1/3-INCH 2-MEGAPIXEL CMOS ACTIVE-PIXEL DIGITAL IMAGE SENSOR

PART NUMBERS: MT9D011W00STC
MT9D011D00STC

Features

- DigitalClarity™ CMOS Imaging Technology
- High frame rate
- Superior low-light performance
- Low dark current
- Simple two-wire serial interface
- Auto black level calibration
- Support for long integration times
- 2 x 2 binning
- Anti-aliasing function
- Anti-eclipse function
- Operating modes: snapshot and flash control, high frame rate preview, electronic panning
- Programmable controls: gain, frame size/rate, exposure, left-right and top-bottom image reversal, window size, panning, zoom and decimation
- 10-bit analog-to-digital converter (ADC) with three external inputs
- Support for external mechanical shutter
- Internal master clock generated from on-chip phase locked loop (PLL)
- Electronic rolling shutter (ERS)

Applications

- Cellular phones
- PC cameras
- PDAs
- Toys and other battery-powered products

Table 1: Key Performance Parameters

PARAMETER		TYPICAL VALUE
Optical Format		1/3-inch (4:3)
Active Imager Size		4.48mm(H) x 3.36mm(V), 5.60mm Diagonal
Active Pixels		1600H x 1200V
Pixel Size		2.8µm x 2.8µm
Color Filter Array		RGB Bayer Pattern
Shutter Type		Electronic Rolling Shutter (ERS)
Maximum Data Rate/ Master Clock		40 MPS/40 MHz
Frame Rate	UXGA (full frame, 1600H x 1200V)	15 fps at 36 MHz
	SVGA (preview, 800H x 600V)	30 fps at 36 MHz
ADC Resolution		10-bit, on-chip
Responsivity		1.0 V/lux-sec (550nm)
Dynamic Range		68dB
SNR_{MAX}		42dB
Supply Voltage	I/O Digital	1.7V–3.6V
	Core Digital	1.7V–1.9V (1.8V nominal)
	Analog	2.5V–3.1V (2.8V nominal)
Power Consumption		75mW at 30 fps, 36 MHz, Preview mode
		125mW at 15 fps, (VAA, VAAPIX and VDD only) 36 MHz, Full frame mode
Operating Temperature		-30°C to +70°C
Packaging		Wafer or die

Table of Contents

List of Figures

List of Tables

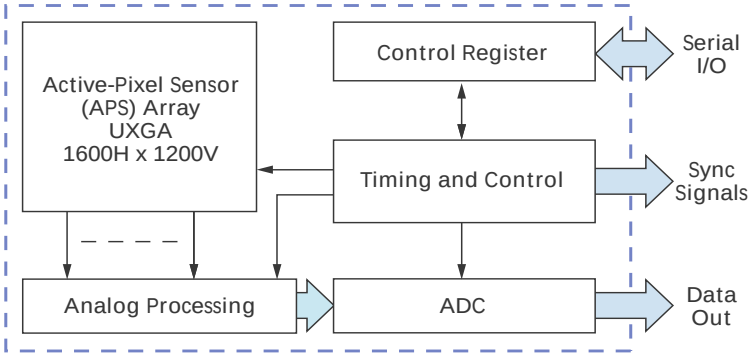
General Description

The Micron® Imaging MT9D011 is an oversize UXGA-format CMOS active-pixel digital image sensor with a pixel array of 1632H x 1216V. It incorporates sophisticated on-chip camera functions such as windowing, mirroring, row skip modes, and snapshot mode. It is programmable through a simple two-wire serial interface and achieves very low power consumption.

The 2-megapixel CMOS image sensor features DigitalClarity—Micron's breakthrough low-noise CMOS imaging technology that achieves CCD image quality (based on signal-to-noise ratio and low-light sensitivity) while maintaining the inherent size, cost, and integration advantages of CMOS.

When operated in default mode, the sensor generates a UXGA image at 15 frames per second (fps). An on-chip ADC generates a 10-bit value for each pixel. The pixel data is output on a 10-bit output bus and qualified by an output data clock (PIXCLK), together with LINE_VALID and FRAME_VALID signals. A FLASH output strobe is provided to allow an external Xenon or LED light source to synchronize with the sensor exposure time. The sensor can be programmed by the user to control the frame size, exposure, gain setting, and other parameters.

Figure 1: Block Diagram



Introduction

The MT9D011 is a progressive-scan sensor that generates a stream of pixel data qualified by LINE_VALID and FRAME_VALID signals. An on-chip PLL generates the master clock from an input clock of 4 MHz to 40 MHz. In default mode, the data rate (pixel clock) is the same as the master clock frequency, which means that one pixel is generated every master clock cycle. The sensor block diagram is shown in Figure 1.

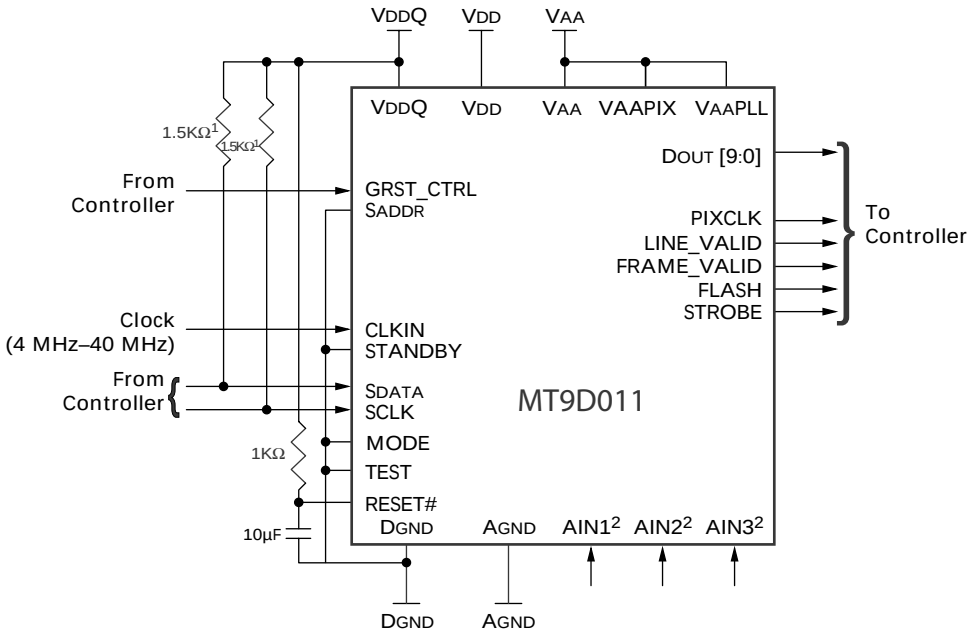
The core of the sensor is an active-pixel array. The timing and control circuitry sequences through the rows of the array, resetting and then reading each row. In the time interval between resetting a row and reading that row, the pixels in that row integrate incident light. The exposure is controlled by varying the time interval between reset and readout. After a row is 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 10-bit value for each pixel in the array. The pixel array contains optically active and light-shielded "black" pixels. The black pixels are used to provide data for on-chip offset correction algorithms ("black level" control).

The sensor contains a set of 16-bit control and status registers that can be used to control many aspects of the sensor operations. These registers can be accessed through a two-wire serial interface. In this document, registers are specified either by name (e.g., column start) or by register address (e.g., Reg0x04). Fields within a register are specified by bit or by bit range (e.g., Reg0x20[0] or Reg0x0B[13:0]). The control and status registers are described in "Registers" on page 17.

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

The sensor generates a UXGA-sized image by default, with 10 parallel data outputs per pixel, and separate LINE_VALID, FRAME_VALID, and pixel clock outputs. All timing control is performed on-chip.

Figure 2: Typical Configuration



NOTE:

1. Resistor value 1.5KΩ is recommended, but may be greater for slower two-wire speed.
2. If not used, leave unconnected.

Table 2: Signal Description

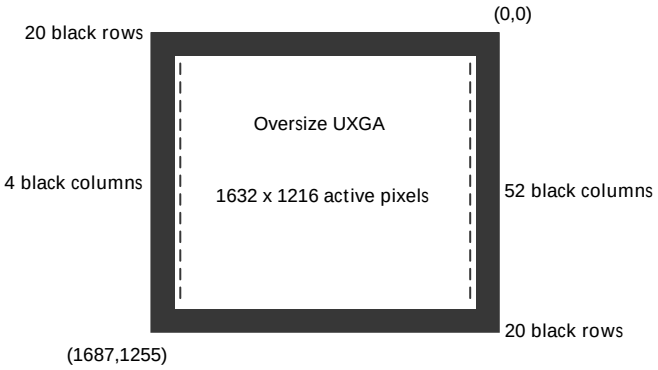
NAME	TYPE	DESCRIPTION
RESET#	Input	Asynchronous active-low reset. When asserted, data output stops and all internal registers are restored to their factory default settings.
AIN1	Input	Analog input port. If enabled by Reg0xE3, the pin is sampled once per row by the on-chip analog-to-digital converter (ADC).
AIN2	Input	Similar to AIN1.
AIN3	Input	Similar to AIN1.
SCLK	Input	Serial clock for access to control and status registers.
SADDR	Input	Selects the device address for the serial interface. See "Slave Address" on page 13.
STANDBY	Input	Multifunction pin to control device addressing, power-down, and pin tri-state functions. When LOW, the sensor functions normally. When HIGH, the sensor may enter a low-power state and may put certain outputs in a High-Z. "Power-Saving Modes" on page 53, "Output Enable Control" on page 51, and "Slave Address" on page 13.
CLKIN	Input	Input clock to PLL or master clock.
TEST	Input	Enable manufacturing test modes. Wire to DGND for functional operation.
MODE	Input	Tie to DGND for normal operation.
SDATA	I/O	Serial data for reads from and writes to control and status registers.
DOUT9	Output	Pixel data output 9 (most significant bit (MSB)).
DOUT8	Output	Pixel data output 8.
DOUT7	Output	Pixel data output 7.
DOUT6	Output	Pixel data output 6.
DOUT5	Output	Pixel data output 5.
DOUT4	Output	Pixel data output 4.
DOUT3	Output	Pixel data output 3.
DOUT2	Output	Pixel data output 2.
DOUT1	Output	Pixel data output 1.
DOUT0	Output	Pixel data output 0 (least significant bit (LSB)).
LINE_VALID	Output	LINE_VALID. Asserted during a line of valid pixel data. (The operation of this signal can be controlled by Reg0x25[15:14].)
FRAME_VALID	Output	FRAME_VALID. Asserted during a frame of valid pixel data.
PIXCLK	Output	Pixel clock. By default, pixel data, LINE_VALID, and FRAME_VALID are valid on the rising edge of this clock. This signal can be inverted and delayed under the control of Reg0x0A.
STROBE	Output	Synchronization pulse for mechanical shutter in global reset mode.
GRST_CTR	Output	Controls the global reset operation.
FLASH	Output	Synchronization pulse for external light source.
VDDQ	Power	I/O power.
VDD	Power	Digital power.
DGND	Power	Digital and I/O ground.
VAAPLL	Power	PLL power.
VAA	Power	Analog power.
AGND	Power	Analog ground.
VAAPIX	Power	Analog power for pixel array.

Pixel Array Structure

The MT9D011 pixel array is configured as 1688 columns by 1256 rows (shown in Figure 3). The first 52 columns and the first and the last 20 rows of pixels are optically black and are used for the automatic black level adjustment. The last four columns are also optically black.

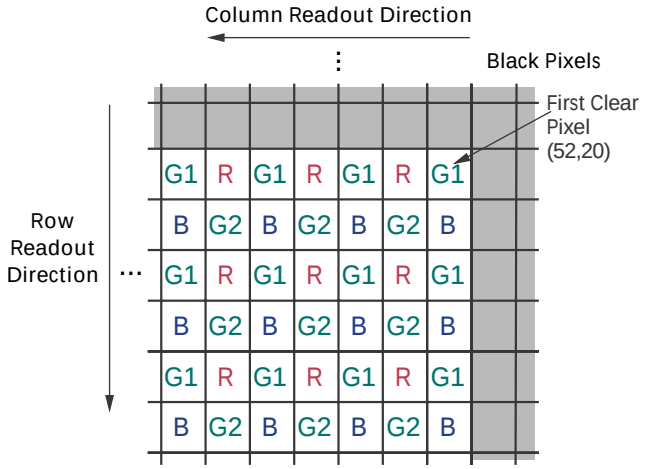
The optically active pixels are used as follows: In default mode a UXGA image (1600 columns by 1200 rows) is generated, starting at row 28, column 60. A four-pixel boundary of active pixels can be enabled around the image to avoid boundary effects during color interpolation and correction. During mirrored readout, the region of active pixels used to generate the image is offset by one pixel in each mirrored direction so that the readout always starts on the same color pixel.

Figure 3: Pixel Array



The MT9D011 uses a Bayer color pattern as shown in Figure 4. The even-numbered rows contain green and red color pixels; odd-numbered rows contain blue and green color pixels. Even-numbered columns contain green and blue color pixels; odd-numbered columns contain red and green color pixels. The color order is preserved during mirrored readout.

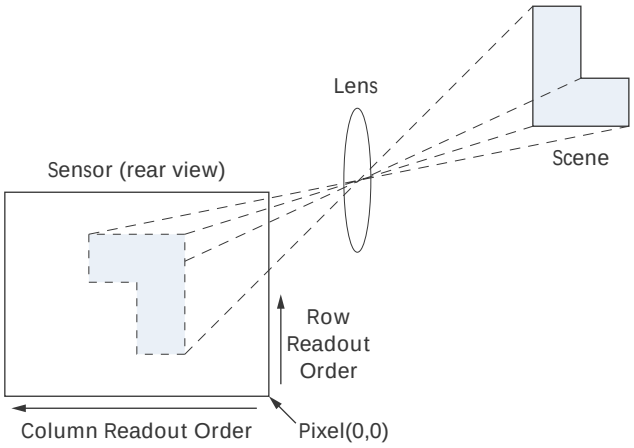
Figure 4: Pixel Color Pattern Detail (Top Right Corner)



Default Readout Order

By convention, the MT9D011 pixel array is shown with pixel (0,0) in the top right-hand corner (see Figure 4). This reflects the actual layout of the array on the die. When the sensor is imaging, the active surface of the sensor faces the scene as shown in Figure 5. 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 3. By convention, data from the sensor is shown with the first pixel read out—pixel (52,20) in the case of the MT9D011—in the top left-hand corner. See Figure 6.

Figure 5: Imaging a Scene



Output Data Format

MT9D011 image data is read out in a progressive scan. Valid image data is surrounded by horizontal blanking and vertical blanking as shown in Figure 6. The amount of horizontal 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 6: Spatial Illustration of Image Readout

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

Output Data Timing

MT9D011 output data is synchronized with the PIXCLK output. When LINE_VALID is HIGH, one pixel datum is output on the 10-bit DOUT output every PIXCLK period. By default, the PIXCLK signal runs at the same frequency as the master clock, and its rising edges occur one-half of a master clock period after transitions on LINE_VALID, FRAME_VALID, and DOUT (see Figure 7). This allows PIXCLK to be used as a clock to sample the data. PIXCLK is continuously enabled, even during the blanking period. The MT9D011 can be programmed to delay the PIXCLK edge relative to the DOUT transitions from 0 to 3.5 master clocks, in steps of one-half of a master clock. This can be achieved by programming the corresponding bits in Reg0x0A. The parameters P, A, and Q in Figure 8 are defined in Table 3 on page 12.

Figure 7: Pixel Data Timing Example

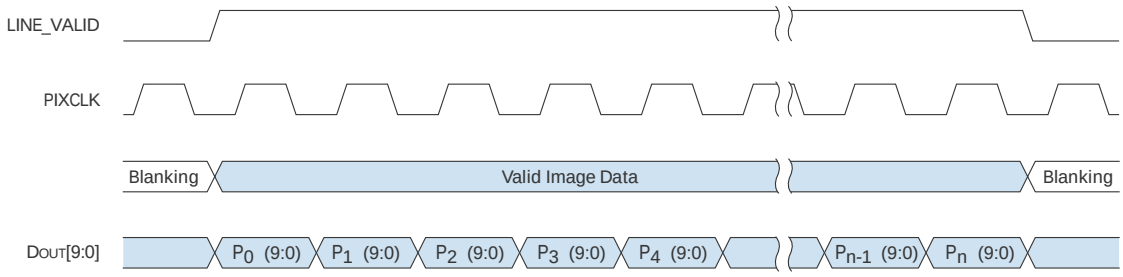
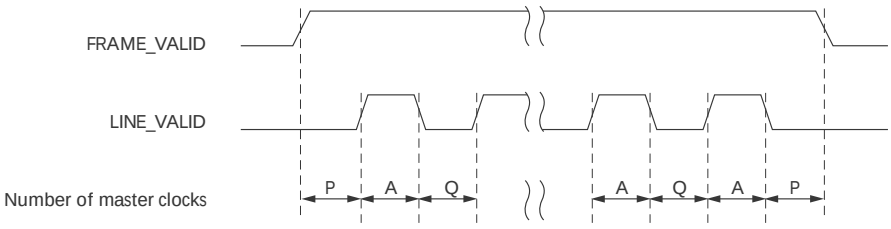


Figure 8: Row Timing and FRAME_VALID/LINE_VALID Signals



The sensor timing (Table 3) is shown in terms of pixel-clock and master-clock cycles (see Figure 7 on page 10). The recommended master clock frequency is 36 MHz. Increasing the integration time to more than one frame will cause the frame time to be extended. The equations in Table 3 assume integration time is less than the number of rows in a frame (Reg0x09 < Reg0x03/S + BORDER + VBLANK_REG). If this is not the case, the number of integration rows must be used instead to determine the frame time as shown in Table 4.

Table 3: Frame Time

PARAMETER	NAME	EQUATION	DEFAULT TIMING AT 36 MHZ, DUAL ADC MODE
HBLANK_REG	Horizontal Blanking Register	Reg0x07 if Reg0xF2[0] = 0 Reg0x05 if Reg0xF2[0] = 1	0x15C = 348 pixels
VBLANK_REG	Vertical Blanking Register	Reg0x8 if Reg0xF2[1] = 0 Reg0x6 if Reg0xF2[1] = 1	0x20 = 32 rows
ADC_MODE	ADC mode	Reg0xF2[3] = 0: Reg0x20[10] Reg0xF2[3] = 1: Reg0x21[10]	
PIXCLK_PERIOD	Pixel clock period	ADC_MODE = 0: Reg0x0A[2:0] ADC_MODE = 1: Reg0x0A[2:0]*2	1 ADC_MODE: 55.556ns 2 ADC_MODE: 27.778ns
S	Skip Factor	For skip 2x mode: S = 2 For skip 4x mode: S = 4 For skip 8x mode: S = 8 For skip 16x mode: S = 16 otherwise, S = 1	1
A	Active Data Time	(Reg0x04/S) * PIXCLK_PERIOD	1,600 pixel clocks = 1,600 master = 44.44µs
P	Frame Start/End Blanking	6 * PIXCLK_PERIOD (can be controlled by Reg0x1F)	6 pixel clocks = 12 master = 0.166µs
Q	Horizontal Blanking	HBLANK_REG * PIXCLK_PERIOD	348 pixel clocks = 348 master = 9.667µs
A + Q	RowTime	((Reg0x04/S) + HBLANK_REG) * PIXCLK_PERIOD	1,948 pixel clocks = 1,948 master = 54.112µs
V	Vertical Blanking	VBLANK_REG * (A + Q) + (Q - 2*P)	62,672 pixel clocks = 62,672 master = 1.741ms
Nrows * (A+Q)	Frame Valid Time	(Reg0x03/S) * (A + Q) - (Q - 2*P)	2,337,264 pixel clocks = 2,337,264 master = 64.925ms
F	Total Frame Time	((Reg0x03/S) + VBLANK_REG) * (A + Q)	2,399,936 pixel clocks = 2,399,936 master = 66.665ms

NOTE:
1. Skip factor should be multiplied by 2 if binning is enabled.

Table 4: Frame—Long Integration Time

PARAMETER	NAME	EQUATION (MASTER CLOCK)
V'	Vertical Blanking (long integration time)	(Reg0x09–(Reg0x03)/S)*(A + Q) + (Q - 2*P)
F'	Total Frame Time (long integration time)	(Reg0x09)*(A + Q)

Two-Wire Serial Register Interface

This section describes the two-wire serial interface bus that can be used in any functional sensor mode.

The two-wire serial interface bus enables R/W access to control and status registers within the MT9D011.

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. The master generates a clock (SCLK) that is an input to the sensor and used to synchronize transfers. The master is responsible for driving a valid logic level on SCLK at all times. Data is transferred between the master and the slave on a bi-directional signal (SDATA). The SDATA signal is pulled up to VDD off-chip by a 1.5kΩ resistor. Either the slave or master device can drive the SDATA line low—the interface protocol determines which device is allowed to drive the SDATA line at any given time.

Protocol

The two-wire serial interface bus defines the transmission codes as follows:

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

Sequence

A typical read or write sequence is executed as follows:

1. The master sends a start bit.
2. The master sends the 8-bit slave device 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.
3. The slave device acknowledges receipt of the address by sending an acknowledge bit to the master.
4. If the request is a write, the master then transfers the 8-bit register address, indicating where the write takes place.
5. The slave sends an acknowledge bit, indicating that the register address has been received.
6. The master then transfers the data, eight bits at a time, with the slave sending an acknowledge bit after each eight bits.

The MT9D011 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.

1. The master sends the write-mode slave address and 8-bit register address, just as in the write request.
2. The master then sends a start bit and the read-mode slave address, and clocks out the register data, eight bits at a time.
3. The master sends an acknowledge bit after each 8-bit transfer. The register address is auto-incremented after every 16 bits is transferred.
4. 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 start and stop bits.

Start Bit

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

Stop Bit

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

Slave Address

The 8-bit address of a two-wire serial interface device consists of seven bits of address and one bit of direction. A “0” in the LSB (least significant bit) of the address indicates write mode, and a “1” indicates read mode. The default slave addresses used by the MT9D011 are 0xBA (write address) and 0xBB (read address). Reg0x0D[10] or the SADDR pin can be used to select the alternate slave addresses 0x90 (write address) and 0x91 (read address).

Writes to Reg0x0D[10] are inhibited when the standby pin is asserted (all other writes proceed normally). This allows two sensors to co-exist as slaves on this interface, but they must be addressed independently. Enable this capability as follows:

After RESET, both sensors use the default slave address. Reads or writes on the serial register interface to the default slave address are decoded by both sensors simultaneously.

1. After RESET, assert the STANDBY signal to one sensor and negate the STANDBY signal to the other sensor.
2. Perform a write to Reg0x0D with bit 10 set. The sensor with STANDBY asserted ignores the write to bit 10 and continues to decode at the default slave address.

The sensor with STANDBY negated has its Reg0x0D[10] set and responds to the alternate slave address for all subsequent read and write operations, as shown in See Table 5.

Table 5: Slave Address Options

SADDR PIN	REG0XD[10]	SLAVE ADDRESS	
		WRITE	READ
0	0	0x090	0x091
0	1	0x0BA	0x0BB
1	0	0x0BA	0x0BB
1	1	0x090	0x091

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 eight 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.

Page Register

The MT9D011 two-wire serial interface and its associated protocols support an address space of 256 16-bit locations. This address space can be extended by a 3-bit page prefix, and controlled through accesses to Reg0xF0.

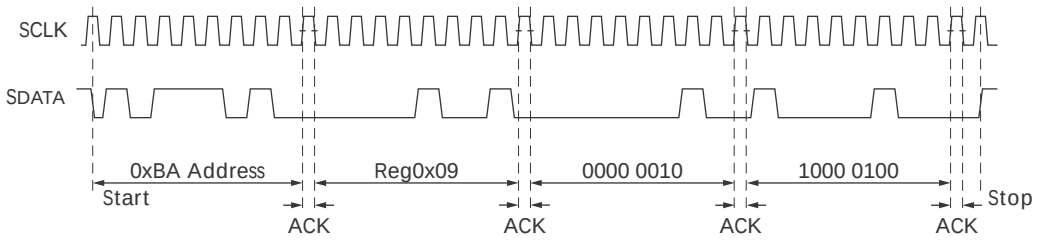
The paging mechanism is intended to allow access to other sets of registers when the sensor is embedded as part of a more complex integrated subsystem, for example, in an SOC. All registers within the MT9D011 are accessible on page 0 (the default page).

Sample Write and Read Sequences

16-Bit Write Sequence

A typical write sequence for writing 16 bits to a register is shown in Figure 9. A start bit given by the master starts the sequence, followed by the write address. 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.

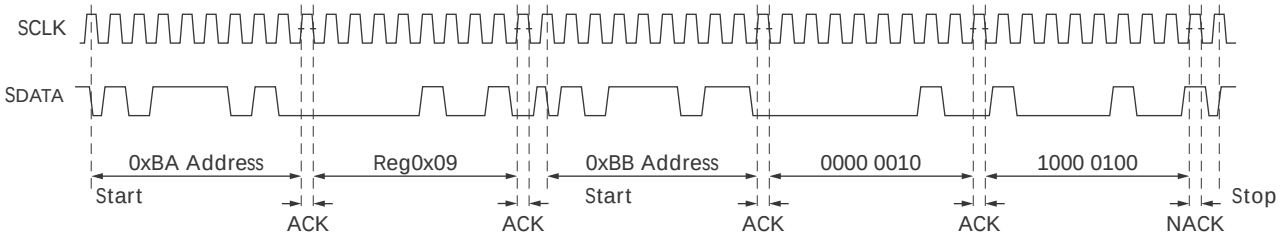
Figure 9: WRITE Timing to R0x09:0—Value 0x0284



16-Bit Read Sequence

A typical read sequence is shown in Figure 10. First the master writes 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 clocks out the register data, eight bits at a time. The master sends an acknowledge bit after each 8-bit transfer. The register address should be incremented after every 16 bits is transferred. The data transfer is stopped when the master sends a no-acknowledge bit.

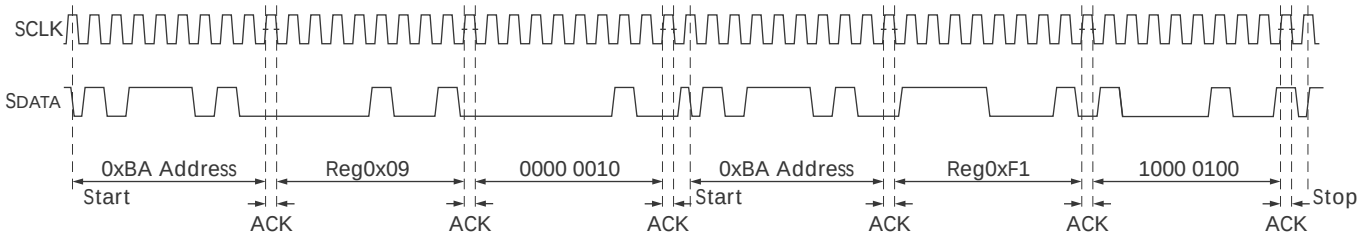
Figure 10: READ Timing from R0x09:0; Returned Value 0x0284



8-Bit Write Sequence

To be able to write one byte at a time to the register, a special register address is added. The 8-bit write is done by writing the upper 8 bits to the desired register, then writing the lower 8 bits to the special register address (Reg0xF1). The register is not updated until all 16 bits have been written. It is not possible to update just half of a register. In Figure 11, a typical sequence for 8-bit writes is shown. The second byte is written to the special register (Reg0xF1).

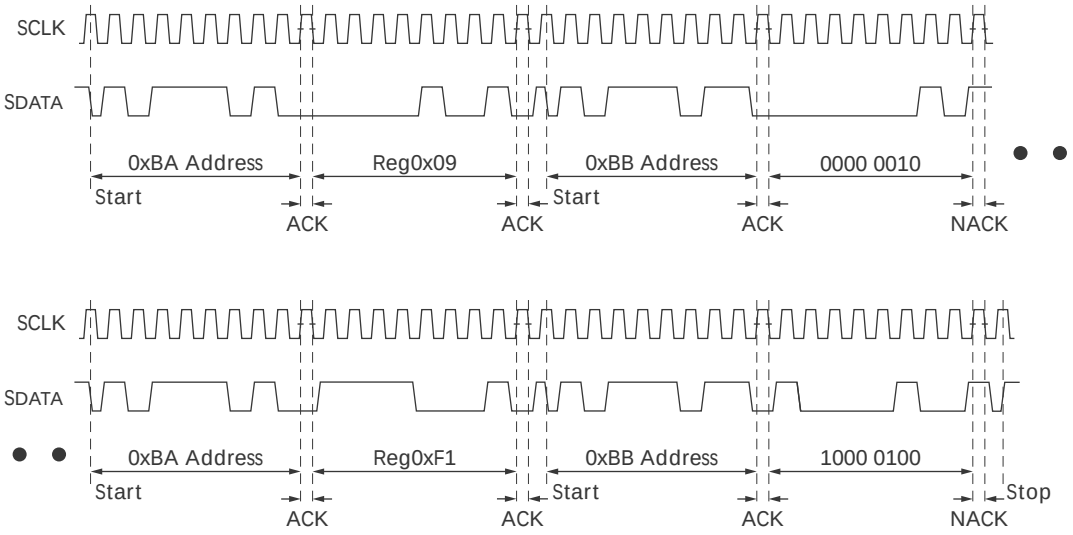
Figure 11: WRITE Timing to R0x09:0—Value 0x0284



8-Bit Read Sequence

To read one byte at a time, the same special register address is used for the lower byte. The upper 8 bits are read from the desired register. By following this with a read from the special register (Reg0xF1), the lower 8 bits are accessed (Figure 12). The master sets the no-acknowledge bits shown.

Figure 12: READ Timing from Reg0x09; Returned Value 0x0284



Registers

The MT9D011 provides a register address space of 256 locations.

Register Map

Table 6 shows the locations used within the address space. Locations that are not shown in the table are reserved for future use; they should not be read from or written to. Locations that are shown in the table as Reserved are not to be read from or written to. The effect of doing so is UNDEFINED and may include the possibility of causing permanent electrical damage to the sensor.

Table 6: Register List and Default Value Descriptions

REGISTER #DEC (HEX)	DESCRIPTION	DATA FORMAT (BINARY)	DEFAULT VALUE (HEX)
0 (0x00)	Chip Version	0001 0101 0001 0001 (LSB)	0x1511
1 (0x01)	Row Start	0000 0ddd dddd dddd	0x001C
2 (0x02)	Column Start	0000 0ddd dddd dddd	0x003C
3 (0x03)	Row Width	0000 0ddd dddd dddd	0x04B0
4 (0x04)	Column Width	0000 0ddd dddd dddd	0x0640
5 (0x05)	Horizontal Blanking – Context B	00dd dddd dddd dddd	0x015C
6 (0x06)	Vertical Blanking – Context B	0ddd dddd dddd dddd	0x0020
7 (0x07)	Horizontal Blanking—Context A	00dd dddd dddd dddd	0x00AE
8 (0x08)	Vertical Blanking—Context A	0ddd dddd dddd dddd	0x0010
9 (0x09)	Shutter Width	dddd dddd dddd dddd	0x04D0
10 (0x0A)	Row Speed	ddd0 0000 dddd 0ddd	0x0011
11 (0x0B)	Extra Delay	00dd dddd dddd dddd	0x0000
12 (0x0C)	Shutter Delay	00dd dddd dddd dddd	0x0000
13 (0x0D)	Reset	d000 0ddd dddd 0ddd	0x0000
31 (0x1F)	FRAME_VALID Control	dddd dddd dddd dddd	0x0000
32 (0x20)	Read Mode—Context B	d0dd dddd dddd dddd	0x0000
33 (0x21)	Read Mode—Context A	d000 0d00 dddd dd00	0x0490
34 (0x22)	Dark Columns/Rows	0000 0ddd dddd dddd	0x010F
35 (0x23)	Flash	??dd dddd dddd dddd	0x0608
36 (0x24)	Extra Reset	dd00 0000 0000 0000	0x8000
37 (0x25)	LINE_VALID Control	dd00 0000 0000 0000	0x0000
38 (0x26)	Bottom Dark Rows	0000 0000 dddd dddd	0x0007
43 (0x2B)	Green1 Gain	0000 dddd dddd dddd	0x0020
44 (0x2C)	Blue Gain	0000 dddd dddd dddd	0x0020
45 (0x2D)	Red Gain	0000 dddd dddd dddd	0x0020
46 (0x2E)	Green2 Gain	0000 dddd dddd dddd	0x0020
47 (0x2F)	Global Gain	0000 dddd dddd dddd	0x0020
48 (0x30)	Row Noise	dddd dddd dddd dddd	0x042A
49 (0x31)	Reserved	—	0x0000
50 (0x32)	Reserved	—	0x02AA
51 (0x33)	Reserved	—	0x0341
52 (0x34)	Reserved	—	0x000F
53 (0x35)	Reserved	—	0x0EE8
54 (0x36)	Reserved	—	0xF0F0
56 (0x38)	Reserved	—	0x0808
59 (0x3B)	Reserved	—	0x0020
60 (0x3C)	Reserved	—	0x2020

Table 6: Register List and Default Value Descriptions (continued)

REGISTER #DEC (HEX)	DESCRIPTION	DATA FORMAT (BINARY)	DEFAULT VALUE (HEX)
61 (0x3D)	Reserved	—	0x2000
62 (0x3E)	Reserved	—	0x0020
63 (0x3F)	Reserved	—	0x1000
63 (0x40)	Reserved	—	0x0000
65 (0x41)	Reserved	—	0x00D7
66 (0x42)	Reserved	—	0x0077
86 (0x56)	Reserved	—	0x87FF
87 (0x57)	Reserved	—	0x0002
88 (0x58)	Reserved	—	0x0000
89 (0x59)	Black Rows	0000 0000 dddd dddd	0x00FF
90 (0x5A)	Reserved	—	0xE2DF
91 (0x5B)	Dark G1 Average	0000 0000 0??? ????	N/A
92 (0x5C)	Dark B Average	0000 0000 0??? ????	N/A
93 (0x5D)	Dark R Average	0000 0000 0??? ????	N/A
94 (0x5E)	Dark G2 Average	0000 0000 0??? ????	N/A
95 (0x5F)	Calib Threshold	0ddd dddd 0ddd dddd	0x231D
96 (0x60)	Calib Control	d00d 0ddd dddd dddd	0x0080
97 (0x61)	Calib Green1	0000 000d dddd dddd	0x0000
98 (0x62)	Calib Blue	0000 000d dddd dddd	0x0000
99 (0x63)	Calib Red	0000 000d dddd dddd	0x0000
100 (0x64)	Calib Green2	0000 000d dddd dddd	0x0000
101 (0x65)	Clock Control	ddd0 0000 0000 dddd	0xe000
102 (0x66)	PLL Control 1	dddd dddd 00dd dddd	0x2809
103 (0x67)	PLL Control 2	0000 dddd 0ddd dddd	0x0501
110 (0x6E)	Reserved	—	0xAD01
111 (0x6F)	Reserved	—	0x893F
112 (0x70)	Reserved	—	0xB502
113 (0x71)	Reserved	—	0xB502
114 (0x72)	Reserved	—	0x2B03
115 (0x73)	Reserved	—	0x1A12
116 (0x74)	Reserved	—	0x8850
117 (0x75)	Reserved	—	0x705F
118 (0x76)	Reserved	—	0xAC57
119 (0x77)	Reserved	—	0x4F28
120 (0x78)	Reserved	—	0xB301
121 (0x79)	Reserved	—	0xB309
122 (0x7A)	Reserved	—	0xB00F
123 (0x7B)	Reserved	—	0xFF00
124 (0x7C)	Reserved	—	0xB601
125 (0x7D)	Reserved	—	0xAE56
126 (0x7E)	Reserved	—	0x00B7
127 (0x7F)	Reserved	—	0xAD1A
128 (0x80)	Reserved	—	0x8402
129 (0x81)	Reserved	—	0x8404
130 (0x82)	Reserved	—	0x8103

Table 6: Register List and Default Value Descriptions (continued)

REGISTER #DEC (HEX)	DESCRIPTION	DATA FORMAT (BINARY)	DEFAULT VALUE (HEX)
131 (0x83)	Reserved	—	0x261C
132 (0x84)	Reserved	—	0x6D06
133 (0x85)	Reserved	—	0x3423
134 (0x86)	Reserved	—	0x8701
135 (0x87)	Reserved	—	0x0088
136 (0x88)	Reserved	—	0x6E03
137 (0x89)	Reserved	—	0x8201
144 (0x90)	Reserved	—	0x7B6F
145 (0x91)	Reserved	—	0xFFFF
146 (0x92)	Reserved	—	0x715E
147 (0x93)	Reserved	—	0x8A79
148 (0x94)	Reserved	—	0x897A
149 (0x95)	Reserved	—	0x2319
150 (0x96)	Reserved	—	0xFFFF
151 (0x97)	Reserved	—	0x1B11
152 (0x98)	Reserved	—	0xAE21
153 (0x99)	Reserved	—	0xAD22
160 (0xA0)	Reserved	—	0x4733
161 (0xA1)	Reserved	—	0xFFFF
162 (0xA2)	Reserved	—	0x3522
163 (0xA3)	Reserved	—	0x6F45
164 (0xA4)	Reserved	—	0x6E46
165 (0xA5)	Reserved	—	0x3525
166 (0xA6)	Reserved	—	0xFFFF
167 (0xA7)	Reserved	—	0x271B
168 (0xA8)	Reserved	—	0x8333
169 (0xA9)	Reserved	—	0x8234
176 (0xB0)	Reserved	—	0x1A00
177 (0xB1)	Reserved	—	0x1901
178 (0xB2)	Reserved	—	0x1802
179 (0xB3)	Reserved	—	0x1A00
180 (0xB4)	Reserved	—	0x1901
181 (0xB5)	Reserved	—	0x1802
182 (0xB6)	Reserved	—	0x002C
183 (0xB7)	Reserved	—	0x001A
192 (0xC0)	Global Shutter Control	d000 0000 0000 0ddd	0x0000
193 (0xC1)	Start Integration (T1)	dddd dddd dddd dddd	0x0064
194 (0xC2)	Start Readout (T2)	dddd dddd dddd dddd	0x0064
195 (0xC3)	Assert Strobe (T3)	dddd dddd dddd dddd	0x0096
196 (0xC4)	De-assert Strobe (T4)	dddd dddd dddd dddd	0x00C8
197 (0xC5)	Assert Flash	dddd dddd dddd dddd	0x0064
198 (0xC6)	De-assert Flash	dddd dddd dddd dddd	0x0078
199 (0xC7)	Reserved	—	0x4E20
200 (0xC8)	Reserved	—	0x0258
201 (0xC9)	Reserved	—	0x1F40

Table 6: Register List and Default Value Descriptions (continued)

REGISTER #DEC (HEX)	DESCRIPTION	DATA FORMAT (BINARY)	DEFAULT VALUE (HEX)
202 (0xCA)	Reserved	—	0x001E
208 (0xD0)	Reserved	—	0x001C
209 (0xD1)	Reserved	—	0x003C
210 (0xD2)	Reserved	—	0x04BC
211 (0xD3)	Reserved	—	0x0654
212 (0xD4)	Reserved	—	0x00B0
224 (0xE0)	External Sample 1	0000 00?? ???? ????	N/A
225 (0xE1)	External Sample 2	0000 00?? ???? ????	N/A
226 (0xE2)	External Sample 3	0000 00?? ???? ????	N/A
227 (0xE3)	External Sampling Control	dd00 0000 0000 0000	0x0000
240 (0xF0)	Page Register	0000 0000 0000 0ddd	0x0000
241 (0xF1)	Bytewise Address	0000 0000 0000 0000	0x0000
242 (0xF2)	Context Control	d000 0000 d000 dddd	0x000B
245 (0xF5)	Reserved	—	0x07FF
246 (0xF6)	Reserved	—	0x07FF
247 (0xF7)	Reserved	—	0x0000
248 (0xF8)	Reserved	—	0x0000
249 (0xF9)	Reserved	—	0x0000
250 (0xFA)	Reserved	—	0x0000
251 (0xFB)	Reserved	—	0x0000
252 (0xFC)	Reserved	—	0x0000
253 (0xFD)	Reserved	—	0x0000
255 (0xFF)	Chip Version	0001 0101 0001 0001	0x1511

NOTE:

1 = always 1
0 = always 0
d = programmable
? = read-only (R/O)
R/W = Read/Write

Register Description

Table 7 provides a detailed description of the registers. Bit fields that are not identified in the table are read only.

Double-Buffered Registers

Some sensor settings cannot be changed during frame readout. For example, changing row width Reg0x03 part way through frame readout results in inconsistent LINE_VALID behavior. To avoid this, the MT9D011 double buffers many registers by implementing a "pending" and a "live" version. Reads and writes access the pending register. The live register controls the sensor operation.

The value in the pending register is transferred to a live register at a fixed point in the frame timing, called "frame start." Frame start is defined as the point at which the first dark row is read out. By default, this occurs ten row times before FRAME_VALID goes high. Reg0x22 enables the dark rows to be shown in the image, but this has no effect on the position of frame start.

To determine which registers or register fields are double-buffered in this way, see Table 7, the "sync'd-to-frame-start" column.

Reg0x0D[15] can be used to inhibit transfers from the pending to the live registers. This control bit should be used when making many register changes that must take effect simultaneously.

Bad Frames

A bad frame is a frame where all rows do not have the same integration time, or where offsets to the pixel values changed during the frame.

Many changes to the sensor register settings can cause a bad frame. For example, when row width Reg0x03 is changed, the new register value does not affect sensor behavior until the next frame start. However, the frame that would be read out at that frame-start has been integrated using the old row width. Consequently, reading it out using the new row width results in a frame with an incorrect integration time.

By default, most bad frames are masked: LINE_VALID and FRAME_VALID are inhibited for these frames so that the vertical blanking time between frames is extended by the frame time.

To determine which register or register field changes can produce a bad frame, see Table 7, the "bad frame" column, and these notations:

- N—No. Changing the register value does not produce a bad frame.
- Y—Yes. Changing the register value might produce a bad frame.
- YM—Yes; but the bad frame is masked out unless the "show bad frames" feature (Reg0x0D[8]) is enabled.

Changes to Integration Time

If the integration time (Reg0x09) is changed while FRAME_VALID 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 Reg0x09 pending register.
2. At the start of frame (*n+1*), the new integration time is transferred to the Reg0x09 live register.

Integration for each row of frame (*n+1*) has been completed using the old integration time. 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*). The actual time that rows start integrating using the new integration time is dependent on the new value of the integration time.

3. When frame (*n+1*) is read out, it is integrated using the new integration time.

If the integration time is changed (Reg0x09 written) on successive frames, each value written is applied to a single frame; the latency between writing a value and it affecting the frame readout remains at two frames.

Changes to Gain Settings

When the gain settings (Reg0x2B, Reg0x2C, Reg0x2D, Reg0x2E, and Reg0x2F) are changed, the gain is usually updated on the next frame start. When the integration time and the gain are changed simultaneously, 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.

Table 7: Register Description

BIT FIELD	DESCRIPTION		DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
R0—0X00 - CHIP VERSION (R/O)					
Bits 15:0	Chip Version	Chip version.	1511		
R1—0X01 - ROW START (R/W)					
Bits 10:0	Row Start	The first row to be read out, excluding any dark rows that may be read. To window the image down, set this register to the starting "Y" value. Setting a value less than 20 is not recommended because the dark rows should be read using Reg0x22.	1C	Y	YM
R2—0X02 - COLUMN START (R/W)					
Bits 10:0	Column Start	The first column to be read out, excluding dark columns that may be read. To window the image down, set this register to the starting X value. Setting a value below 52 is not recommended because readout of dark columns should be controlled by Reg0x22.	3C	Y	YM
R3—0X03 - ROW WIDTH (R/W)					
Bits 10:0	Row Width	Number of rows in the image to be read out, excluding any dark rows or border rows that may be read. The minimum supported value is 2.	4B0	Y	YM
R4—0X04 - COLUMN WIDTH (R/W)					
Bits 10:0	Column Width	Number of columns in image to be read out, excluding any dark columns or border columns that may be read. The minimum supported value is 9 in 1 ADC mode and 17 in 2 ADC mode.	640	Y	YM
R5—0X05 - HORIZONTAL BLANKING—CONTEXT B (R/W)					
Bits 13:0	Horizontal Blanking—Context B	Number of blank columns in a row when Context B is selected (Reg0xF2[0] = 1). The extra columns are added at the beginning of a row. "Frame Rate Control" on page 43 for more information on supported register values.	15C	Y	YM
R6—0X06 - VERTICAL BLANKING—CONTEXT B (R/W)					
Bits 14:0	Vertical Blanking—Context B	Number of blank rows in a frame when Context B is selected (Reg0xF2[1] = 1). The minimum supported value is (4 + Reg0x22[2:0]). The actual vertical blanking time may be controlled by the shutter width (Reg0x9); see "Output Data Timing" on page 10.	20	Y	N
R7—0X07 - HORIZONTAL BLANKING—CONTEXT A (R/W)					
Bits 13:0	Horizontal Blanking—Context A	Number of blank columns in a row when Context A is selected (Reg0xF2[0] = 0). The extra columns are added at the beginning of a row. "Frame Rate Control" on page 43 for more information on supported register values.	AE	Y	YM

Table 7: Register Description (continued)

BIT FIELD		DESCRIPTION	DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
R8—0X08 - VERTICAL BLANKING—CONTEXT A (R/W)					
Bits 14:0	Vertical Blanking—Context A	Number of blank rows in a frame when Context A is chosen (Reg0xF2[1] = 1). The minimum supported value is (4 + Reg0x22[2:0]). The actual vertical blanking time may be controlled by the shutter width (Reg0x9); see "Output Data Timing" on page 10.	10	Y	N
R9—0X09 - SHUTTER WIDTH (R/W)					
Bits 15:0	Shutter Width	Integration time in number of rows. The integration time is also influenced by the shutter delay (Reg0x0C) and the overhead time.	4D0	Y	N
R10—0X0A - ROW SPEED (R/W)					
Bits 15:14	Reserved	Do not change from default value.			
Bit 13	Reserved	Do not change from default value.			
Bit 8	Invert Pixel Clock	Invert PIXCLK. When clear, FRAME_VALID, LINE_VALID, and DOUT are set up relative to the delayed rising edge of PIXCLK. When set, FRAME_VALID, LINE_VALID, and DOUT are set up relative to the delayed falling edge of PIXCLK.	0	N	N
Bits 7:4	Delay Pixel Clock	Number of half master clock cycle increments to delay the rising edge of PIXCLK relative to transitions on FRAME_VALID, LINE_VALID, and DOUT.	1	N	N
Bit 3	Reserved	Do not change from default value. .			
Bits 2:0	Pixel Clock Speed	A programmed value of *N* gives a pixel clock period of *N* master clocks in 2 ADC mode and *2*N* master clocks in 1 ADC mode. A value of "0" is treated like (and reads back as) a value of "1."	1	Y	YM
R11—0X0B - EXTRA DELAY (R/W)					
Bits 13:0	Extra Delay	Extra blanking inserted between frames. A programmed value of *N* increases the vertical blanking time by *N* pixel clock periods. Can be used to get a more exact frame rate. It may affect the integration times of parts of the image when the integration time is less than one frame.	0	Y	N^2
R12—0X0C - SHUTTER DELAY (R/W)					
Bits 13:0	Shutter Delay	The amount of time from the end of the sampling sequence to the beginning of the pixel reset sequence. If the value in this register exceeds the row time, the reset of the row does not complete before the associated row is sampled, and the sensor does not generate an image. A programmed value of *N* reduces the integration time by (*N/2*) pixel clock periods in 1 ADC mode and by *N* pixel clock periods in 2 ADC mode.	0	Y	N

Table 7: Register Description (continued)

BIT FIELD		DESCRIPTION	DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
R13—0X0D - RESET (R/W)					
Bit 15	Synchronize Changes	By default, update of many registers are synchronized to frame start. Setting this bit inhibits this update; register changes remain pending until this bit is returned to "0." When this bit is returned to "0," all pending register updates are made on the next frame start.	0	N	N
Bit 10	Toggle SADDR	By default, the sensor serial bus responds to addresses 0xBA and 0xBB. When this bit is set, the sensor serial bus responds to addresses 0x90 and 0x91. Writes to this bit are ignored when STANDBY is asserted. "Slave Address" on page 13.	0	N	N
Bit 9	Restart Bad Frames	When set, a restart is forced to take place whenever a bad frame is detected. This can shorten the delay when waiting for a good frame because the delay, when masking out a bad frame, is the integration time rather than the full frame time.	0	N	N
Bit 8	Show Bad Frames	1: Output all frames (including bad frames). 0: Only output good frames (default). A bad frame is defined as the first frame following a change to: window size or position, horizontal blanking, pixel clock speed, zoom, row or column skip, binning, mirroring, or use of border.	0	N	N
Bit 7	Inhibit Standby	By default, asserting STANDBY places the sensor in a low-power state. Setting this bit stops STANDBY from affecting entry to or exit from the low-power state. See "Power-Saving Modes" on page 53.	0	N	N
Bit 6	Drive Pins	By default, asserting STANDBY causes the pin interface to enter a High-Z. Setting this bit stops STANDBY from contributing to output-enable control. See "Output Enable Control" on page 51.	0	N	N
Bit 5	Reserved	Do not change from default value.			
Bit 4	Output Disable	Setting this bit puts the pin interface in a High-Z. See "Output Enable Control" on page 51.			
Bit 3	Reserved	Do not change from default value.			
Bit 2	Standby	Setting this bit places the sensor in a low-power state. See "Power-Saving Modes" on page 53.	0	N	YM
Bit 1	Restart	Setting this bit causes the sensor to truncate the current frame and start resetting the first row. The delay before the first valid frame is read out is equal to the integration time. This bit is write -1 but always reads back as 0.	0	N	YM

Table 7: Register Description (continued)

BIT FIELD	DESCRIPTION		DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
Bit 0	Reset	Setting this bit puts the sensor in reset; the frame being generated is truncated and the pin interface goes to an idle state. All internal registers (except for this bit) go to the default power-up state. Clearing this bit resumes normal operation.	0	N	YM
R31—0X1F - FRAME_VALID CONTROL (R/W)					
Bit 15	Enable Early FRAME_VALID Fall	1: Enables the early disabling of FRAME_VALID as set in bits 14:8. Note that LINE_VALID is still generated for all active rows. 0: Default. FRAME_VALID goes low 6 pixel clocks after last LINE_VALID.	0	N	N
Bits 14:8	Early FRAME_VALID Fall	When enabled, the FRAME_VALID falling edge occurs within the programmed number of rows before the end of the last LINE_VALID. (1 + bits 14:8)*row time + constant (constant = 3 in default mode) The value of this field must not be larger than row width Reg0x03.	0	N	N
Bit 7	Enable Early FRAME_VALID Rise	1: Enables the early rise of FRAME_VALID as set in bits 6:0. 0: Default. FRAME_VALID goes high 6 pixel clocks before first LINE_VALID.	0	N	N
Bits 6:0	Early FRAME_VALID Rise	When enabled, the FRAME_VALID rising edge is set HIGH the programmed number of rows before the first LINE_VALID: (1 + bits 6:0)*row time + horizontal blank + constant (constant = 3 in default mode).	0	N	N
R32—0X20 - READ MODE—CONTEXT B (R/W)					
Bit 15	Binning—Context B	When Read mode Context B is selected (Reg0xF2[3] = 1): 0: Normal operation. 1: Binning enabled. See "Binning" on page 42 and "Frame Rate Control" on page 43 for a full description.	0	Y	YM
Bit 13	Zoom Enable	0: Normal operation. 1: Zoom is enabled, with zoom factor [zoom] defined in bits 12:11. In zoom mode, the pixel data rate is slowed by a factor of [zoom]. This is achieved by outputting [zoom-1] blank rows between each output row. Setting this mode allows the user to fill a window that is [zoom] times larger with interpolated data. The pixel clock speed is not affected by this operation, and the output data for each pixel is valid for [zoom] pixel clocks. Every row is followed by [zoom-1] blank rows (with their own LINE_VALID, but all data bits = 0) of equal time. The combination of this register and an appropriate change to the window sizing registers allows the user to zoom to a region of interest without affecting the frame rate.	0	Y	YM

Table 7: Register Description (continued)

BIT FIELD	DESCRIPTION		DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
Bits 12:11	Zoom	When zoom is enabled by bit 13, this field determines the zoom amount: "00"—Zoom 2x. "01"—Zoom 4x. "10"—Zoom 8x. "11"—Zoom 16x.	0	Y	YM
Bit 10	Use 1 ADC—Context B	When Read mode Context B is selected (bit 3, Reg0xF2 = 1): 0: Use both ADCs to achieve maximum speed. 1: Use 1 ADC to reduce power. Maximum readout frequency is now half the master clock frequency, and the pixel clock is automatically adjusted as described for the pixel clock speed register.	0	Y	YM
Bit 9	Show Border	This bit indicates whether to show the border enabled by bit 8. "X0"—Normal behavior, no border. "01"—Border is enabled but not shown; vertical blanking is increased by eight rows and horizontal blanking is increased by eight pixels. "11"—border is enabled and shown; FRAME_VALID time is extended by 8 rows and LINE_VALID is extended by 8 pixels. See "Pixel Border" on page 36.	0	N	N
Bit 8	Over Sized	When this bit is set, a four-pixel border is output around the active image array independent of readout mode (skip, zoom, mirror, etc.). Setting this bit adds eight to the number of rows and columns in the frame.	0	Y	YM
Bit 7	Column Skip Enable—Context B	When Read mode Context B is selected (Reg0xF2[3] = 1): 1: Enable column skip. 0: Normal readout.	0	Y	YM
Bits 6:5	Column Skip—Context B	When Read mode Context B is selected (Reg0xF2[3] = 1) and column skip is enabled (bit 7 = 1): "00"—Column Skip 2x. "01"—Column Skip 4x. "10"—Column Skip 8x. "11"—Column Skip 16x. See "Column and Row Skip" on page 38 for more information.	0	Y	YM
Bit 4	Row Skip Enable—Context B	When Read mode Context B is selected (Reg0xF2[3] = 1): 1: Enable row skip. 0: Normal readout.	0	Y	YM
Bits 3:2	Row Skip—Context B	When Read mode Context B is selected (Reg0xF2[3] = 1) and Row skip is enabled (bit 4 = 1): "00"—Row Skip 2x. "01"—Row Skip 4x. "10"—Row Skip 8x. "11"—Row Skip 16x. See "Column and Row Skip" on page 38 for more information.	0	Y	YM

Table 7: Register Description (continued)

BIT FIELD	DESCRIPTION		DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
Bit 1	Mirror Columns	Read out columns from right to left (mirrored). When set, column readout starts from column (Column Start + Column Size) and continues down to (Column Start + 1). When clear, readout starts at Column Start and continues to (Column Start + Column Size - 1). This ensures that the starting color is maintained.	0	Y	YM
Bit 0	Mirror Rows	Read out rows from bottom to top (upside down). When set, row readout starts from row (Row Start + Row Size) and continues down to (Row Start + 1). When clear, readout starts at Row Start and continues to (Row Start + Row Size - 1). This ensures that the starting color is maintained.	0	Y	YM
R33—0X21 - READ MODE—CONTEXT A (R/W)					
Bit 15	Binning—Context A	When Read mode Context A is selected (Reg0xF2[3] = 0): 0: Normal operation. 1: Binning enabled. "Binning" on page 42.	0	Y	YM
Bit 10	Use 1 ADC—Context A	When Read mode Context A is selected (Reg0xF2[3] = 0): 0: Use both ADCs to achieve maximum speed. 1: Use one ADC to reduce power. Maximum readout frequency is now half of the master clock, and the pixel clock is automatically adjusted as described for the pixel clock speed register.	1	Y	YM
Bit 7	Column Skip Enable—Context A	When Read mode Context A is selected (Reg0xF2[3] = 0): 1: Enable column skip. 0: Normal readout.	1	Y	YM
Bits 6:5	Column Skip—Context A	When Read mode Context A is selected (Reg0xF2[3] = 0) and column skip is enabled (bit 7 = 1): "00"—Column Skip 2x. "01"—Column Skip 4x. "10"—Column Skip 8x. "11"—Column Skip 16x. See "Column and Row Skip" on page 38 for more information.	0	Y	YM
Bit 4	Row Skip Enable—Context A	When Read mode Context A is selected (Reg0xF2[3] = 0): 1: Enable row skip. 0: Normal readout.	1	Y	YM
Bits 3:2	Row Skip—Context A	When Read mode Context A is selected (Reg0xF2[3] = 0) and Row skip is enabled (bit 4 = 1): "00"—Row Skip 2x. "01"—Row Skip 4x. "10"—Row Skip 8x. "11"—Row Skip 16x. See "Column and Row Skip" on page 38 for more information.	0	Y	YM
R34—0X22 - SHOW CONTROL (R/W)					
Bit 10	Number of Dark Columns	MT9D011 has 40 dark columns. 1: Read out 36 dark columns (4–39). Ignored during binning, where all 40 dark columns are used. 0: Read out 20 dark columns (4–23).	0	N	N

Table 7: Register Description (continued)

BIT FIELD	DESCRIPTION		DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
Bit 9	Show Dark Columns	When set, the 20/36 (dependent on bit 10) dark columns are output before the active pixels in a line. There is an idle period of two pixels between readout of the dark columns and readout of the active image. Therefore, when set, LINE_VALID is asserted 22 pixel times earlier than normal, and the horizontal blanking time is decreased by the same amount.	0	N	N
Bit 8	Read Dark Columns	1: Enables the readout of dark columns for use in the row-wise noise correction algorithm. The number of columns used are 40 in binning mode, and otherwise determined by bit 10. 0: When disabled, an arbitrary number of dark columns can be read out by including them in the active image. Enabling the dark columns increases the minimum value for horizontal blanking but does not affect the row time.	1	N	Y
Bit 7	Show Dark Rows	When set, the programmed dark rows is output before the active window. FRAME_VALID is thus asserted earlier than normal. This has no effect on integration time or frame rate.	0	N	N
Bits 6:4	Dark Start Address	The start address for the dark rows within the eight available rows (an offset of four is added to compensate for the guard pixels). Must be set so all dark rows read out falls in the address space 0:7.	0	N	N
Bit 3	Reserved	Do not change from default value.			
Bits 2:0	Num Dark Rows	A value of *N* causes (*n+1*) dark rows to be read out at the start of each frame when dark row readout is enabled (bit 3).	7	N	Y
R35—0X23 - FLASH CONTROL (R/W)					
Bit 15	FLASH	Reflects the current state of the FLASH output pin.	0		
Bit 14	Triggered	Indicates that the FLASH output pin is asserted for the current frame.	0		
Bit 13	Xenon Flash	Enable Xenon flash. When set, the FLASH output pin asserts for the programmed period (bits 7:0) during vertical blank. This is achieved by keeping the integration time equal to one frame, and the pulse width less than the vertical blank time.	0	Y	N[1]
Bits 12:11	Frame Delay	Delay of the flash pulse measured in frames.	0	N	N
Bit 10	End of Reset	1: In Xenon mode the flash is triggered after the resetting of a frame. 0: In Xenon mode the flash is triggered after the readout of a frame.	1	N	N
Bit 9	Every Frame	1: Flash should be enabled every frame. 0: Flash should be enabled for one frame only.	1	N	N
Bit 8	LED Flash	Enable LED flash. When set, the FLASH output pin asserts prior to the start of the resetting of a frame and remains asserted until the end of the readout of the frame.	0	Y	Y[1]

Table 7: Register Description (continued)

BIT FIELD	DESCRIPTION		DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
Bits 7:0	Xenon Count	Length of FLASH pulse when Xenon flash is enabled. The value specifies the length in units of 1024*PIXCLK cycle increments. When the Xenon count is set to its maximum value (0xFF), the FLASH pulse is automatically truncated prior to the readout of the first row, giving the longest pulse possible.	8	N	N
R36—0X24 - EXTRA RESET (R/W)					
Bit 15	Extra Reset Enable	0: Only programmed window (set by Reg0x01 through Reg0x04) and black pixels are read. 1: Two additional rows are read and reset above and below programmed window to prevent blooming to active area.	1	N	N
Bit 14	Next Row Reset	When set, and the integration time is less than one frame time, row (*n+1*) is reset immediately prior to resetting row (n). This is intended to prevent blooming across rows under conditions of very high illumination.	0	N	N
Bits 13:0	Reserved	Do not change from default value.			
R37—0X25 - LINE_VALID CONTROL (R/W)					
Bit 15	Xor LINE_VALID	1: LINE_VALID = "continuous" LINE_VALID XOR FRAME_VALID. 0: Normal LINE_VALID (default, no XORing of LINE_VALID). Ineffective if continuous LINE_VALID is set.	0	N	N
Bit 14	Continuous LINE_VALID	1: "Continuous" LINE_VALID (continue producing LINE_VALID during vertical blank). 0: Normal LINE_VALID (default, no LINE_VALID during vertical blank).	0	N	N[3]
R38—0X26 - BOTTOM DARK ROWS (R/W)					
Bit 7	Show	The bottom dark rows are visible in the image if the bit is set.	0	N	N
Bits 6:4	Start Address	Defines the start address within the eight bottom dark rows.	0	N	N
Bit 3	Enable Readout	Enable readout of the bottom dark rows.	0	N	Y
Bits 2:0	Number of Dark Rows	Defines the number of bottom dark rows to be used. (The number of rows used is the specified value +1.)	7	N	Y
R43—0X2B - GREEN1 GAIN (R/W)					
Bits 11:9	Digital Gain	Total gain = (bit 9 + 1)*(bit 10 + 1)*(bit 11 + 1)*analog gain (each bit gives 2x gain).	0	Y	N
Bits 8:7	Analog Gain	Analog gain = (bit 8 + 1)*(bit 7 + 1)*initial gain (each bit gives 2x gain).	0	Y	N
Bits 6:0	Initial Gain	Initial gain = bits 6:0*0.03125.	20	Y	N
R44—0X2C - BLUE GAIN (R/W)					
Bits 11:9	Digital Gain	Total gain = (bit 9 + 1)*(bit 10 + 1)*(bit 11 + 1)*analog gain (each bit gives 2x gain).	0	Y	N
Bits 6:0	Initial Gain	Initial gain = bits [6:0]*0.03125.	20	Y	N
Bits 8:7	Analog Gain	Analog gain = (bit 8 + 1)*(bit 7 + 1)*initial gain (each bit gives 2x gain).	0	Y	N

Table 7: Register Description (continued)

BIT FIELD		DESCRIPTION	DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
R45—0X2D - RED GAIN (R/W)					
Bits 11:9	Digital Gain	Total gain = (bit 9 + 1)*(bit 10] + 1)*(bit 11 + 1)*analog gain (each bit gives 2x gain).	0	Y	N
Bits 8:7	Analog Gain	Analog gain = (bit 8 + 1)*(bit 7 + 1)*initial gain (each bit gives 2x gain).	0	Y	N
Bits 6:0	Initial Gain	Initial gain = bits 6:0*0.03125.	20	Y	N
R46—0X2E - GREEN2 GAIN (R/W)					
Bits 11:9	Digital Gain	Total gain = (bit 9 + 1)*(bit 10 + 1)*(bit 11 + 1)*analog gain (each bit gives 2x gain).	0	Y	N
Bits 8:7	Analog Gain	Analog gain = (bit 8 + 1)*(bit 7 + 1)*initial gain (each bit gives 2x gain).	0	Y	N
Bits 6:0	Initial Gain	Initial gain = bits 6:0*0.03125.	20	Y	N
R47—0X2F - GLOBAL GAIN (R/W)					
Bits 11:0	Global Gain	This register can be used to simultaneously set all four gains. When read, it returns the value stored in Reg0x2B.	20	Y	N
R48—0X30 - ROW NOISE (R/W)					
Bit 15	Frame-wise Digital Correction	By default, the row noise is calculated and compensated for individually for each color of each row. When this bit is set, the row noise is calculated and applied for each color of each of the first two rows (two pairs of values) and the same values are applied to each subsequent row, so that new values are calculated and applied once per frame.	0	N	N
Bits 14:12	Gain Threshold	When the upper analog gain bits are equal to or larger than this threshold, the dark column average is used in the row noise correction algorithm. Otherwise, the subtracted value is determined by bit 11. This check is independently performed for each color, and is a means to turn off the black level algorithm for lower gains.	0	N	N
Bit 11	Use Black Level Average	1: Use black level frame average from the dark rows in the row noise correction algorithm for low gains. Note: this frame average was taken before the last adjustment of the offset DAC for that frame, so it might be slightly off. 0: Use mean of black level programmed threshold in the row noise correction algorithm for low gains.	0	N	Y
Bit 10	Enable Correction	1: Enable row noise cancellation algorithm. When this bit is set, the average value of the dark columns read out is used as a correction for the whole row. The dark average is subtracted from each pixel on the row, and then a constant is added (bits 9:0). 0: Normal operation.	1	N	Y
Bits 9:0	Row Noise Constant	Constant used in the row noise cancellation algorithm. It should be set to the dark level targeted by the black level algorithm plus the noise expected between the averaged values of the dark columns. The default constant is set to 42 LSB.	2A	N	Y

Table 7: Register Description (continued)

BIT FIELD		DESCRIPTION	DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
R89—0X59 - BLACK ROWS (R/W)					
Bits 7:0	Black Rows	For each bit set, the corresponding dark row (rows 0–7) are used in the black level algorithm. For this to occur, the reading of those rows must be enabled by the settings in Reg0x22.	FF	N	N
R91—0X5B - GREEN1 FRAME AVERAGE (R/O)					
Bits 6:0	Green1 Frame Average	The frame-averaged green1 black level that is used in the black level calibration algorithm.			
R92—0X5C - BLUE FRAME AVERAGE (R/O)					
Bits 6:0	Blue Frame Average	The frame-averaged blue black level that is used in the black level calibration algorithm.			
R93—0X5D - RED FRAME AVERAGE (R/O)					
Bits 6:0	Red Frame Average	The frame-averaged red black level that is used in the black level calibration algorithm.			
R94—0X5E - GREEN2 FRAME AVERAGE (R/O)					
Bits 6:0	Green2 Frame Average	The frame-averaged green2 black level that is used in the black level calibration algorithm.			
R95—0X5F - THRESHOLD (R/W)					
Bits 14:8	Upper Threshold	Upper threshold for targeted black level in ADC LSBs.	23	N	N
Bits 6:0	Lower Threshold	Lower threshold for targeted black level in ADC LSBs.	1D	N	N
R96—0X60 - CALIBRATION CONTROL (R/W)					
Bit 15	Disable Rapid Sweep Mode	Disables the rapid sweep mode in the black level algorithm. The averaging mode remains enabled.	0	Y	N
Bit 12	Recalculate	When set, the rapid sweep mode is triggered if enabled, and the running frame average is reset to the current frame average. This bit is write-1, but always reads back as 0.	0	Y	N
Bit 10	Limit Rapid Sweep	1: Dark rows 8–11 are not used for the black level algorithm controlling the calibration value. Instead, these rows are used to calculate dark averages that can be a starting point for the digital frame-wise black level algorithm. 0: All dark rows can be used for the black level algorithm. This means that the internal average might not correspond to the calibration value used for the frame, so the dark row average should in this case not be used as the starting point for the frame-wise black level algorithm.	0	N	N
Bit 9	Freeze Calibration	When set, does not let the averaging mode of the black level algorithm change the calibration value. Use this with the feature in the frame-wise black level algorithm that allows you to trigger the rapid sweep mode when the dark column average gets away from the black level target.	0	N	N
Bit 8	Sweep Mode	When set, the calibration value is increased by one every frame, and all channels are the same. This can be used to get a ramp input to the ADC from the calibration DACs.	0	N	N

Table 7: Register Description (continued)

BIT FIELD	DESCRIPTION		DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
Bits 7:5	Frames To Average Over	Two to the power of this value determines how many frames to average when the black level algorithm is in the averaging mode. In this mode, the running frame average is calculated from the following formula: Running frame ave = old running frame ave (old running frame ave)/2^n + (new frame ave)/ 2^n.	4	N	N
Bit 4	Step Size Forced To 1	When set, the step size is forced to 1 for the rapid sweep algorithm. Default operation (0) is to start at a higher step size when in rapid sweep mode, to converge faster to the correct value.	0	N	N
Bit 3	Switch Calibration Values	When set, the calibration values applied to the two channels are switched. This is not recommended and should not be used.	0		
Bit 2	Same Red/Blue	When this bit is set, the same calibration value is used for red and blue pixels: Calib blue = calib red.	0	N	Y
Bit 1	Same Green	When this bit is set, the same calibration value is used for all green pixels: Calib green2 = calib green1.	0	N	Y
Bit 0	Manual Override	Manual override of black level correction. 1: Override automatic black level correction with programmed values. (Reg0x61–Reg0x64). 0: Normal operation (default).	0	N	Y
R97—0X61 - GREEN1 CALIBRATION VALUE (R/W)					
Bits 8:0	Green1 Calibration Value	Analog calibration offset for green1 pixels, represented as a two's complement signed 8-bit value (if bit 8 is clear, the offset is positive and the magnitude is given by bits 7:0. If bit 8 is set, the offset is negative and the magnitude is given by not ([7:0]) + 1). If Reg0x60[0] = 0, this register is R/O and returns the current value computed by the black level calibration algorithm. If Reg0x60[0] = 1, this register is R/W and can be used to set the calibration offset manually. Green1 pixels share rows with red pixels.	0	N	Y
R98—X62 - BLUE CALIBRATION VALUE (R/W)					
Bits 8:0	Blue Calibration Value	Analog calibration offset for blue pixels, represented as a two's complement signed 8-bit value (if bit 8 is clear, the offset is positive and the magnitude is given by bits 7:0. If bit 8 is set, the offset is negative and the magnitude is given by not ([7:0]) + 1). If Reg0x60[0] = 0, this register is R/O and returns the current value computed by the black level calibration algorithm. If Reg0x60[0] = 1, this register is R/W and can be used to set the calibration offset manually.	0	N	Y
R99—0X63 - RED CALIBRATION VALUE (R/W)					
Bits 8:0	Red Calibration Value	Analog calibration offset for red pixels, represented as a two's complement signed 8-bit value (if bit 8 is clear, the offset is positive and the magnitude is given by bits 7:0. If bit 8 is set, the offset is negative and the magnitude is given by not([7:0]) + 1). If Reg0x60[0] = 0, this register is R/O and returns the current value computed by the black level calibration algorithm. If Reg0x60[0] = 1, this register is R/W and can be used to manually set the calibration offset.	0	N	Y

Table 7: Register Description (continued)

BIT FIELD		DESCRIPTION	DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
R100—0X64 - GREEN2 CALIBRATION VALUE (R/W)					
Bits 8:0	Green2 Calibration Value	Analog calibration offset for green2 pixels, represented as a two's complement signed 8-bit value (if bit 8 is clear, the offset is positive and the magnitude is given by bits 7:0. If bit 8 is set, the offset is negative and the magnitude is given by not ([7:0]) + 1.) If Reg0x60[0] = 0, this register is R/O and returns the current value computed by the black level calibration algorithm. If Reg0x60[0] = 1, this register is R/W and can be used to manually set the calibration offset. Green2 pixels share rows with blue pixels.	0	N	Y
R101—0X65 - CLOCK (R/W)					
Bit 15	PLL Bypass	1: Bypass the PLL. Use CLKIN port as master clock. 0: Use clock produced by PLL as master clock.	1	N	N
Bit 14	PLL Power-down	1: Keep PLL in power-down to save power (default). 0: PLL powered-up.	1	N	N
Bit 13	Power-down PLL During Standby	This register only has an effect when bit 14 = 0. 1: Turn off PLL (power-down) during Standby to save power (default). 0: PLL powered-up during Standby.	1	N	N
Bit 2	clk_newrow	Force clk_newrow to be on continuously.	0	N	N
Bit 1	clk_newframe	Force clk_newframe to be on continuously.	0	N	N
Bit 0	clk_ship	Force clk_ship to be on continuously.	0	N	N
R102—0X66 - PLL CONTROL 1 (R/W)					
Bits 15:8	M	M value for PLL must be 16 or higher.	28	N	N
Bits 5:0	N	N value for PLL.	9	N	N
R103—0X67 - PLL CONTROL 2 (R/W)					
Bits 11:8	Reserved	Do not change from default value.			
Bits 6:0	P	P value for PLL.	1	N	N
R192—0XC0 - GLOBAL RESET CONTROL (R/W)					
Bit 15	Global Reset Enable	Enter global reset. Alternative to using GRST_CTR pin. This bit is write -1 only and is always read 0.	0	N	N[4]
Bit 2	Global Reset Flash Control	1: Flash is de-asserted at end of readout. 0: Flash is de-asserted by Reg0xB6 (de-assert flash).	0	N	N
Bit 1	Global Reset Strobe Control	1: Strobe is de-asserted at end of readout. 0: Strobe is de-asserted by Reg0xC4 (de-assert strobe).	0	N	N
Bit 0	Global Reset Readout Control	1: Start of readout is controlled by falling edge of GRST_CTR pin. 0: Start of readout is controlled by Reg0xC2 (start readout time).	0	N	N
R193—0XC1 - START INTEGRATION TIME (T1) (R/W)					
Bits 15:0	Start Integration Time (T1)	These 16 bits are compared to the upper bits of a 24-bit counter, which starts counting master clocks when global reset starts. When this value is reached, global reset is de-asserted, and integration time starts. Note: there is a minimum time period for which global reset is always held. This time is defined by the physical properties of the boost circuit.	64	N	N

Table 7: Register Description (continued)

BIT FIELD		DESCRIPTION	DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
R194—0XC2 (194) START READOUT TIME (T2) (R/W)					
Bits 15:0	Start Readout Time (T2)	These 16 bits are added to Reg0xC1 (start integration time) and compared to the 24-bit counter mentioned for Reg0xC1. The value defines the time from when integration time starts to when it is guaranteed to end. Readout then commences.	64	N	N
R195—0XC3 - ASSERT STROBE TIME (T3) (R/W)					
Bits 15:0	Assert Strobe Time (T3)	These 16 bits are compared to the upper bits of a 24-bit counter, which starts counting master clocks when global reset starts. When this value is reached, the strobe is asserted.	96	N	N
R196—0XC4 - DE-ASSERT STROBE TIME (T4) (R/W)					
Bits 15:0	De-assert Strobe Time (T4)	These 16 bits are compared to the upper bits of a 24-bit counter, which starts counting master clocks when global reset starts. When this value is reached, the strobe is de-asserted if strobe control is 0 (Reg0xC0[1]).	C8	N	N
R197—0XC5 - ASSERT FLASH TIME (R/W)					
Bits 15:0	Assert Flash Time	These 16 bits are compared to the upper bits of a 24-bit counter, which starts counting master clocks when global reset starts. When this value is reached, the flash is asserted.	64	N	N
R198—0XC6 - DE-ASSERT FLASH TIME (R/W)					
Bits 15:0	De-assert Flash Time	These 16 bits are compared to the upper bits of a 24-bit counter, which starts counting master clocks when global reset starts. When this value is reached, the flash is de-asserted if flash control is 0 (Reg0xC0[2]).	78	N	N
R224—0XE0 - EXTERNAL SAMPLING 3 (R/O)					
Bits 9:0	External Sampling 3	Contains sample of AIN3 if external sampling is enabled (Reg0xE3[15] = 1). See "Analog Inputs AIN1–AIN3" on page 54.			
R225—0XE1 - EXTERNAL SAMPLING 2 (R/O)					
Bits 9:0	External Sampling 2	Contains sample of AIN2 if external sampling is enabled (Reg0xE3[15] = 1).			
R226—0XE2 - EXTERNAL SAMPLING 1 (R/O)					
Bits 9:0	External Sampling 1	Contains sample of AIN1 if external sampling is enabled (Reg0xE3[15] = 1).			
R227—0XE3 - EXTERNAL SAMPLING CONTROL (R/W)					
Bit 15	Enable Sampling	1: Enable external sampling. 0: Disable external sampling.	0	N	N
Bit 14	Show Sample	If external sampling is enabled (Reg0xE3[15] = 1): 1: Show the external samples in the data stream after LINE_VALID goes low. 0: Don't show external samples in data stream.	0	N	N
R240—0XF0 - PAGE REGISTER (R/W)					
Bits 2:0	Page Register	Must be 0.	0	N	N

Table 7: Register Description (continued)

BIT FIELD		DESCRIPTION	DEFAULT (HEX)	SYNC'D TO FRAME START	BAD FRAME
R241—0XF1 - BYTEWISE ADDRESS (R/W)					
Bits 15:0	Bytewise Address	Special address to perform 16-bit reads and writes to the sensor in 8-bit chunks. See "8-Bit Write Sequence" on page 15.	0	N	N
R242—0XF2 - CONTEXT CONTROL (R/W)					
Bit 15	Restart	Setting this bit causes the sensor to abandon the current frame and start resetting the first row. Same physical register as Reg0x0D[1].	0	N	YM
Bit 7	Xenon Flash Enable	Enable Xenon flash. Same physical register as Reg0x23[13].	0	Y	N
Bit 3	Read Mode Select	1: Use Read mode Context B, Reg0x20. 0: Use Read mode Context A, Reg0x21. Note that bits only found in Read mode Context B register are always taken from that register.	1	Y	YM
Bit 2	LED Flash Enable	Enable LED flash. Same physical register as Reg0x23[8].	0	Y	Y
Bit 1	Vertical Blank Select	1: Use Vertical Blank Context B, Reg0x06. 0: Use Vertical Blank Context A, Reg0x08.	1	Y	YM
Bit 0	Horizontal Blank Select	1: Use Horizontal Blank Context B, Reg0x05. 0: Use Horizontal Blank Context A, Reg0x07.	1	Y	YM
R255—0XFF - CHIP VERSION (R/O)					
Bits 15:0	Chip Version	Chip version.	1511		

NOTE:

[1]See "Flash STROBE" on page 48.

[2]Unless integration time is less than one frame.

[3]If enabled in bit 3.

[4]Will cause current frame to stop if triggered during a frame.

Notation used in the register description table:

Sync'd to frame start

N = No. The register value is updated and used immediately.

Y = Yes. The register value is updated at next frame start as long as the synchronize changes bit is 0. Note also that frame start is defined as when the first dark row is read out. By default, this is 8 rows before FRAME_VALID goes HIGH.

Bad frame

A bad frame is a frame where all rows do not have the same integration time, or offsets to the pixel values changed during the frame.

N = No. Changing the register value does not produce a bad frame.

Y = Yes. Changing the register value might produce a bad frame.

YM = Yes, but the bad frame is masked out unless the "show bad frames" feature is enabled.

Feature Description

PLL Generated Master Clock

The MT9D011 has an on-chip PLL that can generate a master clock in the range of 36 MHz to 40 MHz from an input reference clock of 4 MHz to 40 MHz. It is possible to bypass the PLL and use CLKIN as master clock. This is controlled by Reg0x65[15]. When the PLL is bypassed (Reg0x65[15] = 1), it is recommended to set the PLL in power down mode by setting Reg0x65[14] = 1. Default mode is PLL bypassed and in powerdown mode.

Reg0x66 and Reg0x67 controls the frequency setting of the generated clock.

PLL Settings

The PLL is controlled through its M, N and P parameters, as set in registers 0x66 and 0x67. The PLL output frequency (fout) has the following relationship to the input frequency (fin):

fout = fin*M/(2*(N+1)*(P+1))

Not all possible settings are allowed. M must be 16 or higher. Also, the following restrictions on frequencies must be obeyed:

FREQUENCY	EQUATION	MIN [MHZ]	MAX [MHZ]
fPFD	fin/(N+1)	2	16
fVCO	fPFD*M	110	220
fout	fVCO/(2*(P+1)	36	40

PLL Power-up

The PLL takes time to power up. During this time, the behavior of its output clock is not guaranteed. The PLL is in power-down by default and must be turned on manually. When using the PLL, the correct power-up sequence after chip reset is as follows:

1. Program PLL frequency settings (Reg0x66 and Reg0x67)
2. Power up PLL (Reg0x65[14] = 0)
3. Wait for PLL settling time > 150µs
4. Turn off PLL bypass (Reg0x65[15] = 0)

Window Control

Window Start

The row and column start address of the displayed image can be set by Reg0x01 (Row Start) and Reg0x02 (Column Start).

Window Size

The size of the displayed image can be set by Row Width Reg0x03 and Column Width Reg0x04. The default image size is 1600 columns and 1200 rows (UXGA).

The window start and size registers can be used to configure an image size between 17 and 1632 columns and between 2 and 1216 rows.

Pixel Border

When Reg0x20[9:8] are both set, a four pixel border is added around the specified image. This border can be used as extra pixels for image processing algorithms. The border is independent of the readout mode, which means that even in skip, zoom, and binning modes, a four pixel border is output in the image. When enabled, the row and column widths are eight pixels larger than the values programmed in Reg0x03 and Reg0x04. If the border is enabled but not shown in the image (Rex0x20[9:8] = 01), the horizontal blanking and vertical blanking values are eight pixels larger than the values programmed in the blanking registers.

Readout Modes

Readout Speeds and Power Savings

The MT9D011 has two ADCs to convert the pixel values to digital data. Because the ADCs run at half the master clock frequency, it is possible to achieve a data rate equal to the master clock frequency. By turning off one of the ADCs, the power consumption of the sensor is reduced. The pixel clock is then reduced by a factor of two.

In Reg0x20 or Reg0x21, bit 10 chooses between the two modes:

0: Use both ADCs and read out at the set pixel clock frequency (Reg0x0A, bits 3:0).

1: Use 1 ADC and read out at half the set pixel clock frequency (Reg0x0A, bits 3:0).

This can be used, for instance, when the camera is in preview mode. To make the transitions between two sensor settings easier, some simple context switching is described in "Context Switching" on page 44.

Column Mirror Image

By setting Reg0x20[1] = 1 (Reg0x21 in Context A), the readout order of the columns are reversed as shown in Figure 13. The starting color is preserved when mirroring the columns.

Row Mirror Image

By setting Reg0x20[0] = 1 (Reg0x21 in Context A), the readout order of the rows are reversed as shown in Figure 14. The starting color is preserved when mirroring the rows.

Figure 13: Six Pixels in Normal and Column Mirror Readout Modes

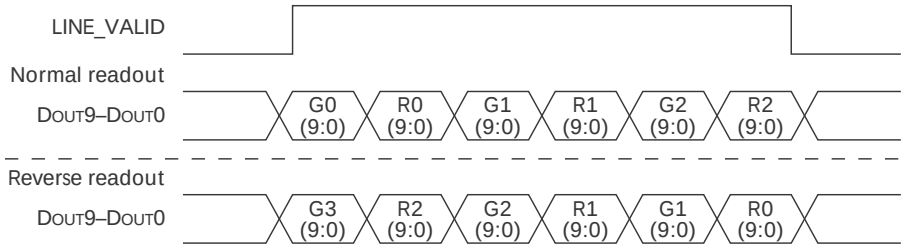
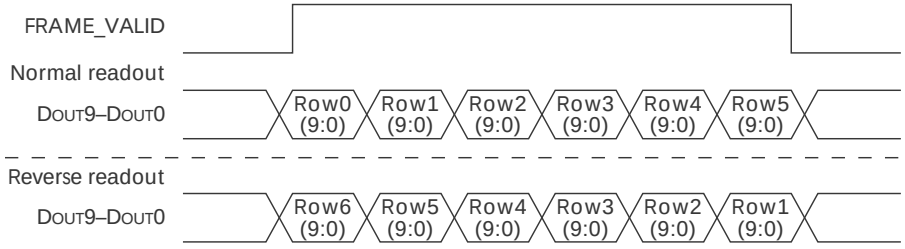


Figure 14: Six Rows in Normal and Row Mirror Readout Modes



Column and Row Skip

This section assumes Context B. If Context A is used, replace all references to Reg0x20 with Reg0x21.

By setting Reg0x20[4] = 1 or Reg0x20[7] = 1, skip is enabled for rows or columns, respectively. When skip is enabled, the image is subsampled. The amount of skipping is set by Reg0x20[3:2] (rows) and Reg0x20[6:5] (columns) according to Table 8.

The number of rows or columns read out is what is set in Reg0x03 or Reg0x04, respectively, divided by the Skip Value in this table.

In all cases, the row and column sequencing ensures that the Bayer pattern is preserved.

Column skip examples are shown in Figures 15 through 18.

Table 8: Skip Values

BIT VALUES	SKIP VALUE
00	2
01	4
10	8
11	16

Figure 15: Eight Pixels in Normal and Column Skip 2x Readout Modes

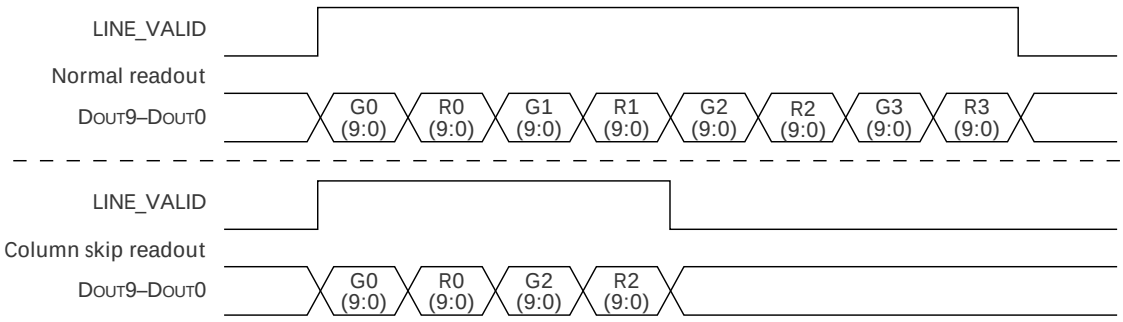


Figure 16: Sixteen Pixels in Normal and Column Skip 4x Readout Modes

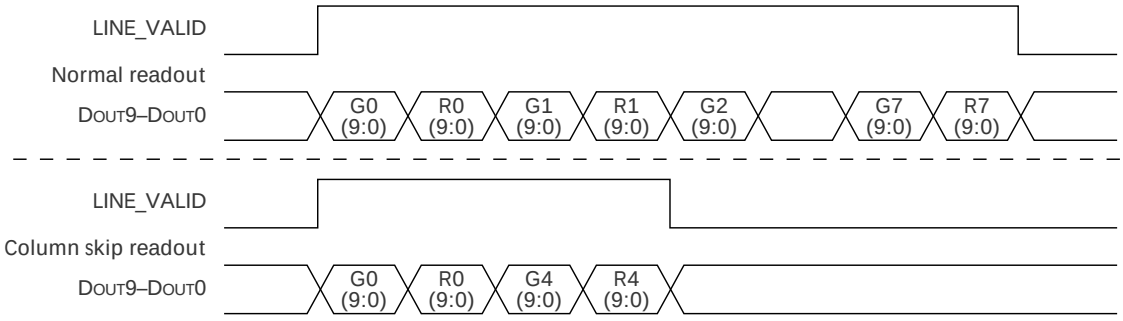


Figure 17: Thirty-two Pixels in Normal and Column Skip 8x Readout Modes

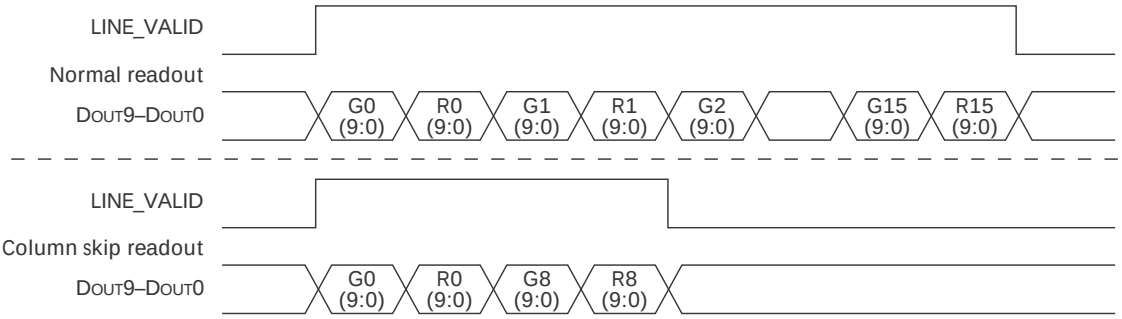
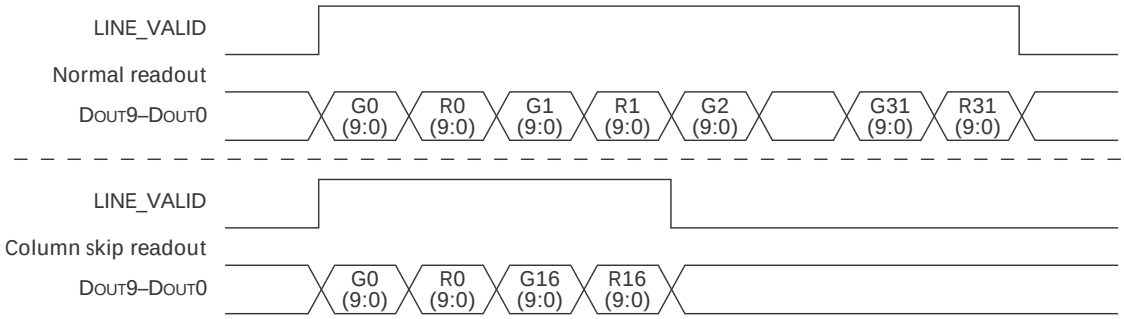


Figure 18: Sixty-four Pixels in Normal and Column Skip 16x Readout Modes



Digital Zoom

Reg0x20[13] enables a digital zoom of 2x to 16x to be applied. The zoom value is set in Reg0x20[12:11] according to Table 9.

In zoom mode, the pixel data rate is slowed by the zoom factor, and a number of additional blank rows are added between output rows (see Table 9). This is designed to give the controller logic the necessary time to repeat data, filling in a larger window with repeated data.

Table 9: Zoom Values

BIT VALUES	ZOOM VALUE	BLANK ROWS
00	2	1
01	4	3
10	8	7
11	16	15

The pixel clock speed is not affected by this operation, therefore the output data for each pixel is valid for zoom factor number of pixel clocks. Every row is followed by a number of blank rows (with their own LINE_VALID, but all data bits = 0) of equal time.

In zoom modes, Reg0x03 and Reg0x04 still specifies the window size out of the sensor including the extra blanking, so the active image read out is, in effect, smaller than the output image.

Figures 19 through 23 show the data coming from the sensor in the different zoom modes. The colors represent from which colored pixel the data comes. Black represents data = 0.

Figure 19: Data from Pixel Array in Normal Mode

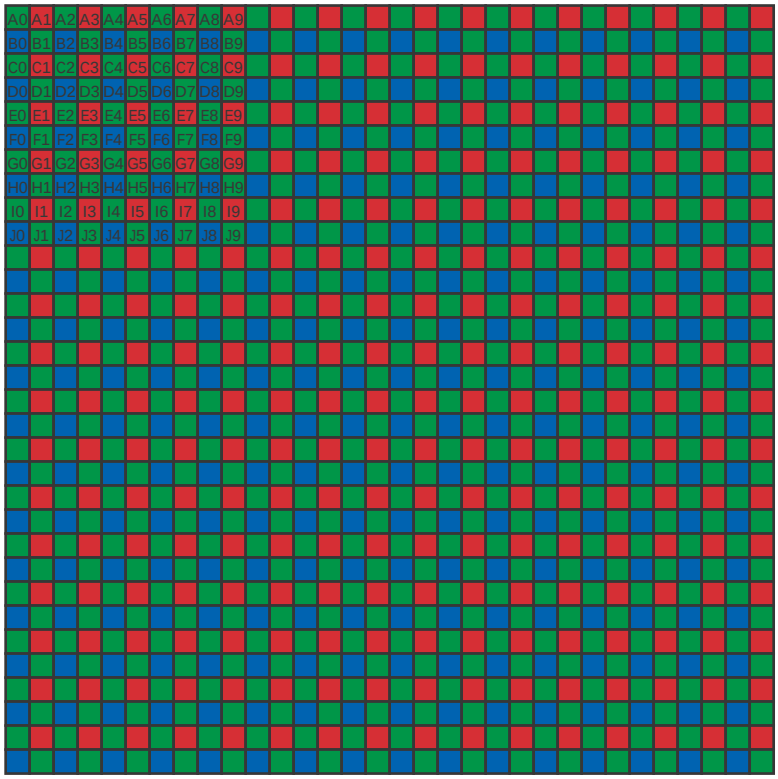


Figure 20: Data from Pixel Array in Zoom 2x Mode

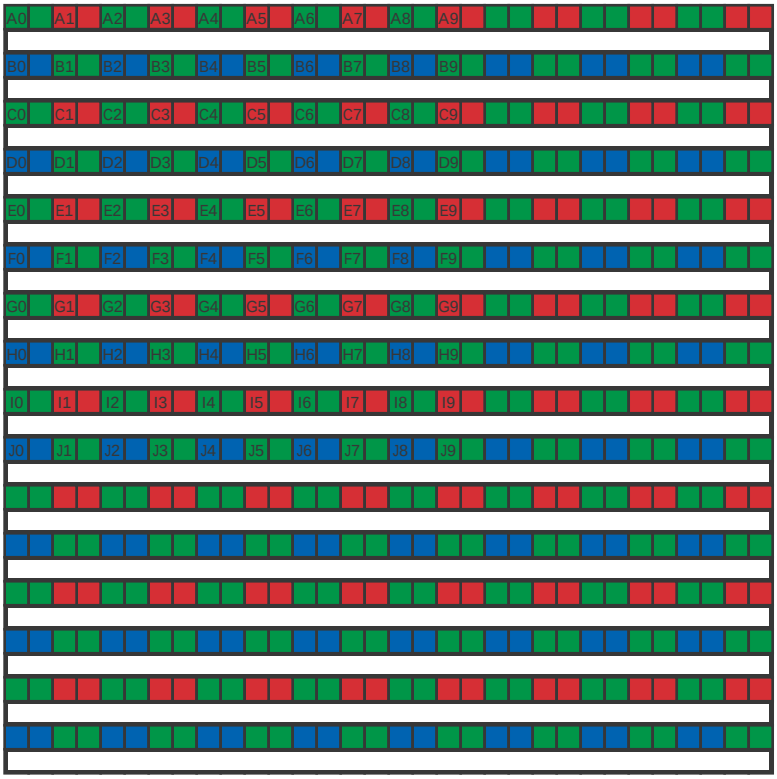


Figure 21: Data from Pixel Array in Zoom 4x Mode

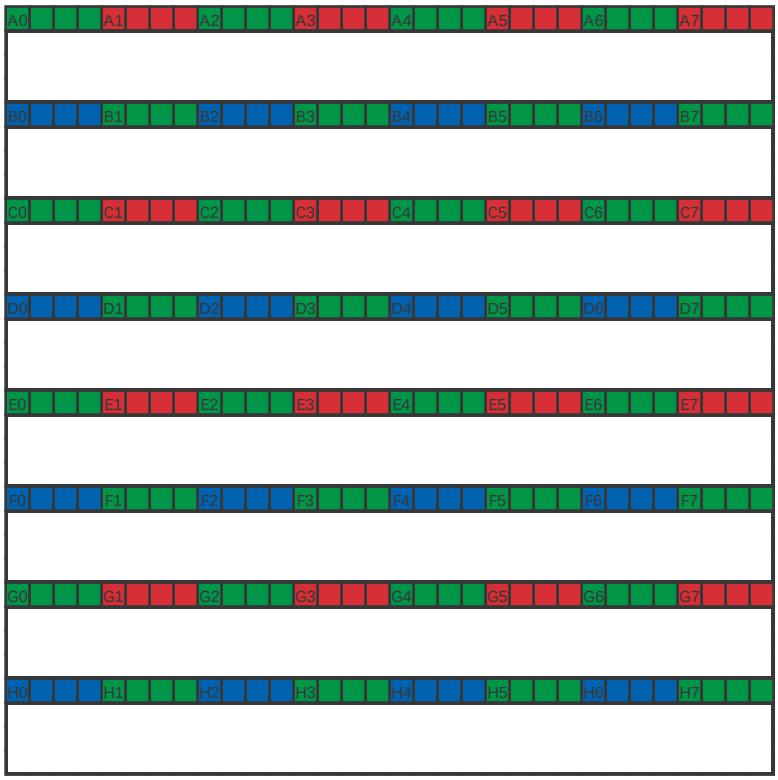


Figure 22: Data from Pixel Array in Zoom 8x Mode

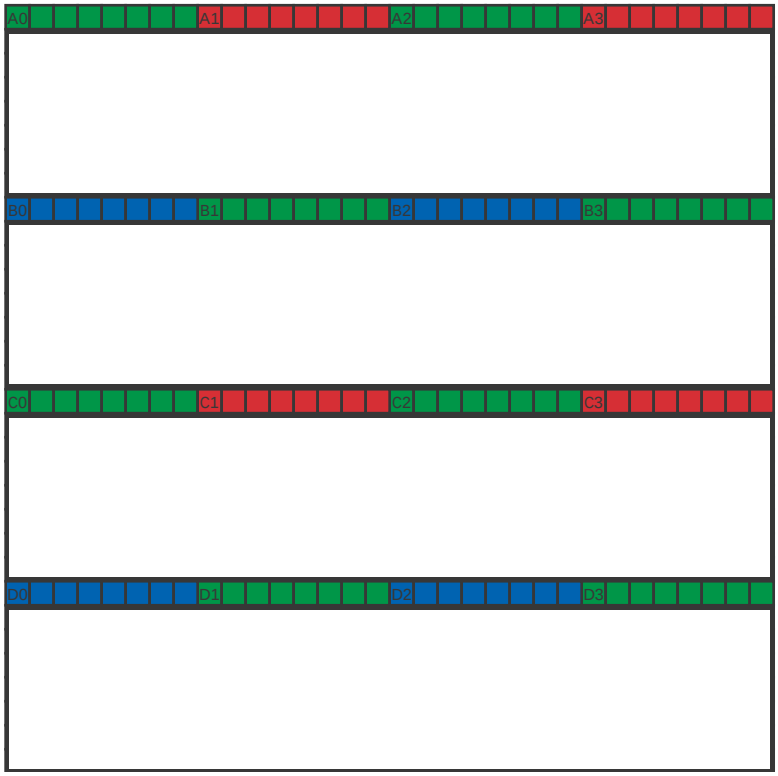
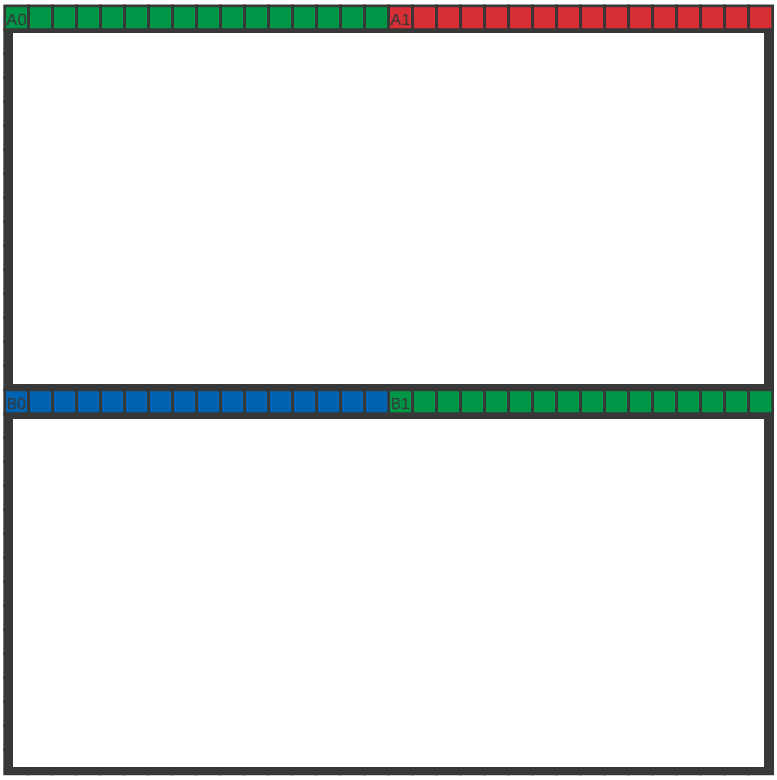


Figure 23: Data from Pixel Array in Zoom 16x Mode



Binning

The MT9D011 supports 2 x 2 binning of four pixels of the same color. This mode can be activated by asserting Reg0x20[15] (Reg0x21 if Context A is used).

Binning is primarily used instead of 2x skip as a way of decimating the picture without losing information. The effect of aliasing in preview mode is eliminated when binning is used instead of just skipping rows and columns.

Activating binning has several implications.

- It adds a level of skip, so the picture that comes out has the same dimensions as a picture read out with the next higher skip setting.
- It increases the minimum hblank and minimum row time requirements (see Table 10 and Table 11).

Binning Limitations

To achieve correct operation, the following conditions must be met:

- Start address must be divisible by four (row and column).
- Window size must be divisible by four in both directions, after dividing by zoom factor and skip factor (because they both reduce the effective window size from the sensor's point of view).

Example: Default row size = 1200. 8x zoom means the actual window on the sensor is divided by 8, so 8x zoom and binning is not allowed with default window size, because 1200 / 8 = 150, which is not divisible by 4.

- Binning can be seen as an extra level of skip. The combination binning/16x skip is therefore not legal.

Frame Rate Control

For a given window size, the blanking registers (Reg0x05-Reg0x08) along with the row speed register (Reg0x0A) can be used to set a particular frame rate.

The frame timing equations (Table 3 and Table 4 on page 12) can be rearranged to express the horizontal blanking or vertical blanking values as a function of the frame rate:

HBLANK_REG = master clock freq / (frame rate*
((Reg0x03/S + BORDER) + VBLANK_REG)*PIXCLK_PERIOD) - (Reg0x04/S + BORDER)

VBLANK_REG = master clock freq / (frame rate*
((Reg0x04/S + BORDER) + HBLANK_REG)*PIXCLK_PERIOD) - (Reg0x03/S + BORDER)

The HBLANK_REG value allows the frame rate to be adjusted with a minimum resolution of one PIXCLK_PERIOD multiplied by the total number of rows (displayed plus blanking). When finer resolution is required, Reg0x0b (Extra Delay) can be used. Reg0x0b allows the frame time to be changed in increments of pixel clocks.

Minimum Horizontal Blanking

The minimum horizontal blanking value is constrained by the time used for sampling a row of pixels and the overhead in the row readout. This is expressed in Table 10.

Table 10: Minimum Horizontal Blanking Parameters

PARAMETER	DEFAULT / 2 ADC MODE, NO BINNING	1 ADC MODE, NO BINNING	2 ADC MODE, BINNING	1 ADC MODE, BINNING
HBLANK(MIN)	286 mclks	324 mclks = 162 pixclks	470 mclks	508 mclks = 254 pixclks

Minimum Row Time Requirement

The total row time must be sufficient to allow all row operations (readout and shutter operations). The row time is the sum of column width (halved during binning divided by column skip factor) and horizontal blanking, and can therefore be adjusted by programming these.

Table 11 shows minimum row time as a function of mode of operation.

Note that this is a particularly strict requirement during binning because twice as many row operations are required per row and the column width is halved.

Table 11: Minimum Row Time Parameters

PARAMETER	DEFAULT / 2 ADC MODE, NO BINNING	1 ADC MODE, NO BINNING	2 ADC MODE, BINNING	1 ADC MODE, BINNING
ROW_TIME(MIN)	473 mclks	488 mclks = 244 pixclks	931 mclks	946 mclks = 473 pixclks
pointer_operations	461 mclks	464 mclks	919 mclks	922 mclks

Context Switching

Reg0xF2 is designed to enable easy switching between sensor modes. Some key registers and bits in the sensor have two physical register locations, called contexts. Bits 0, 1, and 3 of Reg0xF2 control which context register context is currently in use. A "1" in a bit selects Context B, while a "0" selects Context A for this parameter. The select bits can be used in any combination, but by default are setup to allow easy switching between preview mode and full resolution mode:

CONTEXT B (DEFAULT CONTEXT)

Reg0xF2	= 0x000B	(Context B)
Reg0x05	= 0x015C	(Horizontal Blanking, Context B)
Reg0x06	= 0x0020	(Vertical Blanking, Context B)
Reg0x20	= 0x0000	(2 ADCs, no column or row skip)

DESCRIPTION: Full-resolution UXGA (1600 x 1200) image at 15 fps

CONTEXT A (ALTERNATE CONTEXT, PREVIEW MODE)

Reg0xF2	= 0x0000	(Context A)
Reg0x07	= 0x00AE	(Horizontal blanking, Context A)
Reg0x08	= 0x0010	(Vertical blanking, Context A)
Reg0x21	= 0x0490	(1 ADC, 2x column and row skip)

DESCRIPTION: Half-resolution SVGA (800 x 600) image at 30 fps

The horizontal blanking and vertical blanking values for the two contexts are chosen so that row time is preserved between contexts. This ensures that changing contexts does not affect integration time. A few more control bits are also available through the context register (Reg0xF2) so that flash and restarting the sensor can be done simultaneously with changing contexts. See Table 7 on page 22 for more information.

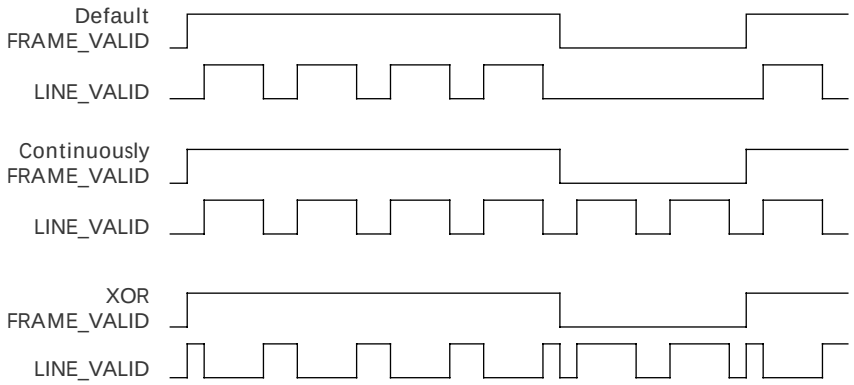
Settings for skip, 1 ADC mode, and binning can be set separately for Context B and Context A using Reg0x20 and Reg0x21, respectively. When these settings are referred to in this document, the register is dependent on what context is set in Reg0xF2.

Valid Data Signals Options

LINE_VALID Signal

By setting bits 14–15 of Reg0x25, the LINE_VALID signal is programmed for three different output formats. The formats shown in Figure 24 illustrate reading out four rows and two vertical blanking rows. In the last format, the LINE_VALID signal is the XOR between the continuous LINE_VALID signal and the FRAME_VALID signal.

Figure 24: LINE_VALID Formats



FRAME_VALID Signal

Reg0x1F allows the user to move FRAME_VALID with respect to data (and LINE_VALID). The rising and falling edges of FRAME_VALID are separately programmable, and can be moved earlier by a number of row_times (plus some overhead).

By programming a value of N in bits [6:0] and enabling it in bit 7, FRAME_VALID will rise before the horizontal blanking N+1 rows earlier. This is shown in Figure 25. N should not be set to higher than 9. Similarly, by programming a value of M in bits [14:8] and enabling it in bit 15, FRAME_VALID will fall M+1 rows earlier. This is shown in Figure 26. M must not be set so FRAME_VALID would fall before LINE_VALID starts toggling. This is avoided by keeping M smaller than Row Width (Reg0x03/S + BORDER).

Figure 25: Early FRAME_VALID Rise

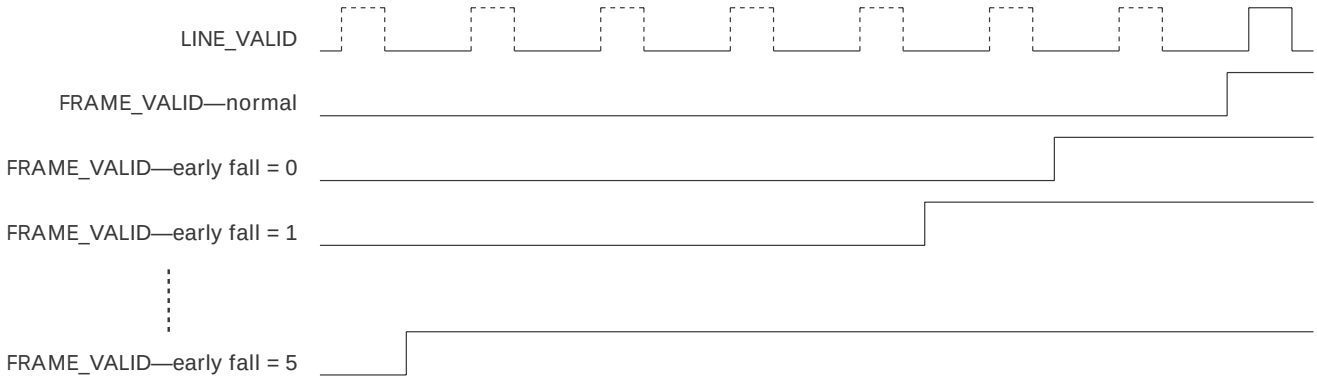
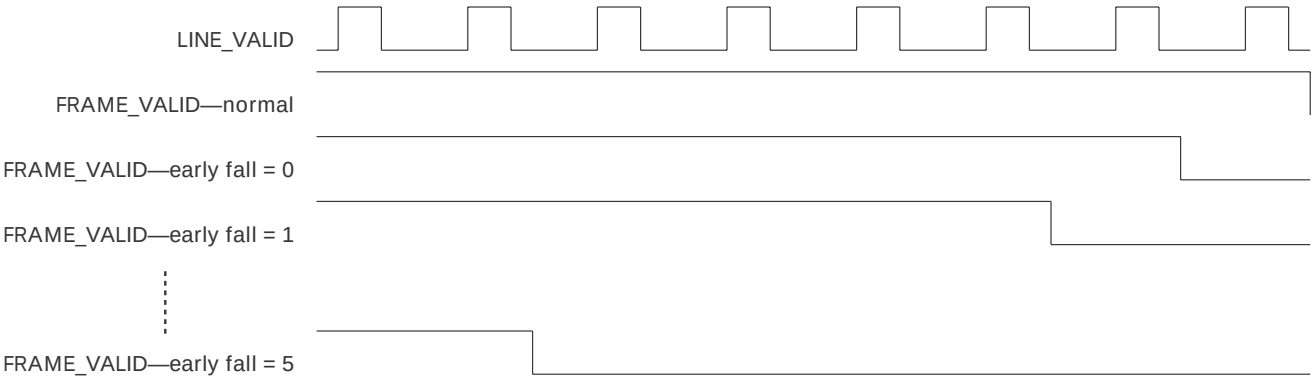


Figure 26: Early FRAME_VALID Fall



Integration Time

Integration time is controlled by Reg0x09 (shutter width in multiples of the row time) and Reg0x0C (shutter delay, in PIXCLK_PERIOD/2). Reg0x0C is used to control sub-row integration times and only has a visible effect for small values of Reg0x09. The total integration time, ^{t}INT, is shown in the equation below:

^{t}INT = Reg0x09*Row Time - Integration Overhead - Shutter Delay

where:

Row Time = (Reg0x04/S + BORDER + HBLANK_REG)*PIXCLK_PERIOD master clock periods (from Table 3 on page 12)
S = Skip Factor, multiplied by 2 if binning is enabled
Overhead Time = 260 master clock periods (262 in 1 ADC mode)
Shutter Delay = Reg0x0C*PIXCLK_PERIOD master clock periods (/2 in 1 ADC mode)

with default settings:

^{t}INT = (1232*(1600 + 348)) - 260 - 0
= 2,399,676 master clock periods = 66.66ms@36 MHz

In the equation, the Integration Overhead corresponds to the delay between the row reset sequence and the row sample (read) sequence.

Typically, the value of Reg0x09 is limited to the number of rows per frame (which includes vertical blanking rows), so that the frame rate is not affected by the integration time. If Reg0x09 is increased beyond the total number of rows per frame, the sensor adds blanking rows as needed. Additionally, ^{t}INT must be adjusted to avoid banding in the image caused by light flicker. Therefore, ^{t}INT must be a multiple of 1/120 of a second under 60Hz flicker, and a multiple of 1/100 of a second under 50Hz flicker.

Maximum Shutter Delay

The shutter delay can be used to reduce the integration time. A programmed value of *N* reduces the integration time by N master clock periods. The maximum shutter delay is set by the row time and the sample time, as shown in the equation below:

Maximum shutter delay = (Row Time - pointer_operations)

where:

Row Time = (Reg0x04/S + BORDER + HBLANK_REG)*PIXCLK_PERIOD master clock periods (from Table 3 on page 12)
S = Skip Factor, multiplied by 2 if binning is enabled
pointer_operations = see Table 11 on page 43.

with default settings:

Maximum shutter delay = (1600 + 348) - 461
= 1487 (master clock periods)

If the value in this register exceeds the maximum value given by this equation, the sensor may not generate an image.

Flash STROBE

The MT9D011 supports both Xenon and LED flash through the FLASH output pin. The timing of the FLASH pin with the default settings is shown in Figure Figure 27, Figure 28, and Figure 29. Reg0x23 allows the timing of the flash to be changed. The flash can be programmed to: fire only once; be delayed by a few frames when asserted; and (for Xenon flash) the flash duration can be programmed.

When Xenon flash is enabled, an integration time significantly smaller than one frame will cause uneven exposure of the image, as will setting a flash pulse width larger than Vertical Blanking.

Enabling the LED flash causes one bad frame in which several rows have the flash on during only part of their integration time. This can be avoided by forcing a restart (write Reg0x0D[1] = 1) immediately after enabling the flash; the first bad frame is then masked out as shown in Figure 29. Read-only bit Reg0x23[14] is set during frames that are correctly integrated; the state of this bit is shown below.

Figure 27: Xenon Flash Enabled

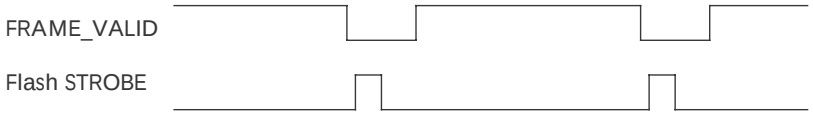
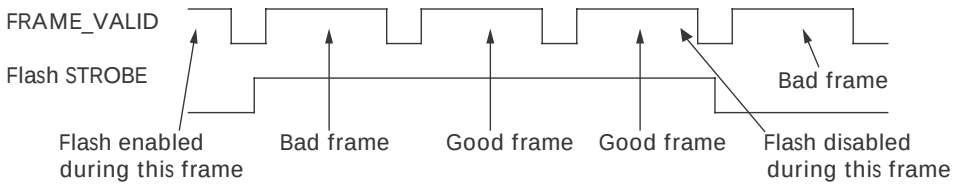
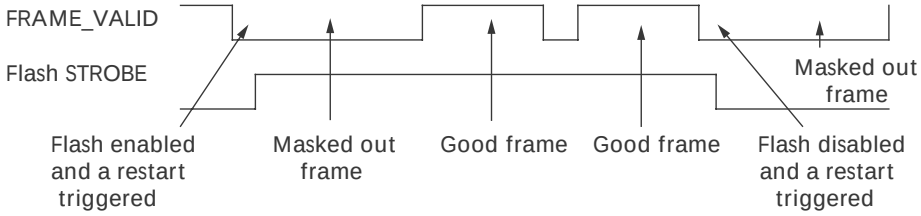


Figure 28: LED Flash Enabled



NOTE:
Integration time = number of rows in a frame.

Figure 29: LED Flash Enabled, Using Restart



NOTE:
Integration time = number of rows in a frame.

Global Reset

The MT9D011 provides a global reset mode in which the pixel integration time is controlled by an external mechanical shutter. The sensor can then operate on a lower clock frequency, reducing the bandwidth on the interface between the sensor and the host processor without losing image quality.

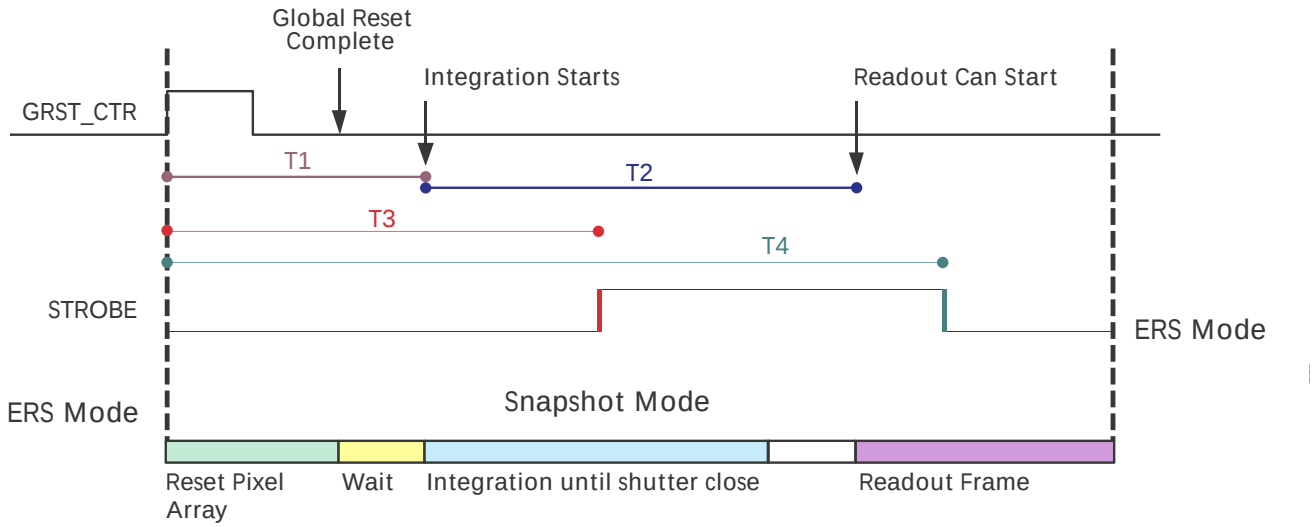
The basic operation is as follows: The sensor operates in either preview or full-frame mode (Electronic Rolling Shutter or ERS). A rising edge on the signal GRST_CTR or a write to an internal register starts the global reset sequence. The sensor now enters the snapshot mode and after a certain time, all the lines in the sensor array is reset and kept in a reset state until the integration starts. The start of the integration (exposure) period, the assertion of the STROBE signal, the start of the readout and the de-assertion of the STROBE signal can be controlled by internal registers (T1, T2, T3 and T4 shown in Figure 30).

The MT9D011 provides an output signal, STROBE, that can be used to control the mechanical shutter. This signal can be programmed to occur in a specified window around the actual start of integration.

During Global Reset, the FLASH pin is programmed in a different way than during normal ERS operation. Normally, the FLASH behavior is programmed using Reg0x23. In Global Reset mode, the FLASH strobe is programmed in the same way as the STROBE pin showed in Figure 30, using registers Reg0xC5 and Reg0xC6.

Reg0xC0[0] controls the mechanism for starting the readout after a global reset operation. If this bit is high, the integration time is directly controlled by the GRST_CTR pin. Very long integration times can be achieved this way.

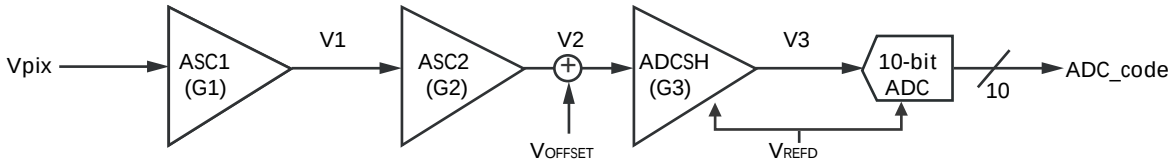
Figure 30: Global Reset Operation



Analog Signal Path

The MT9D011 features two identical analog readout channels. A block diagram for one channel is shown in Figure 31. The readout channel consists of two gain stages (ASC1 and ASC2), a sample-and-hold (ADCSH) stage with black level calibration capability (VOFFSET), and a 10-bit ADC.

Figure 31: Analog Readout Channel



Stage-by-Stage Transfer Functions

Transfer functions proceed stage-by-stage, as follows:

Let VPIX be the input of the signal path:	VPIX = pixel output voltage = signal path input voltage,	
The output voltage of ASC 1st stage is:	V1 = -1*G1*VPIX	(1)
The output voltage of ASC 2nd stage is:	V2 = -1*G2*V1	(2)
The output voltage of ADC Sample-and-Hold stage is:	V3 = 2*G3*V2 - VREFD + VOFFSET	(3)
and the ADC output code is:	ADC output code = 511*(1 + (V3 / VREFD))	(4)
From (1) to (4), the ADC output code can also be written as:	ADC code = (1022/VREFD)*[G1*G2*G3*VPIX + (Voffset/(2*G3))]	(5)

Where G1, G2, and G3 are the gain settings, VOFFSET is the offset (calibration) voltage, and VREFD is the reference voltage of the ADC. The gain setting G3 is applied to the signal but is not applied to VOFFSET. The parameters VREFD, G1, G2, G3, and VOFFSET are described next.

VREFD

The VREFD parameters are as follows:

The ADC reference voltage VREFD is:	VREFD = VREF_HI - VREF_LO	(6)
where	VREF_HI = 55.5mV*(Reg0x41[7:4] + 23)	(7)
using default register values:	VREF_HI = 55.5mV*(13 + 23) = 1.998V	
and	VREF_LO = 55.5mV*(Reg0x41[3:0] +11)	(8)
using default register values:	VREF_LO = 55.5mV*(7 +11) = 0.999V	
so	VREFD = 55.5mV*(Reg0x41[7:4] - Reg0x41[3:0] + 12)	(9)
using default register values	VREFD = 1.998 - 0.999 = 0.999V	

Gain Settings: G1, G2, G3

The gains for green1, blue, red, and green2 pixels are set by registers Reg0x2B, Reg0x2C, Reg0x2D, and Reg0x2D, respectively. Gain can also be globally set by Reg0x2F. The analog gain is set by bits 8:0 of the corresponding register as follows:

G1 = bit 7 + 1 (10)

G2 = bit 6:0 / 32 (11)

G3 = bit 8 + 1 (12)

Digital gain is set by bits 11:9 of the same registers.

Offset Voltage: VOFFSET

The offset voltage provides a constant offset to the ADC to fully utilize the ADC input dynamic range. The offset voltages for green1, blue, red, and green2 pixels are manually set by registers Reg0x61, Reg0x62, Reg0x63, and Reg0x64, respectively. Note that the offset voltages also can be automatically set by the black-level calibration loop.

For a given color, the offset voltage, VOFFSET, is determined by:

VOFFSET = 0.50V*offset_gain*offset_sign*offset_code[7:0]/255 (13)

where: "offset_sign" is determined by bit 8 as:

if bit 8 = 0, offset_sign = +1 (14)

if bit 8 = 1, offset_sign = -1 (15)

"offset_code" is the decimal value of bit<7:0>

"Offset_gain" is determined by the 2-bit code from Reg0x5A[1:0], as shown in Table 12. These step sizes are not exact; increasing the stage0 ADC gain from 2 to 4 decreases the step size significance; decreasing the ADC VREFD increases the step size significance.

Table 12: Offset Gain

REG0X5A[1:0]	OFFSET_GAIN
00	OFFSET_GAIN = 0 (no calibration voltage is applied)
01	OFFSET_GAIN = 0.25 (1 calibration LSB is equal to 0.5 ADC LSB when VREFD = 1V)
10	OFFSET_GAIN = 0.50 (1 calibration LSB is equal to 1 ADC LSB when VREFD = 1V)
11	OFFSET_GAIN = 1 (1 calibration LSB is equal to 2 ADC LSB when VREFD = 1V)

Recommended Gain Settings

The analog gain circuitry in the MT9D011 provides signal gains from 1 to 15.875.

Table 13: Recommended Gain Settings

DESIRED GAIN	RECOMMENDED GAIN REGISTER SETTING
1–1.969	0x020–0x03F
2–7.938	0x0A0–0x0FF
8–15.875	0x1C0–0x1FF

Output Enable Control

When the sensor is configured to operate in Default mode, the DOUT, FRAME_VALID, LINE_VALID, PIXCLK, and flash outputs can be placed in a high-impedance state under hardware or software control, as shown in Table 14.

Table 14: Output-Enable Control

STANDBY	REG0X0D[4] (OUTPUT_DIS)	REG0X0D[6] (DRIVE_PINS)	PIN STATE
0	0 (default)	0 (default)	Driven
1	0 (default)	0 (default)	High-Z
don't care	0 (default)	1	Driven
don't care	1	don't care	High-Z

The pin transition between driven and High-Z always occurs asynchronously. Output-enable control is provided as a mechanism to allow multiple sensors to share a single set of interface pins with a host controller.

NOTE: There is no benefit in placing the pins in a High-Z while the sensor is in its low power standby state. Therefore, in single-sensor applications that use the STANDBY pin to

enter and leave the standby state, programming Reg0x0D[6] = 1 is recommended.

Power-Saving Modes

The sensor can be placed in a low power standby state by either of these mechanisms:

- Asserting STANDBY input pin (provided that Reg0x0D[7] = 0)
- Setting Reg0x0D[2] = 1 by performing a register write through the serial register interface.

These two methods are equivalent and have the same effect:

- The source of standby is synchronized and latched. Once latched, the full standby sequence is completed even if the source of standby is removed.
- The readout of the current row is completed.
- Internal clocks are gated off.
- The analog signal chain and associated current and voltage sources are placed in a low power state.

The standby state is maintained for as long as the standby source remains asserted. Table 15 shows the state of the pin interface while in standby state.

Table 15: Signal State During Standby

SIGNAL	STATE
LINE_VALID	0
FRAME_VALID	0
LINE_VALID	0
PIXCLK	0
FLASH	0
DOUT9–DOUT0	0

Output-enable control can be used to place the pin interface in a high-impedance state (see "Output Enable Control" on page 51).

While in standby, the state of the internal registers is maintained and the sensor continues responding to accesses through its serial register interface. An even lower power standby state can be achieved by stopping the input clock (CLKIN) while in standby. If the input clock is stopped, the sensor does not respond to accesses through its serial register interface.

Exit from standby must be through the same mechanism as entry to standby. When the standby source is negated, this sequence occurs:

1. The internal clocks are restarted.
2. The analog circuitry is restored to its normal operating state.
3. The timing and control circuitry performs a restart equivalent to writing Reg0x0D[1] = 1.

After this sequence is complete, normal operation resumes. If the input clock is stopped during standby, it must be restarted before leaving standby.

PLL and Standby

If the PLL is used to generate master clock, special care must be taken when entering standby mode. The PLL uses relatively high power, so allowing the PLL to power down during standby is recommended. This can be controlled in Reg0x65[13]. By default the PLL powers down whenever MT9D011 enters standby. The operation of the circuit cannot be guaranteed if the PLL is driving the master clock when it powers down.

To safely allow the PLL to power down when entering standby, turn on PLL bypass before triggering standby (controlled by Reg0x65[15]). When coming out of standby mode, the normal PLL power-up sequence must be followed as specified in "PLL Power-up" on page 36.

Floating Inputs

Many MT9D011 signals use bi-directional pins (shown in Table 4 on page 12) for the following three reasons:

- The signal associated with the pin is bi-directional in normal use (the only signal in this category is SDATA).
- The pin is normally used as an output, but is used as an input during manufacturing test modes (e.g., DOUT[9:0]).
- Standard design practice dictates that signal inputs should not be allowed to float for long periods of time.

This leads to two areas where the design application should be reviewed:

- When using the output-enable control. All MT9D011 bi-directional pins that enter a high-impedance state must be driven to a valid logic level. ("Output Enable Control" on page 51.)
- When input pins are allowed to float. The MT9D011 does not include on-chip pull-down resistors, therefore, no input pins should be allowed to float.

Dark Row/Column Display

Optically black rows 7 through 0 are used to provide data for black level calibration and are not normally visible in the displayed image. Setting Reg0x22[7] = 1 makes these rows visible in the displayed image. This is achieved by asserting FRAME_VALID earlier than normal, and keeping it asserted longer, so that the following rows are displayed:

- The optically black rows at the start of the pixel array (controlled by Reg0x22[2:0]).
- Two rows before the visible rows.
- The visible rows (controlled by Reg0x01, Reg0x03 and Reg0x20).

The result of setting Reg0x22[7] = 1 is a larger image (more rows) than is programmed by Reg0x03.

Optically black columns 23 through 0 are used to provide data for row-wise noise cancellations, and are not normally visible in the displayed image. Two methods for making them visible in the displayed image are:

- Set Reg0x22[8] = 0 (to disable readout of the dark columns); set Reg0x30[10] = 0 (to disable row-wise correction); then adjust Reg0x02.
- Set Reg0x22[9] = 1. When Reg0x22[9] = 1, LINE_VALID is asserted 22 pixel clocks earlier than normal. Data from columns 21 through 2 (20 columns) is followed by two pixel clocks of undefined data, then by data from the visible columns (controlled by Reg0x02, Reg0x04, and Reg0x20).

Clock Control

The MT9D011 uses an aggressive clock-gating methodology to reduce power consumption: the clocked logic is divided into a number of separate domains, each of which is only clocked as required. Reg0x65 can be used to bypass the clock gating, so that clocks to individual domains run continuously.

When the MT9D011 enters a low power state, almost all of the internal clocks are "gated off." The only exception is that a small amount of logic (approximately 10 flip-flops) is clocked so that access to the two-wire serial interface continues to function correctly. "Power-Saving Modes" on page 53 for more information.

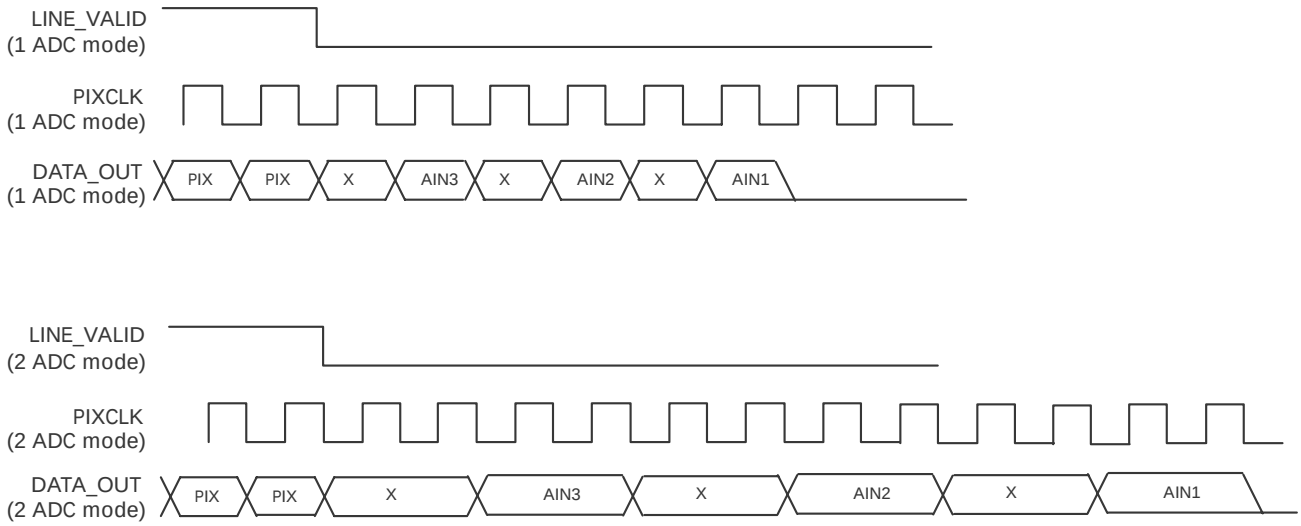
Analog Inputs AIN1–AIN3

MT9D011 can share its on-chip ADC resources, such as for use in auto focus applications. If Reg0xE3[15] is set, the chip samples AIN1–AIN3 once per row (after reading out the data from the row). The digital data from this sampling is available to the user in two ways:

- Data can be read in registers Reg0xE0 to Reg0xE2
- Data is present in the data stream after LINE_VALID goes low if Reg0xE3[14] is set

The nominal range of the AIN pins are 0V + VOFFSET to VREFD + VOFFSET. VREFD is the ADC reference voltage (nominally 1V), but can be programmed. ("Analog Signal Path" on page 50.) VOFFSET is the offset in the ADC and is typically ±10mV to 20mV. If required, the offset can be measured by converting a calibrated reference voltage, which can be used to compensate at the input. The ADC is designed to operate with differential inputs. Since AIN1-AIN3 are used as single-ended inputs to the ADC, it is recommended to average values from several samples (if possible, a whole frame) to cancel out noise.

Figure 32: Timing Diagram AIN1–AIN3 Sample



Power-up Sequence

There are no specific requirements to the order in which different supplies are turned on. The reset sequence cannot start before the last supply is stable within the valid ranges as defined in Table 16: DC Electrical Characteristics on page 56.

Hard Reset Sequence

After power-up, a hard reset is required. Assuming all supplies are stable, the assertion of the RESET# pin to logic “0” will set device in reset mode ~30ns after assertion. The input clock does not have to run while RESET# is active. Release of RESET# will require that the clock is running and after 3 clock cycles (CLKIN), the serial interface is ready to accept commands on the two-wire serial interface.

Soft Reset Sequence

At any time during normal operation or standby, the user can do a soft reset by writing a logic "1" to Reg0x0D[0] using the two-wire serial interface. This will also put the device in reset mode and all registers (including PLL state and settings) will get their default values. Writing a logic "0" to the same register will release the soft reset, and normal operation can be resumed once the write operation on the serial interface is completed.

Electrical Specifications

Table 16: DC Electrical Characteristics

(V_{DD} = 1.8V ±0.1V; V_{AA} = VAAPIX = V_{AAPLL} = V_{DDQ} = 2.8V ±0.3V; T_A = Ambient = 25°C)

SYMBOL	DEFINITION	CONDITION	MIN	TYP	MAX	UNITS
V_{IH}	Input High Voltage		TBD	TBD	TBD	V
V_{IL}	Input Low Voltage		TBD	TBD	TBD	V
I_{IN}	Input Leakage Current	No Pull-up Resistor; V_{IN} = V_{DD} or D_{GND}	-15		15	µA
V_{OH}	Output High Voltage		TBD	TBD		V
V_{OL}	Output Low Voltage		TBD	TBD	TBD	V
I_{OZ}	Tri-state Output Leakage Current				15	µA
I_{PWR}	Total Quiescent Supply Current[2]	CLKIN = 36 MHz; default settings	TBD	TBD	TBD	mA
I_{PWR} Standby	Total Standby Supply Current[1]	STANDBY = V_{DDQ}, CLKIN = 0 MHz	TBD	TBD	10	µA

NOTE:

1. To place the chip in standby mode, first raise STANDBY to V_{DDQ}, wait until FRAME_VALID and LINE_VALID are de-asserted, then wait two master clock cycles before turning off the master clock.
2. Summation of currents for all power supplies. Typical operating power does not include the I/O power or the PLL. V_{AAPLL} off, conditions are dark.

Table 17: AC Electrical Characteristics

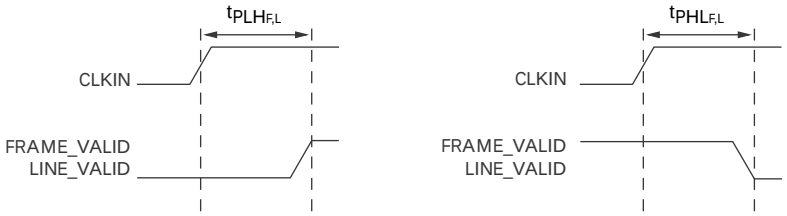
(V_{DD} = 1.8V ±0.1V; $V_{DD}Q$ = V_{AA} = VAAPIX = V_{AA}PLL = 2.8V ±0.3V; T_A = Ambient = 25°C; Load Capacitance = TBD)

SYMBOL	DEFINITION	CONDITION	MIN	TYP	MAX	UNITS
CLKIN	Input Clock Frequency		4		40	MHz
	Duty Cycle					%
t_R	Input Clock Rise Time			TBD		ns
t_F	Input Clock Fall Time			TBD		ns
t_{PLHP}	CLKIN to PIXCLK propagation delay, low-to-high	TBD				ns
t_{PHLP}	CLKIN to PIXCLK propagation delay, high-to-low	TBD				ns
t_{PLHD}	CLKIN to D_{OUT}[9:0]propagation delay, low-to-high	TBD				ns
t_{PHLD}	CLKIN to D_{OUT}[9:0] propagation delay, high-to-low	TBD				ns
t_{OH}	Data Hold Time from CLKIN					ns
$t_{PLHF,L}$	CLKIN to FRAME_VALID and LINE_VALID propagation, low-to-high	TBD		TBD		ns
$t_{PHLF,L}$	CLKIN to FRAME_VALID and LINE_VALID propagation, high-to-low			TBD		ns
t_{PHLF}	CLKIN to FLASH propagation delay, low-to-high			TBD		ns
t_{PHLF}	CLKIN to FLASH propagation delay, high-to-low			TBD		ns

Propagation Delay for FRAME_VALID and LINE_VALID

The LINE_VALID and FRAME_VALID signals change on the rising edge of the master input clock, as shown in Figure 33.

Figure 33: Propagation Delay for FRAME_VALID and LINE_VALID

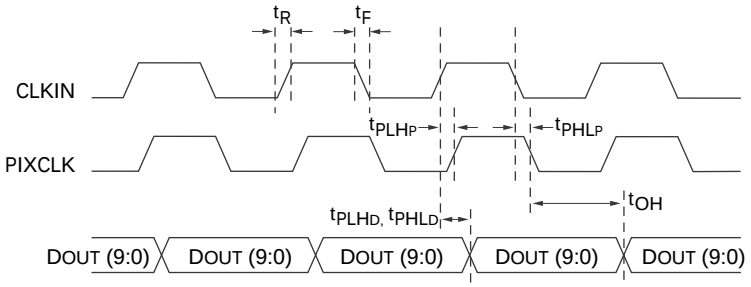


Propagation Delay for PIXCLK and DOUT

The DOUT signals change on the rising edge of the master input clock, as shown in Figure 34. LINE_VALID asserts at the same time as the first valid pixel data, at the start of a line, and remains asserted until the end of the final valid pixel data for the line.

The timing and behavior of PIXCLK depends on the Reg0x0A settings

Figure 34: Propagation Delays for PIXCLK and DOUT Signals



Two-Wire Serial Bus Timing

The two-wire serial bus operation requires certain minimum master clock cycles between transitions. These are specified in the following diagrams in master clock cycles.

Figure 35: Serial Host Interface Start Condition Timing

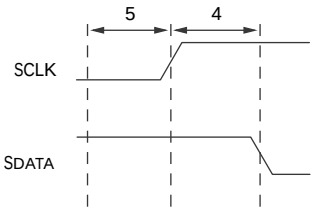
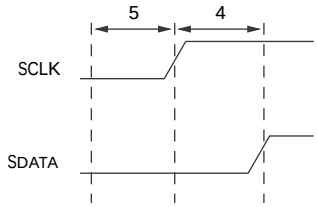
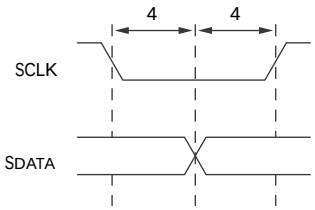


Figure 36: Serial Host Interface Stop Condition Timing



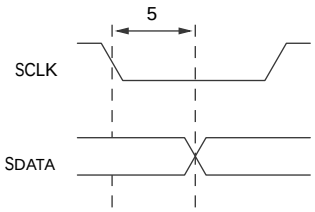
NOTE:
All timing in master clock cycle units.

Figure 37: Serial Host Interface Write Data Timing



NOTE:
SDATA is driven by an off-chip transmitter.

Figure 38: Serial Host Interface Read Data Timing



NOTE:
SDATA is pulled low by the sensor, or allowed to be pulled high by an off-chip pull-up resistor.

Figure 39: Acknowledge Signal Timing Following 8-Bit Write to Sensor

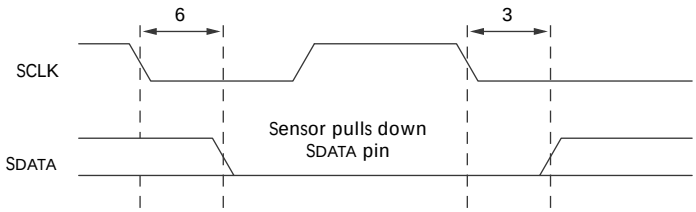
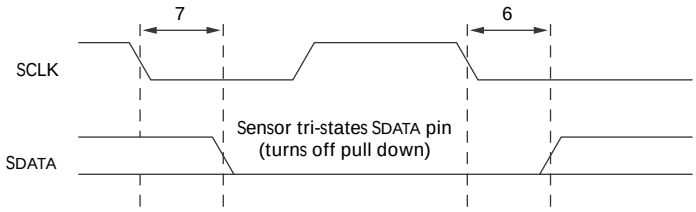


Figure 40: Acknowledge Signal Timing Following 8-Bit Read from Sensor



NOTE:
After a read, the master receiver must pull down SDATA to acknowledge receipt of data bits. When read sequence is complete, the master must generate a no-acknowledge by leaving SDATA to float high. On the following cycle, a start or stop bit can be used.

Figure 41: Spectral Response (TBD)

Data Sheet Designation

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

8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900
E-mail: prodmktg@micron.com, Internet: http://www.micron.com, Customer Comment Line: 800-932-4992

Revision History

Rev A, Preliminary 11/04

- Initial Release of document

8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900
E-mail: prodmktg@micron.com, Internet: http://www.micron.com, Customer Comment Line: 800-932-4992