



1/4.5-Inch 1.6Mp CMOS Digital Image Sensor

MT9M003 Data Sheet

For the latest MT9M003 data sheet, refer to Aptina's Web site: www.aplina.com

Features

- DigitalClarity[®] CMOS imaging technology
- Maximum frame rate (1284H x 812V/60 fps at 99 MHz)
- Superior low-light performance
- Low dark current
- Global reset release (GRR), which starts the exposure of all rows simultaneously
- Simple two-wire serial interface
- Programmable controls: gain, frame rate, frame size, exposure
- Horizontal and vertical mirror image
- Automatic black level calibration
- On-chip phase-locked loop (PLL) oscillator
- Bulb exposure mode for arbitrary exposure times
- Snapshot mode to take frames on demand
- Parallel data output
- Electronic rolling shutter (ERS), progressive scan
- Programmable power-down mode (mode A or mode B)
- Xenon and LED flash support with fast exposure adaptation
- Flexible support for mechanical shutter

Applications

- 720p high-definition digital video camcorder
- Solid state (flash) pocket DVC
- Digital still cameras
- PC cameras
- Cellular phones

Table 1: Key Performance Parameters

Parameter	Value	
Optical format	1/4.5-inch (4:3)	
Active imager size	3.24mm(H) x 2.41mm(V)	
Active pixels	1472H x 1096V	
Pixel size	2.2 x 2.2 μ m	
Chief ray angle	9.4°	
Color filter array	RGB Bayer pattern, mono	
Shutter type	Global reset release (GRR) (snapshot only), electronic rolling shutter (ERS)	
Maximum data rate/master clock	99 Mp/s at 49.5 MHz (parallel)	
Frame rate	1440H x 1080V	Programmable up to 30 fps
	1280H x 720V	Programmable up to 60 fps
ADC resolution	12-bit, on-chip	
Responsivity	1.5 V/lux-sec (550nm)	
Dynamic range	71 dB	
SNR _{MAX}	39.6 dB	
Supply voltage	Digital	1.7–1.9V
	I/O	2.6–3.1V
	PLL	2.6–3.1V
	Analog	2.6–3.1V
Power consumption	364.6mW at 2.8V (parallel)	
Operating temperature	–30°C to +70°C	
Packaging	48-pin CLCC	

Ordering Information

Table 2: Available Part Numbers

Part Number	Description
MT9M003C12STC	48-pin lead-free CLCC/color/parallel/ 9.4° CRA



Table of Contents

Features	1
Applications	1
Ordering Information	1
General Description	6
Functional Overview	6
Signal Descriptions	8
Typical Connections	10
Pixel Array Structure	11
Default Readout Order	11
Output Data Format	13
Parallel Pixel Data Interface	13
Output Data Timing (Parallel Pixel Data Interface)	14
Frame Timing Details	15
Serial Bus Description	17
Protocol	17
Sequence	17
Bus Idle State	17
Start Bit	17
Stop Bit	18
Slave Address	18
Data Bit Transfer	18
Acknowledge Bit	18
No-Acknowledge Bit	18
Two-Wire Serial Interface Sample Write and Read Sequences	19
16-Bit Write Sequence	19
16-Bit Read Sequence	19
Signal Chain and Datapath	20
Gains	20
Analog Gain	20
Digital Gain	21
Offset	21
Analog Black Level Calibration	21
Digital Black Level Calibration	21
Features	22
PLL-Generated Master Clock	22
PLL Setup Sequence	23
PLL Setup Sample Code for Parallel Mode After Power-Up	23
Maintaining a Constant Frame Rate	24
Synchronizing Register Writes to Frame Boundaries	24
Restart	25
Window Size	25
Readout Modes	25
Mirror Mode	25
Row Mirror	26
Image Acquisition Modes	27
Electronic Rolling Shutter	27
Global Reset Release	27
Exposure	27
Operating Modes	28
Strobe Control	30
Timing Specifications	31


 MT9M003: 1/4.5-Inch 1.6Mp CMOS Digital Image Sensor
 Table of Contents

Power-Up Sequence	31
Power-Down Sequence	32
Reset	33
Hard Reset	33
Soft Reset	33
Signal State During Reset	33
Standby and Chip Enable (Power Save Mode)	34
Spectral Characteristics	35
CRA Characteristics	36
Electrical Specifications	37
Two-Wire Serial Register Interface	37
I/O Timing	38
DC Electrical Characteristics	40
Absolute Maximum Ratings	41
Package Dimensions	42
Revision History	43



List of Figures

Figure 1:	Block Diagram – Parallel Output	.6
Figure 2:	48-Pin CLCC 11.43 x 11.43 Package Pinout Diagram (Top View)	.9
Figure 3:	Typical Configuration – Parallel Connection	.10
Figure 4:	Pixel Array Description	.11
Figure 5:	Pixel Color Pattern Detail (Top Right Corner)	.12
Figure 6:	Imaging a Scene	.12
Figure 7:	Spatial Illustration of Image Readout - Parallel Interface	.13
Figure 8:	Pixel Data Timing Example	.14
Figure 9:	Row Timing and FRAME_VALID/LINE_VALID Signals	.14
Figure 10:	1280x720/60 fps Row Timing Details	.15
Figure 11:	1440x1080/30 fps Mode	.15
Figure 12:	Frame Timing Overview	.16
Figure 13:	Timing Diagram Showing a Write to R0x09 with the Value 0x0284	.19
Figure 14:	Timing Diagram Showing a Read from R0x09; Returned Value 0x0284	.19
Figure 15:	Signal Path	.20
Figure 16:	PLL-Generated Master Clock	.22
Figure 17:	Six Pixels in Normal and Column Mirror Readout Modes (Color)	.26
Figure 18:	Six Pixels in Normal and Column Mirror Readout Modes (Mono)	.26
Figure 19:	Six Rows in Normal and Row Mirror Readout Modes	.26
Figure 20:	ERS Snapshot Timing	.29
Figure 21:	GRR Snapshot Timing	.30
Figure 22:	Power Supply Power-Up Sequence	.31
Figure 23:	Power Supply Power-Down Sequence	.32
Figure 24:	Typical Color Spectral Characteristics	.35
Figure 25:	Chief Ray Angle (CRA) vs. Image Height	.36
Figure 26:	Two-Wire Serial Bus Timing Parameters	.37
Figure 27:	Parallel I/O Timing Diagram	.38
Figure 28:	48-Pin CLCC Package Outline	.42



List of Tables

Table 1:	Key Performance Parameters	1
Table 2:	Available Part Numbers	1
Table 3:	Signal Descriptions for MT9M003 in CLCC Package	8
Table 4:	Pixel Type by Column	11
Table 5:	Pixel Type by Row	11
Table 6:	Device Addresses	18
Table 7:	Frequency Parameters	23
Table 8:	Operating Modes	28
Table 9:	STROBE Timepoints	30
Table 10:	Power Supply Power-Up Timing	32
Table 11:	Power Supply Power-Down Timing	32
Table 12:	Signal State During Reset	33
Table 13:	Standby Modes	34
Table 14:	Two-Wire Serial Bus Characteristics	37
Table 15:	I/O Timing Characteristics	39
Table 16:	DC Electrical Characteristics	40
Table 17:	Power Consumption – Parallel	40
Table 18:	Absolute Maximum Values	41



General Description

The Aptina MT9M003 is a 1/4.5-inch format CMOS active-pixel digital image sensor with a pixel array of 1472H x 1096V. The default active imaging array size is 1440 x 1080. It incorporates sophisticated on-chip camera functions such as windowing, mirroring, and snapshot mode. It is programmable through a simple two-wire serial interface and has very low power consumption.

The MT9M003 digital image sensor features DigitalClarity—Aptina’s breakthrough low-noise CMOS imaging technology that achieves near-CCD image quality (based on signal-to-noise ratio and low-light sensitivity) while maintaining the inherent size, cost, and integration advantages of CMOS.

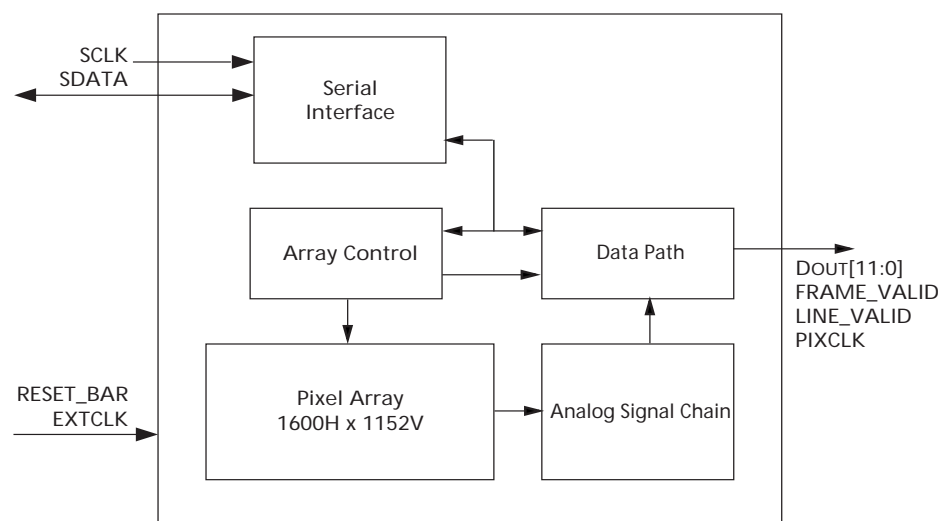
Functional Overview

The MT9M003 is a progressive-scan sensor that generates a stream of pixel data at a constant frame rate. It uses an on-chip, phase-locked loop (PLL) to generate all internal clocks from a single master input clock running between 8 and 16.5 MHz.

User interaction with the sensor is through the two-wire serial bus, which communicates with the array control, analog signal chain, and digital signal chain. The core of the sensor is a 1.6Mp 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 is sequenced through an analog signal chain (providing offset correction and gain), and then through an ADC. The output from the ADC is a 12-bit value for each pixel in the array. The ADC output passes through a digital processing signal chain (which provides further data path corrections and applies digital gain). The pixel data are output at a rate of up to 99 Mp/s, in addition to frame and line synchronization signals in parallel mode corresponding to a pixel clock rate of 99 MHz. Figure 1 shows the block diagram of the sensor.

Figure 1: Block Diagram – Parallel Output





MT9M003: 1/4.5-Inch 1.6Mp CMOS Digital Image Sensor Functional Overview

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

The sensor 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 setting. These registers can be accessed through a two-wire serial interface.

The output from the sensor (MT9M003I12STC) 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 and to support the provision of an external mechanical shutter.



Signal Descriptions

Table 3 provides signal descriptions for the MT9M003 in a CLCC package.

Table 3: Signal Descriptions for MT9M003 in CLCC Package

Pin Numbers CLCC	Name	Type	Description
26	SCLK	Input	Serial clock. Pull to VDD_IO with a 1.5kΩ resistor (depending on bus loading).
45	SADDR0	Input	Serial address. Pull to VDD_IO with a 1.5kΩ resistor (depending on bus loading).
28	SADDR1	Input	Serial address. Pull to VDD_IO with a 1.5kΩ resistor (depending on bus loading).
21	RESET_BAR	Input	Master reset signal, active LOW.
33	EXTCLK	Input	Input clock signal 8–16.5 MHz.
5	TRIGGER	Input	Snapshot trigger. Used to trigger one frame of output in snapshot modes.
23, 25	TEST	Input	Enables manufacturing test modes. Tie to digital GND for functional operation.
27	SDATA	I/O	Serial data. Pull to VDD_IO with a 1.5kΩ resistor (depending on bus loading).
1	STROBE	Output	Snapshot strobe. Driven HIGH when all pixels are exposing in snapshot modes.
4	DOUT[0]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
48	DOUT[1]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
46	DOUT[2]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
20	DOUT[3]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
22	DOUT[4]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
24	DOUT[5]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
37	DOUT[6]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
35	DOUT[7]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
34	DOUT[8]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
38	DOUT[9]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
40	DOUT[10]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
41	DOUT[11]	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
47	PIXCLK	Output	Pixel clock. Used to qualify the LINE_VALID, FRAME_VALID, and DOUT(11:0). These outputs should be captured on the falling edge of this signal.
3	FRAME_VALID	Output	Frame valid. Qualified by PIXCLK. Driven HIGH during active pixels and horizontal blanking of each frame and LOW during vertical blanking.
2	LINE_VALID	Output	Line valid output. Qualified by PIXCLK. Driven HIGH with active pixels of each line and LOW during horizontal blanking periods. External pull down resistor to DGND (typical 10k–100k) required for proper initialization sequence.
29, 44	VDD	Supply	Digital power 1.8V nominal.
10, 11	VAA_PIX	Supply	Pixel array power 2.8V nominal.

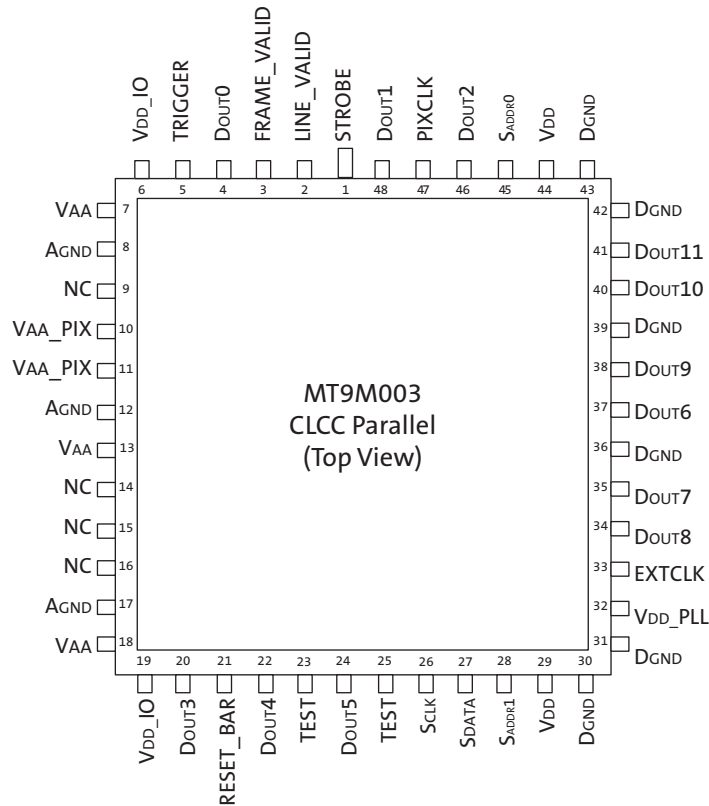


MT9M003: 1/4.5-Inch 1.6Mp CMOS Digital Image Sensor
Signal Descriptions

Table 3: Signal Descriptions for MT9M003 in CLCC Package (continued)

Pin Numbers CLCC	Name	Type	Description
7, 13, 18	VAA	Supply	Analog power 2.8V nominal.
32	VDD_PLL	Supply	PLL power 2.8V nominal.
6, 19	VDD_IO	Supply	I/O power supply 2.8V nominal.
30, 31, 36, 39, 42, 43	DGND	Supply	Digital ground.
8, 12, 17	AGND	Supply	Analog ground.
9, 14, 15, 16	NC	–	No connect.

Figure 2: 48-Pin CLCC 11.43 x 11.43 Package Pinout Diagram (Top View)





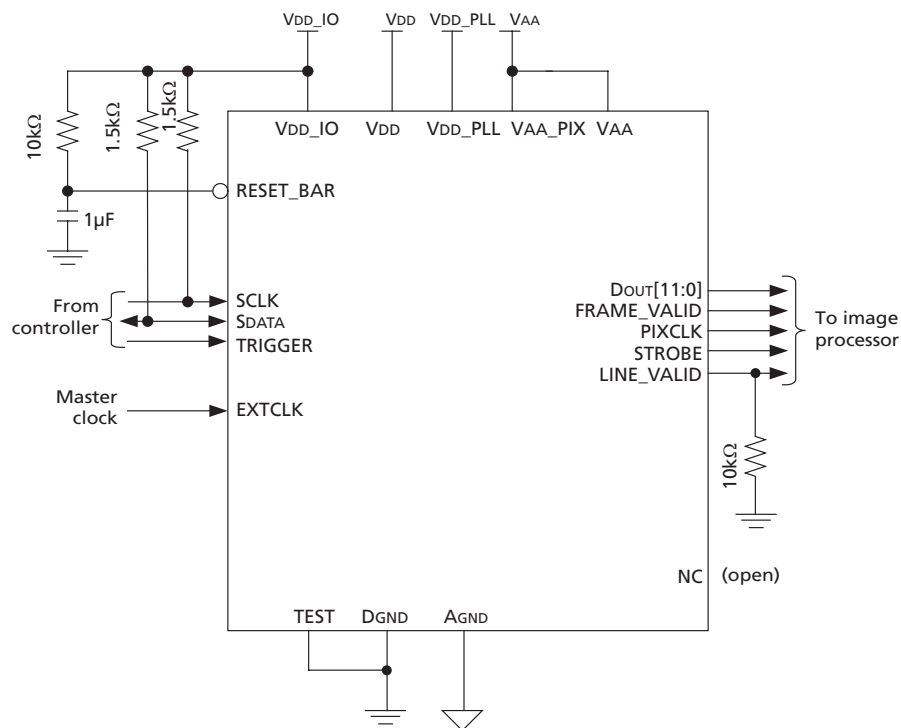
MT9M003: 1/4.5-Inch 1.6Mp CMOS Digital Image Sensor Typical Connections

Typical Connections

Figure 3 shows typical connections for the MT9M003 sensor. For low-noise operation, the MT9M003 requires separate power supplies for analog and digital. Incoming digital and analog ground conductors can be tied together next to the die. Both power supply rails should be decoupled from ground using capacitors as close as possible to the die. The use of inductance filters is not recommended on the power supplies or output signals.

The MT9M003 also supports different digital core (VDD/DGND) and I/O power (VDD_IO/DGND) power domains that can be at different voltages. PLL requires a clean power source (VDD_PLL).

Figure 3: Typical Configuration – Parallel Connection



- Notes:
1. Typical connection shows only one scenario out of multiple possible variations for this sensor.
 2. All inputs must be configured with VDD_IO.
 3. VAA and VAA_PIX must be tied together.



Pixel Array Structure

The MT9M003 pixel array consists of a 1600-column by 1152-row matrix of pixels addressed by column and row. The address (column 0, row 0) represents the upper-right corner of the entire array, looking at the sensor, as shown in Figure 4.

The array consists of a 1440-column by 1080-row active region in the center representing the default output image resolution, surrounded by a boundary region (also active), surrounded by a border of dark pixels (see Table 4 and Table 5). The boundary region can be used to avoid edge effects when doing color processing, while the optically black column and rows can be used to monitor the black level.

Table 4: Pixel Type by Column

Column	Pixel Type
0–15	Active boundary (16)
16–1455	Active image (1440)
1456–1471	Active boundary (16)
1472–1599	Black (128)

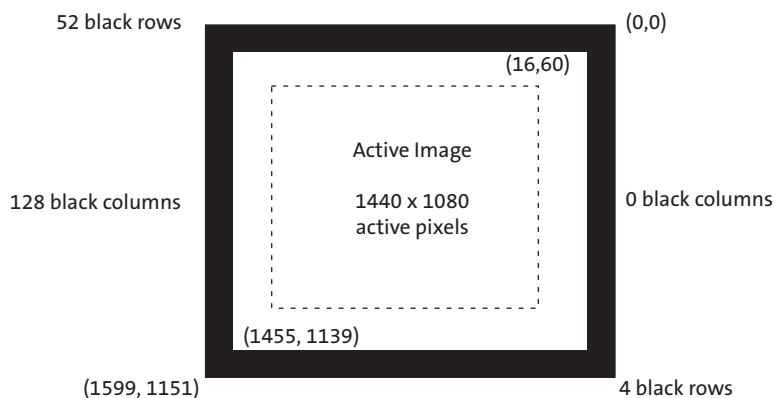
Table 5: Pixel Type by Row

Row	Pixel Type
0–51	Black (52)
53–59	Active boundary (8)
60–1139	Active image (1,080)
1140–1147	Active boundary (8)
1148–1151	Black (4)

Default Readout Order

By convention, the sensor core pixel array is shown with pixel (0,0) in the top right corner (see Figure 4). This reflects the actual layout of the array on the die. Also, the first pixel data read out of the sensor in default condition is that of pixel (16,60).

Figure 4: Pixel Array Description

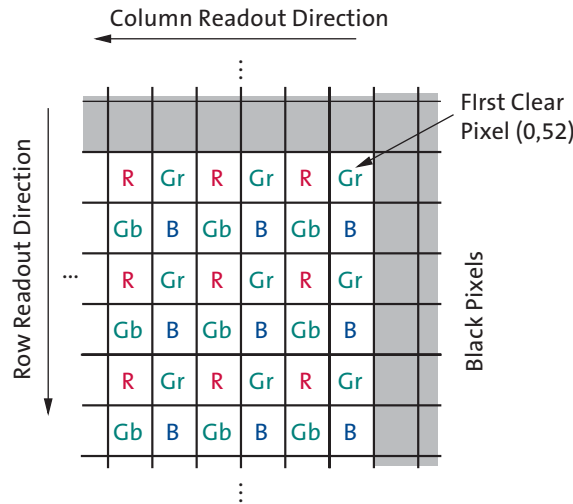




MT9M003: 1/4.5-Inch 1.6Mp CMOS Digital Image Sensor Pixel Array Structure

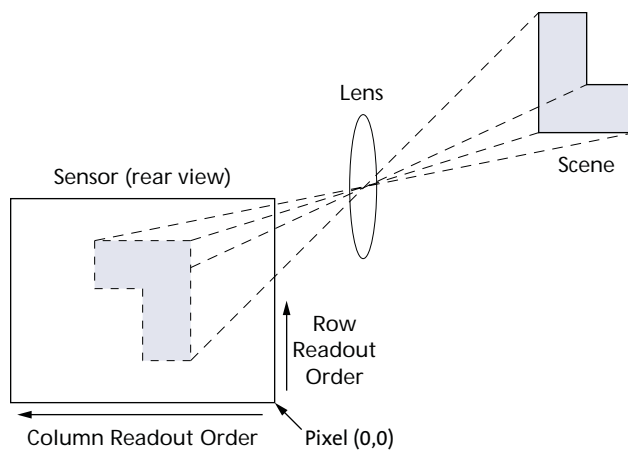
Sensor pixels are output in a Bayer pattern format consisting of four “colors”—GreenR, GreenB, Red, and Blue (Gr, Gb, R, B)—representing three filter colors. When no mirror modes are enabled, even-numbered rows contain alternate green1 and red pixels; odd-numbered rows contain alternate blue and green2 pixels. Even-numbered columns contain greenR and blue pixels; odd-numbered columns contain red and greenB pixels. The GreenR and GreenB pixels have the same color filter, but they are treated as separate colors by the data path and analog signal chain.

Figure 5: Pixel Color Pattern Detail (Top Right Corner)



When the sensor is imaging, the active surface of the sensor faces the scene, as shown in Figure 6. When the image is read out of the sensor, it is read one row at a time, with the rows and columns sequenced, as shown in Figure 5.

Figure 6: Imaging a Scene



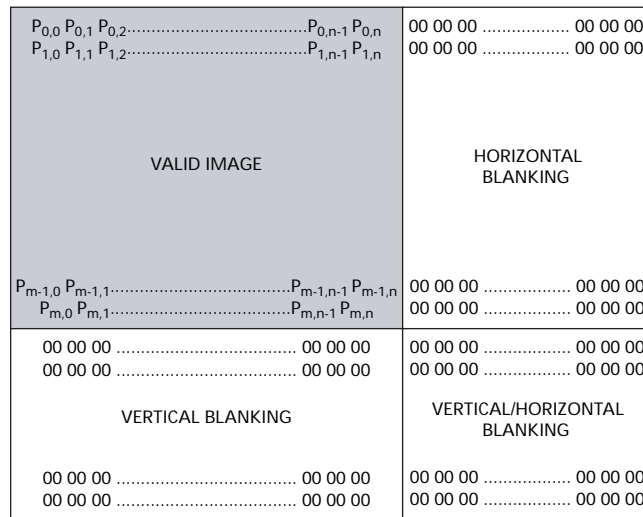


Output Data Format

Parallel Pixel Data Interface

MT9M003 image data is read out in a progressive scan. Valid image data is surrounded by horizontal blanking and vertical blanking, as shown in Figure 7. The amount of horizontal blanking and vertical blanking is programmable; LINE_VALID is HIGH during the shaded region of the figure. FRAME_VALID timing is described in the next section.

Figure 7: Spatial Illustration of Image Readout - Parallel Interface





Output Data Timing (Parallel Pixel Data Interface)

The sensor core output data is synchronized with the PIXCLK output. When LINE_VALID is HIGH, one pixel data is output on the 12-bit DOUT output every PIXCLK period. By default, the internal PLL is used and PIXCLK runs at the 2X master clock. The falling edge of PIXCLK appears at the center of the DOUT. This allows PIXCLK to be used as a clock to sample the data.

By default, PIXCLK is not enabled, and its on or off is register controllable. When on, PIXCLK is continuously enabled, even during the blanking period. The MT9M003 can be programmed to delay the PIXCLK edge relative to the DOUT transitions. This can be achieved by programming the corresponding register bits.

Figure 8: Pixel Data Timing Example

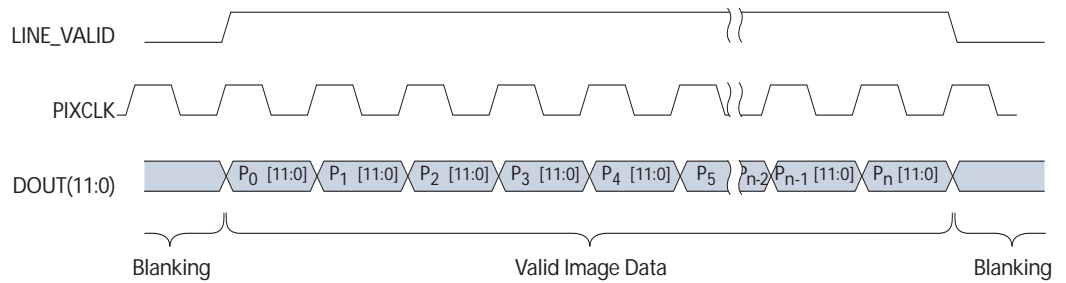
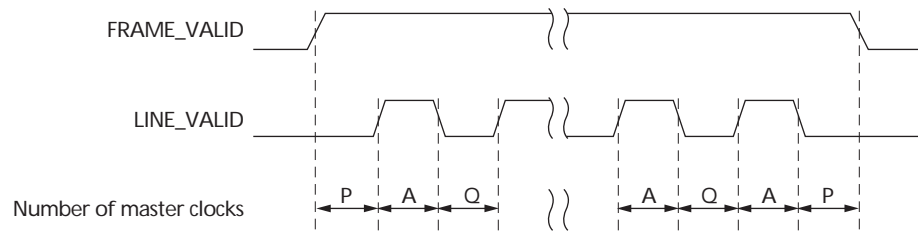


Figure 9: Row Timing and FRAME_VALID/LINE_VALID Signals



The sensor timing is shown in terms of pixel clock and master clock cycles (Figure 8 and Figure 9).

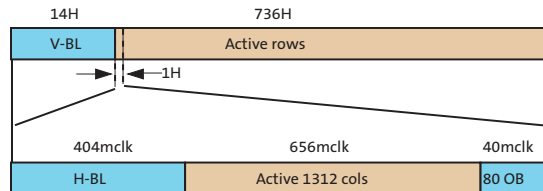


MT9M003: 1/4.5-Inch 1.6Mp CMOS Digital Image Sensor Output Data Timing (Parallel Pixel Data Interface)

Frame Timing Details

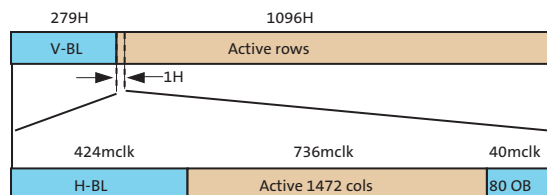
This section discusses row timing for 1440 x 1080 30fps and 1280 x 720 60 fps modes. In Figure 10 and Figure 11, H-BL is horizontal blanking, OB is optically black columns, H is one row, and V-BL is vertical blanking. The pixel clock is two times master clock (PCLK = 99 MHz, MCLK = 49.5 MHz). Each PCLK outputs one active pixel or one black pixel. H-BL setting value uses MCLK as a unit (one horizontal blank needs two PCLKs).

Figure 10: 1280x720/60 fps Row Timing Details



- $mclk = 49.5 \text{ MHz}$
 $H = H\text{-BL} + \text{Active Cols}/2 + \text{OB}/2$
 $H = 404 \text{ mclks} + 656 \text{ mclks} + 40 \text{ mclks} = 1100 \text{ mclks} = 22.22\mu\text{s}$
- $V = 14H + 736H = 750H$
 $t_{\text{FRAME}} = H \times V$
 $= 1100 \times 750 = 825000 \text{ mclks} = 825000 \text{ mclks}/49.5 \text{ MHz} = 16.66\text{ms}$
- Frame rate = $1/t_{\text{FRAME}}$
 $= 1/16.66\text{ms} = 60 \text{ fps}$
- Active readout window is 1312 (1280+32 boundary) columns x 736 (720+16 boundary) rows

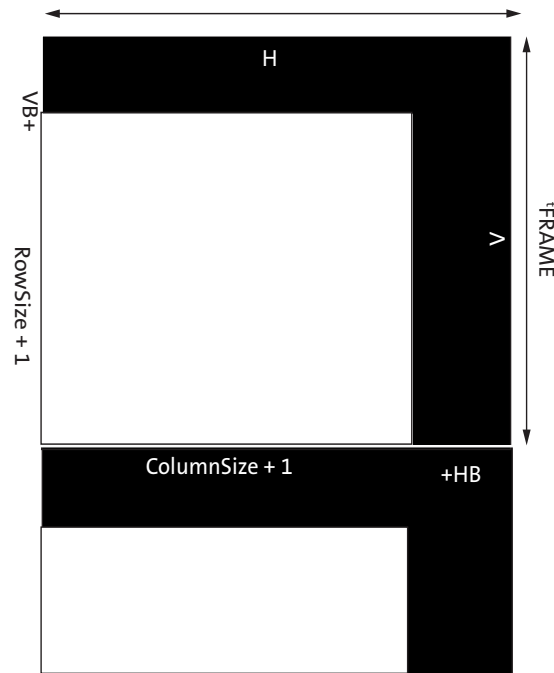
Figure 11: 1440x1080/30 fps Mode



- $mclk = 49.5 \text{ MHz}$
- $H = H\text{-BL} + \text{Active Cols}/2 + \text{OB}/2$
 $= 424 \text{ clks} + 736 \text{ mclks} + 40 \text{ mclks} = 1200 \text{ mclks} = 24.24\mu\text{s}$
- $V = 279H + 1096H = 1375H$
- $t_{\text{FRAME}} = H \times V$
 $= 1200 \times 1375 = 1650000 \text{ mclks} = 1650000 \text{ mclks}/49.5 \text{ MHz} = 33.33\text{ms}$
- Frame rate = $1/t_{\text{FRAME}}$
 $= 1/33.33\text{ms} = 30 \text{ fps}$
- Active readout window is 1472 (1440+32 boundary) columns x 1096 (1080+16 boundary) rows



Figure 12: Frame Timing Overview



- $t_{FRAME} = V * H$
 $V = VB + (RowSize + 1)$
- $H = t_{ROW} = ((ColumnSize + 2 * HB)) * t_{PIXCLK}$
 $H_{bmin} = 353$
- H_{bmin} is dependent on the readout mode and may be smaller than 353 for some readout modes. If HB is configured to a value of less than H_{bmin} , then the MT9M003 will operate with $HB = H_{bmin}$.

Example Timing Configuration for $t_{PIXCLK} = 96$ MHz:

VB (R0x06) = 885, $RowSize$ (R0x03) = 719,
 $V = 885 + (719 + 1) = 1605$

$ColumnSize$ (R0x04) = 1279, HB (R0x05) = 353

- $t_{ROW} = ((1279 + 1) + (2 * 353)) / 96 \text{ MHz} = 20.69 \mu\text{s}$
- $t_{FRAME} = 1605 * 20.69 \mu\text{s} = 0.332 \text{ ms}$
- $Frame \text{ Rate} = 1 / t_{FRAME} = 30 \text{ fps}$



Serial Bus Description

Registers are written to and read from the MT9M003 through the two-wire serial interface bus. The MT9M003 is a serial interface slave and is controlled by the serial clock (SCLK), which is driven by the serial interface master. Data is transferred into and out of the MT9M003 through the serial data (SDATA) line. The SDATA line is pulled up to VDD_IO off-chip by a 1.5k Ω resistor. Either the slave or master device can pull the SDATA line LOW—the serial interface protocol determines which device is allowed to pull the SDATA line down at any given time.

Protocol

The two-wire serial defines several different transmission codes, as follows:

- a start bit
- the slave device 8-bit address
- an (a no) acknowledge bit
- an 8-bit message
- a stop bit

Sequence

A typical read or write sequence begins by the master sending a start bit. After the start bit, the master sends the slave device's 8-bit address. The last bit of the address determines if the request is a READ or a WRITE, where a "0" indicates a WRITE and a "1" indicates a READ. The slave device acknowledges its address by sending an acknowledge bit back to the master.

If the request was a WRITE, the master then transfers the 8-bit register address to which a WRITE should take place. The slave sends an acknowledge bit to indicate that the register address has been received. The master then transfers the data 8 bits at a time, with the slave sending an acknowledge bit after each 8 bits. The MT9M003 uses 16-bit data for its internal registers, thus requiring two 8-bit transfers to write to one register. After 16 bits are transferred, the register address is automatically incremented, so that the next 16 bits are written to the next register address. The master stops writing by sending a start or stop bit.

A typical READ sequence is executed as follows. First the master sends the write-mode slave address and 8-bit register address, just as in the write request. The master then sends a start bit and the read-mode slave address. The master then clocks out the register data 8 bits at a time. The master sends an acknowledge bit after each 8-bit transfer. The register address is automatically incremented after every 16 bits is transferred. The data transfer is stopped when the master sends a no-acknowledge bit.

Bus Idle State

The bus is idle when both the data and clock lines are HIGH. Control of the bus is initiated with a start bit, and the bus is released with a stop bit. Only the master can generate the start and stop bits.

Start Bit

The start bit is defined as a HIGH-to-LOW transition of the data line while the clock line is HIGH.



Stop Bit

The stop bit is defined as a LOW-to-HIGH transition of the data line while the clock line is HIGH.

Slave Address

The 8-bit address of the two-wire serial interface device consists of 7 bits of address and 1 bit of direction. A “0” in the LSB (least significant bit) of the address indicates write mode (0xB8), and a “1” indicates read mode (0xB9).

The two-wire serial interface device addresses consists of 7 bits. The MT9M002 allows for multiple device addresses in master/slave mode, as shown in Table 6. The 2 LSBs of the device address are defined by SADDR0 and SADDR1 input port values.

Table 6: Device Addresses

SADDR1	SADDR0	Device Address
0	0	0xB8
0	1	0xBA
1	0	0xBC
1	1	0xBE

Data Bit Transfer

One data bit is transferred during each clock pulse. The serial interface clock pulse is provided by the master. The data must be stable during the HIGH period of the two-wire serial interface clock—it can only change when the serial clock is LOW. Data is transferred 8 bits at a time, followed by an acknowledge bit.

Acknowledge Bit

The master generates the acknowledge clock pulse. The transmitter (which is the master when writing, or the slave when reading) releases the data line, and the receiver indicates an acknowledge bit by pulling the data line LOW during the acknowledge clock pulse.

No-Acknowledge Bit

The no-acknowledge bit is generated when the data line is not pulled down by the receiver during the acknowledge clock pulse. A no-acknowledge bit is used to terminate a read sequence.

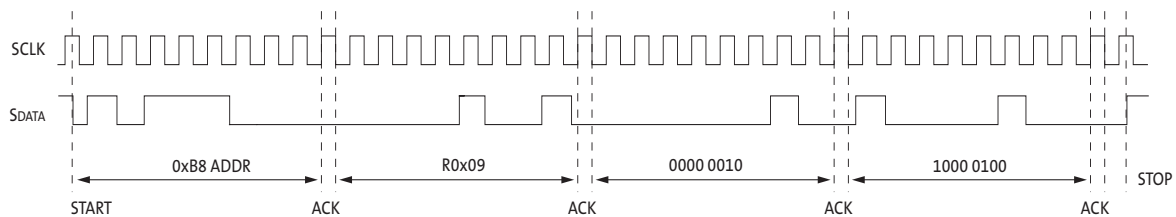


Two-Wire Serial Interface Sample Write and Read Sequences

16-Bit Write Sequence

A typical write sequence for writing 16 bits to a register is shown in Figure 13. A start bit given by the master, followed by the write address, starts the sequence. The image sensor then sends an acknowledge bit and expects the register address to come first, followed by the 16-bit data. After each 8-bit transfer, the image sensor sends an acknowledge bit. All 16 bits must be written before the register is updated. After 16 bits are transferred, the register address is automatically incremented so that the next 16 bits are written to the next register. The master stops writing by sending a start or stop bit.

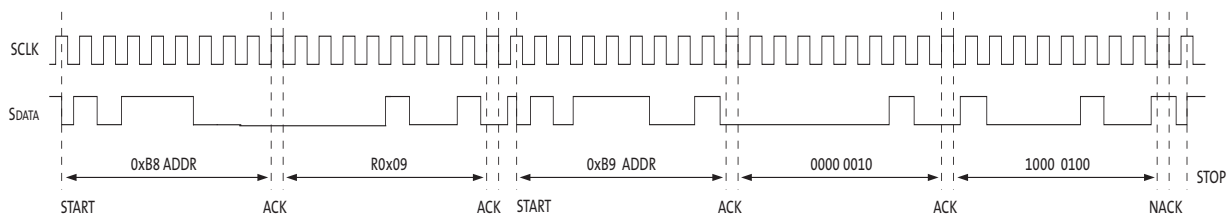
Figure 13: Timing Diagram Showing a Write to R0x09 with the Value 0x0284



16-Bit Read Sequence

A typical read sequence is shown in Figure 14. First the master has to write the register address, as in a write sequence. Then a start bit and the read address specify that a READ is about to happen from the register. The master then clocks out the register data 8 bits at a time. The master sends an acknowledge bit after each 8-bit transfer. The register address is incremented after every 16 bits is transferred. The data transfer is stopped when the master sends a no-acknowledge bit.

Figure 14: Timing Diagram Showing a Read from R0x09; Returned Value 0x0284



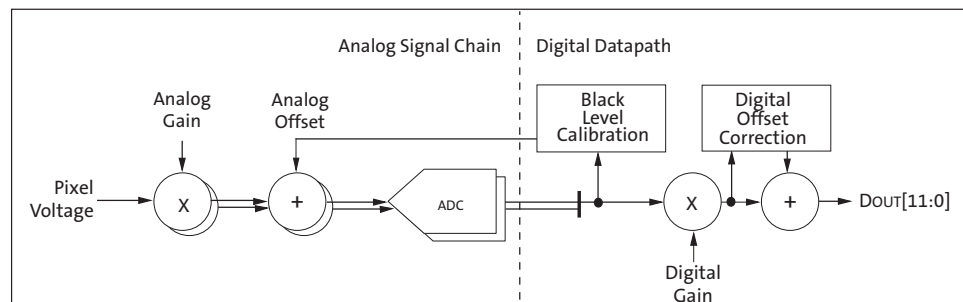


Signal Chain and Datapath

The signal chain and datapath are shown in Figure 15. Each color is processed independently, including separate gain and offset settings. Voltages sampled from the pixel array are first passed through an analog gain stage, which can produce gain factors between 1 and 7.875. An analog offset is then applied, and the signal is sent through a 12-bit analog-to-digital converter. In the digital space, a digital gain factor of between 1 and 16 is applied, and then a digital offset of between -2048 and 2047 is added. The resulting 12-bit pixel value is then output on the DOUT[11:0] ports.

The analog offset applied is determined automatically by the black level calibration algorithm, which attempts to shift the output of the analog signal chain so that black is maintained. The digital offset is a fine-tuning of the analog offset.

Figure 15: Signal Path



Gains

The MT9M003 supports two types of gain: analog gain and digital gain. Combined, gains of between 1 and 126 are possible. It is recommended that analog gain should be maximized before applying digital gain.

The sensor provides per-color gain control as well as the option of global gain control. 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.

The combined gain for a color C is given by:

$$G_C = AG_C \times DG_C \quad (\text{EQ 1})$$

Analog Gain

The analog gain is specified independently for each color channel. There are two components, the gain and the multiplier. The gain is specified by Green1_Analog_Gain, Red_Analog_Gain, Blue_Analog_Gain, and Green2_Analog_Gain. The analog multiplier is specified by Green1_Analog_Multiplier, Red_Analog_Multiplier, Blue_Analog_Multiplier, and Green2_Analog_Multiplier. These combine to form the analog gain for a given color C, as shown in this equation:

$$AG_C = (1 + C_Analog_Multiplier) \times (C_Analog_Gain / 16) \quad (\text{EQ 2})$$

The gain component can range from 0 to 7.875 in steps 0.0625 for <4 gain, and 0.125 for >4 gain, and the multiplier component can be either 0 or 1 (resulting in a multiplier of 1 or 2). However, it is best to keep the gain component between 1 and 4 for the best noise performance, and use the multiplier for gains between 4 and 7.825.



Digital Gain

The digital gain is specified independently for each color channel in steps of 0.125. It is controlled by the register fields Green1_Digital_Gain, Red_Digital_Gain, Blue_Digital_Gain, and Green2_Digital_Gain. The digital gain for a color C is given by:

$$DG_C = 1 + (C_Digital_Gain / 8) \quad (\text{EQ 3})$$

Offset

The MT9M003 sensor can apply an offset or shift to the image data in several ways.

An analog offset can be applied on a color-wise basis to the pixel voltage as it enters the ADC. This makes it possible to adjust for offset introduced in the pixel sampling and gain stages to be removed, centering the resulting voltage swing in the ADC's range. This offset can be automatically determined by the sensor using the automatic black level calibration (BLC) circuit, or it can be set manually by the user. It is a fairly coarse adjustment, with adjustment step sizes of four to eight LSBs.

Digital offset is also added on a color-wise and line-wise basis to fine-tune the black level of the output image. This offset is based on an average black level taken from each row's dark columns, and is automatically determined by the digital row-wise black level calibration (RBLC) circuit. If the RBLC circuit is not used, a user-defined offset can be applied instead. This offset has a resolution of 1 LSB.

A digital offset is added on a color-wise basis to account for channel offsets that can be introduced due to "even" and "odd" pixels of the same color going through a slightly different ADC chain. This offset is automatically determined based on dark row data, but it can also be manually set.

Analog Black Level Calibration

The MT9M003 black level calibration circuitry provides a feedback control system since adjustments to the analog offset are imprecise by nature. The goal is that within the dark row region of any supported output image size, the offset should have been adjusted such that the average black level falls within the specified target thresholds.

The analog offsets normally need a major adjustment only when leaving the Reset state or when there has been a change to a color's analog gain. Factors like shutter width and temperature have lower-order impact, and generally only require a minor adjustment to the analog offsets. The MT9M003 has various calibration modes to keep the system stable while still supporting the need for rapid offset adjustments when necessary.

Digital Black Level Calibration

Digital black level calibration is the final calculation applied to pixel data before it is output. It provides a precise black level to complement the coarser-grained analog black level calibration, and also corrects for black level shift introduced by digital gain. This correction applies to the active columns for all rows, including dark rows.



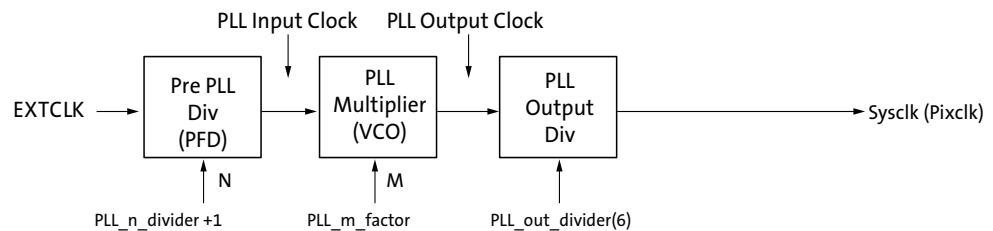
Features

PLL-Generated Master Clock

The PLL can generate a PIXCLK clock signal whose frequency is up to 99 MHz (input clock from 8–16.5 MHz). The PLL-generated clock can be controlled by programming the appropriate register. It is possible to bypass the PLL and use EXTCLK as master clock. By default, the PLL is powered up.

The PLL contains a prescaler to divide the input clock applied on EXTCLK, a VCO to multiply the prescaler output, and PLL output divider stage to generate the output clock. The clocking structure is shown in Figure 16. PLL control can be programmed to generate desired pixel clock frequency.

Figure 16: PLL-Generated Master Clock



Note: The PLL control registers must be programmed while the sensor is in the software Standby state. The effect of programming the PLL divisors while the sensor is in the streaming state is undefined.



PLL Setup Sequence

To use the PLL:

1. Power up the MT9M003; ensure that f_{EXTCLK} is between 8 and 16.5 MHz.
2. Put the MT9M003 into master mode.
3. Set the PLL out divider to 7 (refer to “PLL Setup Sample Code for Parallel Mode After Power-Up” below).

Note: The power-up default PLL out divider setting is 6.

4. Set PLL_m_factor and PLL_n_divider based on the desired input ($f_{EXTCLK} = 13.5$ MHz) and output ($f_{PIXCLK} = 99$ MHz) frequencies.

Using this formula:

$$f_{PIXCLK} = f_{VCO} / 7 \quad (EQ 4)$$

where

$$f_{VCO} = (f_{EXTCLK} \times M) / N$$

$$M = \text{PLL_m_factor},$$

$$N = (\text{PLL_n_divider} + 1)$$

5. Wait 1ms to ensure that the VCO has locked.
6. Select the PLL as the clock source.
7. Enable the parallel data output.

PLL Setup Sample Code for Parallel Mode After Power-Up

(with input clock frequency 13.5 MHz)

1. Set R0x1E = 0x8006 // Master Mode
2. Set R0x9F = 0x0070 // Set-up for changing to 14-bit mode.
3. Set R0x9E = 0x101E // Set 14-bit mode, select 7 divider, parallel mode.
4. Set R0x11 = 0x9A02 // Assuming an input EXTCLK of 13.5 MHz, generates an output PIXCLK of 99 MHz.
5. Delay = 1ms // Ensures VCO has locked.
6. Set R0x10 = 0x0053 // Select PLL as clock source.
7. Set R0x9F = 0x3070 // Parallel data out.

Table 7: Frequency Parameters

Parameter	Equation	Min	Max	Unit
PLL_n_divider	–	0	63	
PLL_m_factor	–	16	255	
f_{EXTCLK}	–	8	16.5	MHz
f_{PFD}	$f_{EXTCLK} / (\text{PLL_n_divider} + 1)$	2	24	MHz
f_{VCO}	$f_{EXTCLK} * \text{PLL_m_factor} / (\text{PLL_n_divider} + 1)$	320	693	MHz



Maintaining a Constant Frame Rate

Maintaining a constant frame rate while continuing to have the ability to adjust certain parameters is often desired. This is not always possible, however, since register updates are synchronized to the read pointer, and the shutter pointer for a frame is usually active during the readout of the previous frame. Therefore, any register changes that could affect the row time or the set of rows sampled causes the shutter pointer to start over at the beginning of the next frame.

By default, the following register fields cause a "bubble" in the output rate (the vertical blank increases for one frame) if they are written in continuous mode, even if the new value would not change the resulting frame rate:

- Row_Start
- Row_Size
- Column_Size
- Horizontal_Blank
- Vertical_Blank
- Shutter_Delay
- Mirror_Row

The size of this bubble is $(SW \times {}^tROW)$, calculating the row time according to the new settings.

The Shutter_Width_Lower and Shutter_Width_Upper fields may be written without causing a bubble in the output rate under certain circumstances. Since the shutter sequence for the next frame often is active during the output of the current frame, this would not be possible without special provisions in the hardware. Writes to these registers take effect two frames after the frame they are written, which allows the shutter width to increase without interrupting the output or producing a corrupt frame (as long as the change in shutter width does not affect the frame time).

Synchronizing Register Writes to Frame Boundaries

Changes to most register fields that affect the size or brightness of an image take effect on the frame after the one during which they are written. These fields are noted as "synchronized to frame boundaries" in Table 7 on page 23. To ensure that a register update takes effect on the next frame, the WRITE operation must be completed after the leading edge of FV and before the trailing edge of FV.

As a special case, in snapshot modes (see below), register WRITES that occur after FV but before the next trigger will take effect immediately on the next frame, as if there had been a restart. However, if the trigger for the next frame in ERS snapshot mode occurs during FV, register WRITES take effect as with continuous mode.

Additional control over the timing of register updates can be achieved by using Synchronize_Changes. If this bit is set, WRITES to certain register fields that affect the brightness of the output image do not take effect immediately. Instead, the new value is remembered internally. When Synchronize_Changes is cleared, all the updates simultaneously take effect on the next frame (as if they had all been written the instant Synchronize_Changes was cleared). Register fields affected by this bit are identified in Table 2: Core Registers – Register Description on page 10 of the register reference.

Fields not identified as being frame-synchronized or affected by Synchronize_Changes are updated immediately after the register write is completed. The effect of these registers on the next frame can be difficult to predict if they affect the shutter pointer.



Restart

To restart the MT9M003 at any time during the operation of the sensor, write a “1” to the restart register (R0x0B[0] = 1). This has two effects: first, the current frame is interrupted immediately. Second, any WRITES to frame-synchronized registers and the shutter width registers take effect immediately, and a new frame starts (in continuous mode). Register updates being held by Synchronize_Changes do not take effect until that bit is cleared. The current row and one following row complete before the new frame is started, so the time between issuing the restart and the beginning of the next frame can vary by about t_{ROW} .

If Pause_Restart is set, rather than immediately beginning the next frame after a Restart in continuous mode, the sensor pauses at the beginning of the next frame until Pause_Restart is cleared. This can be used to achieve a deterministic time period from clearing the Pause_Restart bit to the beginning of the first frame, meaning that the controller does not need to be tightly synchronized to LV or FV.

Note: When Pause_Restart is cleared, be sure to leave Restart set to “1” for proper operation. The restart bit will be cleared automatically by the device.

Window Size

The output image window of the pixel array (the FOV) is programmable and defined by four register fields. Column_Start and Row_Start define the X and Y coordinates of the upper left corner of the FOV. Column_Size defines the width of the FOV, and Row_Size defines the height of the FOV in array pixels.

The Column_Start and Row_Start fields must be set to an even number. The Column_Size and Row_Size fields must be set to odd numbers (resulting in an even size for the FOV). The Row_Start register should be set no lower than 12 if either Manual_BLC is cleared or Show_Dark_Rows is set. The width of the output image, W, is $Column_Size + 1$ and height, H, is $Row_Size + 1$. In default, a full resolution image size of 1440 x 1080 in output.

Readout Modes

The MT9M003 sensor supports mirror readout mode. Images can be flipped in the vertical and/or mirrored in the horizontal directions.

Mirror Mode

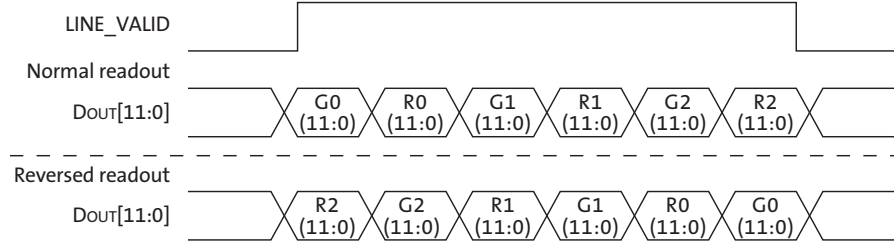
By default, active pixels in an image are output in row-major order (an entire row is output before the next row is begun), from lowest row/column number to highest. Mirror mode allows the output order of the rows and columns to be reversed. This only affects pixels in the active region of the image, not pixels read out as dark rows or dark columns. When the readout direction is reversed, the color order is reversed as well (for example, red, green, red, and so on instead of green, red, green, and so on), thus causing the Bayer order of the output image to change.



Column Mirror (Color)

The readout order of the columns are reversed, as shown in Figure 17 on page 26.

Figure 17: Six Pixels in Normal and Column Mirror Readout Modes (Color)



Row Mirror

The readout order of the rows are reversed, as shown in Figure 18.

Figure 18: Six Pixels in Normal and Column Mirror Readout Modes (Mono)

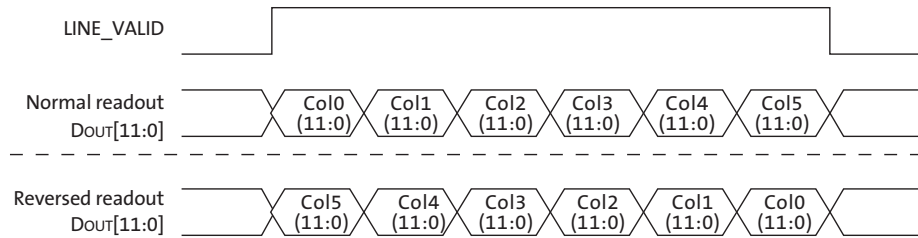


Figure 19: Six Rows in Normal and Row Mirror Readout Modes

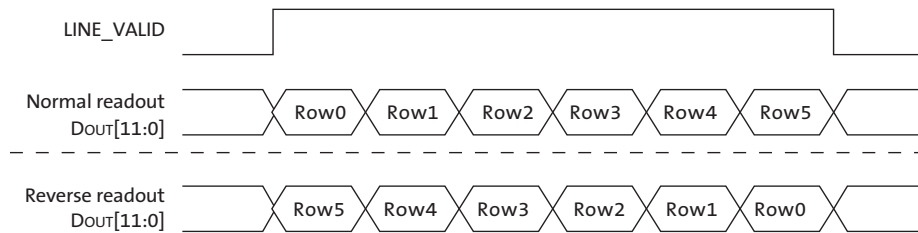




Image Acquisition Modes

The MT9M003 supports two image acquisition modes (shutter types), electronic rolling shutter (ERS), and global reset release (GRR).

Electronic Rolling Shutter

The ERS modes take pictures by scanning the rows of the sensor. On the first scan, each row is released from reset, starting the exposure. On the second scan, the row is sampled, processed, and returned to the reset state. The exposure for any row is therefore the time between the first and second scans. Each row is exposed for the same duration, but at slightly different point in time, which can cause a shear in moving subjects.

Whenever the mode is changed to an ERS mode (even from another ERS mode), and before the first frame following reset, there is an anti-blooming sequence where all rows are placed in reset. This sequence must complete before continuous readout begins.

This delay is:

$${}^tALLRESET = 16 \times 1096 \times {}^tACLK \text{ (where } {}^tACLK \text{ is } 2 * {}^tPIXCLK) \quad (EQ 5)$$

Global Reset Release

The GRR modes attempt to address the shearing effect by starting exposures of all rows at the same time. Instead of the first scan used in ERS mode, the reset to each row is released simultaneously. The second scan occurs as normal, so the exposure time for each row would differ. Typically, an external mechanical shutter would be used to stop the exposure of all rows simultaneously.

In GRR modes, there is a startup overhead before each frame as all rows are initially placed in the reset state (tALLRESET). Unlike ERS mode, this delay always occurs before each frame. However, it occurs as soon as possible after the preceding frame, so typically the time from trigger to the start of exposure does not include this delay. To ensure that this is the case, the first trigger must occur no sooner than tALLRESET after the previous frame is read out.

Exposure

The nominal exposure time, tEXP , is the effective shutter time in ERS modes, and is defined by the shutter width (SW), R8, R9 and the shutter overhead (SO), which includes the effect of Shutter_Delay. Exposure time for other modes is defined relative to this time. Increasing Shutter_Delay (SD), R12 decreases the exposure time. Exposure times are typically specified in units of row time, increasing or decreasing the shutter width (SW) register value will make exposure times increase or decrease in units of row time. It is also possible to fine-tune exposures in units of tACLK s (where ${}^tACLK = 2 * {}^tPIXCLK$) by adjusting the Shutter_Delay (SD) register.

This is expressed in the formula:

$${}^tEXP = SW \times {}^tROW - SO \times 2 \times {}^tPIXCLK \quad (EQ 6)$$

If the SD register value does not change, the $(SO \times 2 \times {}^tPIXCLK)$ in above formula is a constant offset of exposure time. Most applications do not need to change the SD register, except when fine-tuning the exposure time in units of tACLK . Under normal conditions in ERS modes, every pixel should end up with the same exposure time. In global shutter release modes, the exposure times of individual pixels can vary.



In global shutter release modes (described later), exposure time starts simultaneously for all rows, but still ends as defined above. In a real system, the exposure would be stopped by a mechanical shutter, which would effectively stop the exposure to all rows simultaneously. Since this specification does not consider the effect of an external shutter, each output row's exposure time will differ by t_{ROW} from the previous row.

Global shutter modes also introduce a constant added to the shutter time for each row, since the exposure starts during the global shutter sequence, and not during any row's shutter sequence.

In Bulb_Exposure modes (also detailed later), the exposure time is determined by the width of the TRIGGER pulse rather than the shutter width registers. In ERS bulb mode, it will still be a multiple of row times, and the shutter overhead equation still applies. In GRR bulb mode, the exposure time is granular to ACLKs, and shutter overhead (and thus Shutter_Delay) have no effect.

Operating Modes

In the default operating mode, the MT9M003 continuously samples and outputs frames. It can be put in snapshot or triggered mode by setting snapshot, which means that it samples and outputs a frame only when triggered. To leave snapshot mode, it is necessary to first clear snapshot then issue a restart.

When in snapshot mode, the sensor can use the ERS or the GRR. The exposure can be controlled as normal, with the Shutter_Width_Lower and Shutter_Width_Upper registers, or it can be controlled using the external TRIGGER signal. The various operating modes are summarized in Table 8.

Table 8: Operating Modes

Mode	Settings	Description
ERS Continuous	Default	Frames are output continuously at the frame rate defined by t_{FRAME} . ERS is used, and the exposure time is electronically controlled to be t_{EXP} .
ERS Snapshot	Snapshot = 1	Frames are output one at a time, with each frame initiated by a trigger. ERS is used, and the exposure time is electronically controlled to be t_{EXP} .
ERS Bulb	Snapshot = 1; Bulb_Exposure = 1	Frames are output one at a time, with each frame's exposure initiated by a trigger. ERS is used. End of exposure and readout are initiated by a second trigger.
GRR Snapshot	Snapshot = 1; Global_Reset = 1	Frames are output one at a time, with each frame initiated by a trigger. GRR is used. Readout is electronically triggered based on SW.
GRR Bulb	Snapshot = 1; Bulb_Exposure = 1; Global_Reset = 1	Frames are output one at a time, with each frame initiated by a trigger. GRR is used. Readout is initiated by a second trigger.

Notes: 1. In ERS Bulb mode, SW must be greater than 4 (use trigger wider than $t_{ROW} * 4$).

All operating modes share a common set of operations:

8. Wait for the first trigger, then start the exposure.
9. Wait for the second trigger, then start the readout.

The first trigger is by default automatic, producing continuous images. If snapshot is set, the first trigger can either be a low level on the TRIGGER pin or writing a "1" to the trigger register field. If Invert_Trigger is set, the first trigger is a high level on TRIGGER pin (or a "1" written to Trigger register field). Since TRIGGER is level-sensitive, multiple frames can be output (with a frame rate of t_{FRAME}) by holding TRIGGER pin at the triggering level.



The second trigger is also normally automatic, and generally occurs SW row times after the exposure is started. If Bulb_Exposure is set, the second trigger can either be a high level on TRIGGER or a write to Restart. If Invert_Trigger is set, the second trigger is a low level on TRIGGER (or a Restart). In bulb modes, the minimum possible exposure time depends on the mechanical shutter used.

After one frame has been output, the chip will reset back to step 1 above, eventually waiting for the first trigger again. The next trigger may be issued after $((VB - 8) \times t_{ROW})$ in ERS modes or $t_{ALLREST}$ in GRR modes.

The choice of shutter type is made by Global_Reset. If it is set, the GRR shutter is used; otherwise, ERS is used. The two shutters are described in “Electronic Rolling Shutter” on page 27 and “Global Reset Release” on page 27.

The default ERS continuous mode is shown in Figure 4 on page 11. Figure 20 shows default signal timing for ERS snapshot modes, while Figure 21 on page 30 shows default signal timing for GRR snapshot modes.

Figure 20: ERS Snapshot Timing

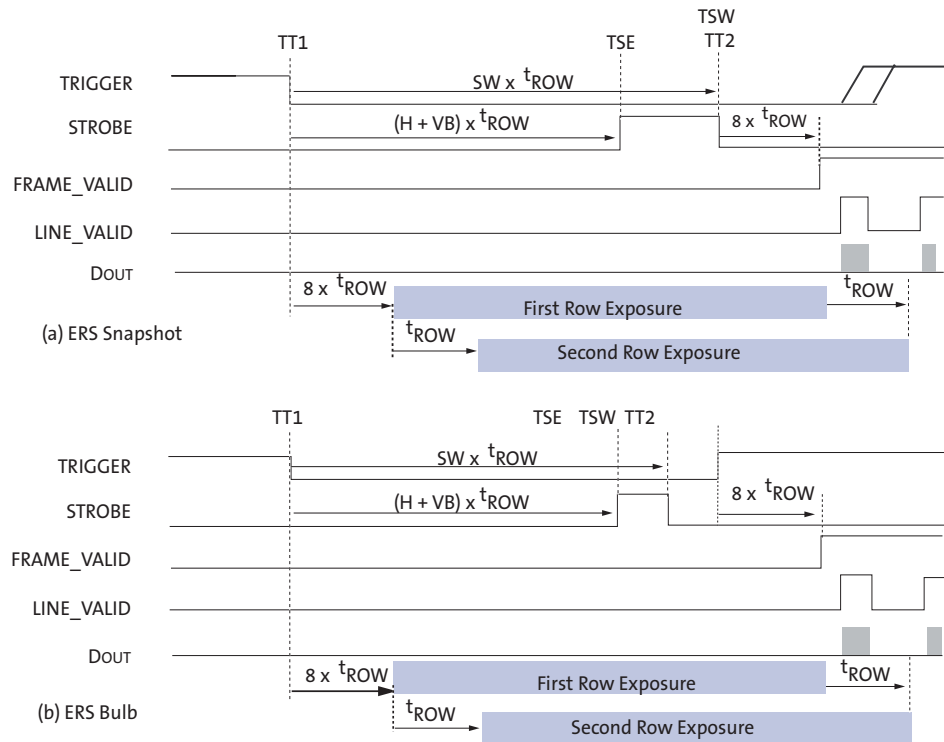
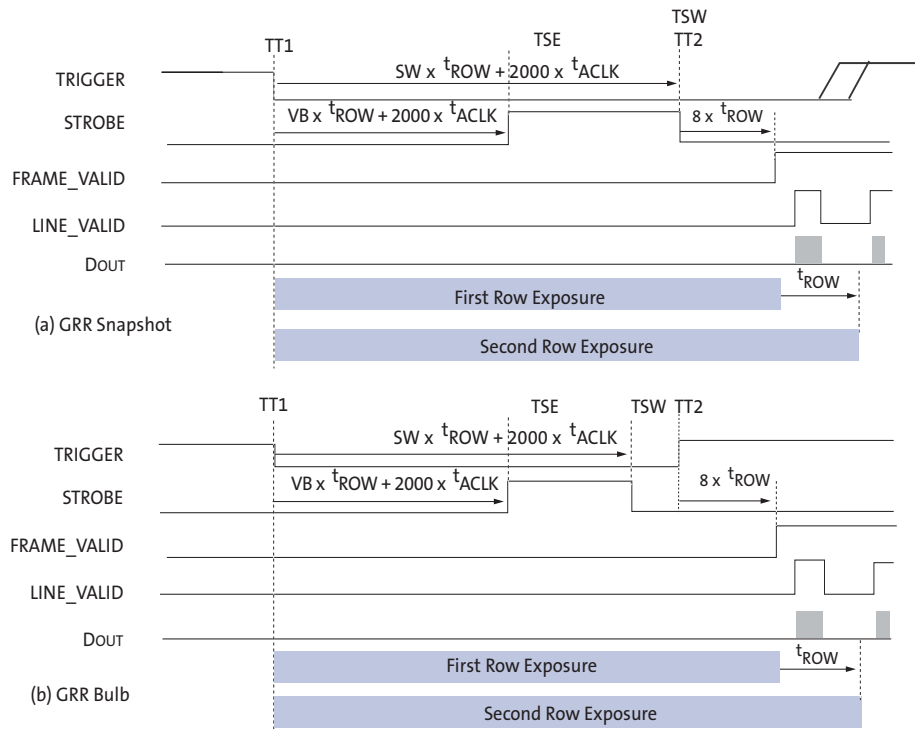




Figure 21: GRR Snapshot Timing



Strobe Control

To support synchronization of the exposure with external events such as a flash or mechanical shutter, the MT9M003 produces a STROBE output. By default, this signal is asserted for approximately the time that all rows are simultaneously exposing, minus the vertical blank time, as shown in Figure 20 on page 29 and Figure 21. Also indicated in these figures are the leading and trailing edges of STROBE, which can be configured to occur at one of several timepoints. The leading edge of STROBE occurs at STROBE_Start, and the trailing edge at STROBE_End, which are set to codes described in Table 9.

Table 9: STROBE Timepoints

Symbol	Timepoint	Code
TT1	Trigger 1 (start of shutter scan)	–
TSE	Start of exposure (all rows simultaneously exposing) offset by VB	1
TSW	End of shutter width (expiration of the internal shutter width counter)	2
TT2	Trigger 2 (start of readout scan)	3

If STROBE_Start and STROBE_End are set to the same timepoint, the strobe is a t_{ROW} wide pulse starting at the STROBE_Start timepoint. If the settings are such that the strobe would occur after the trailing edge of FV, the strobe may be only t_{ACLK} wide; however, since there is no concept of a row at that time. The sense of the STROBE signal can be inverted by setting Invert_Strobe (R0x1E[5] = 1). To use strobe as a flash in snapshot modes or with mechanical shutter, set the Strobe_Enable register bit field (R0x1E[4] = 1).



Timing Specifications

Power-Up Sequence

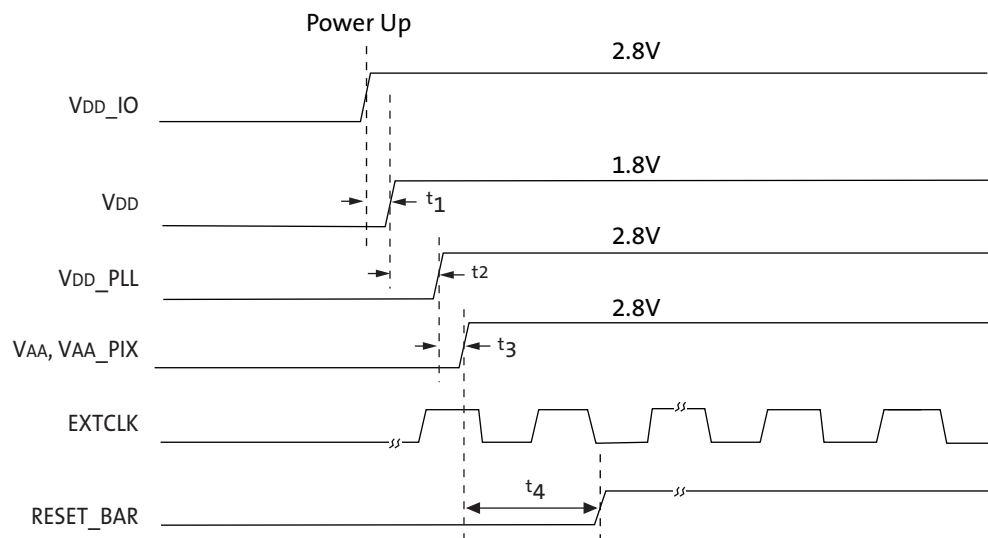
Use the following sequence when powering up the MT9M003:

1. Ensure RESET_BAR is asserted (driven LOW).
2. Bring up all the power supplies at the same time. If both the analog and the digital supplies cannot be brought up simultaneously, ensure the digital supply comes up first. Ensure that all power rails reach minimum voltages.
3. De-assert RESET_BAR (driven HIGH).
4. After reset, the sensor must be activated to generate output image data. To active it, the user must load a set of initial file settings. The simplest set of power on initialial-izations setting is:

```

REG = 0, 0x1E, 0xC006    // Set parallel mode
REG = 0, 0x9F, 0x3070    // Parallel data and clock out
REG = 0, 0x9E, 0x111E    // FV_LV timing adjustment
REG = 0, 0x0B, 0x0001    // restart
DELAY = 100
REG = 0, 0x0B, 0x0000    //restart
  
```

Figure 22: Power Supply Power-Up Sequence



- Notes:
1. The LINE_VALID signal must be connected to an external pull-down resistor (typically from 10k–100k Ω).
 2. The dotted lines are drawn in reference to the minimum voltage of the power supply or minimum V_{IH} for RESET_BAR. Please refer Table 17 on page 41 for DC electrical specifications.
 3. After all power rails reach their minimum voltage value, RESET_BAR should stay at LOW at least one millisecond. At least one stable EXTCLK input is required before RESET_BAR is released.

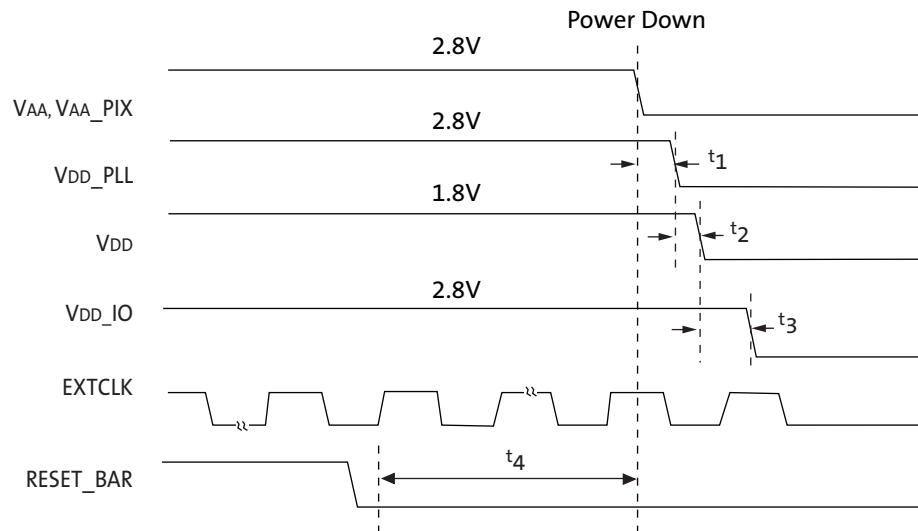
**Table 10: Power Supply Power-Up Timing**

Parameter	Symbol	Min	Typ	Max	Units
VDD_IO to VDD	t_1	0	–	500	ms
VDD to VDD_PLL	t_2	0	–	500	
VDD_PLL to VAA, VAA_PIX	t_3	0	–	500	
Reset activation	t_4	1	–	–	

Power-Down Sequence

Follow this sequence to power down the sensor. See Figure 23 for detailed timing.

1. Assert RESET_BAR (driven LOW).
2. Remove all power supplies simultaneously or at least within the timing parameters specified in Table 11.

Figure 23: Power Supply Power-Down Sequence**Table 11: Power Supply Power-Down Timing**

Parameter	Symbol	Min	Typ	Max	Units
VAA, VAA_PIX to VDD_PLL	t_1	0	–	500	ms
VDD_PLL to VDD	t_2	0	–	500	
VDD to VDD_IO	t_3	0	–	500	
Reset activation	t_4	1	–	–	



Reset

Two types of reset are available:

- A hard reset is issued by toggling RESET_BAR.
- A soft reset is issued by writing commands through the serial interface.

Hard Reset

Assert (LOW) RESET_BAR and apply at least one EXTCLK pulse. All registers return to the factory defaults. When the signal is de-asserted (HIGH), the chip resumes normal operation.

Soft Reset

A soft reset to the sensor has the same affect as the hard reset and can be activated by setting the register field to "1": R0x0D[0] = 1.

All registers except the following will be reset:

- Chip_Enable
- Synchronize_Changes
- Reset
- PLL_m_Factor
- PLL_n_Divider

When the field is returned to "0," the chip resumes normal operation.

Signal State During Reset

Table 12 shows the state of the signal interface during reset (when RESET_BAR is asserted) and during standby (after exit from Reset and before any registers within the sensor have been changed from their default power-up values).

Table 12: Signal State During Reset

Signal Name	Signal Type	Reset Signal State
SCLK	Input	Input
RESET_BAR	Input	Input
EXTCLK	Input	Input
TRIGGER	Input	Input
TEST	Input	Input
SDATA	I/O	Input
STROBE	Output	Tri-state
DOUT[11:0]	Output	Output
PIXCLK	Output	High
FRAME_VALID	Input	Input
LINE_VALID	Input	Input



Standby and Chip Enable (Power Save Mode)

The MT9M003 can be put in a low-power standby state from streaming state by programming R0x07[1]. Two standby modes (STBY_A and STBY_B) are selectable through R0x10[13:12]. Conditions are shown in Table 13. When the sensor is put in standby, all internal clocks are gated, and analog circuitry is put in a state that it draws minimal power.

The two-wire serial interface remains minimally active so that the Chip_Enable bit can subsequently be cleared. READs cannot be performed and only the Chip_Enable register is writable.

If the sensor was in continuous mode when put in standby, it resumes from where it was when standby was deactivated. For maximum power savings in standby mode, EXTCLK should not be toggling. When standby mode is entered, the PLL is disabled automatically or powered down. It must be manually re-enabled when leaving standby as needed.

Note: STBY_B is for master mode in the system, which keeps to output sync (FV / LV) signals. STBY_A is for both modes.

To enter standby STBY_A:

1. Set R0x027[7] = 1
2. Set R0x094[0] = 1
3. Set R0x00B = 0x0003
4. Set R0x007[1] = 0

To enter standby STBY_B:

Set R0x00A[12:13] = 1

To leave standby STBY_A:

1. Set R0x027[7] = 0
2. Set R0x094[0] = 0
3. Set R0x00B = 0x0000
4. Set R0x007[1] = 1
5. Set R0x010[1] = 0
6. Set R0x010[1] = 1

To leave standby STBY_B:

Set R0x00A[12:13] = 0

Table 13: Standby Modes

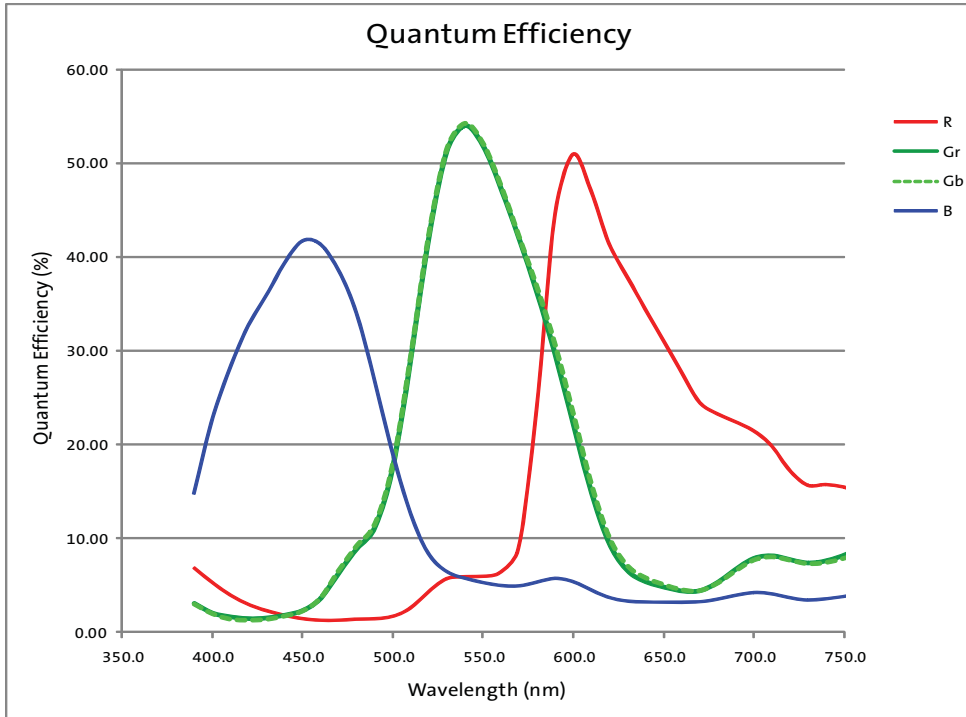
Circuit Type	STBY_A	STBY_B
Analog core	Disable	Disable
Digital data pass array control	Disable	Enable
Digital AC	Disable	Enable
Digital CLK Gen (Gated Clock Ctrl)	Enable (master clock bypass)	Enable
PLL	Disable	Enable

Note: The execution of standby will take place after the completion of the current line by default.



Spectral Characteristics

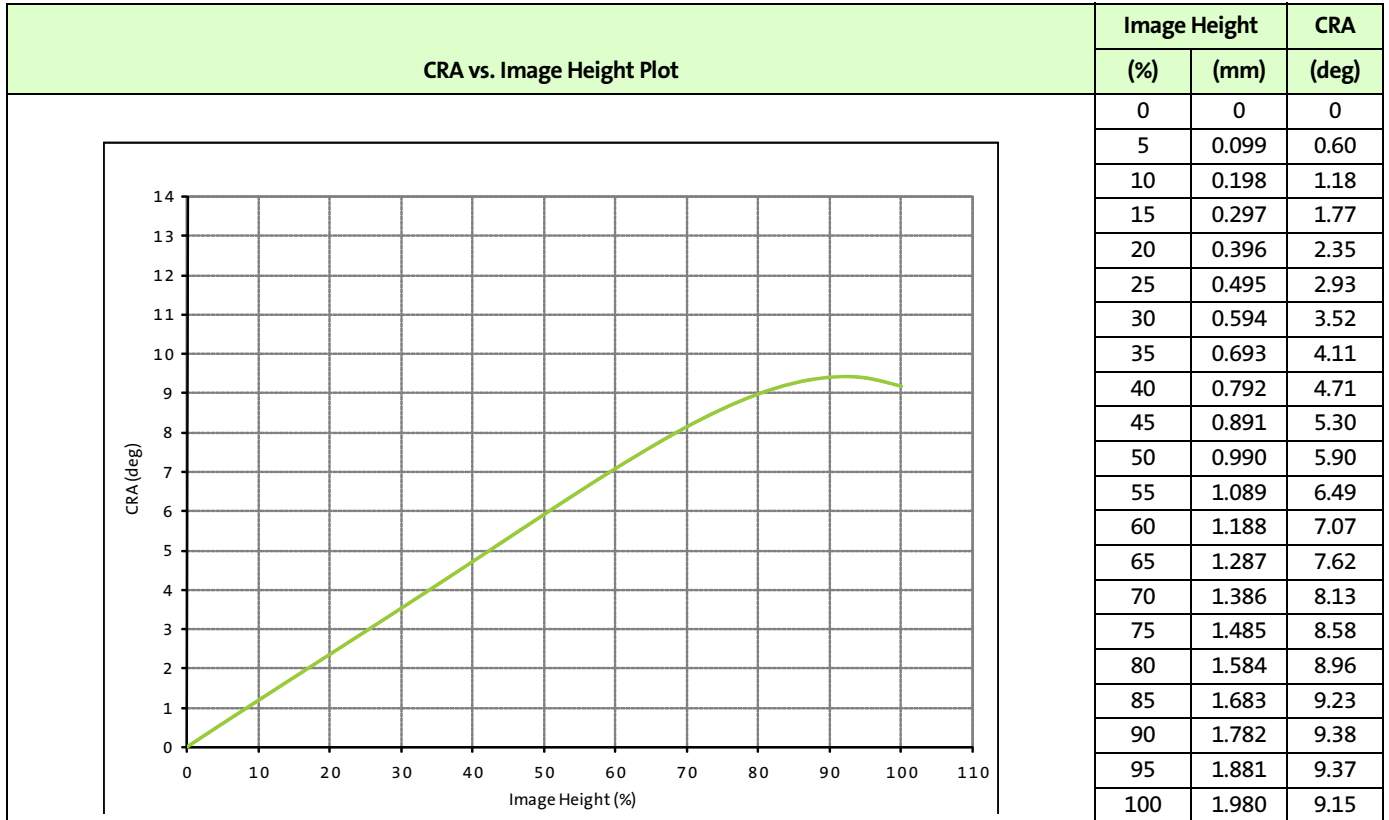
Figure 24: Typical Color Spectral Characteristics





CRA Characteristics

Figure 25: Chief Ray Angle (CRA) vs. Image Height



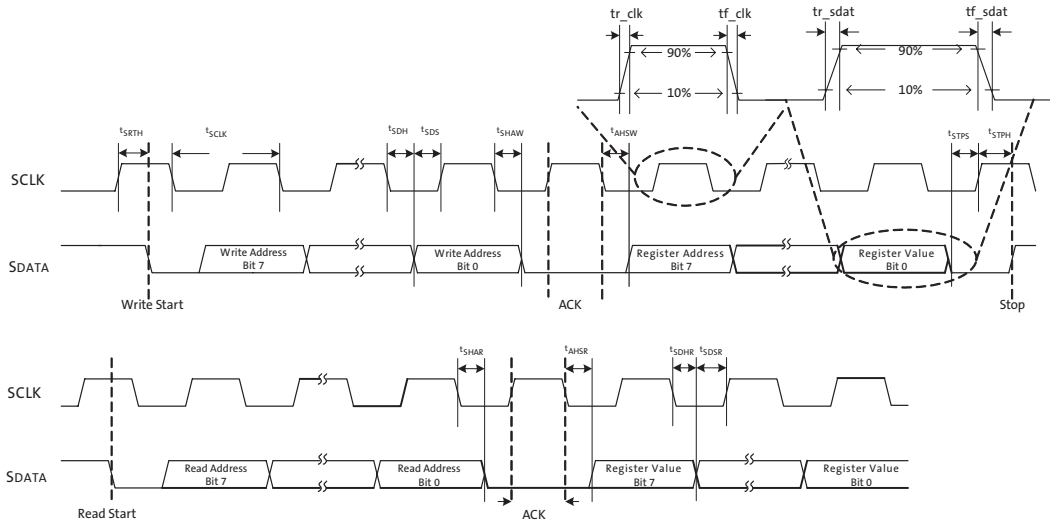


Electrical Specifications

Two-Wire Serial Register Interface

The electrical characteristics of the two-wire serial register interface (SCLK, SDATA) are shown in Figure 26 and Table 14.

Figure 26: Two-Wire Serial Bus Timing Parameters



Note: Read sequence: For an 8-bit READ, read waveforms start after WRITE command and register address are issued.

Table 14: Two-Wire Serial Bus Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Unit
f_{SCLK}	Serial interface input clock frequency	—	—	—	400	kHz
t_{SCLK}	Serial Input clock period	—	2.5	—	—	μ s
	SCLK duty cycle	—	40	50	60	%
t_{r_sclk}	SCLK rise time		—		300	ns
t_{f_sclk}	SCLK fall time		—		300	ns
t_{r_sdat}	SDATA rise time		—		300	ns
t_{f_sdat}	SDATA fall time		—		300	ns
t_{SRTH}	Start hold time	WRITE/READ	600			ns
t_{SDH}	SDATA hold	WRITE				ns
t_{SDS}	SDATA setup	WRITE				ns
t_{SHAW}	SDATA hold to ACK	WRITE				ns
t_{AHSW}	ACK hold to SDATA	WRITE				ns
t_{STPS}	Stop setup time	WRITE/READ				ns
t_{STPH}	Stop hold time	WRITE/READ				ns
t_{SHAR}	SDATA hold to ACK	READ				ns
t_{AHSR}	ACK hold to SDATA	READ				ns
t_{SDHR}	SDATA hold	READ				ns
t_{SDSR}	SDATA setup	READ				ns



Table 14: Two-Wire Serial Bus Characteristics (continued)

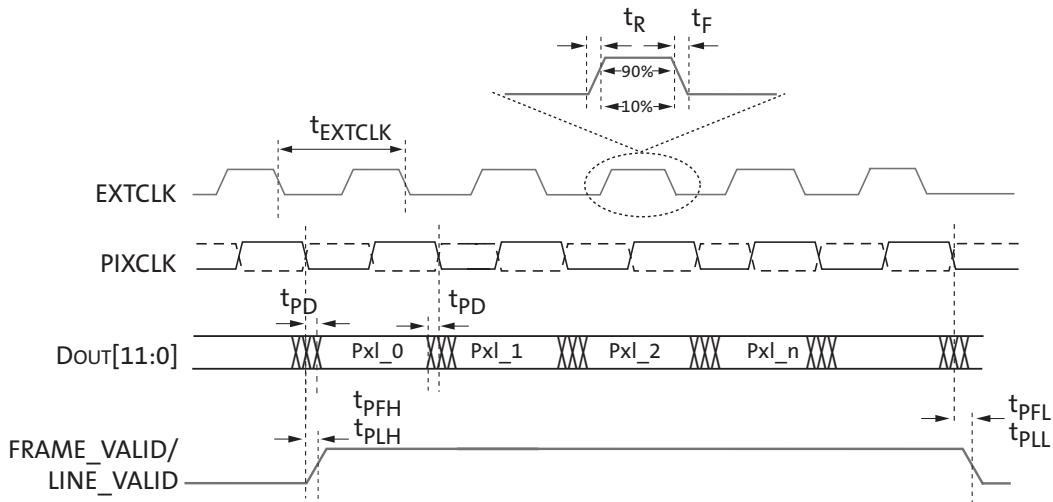
Symbol	Parameter	Condition	Min	Typ	Max	Unit
CIN_SI	Serial interface input pin capacitance	–	–	–	–	pF
CLOAD_SD	SDATA MAX load capacitance	–	–	–	15	pF
RSD	SDATA pull-up resistor	–	–	1500	–	Ω

I/O Timing

By default, the MT9M003 launches pixel data, FRAME_VALID and LINE_VALID with the rising edge of PIXCLK. The expectation is that the user captures DOUT[11:0], FRAME_VALID and LINE_VALID using the falling edge of PIXCLK.

See Figure 27 and Table 15 on page 39 for I/O timing (AC) characteristics.

Figure 27: Parallel I/O Timing Diagram



Note: PLL disabled for t_{CP} .



Table 15: I/O Timing Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Unit
t_{EXTCLK}	Input clock frequency	PLL disabled	8	–	16.5	MHz
t_{EXTCLK}	Input clock period	PLL disabled	60.6	–	125	ns
f_{PIXCLK}	Output clock frequency	PLL enabled	8	–	99	MHz
t_{PIXCLK}	Output clock period	PLL enabled	10.10	–	125	ns
t_{R^1}	Input clock rise time	54–99 MHz, PIXCLK, HIGH–LOW voltage, midlevel condition	1	–	60	ns
t_{F^1}	Input clock fall time		1	–	60	ns
t_{RP^1}	PIXCLK rise time		2.4	–	3	ns
t_{FP^1}	PIXCLK fall time		2.2	–	3	ns
	EXTCLK_Dutycycle		40	50	60	%
	PIXCLK_Duty cycle	99 MHz, midlevel condition (V_{DD} , EXTCLK duty cycle were varied)	40	50	60	%
$t_{(PIX\ JITTER)^2}$	Jitter on PIXCLK	HIGH-LOW voltage, midlevel condition	0.5	0.7	1	ps
t_{JITTER^2}	Input clock jitter @ 8 MHz		–	–	–	ps
t_{JITTER^2}	Input clock jitter @ 16.5 MHz		–	–	–	ps
t_{CP}	EXTCLK to PIXCLK propagation delay	54–99 MHz PixClk, HIGH–LOW voltage, midlevel condition	12.6	14.4	16	ns
f_{PIXCLK}	PIXCLK frequency		54	–	99	MHz
t_{PD}	PIXCLK to data valid		0.6	1.5	2.3	ns
t_{PFH}	PIXCLK to FV HIGH		1.3	1.8	2.2	ns
t_{PLH}	PIXCLK to LV HIGH		0.5	0.7	1.0	ns
t_{PFL}	PIXCLK to FV LOW		1.4	2.1	2.6	ns
t_{PLL}	PIXCLK to LV LOW		0.5	0.7	1.0	ns
C_{LOAD}^3	Output load capacitance		–	6.5	–	pF
C_{IN}^3	Input pin capacitance		–	2.5	–	pF

- Notes:
- EXTCLK 16.5 MHz (min rise = 5ns, max rise = 6.3 ns) V_{PP} = 2.3V midpoint 1.9V.
 - Value equal to jitter on tester.
 - Based on boards currently used for testing.



DC Electrical Characteristics

The DC electrical characteristics are shown in Table 16.

Table 16: DC Electrical Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Unit
VDD	Core digital voltage		1.7	1.8	1.9	V
VDD_IO	I/O digital voltage		2.6	2.8	3.1	V
VAA	Analog voltage		2.6	2.8	3.1	V
VAA_PIX	Pixel supply voltage		2.6	2.8	3.1	V
VDD_PLL	PLL supply voltage		2.6	2.8	3.1	V
VIH	Input HIGH voltage	VDD_IO = 2.8V	2.0	–	3.3	V
VIL	Input LOW voltage	VDD_IO = 2.8V	–0.3	–	0.8	V
IIN	Input leakage current	No pull-up resistor; VIN = VDD_IO or DGND	0.0176	0.5305	1.1425	μA
VOH	Output HIGH voltage	At specified IOH	2.17	2.68	3.05	V
VOL	Output LOW voltage	At specified IOL	0.2	0.28	0.39	V
IOH	Output HIGH current	At specified VOH	–0.014	–	0.014	mA
IOL	Output LOW current	At specified VOL	0.012	–	0.016	mA
IOZ	Tri-state output leakage current	VIN = VDD_IO or GND	–	143	250	nA
IDD	Digital operating current	Streaming, full resolution	20	28.0	30	mA
IDD_IO	I/O digital operating current	Streaming, full resolution	25	27.3	50	mA
IAA	Analog operating current	Streaming, full resolution	60	65.0	100	mA
IAA_PIX	Pixel supply current	Streaming, full resolution	0	2.6705	4	mA
IDD_PLL	PLL supply current	Streaming, full resolution	1	3.0	5	mA
ISTBY_A_OFF	Soft standby current	–	0	0.48	1.6	mA
ISTBY_A_ON		–	0	1.93	3.2	mA
ISTBY_B_OFF		–	0	6.53	8.8	mA
ISTBY_B_ON		–	0	40.65	55.7	mA

Table 17: Power Consumption – Parallel
(at 30 fps, full resolution, 25°C)

Symbol	Parameter	Typ Current (mA)	Typ Voltage (V)	Power Parallel (mW)
PVDD	Digital operating power	28.0	1.8	50.4
PVDDIO1	I/O digital operating power	7.7	2.8	21.6
PVDDIO2 (parallel)	I/O power parallel	19.6	2.8	86.5
PVAA	Analog operating power	65.0	2.8	182.0
PVAAPIX	PLL supply power	5.6	2.8	15.7
PVDDPLL	PLL supply power	3.0	2.8	8.4
PTOTAL	Total power			364.6



Absolute Maximum Ratings

Caution Stresses greater than those listed in Table 18 may cause permanent damage to the device. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

Table 18: Absolute Maximum Values

Symbol	Parameter	Condition	Min	Max	Unit
VDD_MAX	Core digital voltage		-0.3	1.9	V
VDD_IO_MAX	I/O digital voltage		-0.3	3.1	V
VAA_MAX	Analog voltage		-0.3	3.1	V
VAA_PIX_MAX	Pixel supply voltage		-0.3	3.1	V
VDD_PLL_MAX	PLL supply voltage		-0.3	3.1	V
VIN_MAX	Input HIGH voltage		-0.3	VDD_IO + 0.3	V
IDD_MAX	Digital operating current	Worst case current		28.3	mA
IDD_IO_MAX	I/O digital operating current	Worst case current		54.4	mA
IAA_MAX	Analog operating current	Worst case current		103	mA
IAA_PIX_MAX	Pixel supply current	Worst case current		11.6	mA
IDD_PLL_MAX	PLL supply current	Worst case current		7.3	mA
IDD_LVDS_MAX	LVDS operating current	Worst case current			
TOP	Operating temperature	Measure at junction	-30	70	°C
TSTG	Storage temperature		-40	85	°C

Note: This is a stress rating only, and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

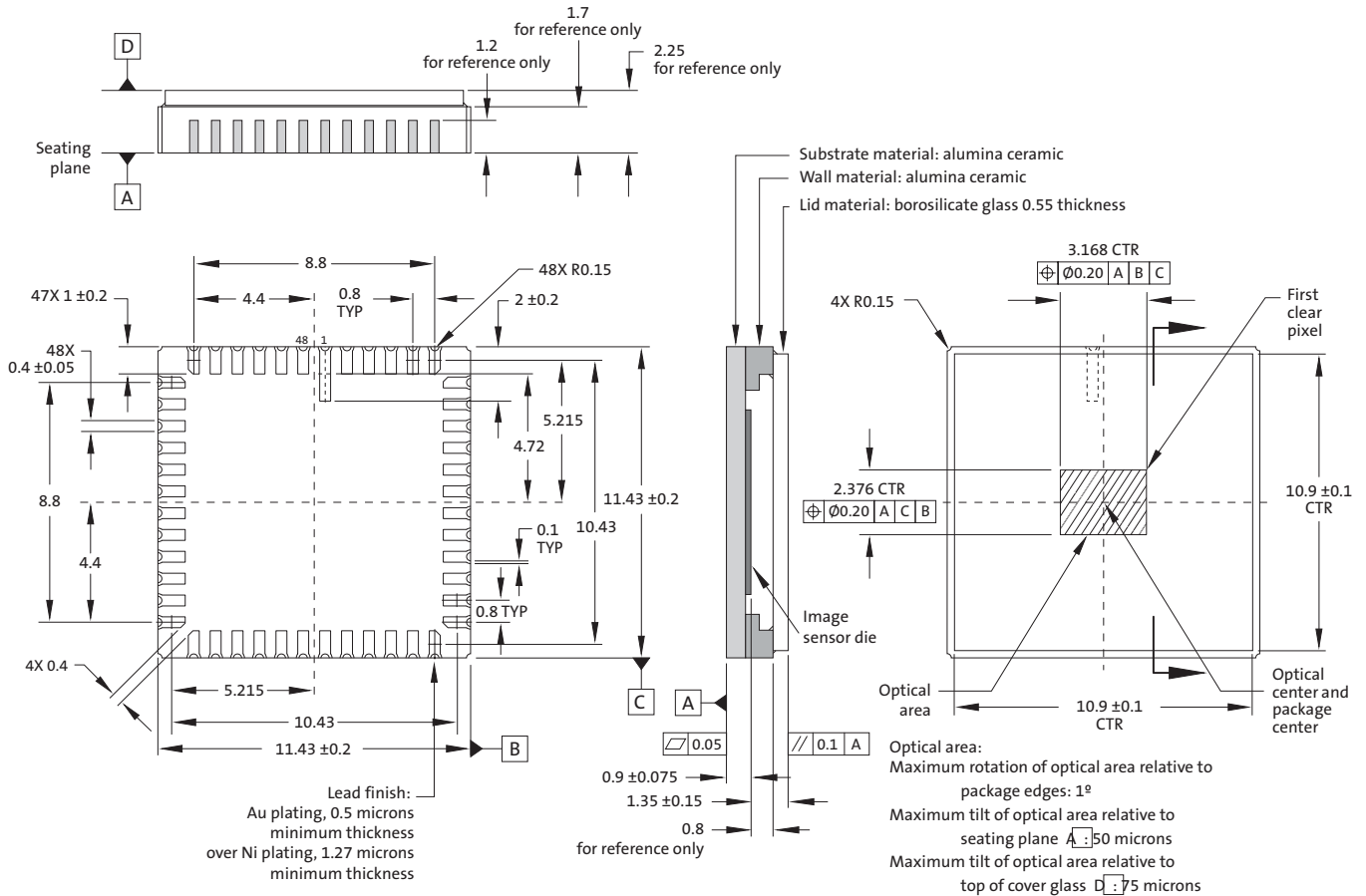


MT9M003: 1/4.5-Inch 1.6Mp CMOS Digital Image Sensor Package Dimensions

Package Dimensions

The 48-pin CLCC package mechanical drawing is shown in Figure 28.

Figure 28: 48-Pin CLCC Package Outline



The optical center is aligned with package center as origin.



Revision History

Rev. B		4/21/10
	<ul style="list-style-type: none"> • Updated Table 1, “Key Performance Parameters,” on page 1 • Updated Figure 2: “48-Pin CLCC 11.43 x 11.43 Package Pinout Diagram (Top View),” on page 9 • Updated Figure 24: “Typical Color Spectral Characteristics,” on page 35 	
Rev. A		3/31/10
	<ul style="list-style-type: none"> • Initial release 	

Sales: Shenzhen Sunnywale Inc, www.sunnywale.com , awin@sunnywale.com , Wechat: 9308762

10 Eunos Road 8 13-40, Singapore Post Center, Singapore 408600 prodmtg@aptina.com www.aptina.com
Aptina, Aptina Imaging, DigitalClarity, and the Aptina logo are the property of Aptina Imaging Corporation
All other trademarks are the property of their respective owners.

Preliminary: This data sheet contains initial characterization limits that are subject to change upon full characterization of production devices.