1/6-Inch 1080P High-Definition (HD) Digital Image Sensor

AR0260 Datasheet, Rev. G

For the latest datasheet, please visit www.onsemi.com

Features

- Superior low-light performance
- Ultra-low-power
- 1080p Full HD video at 30 fps
- 1.4µm pixel with ON Semiconductor A-Pix™ technology
- Internal master clock generated by on-chip phase locked loop (PLL) oscillator
- Electronic rolling shutter (ERS), progressive scan
- Programmable controls: gain, horizontal and vertical blanking, auto black level offset correction, frame size/rate, exposure, left–right and top–bottom image reversal, window size, and panning.
- Arbitrary image scaling with anti-aliasing
- Two-wire serial interface providing access to registers and microcontroller memory
- Selectable output data format: RAW8 and RAW8+2-bit
- Parallel and 1- or 2-lane MIPI data output
- Adaptive polynomial lens shading correction

Applications

- Embedded tablet, notebook, and tethered PC cameras
- Game consoles
- Cell phones, mobile devices
- Consumer video communications

General Description

ON Semiconductor 's AR0260 is a 1/6-inch 2.0Mp Full HD CMOS digital image sensor with an active-pixel array of 1920H x 1080V. It incorporates sophisticated on-chip camera functions such as windowing, mirroring, column and row skip modes, and snapshot mode. It is programmable through a simple two-wire serial interface. It is designed for low light performance.The AR0260 produces extraordinarily clear, sharp digital pictures, making it the perfect choice for a wide range of applications, including PC and notebook cameras, gaming systems, and mobile phones.

Table 1: Key Parameters

Parameter		Typical Value
Optical format		1/6-inch
Active pixels		1920 x 1080
Pixel size		1.4 µm
Color filter array		RGB Bayer
Shutter type		Electronic rolling shutter (ERS)
Input clock range		6 – 54 MHz
Output pixel clock maximum		96 MHz
Output MIPI data rate maximum		768 Mb/s per lane
Frame Rate	1080p (full res)	30 fps
	720p	60 fps
	VGA	60 fps
	QVGA	120 fps
Responsivity		0.64 V/lux-sec
SNR_{MAX}		33 dB
Pixel dynamic range		65 dB
Supply voltage	Digital	1.7 – 1.95 V
	Analog	2.5 – 3.1 V
	I/O	1.7 – 1.95 V or 2.5 – 3.1 V
	PHY	1.7 – 1.95 V
Power consumption[1]		270 mW
Operating temperature		–30°C to +70° C
Chief ray angle		28°
Package options		CSP, Bare die

Notes: 1. Power consumption for typical voltages and full resolution, maximum frame rate output, excluding I/O.

Ordering Information

Table 2: Available Part Numbers

Part Number	Product Description	Orderable Product Attribute Description
AR0260LCSC28SUKAL-A-CR	2 MP 1/6" CIS	Chip Tray without Protective Film

Table of Contents

List of Figures

List of Tables

Functional Description

The ON Semiconductor AR0260 can be operated in its default mode or programmed for frame size, exposure, gain, and other parameters. The default mode output is a 1080p-resolution image at 30 frames per second (fps). It outputs 8-bit or 10-bit raw data, using either the parallel or serial (MIPI) output ports.

FRAME_VALID and LINE_VALID signals are output on dedicated pins, along with a synchronized pixel clock in parallel mode.

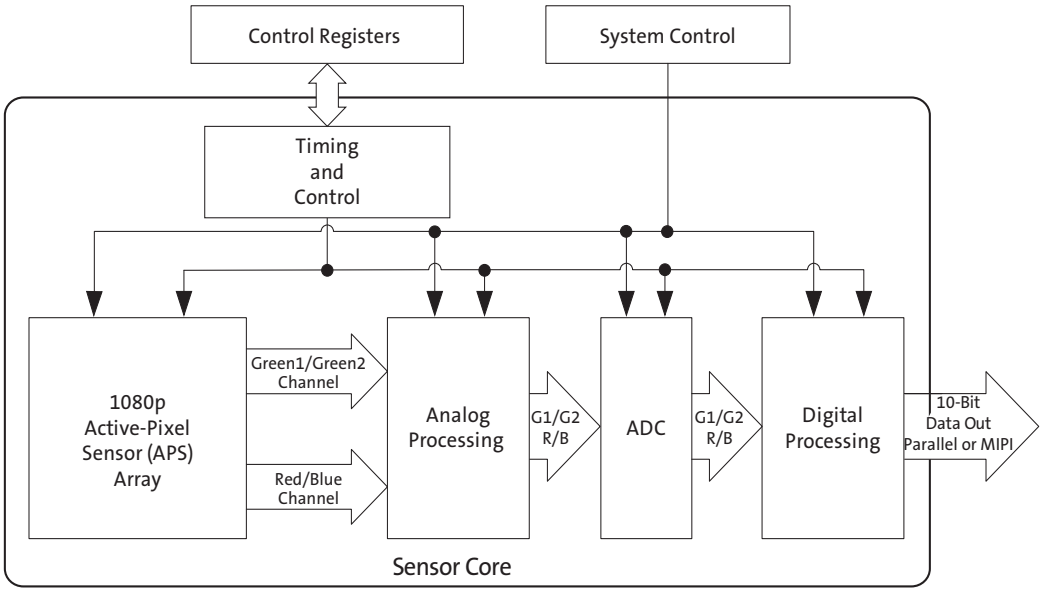
The AR0260 includes additional features to allow application-specific tuning: windowing and offset, auto black level correction, mirroring, and sub-sampling. Optional register information can be embedded in the first and last 2 lines of the image frame.

The sensor is designed to operate in a wide temperature range (–30°C to +70°C).

Architecture Overview

The AR0260 is a 1/6-inch 2.0Mp progressive-scan sensor that generates a stream of pixel data at a constant frame rate. It uses an on-chip, phase-locked loop (PLL) that can be optionally enabled to generate all internal clocks from a single master input clock running between 6 and 54 MHz. The maximum output pixel rate is 96 MHz. Figure 1 shows a block diagram of the sensor.

Figure 1: AR0260 Block Diagram



User interaction with the sensor is through the two-wire serial bus, which communicates with the array control, analog signal chain, and digital signal chain. The core of the sensor is a 2.0 Mp Active- Pixel Sensor array. The timing and control circuitry sequences through the rows of the array, resetting and then reading each row in turn. In the time interval between resetting a row and reading that row, the pixels in the row integrate incident light. The exposure is controlled by varying the time interval between reset and readout. Once a row has been read, the data from the columns is sequenced through an

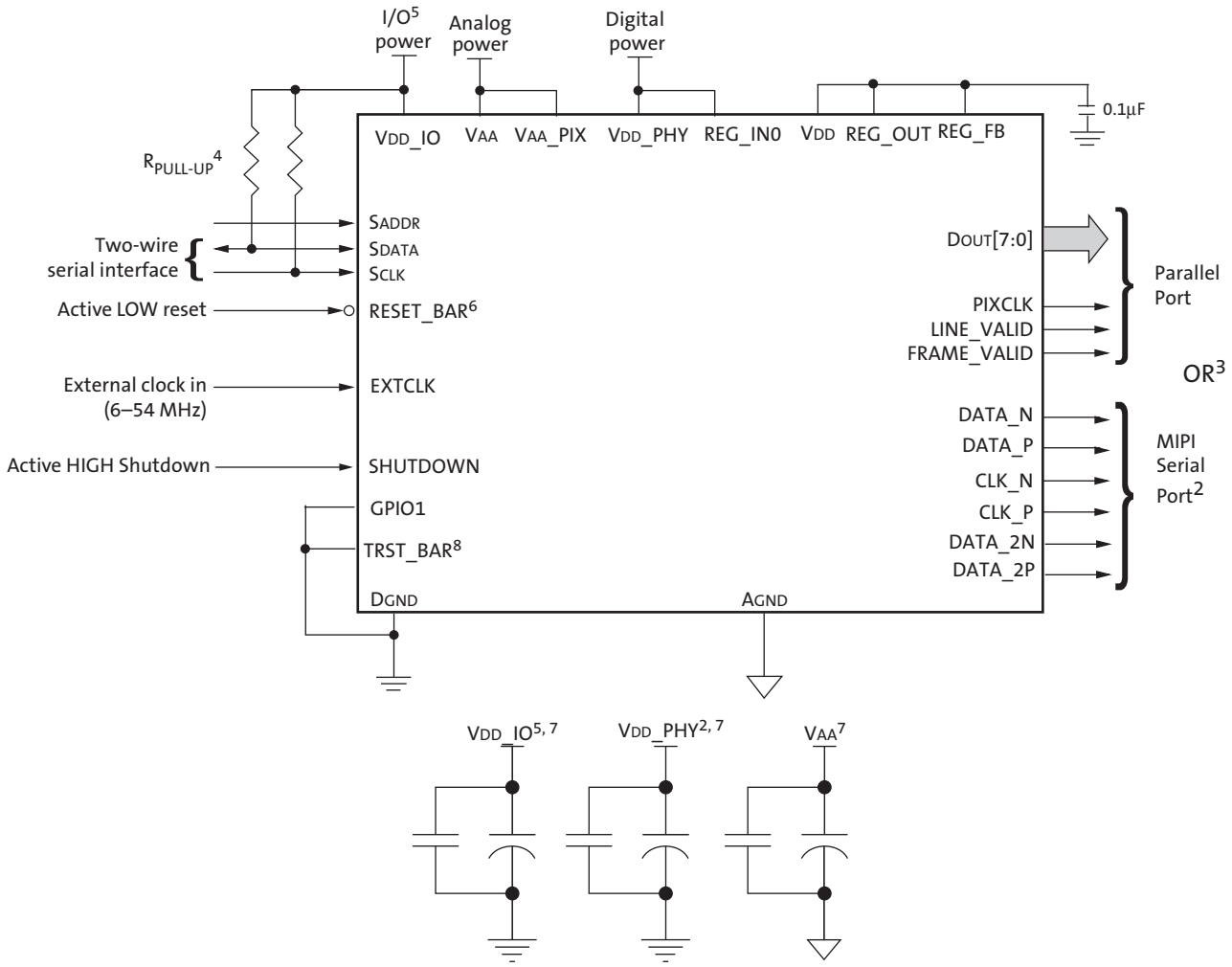
analog signal chain (providing offset correction and gain), and then through an analog-to-digital converter (ADC). The output from the ADC is a 10-bit value for each pixel in the array. The ADC output passes through a digital processing signal chain (which provides further data path corrections and applies digital gain). Finally, the output from the digital processing chain is passed either to the parallel or serial (MIPI) data port for transmission to the host system. The AR0260 also includes programmable I/O slew rate to minimize EMI.

System Interfaces

Figure 2 on page 8 shows typical AR0260 device connections. For low-noise operation, the AR0260 requires separate power supplies for analog and digital sections of the die. Both power supply rails must be decoupled from ground using capacitors as close as possible to the die. The use of inductance filters is not recommended on the power supplies or output signals.

The AR0260 provides dedicated inputs for digital core, PHY, and I/O power domains that can be at different voltages. The PLL and analog circuitry require clean power sources. Table 3 on page 9 provides the signal descriptions for the AR0260.

Figure 2: Typical Configuration



Notes:
1. This typical configuration shows only one scenario out of multiple possible variations for this sensor.
2. If a MIPI Interface is not required, the MIPI serial port must be left floating. The VDD_PHY power signal must always be connected to the 1.8V supply.
3. Only one of the output modes (serial or parallel) can be used at any time.
4. ON Semiconductor recommends a 1.5kΩ resistor value for the two-wire serial interface RPULL-UP; however, greater values may be used for slower transmission speed.
5. All inputs must be configured with VDD_IO.
6. RESET_BAR has an internal pull-up resistor and can be left floating.
7. ON Semiconductor recommends that 0.1µF and 1µF decoupling capacitors for each power supply are mounted as close as possible to the pad. Actual values and numbers may vary depending on layout and design considerations.
8. TRST_BAR and GPIO1 connect to GND for normal operation.

Table 3: Pin Descriptions

Name	Type	Description	Note
EXTCLK	Input	Input clock signal.	
RESET_BAR	Input/PU	Master reset signal, active LOW. This signal has an internal pull up.	
SCLK	Input	Two-wire serial interface clock.	
SDATA	I/O	Two-wire serial interface data.	
SADDR	Input	Selects device address for the two-wire serial interface.	
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.	
CLK_N	Output	Differential MIPI clock (sub-LVDS, negative).	2
CLK_P	Output	Differential MIPI clock (sub-LVDS, positive).	2
DATA_N	Output	Differential MIPI data (sub-LVDS, negative).	2
DATA_P	Output	Differential MIPI data (sub-LVDS, positive).	2
DATA_2N	Output	Differential MIPI data (sub-LVDS, negative).	2
DATA_2P	Output	Differential MIPI data (sub-LVDS, positive).	2
GPIO0	Input/Output	General purpose input/output.	
GPIO1	Input/Output	General purpose input/output.	5
GPIO2	Input/Output	General purpose input/output.	
SHUTDOWN	Input	Low power shutdown control, active HIGH.	
TRST_BAR	Input	Must be tied to DGND in normal operation.	
VDD	Supply	Digital power. Must connect to REG_FB.	4
DGND	Supply	Digital ground.	1
VDD_IO	Supply	I/O power supply.	
VAA	Supply	Analog power.	
VAA_PIX	Supply	Analog pixel power.	
AGND	Supply	Analog ground.	1
VPP	Supply	Reserved. To be left floating in normal operation.	
REG_IN0	Supply	Digital power. Must connect to VDD_PHY.	
REG_OUT	Supply	Digital power output.	
REG_FB	Supply	Digital power. Must connect to VDD.	4
VDD_PHY	Supply	I/O power supply for the MIPI interface. Must connect to REG_IN0	3

Notes:
1. AGND and DGND are not connected internally.
2. To be left floating if not using feature.
3. Must always be connected even when not using MIPI.
4. The VDD and REG_FB pins must be connected together and have a 0.1µF decoupling capacitor attached.
5. GPIO1 pin functions as CONFIG during power-on cycle. GPIO1 pin must be driven low during power-on cycle. GPIO1 pin should be either connected to DGND or needs pull-down resistor.

Decoupling Capacitor Recommendations

It is important to provide clean, well regulated power to each power supply. The ON Semiconductor 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, ON Semiconductor 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. ON Semiconductor 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. ON Semiconductor recommends placing a 10µF capacitor for each supply off-module but close to each supply.
4. Give priority to the V_{AA} 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

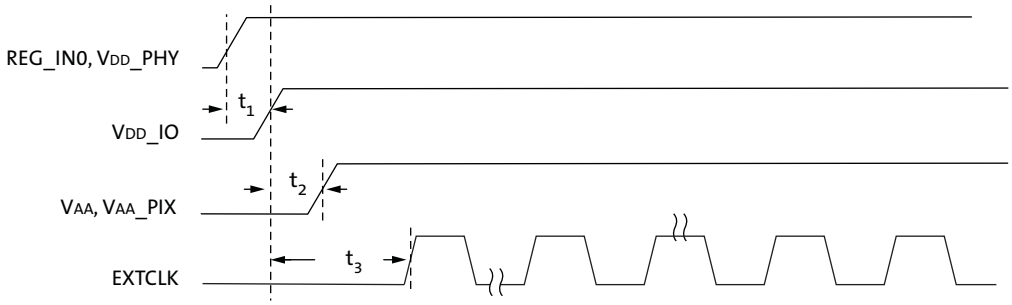


Table 4: Power-Up Signal Timing

Symbol	Parameter	Min	Max	Unit
t_1	Delay from REG_IN0 and V_{DD}_PHY to V_{DD}_IO	50	200	ms
t_2	Delay from V_{DD}_IO to V_{AA} and V_{AA}_PIX	0	50	ms
t_3	EXTCLK activation	0	–	ms

Power-Down Sequence

Powering down the sensor requires voltages to be applied in a particular order, as seen in Figure 4. The timing requirements are shown in Table 5.

Figure 4: Power-Down Sequence

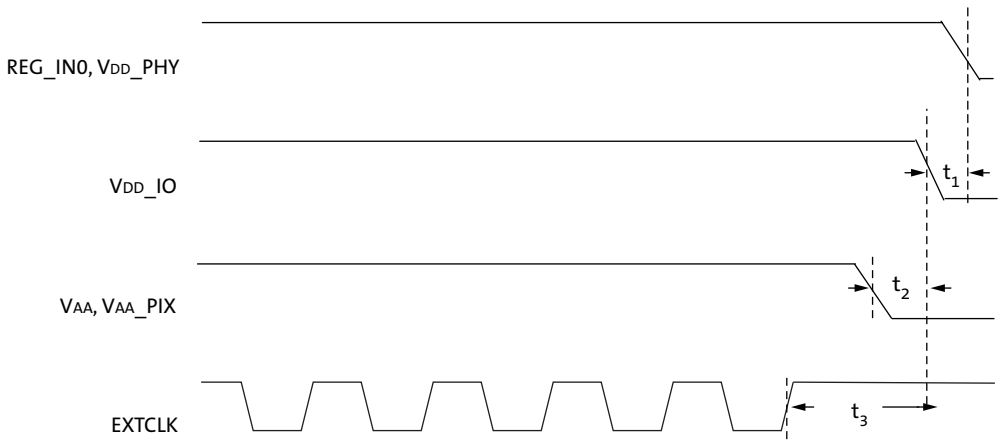


Table 5: Power-Down Signal Timing

Symbol	Parameter	Min	Max	Unit
t_1	Delay from VDD_IO to REG_IN0 and VDD_PHY	0	200	ms
t_2	Delay from VAA and VAA_PIX to VDD_IO	0	50	ms
t_3	EXTCLK deactivation	0	–	ms

Reset Modes

Three types of reset are available:

- An internal power-on reset
- 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

Power-On Reset

The AR0260 includes a power-on reset feature that initiates a reset upon power-up. The RESET_BAR signal has an internal pull-up resistor and can be left floating.

Hard Reset

The AR0260 enters the reset state when the external RESET_BAR signal is asserted LOW, as shown in Figure 5. Parallel data output signals will be in High-Z state.

Figure 5: Hard Reset Operation

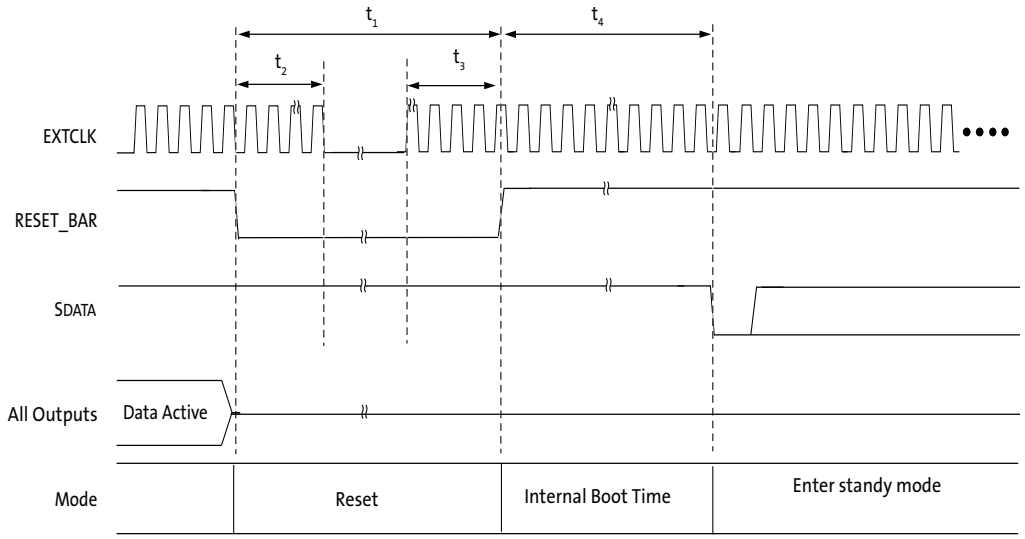


Table 6: Hard Reset

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]	–	–	35	ms

Notes: 1. This delay is dependent on EXTCLK frequency.

The output states during hard reset are shown in Table 7.

Table 7: Status of Output Signals During Hard Reset

Signal	Reset
DOUT[7:0]	High-Z
PIXCLK	High-Z
LV	High-Z
FV	High-Z
DATA_N	0
DATA_P	0
DATA_2N	0
DATA_2P	0
CLK_N	0
CLK_P	0

Soft Reset

The host processor can reset the AR0260 using the two-wire serial interface by writing to SYSCTL 0x001A. SYSCTL 0x001A[0] is used to reset the AR0260 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 up to 35 ms, depending on EXTCLK frequency.

Figure 6: Soft Reset Operation

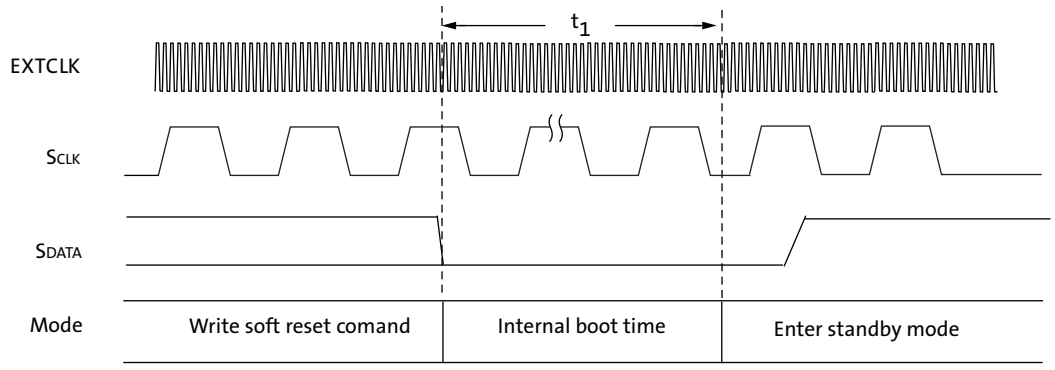


Table 8: Soft Reset Signal Timing

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

Notes: 1. This delay is dependent on EXTCLK frequency.

Shutdown Mode

The shutdown mode is entered when the SHUTDOWN pin is asserted. All power to the AR0260 is disabled and no state, register, or patch information is retained. VDD_IO and REG_IN0/VDD_PHY must be powered during Shutdown. Analog power (VAA and VAA_PIX) is optional but typically would also be left on. All pins hold their state (Input, output or tristate) during shutdown. Inputs are ignored during Shutdown and can be high, low or tristate. EXTCLK can be turned off or left running. An internal POR is automatically generated on exit from Shutdown.

Figure 7: Shutdown Operation

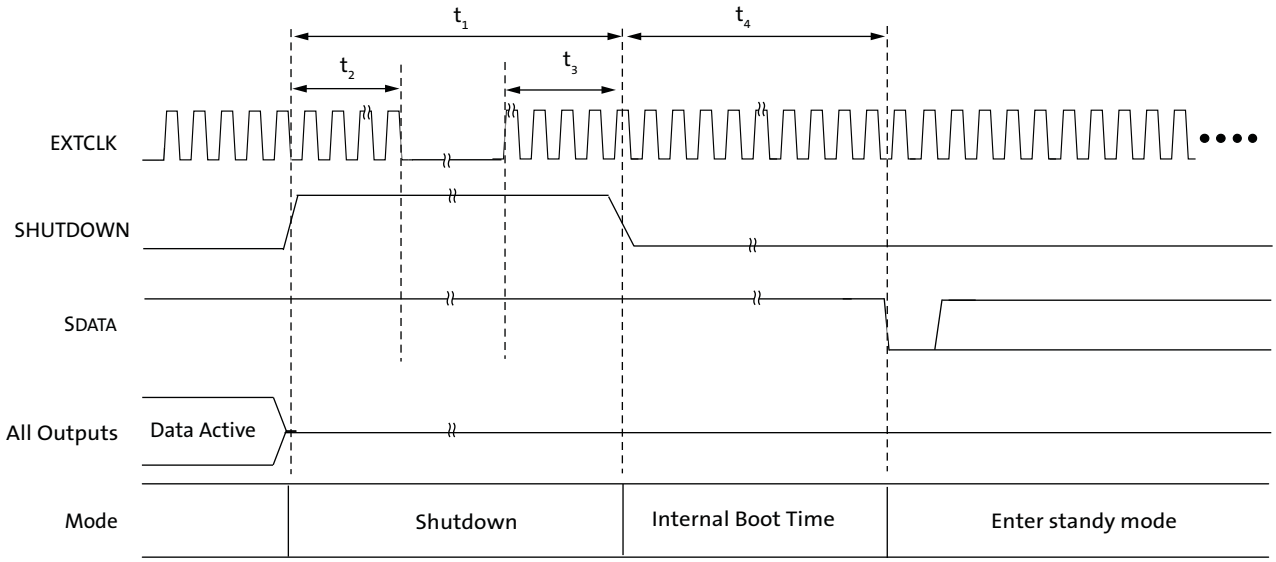


Table 9: Shutdown

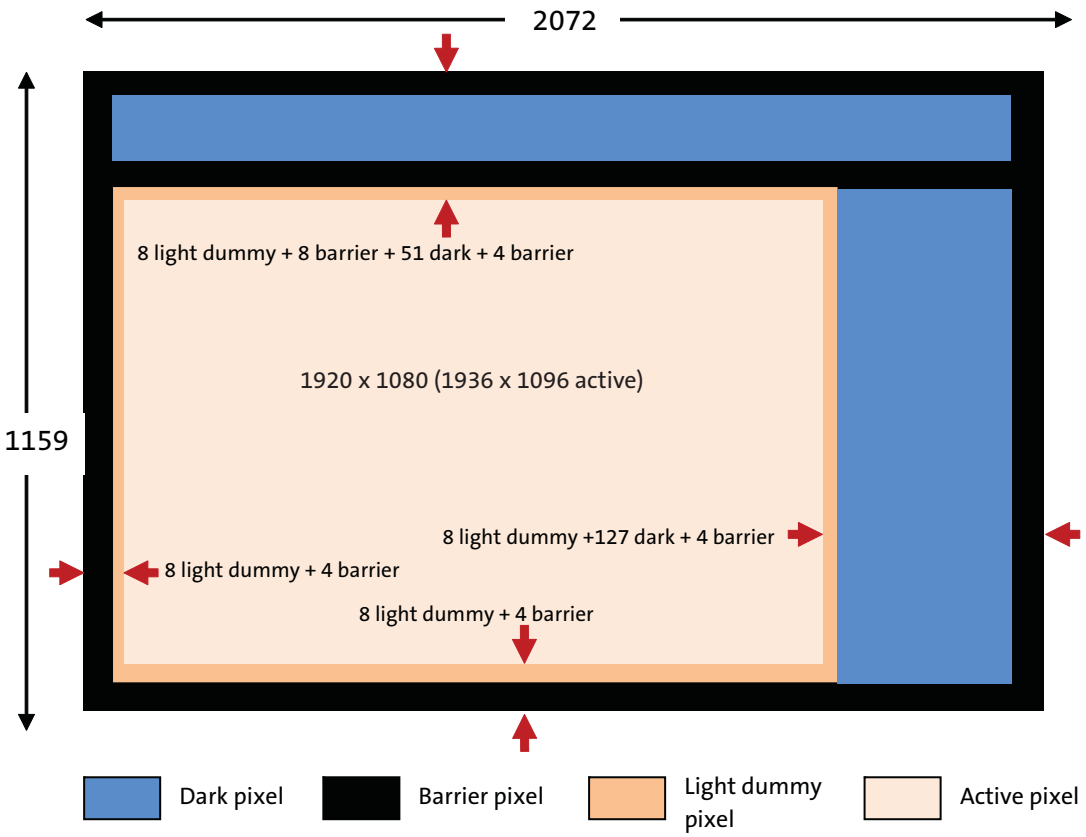
Symbol	Definition	Min	Typ	Max	Unit
t_1	SHUTDOWN pulse width	50	–	–	EXTCLK cycles
t_2	Active EXTCLK required after SHUTDOWN asserted	0	–	–	
t_3	Active EXTCLK required before SHUTDOWN de-asserted	0	–	–	
t_4	Maximum internal boot time[1]	–	–	35	ms

Notes: 1. This delay is dependent on EXTCLK frequency.

Pixel Data Format

The AR0260 pixel array is configured as 1928 columns by 1088 rows, (see Figure 8). The dark pixels are optically black and are used internally to monitor black level. There are 1936 columns by 1096 rows of optically active pixels. While the sensor's format is 1920 x 1080, the additional active columns and active rows are included for use when horizontal or vertical mirrored readout is enabled, to allow readout to start on the same pixel. The active area is surrounded with optically transparent dummy pixels to improve image uniformity within the active area. Not all dummy pixels or dark pixels can be read out.

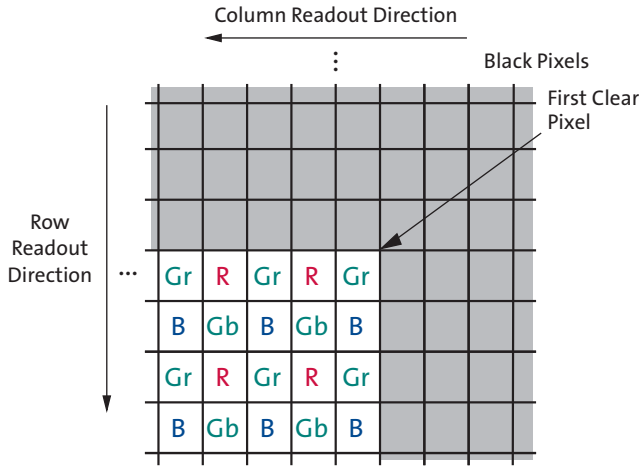
Figure 8: Pixel Array Description



Pixel Pattern Detail

The output from the sensor core is a Bayer pattern, as shown in Figure 9; alternate rows are a sequence of either green and red pixels or blue and green pixels. The analog signal chain provides per-color control of the pixel data.

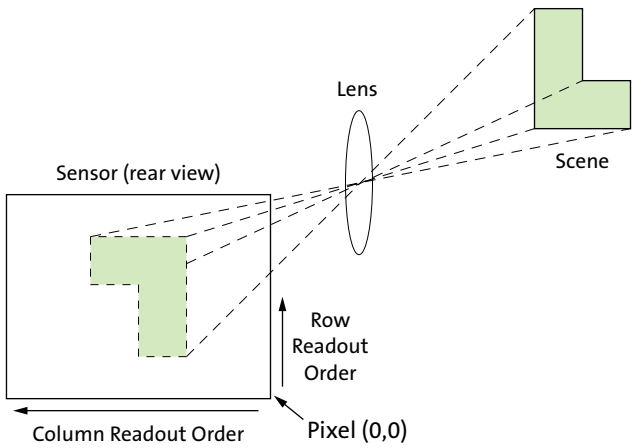
Figure 9: Pixel Color Pattern Detail (Top Right Corner)



The AR0260 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



Output Data Format

The AR0260 image data is read out in a progressive scan. Valid image data is surrounded by horizontal blanking and vertical blanking, as shown in Figure 11.

LINE_VALID is HIGH in the shaded region of the figure.

Figure 11: Spatial Illustration of Image Readout

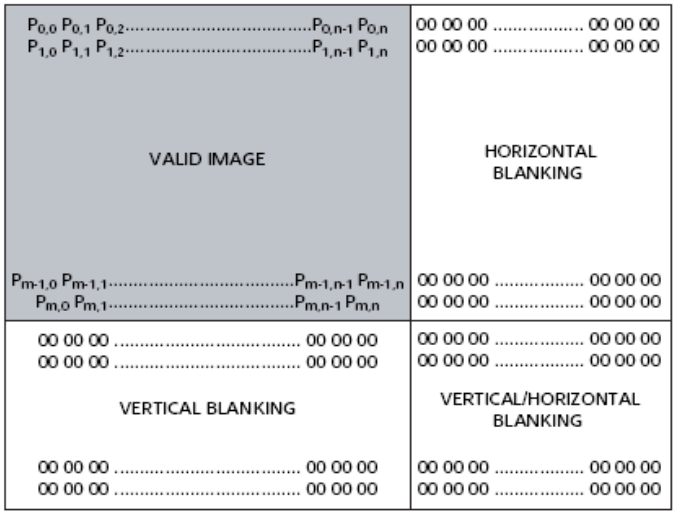


Image Data Output Interface

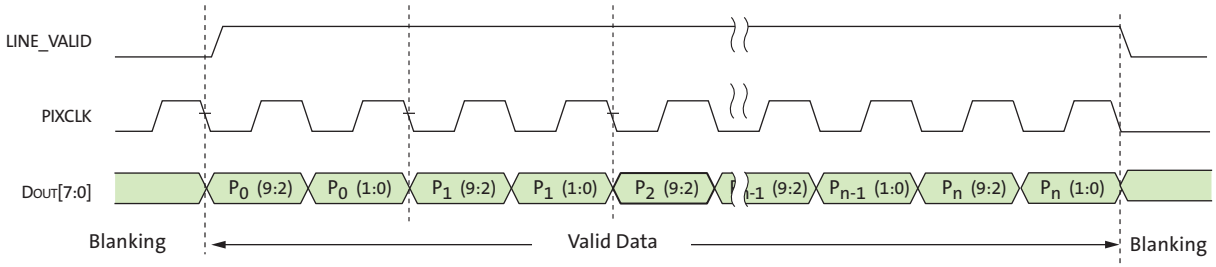
The user can select either the 8-bit parallel or serial MIPI output to transmit the sensor image data to the host system. Only one of the output modes can be used at any time. The AR0260 has an output FIFO to retain a constant pixel output clock.

Parallel Port

The AR0260 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.

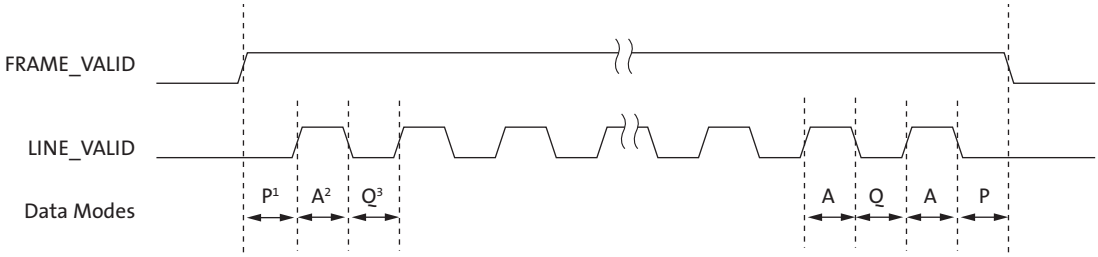
AR0260 output data is synchronized with the PIXCLK output. When LV is HIGH, one pixel value is output on the 8-bit DOUT port every TWO PIXCLK periods as shown in Figure 12. PIXCLK is continuously running, even during the blanking period. PIXCLK phase can be varied by 50 percent, controlled using a register.

Figure 12: Pixel Data Timing Example



Note: Shown is 10-bit Bayer data in 8 + 2 mode.

Figure 13: 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.

MIPI Port

The MIPI output implements a serial differential sub-LVDS transmitter capable of up to 1536 Mbps (768 Mbps/lane). It supports multiple formats, error checking, and custom short packets.

When the sensor is in the hardware standby system state or in the software standby system state, the MIPI signals (CLK_P, CLK_N, DATA_P, DATA_N, DATA_2P, DATA_2N) indicate ultra low power state (ULPS) corresponding to (nominal) 0V levels being driven on CLK_P, CLK_N, DATA_P, DATA_N, DATA_2P, and DATA_2N. This is equivalent to signaling code LP-00.

When the sensor enters the streaming system state, the interface goes through the following transitions:

1. After the PLL has locked and the bias generator for the MIPI drivers has stabilized, the MIPI interface transitions from the ULPS state to the ULPS-exit state (signaling code LP–10).
2. After a delay (TWAKEUP), the MIPI interface transitions from the ULPS-exit state to the TX-stop state (signaling code LP–11).
3. After a short period of time (the programmed integration time plus a fixed overhead), frames of pixel data start to be transmitted on the MIPI interface. Each frame of pixel data is transmitted as a number of high-speed packets. The transition from the TX-stop state to the high-speed signaling states occurs in accordance with the MIPI specifications. Between high-speed packets and between frames, the MIPI interface idles in the TX-stop state. The transition from the high-speed signaling states and the TX-stop state takes place in accordance with the MIPI specifications.

4. If the sensor is reset, any frame in progress is aborted immediately and the MIPI signals switch to indicate the ULPS.
5. If the sensor is taken out of the streaming system state and reset_register[4] = 1 (standby end-of-frame), any frame in progress is completed and the MIPI signals switch to indicate the ULPS.

If the sensor is taken out of the streaming system state and reset_register[4] = 0 (standby end-of-frame), any frame in progress is aborted as follows:

1. Any long packet in transmission is completed.
2. The end of frame short packet is transmitted.

After the frame has been aborted, the MIPI signals switch to indicate the ULPS.

Uncompressed Bayer Bypass Output

Raw or processed 10-bit Bayer data from the sensor core can be output in bypass mode by:

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 10.

Table 10: 2-Byte Bayer Format

2-Byte Bayer Format	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

Readout Order

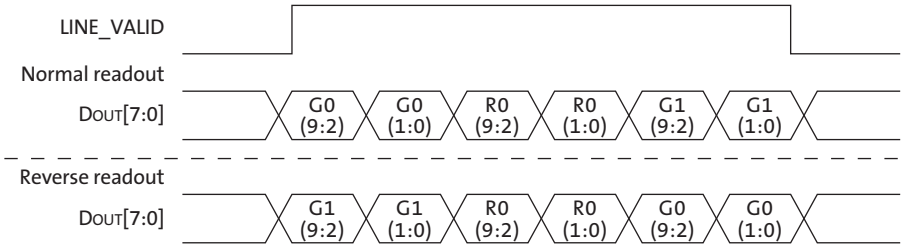
The sensor core supports different readout options to modify the image before it is sent to the host system. The readout can be limited to a specific window size of the original pixel array.

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.

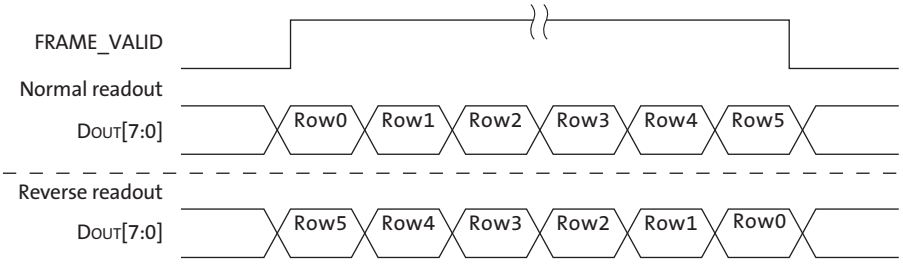
When the sensor is configured to mirror the image horizontally, the order of pixel readout within a row is reversed, so that readout starts from the last column address and ends at the first column address. Figure 14 shows a sequence of 3 pixels being read out with normal readout and reverse readout (Bayer8 + 2 mode shown).

Figure 14: Three Pixels in Normal and Column Mirror Readout Mode



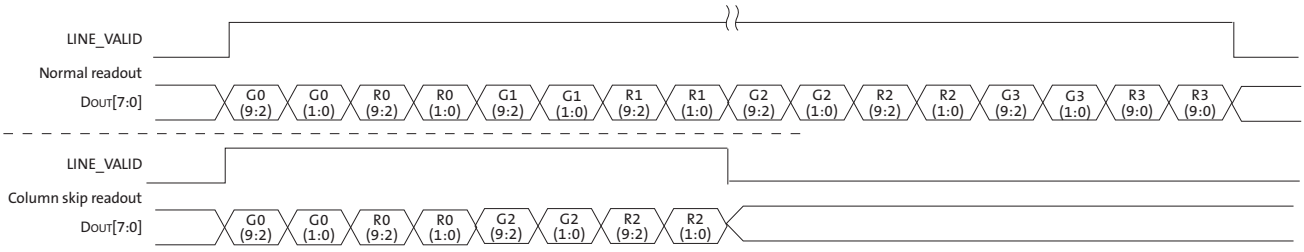
When the sensor is configured to flip the image vertically, the order in which pixel rows are read out is reversed, so that row readout starts from the last row address and ends at the first row address. Figure 15 on page 20 shows a sequence of 6 rows being read out with normal readout and reverse readout.

Figure 15: Six Rows in Normal and Row Mirror Readout Mode



The AR0260 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 16 shows Bayer 8 + 2 readout with 2X skipping.

Figure 16: Eight Pixels in Normal and Column Skip 2X Readout Mode



Pixel Readouts

The following diagrams show a sequence of data being read out with no skipping. The effect of the different subsampling on the pixel array readout is shown in Figure 17 through Figure 21 on page 23.

Figure 17: Pixel Readout (No Skipping)

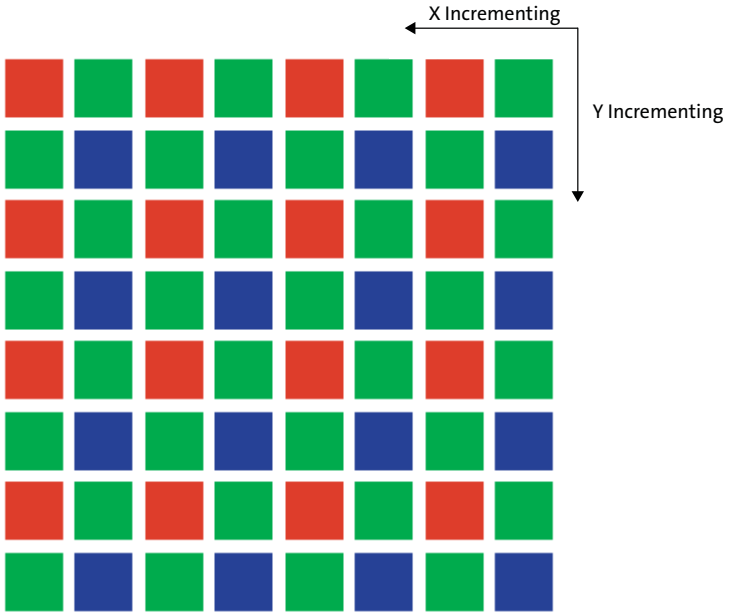


Figure 18: Pixel Readout (Column Skipping)

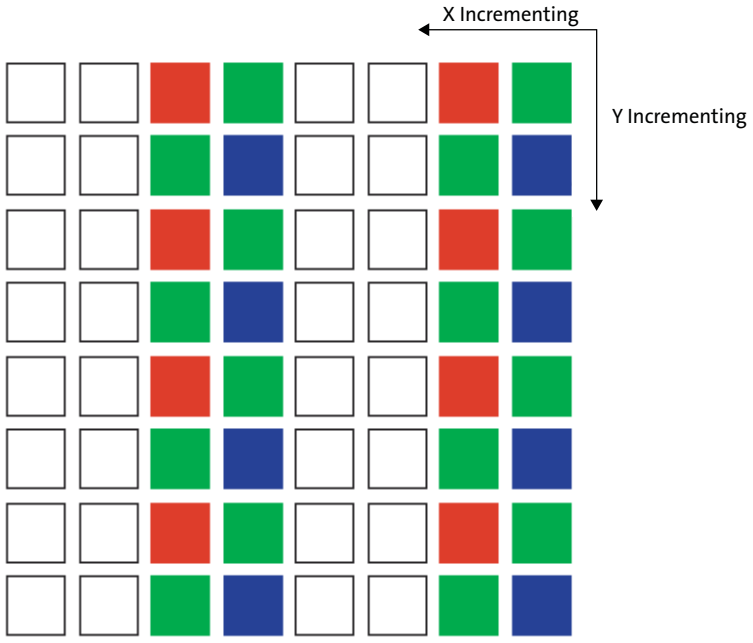


Figure 19: Pixel Readout (Row Skipping)

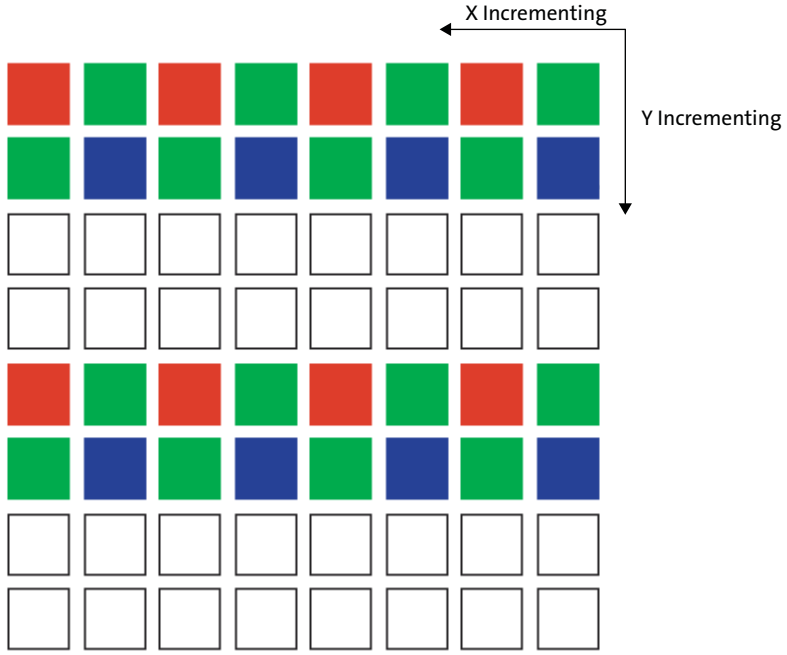
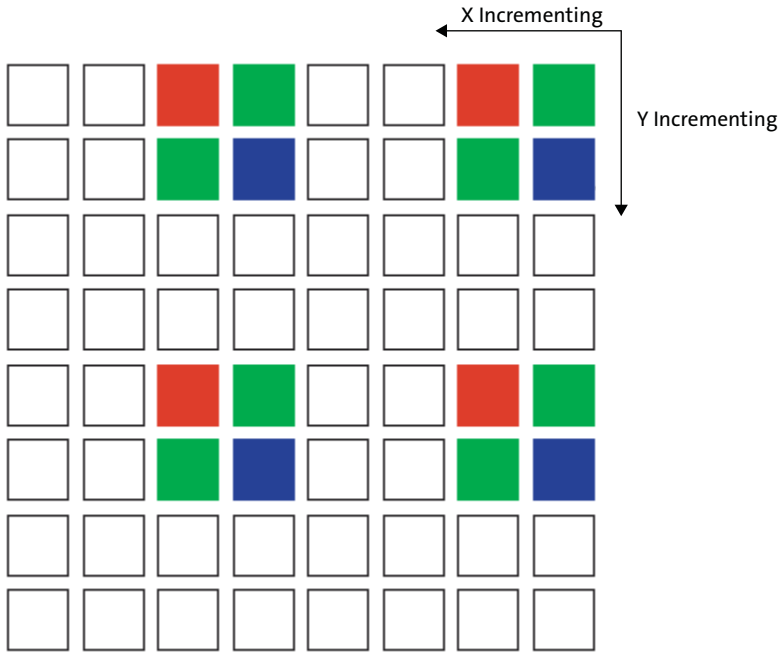


Figure 20: Pixel Readout (Column and Row Skipping)



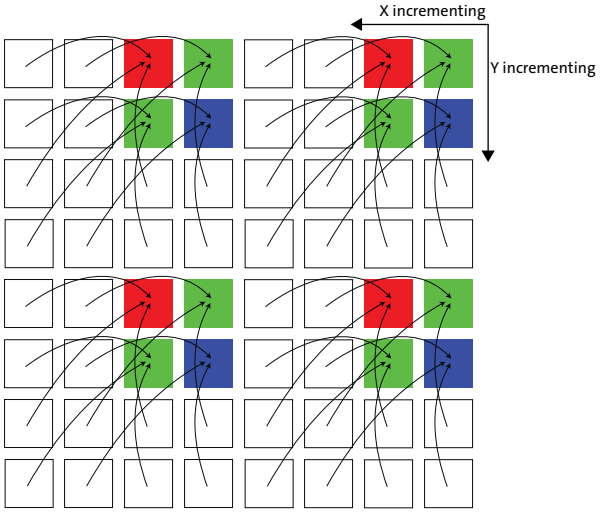
Binning

The AR0260 sensor core supports 2 x 2 binning. Binning has many of the same characteristics as subsampling but because it gathers image data from all pixels in the active window (rather than a subset of them), it achieves superior image quality and avoids the aliasing artifacts that can be a characteristic side effect of subsampling.

Binning is enabled by selecting the appropriate subsampling settings. Subsampling may require sensor window size adjustment when binning is enabled.

The effect of binning is shown in Figure 21.

Figure 21: Pixel Readout (column and row binning)



Analog Gain

The analog gain of the AR0260 sensor consists of three components: a fine (initial) gain, a course (ASC1) gain, and a column amplifier gain. The level of analog gain applied is controlled by the per-color gain and registers. The recommended gain tables are listed in Table 11. It is recommended that these registers are configured before streaming images.

Table 11: Recommended Sensor Analog Gain Tables

GAIN STAGE	2X	3X	4X	5X	6X	7X	8X	16X
Initial	32(1X)	48(1.5X)	32(1X)	40(1.25X)	48(1.5X)	56(1.75X)	32(1X)	64(2X)
ASC1	0(1X)	0(1X)	0(1X)	0(1X)	0(1X)	0(1X)	2(2X)	2(2X)
Column	2(2X)	2(2X)	3(4X)	3(4X)	3(4X)	3(4X)	3(4X)	3(4X)

Frame Rate Control

The formulas for calculating the frame rate of the AR0260 are shown below.

The line length is programmed directly in pixel clock periods through register line_length_pck. For a specific window size, the minimum line length can be found in Equation 1:

$$minimum\ line\_length\_pck = \left(\frac{x\_addr\_end - x\_addr\_start + 1}{subsampling\ factor} + min\_line\_blanking\_pck\right) \quad \text{(EQ 1)}$$

Note that line_length_pck also needs to meet the minimum line length requirement set by min_line_length_pck. The row time can either be limited by the time it takes to sample and reset the pixel array for each row, or by the time it takes to sample and read out a row. Values for min_line_blanking_pck are provided in "Minimum Row Time" on page 24.

The frame length is programmed directly in number of lines in the register frame_length_lines. For a specific window size, the minimum frame length can be found in Equation 2:

$$minimum\ frame\_length\_lines = \left(\frac{y\_addr\_end - y\_addr\_start + 1}{subsampling\ factor} + min\_frame\_blanking\_lines\right) \quad \text{(EQ 2)}$$

The frame rate can be calculated from these variables and the pixel clock speed as shown in Equation 3:

$$frame\ rate = \frac{2 \times cam\_sensor\_cfg\_pixclk}{line\_length\_pck\ x\ frame\_length\_lines} \quad \text{(EQ 3)}$$

If coarse_integration_time is set larger than frame_length_lines the frame size will be expanded to coarse_integration_time + 1.

Minimum Row Time

The minimum row time and blanking values with default register settings are shown in Table 12.

Table 12: Minimum Row Time and Blanking Numbers

	No Row Binning	Row Binning
min_line_blanking_pck	0x0492	0x07C4
min_line_length_pck	0x05D8	0x09C8

In addition, enough time must be given to the output FIFO so it can output all data at the set frequency within one row time.

There are therefore two checks that must all be met when programming line_length_pck:

- line_length_pck ≥ min_line_length_pck in Table 12.
- line_length_pck ≥ (x_addr_end - x_addr_start + x_odd_inc)/((1+x_odd_inc)/2) + min_line_blanking_pck in Table 12.

Minimum Frame Time

The minimum number of rows in the image is 2, so min_frame_length_lines will always equal (min_frame_blanking_lines + 2).

Table 13: Minimum Frame Blanking Numbers

	No Row Binning	Row Binning
min_frame_blanking_lines	0x0051	0x004D

Integration Time

The integration (exposure) time of the AR0260 is controlled by the fine_integration_time and coarse_integration_time registers.

The actual integration time is given by:

$$integration\_time = \frac{((coarse\_integration\_time * line\_length\_pck) + fine\_integration\_time)}{(2 \times cam\_sensor\_cfg\_pixclk)} \quad \text{(EQ 4)}$$

The limits for the coarse integration time are defined by:

$$coarse\_integration\_time > 0 \quad \textit{(EQ 5)}$$

The limits for the fine integration time are defined by:

$$fine\_integration\_time\_min < fine\_integration\_time < (line\_length\_pck - fine\_integration\_time\_max\_margin) \quad \textit{(EQ 6)}$$

Fine Integration Time Limits

The limits for the fine_integration_time can be found from fine_integration_time_min and fine_integration_time_max_margin. Values for different mode combinations are shown in Table 14.

Table 14: fine_integration_time Limits

	No Row Binning	Row Binning
fine_integration_time_min	0x0336	0x06A4
fine_integration_time_max_margin	0x01BE	0x0322

fine_correction

For the fine_integration_time limits, the fine_correction constant will change with the binning mode. It is necessary to fine_correction when binning is enabled. The corresponding fine_correction values are shown in Table 15.

Table 15: fine_correction Values

	No Row Binning	Row Binning
fine_correction	0x00D4	0x01D9

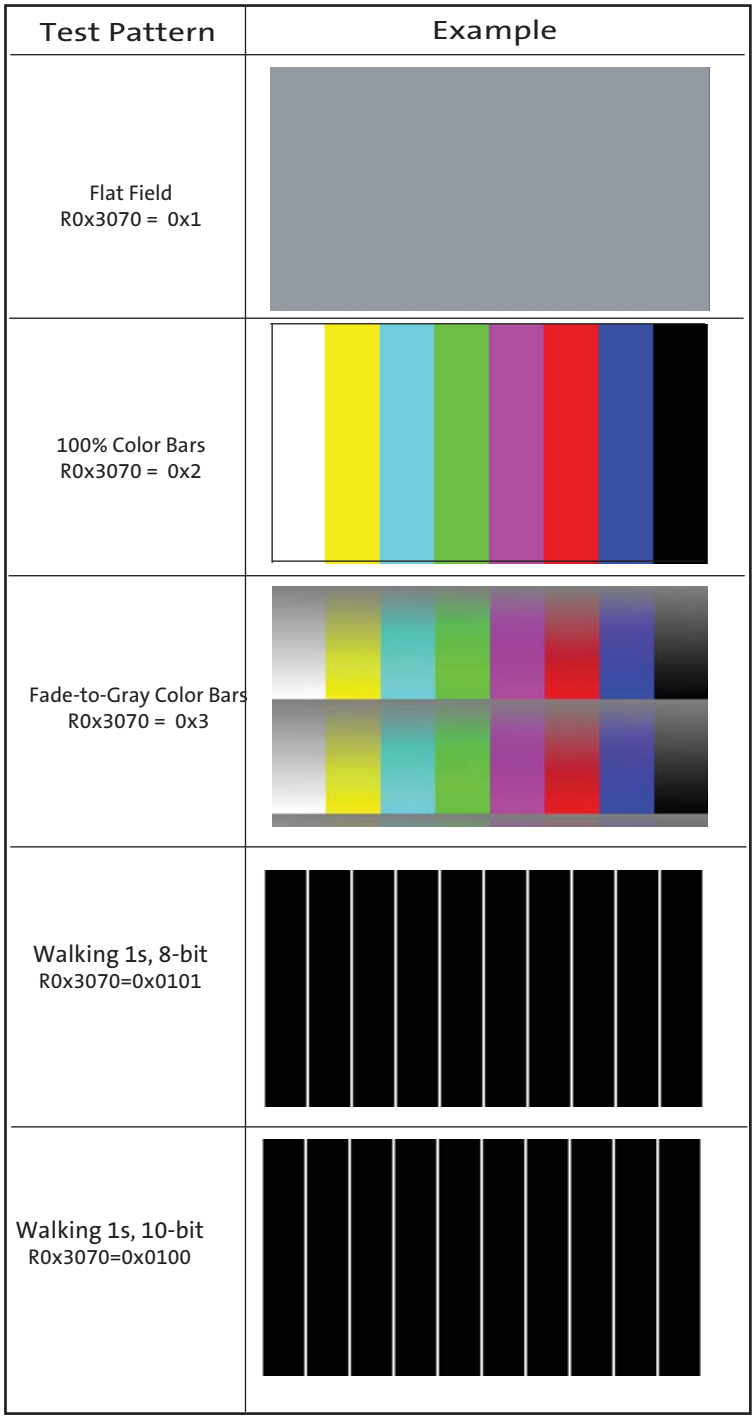
PLL

A PLL is provided to create the required PIXCLK from the input EXTCLK. The PLL is programmed through variable settings.

Digital Processing Chain

The AR0260 features a digital processing chain to provide further data path corrections and apply digital gain.

Figure 22: Color Bar Test Pattern



Digital Gain

Image stream processing starts with multiplication of all pixel values by a programmable digital gain. Independent color channel digital gain can be adjusted with registers.

Adaptive PGA (APGA)

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 AR0260 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.

In some cases, different illuminants can introduce different color shading response. The APGA feature on the AR0260 will compensate for the dependency of the lens shading of the illuminant. The AR0260 will allow for up to three different illuminants to be compensated for.

Image Scaling and Cropping

To ensure that the size of images output by the AR0260 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 including 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.

Camera Control and Auto Functions

General Purpose I/Os

The three general purpose I/Os (GPIOs) of the AR0260 can be configured in multiple ways. Each of the I/Os can be used for multiple purposes and can be programmed from the host. The GPIOs are powered by their own power supply.

Hardware Functions

Two-Wire Serial Interface

The two-wire serial interface bus enables read and write access to control and status registers and variables within the AR0260.

The interface protocol uses a master/slave model in which a master controls one or more slave devices. The AR0260 always operates in slave mode. The host (master) generates a clock (SCLK) that is an input to the AR0260 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 AR0260 are R0x090 (write address) and R0x091 (read address). If the SADDR signal is driven HIGH, then addresses used by the AR0260 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.

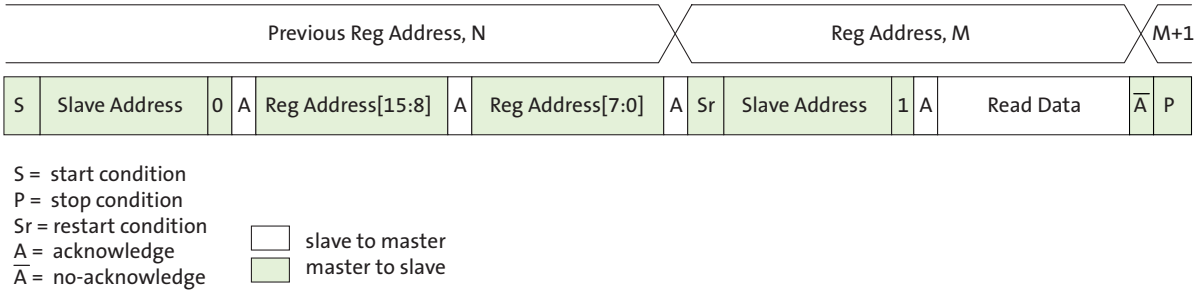
If the request was a read, the master sends the 8-bit write slave address/data direction byte and 16-bit register address, just as in the write request. The master then generates a (re)start condition and the 8-bit read slave address/data direction byte, and clocks out the register data, 8 bits at a 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 23) 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 23 shows how the internal register address maintained by the AR0260 is loaded and incremented as the sequence proceeds.

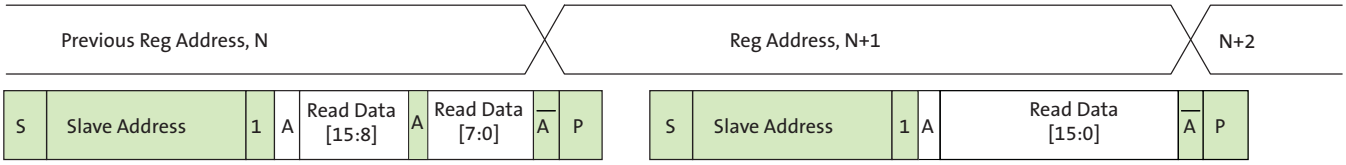
Figure 23: Single Read from Random Location



Single Read from Current Location

This sequence (Figure 24) performs a read using the current value of the AR0260 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.

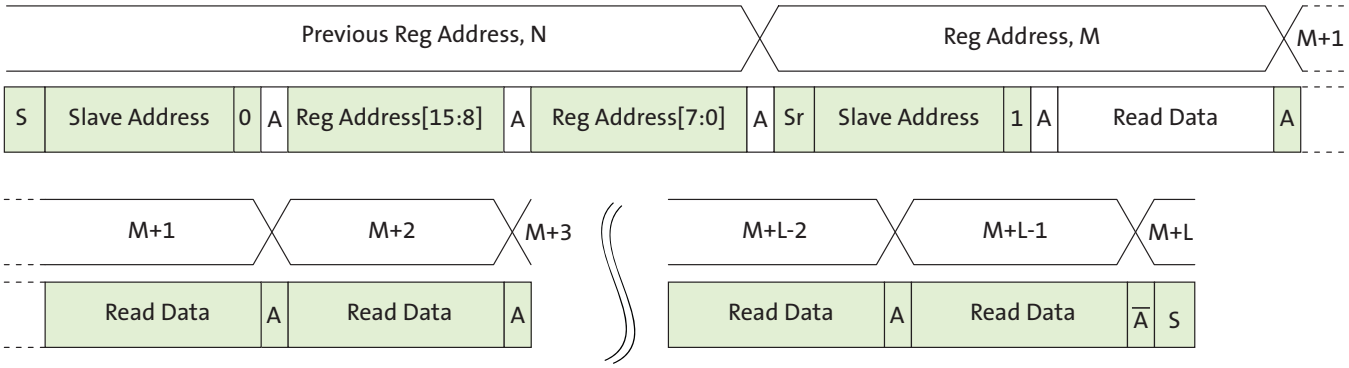
Figure 24: Single Read from Current Location



Sequential Read, Start from Random Location

This sequence (Figure 25) starts in the same way as the single read from random location (Figure 23). 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.

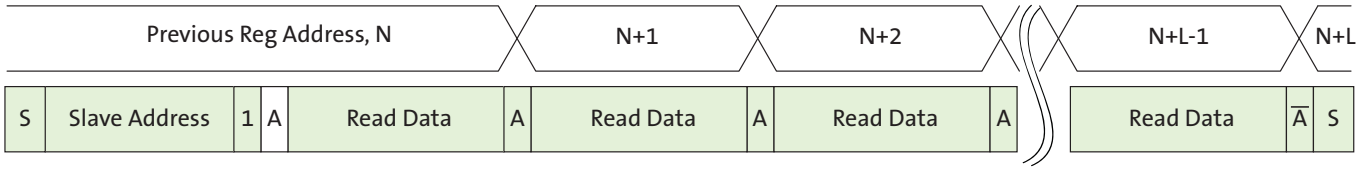
Figure 25: Sequential Read, Start from Random Location



Sequential Read, Start from Current Location

This sequence (Figure 26) starts in the same way as the single read from current location (Figure 24). 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.

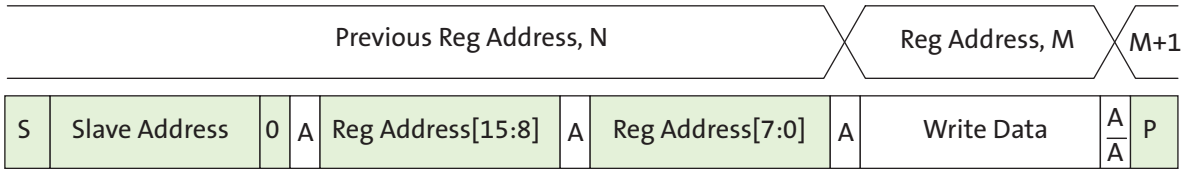
Figure 26: Sequential Read, Start from Current Location



Single Write to Random Location

This sequence (Figure 27) 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.

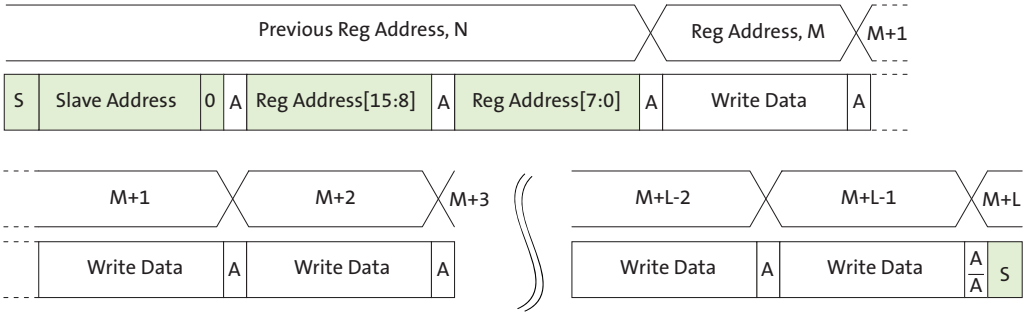
Figure 27: Single Write to Random Location



Sequential Write, Start at Random Location

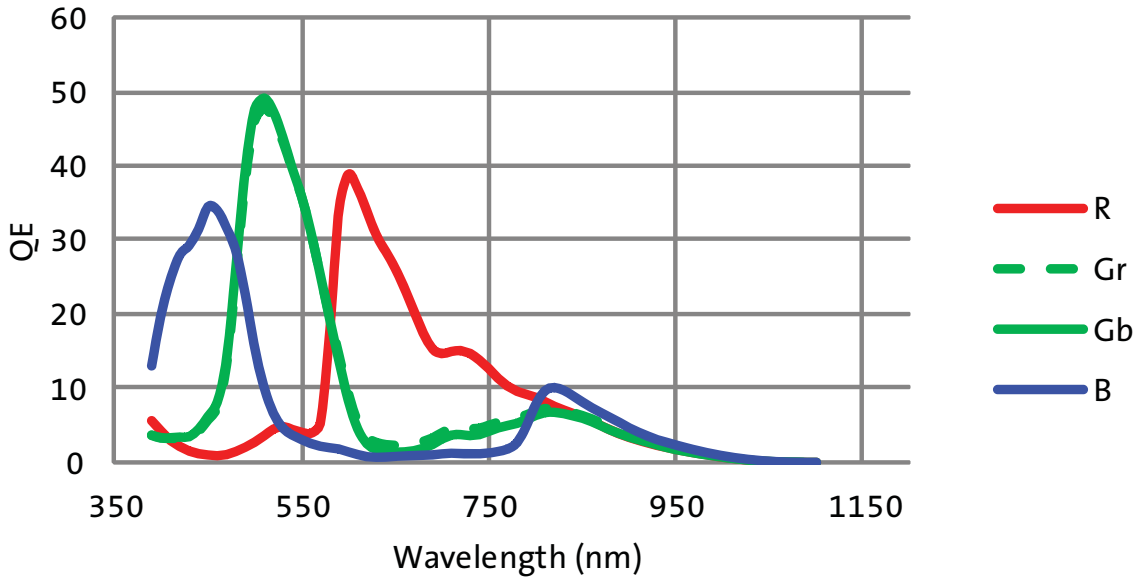
This sequence (Figure 28) starts in the same way as the single write to random location (Figure 27). Instead of generating a no-acknowledge bit after the first byte of data has been transferred, the master generates an acknowledge bit, and continues to perform byte writes until L bytes have been written. The write is terminated by the master generating a stop condition.

Figure 28: Sequential Write, Start at Random Location



Spectral Characteristics

Figure 29: Quantum Efficiency vs. Wavelength



Chief Ray Angle

Table 16: Chief Ray Angle Characteristics

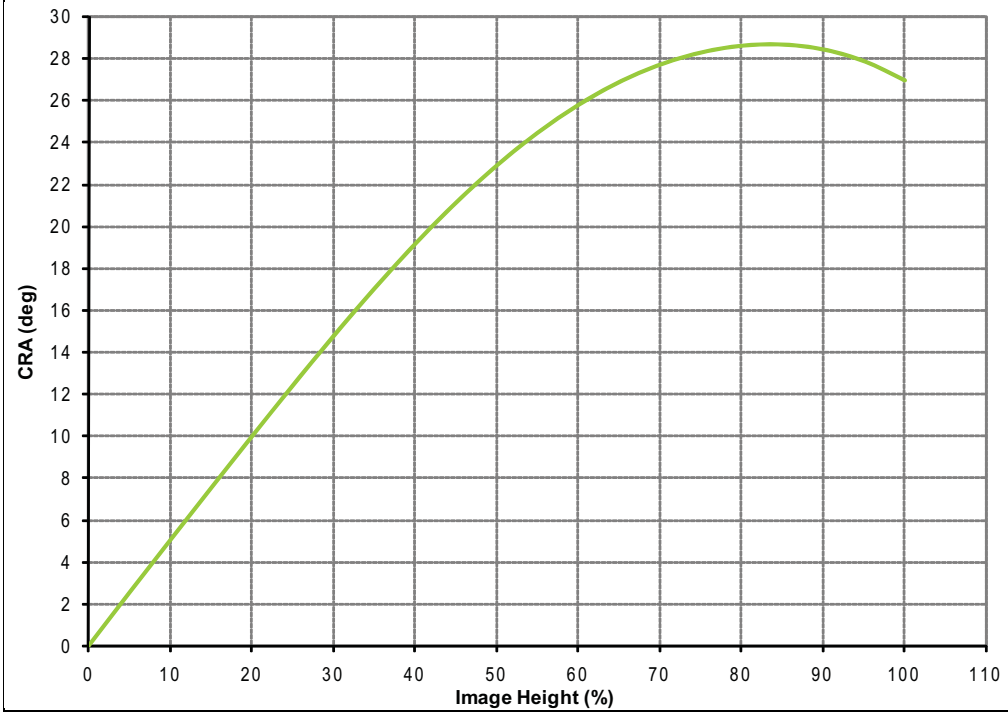


Image Height		CRA
(%)	(mm)	(deg)
0	0	0
5	0.077	2.54
10	0.154	5.04
15	0.231	7.52
20	0.308	9.98
25	0.386	12.40
30	0.463	14.76
35	0.540	17.02
40	0.617	19.15
45	0.694	21.12
50	0.771	22.89
55	0.848	24.45
60	0.925	25.78
65	1.002	26.87
70	1.079	27.70
75	1.157	28.28
80	1.234	28.61
85	1.311	28.66
90	1.388	28.43
95	1.465	27.88
100	1.542	26.95

CSP Package Details

AR0260 sensor is also available in chip scale package (CSP) and this section provides the relevant CSP package details necessary for the optical design of camera system.

Table 17: Package Dimension

Parameter	Symbol	Millimeters			Inches		
		Nominal	Min	Max	Nominal	Min	Max
Package Body Dimension X	A	6.005	5.97955	6.02955	0.23640	0.23542	0.23738
Package Body Dimension Y	B	4.158	4.13255	4.18255	0.16368	0.16270	0.16467
Package Height	C	0.710	0.655	0.765	0.02795	0.02579	0.03012
Cavity wall height	C4	0.0410	0.0370	0.0450	0.00161	0.00146	0.00177
Cavity wall + epoxy thickness glass to the wafer bonding top point)	C5	0.0435	0.0385	0.0485	0.00171	0.00152	0.00191
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	0.140	0.110	0.170	0.00551	0.00433	0.00669
Ball Diameter	D	0.280	0.250	0.310	0.01102	0.00984	0.01220
Total Ball Count		54					
Ball Count X axis	N1	9					
Ball Count Y axis	N2	6					
UBM	U	0.310	0.300	0.320	0.0122	0.01181	0.01260
Pins Pitch X axis	J1	0.620	0.610	0.630	0.02441	0.02402	0.02480
Pins Pitch Y axis	J2	0.620	0.610	0.630	0.02441	0.02402	0.02480
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
BGA ball center to chip center offset in X-direction	X1	0.000	-0.014	0.014	0.000	-0.001	0.001
BGA ball center to chip center offset in Y-direction	Y1	0.000	-0.014	0.014	0.000	-0.001	0.001
Edge to Ball Center Distance along X	S1	0.522	0.492	0.552	0.02056	0.01938	0.02174
Edge to Ball Center Distance along Y	S2	0.529	0.499	0.559	0.02082	0.01964	0.02200

Figure 30: CSP Mechanical Drawing

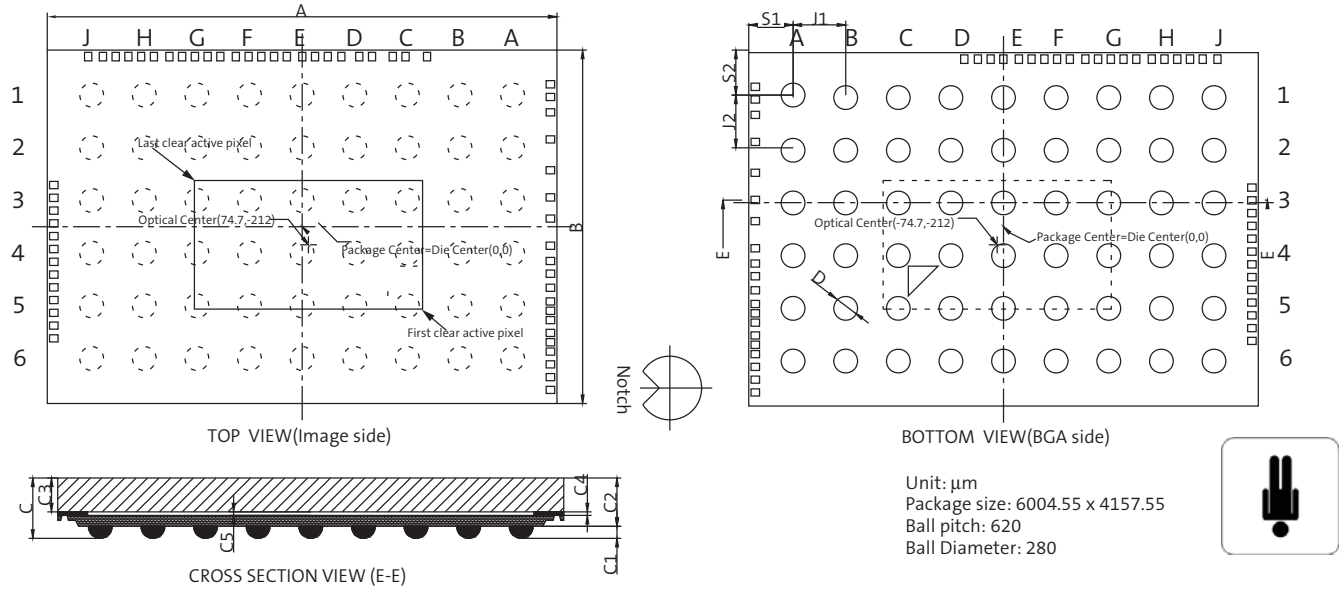


Table 18: Ball Matrix

	J	H	G	F	E	D	C	B	A
1	DGND	DOUT5	DGND	DOUT0	GPIO1	LINE_VALID	REG_IN0	DGND	DGND
2	DGND	DOUTT6	DOUT4	VDD_IO	GPIO0	DGND	REG_OUT	CLK_P	CLK_N
3	GPIO2	DOUT7	VDD_IO	DOUT3	DOUT1	FRAME_VALID	REG_FB	DATA2_N	DATA2_P
4	VDD_IO	PIXCLK	VDD	DOUT2	DGND	VDD_IO	VDD	DATA_N	DATA_P
5	SDATA	EXTCLK	DGND	DGND	SHUTDOWN	AGND	DGND	VDD_PHY	VPP
6	TRST_BAR	Reserved	SCLK	RESET_BAR	SADDR	AGND	AGND	VAA_PIX	VAA

Notes: 1. TRST_BAR connects to DGND for normal operation.
2. Reserved pins must be left floating.

Electrical Specifications

Caution **Stresses above those listed in Table 19 may cause permanent damage to the device.**

Table 19: Absolute Maximum Ratings

Symbol	Parameter	Rating		Unit
		Min	Max	
$V_{DD\_IO\_MAX}$	I/O digital voltage	–0.3	4.0	V
V_{AA_MAX}	Analog voltage	–0.3	4.0	V
$V_{AA\_PIX\_MAX}$	Analog pixel supply voltage	–0.3	4.0	V
REG_IN0_MAX	Digital supply voltage	–0.3	2.4	V
$V_{DD\_PHY\_MAX}$	PHY supply voltage	–0.3	2.4	V
V_{IN}	DC input voltage	–0.3	V_{DD_IO} + 0.3	V
I_{IN}	Transient input current (0.5 sec. duration)	–	150	mA
T_{OP}	Operating temperature (measure at junction)	–30	70	°C
T_{STG}[1]	Storage temperature	–40	85	°C

Notes: 1. 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 20: Operating Conditions

Symbol	Parameter	Min	Typ	Max	Units
V_{DD_IO}	I/O digital voltage	2.5	2.8	3.1	V
		1.7	1.8	1.95	V
V_{AA}	Analog voltage	2.5	2.8	3.1	V
V_{AA_PIX}	Pixel supply voltage	2.5	2.8	3.1	V
REG_IN0	Digital supply voltage	1.7	1.8	1.95	V
V_{DD_PHY}	PHY supply voltage	1.7	1.8	1.95	V
T_J	Operating temperature (at junction)	–30	55	70	°C

Table 21: DC Electrical Characteristics

Symbol	Parameter	Condition	Min	Max	Unit
V_{IH}	Input HIGH voltage		V_{DD_IO} * 0.7	V_{DD_IO} + 0.3	V
V_{IL}	Input LOW voltage		–0.3	V_{DD_IO} * 0.3	V
I_{IN}	Input leakage current	V_{IN} = 0V or V_{IN} = V_{DD_IO}		10	µA
V_{OH}	Output HIGH voltage	V_{DD_IO} = 1.8V, I_{OH} = 2mA	V_{DD_IO} – 0.3		V
V_{OL}	Output LOW voltage	V_{DD_IO} = 1.8V, I_{OH} = 2mA	–	0.4	V

Table 22: Operating Current Consumption (Parallel)
Default Setup Conditions: f_{EXTCLK}= 24 Mhz, f_{PIXCLK} = 96 Mhz, V_{AA} = V_{AA_PIX} = V_{DD_IO} = 2.8V, V_{DD_PHY} = REG_IN0 = 1.8V, Tj = 25°C, unless otherwise stated

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{AA}	Analog supply current	1080p Full resolution 30 fps	–	57	75	mA
		720p, 30 fps	–	53	70	mA
		VGA, 60 fps	–	54	71	mA
I_{AA_PIX}	Pixel supply current	1080p Full resolution 30 fps	–	1.7	2.9	mA
		720p, 30 fps	–	1.2	1.6	mA
		VGA, 60 fps	–	1.6	2.2	mA
I_{REG_IN0} and I_{DD_PHY}	Digital and PHY supply currents	1080p Full resolution 30 fps	–	52	68	mA
		720p, 30 fps	–	42	55	mA
		VGA, 60 fps	–	39	51	mA
I_{DD_IO}	I/O supply current (at 20pF load)	1080p Full resolution 30 fps	–	6.4	8.4	mA
		720p, 30 fps	–	6.3	8.2	mA
		VGA, 60 fps	–	7	9.1	mA
Total power consumption		1080p Full resolution 30 fps	–	277	360	mW
		720p, 30 fps	–	245	318	mW
		VGA, 60 fps	–	245	318	mW

Table 23: Operating Current Consumption (MIPI)
Default Setup Conditions: f_{EXTCLK}= 24 Mhz, f_{PIXCLK} = 96 Mhz, V_{AA} = V_{AA_PIX} = V_{DD_IO} = 2.8V, V_{DD_PHY} = REG_IN0 = 1.8V, Tj = 25°C, unless otherwise stated

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{AA}	Analog supply current	1080p Full resolution 30 fps	–	57	75	mA
		720p, 30 fps	–	53	70	mA
		VGA, 60 fps	–	54	71	mA
I_{AA_PIX}	Pixel supply current	1080p Full resolution 30 fps	–	1.7	2.3	mA
		720p, 30 fps	–	1.2	1.6	mA
		VGA, 60 fps	–	1.6	2.2	mA
I_{REG_IN0} and I_{DD_PHY}	Digital and PHY supply currents	1080p Full resolution 30 fps	–	58	75	mA
		720p, 30 fps	–	47	61	mA
		VGA, 60 fps	–	45	58	mA
I_{DD_IO}	I/O (non-MIPI) supply current	1080p Full resolution 30 fps	–	0.1	0.2	mA
		720p, 30 fps	–	0.1	0.2	mA
		VGA, 60 fps	–	0.1	0.2	mA
Total power consumption		1080p Full resolution 30 fps	–	270	351	mW
		720p, 30 fps	–	236	307	mW
		VGA, 60 fps	–	236	307	mW

Table 24: Non-Operating Current Consumption

Default Setup Conditions: f_{EXTCLK}= 24 Mhz, f_{PIXCLK} = 96 Mhz, V_{AA} = V_{AA_PIX} = V_{DD_IO} = 2.8V, V_{DD_PHY} = REG_IN0 = 1.8V, T**j** = 25°C, unless otherwise stated

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{AA}	Analog supply current	Hard Reset (with EXTCLK)	–	12	16	µA
		Hard Reset (without EXTCLK)	–	12	16	µA
		Shutdown(with EXTCLK)	–	1.0	1.5	µA
		Shutdown(without EXTCLK)	–	1.0	1.5	µA
I_{AA_PIX}	Pixel supply current	Hard Reset (with EXTCLK)	–	0	0	µA
		Hard Reset (without EXTCLK)	–	0	0	µA
		Shutdown(with EXTCLK)	–	0	0	µA
		Shutdown(without EXTCLK)	–	0	0	µA
I_{REG_IN0} and I_{DD_PHY}	Digital and PHY supply currents	Hard Reset (with EXTCLK)	–	2.8	3.6	mA
		Hard Reset (without EXTCLK)	–	2.3	3.0	mA
		Shutdown(with EXTCLK)	–	5.0	6.6	µA
		Shutdown(without EXTCLK)	–	5.0	6.6	µA
I_{DD_IO}	I/O supply current	Hard Reset (with EXTCLK)	–	0.2	0.3	mA
		Hard Reset (without EXTCLK)	–	0.2	0.2	mA
		Shutdown(with EXTCLK)	–	1.3	1.8	µA
		Shutdown(without EXTCLK)	–	1.3	1.8	µA
Total power consumption		Hard Reset (with EXTCLK)	–	5.5	7.2	mW
		Hard Reset (without EXTCLK)	–	4.6	6.0	mW
		Shutdown(with EXTCLK)	–	13	18	µW
		Shutdown(without EXTCLK)	–	13	18	µW

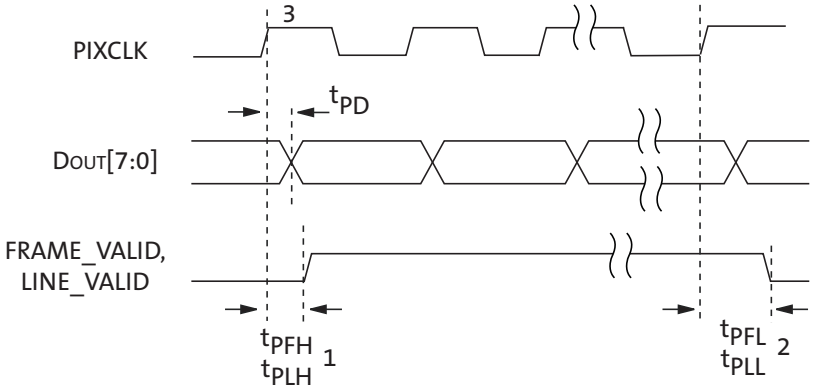
Table 25: AC Electrical Characteristics

EXTCLK = 24 MHz; REG_IN0 = V_{DD}_PHY = 1.8V; V_{DD}_IO = V_{AA} = V_{AA}_PIX = 2.8V; T_J = 25°C unless otherwise stated

Symbol	Parameter	Conditions	Min	Typ	Max	Unit	Notes
f_{EXTCLK}	External clock frequency		6		54	MHz	1
D_{EXTCLK}	External input clock duty cycle		40	50	60	%	
t_{JITTER}	External input clock jitter		–	500	–	ps	
f_{PIXCLK}	Pixel clock frequency		6		96	MHz	
$t_{RPIXCLK}$	Pixel clock rise time	C_{LOAD} = 25pf	–	4		ns	
$t_{FPIXCLK}$	Pixel clock fall time	C_{LOAD} = 25pf	–	4		ns	
t_{PD}	PIXCLK to data valid		–	21	0.1 * t_{PIXCLK}	ns	
t_{PFH}	PIXCLK to FV HIGH		–	21	0.1 * t_{PIXCLK}	ns	
t_{PFL}	PIXCLK to FV LOW		–	21	0.1 * t_{PIXCLK}	ns	
t_{PLH}	PIXCLK to LV HIGH		–	21	0.1 * t_{PIXCLK}	ns	
t_{PLL}	PIXCLK to LV LOW		–	21	0.1 * t_{PIXCLK}	ns	
PIXCLK slew rate							
	Programmable Slew = 7	V_{DD}_IO = 2.8V, C_{LOAD} = 25pf	–	0.83	–	V/ns	
		V_{DD}_IO = 1.8V, C_{LOAD} = 25pf	–	0.45	–	V/ns	
	Programmable Slew = 4	V_{DD}_IO = 2.8V, C_{LOAD} = 25pf	–	0.73	–	V/ns	
		V_{DD}_IO = 1.8V, C_{LOAD} = 25pf	–	0.42	–	V/ns	
	Programmable Slew = 0	V_{DD}_IO = 2.8V, C_{LOAD} = 25pf	–	0.38	–	V/ns	
		V_{DD}_IO = 1.8V, C_{LOAD} = 25pf	–	0.26	–	V/ns	
Output slew rate							
	Programmable Slew = 7	V_{DD}_IO = 2.8V, C_{LOAD} = 25pf	–	0.70	–	V/ns	
		V_{DD}_IO = 1.8V, C_{LOAD} = 25pf	–	0.39	–	V/ns	
	Programmable Slew = 4	V_{DD}_IO = 2.8V, C_{LOAD} = 25pf	–	0.67	–	V/ns	
		V_{DD}_IO = 1.8V, C_{LOAD} = 25pf	–	0.31	–	V/ns	
	Programmable Slew = 0	V_{DD}_IO = 2.8V, C_{LOAD} = 25pf	–	0.35	–	V/ns	
		V_{DD}_IO = 1.8V, C_{LOAD} = 25pf	–	0.16	–	V/ns	

Notes: 1. V_{IH}/V_{IL} restrictions apply.

Figure 31: Parallel Pixel Bus Timing Diagram



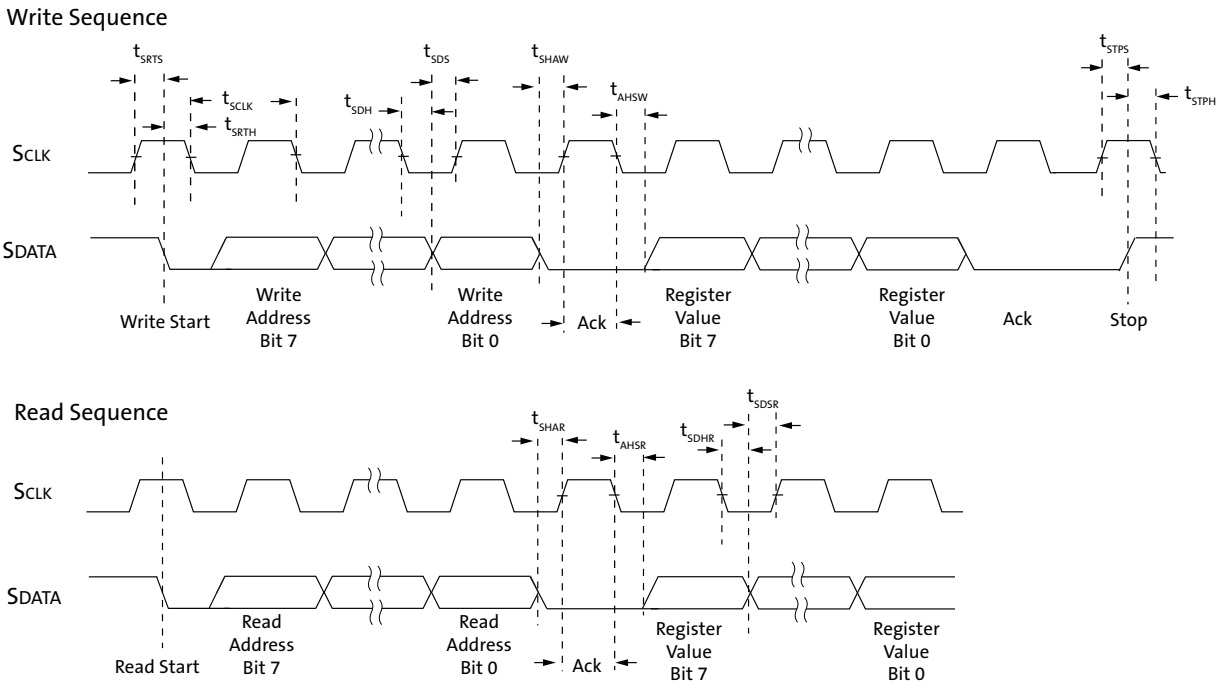
Notes:
1. FRAME_VALID leads LINE_VALID by 6 PIXCLKs.
2. FRAME_VALID trails LINE_VALID by 6 PIXCLKs.
3. DOUT[7:0], FRAME_VALID, and LINE_VALID are shown with respect to the rising edge of PIXCLK. This feature is programmable and DOUT[7:0], FRAME_VALID, and LINE_VALID can be synchronized to the falling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

Table 26: Two-Wire Serial Interface Timing Data

^{f}EXTCLK = 24 MHz; REG_IN0 = 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		–	65	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	–	–	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	–	–	ns
t_{SDSR}	SDATA setup	Master read from slave	350	–	–	ns

Figure 32: Two-Wire Serial Bus Timing Parameters



MIPI AC and DC Electrical Characteristics

Table 27: MIPI High-Speed Transmitter DC Characteristics

Symbol	Parameter	Min	Typ	Max	Unit
V_{OD}	HS transmit differential voltage	140	–	270	mV
VCMTX	HS transmit static common mode voltage	150	–	250	mV
ΔV_{OD}	VOD mismatch when output is Differential-1 or Differential-0	–	–	13	mV
Δ VCMTX(1,0)	VCMTX mismatch when output is Differential-1 or Differential-0	–	–	5	mV
V_{OHHS}	HS output HIGH voltage	–	–	360	mV
Z_{OS}	Single-ended output impedance	40	–	62.5	Ω
ΔZ_{OS}	Single-ended output impedance mismatch	–	–	17	%

Table 28: MIPI High-Speed Transmitter AC Characteristics

Symbol	Parameter	Min	Typ	Max	Unit
	Data bit rate	–	–	768	Mb/s
t_{rise}	20–80% rise time	150	–	500	ps
t_{fall}	20–80% fall time	150	–	500	ps

Table 29: MIPI Low-Power Transmitter DC Characteristics

Symbol	Parameter	Min	Typ	Max	Unit
V_{OL}	Thevenin output low level	–	–	55	mV
V_{OH}	Thevenin output high level	1.1	1.15	1.3	V
Z_{OLP}	Output impedance of LP transmitter	110	–	–	Ω

Table 30: MIPI Low-Power Transmitter AC Characteristics

Symbol	Parameter	Min	Typ	Max	Unit
t_{rise}	15–85% rise time	–	–	25	ns
t_{fall}	15–85% fall time	–	–	25	ns
Slew	Slew rate (C_{LOAD} 5–20pf)	–	–	200	mV/ns
Slew	Slew rate (C_{LOAD} 20–70pf)	–	–	150	mV/ns

MIPI Specification Reference

The AR0260 design and this documentation is based on the following reference documents:

- MIPI Alliance Standard for CSI-2 version 1.0
- MIPI Alliance Standard for D-PHY version 1.0

Revision History

Rev. G 4/13/15

- Updated “Ordering Information” on page 2

Rev. F 3/24/15

- Removed Confidential marking
- Updated Table of Contents to new format

Rev. E 12/2/14

- Updated to ON Semiconductor template

Rev. D 2/5/13

- Updated Table 2, “Available Part Numbers,” on page 1

Rev. C 9/7/12

- Updated to Production
- Updated Table 2, “Available Part Numbers,” on page 1
- Updated Table 3, “Pin Descriptions,” on page 9
- Updated Table 22, “Operating Current Consumption (Parallel),” on page 39
- Updated Table 23, “Operating Current Consumption (MIPI),” on page 39
- Updated Table 24, “Non-Operating Current Consumption,” on page 40
- Updated Table 25, “AC Electrical Characteristics,” on page 41
- Updated Table 32, “Two-Wire Serial Bus Timing Parameters,” on page 43
- Updated Table 27, “MIPI High-Speed Transmitter DC Characteristics,” on page 44
- Updated Table 28, “MIPI High-Speed Transmitter AC Characteristics,” on page 44
- Updated Table 29, “MIPI Low-Power Transmitter DC Characteristics,” on page 44
- Updated Table 30, “MIPI Low-Power Transmitter AC Characteristics,” on page 44

Rev. B 3/29/12

- Updated to Preliminary
- Updated Table 1: “Key Parameters,” on page 1
- Updated “Shutdown Mode” on page 14
- Removed “Soft Standby”
- Updated Table 11, “Recommended Sensor Analog Gain Tables,” on page 23
- Added “fine_correction” on page 26
- Removed “One-Time Programmable Memory”
- Updated Table 20, “Operating Conditions,” on page 38
- Updated Table 21, “DC Electrical Characteristics,” on page 38
- Updated Table 22, “Operating Current Consumption (Parallel),” on page 39
- Updated Table 23, “Operating Current Consumption (MIPI),” on page 39

Rev. A 3/6/12

- Initial release