1/2.5-Inch 5-Megapixel CMOS Digital Image Sensor

MT9P001I12STC

For the latest data sheet, refer to Micron's Web site: www.micron.com/imaging

Features

- DigitalClarity™ imaging technology
- High frame rate
- Superior low-light performance
- Low dark current
- Global reset release, which starts the exposure of all rows simultaneously
- Bulb exposure mode, for arbitrary exposure times
- Snapshot mode to take frames on demand
- Horizontal and vertical mirror image
- Column and row skip modes to reduce image size without reducing field-of-view
- Column and row binning modes to improve image quality when resizing
- Simple two-wire serial interface
- Programmable controls: gain, frame rate, frame size, exposure

Applications

- Digital still cameras
- Digital video cameras
- PC cameras
- Converged DSCs/camcorders
- Cellular phones
- PDAs

General Description

The Micron® Imaging MT9P001 is a 1/2.5-inch CMOS active-pixel digital image sensor with an active imaging pixel array of 2,592H x 1,944V. It incorporates sophisticated camera functions on-chip such as windowing, column and row skip mode, and snapshot mode. It is programmable through a simple two-wire serial interface.

The 5-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.

Table 1: Key Performance Parameters

Parameter		Typical Value
Optical format		1/2.5-inch (4:3)
Active imager size		5.70mm(H) x 4.28mm(V) 7.13mm (diagonal)
Active pixels		2,592H x 1,944V
Pixel size		2.2µm x 2.2µm
Color filter array		RGB Bayer pattern
Shutter type		Global reset release (GRR), Snapshot only Electronic rolling shutter (ERS)
Maximum data rate/ master clock		96 Mp/s at 96 MHz (2.8V I/O) 64 Mp/s at 64 MHz (1.8V I/O)
Frame rate	Full resolution	Programmable up to 12 fps
	VGA (640 x 480)	Programmable up to 30 fps
ADC resolution		12-bit, on-chip
Responsivity		0.53 V/lux-sec (550nm)
Pixel dynamic range		66.5dB
SNR_{MAX}		40.5dB
Supply Voltage	I/O	1.8V—3.1V
	Digital	1.7V—1.9V (1.8V nominal)
	Analog	2.6V—3.1V (2.8V nominal)
Power consumption		<317mW
Operating temperature		-30°C to +70°C
Packaging		48-pin iLCC, Die

Ordering Information

Table 2: Available Part Numbers

Part Number	Description
MT9P001C12STC	48-pin iLCC

Table of Contents

List of Figures

List of Tables

General Description (continued)

The sensor can be operated in its default mode or programmed by the user for frame size, exposure, gain setting, and other parameters. The default mode outputs a full resolution image at 12 frames per second (fps).

An on-chip analog-to-digital converter (ADC) provides 10 bits per pixel. FRAME_VALID and LINE_VALID signals are output on dedicated pins, along with a pixel clock that is synchronous with valid data.

The MT9P001 produces extraordinarily clear, sharp digital pictures, and its ability to capture both continuous video and single frames makes it the perfect choice for a wide range of consumer and industrial applications, including cell phones, digital still cameras, digital video cameras, and PC cameras.

Figure 1: Block Diagram

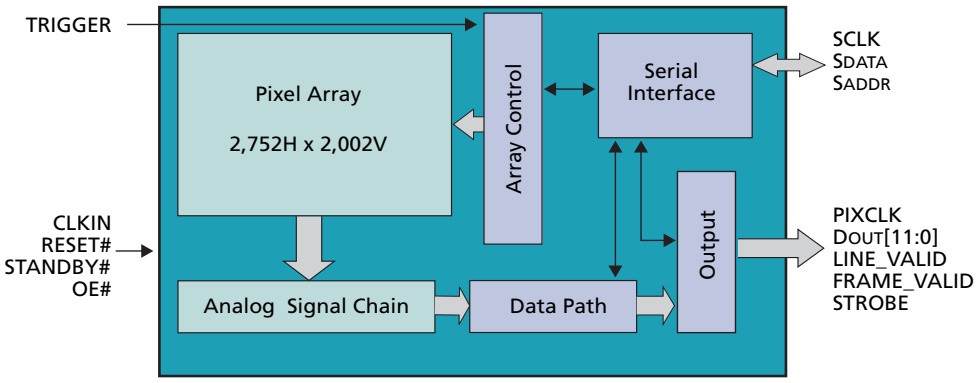
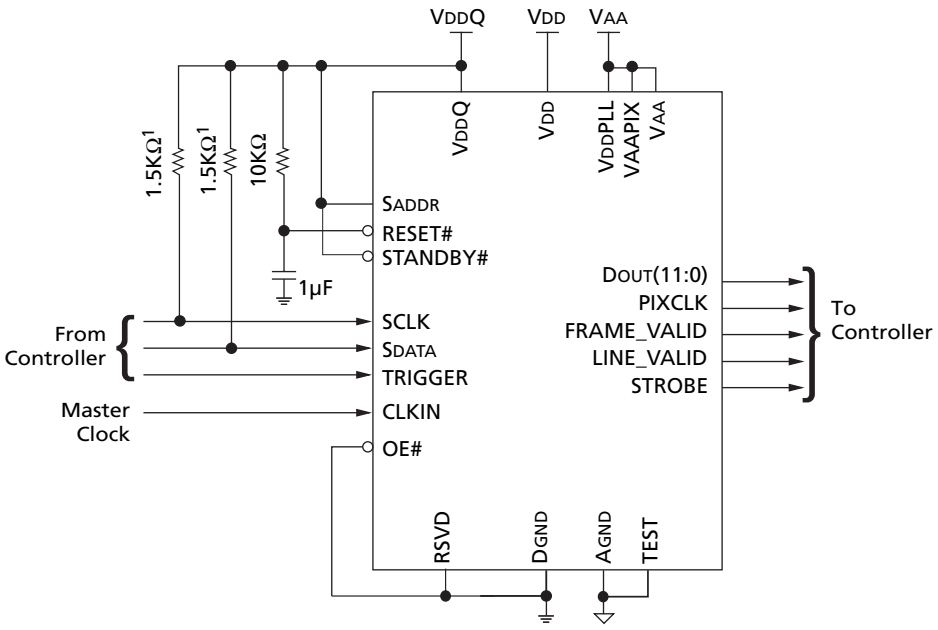


Figure 2: Typical Configuration (Connection)



Notes: 1. Resistor value 1.5KΩ is recommended, but may be greater for slower two-wire speed.
2. All power supplies should be adequately decoupled.

Figure 3: 48-Pin iLCC 10x10 Package Pinout Diagram (Top View)

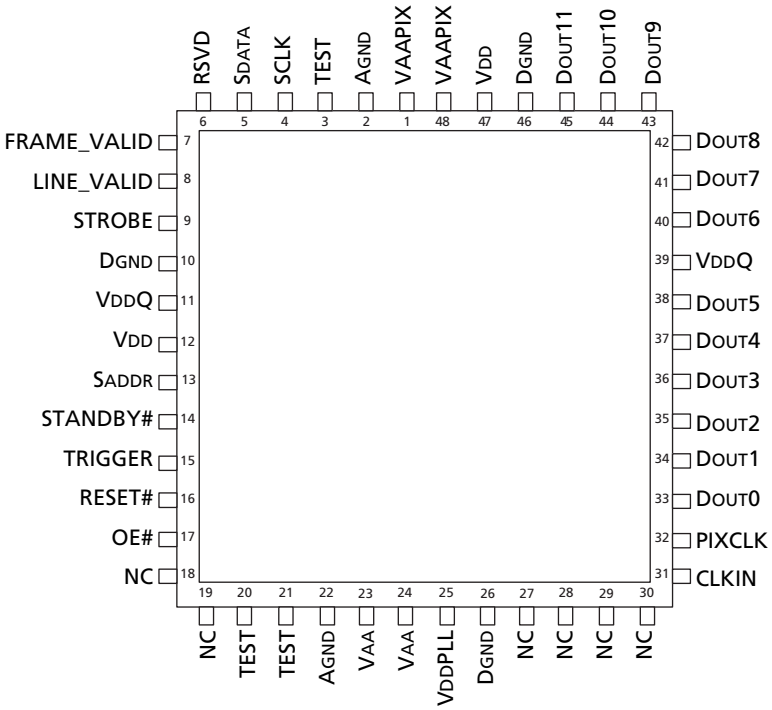


Table 3: Pin Description

Name	Type	Description
RESET#	Input	When LOW, the MT9P001 asynchronously resets. When driven HIGH, it resumes normal operation with all configuration registers set to factory defaults.
CLKIN	Input	Clock in.
SCLK	Input	Serial clock. Pull to VDDQ with a 1.5KΩ resistor.
OE#	Input	When HIGH, the PIXCLK, DOUT, FRAME_VALID, LINE_VALID, and STROBE outputs enter a High-Z. When driven LOW, normal operation resumes.
STANDBY#	Input	Standby. When LOW, the chip enters a low-power standby mode. It resumes normal operation when the pin is driven HIGH.
TRIGGER	Input	Snapshot trigger. Used to trigger one frame of output in snapshot modes, and to indicate the end of exposure in bulb exposure modes.
SADDR	Input	Serial address. When HIGH, the MT9P001 responds to device ID $(BA)_H$. When LOW, it responds to serial device ID $(90)_H$.
SDATA	I/O	Serial data. Pull to VDDQ with a 1.5KΩ resistor.
PIXCLK	Output	Pixel clock. The DOUT, FRAME_VALID, LINE_VALID, and STROBE outputs should be captured on the falling edge of this signal.
DOUT(11:0)	Output	Pixel data. Pixel data is 12-bit. MSB (DOUT11) through LSB (DOUT0) of each pixel, to be captured on the falling edge of PIXCLK.
FRAME_VALID	Output	Frame valid. Driven HIGH during active pixels and horizontal blanking of each frame and LOW during vertical blanking.

Table 3: Pin Description (continued)

Name	Type	Description
LINE_VALID	Output	Line valid. Driven HIGH with active pixels of each line and LOW during blanking periods.
STROBE	Output	Snapshot strobe. Driven HIGH when all pixels are exposing in snapshot modes.
VDD	Supply	Digital supply voltage. Nominally 1.8V.
VDDQ	Supply	IO supply voltage. Nominally 1.8V or 2.8V.
DGND	Supply	Digital ground.
VAA	Supply	Analog supply voltage. Nominally 2.8V.
VAAPIX	Supply	Pixel supply voltage. Nominally 2.8V, connected externally to VAA.
AGND	Supply	Analog ground.
VDDPLL	Supply	PLL supply voltage. Nominally 2.8V, connected externally to VAA.
TEST	—	Tie to AGND for normal device operation (factory use only).
RSVD	—	Tie to DGND for normal device operation (factory use only).
NC	—	No connect.

Pixel Data Format

Pixel Array Structure

The MT9P001 pixel array consists of a 2,752-column by 2,002-row matrix of pixels addressed by column and row. The address (column 0, row 0) represents the upper-right corner of the entire array, looking at the sensor as shown in Figure 4.

The array consists of a 2,592-column by 1,944-row active region in the center representing the default output image, surrounded by a boundary region (also active), surrounded by a border of dark pixels (see Table 4 and Table 5). The boundary region can be used to avoid edge effects when doing color processing to achieve a 2,592 x 1,944 result image, while the optically black column and rows can be used to monitor the black level.

Pixels are output in a Bayer pattern format consisting of four "colors"—Green1, Green2, Red, and Blue (G1, G2, R, B)—representing three filter colors. When no mirror modes are enabled, the first row output alternates between G1 and R pixels, and the second row output alternates between B and G2 pixels. The Green1 and Green2 pixels have the same color filter, but they are treated as separate colors by the data path and analog signal chain.

Table 4: Pixel Type by Column

Column	Pixel Type
0 – 9	Dark (10)
10 – 5	Active boundary (6)
16 – 2,607	Active image (2592)
2,608 – 2,617	Active boundary (10)
2,618 – 2,751	Dark (134)

Table 5: Pixel Type by Row

Row	Pixel Type
0 – 49	Dark (50)
50 – 53	Active boundary (4)
54 – 1997	Active image (1944)
1,998 – 2,000	Active boundary (3)
2,001	Dark (1)

Figure 4: Pixel Array Description

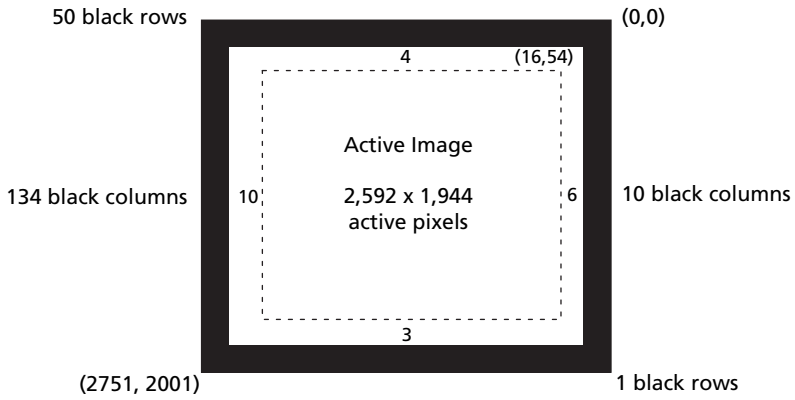
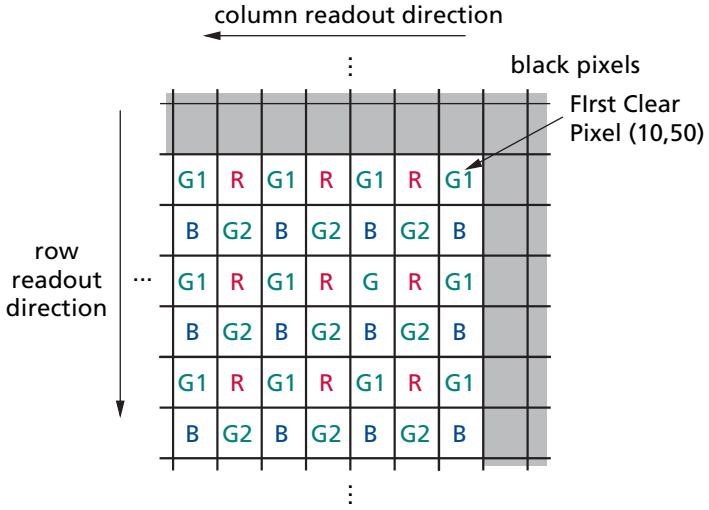


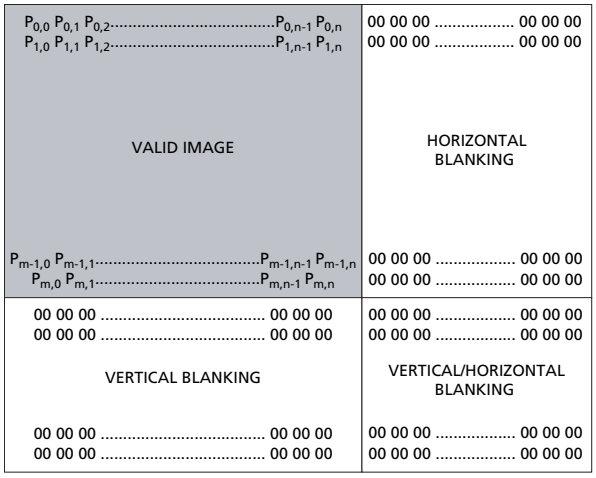
Figure 5: Pixel Color Pattern Detail (Top Right Corner)



Output Data Format

The MT9P001 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 on page 12. LINE_VALID is HIGH during the shaded region of the figure. FRAME_VALID timing is described in “Output Data Timing” on page 12.

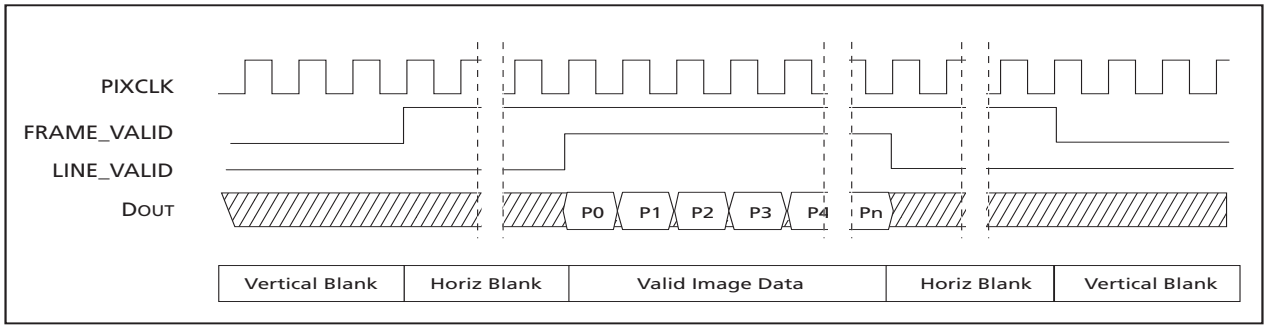
Figure 6: Spatial Illustration of Image Readout



Output Data Timing

The output images are divided into frames, which are further divided into lines. By default, the sensor produces 1,944 rows of 2,592 columns each. The FRAME_VALID and LINE_VALID signals indicate the boundaries between frames and lines, respectively. PIXCLK can be used as a clock to latch the data. For each PIXCLK cycle, one 12-bit pixel datum outputs on the DOUT pins. When both FRAME_VALID and LINE_VALID are asserted, the pixel is valid. PIXCLK cycles that occur when FRAME_VALID is negated are called vertical blanking. PIXCLK cycles that occur when only LINE_VALID is negated are called horizontal blanking.

Figure 7: Timing Example of Pixel Data



Power Up and Power Down

When first powering on the MT9P001, follow this sequence:

1. Ensure RESET# is asserted (LOW).
2. Bring up the supplies. If both the analog and the digital supplies cannot be brought up simultaneously, ensure the digital supply comes up first.
3. Negate RESET#.

To ensure that the I/Os do not load any of the buses that they are connected to, assert RESET#, and then remove the supplies.

Reset

The MT9P001 may be reset by either using RESET# (active LOW) or the reset register.

If RESET# is asserted (LOW), it is not necessary to clock the device. All registers return to the factory defaults. When the pin is negated (HIGH), the chip resumes normal operation.

If the reset register field is set to "1," all registers except the following reset:

- Chip_Enable
- Synchronize_Changes
- Reset
- Use_PLL
- Power_PLL
- PLL_m_Factor
- PLL_n_Divider
- PLL_p1_Divider

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

Clocks

The MT9P001 requires one clock (CLKIN), which is nominally 96 MHz. By default, this results in pixels being output on the DOUT pins at a maximum data rate of 96 Mp/s. With $V_{DD}Q$ = 1.8V, maximum master clock and maximum data rate become 64 MHz and 64 Mp/s, respectively. The CLKIN clock can be divided down internally by setting Divide_Pixel_Clock to a non-zero value. This slows down the operation of the chip as though CLKIN had been divided externally.

The DOUT, LINE_VALID, FRAME_VALID, and STROBE outputs are launched on the rising edge of PIXCLK, and should be captured on the falling edge of PIXCLK. The specific relationship of PIXCLK to these other outputs can be adjusted in two ways. If Invert_Pixel_Clock is set, the sense of PIXCLK is inverted from that shown in Figure 7 on page 12.

Additionally, if the pixel clock has been divided by Divide_Pixel_Clock, it can be shifted relative to the other outputs by setting Shift_Pixel_Clock.

$$f_{PIXCLK} = \begin{cases} f_{CLKIN} & \text{if Divide\_Pixel\_Clock = 0} \\ f_{CLKIN} / (2 \times \text{Divide\_Pixel\_Clock}) & \text{otherwise} \end{cases}$$

The MT9P001 has a PLL which can be used to generate the pixel clock internally. To use the PLL:

1. Bring the MT9P001 up as normal, then power on the PLL by setting Power_PLL. To use the PLL, f_{CLKIN} should be between 6 MHz and 27 MHz.
2. Set PLL_m_Factor, PLL_n_Divider, and PLL_p1_Divider as desired (see below).
3. Wait 1ms to ensure that the VCO has locked.
4. Set Use_PLL to switch from CLKIN to the PLL-generated clock.

To set the PLL parameters, determine the desired input (f_{CLKIN}) and output (f_{PIXCLK}) frequencies. Set the m, n, and p1 values to achieve the desired f_{PIXCLK} using these formulas:

$f_{PIXCLK} = (f_{CLKIN} \times m) / (n \times p1)$

$m = PLL\_m\_Factor$

$n = PLL\_n\_Divider + 1$

$p1 = PLL\_p1\_Divider + 1$

$2\ MHz < f_{CLKIN} / n < 13.5\ MHz$

$180\ MHz < (f_{CLKIN} \times m) / n < 360\ MHz$

It is desirable to keep (f_{CLKIN} / n) as large as possible within the limits. Also, “m” must be between 16 and 255, inclusive.

Standby and Chip Enable

The MT9P001 can be put in a low-power state either by pulling STANDBY# pin LOW or by clearing the Chip_Enable register field. When put in standby, all internal clocks are gated, and analog circuitry is put in a state that is meant to draw minimal power. The two-wire serial interface remains minimally active so that the Chip_Enable bit can subsequently be cleared. Reads cannot be performed and only the Chip_Enable and Invert_Standby registers are writable.

If the sensor was in continuous mode when put in standby, it resumes from where it was when standby is deactivated. Naturally, this frame and the next frame are corrupted, though the sensor itself does not realize this. As this could affect automatic black level calibration, it is recommended that either the chip be paused (by setting Restart_Pause) before being put in standby mode, or it be restarted (setting Restart) upon resumption of operation. For maximum power savings in standby mode, CLKIN should not be toggling.

When standby mode is entered, either by clearing Chip_Enable or by asserting STANDBY# pin, the PLL is disabled automatically or powered down. It must be manually re-enabled when leaving standby as needed.

Registers

Table 6: Register List and Default Values

Register Address (Hex)	Description	Default Value (Hex)
0x00	Chip_Version	0x1800
0x01	Row_Start	0x0036
0x02	Column_Start	0x0010
0x03	Row_Size	0x0797
0x04	Column_Size	0x0A1F
0x05	Horizontal_Blanking	0x0000
0x06	Vertical_Blanking	0x0019
0x07	Output_Control	0x1F82
0x08	Shutter_Width_Upper	0x0000
0x09	Shutter_Width_Lower	0x0797
0x0A	Pixel_Clock_Control	0x0000
0x0B	Restart	0x0000
0x0C	Shutter_Delay	0x0000
0x0D	Reset	0x0000
0x10	PLL_Control	0x0050
0x11	PLL_Config_1	0x6404
0x12	PLL_Config_2	0x0000
0x1E	Read_Mode_1	0x4006
0x20	Read_Mode_2	0x0040
0x22	Row_Address_Mode	0x0000
0x23	Column_Address Mode	0x0000
0x2B	Green1_Gain	0x0008
0x2C	Blue_Gain	0x0008
0x2D	Red_Gain	0x0008
0x2E	Green2_Gain	0x0008
0x35	Global_Gain	0x0008
0x49	Row_Black_Target	0x00A8
0x4B	Row_Black_Default_Offset	0x0028
0x5B	BLC_Sample_Size	0x0001
0x5C	BLC_Tune_1	0x005A
0x5D	BLC_Delta_Thresholds	0x2D13
0x5E	BLC_Tune_2	0x41FF
0x5F	BLC_Target_Thresholds	0x231D
0x60	Green1_Offset	0x0020
0x61	Green2_Offset	0x0020
0x62	Black_Level_Calibration	0x0000
0x63	Red_Offset	0x0020
0x64	Blue_Offset	0x0020
0xFF	Chip_Version_Alt	0x1800

Table 7: Register Description

Bits	Default (hex)	Description
0:0x00 R0 - Chip_Version [0x1800]		
15:8	0x18	Part_ID Two-digit BCD value typically derived from the reticle ID code (C18A). Read-only. Legal values: [0–255].
7:4	0x0	Analog_RevisionConstant value incremented with each mask change for the same Part ID. Read-only. Legal values: [0–15].
3:0	0x0	Digital_Revision Constant value incremented with each digital functionality change for the same Part ID. Read-only. Legal values: [0–15].
0:0x01 R1 - Row_Start [0x0036]		
10:0	0x036	Row_Start The Y coordinate of the upper-left corner of the Field of View. If this register is set to an odd value, the next lower even value will be used. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Causes a Bad Frame if written. Legal values: [0–2004], even.
0:0x02 R2 - Column_Start [0x0010]		
11:0	0x010	Column_Start The X coordinate of the upper-left corner of the Field of View. The value will be rounded down to the nearest multiple of 2 times the column bin factor. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Legal values: [0–2750], even.
0:0x03 R3 - Row_Size [0x0797]		
10:0	0x797	Row_Size The height of the field of view minus one. If this register is set to an even value, the next higher odd value will be used. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Causes a Bad Frame if written. Legal values: [1–2005], odd.
0:0x04 R4 - Column_Size [0x0A1F]		
11:0	0xA1F	Column_Size The width of the field of view minus one. If this register is set to an even value, the next higher odd value will be used. This register should be set to a multiple of 2 times the column bin factor minus 1. In other words, it should be (2*n*(Column_Bin + 1) - 1) for some integer n. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Causes a Bad Frame if written. Legal values: [1–2751], odd.
0:0x05 R5 - Horizontal_Blank [0x0000]		
11:0	0x000	Horizontal_Blank Extra time added to the end of each row, in pixel clocks. Incrementing this register will increase exposure and decrease frame rate. Setting a value less than the minimum will use the minimum horizontal blank. The minimum horizontal blank depends on the mode of the sensor. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Causes a Bad Frame if written. Legal values: [0–4095].
0:0x06 R6 - Vertical_Blank [0x0019]		
10:0	0x019	Vertical_Blank Extra time added to the end of each frame in rows minus one. Incrementing this register will decrease frame rate, but not affect exposure. Setting a value less than the minimum will use the minimum vertical blank. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Legal values: [8–2047].

Table 7: Register Description (continued)

Bits	Default (hex)	Description
0:0x07 R7 - Output_Control [0x1F82]		
1	0x01	Chip_Enable When clear, sensor readout is stopped and analog circuitry is put in a state which draws minimal power. When set, the chip operates according to the current mode. Writing this bit does not affect the values of any other registers.
0	0x00	Synchronize_Changes When set, changes to certain registers (those with the SC attribute) are delayed until the bit is clear. When cleared, all the delayed writes will happen immediately. Registers with the F attribute will still have the update synchronized to the next frame boundary.
0:0x08 R8 - Shutter_Width_Upper [0x0000]		
3:0	0x0	Shutter_Width_Upper The most significant bits of the shutter width, which are combined with Shutter Width Lower (R9). Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Legal values: [0–15].
0:0x09 R9 - Shutter_Width_Lower [0x797]		
15:0	0x797	Shutter_Width_Lower The least significant bits of the shutter width. This is combined with Shutter_Width_Upper and Shutter_Delay such that the effective shutter width is (((((Shutter_Width_Upper)*65536) + Shutter_Width_Lower)*t_ROW) - Shutter_Delay - C) in pixclks. This should allow a shutter width from about 50 us to about 50 s at default row time. If set to zero, a value of 1 will be used. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Legal values: [1–65535].
0:0x0A R10 - Pixel_Clock_Control [0x0000]		
15	0x0	Invert_Pixel_Clock When set, LINE_VALID, FRAME_VALID, and D_OUT should be captured on the rising edge of PIXCLK. When clear, they should be captured on the falling edge. This is accomplished by inverting the PIXCLK output.
10:8	0x0	Shift_Pixel_Clock Two's complement value representing how far to shift the PIXCLK output pin relative to DOUT, in CLKIN cycles. Positive values shift PIXCLK later in time relative to DOUT (and thus relative to the internal array/datapath clock). No effect unless PIXCLK is divided by Divide Pixel Clock. Legal values: [0–15].
6:0	0x0	Divide_Pixel_Clock Produces a PIXCLK that is divided by the value times two. The value must be zero or a power of 2. This will slow down the internal clock in the array control and datapath blocks, including pixel readout. It will not affect the SHIP interface clock. A value of 0 corresponds to a PIXCLK with the same frequency as CLKIN. A value of 1 means f_PIXCLK = (f_CLKIN / 2); 2 means f_PIXCLK = (f_CLKIN / 4); 64 means f_PIXCLK = (f_CLKIN / 128); etc. Legal values: {0, 1, 2, 4, 8, 16, 32, 64}.
0:0x0B R11 - Restart [0x0000]		
2	0x0	Trigger Setting this bit in Snapshot mode will cause the next trigger to occur as if the TRIGGER pin were properly asserted/negated. Ineffective if not in Snapshot mode. The sense of this bit is NOT affected by Invert Trigger. When using this bit instead of the TRIGGER pin, make sure that either the trigger pin is continuously asserted, or that the pad is continuously negated and Invert_Trigger is set.
1	0x0	Pause_Restart When set, Restart will not automatically be cleared. Instead, the sensor will pause at row 0 after Restart is set. When Pause_Restart is cleared, the sensor will resume. This allows for a repeatable delay from clearing restart to FRAME_VALID.

Table 7: Register Description (continued)

Bits	Default (hex)	Description
0	0x0	Restart Setting this bit will cause the sensor to abandon the current frame and restart from the first row. It will take up to 2*t_ROW for the restart to take effect. This bit resets to 0 automatically unless Pause_Restart is set. Volatile.
0:0x0C R12 - Shutter_Delay [0x0000]		
12:0	0x0000	Shutter_Delay A negative adjustment to the effective shutter width in aclks. See Shutter_Width_Lower. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Legal values: [0–8191].
0:0x0D R13 - Reset [0x0000]		
0	0x0	Reset Setting this bit will put the sensor into reset mode, which will set the sensor to its default power-up state and cause it to halt. Clearing this bit will resume normal operation. This is equivalent to pulling the RESET# pin LOW, except that the two-wire serial interface remains functional.
0:0x10 R16 - PLL_Control [0x0050]		
1	0x0	Use_PLL When set, use the PLL output as the system clock. When clear, use CLKIN as the system clock.
0	0x0	Power_PLL When set, the PLL is powered. When clear, it is not powered.
0:0x11 R17 - PLL_Config_1 [0x6404]		
15:8	0x64	Legal values: [16–255].
5:0	0x04	PLL_n_Divider PLL output frequency divider minus 1. Legal values: [0–63].
0:0x12 R18 - PLL_Config_2 [0x0000]		
4:0	0x00	PLL_p1_Divider PLL system clock divider minus 1. Legal values: [0–127].
0:0x1E R30 - Read_Mode_1 [0x4006]		
14	0x1	Anti_Bloom When set, multiple reset mode will be used to control blooming: rows which are not sampled will be held in reset. When clear, multiple reset mode will be disabled and there will be no anti blooming in row skip modes. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes.
10	0x0	Continuous_Line_Valid When set, produce the LINE_VALID signal even during the vertical blank period. When clear, produce LINE_VALID only when active rows are being read out (that is, only when FRAME_VALID is high).
9	0x0	When set, the sense of the TRIGGER input pin will be inverted. Invert_Trigger.
8	0x0	Snapshot When set, the sensor enters snapshot mode, and will wait for a trigger event between frames. When clear, the sensor is in continuous mode. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes.
7	0x0	Global_Reset When set, the Global Reset Release shutter will be used. When clear, the Electronic Rolling Shutter will be used. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes.

Table 7: Register Description (continued)

Bits	Default (hex)	Description
6	0x0	Bulb_Exposure When set, exposure time will be controlled by an external trigger. When clear, exposure time will be controlled by the Shutter_Width_Lower and Shutter_Width_Upper registers. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes.
5	0x0	Invert_Strobe When set, the STROBE signal will be normally HIGH, except during exposure, when it will be LOW. When clear, the STROBE signal is normally LOW except during exposure.
4	0x0	Strobe_Enable When set, a strobe signal will be generated by the digital logic during integration. When clear, the strobe pin will be set to the value of Invert_Strobe.
3:2	0x1	Strobe_Start Determines the timepoint when the strobe is asserted. 0: first trigger 1: simultaneous exposure 2: shutter width 3: second trigger Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Legal values: [0–3].
1:0	0x2	Strobe_End Determines the timepoint when the strobe is negated. If this is set equal to or less than Strobe_Start, the width of the strobe pulse will be t_ROW. See Strobe_Start. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Legal values: [0–3].
0:0x20 R32 - Read_Mode_2 [0x0040]		
15	0x0	Mirror_Row When set, row readout in the active image occurs in reverse numerical order starting from (Row_Start + Row_Size). When clear, row readout of the active image occurs in numerical order. This has no effect on the readout of the dark rows. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Causes a Bad Frame if written.
14	0x0	Mirror_Column When set, column readout in the active image occurs in reverse numerical order, starting from (Column_Start + Column_Size). When clear, column readout of the active image occurs in numerical order. This has no effect on the readout of the dark columns. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes.
12	0x0	Show_Dark_Columns When set, dark columns will be output to the left of the active image, making the output image wider. This has no effect on integration time or frame rate. When clear, only columns that are part of the active image will be output. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes.
11	0x0	Show_Dark_Rows When set, dark rows will be output before the active image rows, making the output image taller. This has no effect on integration time or frame rate. Issue a Restart after setting this register bit. When clear, only rows from the active image will be output. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes.
6	0x1	Row_BLC When set, digitally compensate for differing black levels between rows by adding Dark Target (R73) and subtracting the average value of the 8 same-color dark pixels at the beginning of the row. When clear, digitally add Row Black Default Offset (R75) to the value of each pixel. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes.

Table 7: Register Description (continued)

Bits	Default (hex)	Description
5	0x0	Column_Sum When set, column summing will be enabled, and in column bin modes, all sampled capacitors will be enabled for column readout, resulting in an effective gain equal to the column bin factor. When clear, column averaging will be done, and there will be no additional gain related to the column bin factor. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes.
4	0x0	Manual_Channel_Offset When set, the channel offsets written to Green1_Channel_Offset, Red_Channel_Offset, Blue_Channel_Offset, and Green2_Channel_Offset will be applied. When clear, the offset values will be calculated automatically.
0:0x22 R34 - Row_Address_Mode [0x0000]		
5:4	0x0	Row_Bin The number of rows to be read and averaged per row output minus one. The rows will be read independently into sampling capacitors, then averaged together before column readout. For normal readout, this should be 0. For Bin 2x, it should be 1; for Bin 3x, it should be 2; for Bin 4x, it should be 3. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Causes a Bad Frame if written. Legal values: [0–3].
2:0	0x0	Row_Skip The number of row-pairs to skip for every row-pair output. A value of zero means to read every row. For Skip 2x, this should be 1; for Skip 3x, it should be 2, and so on. This value should be no less than Row_Bin. For full binning, Row_Skip should equal Row_Bin. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Causes a Bad Frame if written. Legal values: [0–7].
0:0x23 R35 - Column_Address_Mode [0x0000]		
5:4	0x0	Column_Bin The number of columns to be read and averaged per column output minus one. For normal readout, this should be zero. For Bin 2x, it should be 1; for Bin 4x, it should be 3. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Causes a Bad Frame if written. Legal values: {0, 1, 3}.
2:0	0x0	Column_Skip The number of column-pairs to skip for every column-pair output. A value of zero means to read every column in the active image. For Skip 2x, this should be 1; for Skip 3x, this should be 2, and so on. This value should be no less than Column_Bin. For full binning, Column_Skip should equal Column_Bin. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Causes a Bad Frame if written. Legal values: [0–6].
0:0x2B R43 - Green1_Gain [0x0008]		
14:8	0x00	Green1_Digital_Gain Digital Gain for the Green1 channel minus 1 times 8. The actual digital gain is (1 + value/8), and can range from 1 (a setting of 0) to 16 (a setting of 120) in increments of 1/8. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile. Legal values: [0–120].
6	0x00	Green1_Analog_Multiplier Analog gain multiplier for the Green1 channel minus 1. If 1, an additional analog gain of 2x will be applied. If 0, no additional gain is applied. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile.

Table 7: Register Description (continued)

Bits	Default (hex)	Description
5:0	0x08	Green1_Analog_Gain Analog gain setting for the Green1 channel times 8. The effective gain for the channel is (((Green1_Digital_Gain/8) + 1) * (Green1_Analog_Multiplier + 1) * (Green1_Analog_Gain/8)). Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile. Legal values: [8–63].
0:0x2C R44 - Blue_Gain [0x0008]		
14:8	0x00	Blue_Digital_Gain Digital Gain for the Blue channel minus 1 times 8. The actual digital gain is (1 + value/8), and can range from 1 (a setting of 0) to 16 (a setting of 120) in increments of 1/8. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile. Legal values: [0–120].
6	0x00	Blue_Analog_Multiplier Analog gain multiplier for the Blue channel minus 1. If 1, an additional analog gain of 2x will be applied. If 0, no additional gain is applied. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile.
5:0	0x08	Blue_Analog_Gain Analog gain setting for the Blue channel times 8. The effective gain for the channel is (((Blue_Digital_Gain/8) + 1) * (Blue_Analog_Multiplier + 1) * (Blue_Analog_Gain/8)). Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile. Legal values: [8–63].
0:0x2D R45 - Red_Gain [0x0008]		
14:8	0x00	Red_Digital_Gain Digital Gain for the Red channel minus 1 times 8. The actual digital gain is (1 + value/8), and can range from 1 (a setting of 0) to 16 (a setting of 120) in increments of 1/8. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile. Legal values: [0–120].
6	0x00	Analog gain multiplier for the Red channel minus 1. If 1, an additional analog gain of 2x will be applied. If 0, no additional gain is applied. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile.
5:0	0x08	Red_Analog_Gain Analog gain setting for the Red channel times 8. The effective gain for the channel is (((Red_Digital_Gain/8) + 1) * (Red_Analog_Multiplier + 1) * (Red_Analog_Gain/8)).
0:0x2E R46 - Green2_Gain [0x0008]		
14:8	0x00	Green2_Digital_Gain Digital Gain for the Green2 channel minus 1 times 8. The actual digital gain is (1 + value/8), and can range from 1 (a setting of 0) to 16 (a setting of 120) in increments of 1/8. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile. Legal values: [0–120]. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile. Legal values: [8–63].
6	0x00	Green2_Analog_Multiplier Analog gain multiplier for the Green2 channel minus 1. If 1, an additional analog gain of 2x will be applied. If 0, no additional gain is applied. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile.
5:0	0x08	Green2_Analog_Gain Analog gain setting for the Green2 channel times 8. The effective gain for the channel is (((Green2_Digital_Gain/8) + 1) * (Green2_Analog_Multiplier + 1) * (Green2_Analog_Gain/8)). Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Volatile. Legal values: [8–63].

Table 7: Register Description (continued)

Bits	Default (hex)	Description
0:0x35 R53 - Global_Gain [0x0008]		
15:0	0x0008	Global_Gain Writing the Global_Gain sets all four individual gain registers R43-R46 to the value. This register should not be read. See Green1_Gain (R43) for a description of the various fields. Write-only. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Duplicate. Legal values: special.
0:0x49 R73 - Row_Black_Target [0x00A8]		
11:0	0x0A8	Row_Black_Target The target black level for the Row BLC algorithm. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Legal values: [0–4095].
0:0x4B R75 - Row_Black_Default_Offset [0x0028]		
11:0	0x028	Row_Black_Default_Offset A two's-complement offset digitally added to all active image pixel values when Row BLC (R30[6]) is disabled. Writes are synchronized to frame boundaries. Affected by Synchronize_Changes. Legal values: [-2048–2047].
0:0x5B R91 - BLC_Sample_Size [0x0001]		
0	0x1	If set, the "moving average" calculation in the BLC algorithm will use a sample size of 32. If clear, it will use a sample size of 1 (i.e., each frame's black level will be considered independent of other frames).
0:0x5C R92 - BLC_Tune_1 [0x005A]		
11:8	0x0	BLC_Delta_Damping A number subtracted from the calculated correction's magnitude when in delta mode. Setting this to a positive number will correct by that much less than the delta value. A negative (two's complement) number will correct by more (possibly worsening the overshoot). This applies to the magnitude of the delta, so a positive damping value will be subtracted from a positive delta and added to a negative delta. Writes are synchronized to frame boundaries. Legal values: [-8–7].
7:0	0x5A	BLC_DAC_Settling_Time The number of pixclks it takes for a newly set offset to take effect divided by 2. Used to configure the fast sample algorithm. After setting a calibration value in fast sample mode, (value * 2) pixclks will elapse before the next sample is taken. Writes are synchronized to frame boundaries. Legal values: [0–255].
0:0x5D R93 - BLC_Delta_Thresholds [0x2D13]		
14:8	0x2D	BLC_High_Delta_Threshold Upper delta threshold divided by 4. If the average black value for a color is higher than this value times 4 or lower than BLC_Low_Delta_Threshold times 4, the fast sampling and binary search modes will be activated (if enabled). Once the black level is between the BLC_High_Delta_Threshold and the BLC_Low_Delta_Threshold, the delta adjustment mode will be used (though fast sample mode will continue until the end of the frame). This value should be set no lower than BLC High Target Threshold. Writes are synchronized to frame boundaries. Legal values: [0–127].
6:0	0x13	BLC_Low_Delta_Threshold Lower delta threshold divided by 4. See BLC_High_Delta_Threshold. Should be no higher than BLC_Low_Target_Threshold. Writes are synchronized to frame boundaries. Legal values: [0–127].

Table 7: Register Description (continued)

Bits	Default (hex)	Description
0:0x5E R94 - BLC_Tune_2 [0x41FF]		
14:12	0x4	BLC_Step_Size Base 2 log of the change in pixel value (in LSBs) of a pixel when the analog offset is changed by one Legal values: [0–4].
8:0	0x1FF	BLC_Max_Adjust The maximum adjustment (positive or negative) that the BLC delta adjustment mode is allowed to make to the analog offset. Legal values: [1– 511].
0:0x5F R95 - BLC_Target_Thresholds [0x231D]		
14:8	0x23	BLC_High_Target_Threshold The upper target threshold of the BLC algorithm divided by 4. The target black value is 4 times the average of the BLC_High_Target_Threshold and the BLC_Low_Target_Threshold. When the black value for a color is within these thresholds, it will be considered to be on target. Writes are synchronized to frame boundaries. Legal values: [0–127].
6:0	0x1D	BLC_Low_Target_Threshold The lower target threshold for the BLC algorithm divided by 4. See BLC High Target Threshold above. Writes are synchronized to frame boundaries. Legal values: [0–27].
0:0x60 R96 - Green1_Offset [0x0020]		
8:0	0x020	Green1_Offset Two's-complement representation of the analog offset value for Green1. If Manual_BLC (R98[0]) is set, this value will be used as the analog offset. Otherwise, the value may be overridden by the BLC algorithm. When read, this register returns the offset currently in use. The user-programmed value is always retained internally, and may be read by setting Manual_BLC. A value of -256 will set the offset to -255. Writes are synchronized to frame boundaries. Legal values: [-255–255].
0:0x61 R97 - Green2_Offset [0x0020]		
8:0	0x020	Green2_Offset Two's-complement representation of the analog offset value for Green2. See Green1_Offset. Writes are synchronized to frame boundaries. Legal values: [-255–255].
0:0x62 R98 - Black_Level_Calibration [0x0000]		
15	0x0	Disable_Fast_Sample When set, the fast sampling mode (multiple samples per frame) will not be used if the black level falls outside the delta thresholds; instead, only one sample-adjust will take place per frame. Binary search mode may still be used. When clear, fast sample mode will be used when necessary. Writes are synchronized to frame boundaries.
14	0x0	Lock_Green_Calibration When set, the calibration offset chosen for Green1 will be used for Green2 pixels as well. Only effective if Green1_Analog_Gain equals Green2_Analog_Gain and Green1_Analog_Multiplier equals Green2_Analog_Multiplier. Writes are synchronized to frame boundaries.
13	0x0	Lock_Red_Blue_Calibration When set, the calibration offset chosen for Red will be used for Blue pixels as well. Only effective if Red_Analog_Gain equals Blue_Analog_Gain and Red_Analog_Multiplier equals Blue_Analog_Multiplier. Writes are synchronized to frame boundaries.
12	0x0	Recalculate_Black_Level When set, any running averages will be reset and the fast sample and binary search modes will be activated (if enabled). This bit always reads “0.” Writes are synchronized to frame boundaries.

Table 7: Register Description (continued)

Bits	Default (hex)	Description
11	0x0	Disable_Binary_Search When set, binary search mode will not be used when the black level falls outside the delta thresholds; instead the delta mode will be used. Fast sampling mode may still be used if enabled. Writes are synchronized to frame boundaries.
1	0x0	Disable_Calibration When set, analog calibration is disabled. When clear, the programmed or automatic offsets will be used.
0	0x0	Manual_BLC When set, the user programmed calibration offsets from R96-R97 and R99-R100 will be used. Also, black level calculation will be disabled. When clear, the BLC algorithm will adjust the offsets to maintain the target black level. Issue a Restart after clearing this register to avoid updating offsets based on corrupt black rows. If this bit is 1, Show_Dark_Rows must be set to allow channel offset correction to function properly. Writes are synchronized to frame boundaries.
0:0x63 R99 - Red_Offset [0x0020]		
8:0	0x020	Red_Offset Two's-complement representation of the analog offset value for Red. See Green1_Offset. Writes are synchronized to frame boundaries. Legal values: [-255–255].
0:0x64 R100 - Blue_Offset [0x0020]		
8:0	0x020	Blue_Offset Two's-complement representation of the analog offset value for Blue. See Green1_Offset. Writes are synchronized to frame boundaries. Legal values: [-255–255].
0:0xFF R255 - Chip_Version_Alt [0x1800]		
15:0	0x1800	Chip_Version_Alt Mirror of R0[15:0]. Read-only. Duplicate. Appears in all pages. Legal values: special.

Feature Description

Readout Sequence

Typically, the readout window is set to a region including only active pixels. The user has the option of reading out dark regions of the array, but if this is done, consideration must be given to how the sensor reads the dark regions for its own purposes.

Rows are read from the array in the following order:

1. Dark rows: If Show_Dark_Rows is set, or if Manual_BLC is clear, dark rows on the top of the array is read out. The set of rows sampled are adjusted based on the Row_Bin setting such that there are 8 rows after binning, as shown in the Table 8.

The Row_Skip setting is ignored for the dark row region.

If Show_Dark_Rows is clear and Manual_BLC is set, no dark rows are read from the array as part of this step, allowing all rows to be part of the active image. This does not change the frame time, as H_{DR} is included in the vertical blank period.

2. Active image: The rows defined by the row start, row size, bin, skip, and row mirror settings are read out. If this set of rows includes rows read out above, those rows are resampled, meaning that the data is invalid.

Table 8: Dark Rows Sampled as a Function of Row_Bin

Row_Bin	Dark Rows Sampled	HDR (Dark rows after binning)	BLC Sample Rows
0	2 – 9	8	4 – 7
1	2 – 17	8	6 – 13
3	2 – 33	8	10 – 25

Columns are read out in the following order:

1. Dark columns: If either Show_Dark_Columns or Row_BLC is set, dark columns on the left side of the image is read out followed by those on the right side. The set of columns read is shown in Table 9.

The Column_Skip setting is ignored for the dark columns.

If neither Show_Dark_Columns nor Row_BLC is set, no dark columns are read, allowing all columns to be part of the active image. This does not change the row time, as W_{DC} is included in the vertical blank period.

2. Active image: The columns defined by column start, column size, bin, skip, and column mirror settings are read out. If this set of columns includes the columns read out above, these columns are resampled, meaning the data is invalid.

Table 9: Dark Columns Sampled as a Function of Column_Bin

Column_Bin	Dark Columns Sampled	WDC (Dark columns after binning)	RBLC Sample Columns
0	2656 – 2735	80	2656 – 2727
1	2656 – 2735	40	2656 – 2735
3	2656 – 2735	20	2656 – 2735

Full-Array Readout

To read the entire array, including dark pixels, without digital processing or automatic black level adjustments, do the following:

1. Set Row_Start and Column_Start to 0.
2. Set Row_Size to 2,001.
3. Set Column_Size to 2,751.
4. Set Manual_BLC to 1.
5. Set Row_BLC to 0.
6. Set Row_Black_Default_Offset to 0.
7. Set Show_Dark_Rows and Show_Dark_Columns to 0.

If automatic analog (coarse) BLC is desired, but no digital processing, modify the above settings as follows:

1. Set Row_Start to 12.
2. Set Row_Size to 1,993.
3. Set Manual_BLC to 0.

These settings result in the same array layout as above, but only 22 dark rows are available at the top of the array; the first 8 are used in the black level algorithm, and there should be a two-row buffer between the black region and the active region in Anti_Bloom mode (the default).

Field of View

The extent of the pixel array that is used to create the output image (the field of view or FOV) is defined by four register fields. Column_Start and Row_Start define the X and Y coordinates of the upper-left corner of the FOV. Column_Size defines the width of the FOV, and Row_Size defines the height of the FOV in array pixels.

The Column_Start and Row_Start fields must be set to an even number. The Column_Size and Row_Size fields must be set to odd numbers (resulting in an even size for the FOV). The Row_Start register should be set no lower than 12 if either Manual_BLC is clear or Show_Dark_Rows is set.

If no special resolution modes are set (see below), the width of the output image, W, is (Column_Size + 1) and the height, H, is (Row_Size + 1).

Skip and Bin Modes

By default, the resolution of the output image is the full width and height of the FOV as defined above. The output resolution can be reduced by two methods: skipping and binning.

Row and column skip modes use subsampling to reduce the output resolution without reducing field-of-view. The MT9P001 also has row and column binning modes, which can reduce the impact of aliasing introduced by the use of skip modes. This is achieved by the averaging of 2 or 3 adjacent rows and columns (adjacent same-color pixels). Both 2x and 4x binning modes are supported. Rows and columns can be binned independently.

Skipping

Skipping reduces resolution by using only selected pixels from the FOV in the output image. A skip 2x mode skips one Bayer pair of pixels for every pair output. Skip 3x skips two pairs for each one pair output. Rows and columns are always read out in pairs. If skip 2x mode is enabled with otherwise default sensor settings, the columns in the output image corresponds to the pixel array columns 16, 17, 20, 21, 24, 25...

Skipping can be enabled separately for rows and columns. To enable skip mode, set either or both of Row_Skip and Column_Skip to the number of pixel pairs that should be skipped for each pair used in the output image. For example, to set column skip 2x mode, set Column_Skip to "1."

The size of the output image is reduced by the skip mode as shown in the following two equations:

W = 2 x ceil((Column_Size + 1) / (2 x (Column_Skip + 1)))

H = 2 x ceil((Row_Size + 1) / (2 x (Row_Skip + 1)))

Binning

Binning reduces resolution by combining adjacent same-color imager pixels to produce one output pixel. This can result in a more pleasing output image with reduced sub-sampling artifacts. It also improves low-light performance. For columns, the combination step can be either an averaging or summing operation. Depending on lighting conditions, one or the other may be desirable. In low-light conditions, summing produces a gain roughly equivalent to the column bin factor. Column summing may be enabled by setting Column_Sum.

Binning works in conjunction with skipping. Pixels that would be skipped because of the Column_Skip and Row_Skip settings can be averaged instead by setting Column_Bin and Row_Bin to the number of neighbor pixels to be averaged with each output pixel. For example, to set bin 2x mode, set Column_Skip and Column_Bin to 1. Additionally, Column_Start must be a multiple of (2 * (Column_Bin + 1)) and Row_Start must be a multiple of (2 * (Row_Bin + 1)).

Only certain combinations of binning and skipping are allowed.

These are shown in Table 10. If an illegal skip value is selected for a bin mode, a legal value is selected instead.

Table 10: Legal Values for Column_Skip Based on Column_Bin

Column_Bin	Legal Values for Column_Skip
0 (no binning)	0, 1, 2, 3, 4, 5, 6
1 (Bin 2x)	1, 3, 5
3 (Bin 4x)	3

Skip and bin modes are illustrated in Figure 8, Column Skip 2x; Row Skip 2x Enabled, Figure , , and Figure 10 on page 28. In skip mode, entire rows and columns of pixels are not sampled, resulting in a lower resolution output image. In bin mode, adjacent same-color pixels are averaged together to create one output pixel. All of the pixels in the FOV contribute to the output image in bin mode.

Figure 8: Column Skip 2x; Row Skip 2x Enabled

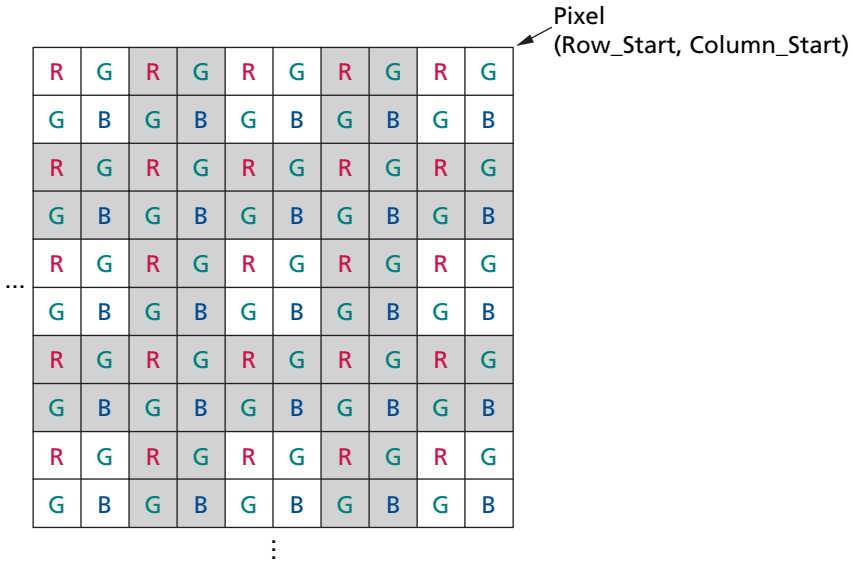
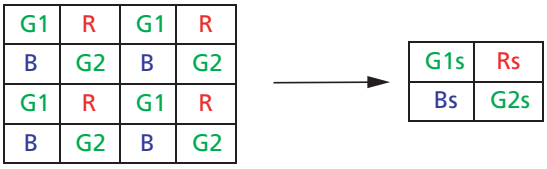
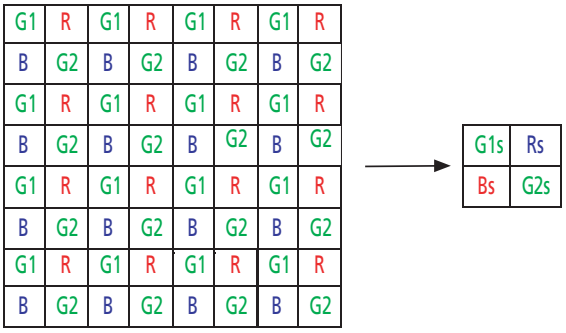


Figure 9: Bin 2x



Note: Gs1 = binning of 4 G1 pixels in a 4 x 4 window; Gs2 = binning of 4 G2 pixels in a 4 x 4 window.
Rs = binning of 4 R pixels in a 4 x 4 window; Bs = binning of 4 B pixels in a 4 x 4 window.

Figure 10: Bin 4x



Note: G1s = binning of 16 G1 pixels in an 8 x 8 window; G2s = binning of 16 G2 pixels in an 8 x 8 window.
Rs = binning of 16 R pixels in an 8 x 8 window; Bs = binning of B pixels in an 8 x 8 window.

Mirror

By default, active pixels in the resulting image are output in row-major order (an entire row is output before the next row is begun), from lowest row/column number to highest. If desired, the output (and sampling) order of the rows and columns can be reversed. This only affects pixels in the active region defined above, not any pixels read out as dark rows or dark columns. When the readout direction is reversed, the color order is reversed as well (red, green, red, etc., instead of green, red, green, etc. for example).

If row binning is combined with row mirroring, the binning is still done in the positive direction. Therefore, if the first output row in bin 2x + row mirror was 1,997, pixels on rows 1,997 and 1,999 would be averaged together. The next pixel output would be from rows 1,996 and 1,998, followed by the average of 1,993 and 1,995.

For column mirroring plus binning, the span of pixels used should be the same as with non-mirror mode.

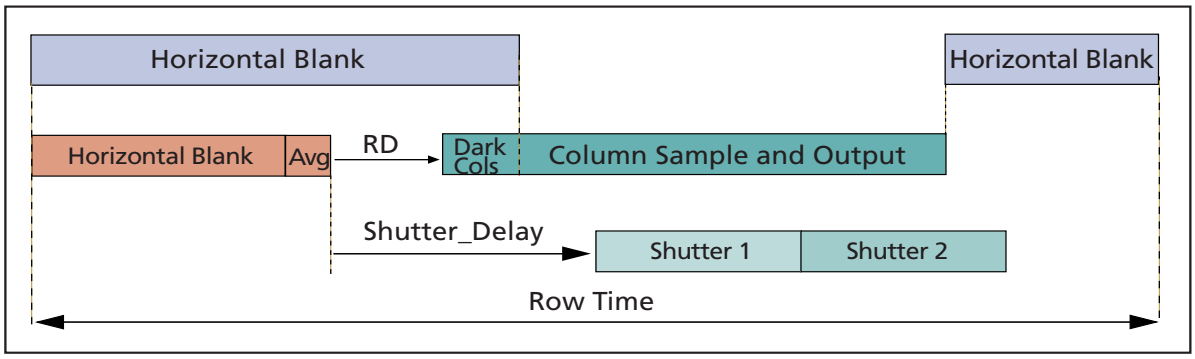
Frame Time and Exposure

Frame Time

Sensor timing can be described in terms of 3 time units: pixel clocks, array clocks, and row time. The pixel clock (PIXCLK) represents the time needed to sample 1 pixel from the array, and is typically equal to 1 CLKIN period. The sensor outputs data at the maximum rate of one pixel per PIXCLK. The array clock (ACLK) represents the sensor control time unit. It is half the frequency of the pixel clock. One row time (^{t}ROW) is the period from the first pixel output in a row to the first pixel output in the next row. Exposure times are typically specified in units of row time, although it is possible to fine tune exposures in units of ACLKs.

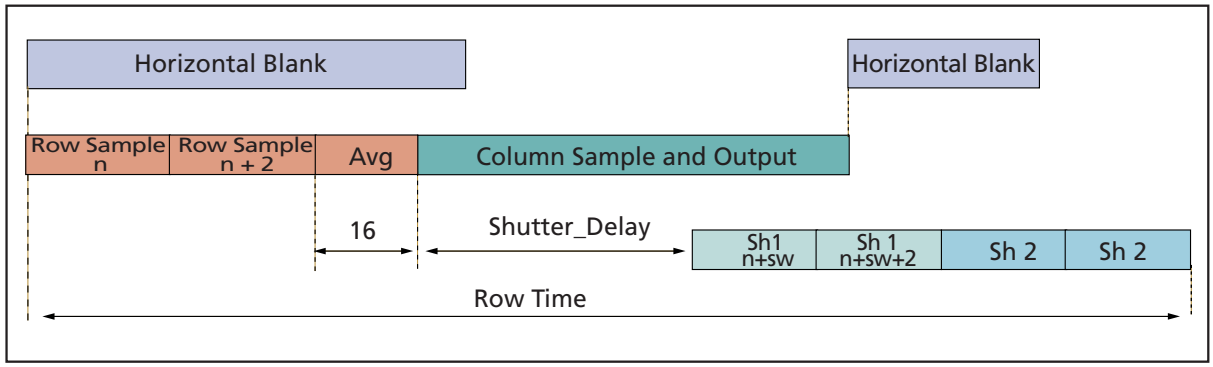
During each row time, a number of events must occur. The row time is the time required to complete all of these events. One row time is illustrated in Figure 11 on page 29. First, an entire row is sampled into holding capacitors. There are 16 ACLK cycles added after the row sample period which are used for the averaging operation in binning modes. Next, after some readout delay, each column is sampled in sequence, starting with the dark columns, and the pixel value is sent to the data path. Simultaneously, the shutter (and possibly second shutter—see below) clears out a future row. Any Horizontal_Blank time that was not used up during the row sample or readout delay is added to the end of the column sample time, allowing a user to ensure that the row time is not affected by changes to Shutter_Delay or other parameters besides the width of the output image.

Figure 11: Row Time



When row binning is used, multiple row sample and shutter sequences are required. Figure shows how the additional sample and shutter steps are inserted into the row time. In the example, the binning operation has pushed the final shutter past the end of horizontal blanking. Since the column sample is likely to be shorter in a binning mode, and there is extra time needed for the shutter, the shutter time can dominate the row time unless Horizontal_Blank is increased sufficiently to account for it.

Figure 12: Row Time with Row Bin 2x



The row time and frame time are defined by the following equations:

$$
\begin{aligned}
{}^{t}FRAME &= (H + \max(VB, VB_{min})) \times {}^{t}ROW \\
{}^{t}ROW &= {}^{L}ROW \times {}^{t}ACLK \\
{}^{L}ROW &= (W / 2) + \max(HB, HB_{min}) \\
VB &= Vertical\_Blank + 1 \\
HB &= Horizontal\_Blank + 1 \\
{}^{t}ACLK &= 2 \times {}^{t}PIXCLK \\
{}^{t}PIXCLK &= 1 / {}^{f}PIXCLK
\end{aligned}
$$

$$
{}^{f}PIXCLK = \begin{cases} {}^{f}CLKIN & \text{if Divide\_Pixel\_Clock) = 0} \\ {}^{f}CLKIN / (2x\ Divide\_Pixel\_Clock) & \text{otherwise} \end{cases}
$$

The minimum horizontal blank, HB_{min}, is determined by the binning modes and array timing configuration registers.

$HBmin = (696 * Row\_Bin) + 46 + Readout\_Delay + (W_{DC} / 2)$

The following constraint must also be observed when in Anti_Bloom + Row_Bin + Row_Mirror mode with (SW <4), which creates a maximum Shutter_Delay for a given Column_Size:

$(105 + W_{DC}/2 + W/2) > 57 + Shutter\_Delay + Row\_Bin \times 225$

Table 11: HB_{MIN} Values for Row_bin vs. Column_bin Settings

	Column_bin (W_{DC})			
Row_bin		**0**	**1**	**3**
	0	782	762	752
	1	1478	1458	1448
	3	2870	2850	2840

The minimum vertical blank, VB_{min}, is determined by the shutter width, SW, which is defined as follow:

$$VB_{min} = \max(H_{DR}, SW - H) + 1$$

$$SW = \max(1, (2^{16} \times Shutter\_Width\_Upper) + Shutter\_Width\_Lower)$$

LINE_VALID and FRAME_VALID

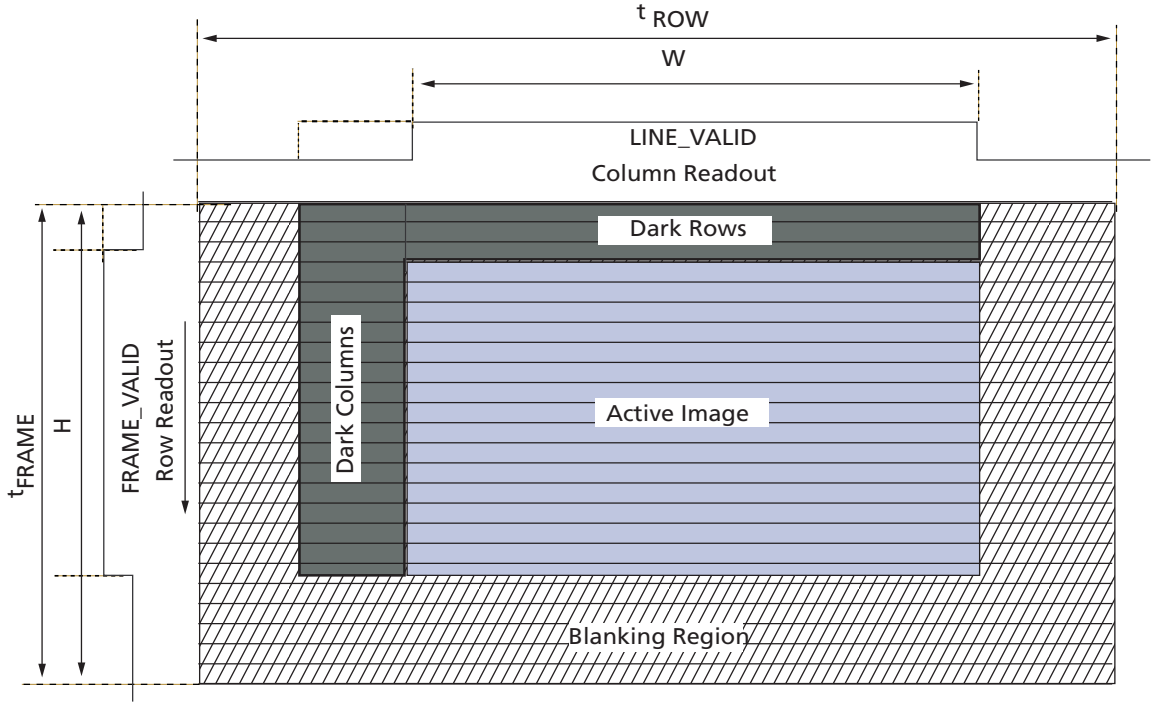
The timing of the FRAME_VALID and LINE_VALID outputs is closely related to the row time and the frame time.

FRAME_VALID is asserted for an integral number of row times, which is normally equal to the height of the output image. If Show_Dark_Rows is set, the dark sample rows are output before the active image, and FRAME_VALID is extended to include them. If Show_Dark_Rows is set, FRAME_VALID's leading edge happens at time 0. When Show_Dark_Rows is set and Vertical_Blank is less than 10, the length of FRAME_VALID would normally equal the frame time; instead, FRAME_VALID is negated for one pixclk cycle to guarantee a rising edge at the beginning of the frame.

LINE_VALID is asserted during the valid pixels of each row. If Show_Dark_Columns is set, the dark columns are output before the image pixels, and LINE_VALID are extended back to include them; in this case, the first pixel of the active image still occurs at the same position relative to the leading edge of FRAME_VALID. Normally, LINE_VALID is only asserted if FRAME_VALID is asserted; this is configurable as described below.

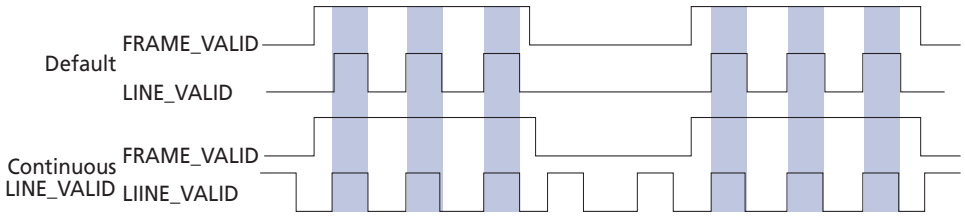
The timing of an entire frame is shown in Figure .

Figure 13: Frame Timing



The default situation is for LINE_VALID to be negated when FRAME_VALID is negated. The other option available is shown in Figure . If Continuous_LINE_VALID is set, LINE_VALID is asserted even when FRAME_VALID is not, with the same period and duty cycle.

Figure 14: LINE_VALID Format Options



Maintaining a Constant Frame Rate

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

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

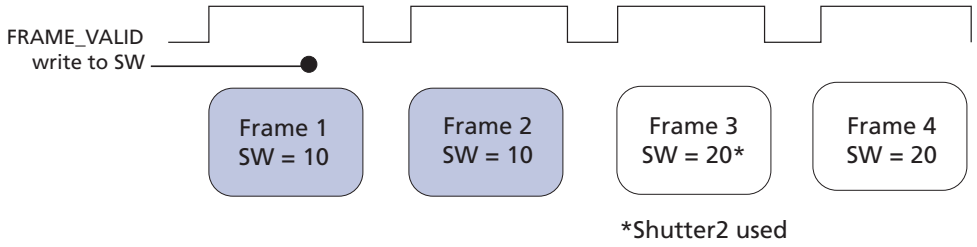
- Row_Start
- Row_Size
- Column_Size
- Horizontal_Blank
- Vertical_Blank
- Shutter_Delay
- Mirror_Row
- Row_Bin
- Row_Skip
- Column_Skip

The size of this bubble is (SW × t_{ROW}), calculating the row time according to the new settings.

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

Figure 15 shows the timing of writes to the shutter width registers. In the example, the shutter width is changed from 10 to 20 during Frame 1. The effective shutter width of Frame 3 is smaller than that of Frame 4 in the diagram, since Frame 3 was exposed using Shutter2 and Frame 4 was exposed using Shutter1.

Figure 15: Writing Shutter Width Registers



Synchronizing Register Writes to Frame Boundaries

Changes to most register fields that affect the size or brightness of an image take effect on the frame after the one during which they are written. These fields are noted as "synchronized to frame boundaries" in Table 6: Register List and Default Values on page 15. To ensure that a register update takes effect on the next frame, the write operation must be completed after the leading edge of FRAME_VALID and before the trailing edge of FRAME_VALID.

Additional control over the timing of register updates can be achieved by using Synchronize_Changes. If this bit is set, writes to certain register fields that affect the brightness of the output image do not take effect immediately. Instead, the new value is remembered internally. When Synchronize_Changes is cleared, all the updates simultaneously take effect on the next frame (as if they had all been written the instant Synchronize_Changes was cleared). Register fields affected by this bit are identified in Table 7: Register Description on page 16.

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

Restart

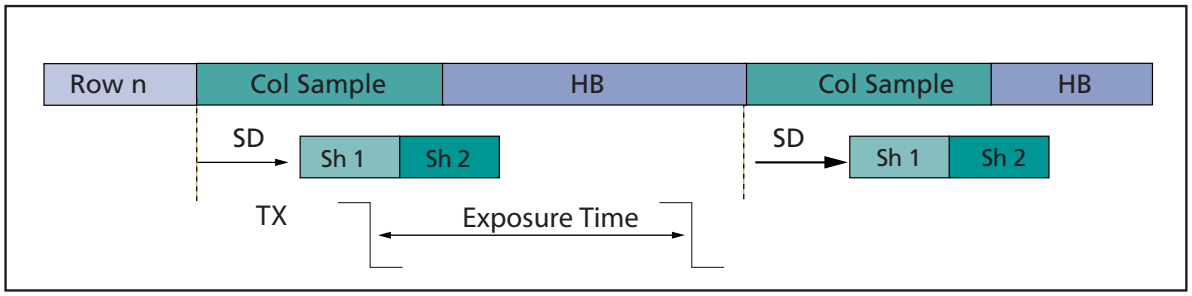
At any time during the operation of the sensor, a user may write a "1" to the Restart register. This has two effects: first, the current frame is interrupted immediately. Second, any writes to frame-synchronized registers and the shutter width registers takes effect immediately, and a new frame starts (in continuous mode). Register updates being held by Synchronize_Changes do not take effect until that bit is cleared. The current row and one following row completes before the new frame is started, so the time between issuing the Restart and the beginning of the next frame can vary by about ^{t}ROW.

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

Exposure

Exposure is defined in Figure . The time is measured from the trailing edge of TX (an internal array control signal) in the shutter sequence during the read of row n to the trailing edge of TX in the read sequence of row (n + SW). Increasing Shutter_Delay (SD) decreases the exposure time.

Figure 16: Exposure Time Shutter (Shutter_Width = 1)



The nominal exposure time, ^{t}EXP, is the effective shutter time in ERS modes, and is defined by the shutter width, SW, and the shutter overhead, SO, which includes the effect of Shutter_Delay. Exposure time for other modes is defined relative to this time.

$$^{t}EXP = (SW * {}^{t}ROW) - (SO * 2 * {}^{t}PIXCLK)$$

$$SO = 398 + (696 * Row\_Bin) + \min(Shutter\_Delay, SD\_max)$$

$$SD_{max} = L_{ROW} + (696 * Row\_Bin) + 46$$

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

Operating Modes

In the default operating mode, the MT9P001 continuously samples and outputs frames. It can be put in "snapshot" or triggered mode by setting Snapshot, which means that it samples and outputs a frame only when triggered. Toswitch between operating modes, it is necessary to first issue a Restart.

When in snapshot mode, the sensor can use the ERS or the GRR. The exposure can be controlled as normal, with the Shutter_Width_Lower and Shutter_Width_Upper registers, or it can be controlled using the external TRIGGER signal. The various operating modes are summarized in Table 12.

Table 12: Operating Modes

Mode	Settings	Description
ERS Continuous	Default	Frames are output continuously at the frame rate defined by $^{t}FRAME$. ERS is used, and the exposure time is electronically controlled to be ^{t}EXP.
ERS Snapshot	Snapshot = 1	Frames are output one at a time, with each frame initiated by a trigger. ERS is used, and the exposure time is electronically controlled to be ^{t}EXP.
ERS Bulb	Snapshot = 1; Bulb_Exposure = 1	Frames are output one at a time, with each frame's exposure initiated by a trigger. ERS is used. End of exposure and readout are initiated by a second trigger.
GRR Snapshot	Snapshot = 1; Global_Reset = 1	Frames are output one at a time, with each frame initiated by a trigger. GRR is used. Readout is electronically triggered based on SW.
GRR Bulb	Snapshot = 1; Bulb_Exposure = 1; Global_Reset = 1	Frames are output one at a time, with each frame initiated by a trigger. GRR is used. Readout is initiated by a second trigger.

Note: In ERS Bulb mode, SW must be greater than 5 (use trigger wider than $^{t}ROW*5$).

All operating modes share a common set of operations:

1. If necessary, place all rows in the reset state.
2. Wait for the first trigger, then start the exposure.
3. Wait for the second trigger, then start the readout.

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

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

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

The choice of shutter type is made by Global_Reset. If it is set, the GRR shutter is used; otherwise, ERS is used. The two shutters are described in "Electronic Rolling Shutter" on page 37 and "Global Reset Release" on page 38.

The default ERS continuous mode is shown in Figure 7 on page 12. In Figure 17 on page 36, (a) shows default signal timing for ERS snapshot modes, while (b) shows default signal timing for GRR snapshot modes.

Figure 17: ERS Snapshot Timing

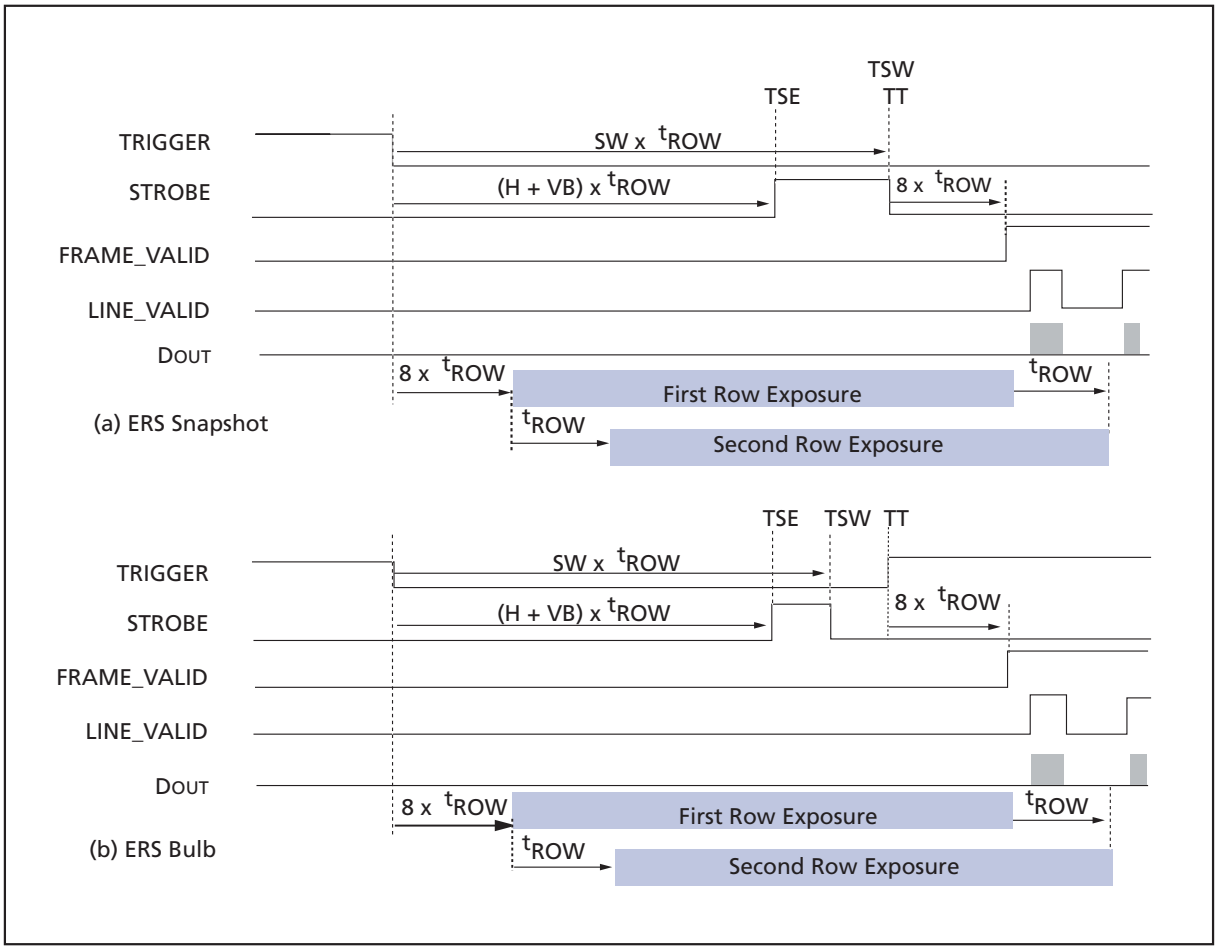
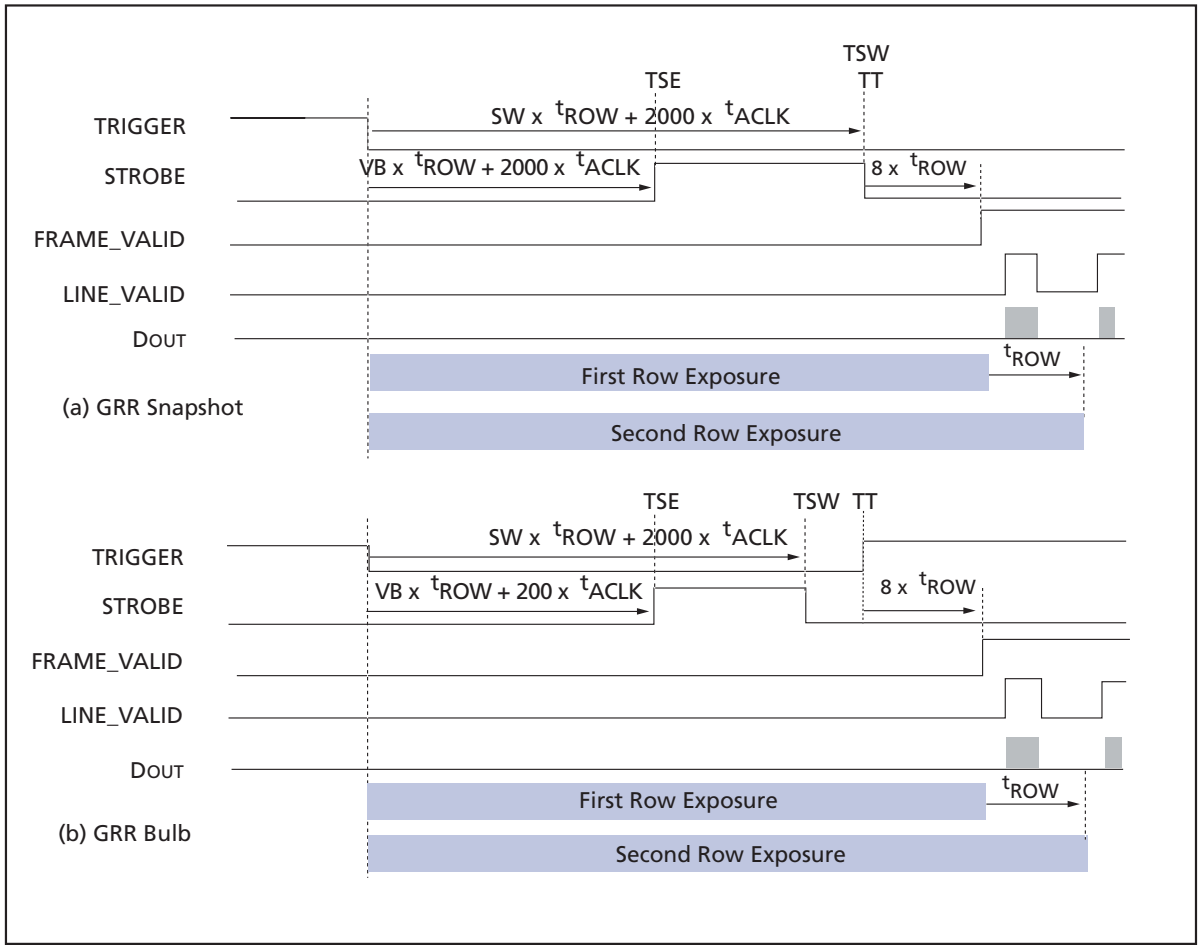


Figure 18: GRR Snapshot Timing



Electronic Rolling Shutter

The ERS modes take pictures by scanning the rows of the sensor twice in the order described in "Readout Sequence" on page 25. On the first scan, each row is released from reset, starting the exposure. On the second scan, the row is sampled, processed, and returned to the reset state. The exposure for any row is therefore the time between the first and second scans. Each row is exposed for the same duration, but at slightly different point in time, which can cause a shear in moving subjects.

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

$$^{t}ALLRESET = 16 \times 2004 \times {}^{t}ACLK$$

If Anti_Bloom is clear, this sequence does not occur.

Global Reset Release

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

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

Strobe Control

To support synchronization of the exposure with external events such as a flash or mechanical shutter, the MT9P001 produces a STROBE output. By default, this signal is asserted for approximately the time that all rows are simultaneously exposing, minus the vertical blank time, as shown in Figure 17 on page 36 and Figure 18 on page 37. Also indicated in these figures is the leading and trailing edges of STROBE can be configured to occur at one of several timepoints. The leading edge of STROBE occurs at STROBE_Start, and the trailing edge at STROBE_End, which are set to codes described in Table 13.

Table 13: STROBE Timepoints

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

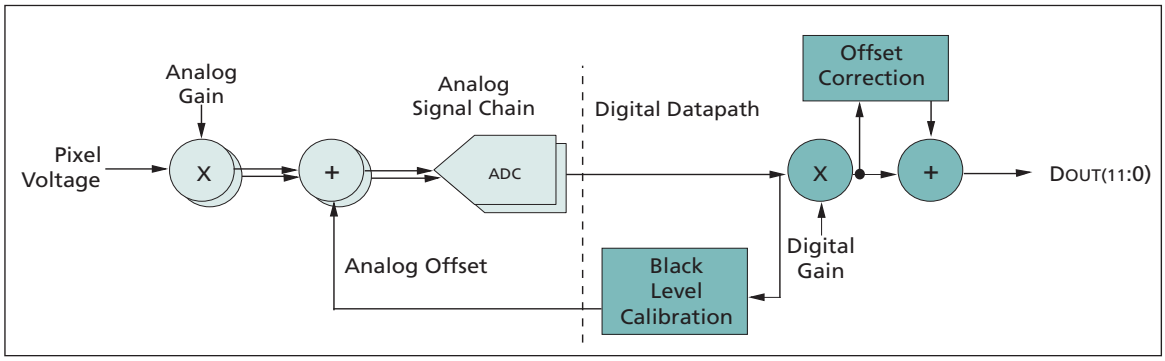
If STROBE_Start and STROBE_End are set to the same timepoint, the strobe is a ^{t}ROW wide pulse starting at the STROBE_Start timepoint. The sense of the STROBE port can be inverted by setting Invert_Strobe. In any case, a STROBE signal is only generated if Strobe_Enable is set.

Signal Path

The MT9P001 sensor signal chain and data path are shown below. Each color is processed independently, including separate gain and offset settings (though circuitry is shared where possible). Voltages sampled from the pixel array are first passed through an analog gain stage, which can produce gain factors between 1 and 8. An analog offset is then applied, and the signal is sent through a 12-bit analog-to-digital converter. Next, a digital gain factor of between 1 and 16 is applied, and then a digital offset of between -512 and 511 is added. The resulting 12-bit pixel value is then output on the Dout(11:0) ports.

The analog offset applied is determined automatically by the black level calibration algorithm, which attempts to shift the output of the analog signal chain so that black is at a particular level. The digital offset is a fine-tuning of the analog offset, and is calculated on a row-wise rather than a frame-wise basis.

Figure 19: Signal Path



Gain

There are two types of gain supported: analog gain and digital gain. Combined, gains of between 1 and 128 are possible. The recommended gain settings are shown in Table 14.

Table 14: Gain Increment Settings

Gain Range	Increments	Digital Gain	Analog Multipier	Analog Gain
1 – 4	0.125	0	0	8 – 32
4.25 – 8	0.25	0	1	17 – 32
9 – 128	1	1 – 120	1	32

Note: Analog gain should be maximized before applying digital gain.

The combined gain for a color C is given by:

$G_C = AG_C \times DG_C$.

Analog Gain

The analog gain is specified independently for each color channel. There are two components, the gain and the multiplier. The gain is specified by Green1_Analog_Gain, Red_Analog_Gain, Blue_Analog_Gain, and Green2_Analog_Gain in steps of 0.125. The analog multiplier is specified by Green1_Analog_Multiplier, Red_Analog_Multiplier, Blue_Analog_Multiplier, and Green2_Analog_Multiplier. These combine to form the analog gain for a given color C as shown in this equation:

$AG_C = (1 + C\_Analog\_Multiplier) \times (C\_Analog\_Gain / 8)$

The gain component can range from 0 to 7.875 in steps of 0.125, and the multiplier component can be either 0 or 1 (resulting in a multiplier of 1 or 2). However, it is best to keep the "gain" component between 1 and 4 for the best noise performance, and use the multiplier for gains between 4 and 8.

Digital Gain

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

$DG_C = 1 + (C\_Digital\_Gain / 8)$

Analog Black Level Calibration

The MT9P001 black level calibration circuitry is necessarily a feedback control system since adjustments to the analog offset are imprecise by nature. The goal is that within the dark row region of any supported output image size, the offset should have been adjusted such that the average black level falls within the specified target thresholds.

The analog offsets normally need a major adjustment only when leaving the Reset state or when there has been a change to a color's analog gain. Factors like shutter width and temperature have lower-order impact, and generally only require a minor adjustment to the analog offsets. The MT9P001 has various calibration modes to keep the system stable while still supporting the need for rapid offset adjustments when necessary.

The two basic steps of black level calibration are:

1. Take a sample.
2. If necessary, adjust the analog offset.

Black level calibration is normally done separately for each color channel, and different channels can be using different sample or adjustment methods at the same time. However, since both Green1 and Green2 pixels go through the same signal chain, and Red and Blue pixels likewise go through the same signal chain, it is expected that the chosen offset for these pairs should be the same as long as the gains are the same. If Lock_Green_Calibration is set, and (Green1_Analog_Gain = Green2_Analog_Gain) and (Green1_Analog_Multiplier = Green2_Analog_Multiplier), the calculated or user-specified Green1_Offset is used for both green channels. Similarly, if Lock_Red/Blue_Calibration is set, and (Red_Analog_Gain = Blue_Analog_Gain) and (Red_Analog_Multiplier = Blue_Analog_Multiplier), the calculated or user-specified Red_Offset is used for both the red and blue channels.

The currently-in-effect values of the offsets can be read from the Green1_Offset, Red_Offset, Blue_Offset, and Green2_Offset registers. Writes to these registers when Manual_BLC is set changes the offsets being used. In automatic BLC mode, writes to these registers are effective when manual mode is re-entered. In Manual_BLC mode, no sampling or adjusting takes place for any color.

Digital Black Level Calibration

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

The amount of shift necessary is determined by sampling the shielded dark pixels on either side of the array, which are always read out before the active pixels. A separate value is calculated for each color. Each of the active pixels Pi in the output image is then adjusted to take this black level into account:

$$Pcorr = Pi + Row\_Black\_Target - RBL$$

To avoid pulling saturated pixels out of saturation, Row_Black_Target should be set such that ((DG - 1) × 4096 + Row_Black_Target - RBL) is always positive (where DG is the effective digital gain for that color).

If digital black level calibration is not desired, Row_BLC can be cleared. In this case, a constant offset (Row_Black_Default_Offset) is added to each pixel. This default offset is the same for all colors. It can be used to account for black level shift due to digital gain, or to maintain a constant black level when switching Row_BLC on and off. To achieve the latter, use roughly:

$$Row\_Black\_Default\_Offset = Row\_Black\_Target - MBL$$

Serial Bus Description

Registers are written to and read from the MT9P001 through the two-wire serial interface bus. The MT9P001 is a serial interface slave and is controlled by the serial clock (SCLK), which is driven by the serial interface master. Data is transferred into and out of the MT9P001 through the serial data (SDATA) line. The SDATA line is pulled up to VDDQ off-chip by a 1.5KΩ resistor. Either the slave or master device can pull the SDATA line LOW—the serial interface protocol determines which device is allowed to pull the SDATA line down at any given time.

Protocol

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

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

Sequence

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

If the request was a write, the master then transfers the 8-bit register address to which a write should take place. The slave sends an acknowledge bit to indicate that the register address has been received. The master then transfers the data eight bits at a time, with the slave sending an acknowledge bit after each eight bits. The MT9P001 uses 16-bit data for its internal registers, thus requiring two 8-bit transfers to write to one register. After 16 bits are transferred, the register address is automatically incremented, so that the next 16 bits are written to the next register address. The master stops writing by sending a start or stop bit.

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

Bus Idle State

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

Start Bit

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

Stop Bit

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

Slave Address

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

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

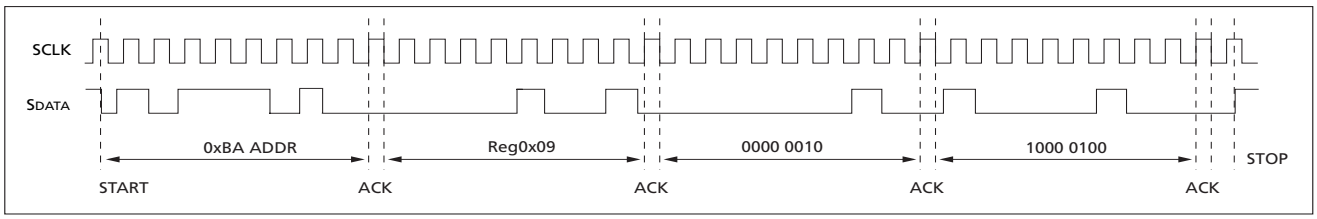
The no-acknowledge bit is generated when the data line is not pulled down by the receiver during the acknowledge clock pulse. A no-acknowledge bit is used to terminate a read sequence.

Two-Wire Serial Interface Sample Write and Read Sequences

16-Bit Write Sequence

A typical write sequence for writing 16 bits to a register is shown in Figure 20. A start bit given by the master, followed by the write address, starts the sequence. The image sensor then gives 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 gives 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.

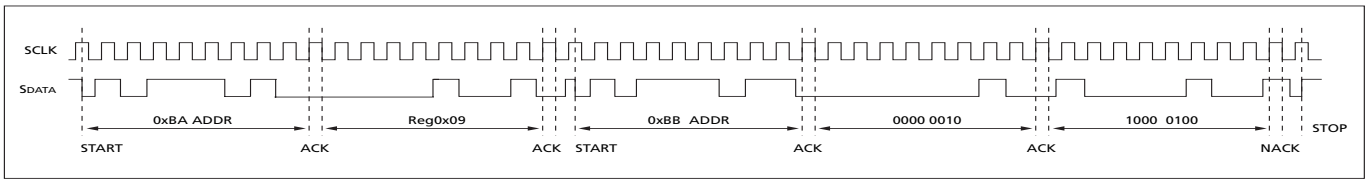
Figure 20: Timing Diagram Showing a Write to Reg0x09 with the Value 0x0284



16-Bit Read Sequence

A typical read sequence is shown in Figure 21. First the master has to write the register address, as in a write sequence. Then a start bit and the read address specifies that a read is about to happen from the register. The master then 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 21: Timing Diagram Showing a Read from Reg0x09; Returned Value 0x0284



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 22: Serial Host Interface Start Condition Timing

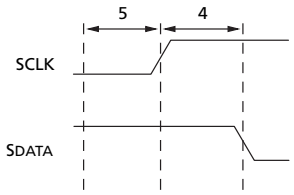
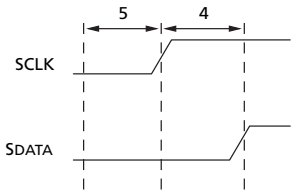
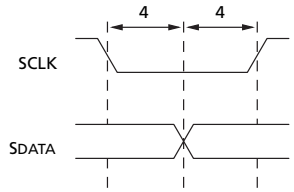


Figure 23: Serial Host Interface Stop Condition Timing



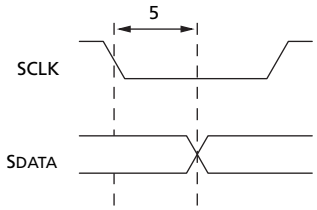
Note: All timing are in units of master clock cycle.

Figure 24: Serial Host Interface Data Timing for Write



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

Figure 25: Serial Host Interface Data Timing for Read



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

Figure 26: Acknowledge Signal Timing After an 8-Bit Write to the Sensor

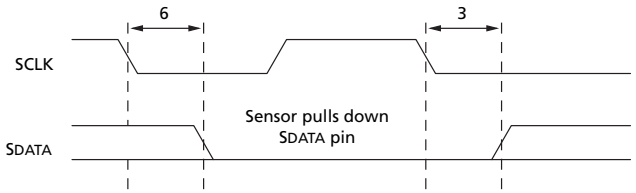
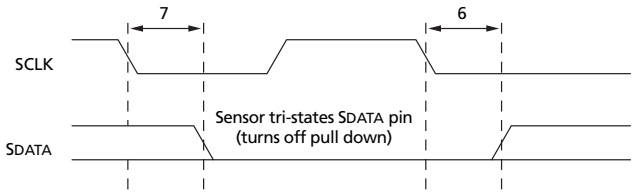
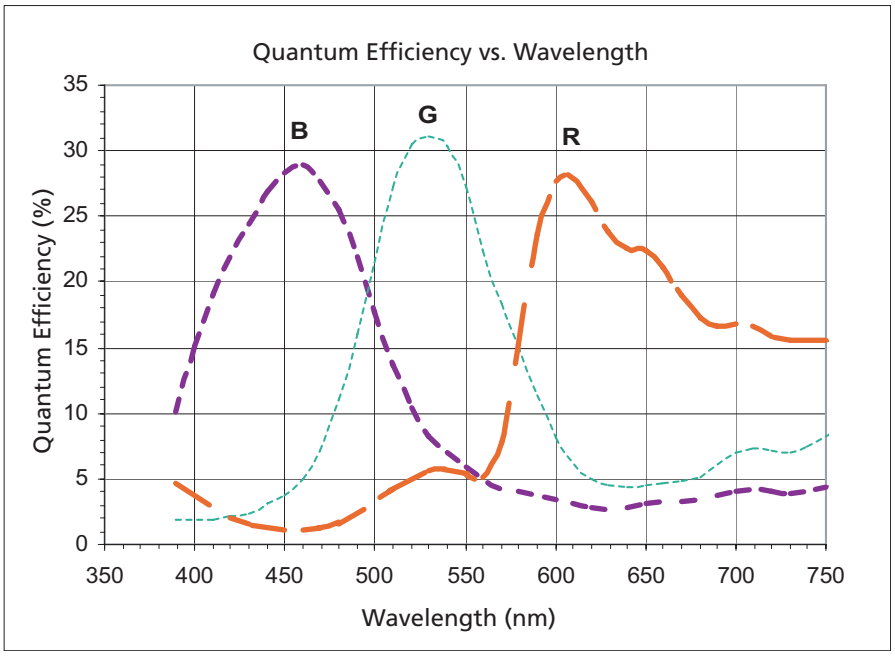


Figure 27: Acknowledge Signal Timing After an 8-Bit Read from the Sensor



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 may be used.

Figure 28: Typical Spectral Characterizations



Electrical Specifications

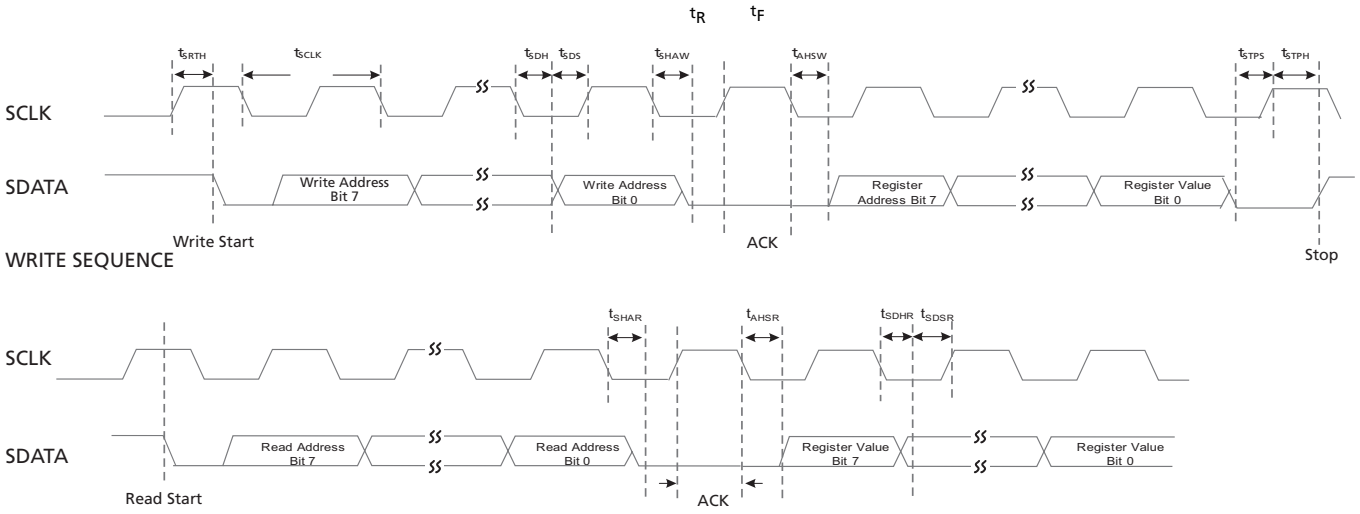
Data Output and Propagation Delays

By default, the MT9P001 launches pixel data, FRAME_VALID and LINE_VALID with the rising edge of PIXCLK. The expectation is that the user captures DOUT[11:0], FRAME_VALID and LINE_VALID using the rising edge of PIXCLK.

Table 15: Two-Wire Serial Bus Characteristics

Symbol	Definition	Condition	Min	Typ	Max	Unit
^{f}SCLK	Serial interface input clock frequency	–	–	–	400	kHz
^{t}SCLK	Serial Input clock period	–	–	–	2.5	µsec
	SCLK Duty Cycle	–	40	50	60	%
^{t}SRTH	Start hold time	WRITE/READ	10*^{t}SCLK	–	–	µs
^{t}SDH	SDATA hold	WRITE	10*^{t}SCLK	–	–	µs
^{t}SDS	SDATA setup	WRITE	10*^{t}SCLK	–	–	µs
^{t}SHAW	SDATA hold to ACK	WRITE	10*^{t}SCLK	–	–	µs
^{t}AHSW	ACK hold to SDATA	WRITE	10*^{t}SCLK	–	–	µs
^{t}STPS	Stop setup time	WRITE/READ	10*^{t}SCLK	–	–	µs
^{t}STPH	Stop hold time	WRITE/READ	10*^{t}SCLK	–	–	µs
^{t}SHAR	SDATA hold to ACK	READ	10*^{t}SCLK	–	–	µs
^{t}AHSR	ACK hold to SDATA	READ	10*^{t}SCLK	–	–	µs
^{t}SDHR	SDATA hold	READ	10*^{t}SCLK	–	–	µs
^{t}SDSR	SDATA setup	READ	10*^{t}SCLK	–	–	µs
CIN_SI	Serial interface input pin capacitance	–	–	3.5	–	pF
CLOAD_SD	SDATA max load capacitance	–	–	15	–	pF
RSD	SDATA pull-up resistor	–	–	1.5	–	kΩ

Figure 29: Two-Wire Serial Bus Timing Parameters



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

Figure 30: I/O Timing Diagram

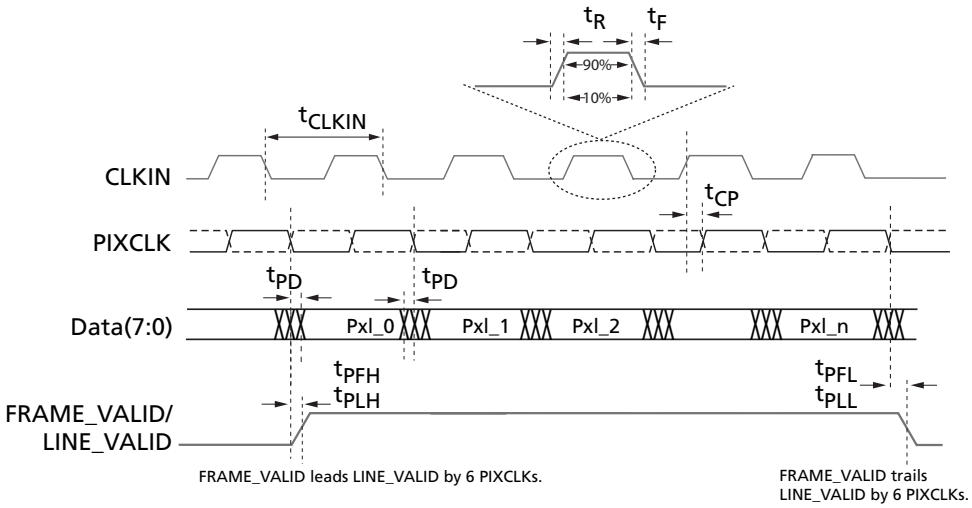


Table 16: I/O Timing Characteristics

Symbol	Definition	Conditions	Min	Typ	Max	Units
^{f}CLKIN1	Input clock frequency	PLL enabled	6	–	27	MHz
^{t}CLKIN1	Input clock period	PLL enabled	166	–	37	ns
^{f}CLKIN2	Input clock frequency	PLL disabled	8	–	96	MHz
^{t}CLKIN2	Input clock period	PLL disabled	125	–	10.4	ns
^{t}R	Input clock rise time		0.03	–	1	V/ns
^{t}F	Input clock fall time		0.03	–	1	V/ns
	Clock duty cycle		40	50	60	%
^{t}JITTER	Input clock jitter	PLL disabled			300	ps
^{t}CP	CLKIN to PIXCLK propagation delay	PLL disabled	10	14.3	20	ns
^{f}PIXCLK	PIXCLK frequency	PLL disabled	8		96	MHz
^{t}PD	PIXCLK to data valid		–	3	2/pixclk	ns
^{t}PFH	PIXCLK to FV HIGH		–	1	2/pixclk	ns
^{t}PLH	PIXCLK to LV HIGH		–	1	2/pixclk	ns
^{t}PFL	PIXCLK to FV LOW		–	1	2/pixclk	ns
^{t}PLL	PIXCLK to LV LOW		–	1	2/pixclk	ns
CLOAD	Output load capacitance		–	6.5	–	pF
CIN	Input pin capacitance		–	2.5	–	pF

Table 17: DC Electrical Characteristics

Symbol	Definition	Conditions	Min	Typ	Max	Units
VDD	Core digital voltage		1.7	1.8	1.9	V
VDDQ	I/O digital voltage		1.7	1.8/2.8	3.1	V
VAA	Analog voltage		2.6	2.8	3.1	V
VAAPIX	Pixel supply voltage		2.6	2.8	3.1	V
VDDPLL	PLL supply voltage		2.6	2.8	3.1	V
			–	–	–	
VIH	Input high voltage	VDDQ=2.8V	2.4	–	VDDQ+0.3	V
		VDDQ=1.8V	1.4	–	VDDQ+0.3	
V_{IL}	Input low voltage	VDDQ=2.8V	GND-0.3	–	0.8	V
		VDDQ=1.8V	GND-0.3	–	0.5	
IIN	Input leakage current	No pull-up resistor; VIN = VDDQ or DGND	-10	+/-0.5	10	uA
VOH	Output high voltage	At specified IOH	VDDQ-0.4	–	–	V
VOL	Output low voltage	At specified IOL		–	0.4	V
IOH	Output high current	At specified VOH = VDDQ-400mv at 1.7V VDDQ	-7	–		mA
I_{OL}	Output low current	At specified VOL=400mv at 1.7V VDDQ	7	–		mA
IOZ	Tri-state output leakage current	VIN=VDDQ or GND	-10	+/-0.5	10	uA
IDD1	Digital operating current	Parallel Mode 96 MHz Full Resolution Nominal Voltage	–	22	40	mA
IDDQ1	I/O digital operating current	Parallel Mode 96 MHz Full Resolution Nominal Voltage	–	12	75	mA
IAA1	Analog operating current	Parallel Mode 96 MHz Full Resolution Nominal Voltage	–	85	105	mA
IAAPIX1	Pixel supply current	Parallel Mode 96 MHz Full Resolution Nominal Voltage	–	2	6	mA
IDDPLL1	PLL supply current	Parallel Mode 96 MHz Full Resolution Nominal Voltage	–	0.16	1	mA
IDD2	Digital operating current	Parallel Mode 96 MHz 4x Binning Nominal Voltage	–	10	40	mA
IDDQ2	I/O digital operating current	Parallel Mode 96 MHz 4x Binning Nominal Voltage	–	13	75	mA
IAA2	Analog operating current	Parallel Mode 96 MHz 4x Binning Nominal Voltage	–	80	105	mA
IAAPIX2	Pixel supply current	Parallel Mode 96 MHz 4x Binning Nominal Voltage	–	3	6	mA
IDDPLL2	PLL supply current	Parallel Mode 96 MHz 4x Binning Nominal Voltage	–	0.16	1	mA
ISTBY1	Hard standby current PLL enabled	CLKIN Enabled	–	TBD	–	uA
ISTBY2	Hard standby current PLL disabled	CLKIN Disabled	–	TBD	–	uA
ISTBY3	Soft standby current PLL enabled	CLKIN Enabled (PLL Enabled)	–	TBD	–	uA
ISTBY4	Soft standby current PLL disabled	CLKIN Enabled (PLL Disabled)	–	TBD	–	uA

Table 18: Power Consumption

Mode	Full Resolution (12 fps)	4x Binning	Units
Streaming	317	287	mW
Standby	TBD	TBD	uW

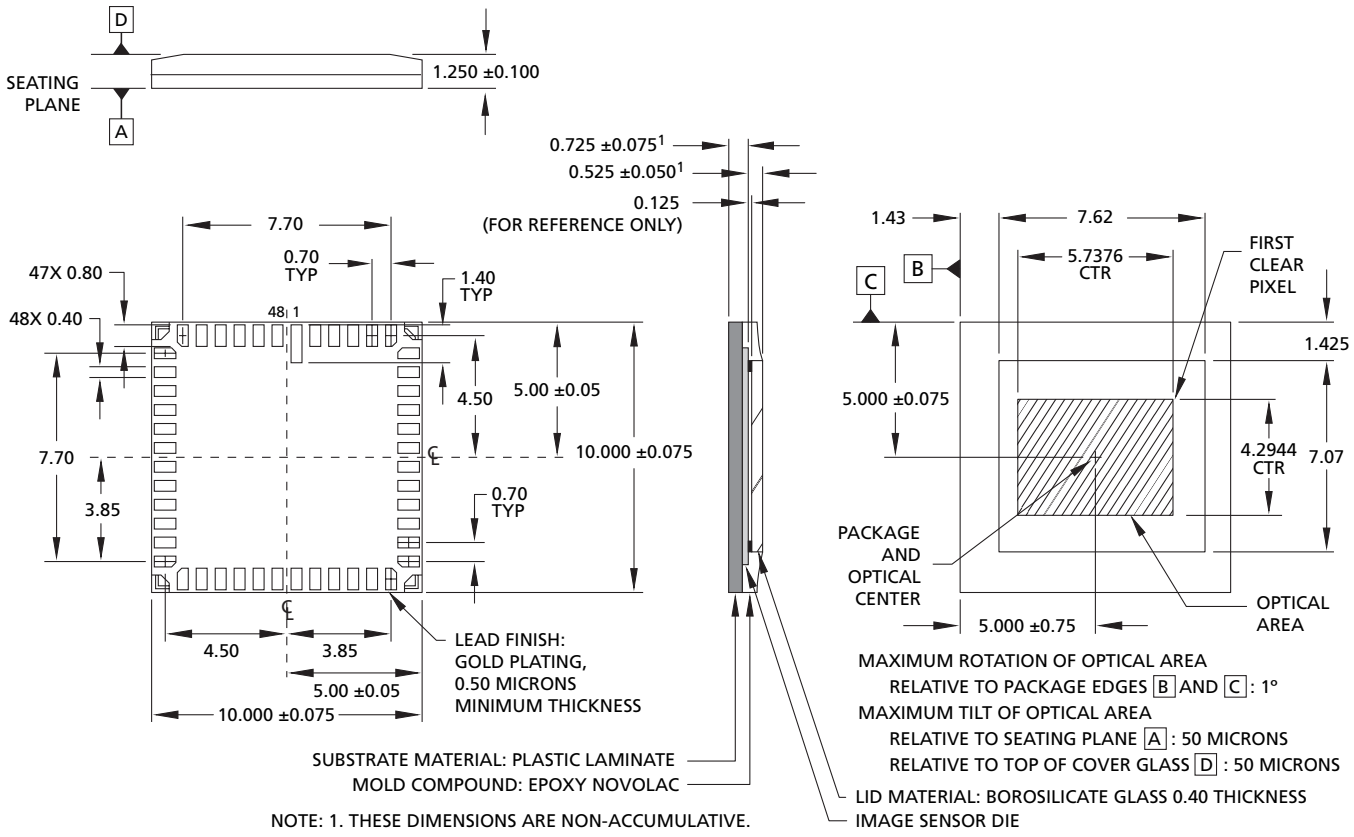
Table 19: Absolute Maximum Ratings

Symbol	Definition	Conditions	Min	Max	Units
V_{DD}_MAX	Core digital voltage		-0.3	1.9	V
V_{DDQ}_MAX	I/O digital voltage		-0.3	3.1	V
V_{AA}_MAX	Analog voltage		-0.3	3.1	V
V_{AAPIX}_MAX	Pixel supply voltage		-0.3	3.1	V
V_{DDPLL}_MAX	PLL supply voltage		-0.3	3.1	V
V_{IN}_MAX	Input voltage		-0.3	V_{DDQ} + 0.3	V
I_{DD}_MAX	Digital operating current		–	TBD	mA
I_{DDQ}_MAX	I/O digital operating current		–	TBD	mA
I_{AA}_MAX	Analog operating current		–	TBD	mA
I_{AAPIX}_MAX	Pixel supply current		–	TBD	mA
I_{DDPLL}_MAX	PLL supply current		–	TBD	mA
T_{OP}	Operating temperature	Measure at junction	-30	70	°C
T_{ST}[1]	Storage temperature		-40	125	°C

Notes: 1. Stresses greater than those listed may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

Package Dimensions

Figure 31: 48-Pin iLCC Package Outline Drawing



Note: All dimensions in millimeters.

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

Revision History

Rev. C, Advance 9/05

- Changed 10-bit to 12-bit
- Update Table 1, Key Performance Parameters, on page 1
- Add Table 2, Available Part Numbers, on page 1
- Update Figure 1, Block Diagram, on page 6
- Update Figure 2, Typical Configuration (Connection), on page 7
- Update Figure 3, 48-Pin iLCC 10x10 Package Pinout Diagram (Top View), on page 8
- Update Table 3, Pin Description, on page 8
- Update "Pixel Array Structure" on page 10
- Update Table 5, Pixel Type by Row, on page 10
- Update Figure 4, Pixel Array Description, on page 11
- Update "Standby and Chip Enable" on page 14
- Update Table 6, Register List and Default Values, on page 15
- Update Table 7, Register Description, on page 16
- Update "Readout Sequence" on page 25
- Update Figure 10, Bin 4x, on page 28
- Update "Mirror" on page 29
- Remove Column Mirror Mode with Binning Figure
- Update Figure 11, Row Time, on page 29
- Update "Frame Time" on page 29
- Update Figure 12, Row Time with Row Bin 2x, on page 30
- Update "LINE_VALID and FRAME_VALID" on page 31
- Update "Restart" on page 33
- Update "Exposure" on page 33
- Update "Operating Modes" on page 34
- Update "Gain" on page 39
- Update Table 14, Gain Increment Settings, on page 39
- Add Figure 28, Typical Spectral Characterizations, on page 45
- Add Table 15, Two-Wire Serial Bus Characteristics, on page 46
- Add Table 16, I/O Timing Characteristics, on page 48
- Update Table 17, DC Electrical Characteristics, on page 49
- Add Table 18, Power Consumption, on page 50
- Update Table 19, Absolute Maximum Ratings, on page 50

Rev. B, Advance 7/05

- Change 12-bit ADC and DOUT(11:0) to 10-bit ADC and DOUT(9:0)
- Added Figure 2: Typical Configuration (Connection) on page 7
- Added Figure 3: 48-Pin iLCC 10x10 Package Pinout Diagram (Top View) on page 8
- Added TEST, RSVD and NC to Table 3: Pin Description on page 8
- Updated "Pixel Data Format" on page 10
- Updated section "Pixel Array Structure" on page 10
- Updated section "Output Data Timing" on page 12
- Added section "Power Up and Power Down" on page 13
- Added section "Reset" on page 13; "Clocks" on page 13
- Added section "Standby and Chip Enable" on page 14
- Added section "Registers" on page 15
- Added Table 6: Register List and Default Values on page 15
- Added Table 7: Register Description on page 16
- Added section "Restart" on page 33
- Added section “Analog Black Level Calibration” on page 40
- Added section “Digital Black Level Calibration” on page 40

- Added Table 17: Absolute Maximum Ratings

Rev. A, Advance 4/05

- Initial release