1/6-Inch VGA System-On-A-Chip (SOC) CMOS Digital Image Sensor

MT9V117 Data Sheet

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

Features

- System-on-a-chip (SOC)—Completely integrated camera system
- Ultra low-power, low-cost CMOS image sensor
- Superior low-light performance
- Electronic rolling shutter (ERS)
- Up to 60 fps progressive scan for high-quality video at VGA resolution
- On-die image flow processor (IFP) performs sophisticated processing: color recovery and correction, sharpening, gamma, lens shading correction, on-the-fly defect correction, zoom
- Image decimation to arbitrary size with smooth, continuous zoom and pan
- Global hue control
- Image flip and mirror
- Parallel interface
- Automatic exposure, white balance and black level compensation, color saturation, and defect identification and correction, aperture correction
- Two-wire serial programming interface
- Progressive ITU-R BT.656 (YCbCr), YUV, 565RGB, 555RGB, 444RGB, RAW10, or processed 8 + 2 output data formats

Applications

- Tethered PC cameras
- Embedded notebook, netbook, and desktop monitor cameras
- Game consoles
- Consumer video communications and security cameras

Table 1: Key Performance Parameters

Parameter		Value
Optical format		1/6-inch (4:3)
Full resolution		640 x 480 pixels (VGA)
Pixel size		3.6 x 3.6µm
Pixel dynamic range		82dB
SNR MAX		42.6dB
Responsivity		5.62 V/lux-sec
Chief ray angle		26° MAX at 100% image height
Color filter array		RGB Bayer pattern
Active pixel array area		2.3mm x 1.73mm
Shutter type		Electronic rolling shutter (ERS)
Input clock frequency		6–54 MHz
Maximum frame rate		60 fps at full resolution
Maximum pixel data output		27 Mp/s
Maximum pixel clock frequency		54 MHz
Supply voltage	Analog	2.50–3.10V
	Digital	1.70–1.95V
	I/O	1.70–1.95V or 2.50–3.10V
ADC resolution		10-bit, on-die
Typical power consumption[1]		120mW, operating mode, typical voltages, and EXTCLK = 27 Mhz
Operating temperature		–30°C to +70°C (at junction)

Note 1: Excluding I/O current

Ordering Information

Table 2: Available Part Numbers

Part Number	Description
MT9V117D00STCK22CC1	Die, MP
MT9V117PACSTCH	Headboard
MT9V117PACSTCD	Demo kit
MT9V117EBJSTC5	CSP, MP

Aptina reserves the right to change products or specifications without notice.

Table of Contents

List of Figures

List of Tables

Functional Description

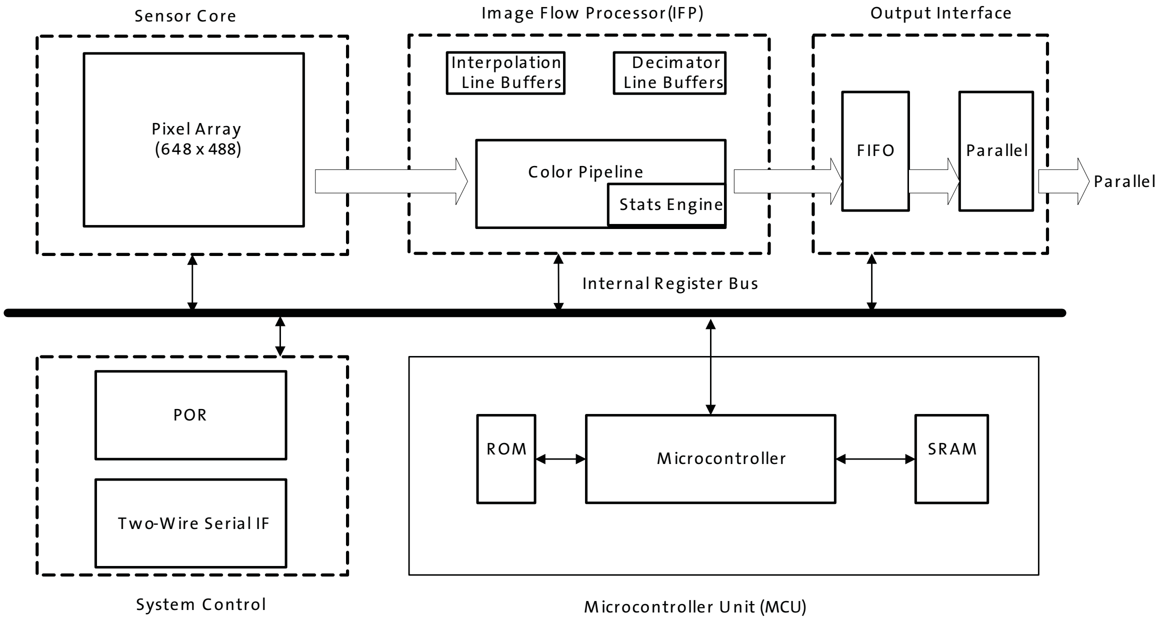
Aptina's MT9V117 is a 1/6-inch VGA CMOS digital image sensor with an integrated advanced camera system. This camera system features a microcontroller (MCU), a sophisticated image flow processor (IFP), and parallel output port. The microcontroller manages all functions of the camera system and sets key operation parameters for the sensor core to optimize the quality of raw image data entering the IFP. The sensor core consists of an active pixel array of 640 x 480 pixels with programmable timing and control circuitry. It also includes an analog signal chain with automatic offset correction, programmable gain, and a 10-bit analog-to-digital converter (ADC).

The entire system-on-a-chip (SOC) has superior low-light performance that is particularly suitable for PC camera applications. The MT9V117 features Aptina's breakthrough low-noise CMOS imaging technology that achieves near-CCD image quality (based on signal-to-noise ratio and low-light sensitivity) while maintaining the inherent size, cost, and integration advantages of CMOS.

Architecture Overview

The MT9V117 combines a VGA sensor core with an IFP to form a stand-alone solution for both image acquisition and processing. Both the sensor core and the IFP have internal registers that can be controlled by the user. In normal operation, an integrated microcontroller autonomously controls most aspects of operation. The processed image data is transmitted to the host system through a parallel interface. Figure 1 shows the major functional blocks of the MT9V117.

Figure 1: MT9V117 Block Diagram



Sensor Core

The MT9V117 has a color image sensor with a Bayer color filter arrangement and a VGA active-pixel array with electronic rolling shutter (ERS). The sensor core readout is 10 bits and can be flipped and/or mirrored. The sensor core also supports separate analog and digital gain for all four color channels (R, Gr, Gb, B).

Image Flow Processor (IFP)

The advanced IFP features and flexible programmability of the MT9V117 can enhance and optimize the image sensor performance. Built-in optimization algorithms enable the MT9V117 to operate with factory settings as a fully automatic and highly adaptable system-on-a-chip (SOC) for most camera systems.

These algorithms include black level conditioning, shading correction, defect correction, color interpolation, edge detection, color correction, aperture correction, and image formatting with cropping and scaling.

Microcontroller Unit (MCU)

The MCU communicates with all functional blocks by way of an internal Aptina proprietary bus interface. The MCU firmware configures all the registers in the sensor core and IFP.

System Control

A two-wire serial interface bus enables read and write access to the MT9V117's internal registers and variables. The internal registers control the sensor core, the color pipeline flow, and the output interface. Variables are located in the microcontroller's RAM memory and are used to configure and control the auto-algorithms and camera control functions.

Output Interface

The output interface block can select either raw data or processed data. Image data is provided to the host system on the 8-bit parallel port. The parallel output port provides up to 10-bit data.

The MT9V117 also includes programmable I/O slew rate to minimize EMI.

System Interfaces

Figure 2 on page 9 shows typical MT9V117 device connections. For low-noise operation, the MT9V117 requires separate power supplies for analog and digital sections of the die. Both power supply rails should be decoupled from ground using capacitors as close as possible to the die.

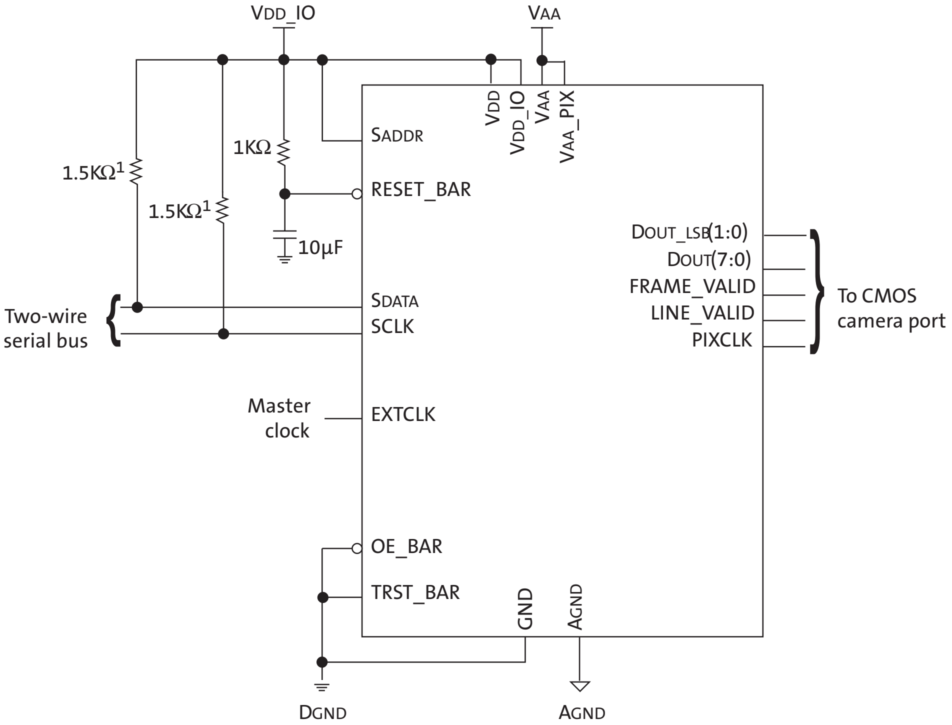
The MT9V117 provides dedicated signals for digital core and I/O power domains that can be at different voltages. The analog circuitry require clean power sources. Table 3 provides the signal descriptions for the MT9V117.

Table 3: Signal Descriptions

Name	Type	Description	Notes
EXTCLK	Input	Input clock signal.	
RESET_BAR	Input/PU	Master reset signal, active LOW. This signal has an internal pull up.	
OE_BAR	Input	Parallel interface enable pad, active LOW.	
SCLK	Input	Two-wire serial interface clock.	
SADDR	Input	Selects device address for the two-wire serial interface.	
SDATA	I/O	Two-wire serial interface data.	
DOUT_LSB[1:0]	I/O	DOUT[1:0] for 10-bit data output mode. For use when in Bypass mode; it is disabled by default.	
FRAME_VALID (FV)	Output	Identifies rows in the active image.	
LINE_VALID (LV)	Output	Identifies pixels in the active line.	
PIXCLK	Output	Pixel clock.	
DOUT[7:0]	Output	DOUT[7:0] for 8-bit image data output or DOUT[9:2] for 10-bit image data output.	
VDD	Supply	Digital power.	
DGND	Supply	Digital ground.	1
VDD_IO	Supply	I/O power supply.	
GND_IO	Supply	I/O ground.	
VAA	Supply	Analog power.	
VAA_PIX	Supply	Pixel array power.	
AGND	Supply	Analog ground.	1
TDI	Input/PU	Internal Test Signal	2
TDO	Output	Internal Test Signal	2
TMS	Input/PU	Internal Test Signal	2
TCK	Input/PU	Internal Test Signal	2
TRST_BAR	Input	Must be tied to GND in normal operation.	
ATEST1	Input	Analog test input.	2
ATEST2	Input	Analog test input.	2

Notes:
1. AGND and DGND are not connected internally.
2. Do not use. These pins can be left floating.

Figure 2: Typical Configuration



Note 1: Aptina recommends a resistor value of 1.5kΩ but a greater value can be used for slower two-wire speed.

Decoupling Capacitor Recommendations

It is important to provide clean, well regulated power to each power supply. The Aptina recommendation for capacitor placement and values are based on our internal demo camera design and verified in hardware.

Note: Because hardware design is influenced by many factors, such as layout, operating conditions, and component selection, the customer is ultimately responsible to ensure that clean power is provided for their own designs.

In order of preference, Aptina recommends:

1. Mount 0.1µF and 1µF decoupling capacitors for each power supply as close as possible to the pad and place a 10 µF capacitor nearby off-module.
2. If module limitations allow for only six decoupling capacitors for a three-regulator design use a 0.1µF and 1µF capacitor for each of the three regulated supplies. Aptina also recommends placing a 10µF capacitor for each supply off-module, but close to each supply.
3. If module limitations allow for only three decoupling capacitors, use a 1µF capacitor (preferred) or a 0.1µF capacitor for each of the three regulated supplies. Aptina recommends placing a 10µF capacitor for each supply off-module but close to each supply.
4. Give priority to the VAA supply for additional decoupling capacitors.
5. Inductive filtering components are not recommended.
6. Follow best practices when performing physical layout. Refer to technical note TN-09-131.

Power-Up Sequence

Powering up the sensor requires voltages to be applied in a particular order, as seen in Figure 3. The timing requirements are shown in Table 4. The sensor includes a power-on reset feature that initiates a reset upon power up of the sensor.

Figure 3: Power-Up Sequence

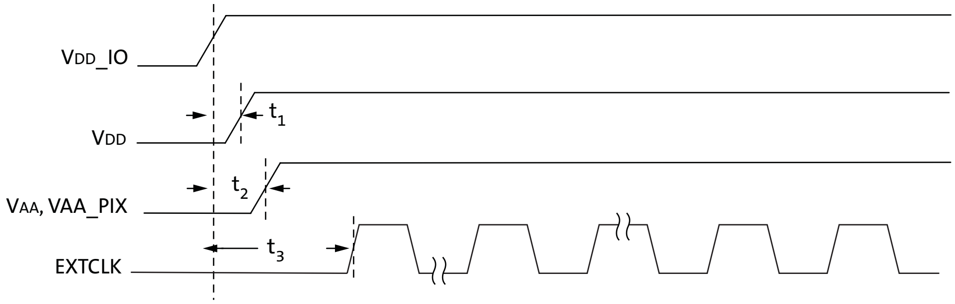


Table 4: Power-Up Signal Timing

Symbol	Parameter	Min	Typ	Max	Unit
t_1	Delay from VDD_IO to VDD	0	–	50	ms
t_2	Delay from VDD_IO to VAA and VAA_PIX	0	–	50	ms
t_3	EXTCLK activation	t_2 + 0	–	–	ms

Power-On Reset

The MT9V117 includes a power-on reset feature that initiates a reset upon power-up.

Three types of reset are available:

- A hard reset is issued by toggling the RESET_BAR signal
- A soft reset is issued by writing commands through the two-wire serial interface
- An internal power-on reset

The output states after hard reset are shown in Table 5.

A soft reset sequence to the sensor has the same effect as the hard reset and can be activated by writing to a register through the two-wire serial interface. On-chip power-on-reset circuitry can generate an internal reset signal in case an external reset is not provided. The RESET_BAR signal has an internal pull-up resistor and can be left floating.

Table 5: Status of Output Signals During Hard Reset

Signal	Reset
DOUT[7:0]	High-Z
PIXCLK	High-Z
LV	High-Z
FV	High-Z
DOUT_LSB[1:0]	High-Z

Hard Reset

The MT9V117 enters the reset state when the external RESET_BAR signal is asserted LOW, as shown in Figure 4. All the output signals will be in High-Z state. When OE_BAR are in HIGH state, the outputs pins will be High-Z during the internal boot time

Figure 4: Hard Reset Operation

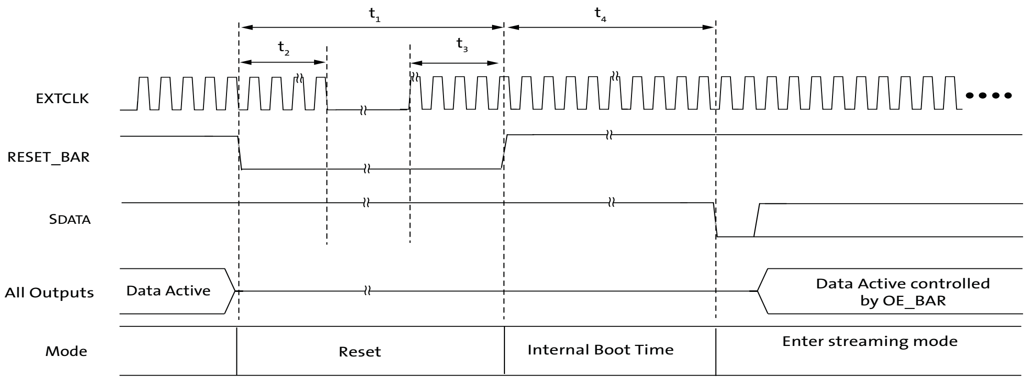


Table 6: Hard Reset

Symbol	Definition	Min	Typ	Max	Unit
t_1	RESET_BAR pulse width	50	–	–	EXTCLK cycles
t_2	Active EXTCLK required after RESET_BAR asserted	10	–	–	
t_3	Active EXTCLK required before RESET_BAR de-asserted	10	–	–	
t_4	Maximum internal boot time[1]	3.5	–	35	ms

Note 1: This delay is dependent on EXTCLK frequency. Minimum equates to 54 MHz and maximum equates to 6 MHz.

Soft Reset

The host processor can reset the MT9V117 using the two-wire serial interface by writing to SYSCTL 0x001A. SYSCTL 0x001A[0] is used to reset the MT9V117 which is similar to external RESET_BAR signal.

1. Set SYSCTL 0x001A[0] to 0x1 to initiate internal reset cycle.
2. Reset SYSCTL 0x001A[0] to 0x0 for normal operation.
3. Delay 3.5 – 35 ms, depending on EXTCLK frequency.

Figure 5: Soft Reset Operation

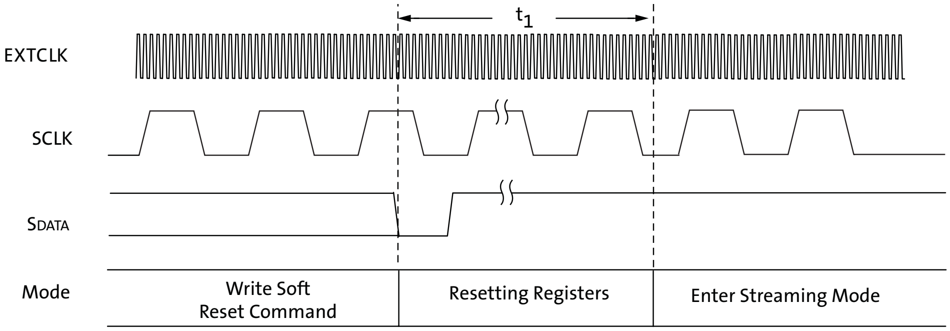


Table 7: Soft Reset Signal Timing

Symbol	Parameter	Min	Typ	Max	Unit
t_1	Maximum soft reset time[1]	3.5	–	35	ms

Note 1: This delay is dependent on EXTCLK frequency. Minimum = 54 MHz; maximum = 6 MHz.

Image Data Output Interface

The user can output the 8-bit parallel image data to host system.

The MT9V117 has an output FIFO to retain a constant pixel output clock independent from the data output rate variations due to scaling factor.

Parallel Port

The MT9V117 image data is read out in a progressive scan mode. Valid image data is surrounded by horizontal blanking and vertical blanking. The amount of horizontal blanking and vertical blanking are programmable.

MT9V117 output data is synchronized with the PIXCLK output. When LV is HIGH, one pixel value is output on the 8-bit DOUT port every TWO PIXCLK periods as shown in Figure 6. PIXCLK is continuously running, even during the blanking period. PIXCLK phase can be varied by 50 percent, controlled using a register.

Figure 6: Pixel Data Timing Example

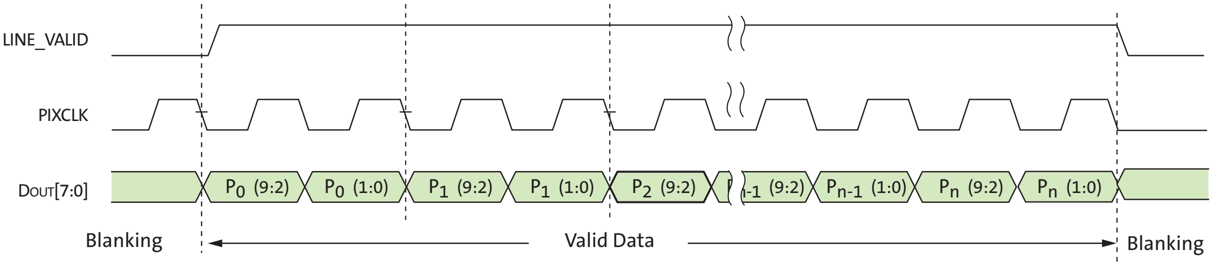
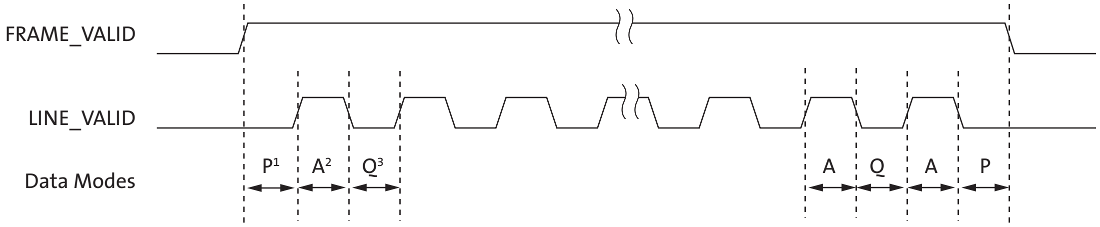


Figure 7: Row Timing, FV, and LV Signals



Notes:
1. P: Frame start and end blanking time.
2. A: Active data time.
3. Q: Horizontal blanking time.

Table 8: Timing Chart

Parameter	Name	Equation	Default @ CLKIN=54 MHz
PIXCLK_PERIOD	Pixel Clock Period	1/fCLKIN	1 pixel clock=18.5ns
A	Active Data Time	2*Ncols*PIXCLK_PERIOD	2*640 pixel clocks=23.7µs
P	Frame Start/End Blanking	6*PIXCLK_PERIOD	6 pixel clocks = 111ns
Q	Horizontal Blanking	(cam_sensor_cfg_line_length_pck*PIXCLK_PERIOD)-A	436 pixel clocks = 8.07µs
A + Q	Row Time	2*cam_sensor_cfg_line_length_pck*PIXCLK_PERIOD	2*858 pixel clocks = 31.77µs
V	Vertical Blanking	F-Tfv	77374 pixel clocks = 1.43ms
Tfv	Frame Valid Time	(Nrows*(A+Q))-Q+(2*P)	823526 pixel clocks = 15.25ms
F	Total Frame Time	2*cam_sensor_cfg_line_length_pck*cam_sensor_cfg_frame_length_lines*PIXCLK_PERIOD	900,900 pixel clocks = 16.68ms

Sensor Control

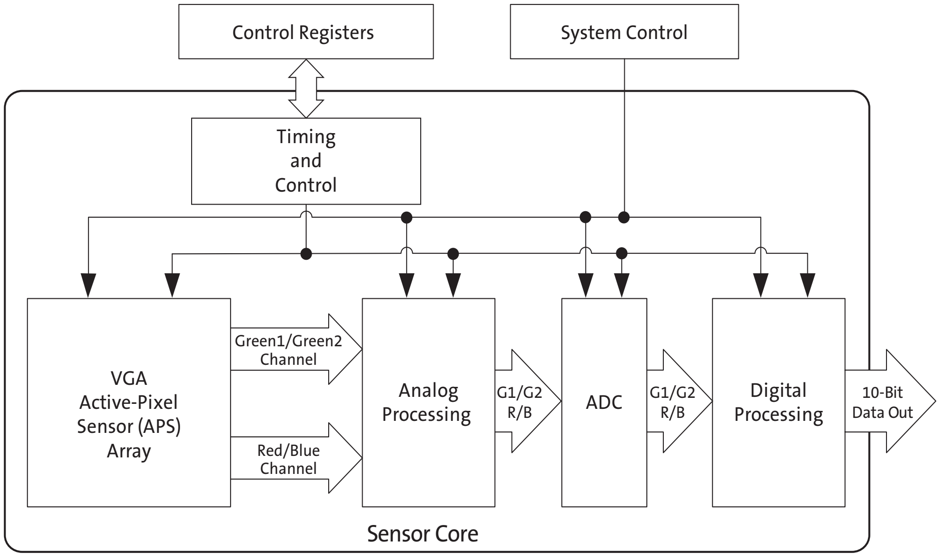
The sensor core of the MT9V117 is a progressive-scan sensor that generates a stream of pixel data at a constant frame rate. Figure 8 shows a block diagram of the sensor core. The timing and control circuitry sequences through the rows of the array, resetting and then reading each row in turn. In the time interval between resetting a row and reading that row, the pixels in the row integrate incident light. The exposure is controlled by varying the time interval between reset and readout. Once a row has been selected, the data from each column is sequenced through an analog signal chain, including offset correction, gain adjustment, and ADC. The final stage of sensor core converts the output of the ADC into 10-bit data for each pixel in the array.

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

The sensor core contains a set of control and status registers that can be used to control many aspects of the sensor behavior including the frame size, exposure, and gain setting. These registers are controlled by the MCU firmware and are also accessible by the host processor through the two-wire serial interface.

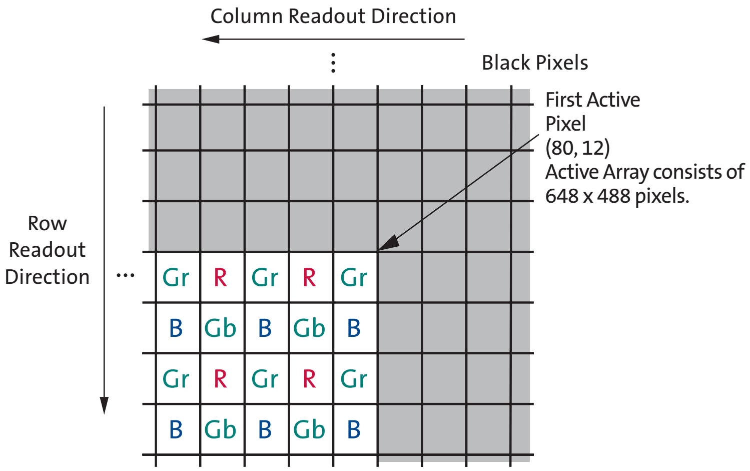
The output from the sensor core is a Bayer pattern; alternate rows are a sequence of either green and red pixels or blue and green pixels. The offset and gain stages of the analog signal chain provide per-color control of the pixel data.

Figure 8: Sensor Core Block Diagram



The sensor core uses a Bayer color pattern, as shown in Figure 9. The even-numbered rows contain green and red pixels; odd-numbered rows contain blue and green pixels. Even-numbered columns contain green and blue pixels; odd-numbered columns contain red and green pixels.

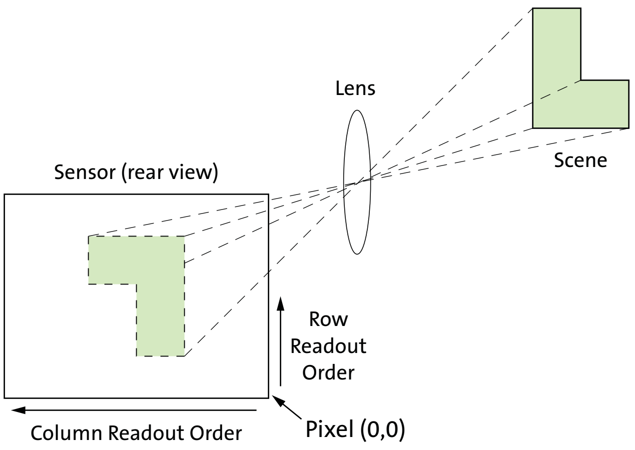
Figure 9: Pixel Color Pattern Detail (Top Right Corner)



The MT9V117 sensor core pixel array is shown with pixel (0,0) in the top right corner, which reflects the actual layout of the array on the die. Figure 10 on page 16 shows the image shown in the sensor during normal operation.

When the image is read out of the sensor, it is read one row at a time, with the rows and columns sequenced.

Figure 10: Imaging a Scene



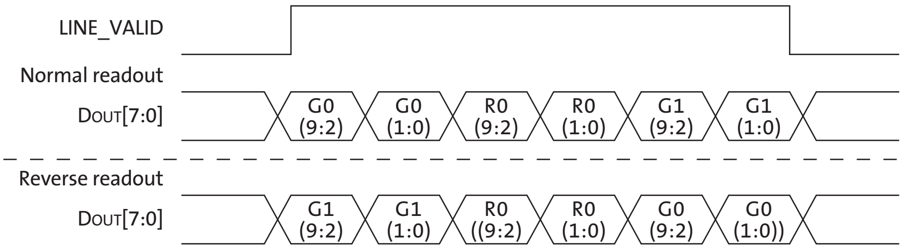
The sensor core supports different readout options to modify the image before it is sent to the IFP. The readout can be limited to a specific window size of the original pixel array.

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

The image output size is set by programming row and column start and end address variables.

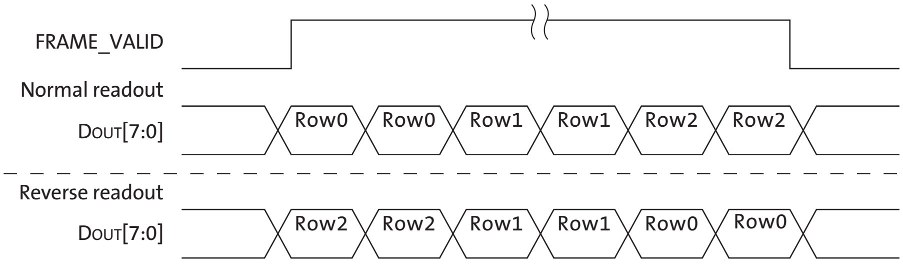
When the sensor is configured to mirror the image horizontally, the order of pixel readout within a row is reversed, so that readout starts from the last column address and ends at the first column address. Figure 11 shows a sequence of 3 pixels being read out with normal readout and reverse readout. This change in sensor core output is corrected by the IFP.

Figure 11: Three Pixels in Normal and Column Mirror Readout Mode



When the sensor is configured to flip the image vertically, the order in which pixel rows are read out is reversed, so that row readout starts from the last row address and ends at the first row address. Figure 12 on page 17 shows a sequence of 3 rows being read out with normal readout and reverse readout. This change in sensor core output is corrected by the IFP.

Figure 12: Three Rows in Normal and Row Mirror Readout Mode



The MT9V117 sensor core supports subsampling with skipping to increase the frame rate. The proper image output size and cropped size must be programmed before enabling subsampling mode. Figure 13 shows the readout with 2X skipping.

Figure 13: Eight Pixels in Normal and Column Skip 2X Readout Mode

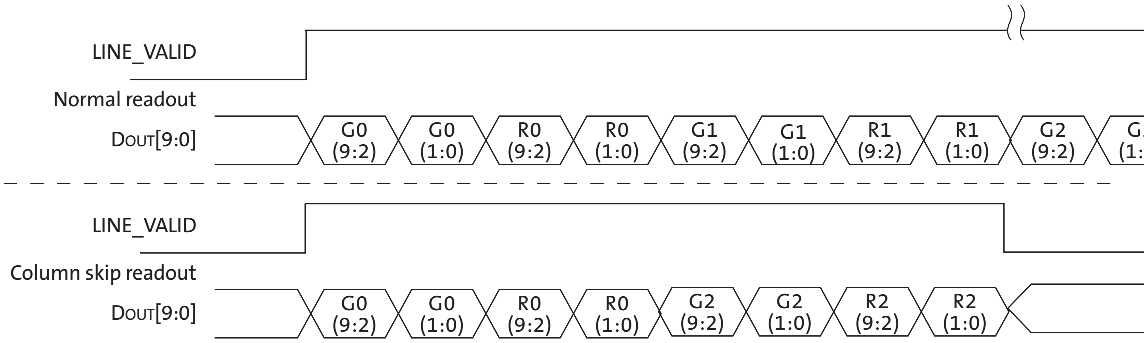


Figure 14 and Figure 15 on page 19 show the different skipping modes supported in MT9V117.

Figure 14: Pixel Readout (no skipping)

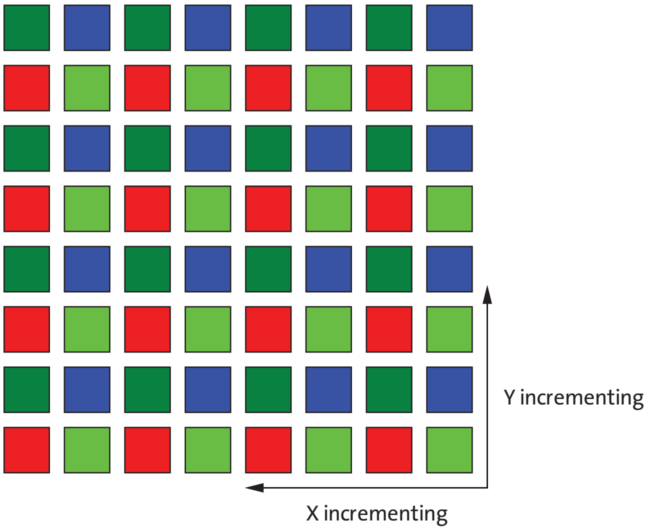


Figure 15: Pixel Readout (y-direction skipping)

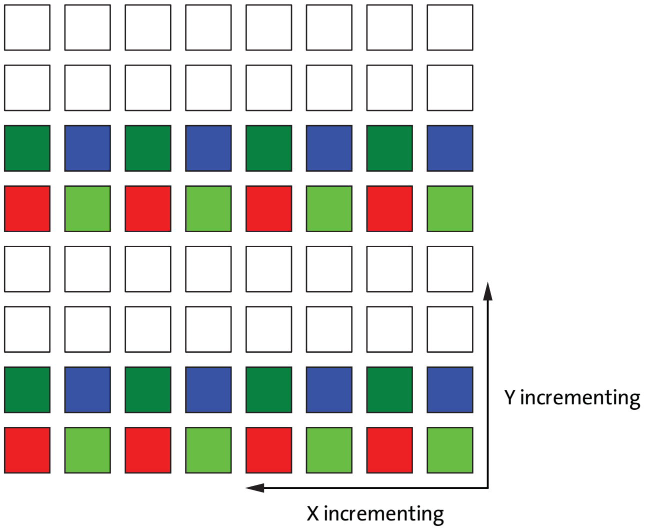
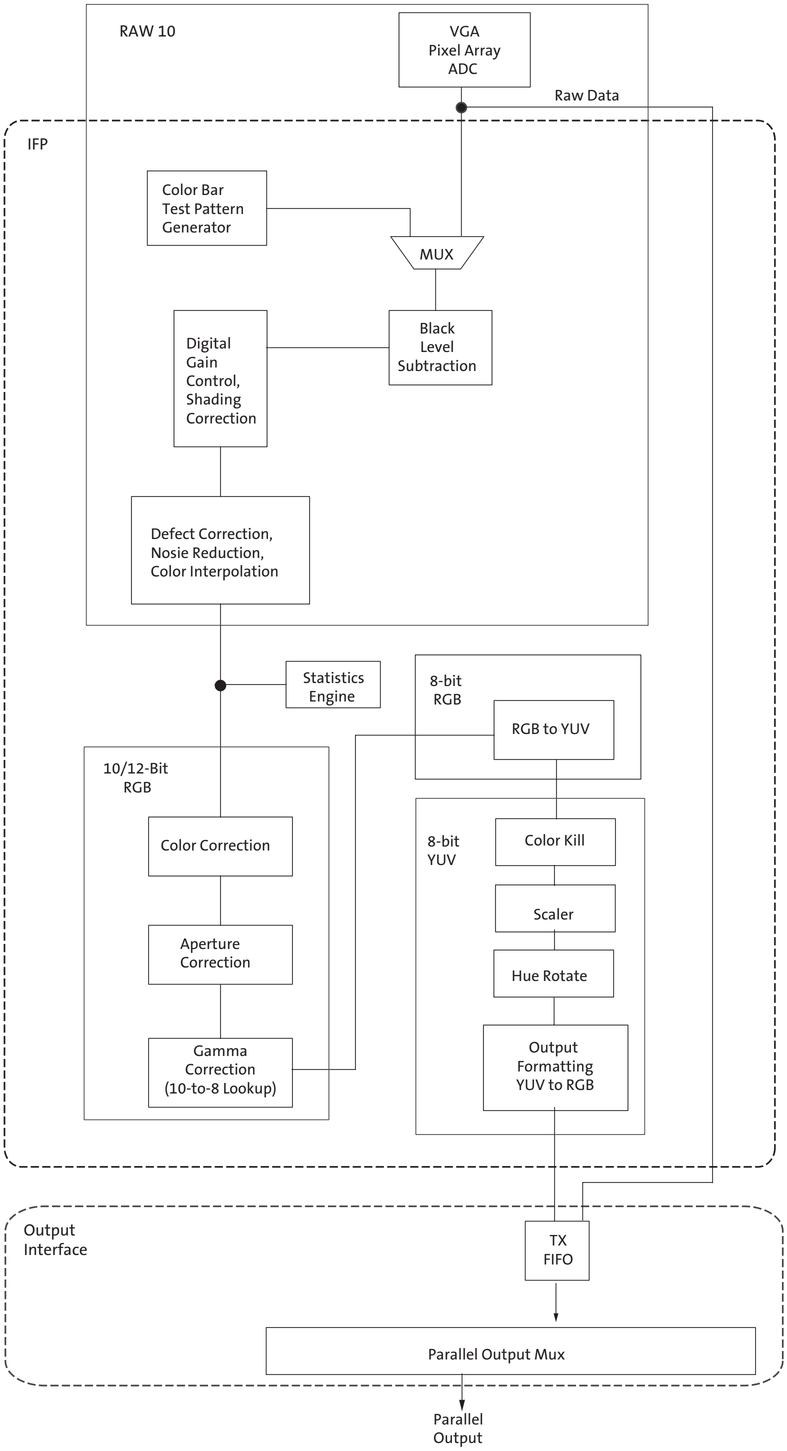


Image Flow Processor

Image control processing in the MT9V117 is implemented in the IFP hardware logic. For normal operation, the microcontroller automatically adjusts the operational parameters of the IFP. Figure 16 shows the image data processing flow within the IFP.

Figure 16: Image Flow Processor



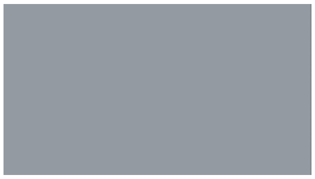
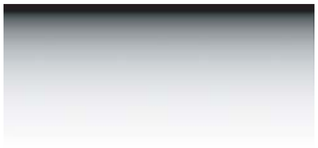
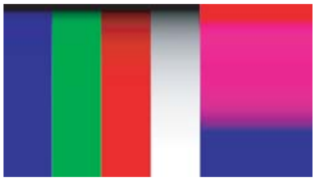
For normal operation of the MT9V117, streams of raw image data from the sensor core are continuously fed into the color pipeline. The MT9V117 features an automatic color bar test pattern generation function to emulate sensor images as shown in Figure 17: "Color Bar Test Pattern," on page 22. The color bar test pattern is fed to the IFP for testing the image pipeline without sensor operation.

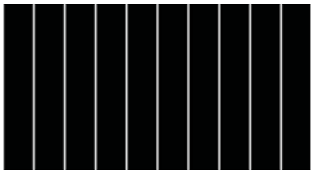
Color bar test pattern generation can be selected by programming registers and variables. The relevant presets selecting test patterns are as follows:

```
[test pattern setup]

VAR=18 ,0x40, 0x0001 //set to test pattern generator
VAR=10 ,0x04, 0x0000 //turn off ae
VAR=11 ,0x04, 0x0000 //turn off awb
LOAD= Config Change
REG= 0x32D4, 0x0080// set digital gain to unity
REG= 0x32D6, 0x0080// set digital gain to unity
REG= 0x32D8, 0x0080// set digital gain to unity
REG= 0x32DA, 0x0080// set digital gain to unity


[Flat field pattern]

LOAD= test pattern setup
VAR=18 ,0x41, 0x0001 //select flat field pattern
LOAD= Refresh


[Vertical Ramp]

LOAD= test pattern setup
VAR=18 ,0x41, 0x0002 //Vertical Ramp
LOAD= Refresh


[Regular Color]

LOAD= test pattern setup
VAR=18 ,0x41, 0x0003 //Regular color
LOAD= Refresh


[Vertical Bars]

LOAD= test pattern setup
VAR=18 ,0x41, 0x0004 //Vertical bars
VAR=18 ,0x42, 0x007F
VAR=18 ,0x44, 0x03FF
LOAD= Refresh


[Random]

LOAD= test pattern setup
VAR=18 ,0x41, 0x0005 //Random
LOAD= Refresh
```

```
[Horizontal Bars]

LOAD= test pattern setup
VAR=18 ,0x41, 0x0006 //Horizontal bars
VAR=18 ,0x42, 0x007F
VAR=18 ,0x44, 0x03FF
LOAD= Refresh


[Exit test pattern]

VAR=18 ,0x40, 0x0000 //Set source to sensor
VAR=10 ,0x04, 0x000F //turn on ae
VAR=11 ,0x04, 0x00BE //turn on awb
LOAD= Refresh
```

Figure 17: Color Bar Test Pattern

Test Pattern	Example
Flat Field VAR = 18, 0x41, 0x0001	
Vertical Ramp VAR = 18, 0x41, 0x0002	
Color Bar VAR = 18, 0x41, 0x0003	

Vertical Bar
VAR = 18, 0x41, 0x0004
VAR = 18, 0x42, 0x007F
VAR = 18, 0x44, 0x03FF
	

Pseudo-Random VAR = 18, 0x41, 0x0005	

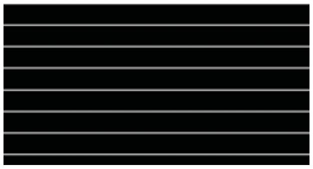
Horizontal Bar
VAR = 18, 0x41, 0x0006
VAR = 18, 0x42, 0x007F
VAR = 18, 0x44, 0x03FF
	

Image Correction

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

Lenses tend to produce images whose brightness is significantly attenuated near the edges. There are also other factors causing fixed pattern signal gradients in images captured by image sensors. The cumulative result of all these factors is known as image shading. The MT9V117 has an embedded shading correction module that can be programmed to counter the shading effects on each individual R, Gb, Gr, and B color signal.

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

Color Interpolation and Edge Detection

In the raw data stream fed by the sensor core to the IFP, each pixel is represented by a 10-bit integer, which can be considered proportional to the pixel’s response to a one-color light stimulus, red, green, or blue, depending on the pixel’s position under the color filter array. Initial data processing steps, up to and including the defect correction, preserve the one-color-per-pixel nature of the data stream, but after the defect correction it must be converted to a three-colors-per-pixel stream appropriate for standard color processing. The conversion is done by an edge-sensitive color interpolation module. The module adds the incomplete color information available for each pixel with information extracted from an appropriate set of neighboring pixels. The algorithm used to select this set and extract the information seeks the best compromise between preserving edges and filtering out high-frequency noise in flat field areas. The edge threshold can be set through variable settings.

Color Correction and Aperture Correction

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

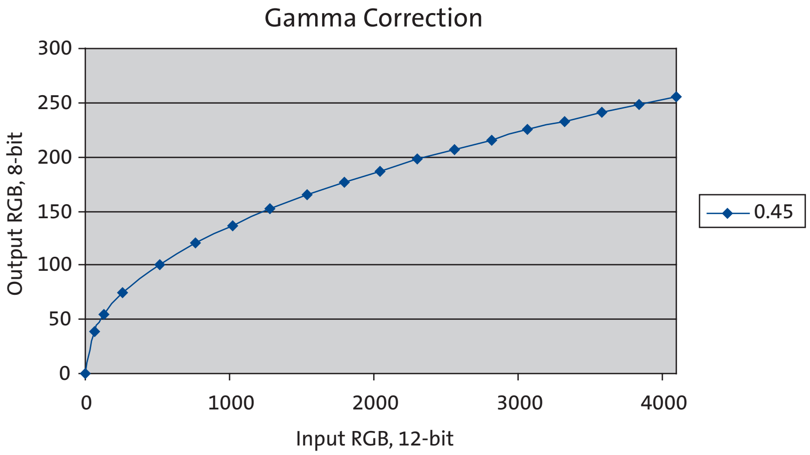
To increase image sharpness, a programmable 2D aperture correction (sharpening filter) is applied to color-corrected image data. The gain and threshold for 2D correction can be defined through variable settings.

Gamma Correction

The gamma correction curve (as shown in Figure 18) is implemented as a piecewise linear function with 19 knee points, taking 12-bit arguments and mapping them to 8-bit output. The abscissas of the knee points are fixed at 0, 64, 128, 256, 512, 768, 1024, 1280, 1536, 1792, 2048, 2304, 2560, 2816, 3072, 3328, 3584, 3840, and 4096. The 8-bit ordinates are programmable through variables.

The MT9V117 IFP includes a block for gamma correction that has the capability to adjust its shape, based on brightness, to enhance the performance under certain lighting conditions. Two custom gamma correction tables may be uploaded, one corresponding to a contrast curve for brighter lighting conditions, the other one corresponding to a noise reduction curve for lower lighting conditions. At power-up, the IFP loads the two tables with default values. The final gamma correction table used depends on the brightness of the scene and can take the form of either uploaded tables or an interpolated version of the two tables. A single (non-adjusting) table for all conditions can also be used.

Figure 18: Gamma Correction Curve



Fade-to-Black

The MT9V117 IFP allows for the image to fade to black under extreme low-light conditions. This feature enables users to optimize the performance of the sensor under low-light conditions. It minimizes the perception of noise and artifacts while the available illumination is diminishing.

This feature has two user set points that reference the brightness of the scene. When the Fade-to-Black starts, it will interpolate to the end point as the light falls until it gets to the end point. When at the end point, the image will be black.

Figure 19: FTB Starting

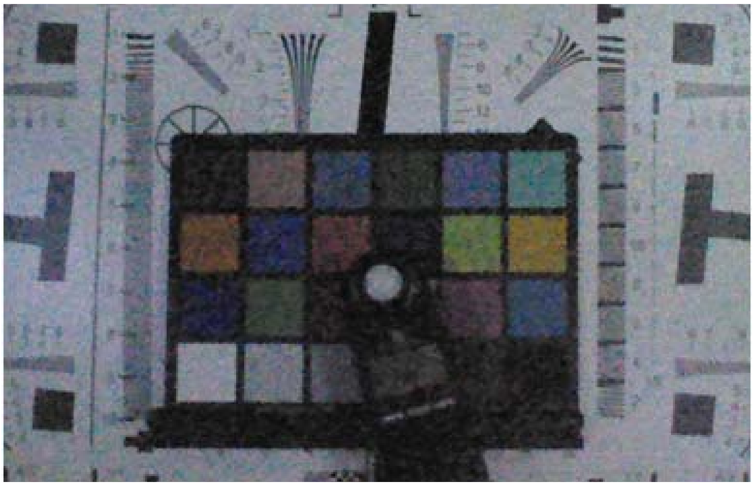


Figure 20: FTB Stop Point Image is Black

Image Scaling and Cropping

To ensure that the size of images output by the MT9V117 can be tailored to the needs of all users, the IFP includes a scaler module. When enabled, this module performs rescaling of incoming images—shrinks them to arbitrarily selected width and height without reducing the field of view and without discarding any pixel values.

By configuring the cropped and output windows to various sizes, different zooming levels for 4X, 2X, and 1X can be achieved. The location of the cropped window is configurable so that panning is also supported. The height and width definitions for the output window must be equal to or smaller than the cropped image. The image cropping and scaler module can be used together to implement a digital zoom and pan.

Hue Rotate

The MT9V117 has integrated hue rotate. This feature will help for improving the color image quality and give customers the flexibility for fine color adjustment and special color effects.

Table 9: Hue Control

Variable	Name	Function
VAR(0x12,0x60)	Enable Hue Rotate	Setting this bit to 1 enables hue rotate
VAR(0x12,0x61)	Hue Angle	Adjusts the global hue angle adjustment (if enabled). 0xEA = –22° 0x00 = 0° 0x16 = +22°

Figure 21: 0° Hue

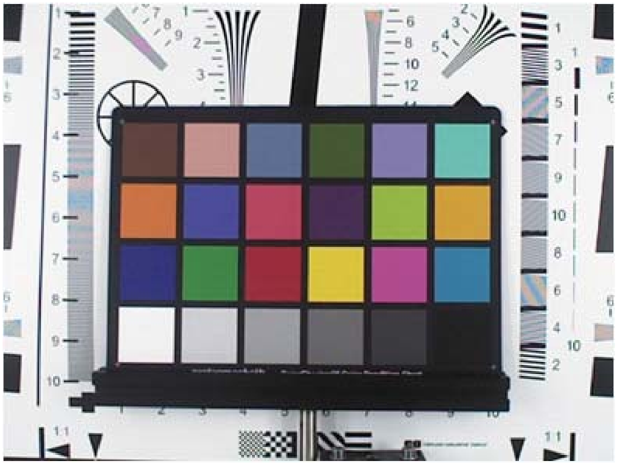


Figure 22: –22° Hue

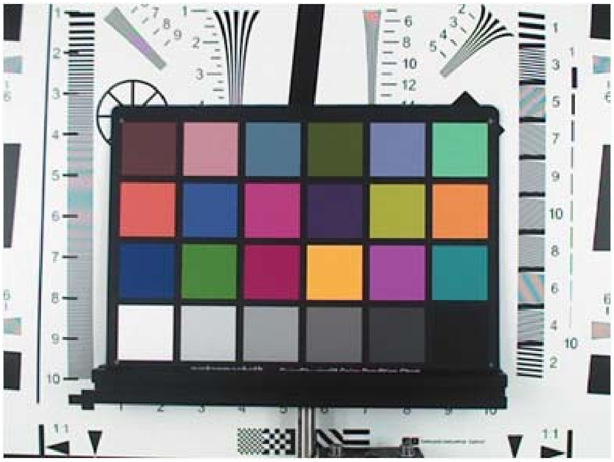
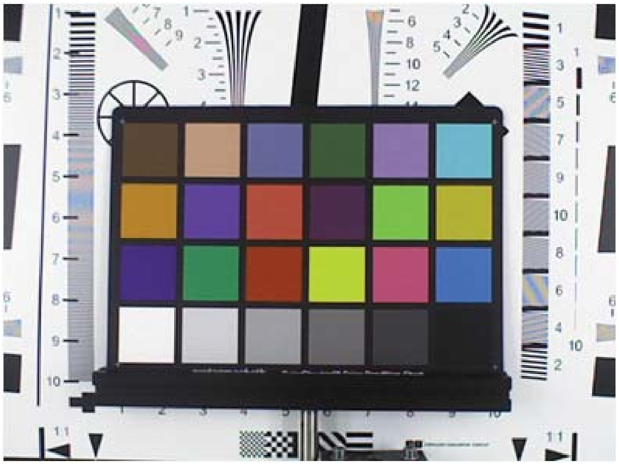


Figure 23: +22° Hue



Camera Control and Auto Functions

Auto Exposure

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

Auto exposure is implemented by a firmware driver that analyzes image statistics collected by the exposure measurement engine, makes a decision, and programs the sensor core and color pipeline to achieve the desired exposure. The measurement engine subdivides the image into 25 windows organized as a 5 x 5 grid.

Two auto exposure algorithm modes are available:

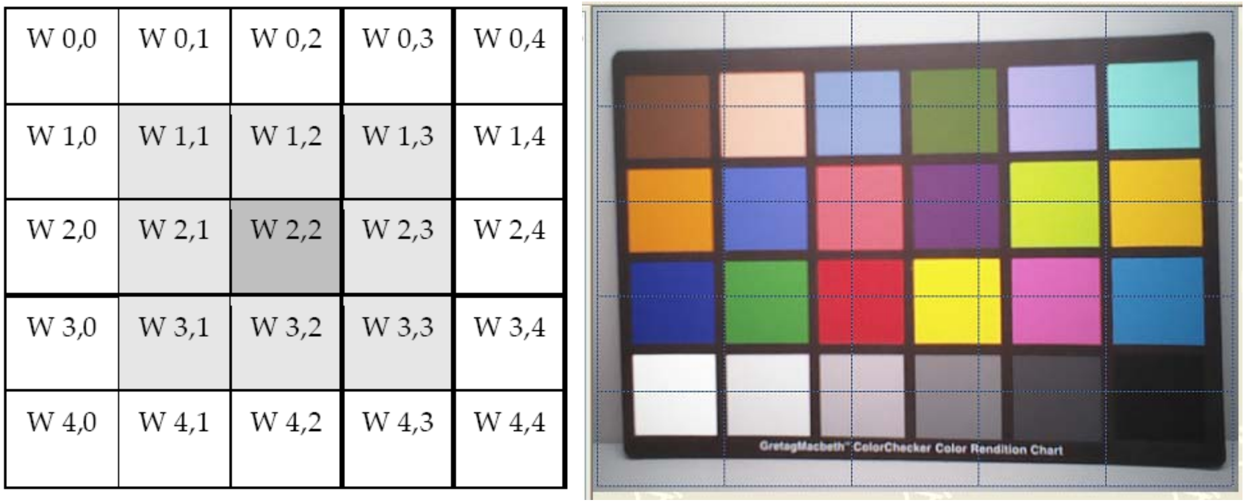
- Average brightness tracking (ABT)
  The average brightness tracking AE uses a constant average tracking algorithm where a target brightness value is compared to a current brightness value, and the gain and integration time are adjusted accordingly to meet the target requirement.
- Weighted Average Brightness
  Each of the 25 windows can be assigned a weight, which can be changed independently of each other. The effect of these weights will allow the center of the image to be weighted higher than the periphery.

AE Driver

Other algorithm features include the rejection of fast fluctuations in illumination (time averaging), control of speed of response, and control of the sensitivity to the small changes. While the default settings are adequate in most situations, the user can program target brightness, measurement window, and other parameters described above.

The driver calculates image brightness based on average luma values received from 25 programmable equal-size rectangular windows forming a 5 x 5grid.

Figure 24: 5 x 5 Grid



W 0,0	W 0,1	W 0,2	W 0,3	W 0,4
W 1,0	W 1,1	W 1,2	W 1,3	W 1,4
W 2,0	W 2,1	W 2,2	W 2,3	W 2,4
W 3,0	W 3,1	W 3,2	W 3,3	W 3,4
W 4,0	W 4,1	W 4,2	W 4,3	W 4,4

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

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

Exposure Control

To achieve the required amount of exposure, the AE driver adjusts the sensor integration time, gains and IFP digital gains. In addition, a variable is available for the user to adjust the overall brightness of the scene. To reject flicker, integration time is typically adjusted in increments of steps. The incremental step specifies the duration in row times equal to one flicker period. Thus, flicker is rejected if integration time is kept a natural factor of the flicker period.

Auto White Balance

The MT9V117 has a built-in AWB algorithm designed to compensate for the effects of changing spectra of the scene illumination on the quality of the color rendition. The algorithm consists of two major parts: a measurement engine performing statistical analysis of the image and a driver performing the selection of the optimal color correction matrix and SOC digital gain. While default settings of these algorithms are adequate in most situations, the user can reprogram base color correction matrices, place limits on color channel gains, and control the speed of both matrix and gain adjustments.

Flicker Avoidance

Flicker occurs when the integration time is not an integer multiple of the period of the light intensity. The MT9V117 can be programmed to avoid flicker for 50 or 60 Hz. For integration times below the light intensity period (10ms for 50Hz environment), flicker cannot be avoided. The MT9V117 supports an indoor AE mode, that will ensure flicker-free operation.

Output Conversion and Formatting

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

Color Conversion Formulas

Y'U'V'

This conversion is BT 601 scaled to make YUV range from 0 through 255. This setting is recommended for JPEG encoding and is the most popular, although it is not well defined and often misused in various operating systems.

$$Y' = 0.299 \times R' + 0.587 \times G' + 0.114 \times B' \qquad \text{(EQ 1)}$$

$$U' = 0.564 \times (B' - Y') + 128 \qquad \text{(EQ 2)}$$

$$V' = 0.713 \times (R' - Y') + 128 \qquad \text{(EQ 3)}$$

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

Y'Cb'Cr' Using sRGB Formulas

The MT9V117 implements the sRGB standard. This option provides YCbCr coefficients for a correct 4:2:2 transmission.

Note: 16 < Y601< 235; 16 < Cb < 240; 16 < Cr < 240; and 0 < = RGB < = 255

$$Y' = (0.2126 \times R' + 0.7152 \times G' + 0.0722 \times B') \times (219 / 256) + 16 \qquad \text{(EQ 4)}$$

$$Cb' = 0.5389 \times (B' - Y') \times (224 / 256) + 128 \qquad \text{(EQ 5)}$$

$$Cr' = 0.635 \times (R' - Y') \times (224 / 256) + 128 \qquad \text{(EQ 6)}$$

Y'U'V' Using sRGB Formulas

These are similar to the previous set of formulas, but have YUV spanning a range of 0 through 255.

$$Y' = 0.2126 \times R' + 0.7152 \times G' + 0.0722 \times B' \qquad \text{(EQ 7)}$$

$$U' = (0.5389 \times (B' - Y') + 128 = -0.1146 \times R' - 0.3854 \times G' + 0.5 \times B' + 128) \qquad \text{(EQ 8)}$$

$$V' = 0.635 \times (R' - Y') + 128 = 0.5 \times R' - 0.4542 \times G' - 0.0458 \times B' + 128 \qquad \text{(EQ 9)}$$

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

$$R' = Y + 1.5748 \times V \qquad \text{(EQ 10)}$$

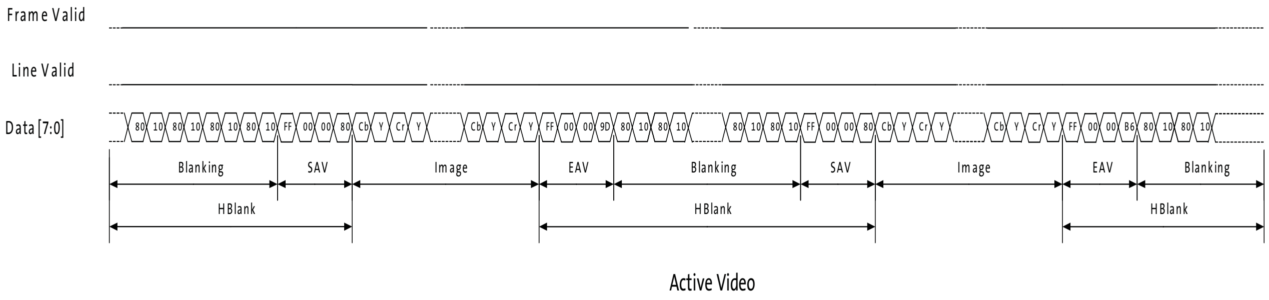
$$G' = Y - 0.1873 \times (U - 128) - 0.4681 \times (V - 128) \qquad \text{(EQ 11)}$$

$$B' = Y + 1.8556 \times (U - 128) \qquad \text{(EQ 12)}$$

BT656

YUV data can also be output in BT656 format with SAV/EAV codes. The BT656 data output will be progressive data and not interlaced (cam_output_format_bt656_enable=1).

Figure 25: BT656 Image Data with SAV/EAV Codes


Frame Valid
Line Valid
Data [7:0]
80 10 80 10 80 10 80 10 FF 00 00 80 Cb Y Cr Y Cb Y Cr Y FF 00 00 9D 80 10 80 10 80 10 80 10 FF 00 00 80 Cb Y Cr Y Cb Y Cr Y FF 00 00 B6 80 10 80 10
Blanking
SAV
Image
EAV
Blanking
SAV
Image
EAV
Blanking
HBlank
HBlank
HBlank
Active Video

Uncompressed YUV/RGB Data Ordering

The MT9V117 supports swapping YCbCr mode, as illustrated in Table 10.

Table 10: YCbCr Output Data Ordering

Mode	Data Sequence			
Default (no swap)	Cb_i	Y_i	Cr_i	Y_{i+1}
Swapped CrCb	Cr_i	Y_i	Cb_i	Y_{i+1}
Swapped YC	Y_i	Cb_i	Y_{i+1}	Cr_i
Swapped CrCb, YC	Y_i	Cr_i	Y_{i+1}	Cb_i

The RGB output data ordering in default mode is shown in Table 11. The odd and even bytes are swapped when luma/chroma swap is enabled. R and B channels are bitwise swapped when chroma swap is enabled.

Table 11: RGB Ordering in Default Mode

Mode (Swap Disabled)	Byte	$D_7D_6D_5D_4D_3D_2D_1D_0$
565RGB	Odd	$R_7R_6R_5R_4R_3G_7G_6G_5$
	Even	$G_4G_3G_2B_7B_6B_5B_4B_3$
555RGB	Odd	$0\ R_7R_6R_5R_4R_3G_7G_6$
	Even	$G_4G_3G_2B_7B_6B_5B_4B_3$
444xRGB	Odd	$R_7R_6R_5R_4G_7G_6G_5G_4$
	Even	$B_7B_6B_5B_4\ 0\,0\,0\,0$
x444RGB	Odd	$0\,0\,0\,0\ R_7R_6R_5R_4$
	Even	$G_7G_6G_5G_4B_7B_6B_5B_4$

Below are the settings required to obtain the range of output formats discussed.

```
[Output Format - YCbCr]

// Assumes full FOV image

FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_FORMAT, 0   // YUV

FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_BT656_ENABLE, 0 // Disable BT.656 codes

// VAR= 18, 0x0058, 0x0010

LOAD= Change-Config

IMAGE= 640, 480, YCbCr


[Output Format - BT.656 with FV/LV]

// Preset assumes full FOV image

FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_FORMAT, 0       // YUV

FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_BT656_ENABLE, 1 // Enable BT.656 codes

FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_FVLV_DISABLE, 0 // Enable FV and LV

// VAR= 18, 0x0058, 0x0018

LOAD= Change-Config
```

```
IMAGE= 640, 480, YCbCr

[Output Format - BT.656 no FV/LV]
// Assumes full FOV image
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_FORMAT, 0       // YUV
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_BT656_ENABLE, 1 // Enable BT.656 codes
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_FVLV_DISABLE, 1 // Disable FV and LV
// VAR= 18, 0x0058, 0x0038
LOAD= Change-Config
IMAGE= 640, 480, YCbCr

[Output Format - RGB565]
// Assumes full FOV image
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_FORMAT, 1      // RGB
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_BT656_ENABLE, 0 // Disable BT.656 codes
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_RGB_FORMAT, 0  // RGB565
// VAR= 18, 0x0058, 0x0130
LOAD= Change-Config
IMAGE= 640, 480, RGB-565

[Output Format - RGB555]
// Assumes full FOV image
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_FORMAT, 1      // RGB
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_BT656_ENABLE, 0 // Disable BT.656 codes
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_RGB_FORMAT, 1  // RGB555
// VAR= 18, 0x0058, 0x1130
LOAD= Change-Config
IMAGE= 640, 480, RGB-555

[Output Format - RGB444x]
// Assumes full FOV image
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_FORMAT, 1      // RGB
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_BT656_ENABLE, 0 // Disable BT.656 codes
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_RGB_FORMAT, 2  // RGB444x
// VAR= 18, 0x0058, 0x2130
LOAD= Change-Config
```

```
IMAGE= 640, 480, RGB-444x


[Output Format - RGBx444]
// Assumes full FOV image
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_FORMAT, 1      // RGB
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_BT656_ENABLE, 0 // Disable BT.656 codes
FIELD_WR= CAM_OUTPUT_FORMAT, CAM_OUTPUT_FORMAT_RGB_FORMAT, 3  // RGBx444
// VAR= 18, 0x0058, 0x3130
LOAD= Change-Config
IMAGE= 640, 480, RGB-x444
```

Uncompressed 10-Bit Bypass Output

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

1. Using both DOUT[7:0] and DOUT_LSB[1:0].
2. Using only DOUT[7:0] with a special 8 + 2 data format, shown in Table 12.

Table 12: 2-Byte Bayer Format

Byte	Bits Used	Bit Sequence
Odd bytes	8 data bits	$D_9D_8D_7D_6D_5D_4D_3D_2$
Even bytes	2 data bits + 6 unused bits	$0\,0\,0\,0\,0\,0\,D_1D_0$

10-Bit Processed Bayer Output

10-bit processed Bayer data from the SOC can be output.

Table 13: 2-Byte Processed Bayer Format

Byte	Bits Used	Bit Sequence
Odd bytes	8 data bits	$D_9D_8D_7D_6D_5D_4D_3D_2$
Even bytes	2 data bits + 6 unused bits	$0\,0\,0\,0\,0\,0\,D_1D_0$

Silicon Revision Identification

It is possible to identify the silicon revision of the MT9V117; this is done by reading R0x31FE.

Two-Wire Serial Interface

The two-wire serial interface bus enables read and write access to control and status registers and variables within the MT9V117.

The interface protocol uses a master/slave model in which a master controls one or more slave devices. The MT9V117 always operates in slave mode. The host (master) generates a clock (SCLK) that is an input to the MT9V117 and is used to synchronize transfers. Data is transferred between the master and the slave on a bidirectional signal (SDATA).

Protocol

Data transfers on the two-wire serial interface bus are performed by a sequence of low-level protocol elements, as follows:

1. a (repeated) start condition
2. a slave address/data direction byte
3. a 16-bit register address (8-bit addresses are not supported)
4. an (a no) acknowledge bit
5. a 16-bit data transfer (8-bit data transfers are not supported)
6. a stop condition

The bus is idle when both SCLK and SDATA are HIGH. Control of the bus is initiated with a start condition, and the bus is released with a stop condition. Only the master can generate the start and stop conditions.

A start condition is defined as a HIGH-to-LOW transition on SDATA while SCLK is HIGH.

At the end of a transfer, the master can generate a start condition without previously generating a stop condition; this is known as a repeated start or restart condition.

A stop condition is defined as a LOW-to-HIGH transition on SDATA while SCLK is HIGH.

Data is transferred serially, 8 bits at a time, with the most significant bit (MSB) transmitted first. Each byte of data is followed by an acknowledge bit or a no-acknowledge bit. This data transfer mechanism is used for the slave address/data direction byte and for message bytes. One data bit is transferred during each SCLK clock period. SDATA can change when SCLK is LOW and must be stable while SCLK is HIGH.

Slave Address

Bits [7:1] of this byte represent the device slave address and bit [0] indicates the data transfer direction. A “0” in bit [0] indicates a WRITE, and a “1” indicates a READ. If the SADDR signal is driven LOW, then addresses used by the MT9V117 are R0x090 (write address) and R0x091 (read address). If the SADDR signal is driven HIGH, then addresses used by the MT9V117 are R0x0BA (write address) and R0x0BB (read address).

Message Byte

Message bytes are used for sending register addresses and register write data to the slave device and for retrieving register read data. The protocol used is outside the scope of the two-wire serial interface specification.

Acknowledge Bit

Each 8-bit data transfer is followed by an acknowledge bit or a no-acknowledge bit in the SCLK clock period following the data transfer. The transmitter (which is the master when writing, or the slave when reading) releases SDATA. The receiver indicates an acknowledge bit by driving SDATA LOW. As for data transfers, SDATA can change when SCLK is LOW and must be stable while SCLK is HIGH.

No-Acknowledge Bit

The no-acknowledge bit is generated when the receiver does not drive SDATA low during the SCLK clock period following a data transfer. A no-acknowledge bit is used to terminate a read sequence.

Stop Condition

A stop condition is defined as a LOW -to-HIGH transition on SDATA while SCLK is HIGH.

Typical Serial Transfer

A typical read or write sequence begins by the master generating a start condition on the bus. After the start condition, the master sends the 8-bit slave address/data direction byte. The last bit indicates whether the request is for a read or a write, where a “0” indicates a write and a “1” indicates a read. If the address matches the address of the slave device, the slave device acknowledges receipt of the address by generating an acknowledge bit on the bus.

If the request was a write, the master then transfers the 16-bit register address to which a write should take place. This transfer takes place as two 8-bit sequences and the slave sends an acknowledge bit after each sequence to indicate that the byte has been received. The master then transfers the data as an 8-bit sequence; the slave sends acknowledge bit at the end of the sequence. After 8 bits have been transferred, the slave’s internal register address is automatically incremented, so that the next 8 bits are written to the next register address. The master stops writing by generating a (re)start or stop condition.

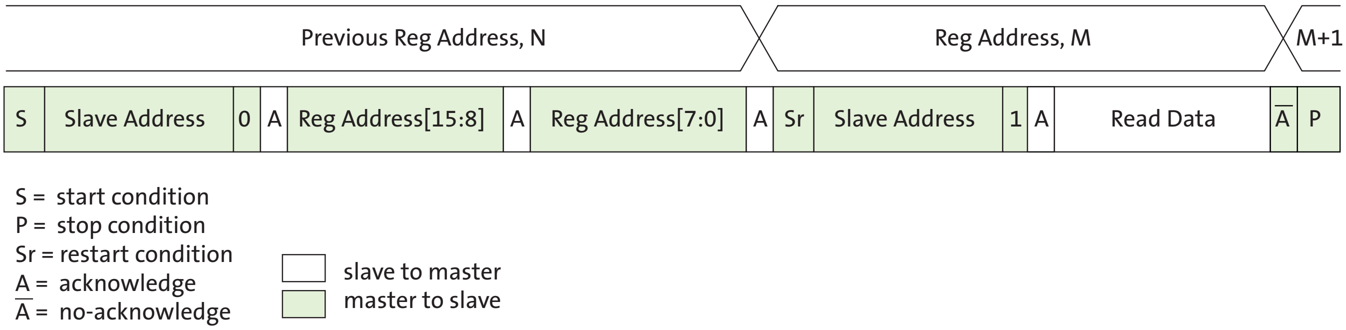
If the request was a read, the master sends the 8-bit write slave address/data direction byte and 16-bit register address, just as in the write request. The master then generates a (re)start condition and the 8-bit read slave address/data direction byte, and clocks out the register data, 8 bits at a time. The master generates an acknowledge bit after each 8-bit transfer. The slave’s internal register address is automatically incremented after every 8 bits are transferred. The data transfer is stopped when the master sends a no-acknowledge bit.

Note: If a customer is using direct memory writes (XDMA), AND the first write ends on an odd address boundary AND the second write starts on an even address boundary AND the first write is not terminated by a STOP, the write data can become corrupted. To avoid this, ensure that a serial write is terminated by a STOP.

Single Read from Random Location

This sequence (see Figure 26) starts with a dummy write to the 16-bit address that is to be used for the read. The master terminates the write by generating a restart condition. The master then sends the 8-bit read slave address/data direction byte and clocks out one byte of register data. The master terminates the read by generating a no-acknowledge bit followed by a stop condition. Figure 26 shows how the internal register address maintained by the MT9V117 is loaded and incremented as the sequence proceeds.

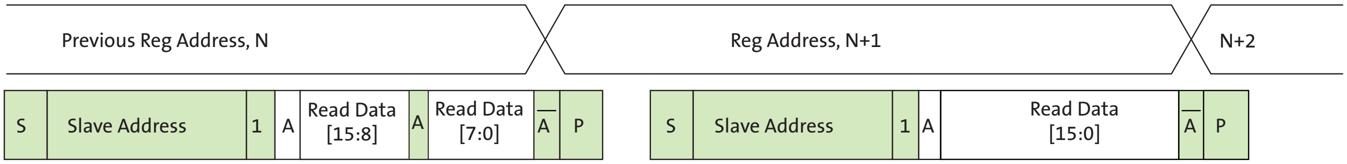
Figure 26: Single Read from Random Location



Single Read from Current Location

This sequence (Figure 27) performs a read using the current value of the MT9V117 internal register address. The master terminates the read by generating a no-acknowledge bit followed by a stop condition. The figure shows two independent read sequences.

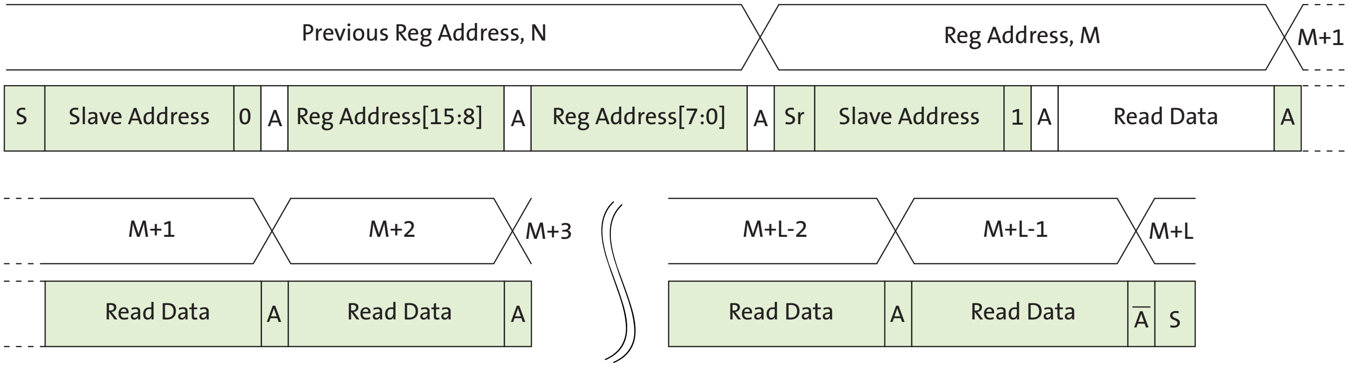
Figure 27: Single Read from Current Location



Sequential Read, Start from Random Location

This sequence (Figure 28) starts in the same way as the single read from random location (Figure 26). Instead of generating a no-acknowledge bit after the first byte of data has been transferred, the master generates an acknowledge bit, and continues to perform byte reads until L bytes have been read.

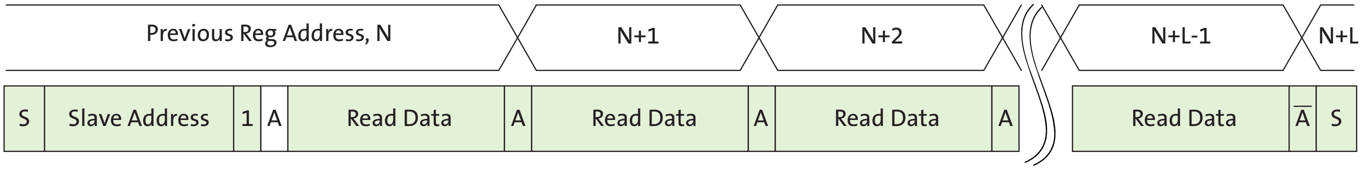
Figure 28: Sequential Read, Start from Random Location



Sequential Read, Start from Current Location

This sequence (Figure 29) starts in the same way as the single read from current location (Figure 27). Instead of generating a no-acknowledge bit after the first byte of data has been transferred, the master generates an acknowledge bit, and continues to perform byte reads until L bytes have been read.

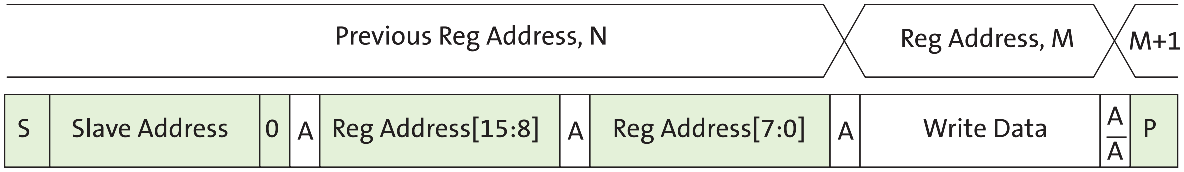
Figure 29: Sequential Read, Start from Current Location



Single Write to Random Location

This sequence (Figure 30) begins with the master generating a start condition. The slave address/data direction byte signals a write and is followed by the high then low bytes of the register address that is to be written. The master follows this with the byte of write data. The write is terminated by the master generating a stop condition.

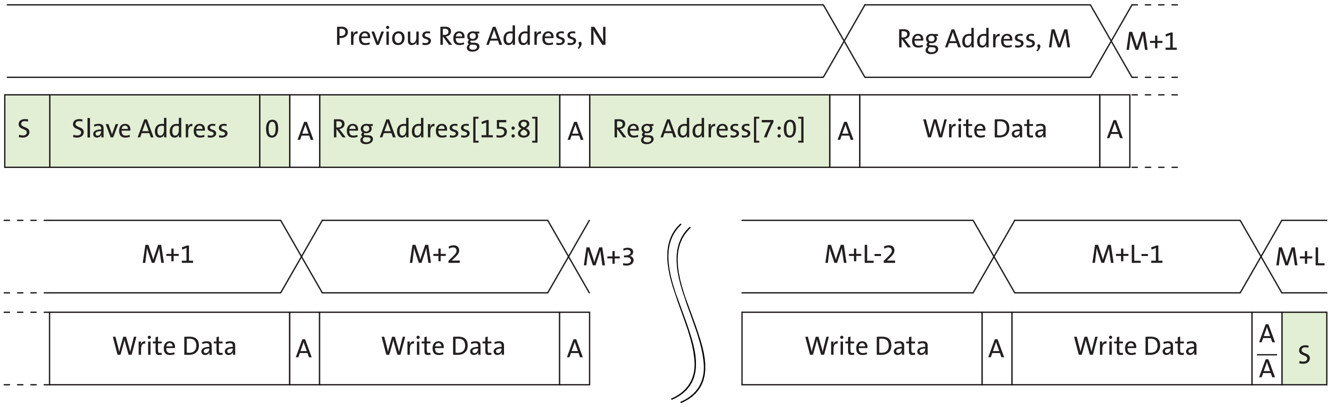
Figure 30: Single Write to Random Location



Sequential Write, Start at Random Location

This sequence (Figure 31) starts in the same way as the single write to random location (Figure 30). Instead of generating a no-acknowledge bit after the first byte of data has been transferred, the master generates an acknowledge bit, and continues to perform byte writes until L bytes have been written. The write is terminated by the master generating a stop condition.

Figure 31: Sequential Write, Start at Random Location



Spectral Characteristic

Figure 32: Chief Ray Angle (CRA) vs. Image Height

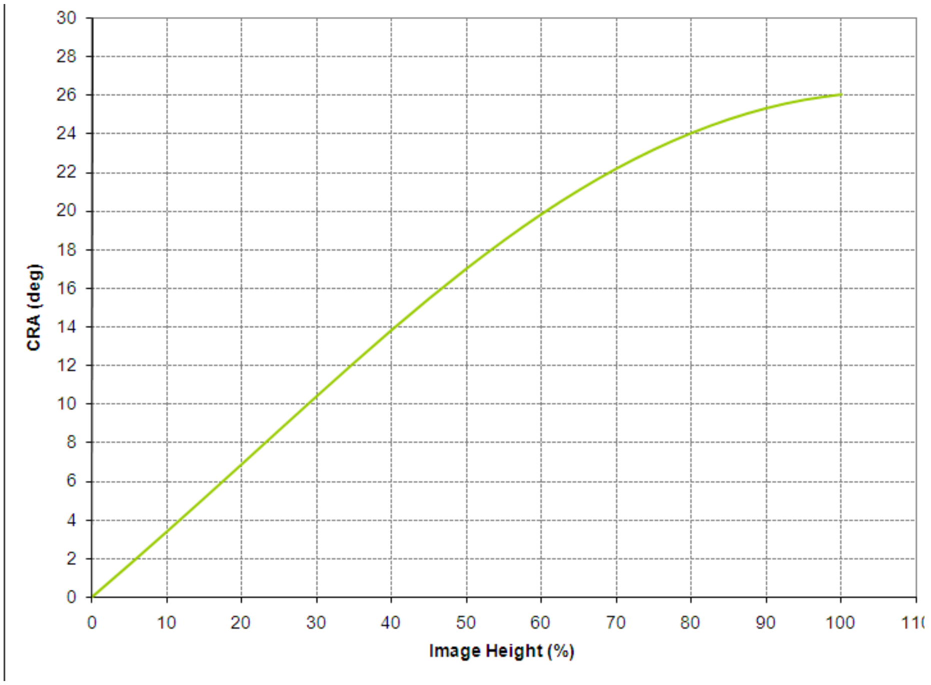
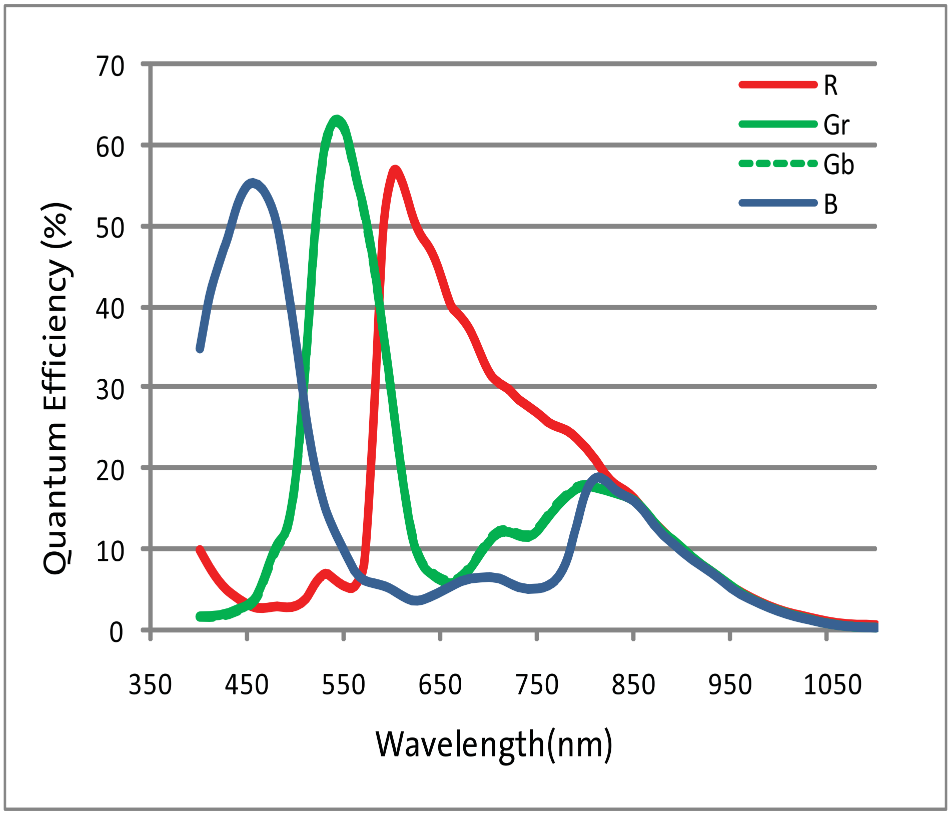
CRA vs. Image Height Plot	Image Height		CRA
	(%)	(mm)	(deg)
	0	0	0
	5	0.072	1.68
	10	0.144	3.39
	15	0.216	5.12
	20	0.288	6.88
	25	0.36	8.64
	30	0.432	10.4
	35	0.504	12.13
	40	0.576	13.82
	45	0.648	15.45
	50	0.72	17
	55	0.792	18.46
	60	0.864	19.82
	65	0.936	21.06
	70	1.008	22.17
	75	1.08	23.16
	80	1.152	24.01
	85	1.224	24.72
	90	1.296	25.29
	95	1.368	25.72
	100	1.44	26

Figure 33: Typical Quantum Efficiency



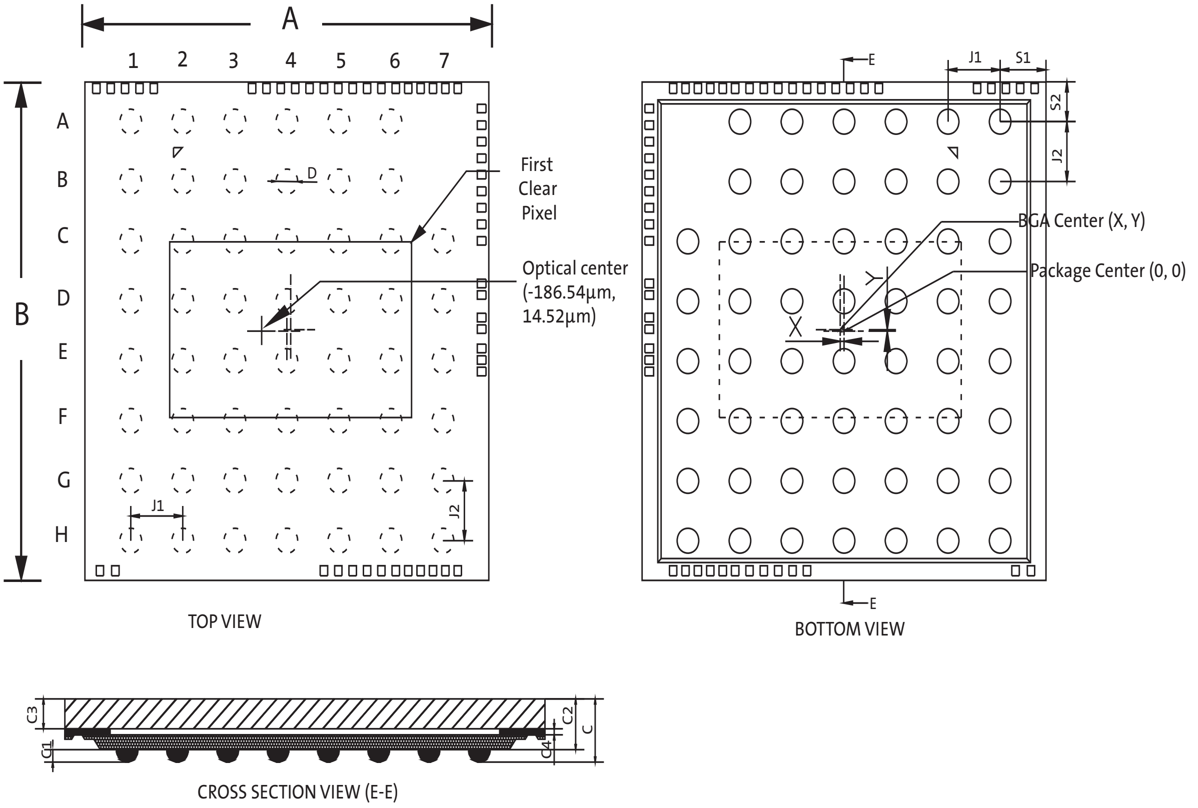
Note: Curve shown is without cover glass.

CSP Package Dimensions

Table 14: Package Dimensions

Parameter	Symbol	Nominal	Min	Max	Nominal	Min	Max
		Millimeters			Inches		
Package Body Dimension X	A	4.688	4.663	4.713	0.18457	0.18359	0.18556
Package Body Dimension Y	B	5.028	5.003	5.053	0.19795	0.19697	0.19894
Package Height	C	0.690	0.645	0.735	0.02717	0.02539	0.02894
Cavity height (glass to pixel distance)	C4	0.041	0.037	0.045	0.00161	0.00146	0.00177
Glass Thickness	C3	0.400	0.390	0.410	0.01575	0.01535	0.01614
Package Body Thickness	C2	0.570	0.535	0.605	0.02244	0.02106	0.02382
Ball Height	C1	0.120	0.100	0.140	0.00472	0.00394	0.00551
Ball Diameter	D	0.250	0.230	0.270	0.00984	0.00906	0.01063
Total Ball Count	N	54					
Ball Count X axis	N1	7					
Ball Count Yaxis	N2	8					
UBM	U	0.270	0.260	0.280	0.01063	0.01024	0.01102
Pins Pitch X axis	J1	0.600	0.590	0.610	0.02362	0.02323	0.02402
Pins Pitch Y axis	J2	0.600	0.590	0.610	0.02362	0.02323	0.02402
BGA ball center to package center offset in X-direction	X	0	-0.025	0.025	0	-0.00098	0.00098
BGA ball center to package center offset in Y-direction	Y	0	-0.025	0.025	0	-0.00098	0.00098
Edge to Ball Center Distance along X	S1	0.544	0.514	0.574	0.02142	0.02024	0.02260
Edge to Ball Center Distance along Y	S2	0.414	0.384	0.444	0.01630	0.01512	0.01748

Figure 34: Package Mechanical Drawing



TOP VIEW

BOTTOM VIEW

CROSS SECTION VIEW (E-E)

Table 15: Ball Matrix

	1	2	3	4	5	6	7
A	GND_IO	EXTCLK	SDATA	DOUT[1]	DOUT[3]	DOUT[5]	–
B	GND_IO	DOUT[0]	DOUT[2]	VDD	VDD_IO	DOUT[7]	–
C	SCLK	VDD_IO	DOUT[4]	DOUT[6]	GND_IO	PIXCLK	LV
D	VDD	VDD_IO	GND_IO	VDD_IO	DOUTLSB[1]	VDD_IO	GND_IO
E	NC	GND_IO	DOUTLSB[0]	AGND	VAA_PIX	VAA_PIX	AGND
F	NC	SADDR	FV	TMS[1]	VAA	VDD	VAA
G	NC	VDD_IO	TDI[1]	TCK[1]	RESET_BAR	TRST_BAR	VDD
H	VDD	GND_IO	VDD_IO	GND_IO	TDO[1]	OE_BAR	GND_IO

Note: 1. Do not use.

Aptina reserves the right to change products or specifications without notice.

Electrical Specifications

Caution **Stresses above those listed in Table 16 may cause permanent damage to the device.**

Table 16: Absolute Maximum Ratings

Symbol	Parameter	Rating		Unit
		Min	Max	
VDD_MAX	Core digital voltage	–0.3	2.4	V
VDD_IO_MAX	I/O digital voltage	–0.3	4.0	V
VAA_MAX	Analog voltage	–0.3	4.0	V
VAA_PIX_MAX	Pixel supply voltage	–0.3	4.0	V
VIN	DC input voltage	–0.3	VDD_IO + 0.3	V
IIN	Transient input current (0.5 sec. duration)	–	150	mA
T_{OP}	Operating temperature (measure at junction)	–30	75	°C
T_{STG}[1]	Storage temperature	–40	85	°C

Note: This is a stress rating only, and functional operation of the device at these or any other conditions above those indicated in the product specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Recommended Operating Conditions

Table 17: Operating Conditions

Symbol	Parameter	Min	Typ	Max	Units
VDD	Core digital voltage	1.7	1.8	1.95	V
VDD_IO	I/O digital voltage	2.5	2.8	3.1	V
		1.7	1.8	1.95	V
VAA	Analog voltage	2.5	2.8	3.1	V
VAA_PIX	Pixel supply voltage	2.5	2.8	3.1	V
T_J	Operating temperature (at junction)	–30	55	70	°C

Table 18: DC Electrical Characteristics

Symbol	Parameter	Condition	Min	Max	Unit
VIH	Input HIGH voltage		VDD_IO * 0.7	VDD_IO + 0.3	V
VIL	Input LOW voltage		–0.3	VDD_IO * 0.3	V
IIN	Input leakage current	VIN = 0V or VIN = VDD_IO		10	µA
VOH	Output HIGH voltage	VDD_IO = 1.8V, IOH = 2mA	VDD_IO – 0.3		V
VOL	Output LOW voltage	VDD_IO = 1.8V, IOH = 2mA	–	0.4	V

Table 19: Operating Current Consumption
Default Setup Conditions: PIXCLK = EXTCLK, VDD = 1.8V, VAA = VAA_PIX = VDD_IO = 2.8V, Tj = 25°C unless otherwise stated

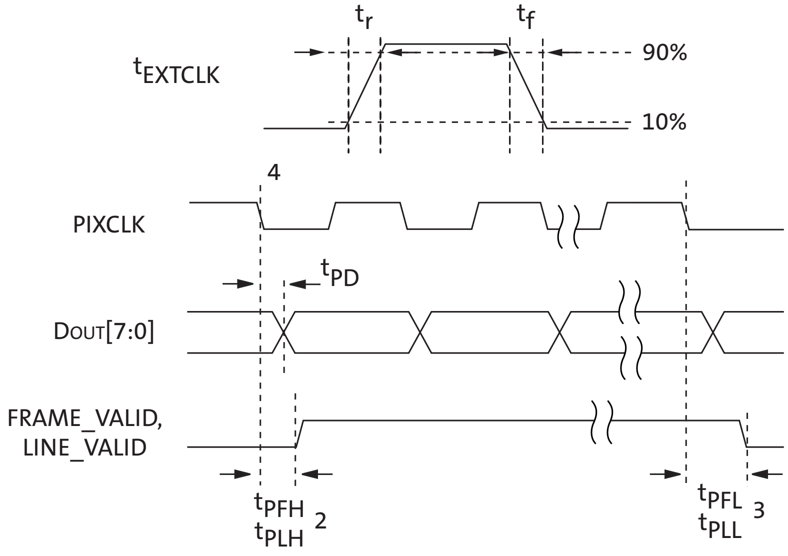
Symbol	Parameter	Conditions	Min	Typ	Max	Unit
VDD	Digital core supply voltage		1.7	1.8	1.95	V
VAA	Analog supply voltage		2.5	2.8	3.1	V
VAA_PIX	Pixel supply voltage		2.5	2.8	3.1	V
VDD_IO	Digital IO supply voltage	VDD_IO = 2.8V	2.5	2.8	3.1	V
		VDD_IO = 1.8V	1.7	1.8	1.95	V
IDD	Digital core supply current	QVGA, 111fps, EXTCLK = 54 MHz		18	25	mA
		VGA, 30 fps, EXTCLK = 27 MHz		9	15	mA
		VGA, 60 fps, EXTCLK = 54 MHz		19	25	mA
IAA	Analog supply current	QVGA, 111 fps, EXTCLK = 54 MHz		37	50	mA
		VGA, 30 fps, EXTCLK = 27 MHz		36	50	mA
		VGA, 60 fps, EXTCLK = 54 MHz		37	50	mA
IAA_PIX	Pixel supply current	QVGA, 111 fps, EXTCLK = 54 MHz		0.66	1.5	mA
		VGA, 30 fps, EXTCLK = 27 MHz		0.65	1.5	mA
		VGA, 60 fps, EXTCLK = 54 MHz		0.66	1.5	mA
	Total power consumption	QVGA, 111 fps, EXTCLK = 54 MHz		138	190	mW
		VGA, 30 fps, EXTCL K= 27 MHz		119	172	mW
		VGA, 60 fps, EXTCLK = 54 MHz		140	190	mW

Table 20: AC Electrical Characteristics
EXTCLK = 6–54 MHz; VDD = 1.8V; VDD_IO = VAA = VAA_PIX = 2.8V; TJ = 25°C unless otherwise stated

Symbol	Parameter	Conditions	Min	Typ	Max	Unit	Notes
^{f}EXTCLK	External clock frequency		6		54	MHz	1
^{t}R	External input clock rise time	10%-90% VDD_IO	–	2	5	ns	
^{t}F	External input clock fall time	90%-10% VDD_IO	–	2	5	ns	
DEXTCLK	External input clock duty cycle		40	50	60	%	
^{t}JITTER	External input clock jitter		–	500	–	ps	
^{t}CP	EXTCLK to PIXCLK propagation delay		–	16	–	ns	
^{f}PIXCLK	Pixel clock frequency		6		54	MHz	
^{t}RPIXCLK	Pixel clock rise time	CLOAD = 35pf	–	2	5	ns	
^{t}FPIXCLK	Pixel clock fall time	CLOAD = 35pf	–	2	5	ns	
^{t}PD	PIXCLK to data valid		–	1	5	ns	
^{t}PFH	PIXCLK to FV HIGH		–	1	5	ns	
^{t}PFL	PIXCLK to FV LOW		–	1	5	ns	
^{t}PLH	PIXCLK to LV HIGH		–	1	5	ns	
^{t}PLL	PIXCLK to LV LOW		–	1	5	ns	
PIXCLK slew rate							
	Programmable Slew = 7	VDD_IO = 2.8V, CLOAD = 35pf	–	0.177	–	V/ns	2
		VDD_IO = 1.8V, CLOAD = 35pf	–	0.100	–	V/ns	
	Programmable Slew = 4	VDD_IO = 2.8V, CLOAD = 35pf	–	0.167	–	V/ns	
		VDD_IO = 1.8V, CLOAD = 35pf	–	0.090	–	V/ns	
	Programmable Slew = 0	VDD_IO = 2.8V, CLOAD = 35pf	–	0.095	–	V/ns	
		VDD_IO = 1.8V, CLOAD = 35pf	–	0.047	–	V/ns	
Output slew rate							
	Programmable Slew = 7	VDD_IO = 2.8V, CLOAD = 35pf	–	0.493	–	V/ns	2
		VDD_IO = 1.8V, CLOAD = 35pf	–	0.226	–	V/ns	
	Programmable Slew = 4	VDD_IO = 2.8V, CLOAD = 35pf	–	0.392	–	V/ns	
		VDD_IO = 1.8V, CLOAD = 35pf	–	0.165	–	V/ns	
	Programmable Slew = 0	VDD_IO = 2.8V, CLOAD = 35pf	–	0.112	–	V/ns	
		VDD_IO = 2.8V, CLOAD = 35pf	–	0.059	–	V/ns	

Notes:
1. VIH/VIL restrictions apply.
2. EXTCLK = 6 MHz

Figure 35: Parallel Pixel Bus Timing Diagram



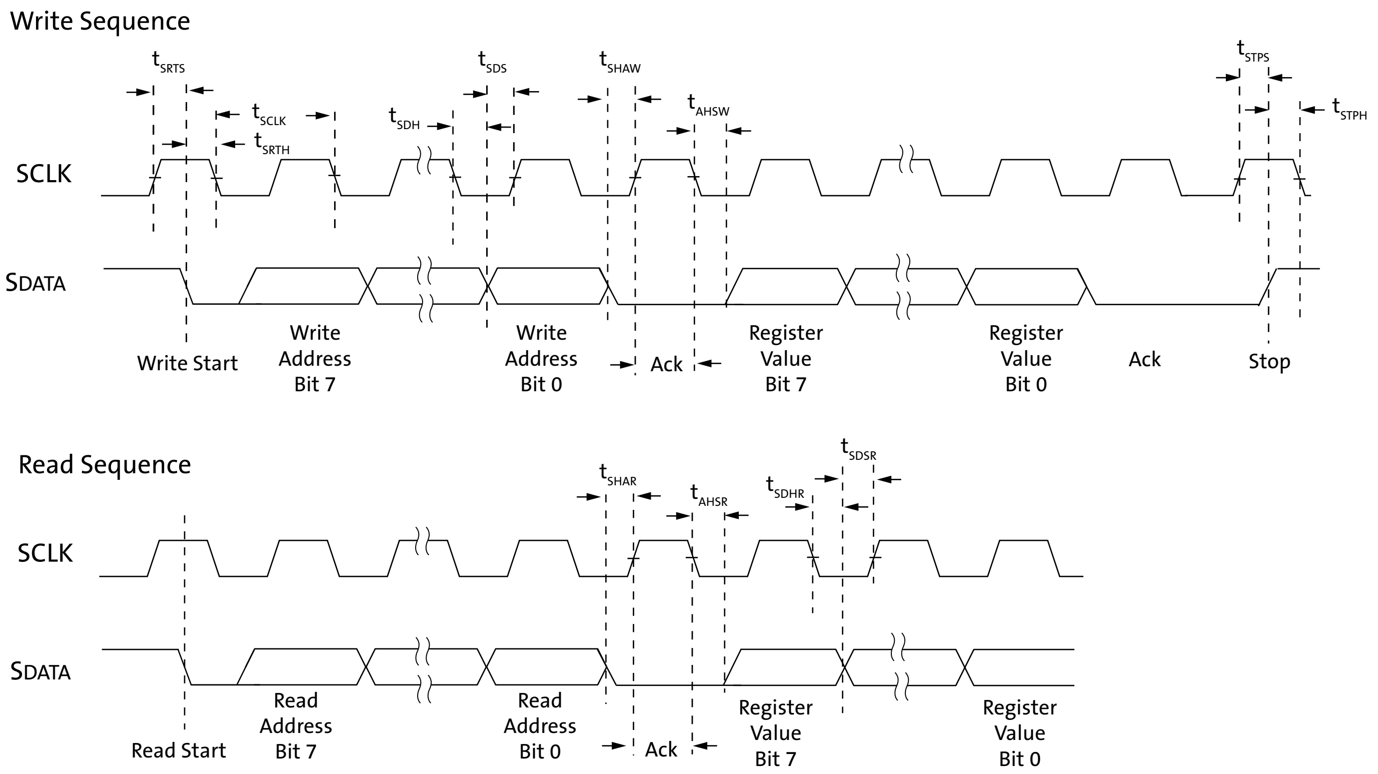
Notes:
1. FRAME_VALID leads LINE_VALID by 6 PIXCLKs.
2. FRAME_VALID trails LINE_VALID by 6 PIXCLKs.
3. DOUT[7:0], FRAME_VALID, and LINE_VALID are shown with respect to the falling edge of PIXCLK. This feature is programmable and DOUT[7:0], FRAME_VALID, and LINE_VALID can be synchronized to the rising edge of PIXCLK.
4. Propagation delay is measured from 50% of rising and falling edges.

Table 21: Two-Wire Serial Interface Timing Data

^{f}EXTCLK = 50 MHz; VDD = 1.8V; VDD_IO = 1.8V; VAA = 2.8V; VAA_PIX = 2.8V; T_J = 70°C; CLOAD = 68.5pF

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
^{f}SCLK	Serial interface input clock frequency		100	–	400	kHz
^{t}SCLK	Serial interface input clock period		2.5	–	10	µs
	SCLK duty cycle		45	50	55	%
^{t}r	SCLK/SDATA rise time		–	–	300	ns
^{t}SRTS	Start setup time	Master write to slave	600	–	–	
^{t}SRTH	Start hold time	Master write to slave	300	–	–	ns
^{t}SDH	SDATA hold	Master write to slave	300	–	650	ns
^{t}SDS	SDATA setup	Master write to slave	300	–	–	ns
^{t}SHAW	SDATA hold to ack	Master read from slave	150	–	–	ns
^{t}AHSW	Ack hold to SDATA	Master read from slave	150	–	–	ns
^{t}STPS	Stop setup time	Master write to slave	300	–	–	ns
^{t}STPH	Stop hold time	Master write to slave	600	–	–	ns
^{t}SHAR	SDATA hold to ack	Master write to slave	300	–	–	ns
^{t}AHSR	Ack hold to SDATA	Master write to slave	300	–	–	ns
^{t}SDHR	SDATA hold	Master read from slave	300	–	650	ns
^{t}SDSR	SDATA setup	Master read from slave	350	–	–	ns

Figure 36: Two-Wire Serial Bus Timing Parameters



Revision History

Rev. F **11/5/10**

- Applied updated Aptina template

Rev. E **1/18/10**

- Updated to Production
- Updated Table 2, "Available Part Numbers," on page 1
- Updated Figure 10: "Imaging a Scene," on page 16
- Updated Figure 11: "Three Pixels in Normal and Column Mirror Readout Mode," on page 17
- Updated Figure 12: "Three Rows in Normal and Row Mirror Readout Mode," on page 17
- Updated Figure 13: "Eight Pixels in Normal and Column Skip 2X Readout Mode," on page 18
- Updated Figure 33: "Typical Quantum Efficiency," on page 42
- Added "CSP Package Dimensions" on page 43

Rev. D **9/3/09**

- Updated to Preliminary
- Updated pixel dynamic range, SNR MAX, and responsivity in Table 1, "Key Performance Parameters," on page 1
- Updated signal names to conform to Aptina standard (changed OE_N to OE_BAR, TRST_N to TRST_BAR) in Table 3, "Signal Descriptions," on page 8 and in Figure 1: "MT9V117 Block Diagram," on page 6; also changed RESET_N to RESET_BAR in Figure 1.
- Added last sentence in 1st paragraph of "Soft Reset" on page 11
- Added Table 8, "Timing Chart," on page 14
- Updated Figure 9: "Pixel Color Pattern Detail (Top Right Corner)," on page 16, Figure 11: "Three Pixels in Normal and Column Mirror Readout Mode," on page 17, Figure 12: "Three Rows in Normal and Row Mirror Readout Mode," on page 17, Figure 13: "Eight Pixels in Normal and Column Skip 2X Readout Mode," on page 18, and Figure 16: "Image Flow Processor," on page 20
- Updated code for [test pattern setup] on page 21 and for [exit test pattern] on page 22
- Added Figure 24: "5 x 5 Grid," on page 28
- Updated Figure 25: "BT656 Image Data with SAV/EAV Codes," on page 31
- Added "Silicon Revision Identification" on page 35
- Updated Figure 33: "Typical Quantum Efficiency," on page 42
- Updated Table 16, "Absolute Maximum Ratings," on page 46, Table 18, "DC Electrical Characteristics," on page 47, Table 19, "Operating Current Consumption," on page 47, Table 20, "AC Electrical Characteristics," on page 48, Table 21, "Two-Wire Serial Interface Timing Data," on page 50

Rev C **7/1/09**

- Updated typical power consumption in Table 1, "Key Performance Parameters," on page 1
- In Table 3 on page 8, added notes not use to test pins
- Updated Figure 9: "Pixel Color Pattern Detail (Top Right Corner)," on page 16
- Updated Table 9, "Hue Control," on page 26
- Updated Figure 22, –22° Hue and Figure 23: "+22° Hue," on page 27

- Updated Figure 27: “Single Read from Current Location,” on page 38
- Updated pages 36 through 40
- Updated Table 16, “Operating/Standby Current Consumption,” on page 43
- Updated Table 20, “AC Electrical Characteristics,” on page 48

Rev. B 3/20/09

- In Table 1, “Key Performance Parameters,” on page 1:
  - Changed Dynamic range (71dB) to Pixel dynamic range (81dB)
  - Changed SNR MAX value to 42dB
  - Changed Responsivity value to 4.23 V/lux-sec
  - Changed Typical Power Consumption to 129mW
- Updated Table 3, “Signal Descriptions,” on page 8
- Deleted section on Tone Control
- Added Figure 33: “Typical Quantum Efficiency,” on page 42

Rev. A 12/17/08

- Initial release

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

This data sheet contains minimum and maximum limits specified over the power supply and temperature range set forth herein. Although considered final, these specifications are subject to change, as further product development and data characterization sometimes occur.