:0	0x0000	seq_state (R/W)
	Current state of sequencer 0: Reserved. 1: Reserved. 2: Enter Preview. 3: Preview . 4: Leave Preview. 5: Reserved. 6: Enter Capture. 7: Capture. 8: Reserved. 9: Reserved.		

**Table 37: Description of Sequencer Variables (Continued)**

Variable Dec(Hex)	Bits	Default	Name
6 0x0006	15:0	0x0000	seq_flash_type (R/W)
	15:8	X	Reserved.
	7	0x0000	seq_flash_type_flash_on_lock Set the flash to LOCK mode. 0: Normal. 1: LOCK mode.
	6:0	0x0000	seq_flash_type_flash_type 0: None. 1: LED. 2: Xenon. 3: Xenon burst; xenon burst is used to turn the flash on every frame till flash is turned off.
Type of flash to be used			
9 0x0009	15:0	0x0010	seq_ae_fast_buff (R/W)
	Weighting factor determining to what extent AE driver averages luma over time. A setting of 32 results in the fastest AE speed; a setting of 1 results in the slowest AE speed.		
10 0x000A	15:0	0x0002	seq_ae_fast_step (R/W)
	Number of steps in which AE driver is expected to reach target brightness. The smaller the number, the faster the AE reaches the target brightness. The bigger the number, the smoother the AE transitions.		
11 0x000B	15:0	0x0008	seq_awb_cont_buff (R/W)
	Weighting factor determining to what extent AWB driver averages digital gains over time. A setting of 32 results in the fastest AWB speed; a setting of 1 results in the slowest AWB speed.		
12 0x000C	15:0	0x0002	seq_awb_cont_step (R/W)
	Number of steps in which AWB driver is expected to reach target color balance. The smaller the number, the faster the AWB reaches the target color balance. The bigger the number, the smoother the AWB transitions.		
17 0x0011	15:0	0x0008	seq_options (R/W)
	15:8	X	Reserved.
	7	0x0000	Reserved.
	6:5	X	Reserved.
	4	0x0000	seq_crop_win_awb Use crop window for AWB statistics.
	3	0x0001	seq_crop_win_ae Use crop window for AE statistics.
	2:0	0x0000	Reserved.
	Options for both Preview (Context A) and Snapshot (Context B).		
19 0x0013	15:0	0x0000	seq_flash_th (R/W)
	Exposure value (EV) threshold. If current EV is below this threshold, flash will be used in autoevaluate flash mode. Changes take effect immediately.		

**Table 37: Description of Sequencer Variables (Continued)**

Variable Dec(Hex)	Bits	Default	Name
21 0x0015	15:0	0x0000	seq_cap_mode (R/W)
	15:7	X	Reserved.
	6	0x0000	seq_cap_mode_video_fd_on Flicker Detection (FD) in Capture and Video Mode. 0: FD is disabled in Capture and Video Mode. 1: FD is enabled in Capture and Video Mode.
	5	0x0000	seq_cap_mode_video_awb_hg_on Auto White Balance (AWB) in Capture and Video Mode. 0: AWB is disabled in Capture and Video Mode. 1: AWB is enabled in Capture and Video Mode.
	4	0x0000	seq_cap_mode_video_ae_hg_on Auto Exposure (AE) in Capture and Video Mode. 0: AE is disabled in Capture and Video Mode. 1: AE is enabled in Capture and Video Mode.
	3	X	Reserved
	2	0x0000	seq_cap_mode_flash_off_last_frame Turn flash off before last frame in capture state.
	1	0x0000	seq_cap_mode_video Video/Capture Mode. When configured as Capture mode, the sequencer returns to Preview mode after the number of frames to be capture is done. When configured as Video mode, the sequencer does not return to Preview mode until Preview mode is selected for switch. 0: Capture mode. 1: Video mode.
	0	0x0000	seq_cap_mode_xenon_flash Capture with Xenon flash. 0: Taking Snapshot does not enable Xenon flash. 1: Taking Snapshot enables Xenon flash.
	Capture mode parameters. Takes effect in Capture state only.		
22 0x0016	15:0	0x0003	seq_cap_num_frames (R/W)
	Number of frames captured in still capture mode. Changes take effect in Capture state only.		
23 0x0017	15:0	0x0000	seq_preview_0_ae (R/W)
	Auto exposure configuration in PreviewEnter state 0: Off. 1: Dynamic range tracking. 2: Brightness tracking. 3: Rules only (Ignore clipping).		
24 0x0018	15:0	0x0000	seq_preview_0_fd (R/W)
	Flicker detection configuration in PreviewEnter state 0: Off. 1: On. 2: Manual.		
25 0x0019	15:0	0x0000	seq_preview_0_awb (R/W)
	Auto white balance configuration in PreviewEnter state. 0: Off. 1: On.		

**Table 37: Description of Sequencer Variables (Continued)**

Variable Dec(Hex)	Bits	Default	Name
26 0x001A	15:0	0x0000	seq_preview_0_hg (R/W)
	Statistics engine configuration in PreviewEnter state. 0: Off. 1: On.		
27 0x001B	15:0	0x0000	seq_preview_0_flash (R/W)
	15:8	X	Reserved.
	7	0x0000	Reserved.
	6:0	0x0000	seq_flash_config_0 Flash configuration in PreviewEnter state. 0: Off. 1: On. 2: Locked/Manual. 3: Auto Evaluate.
PreviewEnter state configuration for Flash.			
28 0x001C	15:0	0x0040	seq_preview_0_skipframe (R/W)
	15:8	X	Reserved.
	7	0x0000	seq_preview_0_skipframe_turn_off_fen 0: Frame valid remains on during skipped frame. 1: Frame valid off output during skipped frame.
	6	0x0001	seq_preview_0_skipframe_skip_state 0: No skip state. 1: Skip state.
	5	0x0000	seq_preview_0_skipframe_skip_led_on Skip first frame when LED On 0: When LED is on, first frame is not skipped. 1: When LED is on, first frame is skipped.
	4:0	X	Reserved.
	PreviewEnter state configuration for skipframe		
29 0x001D	15:0	0x0001	seq_preview_1_ae (R/W)
	Auto exposure configuration in Preview state . 0: Off. 1: Dynamic range tracking. 2: Brightness tracking. 3: Rules only (Ignore clipping).		
30 0x001E	15:0	0x0001	seq_preview_1_fd (R/W)
	Flicker detection configuration in Preview state. 0: Off. 1: On. 2: Manual.		
31 0x001F	15:0	0x0001	seq_preview_1_awb (R/W)
	Auto white balance configuration in Preview state. 0: Off. 1: On.		
32 0x0020	15:0	0x0001	seq_preview_1_hg (R/W)
	Statistics engine configuration in Preview state. 0: Off. 1: On.		

**Table 37: Description of Sequencer Variables (Continued)**

Variable Dec(Hex)	Bits	Default	Name
33 0x0021	15:0	0x0000	seq_preview_1_flash (R/W)
	15:8	X	Reserved.
	7	0x0000	Reserved.
	6:0	0x0000	seq_flash_config_1 Flash configuration in Preview state. 0: Off. 1: On. 2: Locked/Manual. 3: Auto Evaluate.
	Preview state configuration for Flash.		
34 0x0022	15:0	0x0000	seq_preview_1_skipframe (R/W)
	15:8	X	Reserved.
	7	0x0000	seq_preview_1_skipframe_turn_off_fen 0: Frame valid remains on during skipped frame. 1: Frame valid off output during skipped frame.
	6	0x0000	seq_preview_1_skipframe_skip_state 0: No skip state. 1: Skip state.
	5	0x0000	seq_preview_1_skipframe_skip_led_on Skip first frame when LED On 0: When LED is on, first frame is not skipped. 1: When LED is on, first frame is skipped.
	4:0	X	Reserved
Preview state configuration for skipframe			
35 0x0023	15:0	0x0000	seq_preview_2_ae (R/W)
	Auto exposure configuration in PreviewLeave state 0: Off 1: Dynamic range tracking 2: Brightness tracking 3: Rules only (Ignore clipping)		
36 0x0024	15:0	0x0000	seq_preview_2_fd (R/W)
	Flicker detection configuration in PreviewLeave state . 0: Off. 1: On. 2: Manual.		
37 0x0025	15:0	0x0000	seq_preview_2_awb (R/W)
	Auto white balance configuration in PreviewLeave state. 0: Off. 1: On.		
38 0x0026	15:0	0x0000	seq_preview_2_hg (R/W)



Table 37: Description of Sequencer Variables (Continued)

Variable Dec(Hex)	Bits	Default	Name
39 0x0027	15:0	0x0000	seq_preview_2_flash (R/W)
	15:8	X	Reserved.
	7	0x0000	Reserved.
	6:0	0x0000	seq_flash_config_2 Flash configuration in PreviewLeave state. 0: Off. 1: On. 2: Locked/Manual. 3: Auto Evaluate.
	PreviewLeave state configuration for Flash		
40 0x0028	15:0	0x0040	seq_preview_2_skipframe (R/W)
	15:8	X	Reserved.
	7	0x0000	seq_preview_2_skipframe_turn_off_fen 0: Frame valid remains on during skipped frame. 1: Frame valid off output during skipped frame.
	6	0x0001	seq_preview_2_skipframe_skip_state 0: No skip state. 1: Skip state.
	5	0x0000	seq_preview_2_skipframe_skip_led_on Skip first frame when LED On. 0: When LED is on, first frame is not skipped. 1: When LED is on, first frame is skipped.
	4	0x0000	Reserved.
	3:0	X	Reserved.
	PreviewLeave state configuration for skipframe.		
41 0x0029	15:0	0x0000	seq_preview_3_ae (R/W)
	Auto exposure configuration in CaptureEnter state. 0: Off. 1: Dynamic range tracking. 2: Brightness tracking. 3: Rules only (Ignore clipping).		
42 0x002A	15:0	0x0000	seq_preview_3_fd (R/W)
	Flicker detection configuration in CaptureEnter state . 0: Off. 1: On. 2: Manual.		
43 0x002B	15:0	0x0000	seq_preview_3_awb (R/W)
	Auto white balance configuration in CaptureEnter state. 0: Off. 1: On.		
44 0x002C	15:0	0x0000	seq_preview_3_hg (R/W)
	Statistics engine configuration in CaptureEnter state. 0: Off. 1: On.		


Table 37: Description of Sequencer Variables (Continued)

Variable Dec(Hex)	Bits	Default	Name
45 0x002D	15:0	0x0000	seq_preview_3_flash (R/W)
	15:8	X	Reserved.
	7	0x0000	Reserved.
	6:0	0x0000	seq_flash_config_3 Flash configuration in CaptureEnter state. 0: Off. 1: On. 2: Locked/Manual. 3: Auto Evaluate.
	CaptureEnter state configuration for Flash.		
46 0x002E	15:0	0x0040	seq_preview_3_skipframe (R/W)
	15:8	X	Reserved.
	7	0x0000	seq_preview_3_skipframe_turn_off_fen 0: Frame valid remains on during skipped frame. 1: Frame valid off output during skipped frame.
	6	0x0001	seq_preview_3_skipframe_skip_state 0: No skip state. 1: Skip state.
	5	0x0000	seq_preview_3_skipframe_skip_led_on Skip first frame when LED On. 0: When LED is on, first frame is not skipped. 1: When LED is on, first frame is skipped.
	4:0	X	Reserved.
	CaptureEnter state configuration for skipframe.		



Auto Exposure Variables (ID = 2)

Table 38: Auto Exposure Variables Sorted by Name

Variable Dec (Hex)	Name	Data Format (Binary)	Default Value Dec (Hex)
79 (0x004F)	ae_base_target	0000 0000 dddd dddd	54 (0x0036)
24 (0x0018)	ae_current_y	0000 0000 ??? ???? ?	75 (0x004B)
7 (0x0007)	ae_gate	0000 0000 dddd dddd	4 (0x0004)
27 (0x001B)	ae_index	0000 0000 ??? ???? ?	4 (0x0004)
12 (0x000C)	ae_max_index	0000 0000 dddd dddd	24 (0x0018)
14 (0x000E)	ae_max_virt_gain	0000 0000 dddd dddd	128 (0x0080)
11 (0x000B)	ae_min_index	0000 0000 dddd dddd	0 (0x0000)
13 (0x000D)	ae_min_virt_gain	0000 0000 dddd dddd	16 (0x0010)
25 (0x0019)	ae_r12	??? ???? ???? ?	633 (0x0279)
34 (0x0022)	ae_r9	dddd dddd dddd dddd	16 (0x0010)
45 (0x002D)	ae_r9_step	dddd dddd dddd dddd	157 (0x009D)
23 (0x0017)	ae_status	0000 0000 0000 0ddd	0 (0x0000)
76 (0x004C)	ae_target_buffer_speed	0000 0000 dddd dddd	12 (0x000C)
75 (0x004B)	ae_target_max	0000 0000 dddd dddd	150 (0x0096)
74 (0x004A)	ae_target_min	0000 0000 dddd dddd	50 (0x0032)
28 (0x001C)	ae_virt_gain	0000 0000 ??? ???? ?	16 (0x0010)
2 (0x0002)	ae_window_pos	0000 0000 dddd dddd	0 (0x0000)
3 (0x0003)	ae_window_size	0000 0000 dddd dddd	255 (0x00FF)

Table 39: Description of Auto Exposure Variables

Variable Dec (Hex)	Bits	Default	Name
2 0x0002	15:0	0x0000	ae_window_pos (R/W)
	15:8	X	Reserved.
	7:4	0x0000	ae_y0 Y0 in units 1/16 of frame height.
	3:0	0x0000	ae_x0 X0 in units of 1/16 of frame width.
	Position of first (upper left) AE window New position written to this variable takes effect only after REFRESH command is given to the sequencer (seq.cmd = 5).		
3 0x0003	15:0	0x00FF	ae_window_size (R/W)
	15:8	X	Reserved.
	7:4	0x000F	ae_height Height (in units of 1/16 of frame height) decremented by 1.
	3:0	0x000F	ae_width Width (in units of 1/16 of frame width) decremented by 1.
	Size of 4 x 4 array of AE windows New data written to this variable take effect only after REFRESH command is given to the sequencer (seq.cmd = 5).		
7 0x0007	15:0	0x0004	ae_gate (R/W)
	AE sensitivity. Changes take effect immediately.		


Table 39: Description of Auto Exposure Variables (Continued)

Variable Dec (Hex)	Bits	Default	Name
11 0x000B	15:0	0x0000	ae_min_index (R/W)
	Minimum allowed zone number. Changes take effect only after REFRESH command.		
12 0x000C	15:0	0x0018	ae_max_index (R/W)
	Maximum allowed zone number (that is, maximum integration time). Changes take effect only after REFRESH command.		
13 0x000D	15:0	0x0010	ae_min_virt_gain (R/W)
	Minimum allowed virtual gain (control analog gains only). Gain is divided by 32; so a value of 32 corresponds to a gain of 1. Changes take effect only after REFRESH command.		
14 0x000E	15:0	0x0080	ae_max_virt_gain (R/W)
	Maximum allowed virtual gain (control analog gains only). Gain is divided by 32; so a value of 32 corresponds to a gain of 1. Changes take effect only after REFRESH command.		
23 0x0017	15:0	0x0000	ae_status (R/W)
	15:3	X	Reserved.
	2	0x0000	ae_status_ready 0: AE not ready. 1: AE ready.
	1	0x0000	ae_status_r9_changed Indication of R9 register status change, so a frame should be skipped. 0: R9 register has not changed. 1: R9 register has changed; skip a frame.
	0	0x0000	ae_status_ae_at_limit 1: Indicate the gain or integration has reached its limit.
	AE Status.		
24 0x0018	15:0	0x004B	ae_current_y (RO)
	Last measured luminance.		
25 0x0019	15:0	0x0279	ae_r12 (RO)
	Current shutter delay value.		
27 0x001B	15:0	0x0004	ae_index (RO)
	Current zone (integration time).		
28 0x001C	15:0	0x0010	ae_virt_gain (RO)
	Current virtual gain.		
34 0x0022	15:0	0x0010	ae_r9 (R/W)
	Current integration time value.		
45 0x002D	15:0	0x009D	ae_r9_step (R/W)
	Integration time of one zone.		
74 0x004A	15:0	0x0032	ae_target_min (R/W)
	Minimum value for the target.		
75 0x004B	15:0	0x0096	ae_target_max (R/W)
	Maximum value for target.		
76 0x004C	15:0	0x000C	ae_target_buffer_speed (R/W)
	Maximum allowed change. Setting of 32 results in the fastest AE speed while setting of 1 results in the slowest AE speed.		


Table 39: Description of Auto Exposure Variables (Continued)

Variable Dec (Hex)	Bits	Default	Name
79 0x004F	15:0	0x0036	ae_base_target (R/W)
Base target for exposure. This can be used to change the overall brightness.			



Auto White Balance Variables (ID = 3)

Table 40: Auto White Balance Variables Sorted by Name

Variable Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
50 (0x0032)	awb_ccm_0	dddd dddd dddd dddd	442 (0x01BA)
52 (0x0034)	awb_ccm_1	dddd dddd dddd dddd	65371 (0xFF5B)
70 (0x0046)	awb_ccm_10	dddd dddd dddd dddd	43 (0x002B)
54 (0x0036)	awb_ccm_2	dddd dddd dddd dddd	65521 (0xFFF1)
56 (0x0038)	awb_ccm_3	dddd dddd dddd dddd	65479 (0xFFC7)
58 (0x003A)	awb_ccm_4	dddd dddd dddd dddd	441 (0x01B9)
60 (0x003C)	awb_ccm_5	dddd dddd dddd dddd	65415 (0xFF87)
62 (0x003E)	awb_ccm_6	dddd dddd dddd dddd	65529 (0xFFF9)
64 (0x0040)	awb_ccm_7	dddd dddd dddd dddd	65330 (0xFF32)
66 (0x0042)	awb_ccm_8	dddd dddd dddd dddd	476 (0x01DC)
68 (0x0044)	awb_ccm_9	dddd dddd dddd dddd	60 (0x003C)
6 (0x0006)	awb_ccm_l_0	dddd dddd dddd dddd	384 (0x0180)
8 (0x0008)	awb_ccm_l_1	dddd dddd dddd dddd	65280 (0xFF00)
26 (0x001A)	awb_ccm_l_10	dddd dddd dddd dddd	51 (0x0033)
10 (0x000A)	awb_ccm_l_2	dddd dddd dddd dddd	128 (0x0080)
12 (0x000C)	awb_ccm_l_3	dddd dddd dddd dddd	65382 (0xFF66)
14 (0x000E)	awb_ccm_l_4	dddd dddd dddd dddd	384 (0x0180)
16 (0x0010)	awb_ccm_l_5	dddd dddd dddd dddd	65518 (0xFFEE)
18 (0x0012)	awb_ccm_l_6	dddd dddd dddd dddd	65485 (0xFFCD)
20 (0x0014)	awb_ccm_l_7	dddd dddd dddd dddd	65229 (0xFECD)
22 (0x0016)	awb_ccm_l_8	dddd dddd dddd dddd	410 (0x019A)
24 (0x0018)	awb_ccm_l_9	dddd dddd dddd dddd	32 (0x0020)
83 (0x0053)	awb_ccm_position	0000 0000 dddd dddd	64 (0x0040)
82 (0x0052)	awb_ccm_position_max	0000 0000 dddd dddd	127 (0x007F)
81 (0x0051)	awb_ccm_position_min	0000 0000 dddd dddd	0 (0x0000)
28 (0x001C)	awb_ccm_rl_0	dddd dddd dddd dddd	256 (0x0100)
30 (0x001E)	awb_ccm_rl_1	dddd dddd dddd dddd	65434 (0xFF9A)
48 (0x0030)	awb_ccm_rl_10	dddd dddd dddd dddd	65527 (0xFFF7)
32 (0x0020)	awb_ccm_rl_2	dddd dddd dddd dddd	0 (0x0000)
34 (0x0022)	awb_ccm_rl_3	dddd dddd dddd dddd	77 (0x004D)
36 (0x0024)	awb_ccm_rl_4	dddd dddd dddd dddd	65485 (0xFFCD)
38 (0x0026)	awb_ccm_rl_5	dddd dddd dddd dddd	65464 (0xFFB8)
40 (0x0028)	awb_ccm_rl_6	dddd dddd dddd dddd	77 (0x004D)
42 (0x002A)	awb_ccm_rl_7	dddd dddd dddd dddd	128 (0x0080)
44 (0x002C)	awb_ccm_rl_8	dddd dddd dddd dddd	65382 (0xFF66)
46 (0x002E)	awb_ccm_rl_9	dddd dddd dddd dddd	8 (0x0008)
80 (0x0050)	awb_gain_b	0000 0000 dddd dddd	128 (0x0080)
72 (0x0048)	awb_gain_buffer_speed	0000 0000 dddd dddd	8 (0x0008)
88 (0x0058)	awb_gain_b_buf	dddd dddd dddd dddd	0 (0x0000)
79 (0x004F)	awb_gain_g	0000 0000 dddd dddd	128 (0x0080)
77 (0x004D)	awb_gain_max_b	0000 0000 dddd dddd	166 (0x00A6)
75 (0x004B)	awb_gain_max_r	0000 0000 dddd dddd	182 (0x00B6)
76 (0x004C)	awb_gain_min_b	0000 0000 dddd dddd	89 (0x0059)


Table 40: Auto White Balance Variables Sorted by Name (Continued)

Variable Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
74 (0x004A)	awb_gain_min_r	0000 0000 dddd dddd	89 (0x0059)
78 (0x004E)	awb_gain_r	0000 0000 dddd dddd	128 (0x0080)
86 (0x0056)	awb_gain_r_buf	dddd dddd dddd dddd	0 (0x0000)
73 (0x0049)	awb_jump_divisor	0000 0000 dddd dddd	2 (0x0002)
104 (0x0068)	awb_kb_l	0000 0000 dddd dddd	128 (0x0080)
107 (0x006B)	awb_kb_r	0000 0000 dddd dddd	128 (0x0080)
103 (0x0067)	awb_kg_l	0000 0000 dddd dddd	128 (0x0080)
106 (0x006A)	awb_kg_r	0000 0000 dddd dddd	128 (0x0080)
102 (0x0066)	awb_kr_l	0000 0000 dddd dddd	128 (0x0080)
105 (0x0069)	awb_kr_r	0000 0000 dddd dddd	128 (0x0080)
85 (0x0055)	awb_mode	0000 0000 dddd dddd	0 (0x0000)
84 (0x0054)	awb_saturation	0000 0000 dddd dddd	128 (0x0080)
96 (0x0060)	awb_steady_b_gain_in_max	0000 0000 dddd dddd	130 (0x0082)
95 (0x005F)	awb_steady_b_gain_in_min	0000 0000 dddd dddd	126 (0x007E)
94 (0x005E)	awb_steady_b_gain_out_max	0000 0000 dddd dddd	134 (0x0086)
93 (0x005D)	awb_steady_b_gain_out_min	0000 0000 dddd dddd	120 (0x0078)
2 (0x0002)	awb_window_pos	0000 0000 dddd dddd	0 (0x0000)
3 (0x0003)	awb_window_size	0000 0000 dddd dddd	239 (0x00EF)
101 (0x0065)	awb_x0	0000 0000 dddd dddd	16 (0x0010)

Table 41: Description of Auto White Balance Variables

Variable Dec (Hex)	Bits	Default	Name
2 0x0002	15:0	0x0000	awb_window_pos (R/W)
	15:8	X	Reserved.
	7:4	0x0000	awb_window_pos_y0 Y0 in units 1/16 of frame height.
	3:0	0x0000	awb_window_pos_x0 X0 in units of 1/16 of frame width.
	Position of upper left corner of AWB window. New position written to this variable takes effect only after REFRESH command is given to the sequencer (seq.cmd = 5).		
3 0x0003	15:0	0x00EF	awb_window_size (R/W)
	15:8	X	Reserved.
	7:4	0x000E	awb_window_size_height Height (in units of 1/16 of frame height) decremented by 1.
	3:0	0x000F	awb_window_size_width Width (in units of 1/16 of frame width) decremented by 1.
	Size of AWB window. New size written to this variable takes effect only after REFRESH command is given to the sequencer (seq.cmd = 5).		
6 0x0006	15:0	0x0180	awb_ccm_l_0 (R/W)
	Left color correction matrix element K11. Value is encoded in signed two's complement form; 256 means 1.0. Changes take effect immediately.		
8 0x0008	15:0	0xFF00	awb_ccm_l_1 (R/W)
	Left color correction matrix element K12. Changes take effect immediately.		

**Table 41: Description of Auto White Balance Variables (Continued)**

Variable Dec (Hex)	Bits	Default	Name
10 0x000A	15:0	0x0080	awb_ccm_l_2 (R/W)
	Left color correction matrix element K13. Changes take effect immediately.		
12 0x000C	15:0	0xFF66	awb_ccm_l_3 (R/W)
	Left color correction matrix element K21. Changes take effect immediately.		
14 0x000E	15:0	0x0180	awb_ccm_l_4 (R/W)
	Left color correction matrix element K22. Changes take effect immediately.		
16 0x0010	15:0	0xFFEE	awb_ccm_l_5 (R/W)
	Left color correction matrix element K23. Changes take effect immediately.		
18 0x0012	15:0	0xFFCD	awb_ccm_l_6 (R/W)
	Left color correction matrix element K31. Changes take effect immediately.		
20 0x0014	15:0	0xFECD	awb_ccm_l_7 (R/W)
	Left color correction matrix element K32. Changes take effect immediately.		
22 0x0016	15:0	0x019A	awb_ccm_l_8 (R/W)
	Left color correction matrix element K33. Changes take effect immediately.		
24 0x0018	15:0	0x0020	awb_ccm_l_9 (R/W)
	Left color correction matrix red/green gain ratio. Value is interpreted as unsigned; 32 means 1.0. Changes take effect immediately.		
26 0x001A	15:0	0x0033	awb_ccm_l_10 (R/W)
	Left color correction matrix blue/green gain ratio. Value is interpreted as unsigned; 32 means 1.0. Changes take effect immediately.		
28 0x001C	15:0	0x0100	awb_ccm_rl_0 (R/W)
	Delta color correction matrix element D11. Value is encoded in signed two's complement form; 256 means 1.0. Changes take effect immediately.		
30 0x001E	15:0	0xFF9A	awb_ccm_rl_1 (R/W)
	Delta color correction matrix element D12. Changes take effect immediately.		
32 0x0020	15:0	0x0000	awb_ccm_rl_2 (R/W)
	Delta color correction matrix element D13. Changes take effect immediately.		
34 0x0022	15:0	0x004D	awb_ccm_rl_3 (R/W)
	Delta color correction matrix element D21. Changes take effect immediately.		
36 0x0024	15:0	0xFFCD	awb_ccm_rl_4 (R/W)
	Delta color correction matrix element D22. Changes take effect immediately.		
38 0x0026	15:0	0xFFB8	awb_ccm_rl_5 (R/W)
	Delta color correction matrix element D23. Changes take effect immediately.		
40 0x0028	15:0	0x004D	awb_ccm_rl_6 (R/W)
	Delta color correction matrix element D31. Changes take effect immediately.		
42 0x002A	15:0	0x0080	awb_ccm_rl_7 (R/W)
	Delta color correction matrix element D32. Changes take effect immediately.		
44 0x002C	15:0	0xFF66	awb_ccm_rl_8 (R/W)
	Delta color correction matrix element D33. Changes take effect immediately.		

**Table 41: Description of Auto White Balance Variables (Continued)**

Variable Dec (Hex)	Bits	Default	Name
46 0x002E	15:0	0x0008	awb_ccm_rl_9 (R/W)
			Delta color correction matrix red/green gain ratio. Value is interpreted as unsigned; 32 means 1.0. Changes take effect immediately.
48 0x0030	15:0	0xFFFF7	awb_ccm_rl_10 (R/W)
			Delta color correction matrix blue/green gain ratio. Value is interpreted as unsigned; 32 means 1.0. Changes take effect immediately.
50 0x0032	15:0	0x01BA	awb_ccm_0 (R/W)
			Current color correction matrix element C11. Value is stored in signed two's complement form; 256 means 1.0.
52 0x0034	15:0	0xFF5B	awb_ccm_1 (R/W)
			Current color correction matrix element C12.
54 0x0036	15:0	0xFFFF1	awb_ccm_2 (R/W)
			Current color correction matrix element C13.
56 0x0038	15:0	0xFFC7	awb_ccm_3 (R/W)
			Current color correction matrix element C21.
58 0x003A	15:0	0x01B9	awb_ccm_4 (R/W)
			Current color correction matrix element C22.
60 0x003C	15:0	0xFF87	awb_ccm_5 (R/W)
			Current color correction matrix element C23.
62 0x003E	15:0	0xFFFF9	awb_ccm_6 (R/W)
			Current color correction matrix element C31.
64 0x0040	15:0	0xFF32	awb_ccm_7 (R/W)
			Current color correction matrix element C32.
66 0x0042	15:0	0x01DC	awb_ccm_8 (R/W)
			Current color correction matrix element C33.
68 0x0044	15:0	0x003C	awb_ccm_9 (R/W)
			Current color correction matrix red/green gain ratio. Value is interpreted as unsigned; 32 means 1.0.
70 0x0046	15:0	0x002B	awb_ccm_10 (R/W)
			Current color correction matrix blue/green gain ratio. Value is interpreted as unsigned; 32 means 1.0.
72 0x0048	15:0	0x0008	awb_gain_buffer_speed (R/W)
			Speed of AWB digital gains buffering (32=fastest, 1=slowest).
73 0x0049	15:0	0x0002	awb_jump_divisor (R/W)
			Number of steps in which AWB driver is expected to reach color target. The smaller the number, the faster the AWB reaches the color target.
74 0x004A	15:0	0x0059	awb_gain_min_r (R/W)
			Minimum allowed AWB digital Red gain (128 means 1.0) Changes take effect immediately.
75 0x004B	15:0	0x00B6	awb_gain_max_r (R/W)
			Maximum allowed AWB digital Red gain (128 means 1.0). Changes take effect immediately.
76 0x004C	15:0	0x0059	awb_gain_min_b (R/W)
			Minimum allowed AWB digital Blue gain (128 means 1.0). Changes take effect immediately.

**Table 41: Description of Auto White Balance Variables (Continued)**

Variable Dec (Hex)	Bits	Default	Name
77 0x004D	15:0	0x00A6	awb_gain_max_b (R/W)
	Maximum allowed AWB digital Blue gain (128 means 1.0). Changes take effect immediately.		
78 0x004E	15:0	0x0080	awb_gain_r (R/W)
	Current R digital gain (128 means 1.0)		
79	15:0	0x0080	awb_gain_g (R/W)
0x004F	Current G digital gain (128 means 1.0)		
80	15:0	0x0080	awb_gain_b (R/W)
0x0050	Current B digital gain (128 means 1.0)		
81 0x0051	15:0	0x0000	awb_ccm_position_min (R/W)
	Leftmost position of color correction matrix (corresponding to incandescent light). Changes effect immediately.		
82 0x0052	15:0	0x007F	awb_ccm_position_max (R/W)
	Rightmost position of color correction matrix (corresponding to daylight). Changes take effect immediately.		
83 0x0053	15:0	0x0040	awb_ccm_position (R/W)
	Current position of color correction matrix (0 corresponds to incandescent light, 127 to daylight). Changes take effect immediately.		
84 0x0054	15:0	0x0080	awb_saturation (R/W)
	Saturation of color correction matrix (128 means 100%). Changes take effect immediately.		
85 0x0055	15:0	0x0000	awb_mode (R/W)
	15:7	X	Reserved.
	6	0x0000	awb_mode_normccm_off CCM Normalization. 0: Enable CCM normalization. 1: Disable CCM normalization.
	5	0x0000	awb_mode_force_unit_dgains Force AWB digital gains to unity. 0: AWB digital gains are not locked to unity. 1: AWB digital gains are locked to unity; program value into awb.CCMPosition to manually set matrix position.
	4:2	0x0000	Reserved.
	1	0x0000	awb_mode_limits_reached 0: Limit is not reached. 1: AWB limit is reached.
	0	0x0000	awb_mode_steady 0: AWB is not done.1:AWB is done.
	Misc control for AWB		
86 0x0056	15:0	0x0000	awb_gain_r_buf (R/W)
	Time-buffered R gain (0x1000 means 1.0).		
88	15:0	0x0000	awb_gain_b_buf (R/W)
0x0058	Time-buffered B gain (0x1000 means 1.0).		
93 0x005D	15:0	0x0078	awb_steady_b_gain_out_min (R/W)
	Blue gain minimum threshold to start search. When AWB blue digital gain falls below this threshold, the AWB driver starts searching for a new color correction matrix position. A threshold setting of 128 corresponds to gain of 1.0; a higher setting means higher gain. Changes take effect immediately.		

**Table 41: Description of Auto White Balance Variables (Continued)**

Variable Dec (Hex)	Bits	Default	Name
94 0x005E	15:0	0x0086	awb_steady_b_gain_out_max (R/W)
			Blue gain maimum. threshold to start search. When AWB blue digital gain goes above this threshold, the AWB driver starts searching for a new color correction matrix position. A threshold setting of 128 corresponds to gain of 1.0; a higher setting means higher gain. Changes take effect immediately.
95 0x005F	15:0	0x007E	awb_steady_b_gain_in_min (R/W)
			Blue gain minimum threshold to end search. When AWB blue digital gain is between awb.steadyBGainInMin and awb.steadyBGainInMax, the AWB driver maintains the current color correction matrix position. A threshold setting of 128 corresponds to gain of 1.0; a higher setting means higher gain. Changes take effect immediately.
96 0x0060	15:0	0x0082	awb_steady_b_gain_in_max (R/W)
			Blue gain maximum threshold to end search. See awb.steadyBGainInMin above. Changes take effect immediately.
101 0x0065	15:0	0x0010	awb_x0 (R/W)
			CCM position threshold between Day and Incandescent normalization coefficients. Changes take effect immediately.
102 0x0066	15:0	0x0080	awb_kr_l (R/W)
			CCM red row normalization coefficient for left matrix position (128 - minimal number). Changes take effect immediately.
103 0x0067	15:0	0x0080	awb_kg_l (R/W)
			CCM green row normalization coefficient for left matrix position (128 - minimal number). Changes take effect immediately.
104 0x0068	15:0	0x0080	awb_kb_l (R/W)
			CCM blue row normalization coefficient for left matrix position (128 - minimal number). Changes take effect immediately.
105 0x0069	15:0	0x0080	awb_kr_r (R/W)
			CCM red row normalization coefficient for right matrix position (128 - minimal number). Changes take effect immediately.
106 0x006A	15:0	0x0080	awb_kg_r (R/W)
			CCM green row normalization coefficient for right matrix position (128 - minimal number). Changes take effect immediately.
107 0x006B	15:0	0x0080	awb_kb_r (R/W)
			CCM blue row normalization coefficient for right matrix position (128 - minimal number). Changes take effect immediately.

Anti-Flicker Variables(ID = 4)**Table 42: Anti-Flicker Variables Sorted by Name**

Register Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
16 (0x0010)	fd_min_amplitude	0000 0000 dddd dddd	5 (0x0005)
4 (0x0004)	fd_mode	0000 0000 dddd dddd	0 (0x0000)
19 (0x0013)	fd_r9_step_f50_a	dddd dddd dddd dddd	259 (0x0103)
23 (0x0017)	fd_r9_step_f50_b	dddd dddd dddd dddd	184 (0x00B8)

**Table 42: Anti-Flicker Variables Sorted by Name**

Register Dec(Hex)	Name	Data Format (Binary)	Default Value Dec(Hex)
17 (0x0011)	fd_r9_step_f60_a	dddd dddd dddd dddd	212 (0x00D4)
21 (0x0015)	fd_r9_step_f60_b	dddd dddd dddd dddd	157 (0x009D)
8 (0x0008)	fd_search_f1_50	0000 0000 dddd dddd	51 (0x0033)
10 (0x000A)	fd_search_f1_60	0000 0000 dddd dddd	41 (0x0029)
9 (0x0009)	fd_search_f2_50	0000 0000 dddd dddd	53 (0x0035)
11 (0x000B)	fd_search_f2_60	0000 0000 dddd dddd	43 (0x002B)
14 (0x000E)	fd_stat_max	0000 0000 dddd dddd	5 (0x0005)
13 (0x000D)	fd_stat_min	0000 0000 dddd dddd	3 (0x0003)
3 (0x0003)	fd_window_height	0000 0000 dddd dddd	4 (0x0004)
2 (0x0002)	fd_window_posh	0000 0000 dddd dddd	29 (0x001D)

Table 43: Description of Anti-Flicker Variables

Register Dec(Hex)	Bits	Default	Name
2 0x0002	15:0	0x001D	fd_window_posh (R/W)
	15:8	X	Reserved
	7:4	0x0001	fd_width_x0
			Left boundary of the flicker measurement window (in units of 1/16 of frame width).
	3:0	0x000D	fd_width Width (in units of 1/16 of frame width) decremented by 1.
	Width of flicker measurement window and position of its left boundary, X0.		
3 0x0003	15:0	0x0004	fd_window_height (R/W)
	15:6	X	Reserved.
	5:0	0x0004	fd_window_rows Flicker measurement window height in rows.
	Flicker measurement window height.		
4 0x0004	15:0	0x0000	fd_mode (R/W)
	15:8	X	Reserved.
	7	0x0000	fd_mode_manual_mode Manual Flicker Mode Enable. 0: Manual mode disabled. 1: Manual mode enabled.
	6	0x0000	fd_mode_curr_settings Manual Flicker Mode Selection. 0: 60Hz setting 1: 50Hz setting
	5	RO	fd_mode_curr_flicker_state Indicates current flicker detection settings. 0: 60Hz 1: 50Hz
	4	0x0000	Reserved.
	3:0	RO	Reserved.
	Flicker detection switches and indicators.		


Table 43: Description of Anti-Flicker Variables (Continued)

Register Dec(Hex)	Bits	Default	Name
8 0x0008	15:0	0x0033	fd_search_f1_50 (R/W)
	Lower limit of period range of interest in 50Hz flicker detection. If the FD driver is searching for 50Hz flicker and detects in a frame a flicker-like signal whose period in rows is between fl.search_f1_50 and fl.search_f2_50, it counts the frame as one containing flicker.		
9 0x0009	15:0	0x0035	fd_search_f2_50 (R/W)
	Upper limit of period range of interest in 50Hz flicker detection. See fd.search_f1_50 above.		
10 0x000A	15:0	0x0029	fd_search_f1_60 (R/W)
	Lower limit of period range of interest in 60Hz flicker detection. If the FD driver is searching for 60Hz flicker and detects in a frame a flicker-like signal whose period in rows is between fl.search_f1_60 and fl.search_f2_60, it counts the frame as one containing flicker.		
11 0x000B	15:0	0x002B	fd_search_f2_60 (R/W)
	Upper limit of period range of interest in 60Hz flicker detection. See fd.search_f1_60 above.		
13 0x000D	15:0	0x0003	fd_stat_min (R/W)
	Flicker is considered detected if fd.stat_min out of fd.stat_max consecutive frames contain flicker (periodic signal of sought frequency).		
14 0x000E	15:0	0x0005	fd_stat_max (R/W)
	See fd.stat_min.		
16 0x0010	15:0	0x0005	fd_min_amplitude (R/W)
	Signals less than this threshold are ignored.		
17 0x0011	15:0	0x00D4	fd_r9_step_f60_a (R/W)
	Minimal shutter width step for 60Hz AC in context A.		
19 0x0013	15:0	0x0103	fd_r9_step_f50_a (R/W)
	Minimal shutter width step for 50Hz AC in context A		
21 0x0015	15:0	0x009D	fd_r9_step_f60_b (R/W)
	Minimal shutter width step for 60Hz AC in context B.		
23 0x0017	15:0	0x00B8	fd_r9_step_f50_b (R/W)
	Minimal shutter width step for 50Hz AC in context B		

Mode Variables (ID = 7)

Table 44: Mode Variables Sorted by Name

Variable Dec (Hex)	Name	Data Format (Binary)	Default Value Dec (Hex)
99 (0x0063)	mode_common_mode_settings_fx_sepia_settings	dddd dddd dddd dddd	0 (0x0000)
102 (0x0066)	mode_common_mode_settings_test_mode	0000 0000 dddd dddd	0 (0x0000)
57 (0x0039)	mode_crop_x0_a	dddd dddd dddd dddd	0 (0x0000)
71 (0x0047)	mode_crop_x0_b	dddd dddd dddd dddd	0 (0x0000)
59 (0x003B)	mode_crop_x1_a	dddd dddd dddd dddd	799 (0x031F)
73 (0x0049)	mode_crop_x1_b	dddd dddd dddd dddd	1599 (0x063F)
61 (0x003D)	mode_crop_y0_a	dddd dddd dddd dddd	0 (0x0000)
75 (0x004B)	mode_crop_y0_b	dddd dddd dddd dddd	0 (0x0000)
63 (0x003F)	mode_crop_y1_a	dddd dddd dddd dddd	599 (0x0257)

**Table 44: Mode Variables Sorted by Name (Continued)**

Variable Dec (Hex)	Name	Data Format (Binary)	Default Value Dec (Hex)
77 (0x004D)	mode_crop_y1_b	dddd dddd dddd dddd	1199 (0x04AF)
85 (0x0055)	mode_output_format_a	dddd dddd dddd dddd	0 (0x0000)
87 (0x0057)	mode_output_format_b	dddd dddd dddd dddd	0 (0x0000)
5 (0x0005)	mode_output_height_a	dddd dddd dddd dddd	600 (0x0258)
9 (0x0009)	mode_output_height_b	dddd dddd dddd dddd	1200 (0x04B0)
3 (0x0003)	mode_output_width_a	dddd dddd dddd dddd	800 (0x0320)
7 (0x0007)	mode_output_width_b	dddd dddd dddd dddd	1600 (0x0640)
19 (0x0013)	mode_sensor_col_end_a	dddd dddd dddd dddd	1293 (0x050D)
41 (0x0029)	mode_sensor_col_end_b	dddd dddd dddd dddd	1291 (0x050B)
15 (0x000F)	mode_sensor_col_start_a	dddd dddd dddd dddd	0 (0x0000)
37 (0x0025)	mode_sensor_col_start_b	dddd dddd dddd dddd	4 (0x0004)
25 (0x0019)	mode_sensor_fine_correction_a	dddd dddd dddd dddd	123 (0x007B)
47 (0x002F)	mode_sensor_fine_correction_b	dddd dddd dddd dddd	164 (0x00A4)
29 (0x001D)	mode_sensor_fine_it_max_margin_a	dddd dddd dddd dddd	171 (0x00AB)
51 (0x0033)	mode_sensor_fine_it_max_margin_b	dddd dddd dddd dddd	164 (0x00A4)
27 (0x001B)	mode_sensor_fine_it_min_a	dddd dddd dddd dddd	1032 (0x0408)
49 (0x0031)	mode_sensor_fine_it_min_b	dddd dddd dddd dddd	1032 (0x0408)
31 (0x001F)	mode_sensor_frame_length_a	dddd dddd dddd dddd	659 (0x0293)
53 (0x0035)	mode_sensor_frame_length_b	dddd dddd dddd dddd	1261 (0x04ED)
33 (0x0021)	mode_sensor_line_length_pck_a	dddd dddd dddd dddd	2000 (0x07D0)
55 (0x0037)	mode_sensor_line_length_pck_b	dddd dddd dddd dddd	3334 (0x0D06)
23 (0x0017)	mode_sensor_read_mode_a	dddd dddd dddd dddd	1132 (0x046C)
45 (0x002D)	mode_sensor_read_mode_b	dddd dddd dddd dddd	36 (0x0024)
17 (0x0011)	mode_sensor_row_end_a	dddd dddd dddd dddd	1037 (0x040D)
39 (0x0027)	mode_sensor_row_end_b	dddd dddd dddd dddd	1035 (0x040B)
21 (0x0015)	mode_sensor_row_speed_a	0000 0ddd 0ddd 0ddd	274 (0x0112)
43 (0x002B)	mode_sensor_row_speed_b	0000 0ddd 0ddd 0ddd	273 (0x0111)
13 (0x000D)	mode_sensor_row_start_a	dddd dddd dddd dddd	0 (0x0000)
35 (0x0023)	mode_sensor_row_start_b	dddd dddd dddd dddd	4 (0x0004)
89 (0x0059)	mode_spec_effects_a	dddd dddd dddd dddd	25664 (0x6440)
91 (0x005B)	mode_spec_effects_b	dddd dddd dddd dddd	25664 (0x6440)
93 (0x005D)	mode_y_rgb_offset_a	0000 0000 dddd dddd	0 (0x0000)
94 (0x005E)	mode_y_rgb_offset_b	0000 0000 dddd dddd	0 (0x0000)

Table 45: Description of Mode Variables

Variable Dec (Hex)	Bits	Default	Name
3 0x0003	15:0	0x0320	mode_output_width_a (R/W)
	Output size of final image for context A (Preview). Must be equal to or smaller than crop dimension. Changes take effect only after REFRESH command.		
5 0x0005	15:0	0x0258	mode_output_height_a (R/W)
	Output size of final image for context A (Preview). Must be equal to or smaller than crop dimension. Changes take effect only after REFRESH command.		

**Table 45: Description of Mode Variables (Continued)**

Variable Dec (Hex)	Bits	Default	Name
7 0x0007	15:0	0x0640	mode_output_width_b (R/W)
	Output size of final image for context B (Snapshot/Video). Must be equal to or smaller than crop dimension. Changes take effect only after REFRESH command.		
9 0x0009	15:0	0x04B0	mode_output_height_b (R/W)
	Output size of final image for context B (Snapshot/Video). Must be equal to or smaller than crop dimension. Changes take effect only after REFRESH command.		
13 0x000D	15:0	0x0000	mode_sensor_row_start_a (R/W)
	The first row of visible pixels to be read out (not counting any dark rows that may be read) in context A (Preview). To move the image window, set this register to the starting "Y" value. Must be an even value. Changes take effect only after REFRESH_MODE command.		
15 0x000F	15:0	0x0000	mode_sensor_col_start_a (R/W)
	The first column of visible pixels to be read out (not counting any dark columns that may be read) in context A (Preview). To move the image window, set this register to the starting "X" value. Must be an even value. Changes take effect only after REFRESH_MODE command.		
17 0x0011	15:0	0x040D	mode_sensor_row_end_a (R/W)
	The last row of visible pixels to be read out. Must be an odd value in context A (Preview). Changes take effect only after REFRESH_MODE command.		
19 0x0013	15:0	0x050D	mode_sensor_col_end_a (R/W)
	The last column of visible pixels to be read out in context A (Preview). Must be an odd value. Changes take effect only after REFRESH_MODE command.		
21 0x0015	15:0	0x0112	mode_sensor_row_speed_a (R/W)
	15:11	X	Reserved.
	10:8	0x0001	Reserved.
	7	X	Reserved.
	6:4	0x0001	Reserved.
	3	X	Reserved.
	2:0	0x0002	mode_sensor_row_speed_a_pix_clk_speed A programmed value of N gives a pixel clock period of N system clocks. A value of 0 is illegal: it causes the clock to stop.
Context A (Preview) row speed. Changes take effect only after REFRESH_MODE command.			

**Table 45: Description of Mode Variables (Continued)**

Variable Dec (Hex)	Bits	Default	Name
23 0x0017	15:0	0x046C	mode_sensor_read_mode_a (R/W)
	15:12	X	Reserved.
	11	0x0000	mode_sensor_read_mode_a_x_bin_en Enable analog binning in X (column) direction. When set, x_odd_inc must be set to 3 and y_odd_inc must be set to 1.
	10	0x0001	mode_sensor_read_mode_a_xy_bin_en Enable analog binning in X and Y (column and row) directions. When set, x_odd_inc and y_odd_inc must be set to 3.
	9	0x0000	mode_sensor_read_mode_a_low_power_enable Low Power Mode Enable. 0: Low power mode disabled. 1: Low power mode enabled.
	8	X	Reserved.
	7:5	0x0003	mode_sensor_read_mode_a_x_odd_inc Increment applied to odd addresses in X (column) direction. 0: Reserved. 1: Normal readout. 2: Reserved. 3: Read out alternate pixel pairs to halve the amount of horizontal data in a frame.
	4:2	0x0003	mode_sensor_read_mode_a_y_odd_inc Increment applied to odd addresses in Y (row) direction. 0: Reserved. 1: Normal readout. 2: Reserved. 3: Read out alternate pixel pairs to halve the amount of vertical data in a frame.
	1	0x0000	mode_sensor_read_mode_a_vert_flip 0: Normal readout. 1: Readout is flipped (mirrored) vertically so that the row specified by sensor_row_end_a is read out of the sensor first.
	0	0x0000	mode_sensor_read_mode_a_horiz_mirror 0: Normal readout. 1: Readout is mirrored horizontally so that the column specified by sensor_col_end_a is read out of the sensor first.
Context A (Preview) Read Mode register. Changes take effect only after REFRESH_MODE command.			
25 0x0019	15:0	0x007B	mode_sensor_fine_correction_a (R/W)
	Context A (Preview) fine integration time correction factor. This is an offset that is applied to the programmed value of fine_integration_time such that the actual integration time matches the integration time equation. Changes take effect only after REFRESH_MODE command.		
27 0x001B	15:0	0x0408	mode_sensor_fine_it_min_a (R/W)
	Context A (Preview) minimal fine integration time. Changes take effect only after REFRESH_MODE command.		
29 0x001D	15:0	0x00AB	mode_sensor_fine_it_max_margin_a (R/W)
	Context A (Preview) maximal fine integration time margin. Changes take effect only after REFRESH_MODE command.		
31 0x001F	15:0	0x0293	mode_sensor_frame_length_a (R/W)
	The number of complete lines (rows) in the output frame in context A (Preview). This includes visible lines and vertical blanking lines. Changes take effect only after REFRESH_MODE command.		

**Table 45: Description of Mode Variables (Continued)**

Variable Dec (Hex)	Bits	Default	Name
33 0x0021	15:0	0x07D0	mode_sensor_line_length_pck_a (R/W)
	The number of pixel clock periods in one line (row) time in context A (Preview). This includes visible pixels and horizontal blanking time. Changes take effect only after REFRESH_MODE command.		
35 0x0023	15:0	0x0004	mode_sensor_row_start_b (R/W)
	The first row of visible pixels to be read out (not counting any dark rows that may be read) in context B (Snapshot/Video). To move the image window, set this register to the starting "Y" value. Must be an even value. Changes take effect only after REFRESH_MODE command.		
37 0x0025	15:0	0x0004	mode_sensor_col_start_b (R/W)
	The first column of visible pixels to be read out (not counting any dark columns that may be read) in context B (Snapshot/Video). To move the image window, set this register to the starting "X" value. Must be an even value. Changes take effect only after REFRESH_MODE command.		
39 0x0027	15:0	0x040B	mode_sensor_row_end_b (R/W)
	The last row of visible pixels to be read out. Must be an odd value in context B (Snapshot/Video). Changes take effect only after REFRESH_MODE command.		
41 0x0029	15:0	0x050B	mode_sensor_col_end_b (R/W)
	The last column of visible pixels to be read out in context B (Snapshot/Video). Must be an odd value. Changes take effect only after REFRESH_MODE command.		
43 0x002B	15:0	0x0111	mode_sensor_row_speed_b (R/W)
	15:11	X	Reserved.
	10:8	0x0001	Reserved.
	7	X	Reserved.
	6:4	0x0001	Reserved.
	3	X	Reserved.
	2:0	0x0001	mode_sensor_row_speed_b_pix_clk_speed A programmed value of N gives a pixel clock period of N system clocks. A value of 0 is illegal: it causes the clock to stop.
	Context B (Snapshot/Video) row speed. Changes take effect only after REFRESH_MODE command.		

**Table 45: Description of Mode Variables (Continued)**

Variable Dec (Hex)	Bits	Default	Name
45 0x002D	15:0	0x0024	mode_sensor_read_mode_b (R/W)
	15:12	X	Reserved.
	11	0x0000	mode_sensor_read_mode_b_x_bin_en Enable analog binning in X (column) direction. When set, x_odd_inc must be set to 3 and y_odd_inc must be set to 1.
	10	0x0000	mode_sensor_read_mode_b_xy_bin_en Enable analog binning in X and Y (column and row) directions. When set, x_odd_inc and y_odd_inc must be set to 3.
	9	0x0000	mode_sensor_read_mode_b_low_power_mode Low Power Mode Enable. 0: Low power mode disabled. 1: Low power mode enabled.
	8	X	Reserved
	7:5	0x0001	mode_sensor_read_mode_b_x_odd_inc Increment applied to odd addresses in X (column) direction. 1: Normal readout. 3: Read out alternate pixel pairs to halve the amount of horizontal data in a frame.
	4:2	0x0001	mode_sensor_read_mode_b_y_odd_inc Increment applied to odd addresses in Y (row) direction. 1: Normal readout. 3: Read out alternate pixel pairs to halve the amount of vertical data in a frame.
	1	0x0000	mode_sensor_read_mode_b_vert_flip 0: Normal readout. 1: Readout is flipped (mirrored) vertically so that the row specified by sensor_row_end_b is read out of the sensor first.
	0	0x0000	mode_sensor_read_mode_b_horiz_mirror 0: Normal readout. 1: Readout is mirrored horizontally so that the column specified by sensor_col_end_b is read out of the sensor first.
Context B (Snapshot/Video) Read Mode register. Changes take effect only after REFRESH_MODE command.			
47 0x002F	15:0	0x00A4	mode_sensor_fine_correction_b (R/W)
	Context B (Snapshot/Video) fine integration time correction factor. This is an offset that is applied to the programmed value of fine_integration_time such that the actual integration time matches the integration time equation. Changes take effect only after REFRESH_MODE command.		
49 0x0031	15:0	0x0408	mode_sensor_fine_it_min_b (R/W)
	Context B (Snapshot/Video) minimal fine integration time. Changes take effect only after REFRESH_MODE command.		
51 0x0033	15:0	0x00A4	mode_sensor_fine_it_max_margin_b (R/W)
	Context B (Snapshot/Video) maximal fine integration time margin. Changes take effect only after REFRESH_MODE command.		
53 0x0035	15:0	0x04ED	mode_sensor_frame_length_b (R/W)
	The number of complete lines (rows) in the output frame in context B (Snapshot/Video). This includes visible lines and vertical blanking lines. Changes take effect only after REFRESH_MODE command.		
55 0x0037	15:0	0x0D06	mode_sensor_line_length_pck_b (R/W)
	The number of pixel clock periods in one line (row) time in context B (Snapshot/Video). This includes visible pixels and horizontal blanking time. Changes take effect only after REFRESH_MODE command.		


Table 45: Description of Mode Variables (Continued)

Variable Dec (Hex)	Bits	Default	Name
57 0x0039	15:0	0x0000	mode_crop_x0_a (R/W)
	Lower-x decimator zoom window for context A (Preview). Changes take effect only after REFRESH command.		
59 0x003B	15:0	0x031F	mode_crop_x1_a (R/W)
	Upper-x decimator zoom window for context A (Preview). Changes take effect only after REFRESH command.		
61 0x003D	15:0	0x0000	mode_crop_y0_a (R/W)
	Lower-y decimator zoom window for context A (Preview). Changes take effect only after REFRESH command.		
63 0x003F	15:0	0x0257	mode_crop_y1_a (R/W)
	Upper-y decimator zoom window for context A (Preview). Changes take effect only after REFRESH command.		
71 0x0047	15:0	0x0000	mode_crop_x0_b (R/W)
	Lower-x decimator zoom window for context B (Snapshot/Video). Changes take effect only after REFRESH command.		
73 0x0049	15:0	0x063F	mode_crop_x1_b (R/W)
	Upper-x decimator zoom window for context B (Snapshot/Video). Changes take effect only after REFRESH command.		
75 0x004B	15:0	0x0000	mode_crop_y0_b (R/W)
	Lower-y decimator zoom window for context B (Snapshot/Video). Changes take effect only after REFRESH command.		
77 0x004D	15:0	0x04AF	mode_crop_y1_b (R/W)
	Upper-y decimator zoom window for context B (Snapshot/Video). Changes take effect only after REFRESH command.		


Table 45: Description of Mode Variables (Continued)

Variable Dec (Hex)	Bits	Default	Name
85 0x0055	15:0	0x0000	mode_output_format_a (R/W)
	15:9	X	Reserved
	8	0x0000	mode_output_format_a_config_processed_bayer Turn on processed Bayer output. Chip output data in Bayer format. As a result, data rate decreases by two times.
	7:6	0x0000	mode_output_format_a_rgb_format RGB output format: 0: 16-bit 565RGB 1: 15-bit 555RGB 2: 12-bit 444xRGB 3: 12-bit x444RGB
	5	0x0000	mode_output_format_a_output_mode RGB/YUV Output. 0: Output YUV. 1: Output RGB (see mode_output_format_a_output_mode[7:6] for detail).
	4	0x0000	Reserved
	3	0x0000	mode_output_format_a_monochrome This bit forces the output to produce a monochrome output. 0: Monochrome output disabled. 1: Monochrome output enabled.
	2	0x0000	Reserved
	1	0x0000	mode_output_format_a_swap_chrominance_luma Chrominance swap. This bit is subject to synchronous update. 0: Normal mode. 1: In YUV mode, swaps chrominance byte with luminance byte. In RGB mode, swaps odd and even bytes.
	0	0x0000	mode_output_format_a_swap_channels Channel swap for YUV and RGB output. This bit is subject to synchronous update. 0: Normal mode. 1: In YUV output mode, swaps Cb and Cr channels. In RGB mode, swaps R and B.
	Output format configuration for context A (Preview). Changes take effect only after REFRESH command.		


Table 45: Description of Mode Variables (Continued)

Variable Dec (Hex)	Bits	Default	Name
87 0x0057	15:0	0x0000	mode_output_format_b (R/W)
	15:9	X	Reserved
	8	0x0000	mode_output_format_b_config_processed_bayer Turn on processed Bayer output. Chip output data in Bayer format. As a result, data rate decreases by two times.
	7:6	0x0000	mode_output_format_b_rgb_format RGB output format: 0: 16-bit 565RGB. 1: 15-bit 555RGB. 2: 12-bit 444xRGB. 3: 12-bit x444RGB.
	5	0x0000	mode_output_format_b_output_mode RGB/YUV Output. 0: Output YUV. 1: Output RGB (see mode_output_format_a_output_mode[7:6] for detail).
	4	0x0000	Reserved.
	3	0x0000	mode_output_format_b_monochrome This bit forces the output to produce a monochrome output. 0: Monochrome output disabled. 1: Monochrome output enabled.
	2	0x0000	Reserved.
	1	0x0000	mode_output_format_b_swap_chrominance_luma Chrominance swap. This bit is subject to synchronous update. 0: Normal mode. 1: In YUV mode, swaps chrominance byte with luminance byte. In RGB mode, swaps odd and even bytes.
	0	0x0000	mode_output_format_b_swap_channels Channel swap for YUV and RGB output. This bit is subject to synchronous update. 0: Normal mode. 1: In YUV output mode, swaps Cb and Cr channels. In RGB mode, swaps R and B.
Output format configuration for context B (Snapshot/Video). Changes take effect only after REFRESH command.			

**Table 45: Description of Mode Variables (Continued)**

Variable Dec (Hex)	Bits	Default	Name
89 0x0059	15:0	0x6440	mode_spec_effects_a (R/W)
	15:8	0x0064	mode_spec_effects_a_solarization_thresh Threshold to create different solarization effect.
	7	X	Reserved.
	6	0x0001	mode_spec_effects_a_dither_luma Dither Selection. 0: Dither in all color channels. 1: Dither only in luma channel.
	5:3	0x0000	mode_spec_effects_a_dither_bit_width Bit width of dither (valid values 1, 2, 3, and 4; no dither if value = 0, 5, 6, or 7).
	2:0	0x0000	mode_spec_effects_a_selection 0: Disabled. 1: Monochrome. 2: Sepia. 3: Negative. 4: Solarization with unmodified UV. 5: Solarization with -UV.
	Special effects selection for context A (Preview). Changes take effect only after REFRESH command.		
91 0x005B	15:0	0x6440	mode_spec_effects_b (R/W)
	15:8	0x0064	mode_spec_effects_b_solarization_thresh Threshold to create different solarization effect.
	7	X	Reserved.
	6	0x0001	mode_spec_effects_b_dither_luma Dither Selection. 0: Dither in all color channels. 1: Dither only in luma channel
	5:3	0x0000	mode_spec_effects_b_dither_bit_width Bit width of dither (valid values 1, 2, 3, and 4; no dither if value = 0, 5, 6, or 7).
	2:0	0x0000	mode_spec_effects_b_selection 0: Disabled. 1: Monochrome. 2: Sepia. 3: Negative. 4: Solarization with unmodified UV. 5: Solarization with -UV.
	Special effects selection for context B (Snapshot/Video). Changes take effect only after REFRESH command.		
93 0x005D	15:0	0x0000	mode_y_rgb_offset_a (R/W)
	Y/RGB Offset setting for context A (Preview). Changes take effect only after REFRESH command.		
94 0x005E	15:0	0x0000	mode_y_rgb_offset_b (R/W)
	Y/RGB Offset setting for context B (Snapshot/Video). Changes take effect only after REFRESH command.		

**Table 45: Description of Mode Variables (Continued)**

Variable Dec (Hex)	Bits	Default	Name
99 0x0063	15:0	0x0000	mode_common_mode_settings_fx_sepia_settings (R/W)
	15:8	0x0000	mode_sepia_cb Magnitude of Cb in 0.7 fixed point.
	7:0	0x0000	mode_sepia_cr Magnitude of Cr in 0.7 fixed point.
	This register specifies the chroma values for the sepia effect. In sepia mode, the chroma values of each pixel are set to this value. By default, this register contains a brownish color, but it can be set to an arbitrary color. Changes take effect only after REFRESH command.		
102 0x0066	15:0	0x0000	mode_common_mode_settings_test_mode (R/W)
	Test Pattern Generator. Changes take effect only after REFRESH command. 0: Disabled. 1: Solid white. 2: Grey ramp. 3: Color bar ramp. 4: Solid white (color bars). 5: Noise.		

Histogram Variables (ID = 11)**Table 46: Histogram Variables Sorted by Name**

Variable Dec (Hex)	Name	Data Format (Binary)	Default Value Dec (Hex)
6 (0x0006)	hg_black_clip_percent	0000 0000 dddd dddd	3 (0x0003)
27 (0x001B)	hg_brightness_metric	dddd dddd dddd dddd	0 (0x0000)
8 (0x0008)	hg_dark_level	0000 0000 dddd dddd	16 (0x0010)
55 (0x0037)	hg_gamma_morph_ctrl	0000 0000 dddd dddd	3 (0x0003)
56 (0x0038)	hg_gamma_start_morph	dddd dddd dddd dddd	2700 (0x0A8C)
58 (0x003A)	hg_gamma_stop_morph	dddd dddd dddd dddd	13500 (0x34BC)
60 (0x003C)	hg_gamma_table_a_0	0000 0000 dddd dddd	0 (0x0000)
61 (0x003D)	hg_gamma_table_a_1	0000 0000 dddd dddd	27 (0x001B)
70 (0x0046)	hg_gamma_table_a_10	0000 0000 dddd dddd	225 (0x00E1)
71 (0x0047)	hg_gamma_table_a_11	0000 0000 dddd dddd	232 (0x00E8)
72 (0x0048)	hg_gamma_table_a_12	0000 0000 dddd dddd	238 (0x00EE)
73 (0x0049)	hg_gamma_table_a_13	0000 0000 dddd dddd	242 (0x00F2)
74 (0x004A)	hg_gamma_table_a_14	0000 0000 dddd dddd	246 (0x00F6)
75 (0x004B)	hg_gamma_table_a_15	0000 0000 dddd dddd	249 (0x00F9)
76 (0x004C)	hg_gamma_table_a_16	0000 0000 dddd dddd	251 (0x00FB)
77 (0x004D)	hg_gamma_table_a_17	0000 0000 dddd dddd	253 (0x00FD)
78 (0x004E)	hg_gamma_table_a_18	0000 0000 dddd dddd	255 (0x00FF)
62 (0x003E)	hg_gamma_table_a_2	0000 0000 dddd dddd	46 (0x002E)
63 (0x003F)	hg_gamma_table_a_3	0000 0000 dddd dddd	76 (0x004C)
64 (0x0040)	hg_gamma_table_a_4	0000 0000 dddd dddd	120 (0x0078)
65 (0x0041)	hg_gamma_table_a_5	0000 0000 dddd dddd	152 (0x0098)
66 (0x0042)	hg_gamma_table_a_6	0000 0000 dddd dddd	176 (0x00B0)

**Table 46: Histogram Variables Sorted by Name (Continued)**

Variable Dec (Hex)	Name	Data Format (Binary)	Default Value Dec (Hex)
67 (0x0043)	hg_gamma_table_a_7	0000 0000 dddd dddd	232 (0x00E8)
68 (0x0044)	hg_gamma_table_a_8	0000 0000 dddd dddd	207 (0x00CF)
69 (0x0045)	hg_gamma_table_a_9	0000 0000 dddd dddd	217 (0x00D9)
79 (0x004F)	hg_gamma_table_b_0	0000 0000 dddd dddd	0 (0x0000)
80 (0x0050)	hg_gamma_table_b_1	0000 0000 dddd dddd	15 (0x000F)
89 (0x0059)	hg_gamma_table_b_10	0000 0000 dddd dddd	187 (0x00BB)
90 (0x005A)	hg_gamma_table_b_11	0000 0000 dddd dddd	198 (0x00C6)
91 (0x005B)	hg_gamma_table_b_12	0000 0000 dddd dddd	208 (0x00D0)
92 (0x005C)	hg_gamma_table_b_13	0000 0000 dddd dddd	217 (0x00D9)
93 (0x005D)	hg_gamma_table_b_14	0000 0000 dddd dddd	226 (0x00E2)
94 (0x005E)	hg_gamma_table_b_15	0000 0000 dddd dddd	234 (0x00EA)
95 (0x005F)	hg_gamma_table_b_16	0000 0000 dddd dddd	241 (0x00F1)
96 (0x0060)	hg_gamma_table_b_17	0000 0000 dddd dddd	249 (0x00F9)
97 (0x0061)	hg_gamma_table_b_18	0000 0000 dddd dddd	255 (0x00FF)
81 (0x0051)	hg_gamma_table_b_2	0000 0000 dddd dddd	26 (0x001A)
82 (0x0052)	hg_gamma_table_b_3	0000 0000 dddd dddd	46 (0x002E)
83 (0x0053)	hg_gamma_table_b_4	0000 0000 dddd dddd	80 (0x0050)
84 (0x0054)	hg_gamma_table_b_5	0000 0000 dddd dddd	106 (0x006A)
85 (0x0055)	hg_gamma_table_b_6	0000 0000 dddd dddd	128 (0x0080)
86 (0x0056)	hg_gamma_table_b_7	0000 0000 dddd dddd	145 (0x0091)
87 (0x0057)	hg_gamma_table_b_8	0000 0000 dddd dddd	161 (0x00A1)
88 (0x0058)	hg_gamma_table_b_9	0000 0000 dddd dddd	175 (0x00AF)
31 (0x001F)	hg_llmode	0000 0000 d000 dddd	132 (0x0084)
40 (0x0028)	hg_ll_brightness_start	dddd dddd dddd dddd	2700 (0x0A8C)
42 (0x002A)	hg_ll_brightness_stop	dddd dddd dddd dddd	13500 (0x34BC)
4 (0x0004)	hg_max_dark_level	0000 0000 dddd dddd	64 (0x0040)

Table 47: Description of Histogram Variables

Variable Dec(Hex)	Bits	Default	Name
4 0x0004	15:0	0x0040	hg_max_dark_level (R/W)
	Limits DarkLevel variable.		
6 0x0006	15:0	0x0003	hg_black_clip_percent (R/W)
	Percentage of Black Pixels the Dark-Level Subtraction algorithm will try to clip, represented as 10 times the percentage to be clipped, such that a value of 10 means 1 percent and 3 means 0.3 percent.		
8 0x0008	15:0	0x0010	hg_dark_level (R/W)
	Makes image darker or brighter based on target black clip pixels.		
27 0x001B	15:0	0x0000	hg_brightness_metric (R/W)
	A measure of the brightness of a scene. It increases in value as the scene becomes darker. This measure does not correspond to any lux level.		

**Table 47: Description of Histogram Variables (Continued)**

Variable Dec(Hex)	Bits	Default	Name
31 0x001F	15:0	0x0084	hg_llmode (R/W)
	15:8	X	Reserved.
	7	0x0001	Reserved.
	6:4	X	Reserved.
	3:0	0x0004	hg_bm_prescale Brightness Metric Prescaler.
Low Light mode controls.			
40 0x0028	15:0	0x0A8C	hg_ll_brightness_start (R/W)
	Low light settings start based on hg_brightness_metric (0x1B). When hg_brightness_metric < hg_ll_brightness_start, low light settings are not applied. When hg_brightness_metric ≥ hg_ll_brightness_start, interpolated low light settings are applied. Note: hg_brightness_metric increases in value as the scene appears darker.		
	15:0	0x34BC	hg_ll_brightness_stop (R/W)
	Low light settings start based on hg_brightness_metric (0x1B). When hg_brightness_metric > hg_ll_brightness_stop, low light settings are applied. When hg_brightness_metric ≤ hg_ll_brightness_stop, interpolated low light settings are applied. Note: hg_brightness_metric increases in value as the scene appears darker.		
55 0x0037	15:0	0x0003	hg_gamma_morph_ctrl (R/W)
	15:2	X	Reserved
	1:0	0x0003	hg_enable_gamma_morph 0: Disable Gamma morph. 1: Use Table A. 2: Use Table B. 3: Auto-Morph between Table A and B based on brightness metric.
	Gamma Morphing Control		
56 0x0038	15:0	0x0A8C	hg_gamma_start_morph (R/W)
	Starting point for Gamma morphing. When the hg_brightness_metric (0x1B) is < hg_gamma_start_morph, Gamma table A is used. When the hg_brightness_metric is ≥ hg_gamma_start_morph, a interpolation of Gamma tables A and B is used. Note: hg_brightness_metric increases in value as the scene appears darker.		
58 0x003A	15:0	0x34BC	hg_gamma_stop_morph (R/W)
	Ending point for Gamma morphing. When the hg_brightness_metric (0x1B) is > hg_gamma_stop_morph, Gamma table B is used. When the hg_brightness_metric is ≤ hg_gamma_stop_morph, a interpolation of Gamma tables A and B is used. Note: hg_brightness_metric increases in value as the scene appears darker.		
60 0x003C	15:0	0x0000	hg_gamma_table_a_0 (R/W)
	Gamma Table A entry for normal light condition.		
61 0x003D	15:0	0x001B	hg_gamma_table_a_1 (R/W)
	Gamma Table A entry for normal light condition.		
62 0x003E	15:0	0x002E	hg_gamma_table_a_2 (R/W)
	Gamma Table A entry for normal light condition.		
63 0x003F	15:0	0x004C	hg_gamma_table_a_3 (R/W)
	Gamma Table A entry for normal light condition.		
64 0x0040	15:0	0x0078	hg_gamma_table_a_4 (R/W)
	Gamma Table A entry for normal light condition.		
65 0x0041	15:0	0x0098	hg_gamma_table_a_5 (R/W)
	Gamma Table A entry for normal light condition.		

**Table 47: Description of Histogram Variables (Continued)**

Variable Dec(Hex)	Bits	Default	Name
66 0x0042	15:0	0x00B0	hg_gamma_table_a_6 (R/W)
			Gamma Table A entry for normal light condition.
67 0x0043	15:0	0x00E8	hg_gamma_table_a_7 (R/W)
			Gamma Table A entry for normal light condition.
68 0x0044	15:0	0x00CF	hg_gamma_table_a_8 (R/W)
			Gamma Table A entry for normal light condition.
69 0x0045	15:0	0x00D9	hg_gamma_table_a_9 (R/W)
			Gamma Table A entry for normal light condition.
70 0x0046	15:0	0x00E1	hg_gamma_table_a_10 (R/W)
			Gamma Table A entry for normal light condition.
71 0x0047	15:0	0x00E8	hg_gamma_table_a_11 (R/W)
			Gamma Table A entry for normal light condition.
72 0x0048	15:0	0x00EE	hg_gamma_table_a_12 (R/W)
			Gamma Table A entry for normal light condition.
73 0x0049	15:0	0x00F2	hg_gamma_table_a_13 (R/W)
			Gamma Table A entry for normal light condition.
74 0x004A	15:0	0x00F6	hg_gamma_table_a_14 (R/W)
			Gamma Table A entry for normal light condition.
75 0x004B	15:0	0x00F9	hg_gamma_table_a_15 (R/W)
			Gamma Table A entry for normal light condition.
76 0x004C	15:0	0x00FB	hg_gamma_table_a_16 (R/W)
			Gamma Table A entry for normal light condition.
77 0x004D	15:0	0x00FD	hg_gamma_table_a_17 (R/W)
			Gamma Table A entry for normal light condition.
78 0x004E	15:0	0x00FF	hg_gamma_table_a_18 (R/W)
			Gamma Table A entry for normal light condition.
79 0x004F	15:0	0x0000	hg_gamma_table_b_0 (R/W)
			Gamma Table B entry for low light condition.
80 0x0050	15:0	0x000F	hg_gamma_table_b_1 (R/W)
			Gamma Table B entry for low light condition.
81 0x0051	15:0	0x001A	hg_gamma_table_b_2 (R/W)
			Gamma Table B entry for low light condition.
82 0x0052	15:0	0x002E	hg_gamma_table_b_3 (R/W)
			Gamma Table B entry for low light condition.
83 0x0053	15:0	0x0050	hg_gamma_table_b_4 (R/W)
			Gamma Table B entry for low light condition.
84 0x0054	15:0	0x006A	hg_gamma_table_b_5 (R/W)
			Gamma Table B entry for low light condition.
85 0x0055	15:0	0x0080	hg_gamma_table_b_6 (R/W)
			Gamma Table B entry for low light condition.
86 0x0056	15:0	0x0091	hg_gamma_table_b_7 (R/W)
			Gamma Table B entry for low light condition.
87 0x0057	15:0	0x00A1	hg_gamma_table_b_8 (R/W)
			Gamma Table B entry for low light condition.


Table 47: Description of Histogram Variables (Continued)

Variable Dec(Hex)	Bits	Default	Name
88 0x0058	15:0	0x00AF	hg_gamma_table_b_9 (R/W)
			Gamma Table B entry for low light condition.
89 0x0059	15:0	0x00BB	hg_gamma_table_b_10 (R/W)
			Gamma Table B entry for low light condition.
90 0x005A	15:0	0x00C6	hg_gamma_table_b_11 (R/W)
			Gamma Table B entry for low light condition.
91 0x005B	15:0	0x00D0	hg_gamma_table_b_12 (R/W)
			Gamma Table B entry for low light condition.
92 0x005C	15:0	0x00D9	hg_gamma_table_b_13 (R/W)
			Gamma Table B entry for low light condition.
93 0x005D	15:0	0x00E2	hg_gamma_table_b_14 (R/W)
			Gamma Table B entry for low light condition.
94 0x005E	15:0	0x00EA	hg_gamma_table_b_15 (R/W)
			Gamma Table B entry for low light condition.
95 0x005F	15:0	0x00F1	hg_gamma_table_b_16 (R/W)
			Gamma Table B entry for low light condition.
96 0x0060	15:0	0x00F9	hg_gamma_table_b_17 (R/W)
			Gamma Table B entry for low light condition.
97 0x0061	15:0	0x00FF	hg_gamma_table_b_18 (R/W)
			Gamma Table B entry for low light condition.



3080 North 1st Street, San Jose, CA 95134, Tel: 408-834-1200 prodmtg@aptina.com www.aptina.com
Aptina, Aptina Imaging, DigitalClarity, Osmium, and the Aptina logo are the property of Micron Technology, Inc.

All other trademarks are the property of their respective owners.

Advance: This data sheet contains initial descriptions of products still under development.



Revision History

Rev. G	4/23/2008
- Added “Power-On Reset” on page 42 - Added “Hard Standby Mode” on page 47 - Added “GPIO Control” on page 52 - Updated Table 1, “Key Performance Parameters,” on page 1 to show the specific I/O voltages that the sensor can be run at (removed range) - Used VDDIO_TX throughout the document - Updated Table 3, “Signal Description and Direction,” on page 8 and notes - Updated Table 5, “Two-Wire Serial Control Bus Timing Parameters,” on page 12 - Added “Power Modes” on page 41 - Updated Table 13, “Output Interface Timing,” on page 32 - Added Figure 24: “Output Interface Timing Waveforms,” on page 32 - Updated Table 20, “Maximum Rating,” on page 53 - Updated Table 21, “I/O Parameters,” on page 54 - Updated Table 22, “AC Electricals,” on page 54 - Updated Table 23, “DC Electricals,” on page 55	
Rev. F	2/5/2008
- Deleted Note 11 on p. 8 (renumbering 12 to 11) from Table 3, Signal Description and Direction. - Updated Note 11 (formerly 12) from Table 3, Signal Description and Direction	
Rev. E	1/16/2008
- Changed name of R0x3290[6] to “walk_ones_10” and R0x3290[5] to “en_walk_ones_tp” on page 70. - Moved “Histogram: Dark Level Adjustments, Low Light and Tonal Controls” from Control Functions to Camera Functions. - Moved “^fVCO” to Symbol column and “Working VCO clock frequency” to Parameters column in Table 20 on page 53.	
Rev. D	1/10/2008
- Updated Equation 1 on page 16. - Added working ^fVCO frequency to Table 20 on page 53.	
Rev. C	11/15/2007
- Added second sentence to Note 8 of Figure 1 on page 6. - Removed Bonding Pad Location table because it is duplicated in the die data sheet. - Edited first sentence of first paragraph in “External Host Interface (Two-Wire Slave-Only Interface)” on page 11. - Added MIN and MAX for Temp_op in Table 20, “Maximum Rating,” on page 53. Removed rows whose information is duplicated in Table 23, “DC Electricals,” on page 55. - Added MIN for V_{IH}, MAX for V_{IL}, MAX for I_{OH}, AND MAX for I_{OL} in Table 21, “I/O Parameters,” on page 54. - Updated Table 23, “DC Electricals,” on page 55 with MIN, MAX, and TYP values for VDD, VDD_PLL, VAA, VAA_PIX, VDD_IO, and VDDIO_TX.	
Rev. B	11/2/2007
Added equation for PLL frequency (Equation 1 on page 16).	



- Updated Figure 20: "Available Test Patterns," on page 24.
- Updated Figure 21: "IFP Block Diagram," on page 25.
- Updated Table 12, "Parallel Image Bus Timing Parameters," on page 32.
- Added Table 13, "Output Interface Timing," on page 32.
- Updated Figure 25: "Sequencer Finite State Machine," on page 35.
- Added "Mode: Context Information" on page 35.
- Added "To control the output image size, the user can modify the mode driver variables such as output_width_A, output_height_A, output_width_B, and output_height_B. The mode driver will automatically apply any appropriate down-scaling filter to achieve this output image size as well as updating the watermark of the output FIFO. It is important to set up the sensor core to output an image equal to or larger than the crop window size, which in turn is equal to or larger than the desired output image size." on page 36.
- Updated Figure 26: "Chief Ray Angle (CRA) vs. Image Height," on page 40.
- Added "Appendix A - Registers" on page 57 and "Appendix B - Driver Variables" on page 85.

Rev. A 8/3/2007

- Initial release.