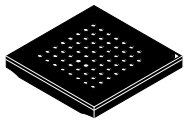
AR0141CS

1/4-inch Digital Image Sensor

ON Semiconductor®

www.onsemi.com



IBGA63 9 × 9
CASE 503AH

ORDERING INFORMATION

See detailed ordering and shipping information on page 2 of this data sheet.

Description

The ON Semiconductor AR0141CS is a 1/4−inch CMOS digital image sensor with an active−pixel array of 1280 H x 800 V. It captures images in linear mode, with a rolling−shutter readout. It includes sophisticated camera functions such as in−pixel binning, windowing and both video and single frame modes. It is designed for low light scene performance. It is programmable through a simple two−wire serial interface. The AR0141CS produces extraordinarily clear, sharp digital pictures, and its ability to capture both continuous video and single frames makes it the perfect choice for a wide range of applications, including surveillance and HD video.

Table 1. KEY PERFORMANCE PARAMETERS

Parameter	Typical Value
Optical Format	1/4-inch
Active Pixels	1280 (H) × 800 (V) (Entire Array)
Pixel Size	3.0 μm × 3.0 μm
Color Filter Array	RGB Bayer, Monochrome, RGB−IR
Shutter Type	Electronic Rolling Shutter and GRR
Input Clock Range	6 − 50 MHz
Output Clock Maximum	148.5 Mp/s (4−lane HiSPi) 74.25 Mp/s (Parallel)
Output Serial Parallel	 HiSPi, 12−bit 10-, 12-bit
Frame Rate 720p	 60 fps
Responsivity	4.0 V/lux−sec
SNR_{MAX}	41 dB
Maximum Dynamic Range	Up to 79 dB
Supply Voltage I/O Digital Analog HiSPi	 1.8 or 2.8 V 1.8 V 2.8 V 0.3 V − 0.6 V, 1.7 V − 1.9 V
Power Consumption (Typical)	326 mW (Linear Mode 1280 x 720 60 fps)
Operating Temperature (Ambient) T_A	−30°C to +70°C
Package Options	9 x 9 mm 63−ball iBGA

Features

- Superior Low-light Performance
- Latest 3.0 μm Pixel with ON Semiconductor DR−Pix Technology
- Linear Range Capture
- 1.0 Mp and 720p (16:9) Images
- Support for External Mechanical Shutter
- Support for External LED or Xenon Flash
- On−chip Phase−locked Loop (PLL) Oscillator
- Integrated Position−based Color and Lens Shading Correction
- Slave Mode for Precise Frame−rate Control
- Stereo/3D Camera Support
- Statistics Engine
- Data Interfaces: Four−lane Serial High−speed Pixel Interface (HiSPi) Differential signaling (SLVS and HiVCM), or Parallel
- Auto Black Level Calibration
- High−speed Context Switching
- Temperature Sensor

Applications

- Video Surveillance
- Scanning
- Industrial
- Stereo Vision
- 720p60 Video Applications

ORDERING INFORMATION

Table 2. AVAILABLE PART NUMBERS

Part Number	Product Description	Orderable Product Attribute Description
AR0141CS2C00SUEA0−DP	Color iBGA	Dry Pack with Protective Film
AR0141CS2C00SUEA0−DR	Color iBGA	Dry Pack without Protective Film
AR0141CS2C00SUEAD3−GEVK	Color iBGA Demo3 Kit	
AR0141CS2C00SUEAH−GEVB	Color iBGA Headboard	
AR0141CS2M00SUEA0 − TPBR	Mono iBGA	Tape and Reel with Protective Film
AR0141CS2M00SUEA0 − DPBR	Mono iBGA	Dry Pack with Protective Film
AR0141CS2M00SUEAD3−GEVK	Mono iBGA Demo3 Kit	
AR0141CS2M00SUEAH−GEVB	Mono iBGA Headboard	
AR0141IRSH00SUEA0−DR	RGB−IR, iBGA, Production	Dry Pack without Protective Film
AR0141IRSH00SUEA0D3−GEVK	RGB−IR, Demo3 Kit	
AR0141IRSH00SUEA0H3−GEVB	RGB−IR, Head Board	
AR0141CSSM21SUEA0−TPBR	Mono, iBGA, 21 Deg Shift	Engineering Sample

See the ON Semiconductor Device Nomenclature document (TND310/D) for a full description of the naming convention used for image sensors. For reference documentation, including information on evaluation kits, please visit our web site at www.onsemi.com.

GENERAL DESCRIPTION

The ON Semiconductor AR0141CS can be operated in its default mode or programmed for frame size, exposure, gain, and other parameters. The default mode output is a 720p−resolution image at 60 frames per second (fps). In linear mode, it outputs 12−bit raw data, using either the parallel or serial (HiSPi) output ports. The device may be operated in video (master) mode or in single frame trigger mode.

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

The AR0141CS includes additional features to allow application−specific tuning: windowing and offset, auto black level correction, and on−board temperature sensor. Optional register information and histogram statistic information can be embedded in the first and last 2 lines of the image frame.

FUNCTIONAL OVERVIEW

The AR0141CS is a progressive−scan sensor that generates a stream of pixel data at a constant frame rate. It uses an on−chip, phase−locked loop (PLL) that can be optionally enabled to generate all internal clocks from a single master input clock running between 6 and 50 MHz. The maximum output pixel rate is 148.5 Mp/s, corresponding to a clock rate of 74.25 MHz. Figure 1 shows a block diagram of the sensor.

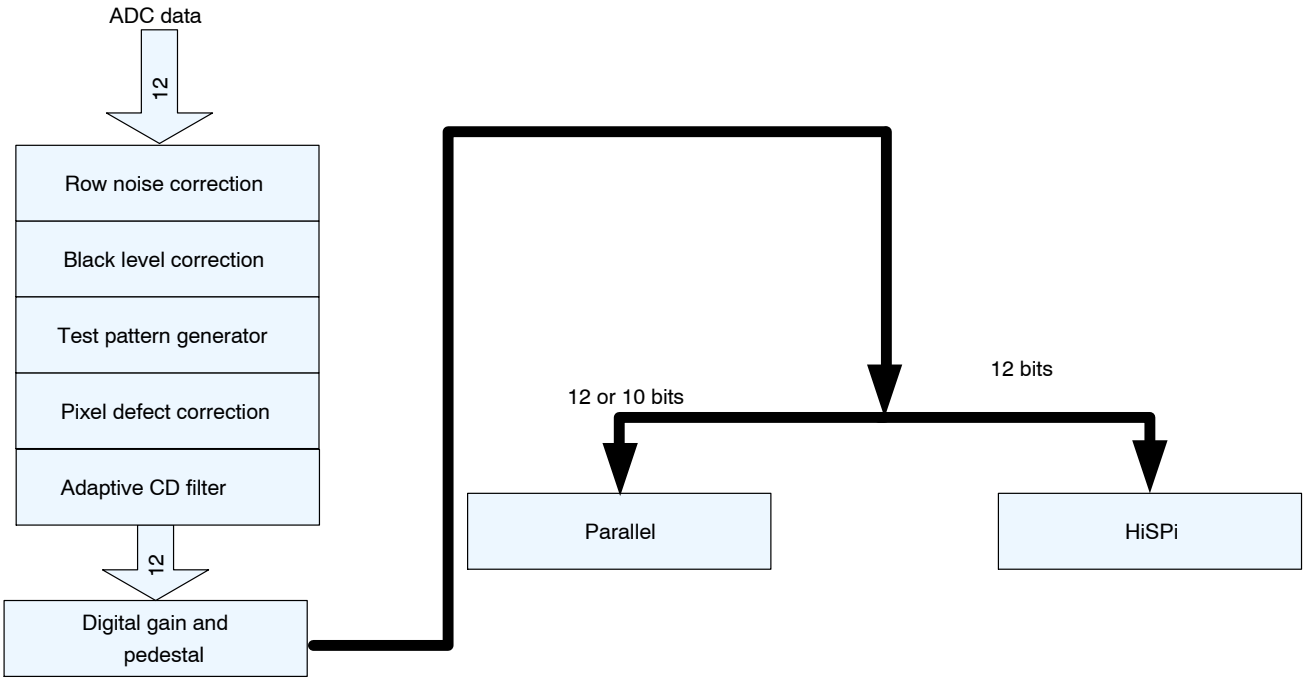
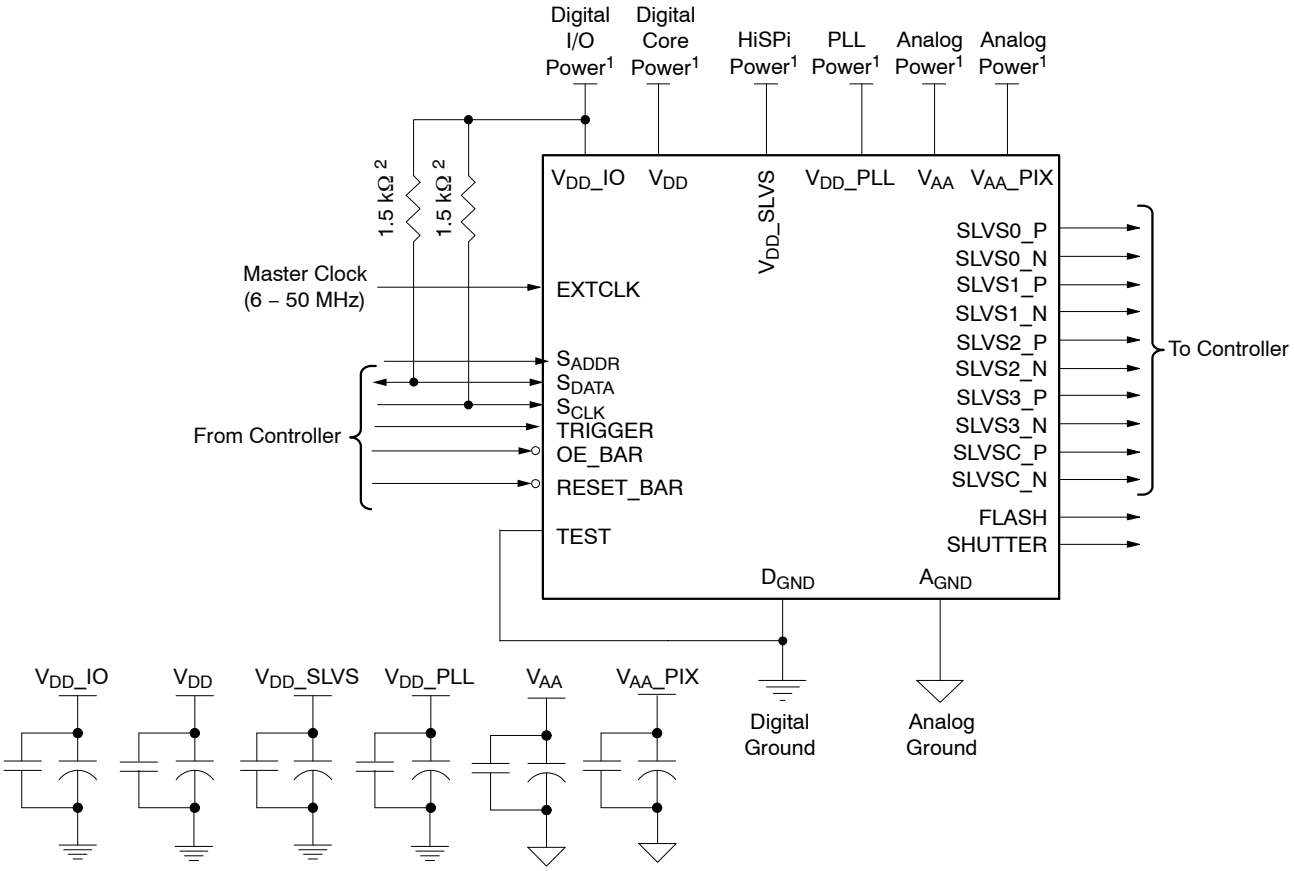


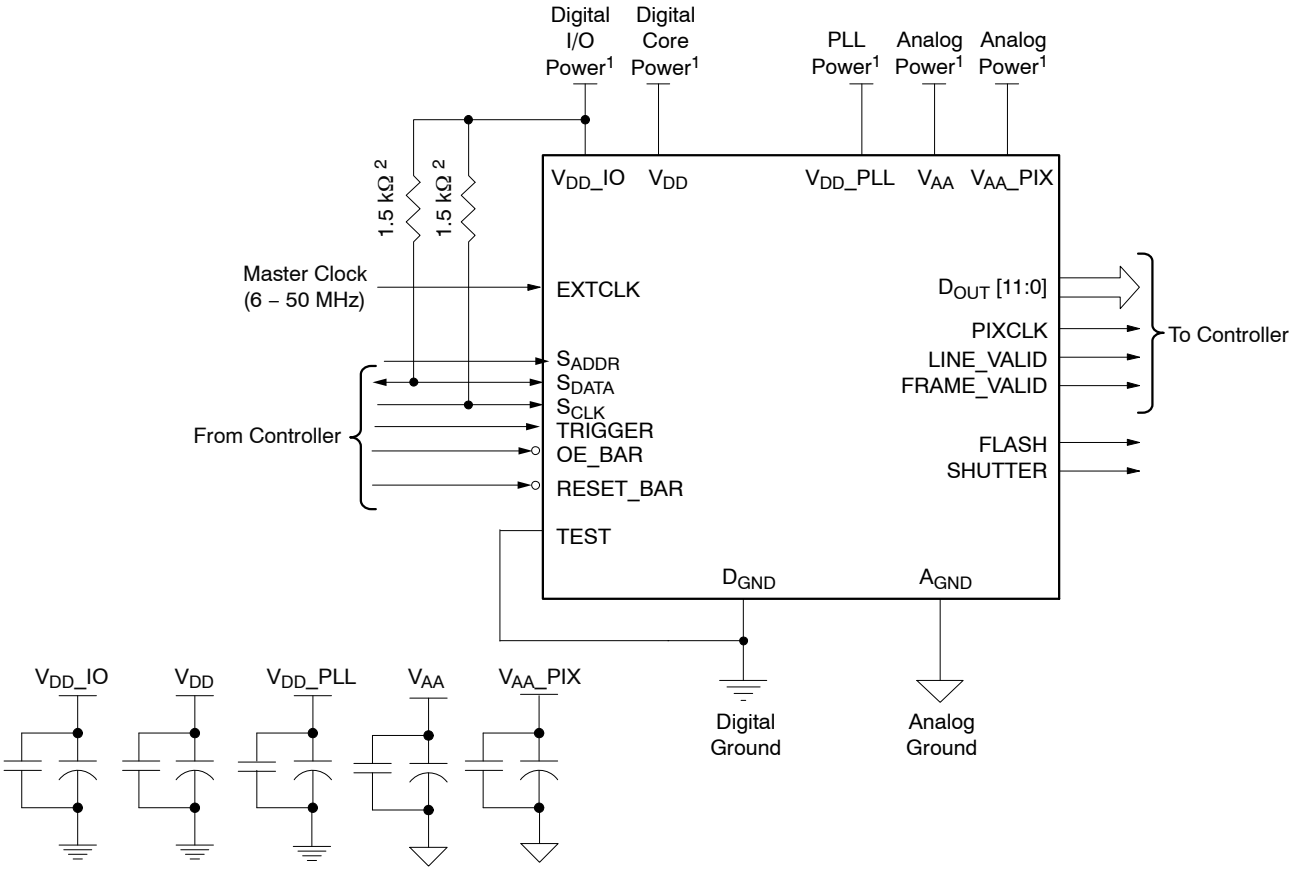
Figure 1. Block Diagram

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 1.1 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 12−bit value for each pixel in the array. The ADC output passes through a digital processing signal chain (which provides further data path corrections and applies digital gain).



Notes:
1. All power supplies must be adequately decoupled.
2. ON Semiconductor recommends a resistor value of 1.5 kΩ, but a greater value may be used for slower two−wire speed.
3. The parallel interface output pads can be left unconnected if the serial output interface is used.
4. ON Semiconductor recommends that 0.1 μF and 10 μF decoupling capacitors for each power supply are mounted as close as possible to the pad. Actual values and results may vary depending on layout and design considerations. Check the AR0141CS demo headboard schematics for circuit recommendations.
5. ON Semiconductor recommends that analog power planes are placed in a manner such that coupling with the digital power planes is minimized.
6. I/O signals voltage must be configured to match V_{DD}_IO voltage to minimize any leakage currents.

Figure 2. Typical Configuration: Serial Four−Lane HiSPi Interface



Notes:
1. All power supplies must be adequately decoupled.
2. ON Semiconductor recommends a resistor value of 1.5 kΩ, but a greater value may be used for slower two−wire speed.
3. The serial interface output pads and V_{DD}_SLVS can be left unconnected if the parallel output interface is used.
4. ON Semiconductor recommends that 0.1 μF and 10 μF decoupling capacitors for each power supply are mounted as close as possible to the pad. Actual values and results may vary depending on layout and design considerations. Check the AR0141CS demo headboard schematics for circuit recommendations.
5. ON Semiconductor recommends that analog power planes are placed in a manner such that coupling with the digital power planes is minimized.
6. I/O signals voltage must be configured to match V_{DD}_IO voltage to minimize any leakage current.
7. The EXTCLK input is limited to 6−50 MHz.

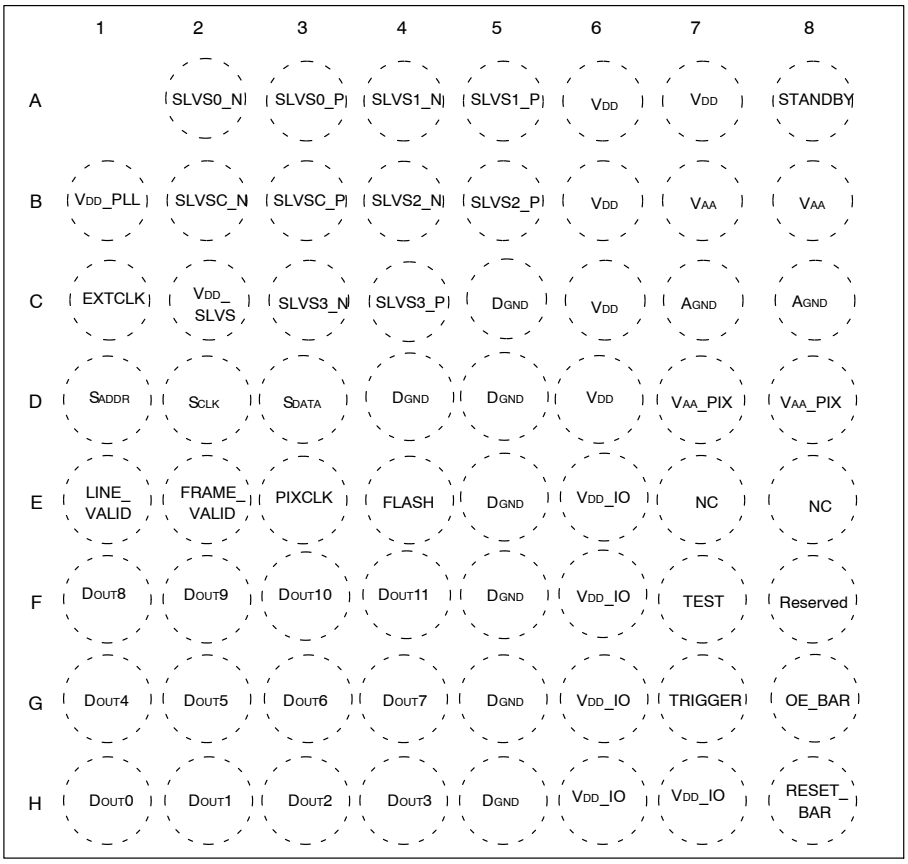
Figure 3. Typical Configuration: Parallel Pixel Data Interface

Table 3. BALL DESCRIPTIONS, 9 X 9 MM, 63−BALL iBGA

Name	iBGA Pin	Type	Description
SLVS0_N	A2	Output	HiSPi serial data, lane 0, differential N
SLVS0_P	A3	Output	HiSPi serial data, lane 0, differential P
SLVS1_N	A4	Output	HiSPi serial data, lane 1, differential N
SLVS1_P	A5	Output	HiSPi serial data, lane 1, differential P
STANDBY	A8	Input	Standby (active high)
V_{DD}_PLL	B1	Power	PLL power
SLVSC_N	B2	Output	HiSPi serial DDR clock differential N
SLVSC_P	B3	Output	HiSPi serial DDR clock differential P
SLVS2_N	B4	Output	HiSPi serial data, lane 2, differential N
SLVS2_P	B5	Output	HiSPi serial data, lane 2, differential P

Table 3. BALL DESCRIPTIONS, 9 X 9 MM, 63−BALL iBGA

Name	iBGA Pin	Type	Description
V_{AA}	B7, B8	Power	Analog power
EXTCLK	C1	Input	External input clock
V_{DD}_SLVS	C2	Power	0.3 V – 0.6 V or 1.7 V – 1.9 V port to HiSPi Output Driver. Set the High_VCM (R0x306E[9]) bit to 1 when configuring V_{DD}_SLVS to 1.7 V – 1.9 V
SLVS3_N	C3	Output	HiSPi serial data, lane 3, differential N
SLVS3_P	C4	Output	HiSPi serial data, lane 3, differential P
DGND	C5, D4, D5, E5, F5, G5, H5	Power	Digital ground
VDD	A6, A7, B6, C6, D6	Power	Digital power
AGND	C7, C8	Power	Analog ground
SADDR	D1	Input	Two−Wire Serial address select. 0: 0x20, 1: 0x30
SCLK	D2	Input	Two−Wire Serial clock input
SDATA	D3	I/O	Two−Wire Serial data I/O
VAA_PIX	D7, D8	Power	Pixel power
LINE_VALID	E1	Output	Asserted when D_{OUT} line data is valid
FRAME_VALID	E2	Output	Asserted when D_{OUT} frame data is valid
PIXCLK	E3	Output	Pixel clock out. D_{OUT} is valid on rising edge of this clock
V_{DD}_IO	E6, F6, G6, H6, H7	Power	I/O supply power
D_{OUT}8	F1	Output	Parallel pixel data output
D_{OUT}9	F2	Output	Parallel pixel data output
D_{OUT}10	F3	Output	Parallel pixel data output
D_{OUT}11	F4	Output	Parallel pixel data output (MSB)
TEST	F7	Input.	Manufacturing test enable pin (connect to D_{GND})
D_{OUT}4	G1	Output	Parallel pixel data output
D_{OUT}5	G2	Output	Parallel pixel data output
D_{OUT}6	G3	Output	Parallel pixel data output
D_{OUT}7	G4	Output	Parallel pixel data output
TRIGGER	G7	Input	Exposure synchronization input
OE_BAR	G8	Input	Output enable (active LOW)
D_{OUT}0	H1	Output	Parallel pixel data output (LSB)
D_{OUT}1	H2	Output	Parallel pixel data output
D_{OUT}2	H3	Output	Parallel pixel data output
D_{OUT}3	H4	Output	Parallel pixel data output
RESET_BAR	H8	Input	Asynchronous reset (active LOW). All settings are restored to factory default
NC	E8		No connection
FLASH	E4	Output	Flash control output
NC	E7		No connection
Reserved	F8		Reserved



Top View

(Ball Down)

Note: No ball on A1 pin, 63 balls in total in actual iBGA package.

Figure 4. 9 x 9 mm 63−Ball iBGA Package

PIXEL DATA FORMAT

Pixel Array Structure

The AR0141CS pixel array consists of 1280 columns by 800 rows of optically active pixels. While the sensor's format is 1344 × 848, 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 pixel adjustment is always performed for monochrome or color versions. The active area is surrounded with optically transparent dummy pixels to improve image uniformity within the active area. Not all dummy pixels or barrier pixels can be read out.

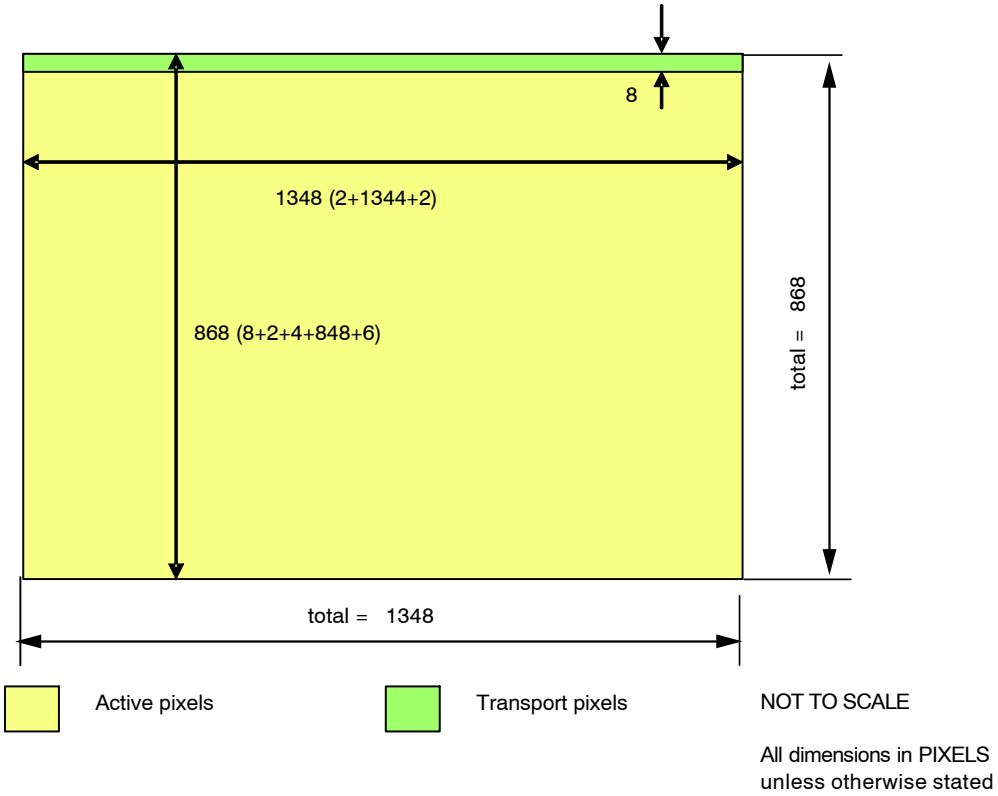


Figure 5. Pixel Array Description

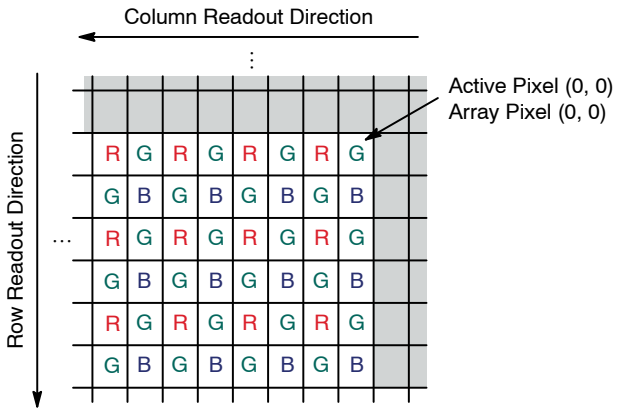


Figure 6. RGB Pixel Color Pattern Detail (Top Right Corner) – AR0141CS

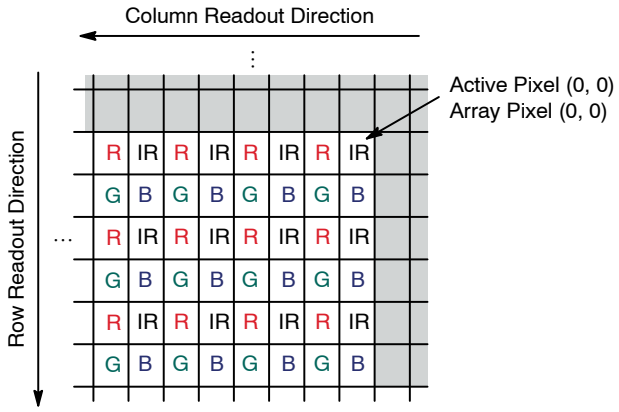


Figure 7. RGB−IR Pixel Color Pattern Detail (Top Right Corner) − AR0141IR

Differentiation from AR0141CS

The AR0141IR can be electrically differentiated from the AR0141CS by reading bits 11:9 in R0x31FA. The AR0141IR contains a unique value of 4 in these bits. It is necessary to set R0x301A[5] = 1 prior to reading R0x31FA[11:9].

Default Readout Order

By convention, the sensor core pixel array is shown with pixel (0,0) in the top right corner (see Figure 6). This reflects the actual layout of the array on the die. Also, the first pixel data read out of the sensor in default condition is that of pixel (0, 0).

When the sensor is imaging, the active surface of the sensor faces the scene as shown in Figure 8. When the image is read out of the sensor, it is read one row at a time, with the rows and columns sequenced as shown in Figure 8.

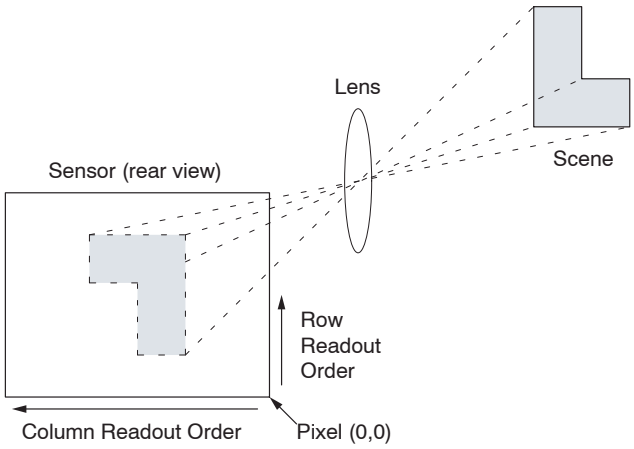


Figure 8. Imaging a Scene

PIXEL OUTPUT INTERFACES

Parallel Interface

The parallel pixel data interface uses these output−only signals:

- FRAME_VALID
- LINE_VALID
- PIXCLK
- D_{OUT}[11:0]

The parallel pixel data interface is disabled by default at power up and after reset. It can be enabled by programming R0x301A. Table 5 shows the recommended settings.

When the parallel pixel data interface is in use, the serial data output signals can be left unconnected. Set reset_register [bit 12 (R0x301A[12] = 1)] to disable the serializer while in parallel output mode.

Output Enable Control

When the parallel pixel data interface is enabled, its signals can be switched asynchronously between the driven and High−Z under pin or register control, as shown in Table 4.

Table 4. OUTPUT ENABLE CONTROL

OE_BAR Pin	Drive Pins R0x301A[6]	Description
Disabled	0	Interface High−Z
Disabled	1	Interface driven
1	0	Interface High−Z
X	1	Interface driven
0	X	Interface driven

Configuration of the Pixel Data Interface

Fields in R0x301A are used to configure the operation of the pixel data interface. The supported combinations are shown in Table 5.

Table 5. CONFIGURATION OF THE PIXEL DATA INTERFACE

Serializer Disable R0x301 A[12]	Parallel Enable R0x301 A[7]	Description
0	0	Power up default. Serial pixel data interface and its clocks are enabled. Transitions to soft standby are synchronized to the end of frames on the serial pixel data interface.
1	1	Parallel pixel data interface, sensor core data output. Serial pixel data interface and its clocks disabled to save power. Transitions to soft standby are synchronized to the end of frames in the parallel pixel data interface.

High Speed Serial Pixel Data Interface

The High Speed Serial Pixel (HiSPi) interface uses four data lanes and one clock as output.

- SLVSC_P
- SLVSC_N
- SLVS0_P
- SLVS0_N
- SLVS1_P
- SLVS1_N
- SLVS2_P
- SLVS2_N
- SLVS3_P
- SLVS3_N

The HiSPi interface supports three protocols, Streaming−S, Streaming−SP, and Packetized SP. The streaming protocols conform to a standard video application where each line of active or intra−frame blanking provided by the sensor is transmitted at the same length. The Packetized SP protocol will transmit only the active data ignoring line−to−line and frame−to−frame blanking data.

These protocols are further described in the High−Speed Serial Pixel (HiSPi) Interface Protocol Specification V1.50.00.

The HiSPi interface building block is a unidirectional differential serial interface with four data and one double data rate (DDR) clock lanes. One clock for every four serial data lanes is provided for phase alignment across multiple lanes. Figure 9 shows the configuration between the HiSPi transmitter and the receiver.

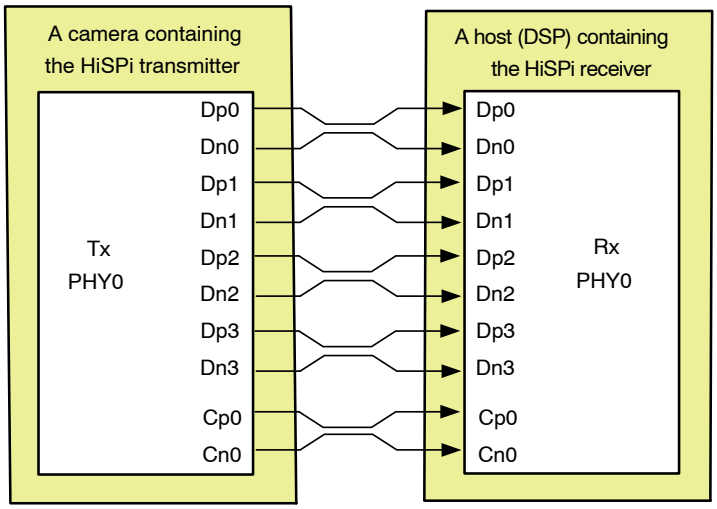


Figure 9. HiSPi Transmitter and Receiver Interface Block Diagram

HiSPi Physical Layer

The HiSPi physical layer has four data lanes and an associated clock lane. Depending on the sensor operating mode and data rate, it can be configured to use either 2, 3, or 4 lanes. The PHY will serialize a 12− to 20−bit data word and transmit each bit of data centered on a rising edge of the clock, the second on the following falling edge of clock. Figure 10 shows bit transmission. In this example, the word is transmitted in order of MSB to LSB. The receiver latches data at the rising and falling edge of the clock.

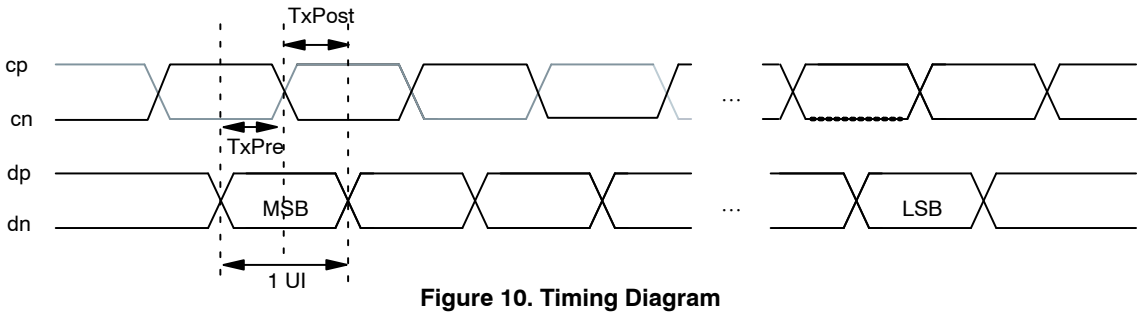


Figure 10. Timing Diagram

DLL Timing Adjustment

The AR0141CS includes a DLL to compensate for differences in group delay for each data lane. The DLL is connected to the clock lane and each data lane, which acts as a control master for the output delay buffers. Once the DLL has gained phase lock, each lane can be delayed in 1/8 unit interval (UI) steps. This additional delay allows the user to increase the setup or hold time at the receiver circuits and can be used to compensate for skew introduced in PCB design.

Delay compensation may be set for clock and/or data lines in the hispi_timing register R0x31C0. If the DLL timing adjustment is not required, the data and clock lane delay settings should be set to a default code of 0x000 to reduce jitter, skew, and power dissipation.

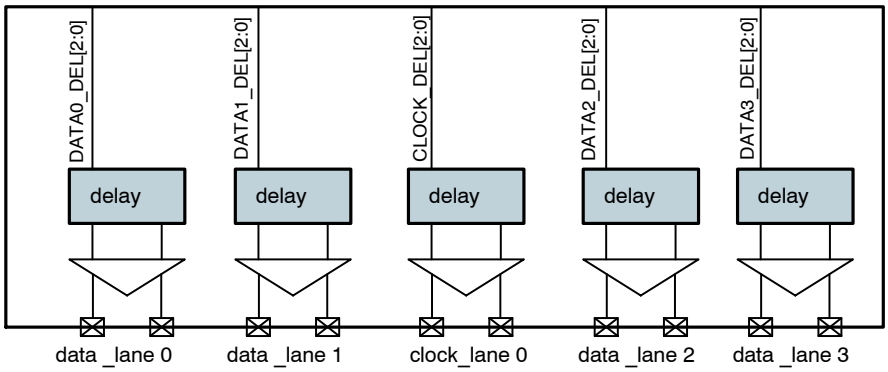


Figure 11. Block Diagram of DLL Timing Adjustments

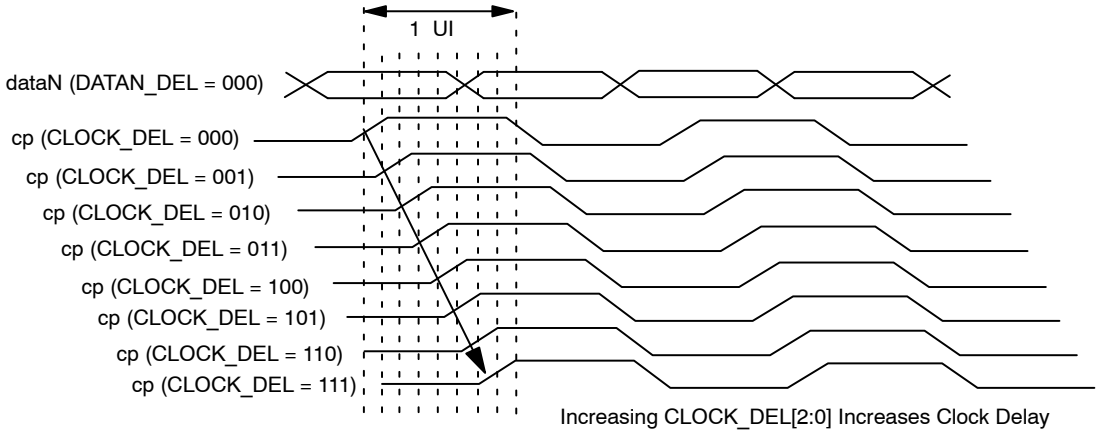


Figure 12. Delaying the Clock with Respect to Data

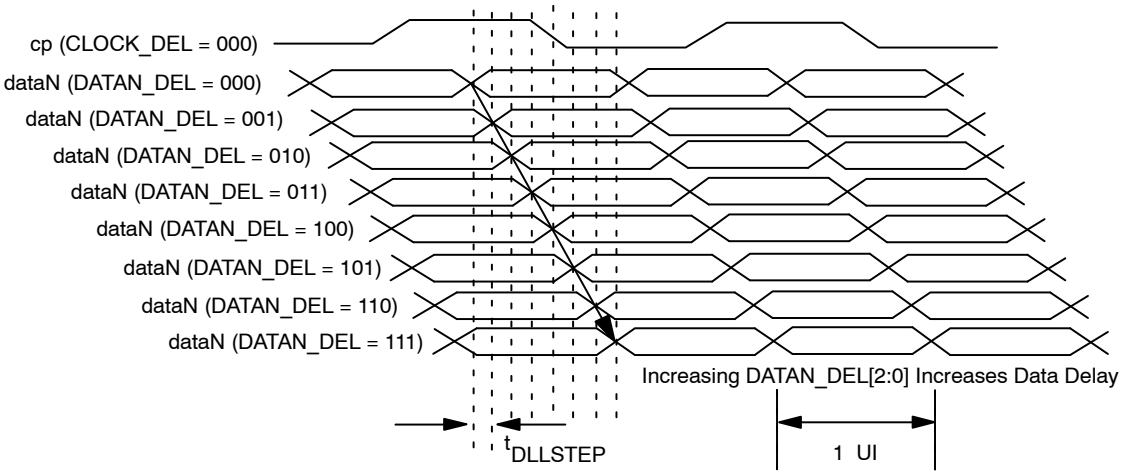


Figure 13. Delaying Data with Respect to the Clock

HiSPi Protocol Layer

The HiSPi protocol is described in the HiSPi Protocol Specification document.

Serial Configuration

The serial format should be configured using R0x31AC. Refer to the AR0141CS Register Reference document for more detail regarding this register.

The serial_format register (R0x31AE) controls which serial format is in use when the serial interface is enabled (reset_register[12] = 0). The following serial formats are supported:

- 0x0304 – Sensor supports quad−lane HiSPi operation
- 0x0302 – Sensor supports dual−lane HiSPi operation

PIXEL SENSITIVITY

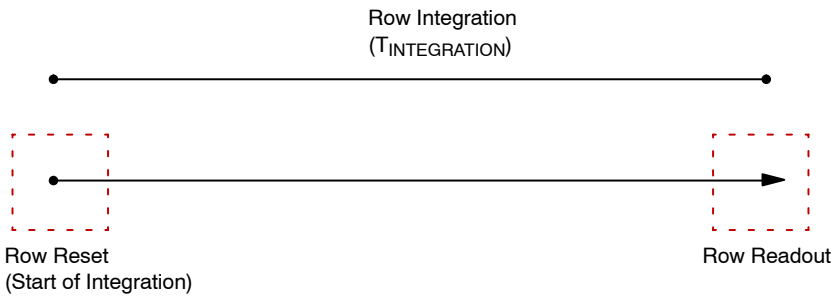


Figure 14. Integration Control in ERS Readout

A pixel's integration time is defined by the number of clock periods between a row's reset and read operation. Both the read followed by the reset operations occur within a row period (T_{ROW}) where the read and reset may be applied to different rows. The read and reset operations will be applied to the rows of the pixel array in a consecutive order.

The coarse integration time is defined by the number of row periods (T_{ROW}) between a row's reset and the row read. The row period is defined as the time between row read operations (see Sensor Frame Rate).

$$T_{COARSE} = T_{ROW} \times \text{coarse\_integration\_time} \quad \text{(eq. 1)}$$

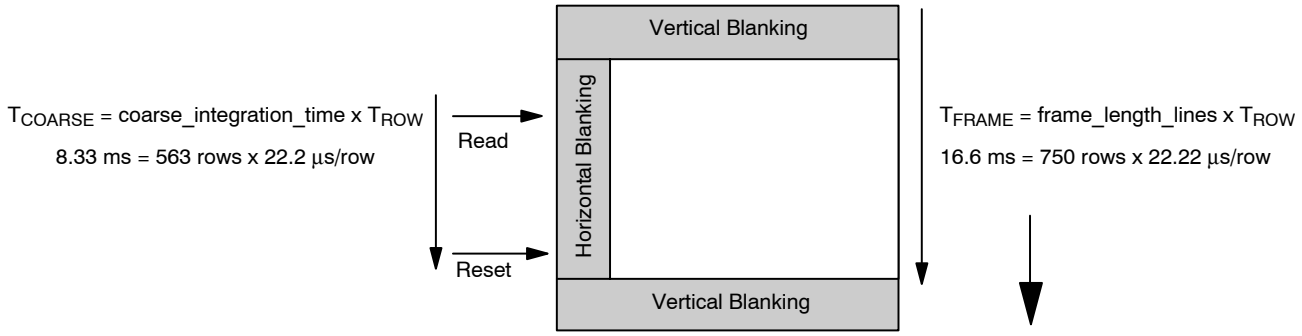


Figure 15. Example of 8.33 ms Integration in 16.6 ms Frame

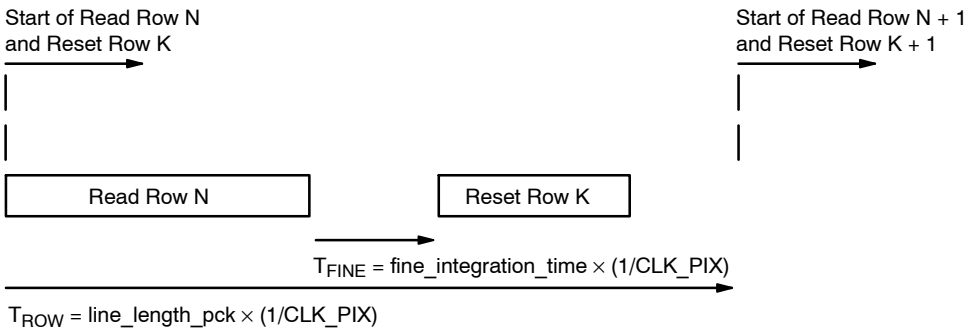


Figure 16. Row Read and Row Reset Showing Fine Integration

$$T_{FINE} = \text{fine\_integration\_time/clk\_pix} \quad \text{(eq. 2)}$$

The maximum allowed value for fine_integration_time is:

$$\text{line\_length\_pck} - \text{fine\_integration\_time\_max\_margin} \quad \text{(eq. 3)}$$

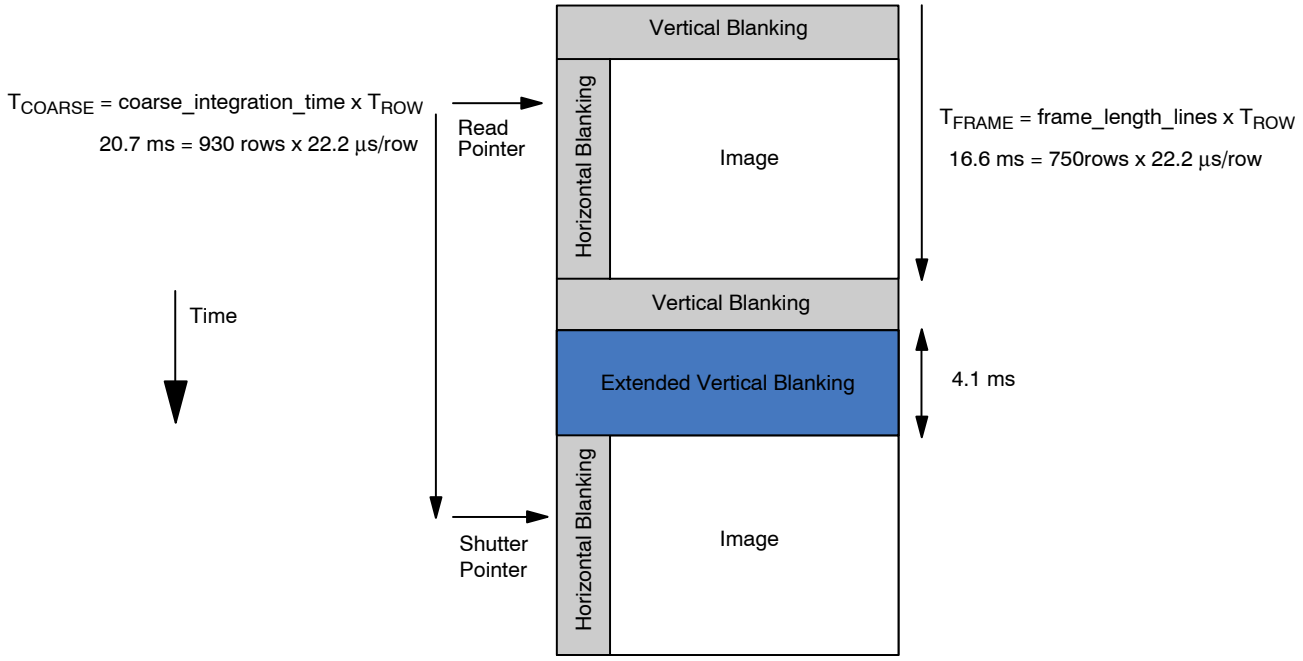


Figure 17. The Row Integration Time is Greater Than the Frame Readout Time

The minimum frame−time is defined by the number of row periods per frame and the row period. The sensor frame−time will increase if the *coarse_integration_time* is set to a value equal to or greater than the *frame_length_lines*.

GAIN STAGES

The sensor analog gain stage will apply the same analog gain to each color channel. Digital gain can be configured to separate levels for each color channel.

The level of analog gain applied is controlled by the coarse_gain and fine_gain at R0x3060 analog gain register. The analog readout circuitry can be configured differently for each analog gain level. Total analog gain is $(2^{coarse\_gain}) \times (1 + fine\_gain / 16)$, where coarse_gain = R0x3060[6:4], fine_gain = R0x3060[3:0].

ON Semiconductor recommends limiting maximum analog gain up to 12x gain for optimal image quality.

Each digital gain can be configured from a gain of 0 to 15.992 using R0x3056, R0x3058, R0x305A, R0x305C, and R0x305E digital gain registers. The digital gain supports 128 gain steps per 6dB of gain. The format of each digital gain register is “xxxx.yyyyyyy” where “xxxx” refers an integer gain of 1 to 15 and “yyyyyyy” is a fractional gain ranging from 0/128 to 127/128.

The sensor includes a digital dithering feature to reduce quantization noise resulting from using digital gain. It can be implemented by setting R0x30BA[5] to 1. The default value is 0.

DATA PEDESTALS

The data pedestal is a constant offset that is added to pixel values at the end of the datapath. The default offset is 168 and is a 12−bit offset. This offset matches the maximum range used by the corrections in the digital readout path. The purpose of the data pedestal is to convert negative values generated by the digital datapath into positive output data.

RESET

The AR0141CS may be reset by the RESET_BAR pin (active LOW) or the reset register.

Hard Reset of Logic

The host system can reset the image sensor by bringing the RESET_BAR pin to a LOW state. Alternatively, the RESET_BAR pin can be connected to an external RC circuit for simplicity. Registers written via the two−wire interface will not be preserved following a hard reset.

Soft Reset of Logic

Soft reset of logic is controlled by the R0x301A Reset register. Bit 0 is used to reset the digital logic of the sensor. Furthermore, by asserting the soft reset, the sensor aborts the current frame it is processing and starts a new frame. This bit is a self−resetting bit and also returns to “0” during two−wire serial interface reads.

CLOCKS

The AR0141CS requires one clock input (EXTCLK).

SENSOR PLL

VCO

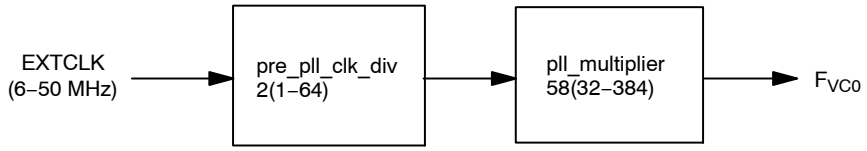


Figure 18. PLL Dividers Affecting VCO Frequency

The sensor contains a phase−locked loop (PLL) that is used for timing generation and control. The required VCO clock frequency is attained through the use of a pre−PLL clock divider followed by a multiplier. The PLL multiplier should be an even integer. If an odd integer (M) is programmed, the PLL will default to the lower (M−1) value to maintain an even multiplier value. The multiplier is followed by a set of dividers used to generate the output clocks required for the sensor array, the pixel analog and digital readout paths, and the output parallel and serial interfaces.

Parallel PLL Configuration

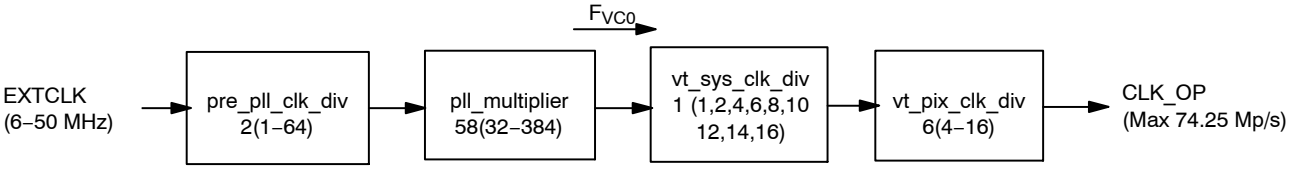


Figure 19. PLL for the Parallel Interface

The maximum output of the parallel interface is 74.25 MPixel/s. The sensor will not use the F_{SERIAL}, F_{SERIAL_CLK}, or CLK_OP when configured to use the parallel interface.

Table 6. PLL PARAMETERS FOR THE PARALLEL INTERFACE

Parameter	Symbol	Min	Max	Unit
External Clock	EXTCLK	6	50	MHz
VCO Clock	F_{VCO}	384	768	MHz
Output Clock	CLK_OP		74.25	Mpixel/s

Table 7. EXAMPLE PLL CONFIGURATION FOR THE PARALLEL INTERFACE

Parameter	Value	Output
F_{VCO}		445.5 MHz (Max)
vt_sys_clk_div	1	
vt_pix_clk_div	6	
CLK_OP		74.25 MPixel/s (= 445.5 MHz / 6)
Output pixel rate		74.25 MPixel/s

Serial PLL Configuration

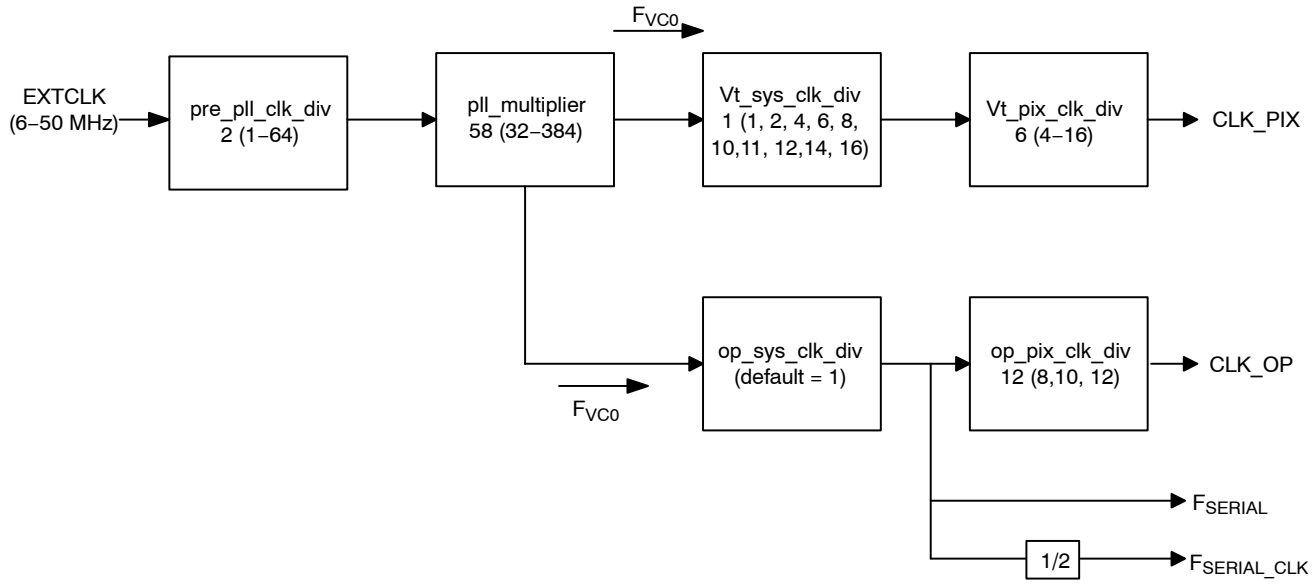


Figure 20. PLL for the Serial Interface

The sensor will use op_sys_clk_div and op_pix_clk_div to configure the output clock per lane (CLK_OP). The configuration will depend on the number of active lanes (1, 2, or 4) configured. To configure the sensor protocol and number of lanes, refer to “Serial Configuration”.

Table 8. PLL PARAMETERS FOR THE SERIAL INTERFACE

Parameter	Symbol	Min	Max	Unit
External Clock	EXTCLK	6	50	MHz
VCO Clock	F_{VCO}	384	768	MHz
Readout Clock	CLK_PIX		74.25	Mpixel/s
Output Serial Data Rate Per Lane	F_{SERIAL}	300 (HiSPi)	600 (HiSPi)	Mbps
Output Serial Clock Speed Per Lane	F_{SERIAL_CLK}	150 (HiSPi)	350(HiSPi)	MHz

Configure the serial output so that it adheres to the following rules:

- The maximum data−rate per lane (F_{SERIAL}) is 600Mbps/lane (HiSPi)
- Configure the output pixel rate per lane (CLK_OP) so that the sensor output pixel rate matches the peak pixel rate (2 × CLK_PIX)
  - — 4−lane: 4 x CLK_OP = 2 × CLK_PIX = Pixel Rate (max: 148.5 Mpixel/s)
  - — 2−lane: 2 x CLK_OP = 2 × CLK_PIX = Pixel Rate (max: 74.25 Mpixel/s)

Table 9. EXAMPLE PLL CONFIGURATIONS FOR THE SERIAL INTERFACE

Parameter	4−lane	2−lane	Units
	12−bit	12−bit	
F_{VCO}	445.5	445.5	MHz
vt_sys_clk_div	1	1	
vt_pix_clk_div	6	12	
op_sys_clk_div	1	1	
op_pix_clk_div	12	12	
F_{SERIAL}	445.5	445.5	MHz
F_{SERIAL_CLK}	222.75	222.75	MHz

Table 9. EXAMPLE PLL CONFIGURATIONS FOR THE SERIAL INTERFACE (continued)

Parameter	4−lane 12−bit	2−lane 12−bit	Units
CLK_PIX	74.25	37.125	Mpixel/s
CLK_OP	37.125	37.125	Mpixel/s
Pixel Rate	148.5	74.25	Mpixel/s

Stream/Standby Control

The sensor supports a soft standby mode. In this mode, the external clock can be optionally disabled to further minimize power consumption. If this is done, then the "Power−Up Sequence" must be followed.

Soft Standby

Soft Standby is a low−power state that is controlled through register R0x301A[2]. Depending on the value of R0x301A[4], the sensor will go to Standby after completion of the current frame readout. When the sensor comes back from Soft Standby, previously written register settings are still maintained. Soft Standby will not occur if the Trigger pin is held high.

A specific sequence needs to be followed to enter and exit from Soft Standby.

Entering Soft Standby:

1. Set R0x301A[12] = 1 if serial mode was used
2. Set R0x301A[2] = 0 and drive Trigger pin low
3. Turn off external clock to further minimize power consumption

Exiting Soft Standby:

1. Enable external clock if it was turned off
2. Set R0x301A[2] = 1 or drive Trigger pin high
3. Set R0x301A[12] = 0 if serial mode is used

SENSOR READOUT

Image Acquisition Modes

The AR0141CS supports two image acquisition modes:

- Electronic rolling shutter (ERS) mode
  This is the normal mode of operation. When the AR0141CS is streaming, it generates frames at a fixed rate, and each frame is integrated (exposed) using the ERS. When the ERS is in use, timing and control logic within the sensor sequences through the rows of the array, resetting and then reading each row in turn. In the time interval between resetting a row and subsequently reading that row, the pixels in the row integrate incident light. The integration (exposure) time is controlled by varying the time between row reset and row readout. For each row in a frame, the time between row reset and row readout is the same, leading to a uniform integration time across the frame. When the integration time is changed (by using the two−wire serial interface to change register settings), the timing and control logic controls the transition from old to new integration time in such a way that the stream of output frames from the AR0141CS switches cleanly from the old integration time to the new while only generating frames with uniform integration. See “Changes to Integration Time” in the AR0141CS Register Reference.
- Global reset mode
  This mode can be used to acquire a single image at the current resolution. In this mode, the end point of the pixel integration time is controlled by an external electromechanical shutter, and the AR0141CS provides control signals to interface to that shutter.
  The benefit of using an external electromechanical shutter is that it eliminates the visual artifacts associated with ERS operation. Visual artifacts arise in ERS operation, particularly at low frame rates, because an ERS image effectively integrates each row of the pixel array at a different point in time.

Window Control

The sequencing of the pixel array is controlled by the x_addr_start, y_addr_start, x_addr_end, and y_addr_end registers.

Readout Modes

Horizontal Mirror

When the horiz_mirror bit (R0x3040[14]) is set in the read_mode register, the order of pixel readout within a row is reversed, so that readout starts from *x_addr_end + 1* and ends at *x_addr_start*. Figure 21 shows a sequence of 6 pixels being read out with R0x3040[14] = 0 and R0x3040[14] = 1.

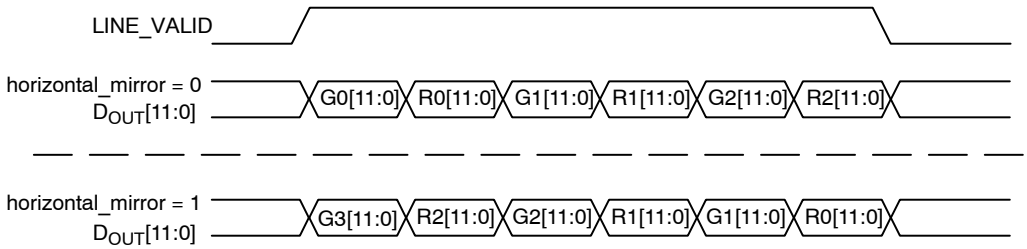


Figure 21. Effect of Horizontal Mirror on Readout Order

Vertical Flip

When the vert_flip bit (R0x3040[15]) is set in the read_mode register, the order in which pixel rows are read out is reversed, so that row readout starts from *y_addr_end* and ends at *y_addr_start*. Figure 30 shows a sequence of 6 rows being read out with R0x3040[15] = 0 and R0x3040[15] = 1.

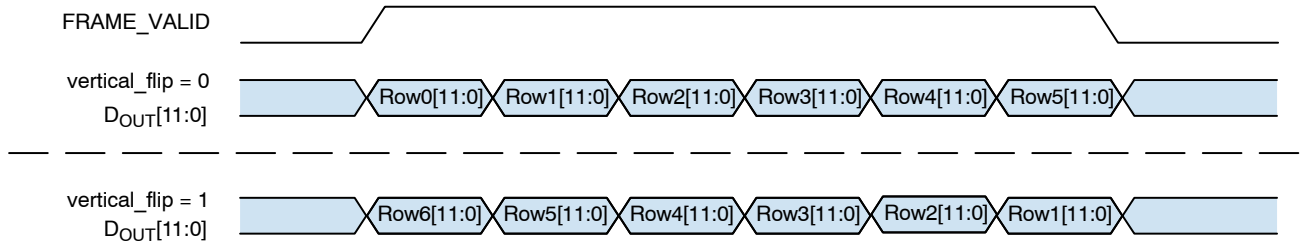


Figure 22. Effect of Vertical Flip on Readout Order

SUBSAMPLING

The AR0141CS supports subsampling. Subsampling allows the sensor to read out a smaller set of active pixels by either skipping, binning, or summing pixels within the readout window.

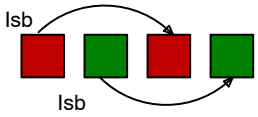


Figure 23. Horizontal Binning in the AR0141CS Sensor

Horizontal binning is achieved either in the pixel readout or the digital readout. The sensor will sample the combined 2x adjacent pixels within the same color plane.

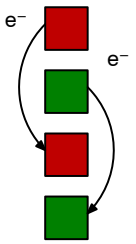


Figure 24. Vertical Row Binning in the AR0141CS Sensor

Vertical row binning is applied in the pixel readout. Row binning can be configured as 2x rows within the same color plane.

Pixel skipping can be configured up to 2x in both the x−direction and y−direction. Skipping pixels in the x−direction will not reduce the row time. Skipping pixels in the y−direction will reduce the number of rows from the sensor effectively reducing the frame time. Skipping will introduce image artifacts from aliasing.

Table 10. AVAILABLE SKIP AND BIN MODES IN THE AR0141CS SENSOR

Subsampling Method	Horizontal	Vertical
Skipping	2x	2x
Binning	2x	2x

The sensor increments its x and y address based on the x_odd_inc and y_odd_inc value. The value indicates the addresses that are skipped after each pair of pixels or rows has been read.

The sensor will increment x and y addresses in multiples of 2. This indicates that a GreenR and Red pixel pair will be read together. As well, that the sensor will read a Gr−R row first followed by a B−Gb row.

$$\text{x subsampling factor} = \frac{1 + \text{x\_odd\_inc}}{2} \quad \text{(eq. 4)}$$

$$\text{y subsampling factor} = \frac{1 + \text{y\_odd\_inc}}{2} \quad \text{(eq. 5)}$$

A value of 1 is used for *x_odd_inc* and *y_odd_inc* when no pixel subsampling is indicated. In this case, the sensor is incrementing x and y addresses by 1 + 1 so that it reads consecutive pixel and row pairs. To implement a 2x skip in the x direction, the *x_odd*_inc is set to 3 so that the x address increment is 1 + 3, meaning that sensor will skip every other Gr−R pair.

Table 11. CONFIGURATION FOR HORIZONTAL SUBSAMPLING

	x_odd_inc	Restrictions
No Subsampling	x_odd_inc = 1 skip = (1+1) × 0.5 = 1x	The horizontal FOV must be programmed to meet the following rule: $\frac{x\_addr\_end - x\_addr\_start + 1}{(x\_odd\_inc + 1)/2}$ = even number
Skip 2x	x_odd_inc = 3 skip = (1+3) ×0.5 = 2x	
Analog Bin 2x	x_odd_inc = 3 skip = (1+3) × 0.5 = 2x col_sf_bin_en = 1	
Digital Bin 2x	x_odd_inc = 3 skip = (1+3) × 0.5 = 2x col_bin = 1	

Table 12. CONFIGURATION FOR VERTICAL SUBSAMPLING

	y_odd_inc	Restrictions
No Subsampling	y_odd_inc = 1 skip = (1+1) × 0.5 = 1x row_bin = 0	The vertical FOV must be programmed to meet the following rule: $\frac{y\_addr\_end - y\_addr\_start + 1}{(y\_odd\_inc + 1)/2}$ = even number
Skip 2x	y_odd_inc = 3 skip = (1+3) × 0.5 = 2x row_bin = 0	
Analog Bin 2x	y_odd_inc = 3 skip = (1+3) × 0.5 =2x row_bin = 1	

1. In skip2 the window size has to be a multiple of 4.

SENSOR FRAME RATE

The time required to read out an image frame (T_{FRAME}) can be derived from the number of clocks required to output each image and the pixel clock.

The frame−rate is the inverse of the frame period.

$$fps = \frac{1}{T_{FRAME}} \quad \text{(eq. 6)}$$

The number of clocks can be simplified further into the following parameters:

- The number of clocks required for each sensor row (*line_length_pck*)

This parameter also determines the sensor row period when referenced to the sensor readout clock. (T_{ROW} = line_length_pck x 1/CLK_PIX)

- The number of row periods per frame (*frame_length_lines*)
- An extra delay between frames used to achieve a specific output frame period (*extra_delay*)

$$T_{FRAME} = 1/(CLK\_PIX) \times [frame\_length\_lines \times line\_length\_pck + extra\_delay] \quad \text{(eq. 7)}$$

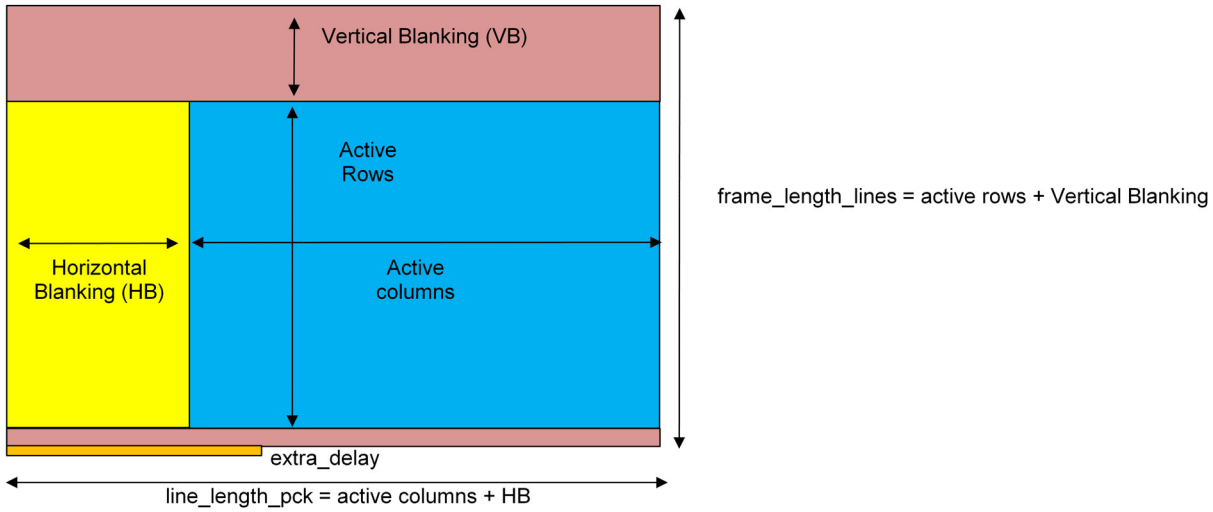


Figure 25. Frame Period Measured in Clocks

Row Period (T_{ROW})

line_length_pck will determine the number of clock periods per row and the row period (T_{ROW}) when combined with the sensor readout clock. *line_length_pck* includes both the active pixels and the horizontal blanking time per row. The sensor utilizes two readout paths, as seen in Figure 1, allowing the sensor to output two pixels during each pixel clock.

Row Periods Per Frame

frame_length_lines determines the number of row periods (T_{ROW}) per frame. This includes both the active and blanking rows. The minimum vertical blanking value is defined by the number of OB rows read per frame, two embedded data rows, and two blank rows.

$$\text{Minimumframe\_length\_lines} = \frac{\text{y\_addr\_end} - \text{y\_addr\_start} + 1}{(\text{y\_odd\_inc} + 1)/2} + \text{min\_vertical\_blanking} \quad \text{(eq. 8)}$$

The sensor is configured to output frame information in two embedded data rows by setting R0x3064[8] to 1 (default). If R0x3064[8] is set to 0, the sensor will instead output two blank rows. The data configured in the two embedded rows is defined in two embedded rows of data at the top of the frame by setting R0x3064[7] and two rows of embedded statistics at the end of the frame by setting R0x3064[7] for exposure calculations. See the section on Embedded Data and Statistics.

Table 13. MINIMUM VERTICAL BLANKING CONFIGURATION

R0x3180[7:4]	OB Rows	min_vertical_blanking
0x8 (Default)	8 OB Rows	8 OB + 8 = 16
0x4	4 OB Rows	4 OB + 8 = 12
0x2	2 OB Rows	2 OB + 8 = 10

The locations of the OB rows, embedded rows, and blank rows within the frame readout are identified in Figure 26: “Slave Mode Active State and Vertical Blanking,”.

SLAVE MODE

The slave mode feature of the AR0141CS supports triggering the start of a frame readout from a VD signal that is supplied from an external device. The slave mode signal allows for precise control of frame rate and register change updates. The VD signal is an edge triggered input to the trigger pin and must be at least 3 PIXCLK cycles wide.

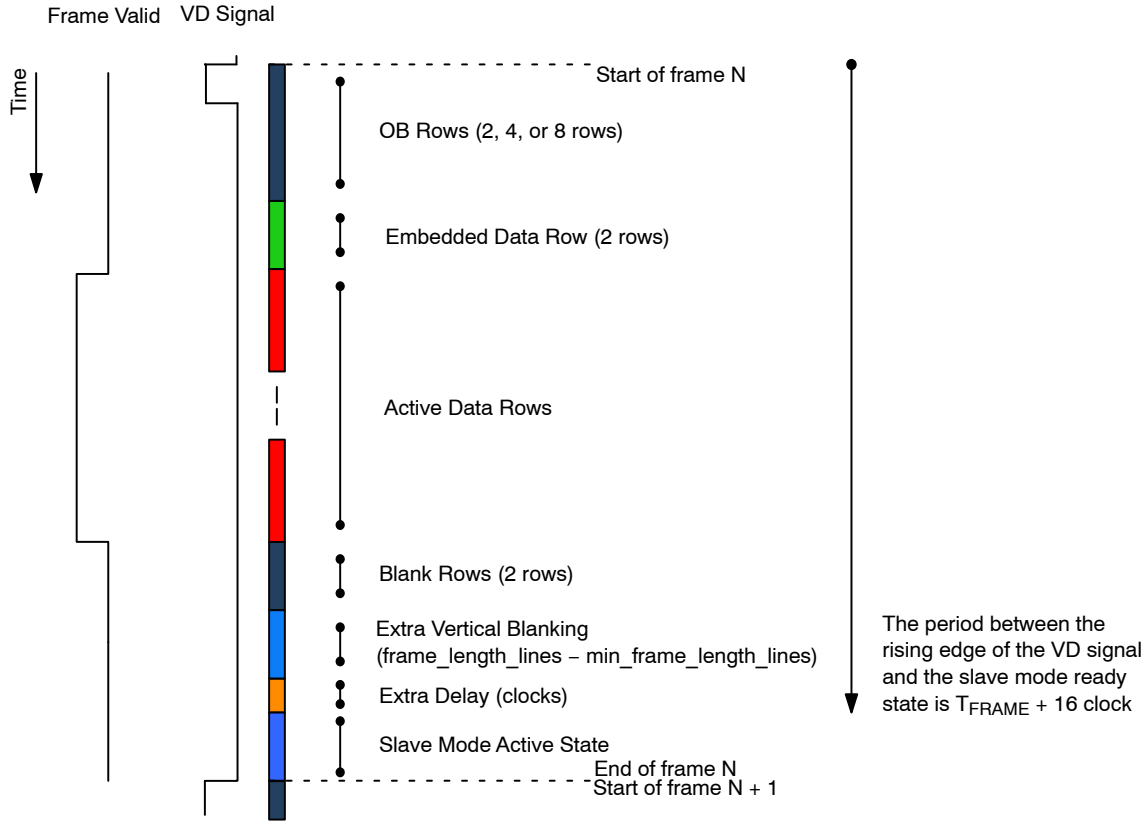
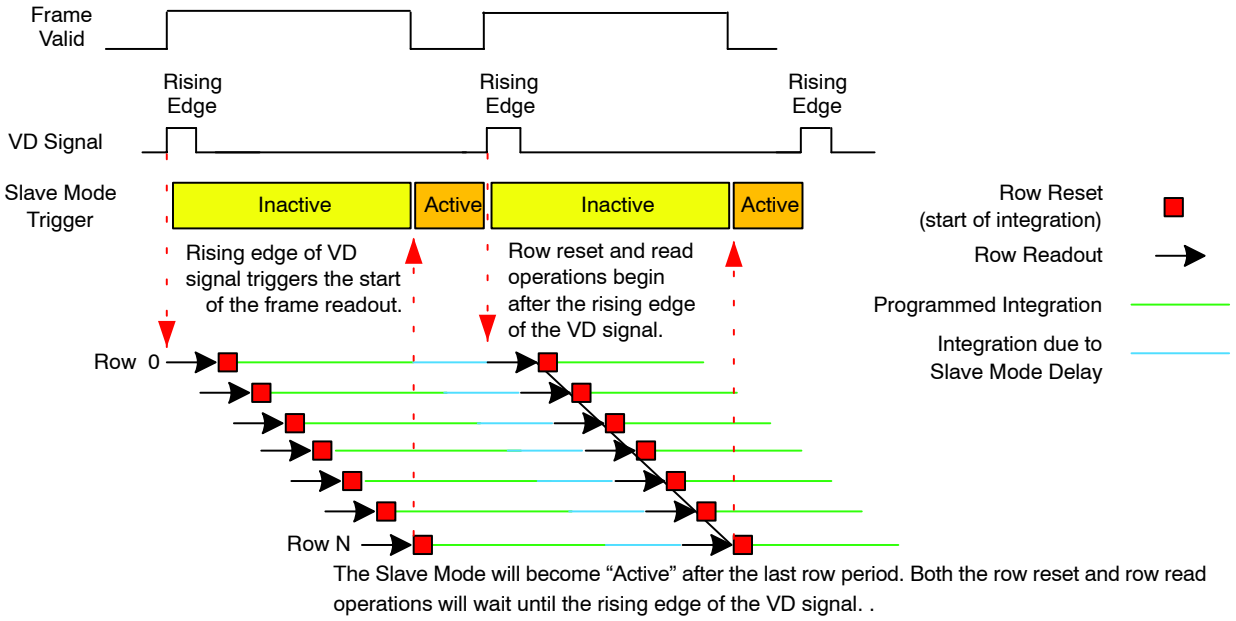


Figure 26. Slave Mode Active State and Vertical Blanking

If the slave mode is disabled, the new frame will begin after the extra delay period is finished.

The slave mode will react to the rising edge of the input VD signal if it is in an active state. When the VD signal is received, the sensor will begin the frame readout and the slave mode will remain inactive for the period of one frame time plus 16 clock periods (T_{FRAME} + (16 / CLK_PIX)). After this period, the slave mode will re−enter the active state and will respond to the VD signal.

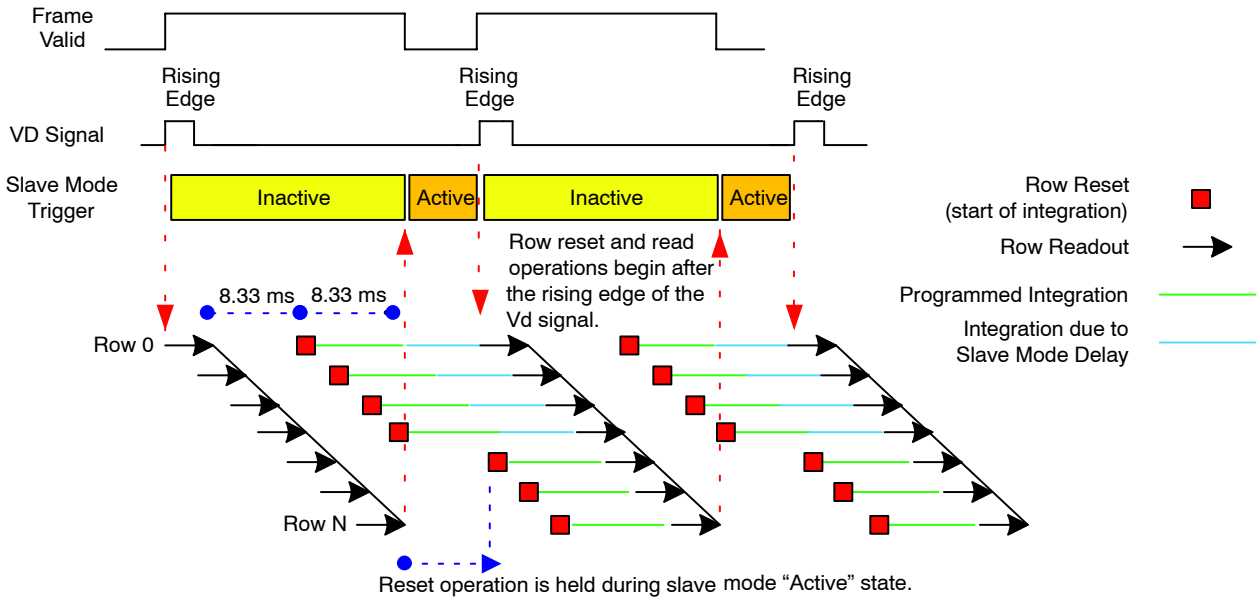


Note: The integration of the last row is started before the end of the programmed integration for the first row.

Figure 27. Slave Mode Example with Equal Integration and Frame Readout Periods

The row shutter and read operations will stop when the slave mode becomes active and is waiting for the VD signal. The following should be considered when configuring the sensor to use the slave mode:

1. The frame period (T_{FRAME}) should be configured to be less than the period of the input VD signal. The sensor will disregard the input VD signal if it appears before the frame readout is finished
2. If the sensor integration time is configured to be less than the frame period, then the sensor will not have reset all of the sensor rows before it begins waiting for the input VD signal. This error can be minimized by configuring the frame period to be as close as possible to the desired frame rate (period between VD signals)



Note: The sensor read pointer will have paused at row 0 while the shutter pointer pauses at row N/2. The extra integration caused by the slave mode delay will only be seen by rows 0 to N/2. The example below is for a frame readout period of 16.6 ms while the integration time is configured to 8.33 ms.

Figure 28. Slave Mode Example Where the Integration Period is Half of the Frame Readout Period

When the slave mode becomes active, the sensor will pause both row read and row reset operations. (Note: The row integration period is defined as the period from row reset to row read.) The frame−time should therefore be configured so that the slave mode “wait period” is as short as possible. In the case where the sensor integration time is shorter than the frame time, the “wait period” will only increase the integration of the rows that have been reset following the last VD pulse.

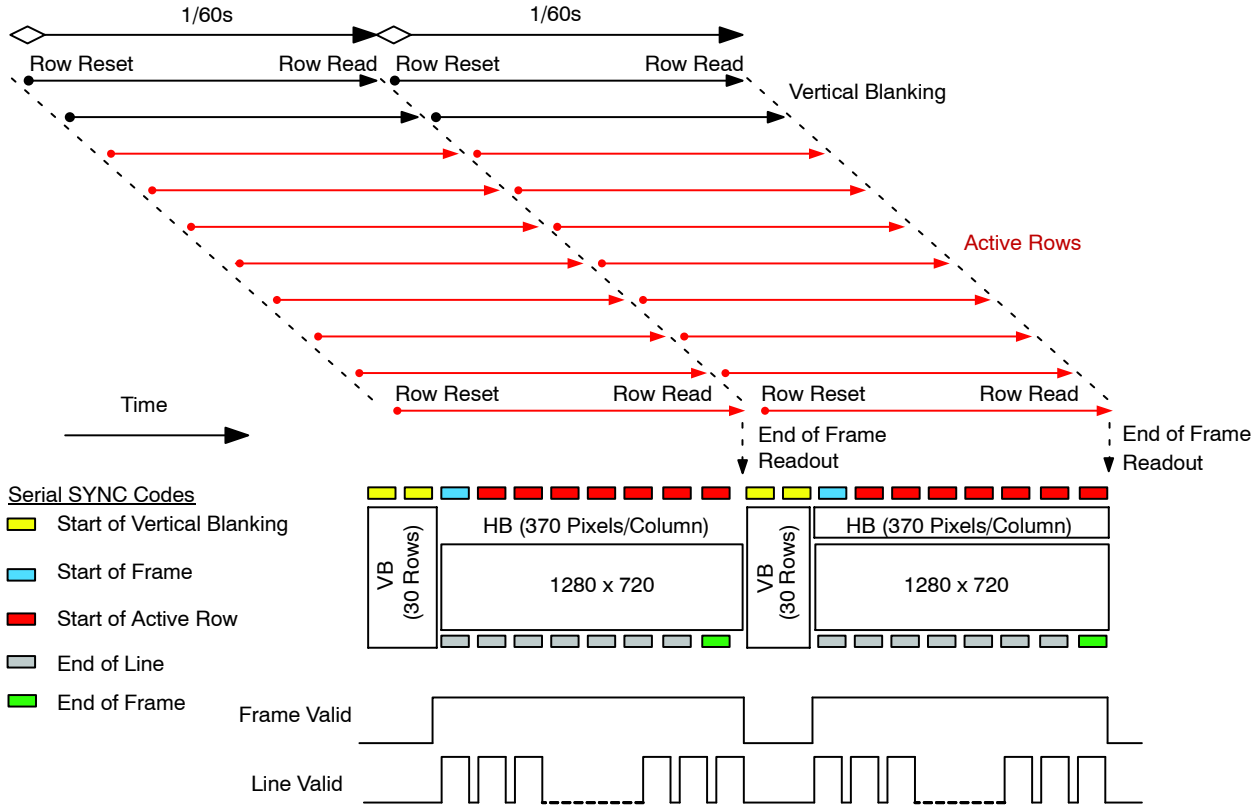
The period between slave mode pulses must also be greater than the frame period. If the rising edge of the VD pulse arrives while the slave mode is inactive, the VD pulse will be ignored and will wait until the next VD pulse has arrived.

To enter slave mode:

1. While in soft−standby, set R0x30CE[4] = 1 to enter slave mode
2. Enable the input pins (TRIGGER) by setting R0x301A[8] = 1
3. Enable streaming by setting R0x301A[2] = 1
4. Apply sync−pulses to the TRIGGER input

FRAME READOUT

The sensor readout begins with vertical blanking rows followed by the active rows. The frame readout period can be defined by the number of row periods within a frame (*frame_length_lines*) and the row period (*line_length_pck*/*clk_pix*). The sensor will read the first vertical blanking row at the beginning of the frame period and the last active row at the end of the row period.



Note: The frame valid and line valid signals mentioned in this diagram represent internal signals within the sensor. The SYNC codes represented in this diagram represent the HiSPi Streaming−SP protocol.

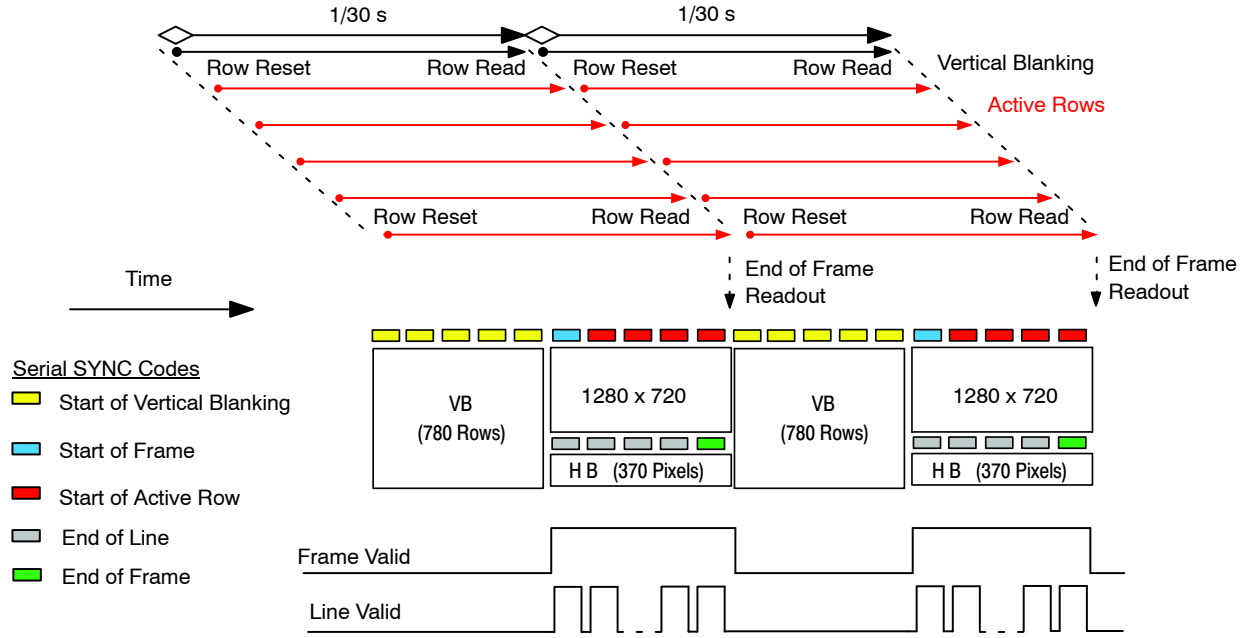
Figure 29. Example of the Sensor Output of a 1280 x 720 Frame at 60 fps

Figure 29 aligns the frame integration and readout operation to the sensor output. It also shows the sensor output using the HiSPi Streaming−SP protocol. Different sensor protocols will list different SYNC codes.

Table 14. SERIAL SYNC CODES INCLUDED WITH EACH PROTOCOL INCLUDED WITH THE AR0141CS SENSOR

Interface/Protocol	Start of Vertical Blanking Row (SOV)	Start of Frame (SOF)	Start of Active Line (SOL)	End of Line (EOL)	End of Frame (EOF)
Parallel	Parallel interface uses FRAME VALID (FV) and LINE VALID (LV) outputs to denote start and end of line and frame.				
HiSPi Streaming−S	Required	Unsupported	Required	Unsupported	Unsupported
HiSPi Streaming−SP	Required	Required	Required	Unsupported	Unsupported
HiSPi Packetized SP	Unsupported	Required	Required	Required	Required

Figure 30 illustrates how the sensor active readout time can be minimized while reducing the frame rate. 750 VB rows were added to the output frame to reduce the 1280 x 720 frame rate from 60 fps to 30 fps without increasing the delay between the readout of the first and last active row.



Note: The frame valid and line valid signals mentioned in this diagram represent internal signals within the sensor. The SYNC codes represented in this diagram represent the HiSPi Streaming−SP protocol.

Figure 30. Example of the Sensor Output of a 1280x 720 Frame at 30 fps

CHANGING SENSOR MODES

Register Changes

All register writes are delayed by one frame. A register that is written to during the readout of frame *n* will not be updated to the new value until the readout of frame *n + 2*. This includes writes to the sensor gain and integration registers.

Real−Time Context Switching

In the AR0141CS, the user may switch between two full register sets A and B by writing to a context switch change bit in R0x30B0[13]. When the context switch is configured to context A the sensor will reference the context A registers. If the context switch is changed from A to B during the readout of frame *n*, the sensor will then reference the context B *coarse_integration_time* registers in frame *n + 1* and all other context B registers at the beginning of reading frame *n + 2*. The sensor will show the same behavior when changing from context B to context A.

Table 15. LIST OF CONFIGURABLE REGISTERS FOR CONTEXT A AND CONTEXT B

Context A		Context B	
Register Description	**Address**	**Register Description**	**Address**
coarse_integration_time	0x3012	coarse_integration_time_cb	0x3016
line_length_pck	0x300C	line_length_pck_cb	0x303E
frame_length_lines	0x300A	frame_length_lines_cb	0x30AA
row_bin	0x3040[12]	row_bin_cb	0x3040[10]
col_bin	0x3040[13]	col_bin_cb	0x3040[11]
fine_gain	0x3060[3:0]	fine_gain_cb	0x3060[11:8]
coarse_gain	0x3060[5:4]	coarse_gain_cb	0x3060[13:12]
x_addr_start	0x3004	x_addr_start_cb	0x308A
y_addr_start	0x3002	y_addr_start_cb	0x308C
x_addr_end	0x3008	x_addr_end_cb	0x308E
y_addr_end	0x3006	y_addr_end_cb	0x3090
y_odd_inc	0x30A6	y_odd_inc_cb	0x30A8
x_odd_inc	0x30A2	x_odd_inc_cb	0x30AE
green1_gain	0x3056	green1_gain_cb	0x30BC
blue_gain	0x3058	blue_gain_cb	0x30BE
red_gain	0x305A	red_gain_cb	0x30C0
green2_gain	0x305C	green2_gain_cb	0x30C2
global_gain	0x305E	global_gain_cb	0x30C4

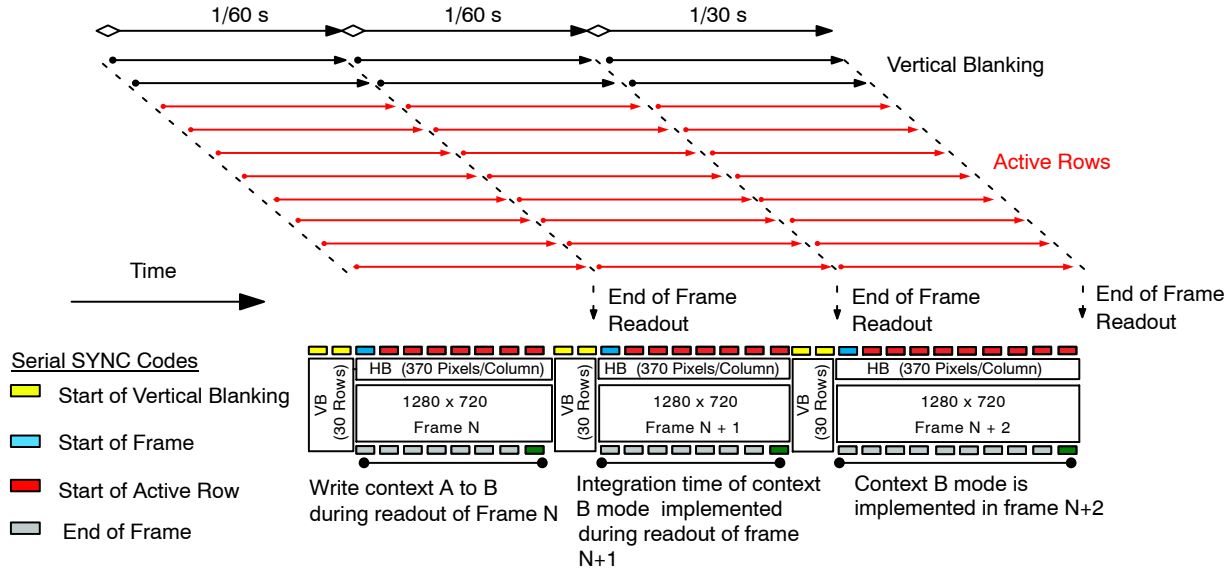


Figure 31. Example of Changing the Sensor from Context A to Context B

Compression

The AR0141CS can optionally compress 12−bit data to 10−bit using A−law compression. The compression is applied after the data pedestal has been added to the data. See "Data Pedestals".

The A−law compression is disabled by default and can be enabled by setting R0x31D0 from "0" to "1" and 0x31AC needs to be set to 0x0C0A.

Table 16. A−LAW COMPRESSION TABLE FOR 12−10 BITS

	Input Values												Compressed Codeword									
Input Range	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0 to 127	0	0	0	0	0	a	b	c	d	e	f	g	0	0	0	a	b	c	d	e	f	g
128 to 255	0	0	0	0	1	a	b	c	d	e	f	g	0	0	1	a	b	c	d	e	f	g
256 to 511	0	0	0	1	a	b	c	d	e	f	g	X	0	1	0	a	b	c	d	e	f	g
512 to 1023	0	0	1	a	b	c	d	e	f	g	X	X	0	1	1	a	b	c	d	e	f	g
1024 to 2047	0	1	a	b	c	d	e	f	g	h	X	X	1	0	a	b	c	d	e	f	g	h
2048 to 4095	1	a	b	c	d	e	f	g	h	X	X	X	1	1	a	b	c	d	e	f	g	h

Temperature Sensor

The AR0141CS sensor has a built−in temperature sensor, accessible through registers, that is capable of measuring die junction temperature.

The temperature sensor can be enabled by writing R0x30B4[0] = 1 and R0x30B4[4] =1. After this, the temperature sensor output value can be read from R0x30B2[9:0].

The value read out from the temperature sensor register is an ADC output value that needs to be converted downstream to a final temperature value in degrees Celsius. Since the PTAT device characteristic response is quite linear in the temperature range of operation required, a simple linear function in the format of the equation below can be used to convert the ADC output value to the final temperature in degrees Celsius.

$$\text{Temperature} = \text{slope} \times \text{R0x30B2[9 : 0]} + T_0 \quad \text{(eq. 9)}$$

For this conversion, a minimum of two known points are needed to construct the line formula by identifying the slope and y−intercept "T_0". These calibration values can be read from registers R0x30C6 and R0x30C8, which correspond to value read at 105°C and 55°C respectively. Once read, the slope and y−intercept values can be calculated and used in Equation 9

For more information on the temperature sensor registers, refer to the AR0141CS Register Reference.

Embedded Data and Statistics

The AR0141CS has the capability to output image data and statistics embedded within the frame timing. There are two types of information embedded within the frame readout.

- Embedded Data:
  If enabled, these are displayed on the two rows immediately before the first active pixel row is displayed
- Embedded Statistics:
  If enabled, these are displayed on the two rows immediately after the last active pixel row is displayed

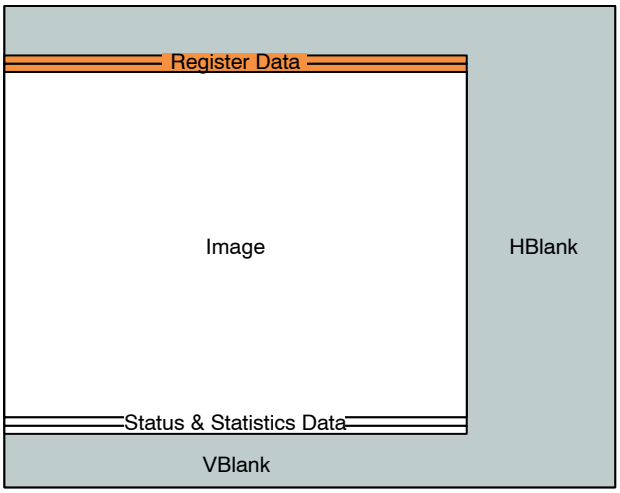


Figure 32. Frame Format with Embedded Data Lines Enabled

Embedded Data

The embedded data contains the configuration of the image being displayed. This includes all register settings used to capture the current frame. The registers embedded in these rows are as follows:

Line 1: Registers R0x3000 to R0x312F.

Line 2: Registers R0x3136 to R0x31BF, R0x31D0 to R0x31FF.

NOTE: All undefined registers will have a value of 0.

In parallel mode, since the pixel word depth is 12 bits/pixel, the sensor 16−bit register data will be transferred over 2 pixels where the register data will be broken up into 8 MSB and 8 LSB. The alignment of the 8−bit data will be on the 8 MSB bits of the 12−bit pixel word. For example, if a register value of 0x1234 is to be transmitted, it will be transmitted over two, 12−bit pixels as follows: 0x120, 0x340.

Embedded Statistics

The embedded statistics contain frame identifiers and histogram information of the image in the frame. This can be used by downstream auto−exposure algorithm blocks to make decisions about exposure adjustment.

This histogram is divided into 244 bins with a bin spacing of 64 evenly spaced bins for digital code values 0 to 2^8, 120 evenly spaced bins for values 2^8 to 2^{12}, 60 evenly spaced bins for values 2^{12} to 2^{16}. It is recommended that auto exposure algorithms be developed using the histogram statistics on line 1.

The first pixel of each line in the embedded statistics is a tag value of 0x0B0. This signifies that all subsequent statistics data is 10 bit data aligned to the MSB of the 12−bit pixel.

Figure 33 summarizes how the embedded statistics transmission looks like. It should be noted that data, as shown in Figure 33, is aligned to the MSB of each word:

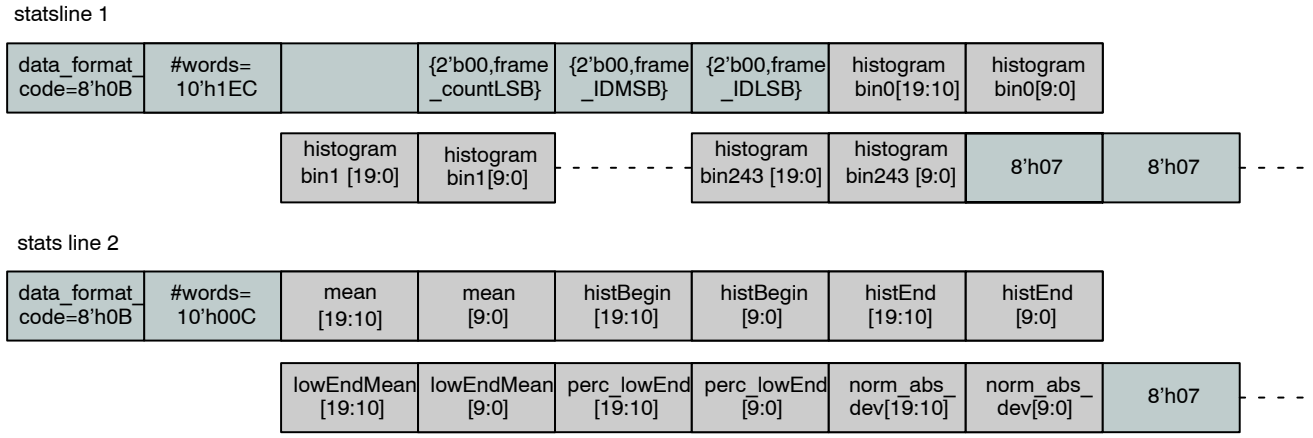


Figure 33. Format of Embedded Statistics Output within a Frame

The statistics embedded in these rows are as follows:

Line 1:

- 0x0B0 – identifier
- Register 0x303A – frame_count
- Register 0x31D2 – frame ID
- Histogram data – histogram bins 0–243

Line 2:

- 0x0B0
- Mean
- Histogram Begin
- Histogram End
- Low End Histogram Mean
- Percentage of Pixels Below Low End Mean
- Normal Absolute Deviation

Test Patterns

The AR0141CS has the capability of injecting a number of test patterns into the top of the datapath to debug the digital logic. With one of the test patterns activated, any of the datapath functions can be enabled to exercise it in a deterministic fashion. Test patterns are selected by Test_Pattern_Mode register (R0x3070). Only one of the test patterns can be enabled at a given point in time by setting the Test_Pattern_Mode register according to Table 17. When test patterns are enabled the active area will receive the value specified by the selected test pattern and the dark pixels will receive the value in Test_Pattern_Green (R0x3074 and R0x3078) for green pixels, Test_Pattern_Blue (R0x3076) for blue pixels, and Test_Pattern_Red (R0x3072) for red pixels. The noise pedestal offset at register 0x30FE impacts on the test pattern output, so the noise_pedestal needs to be set as 0x0000 for normal test pattern output.

Table 17. TEST PATTERN MODES

Test_Pattern_Mode	Test Pattern Output
0	No test pattern (normal operation)
1	Solid color test pattern
2	100% Vertical Color Bars test pattern
3	Fade−to−Gray Vertical Color Bars test pattern
256	Walking 1s test pattern (12−bit)

Solid Color

When the color field mode is selected, the value for each pixel is determined by its color. Green pixels will receive the value in Test_Pattern_Green, red pixels will receive the value in Test_Pattern_Red, and blue pixels will receive the value in Test_Pattern_Blue.

Vertical Color Bars

When the vertical color bars mode is selected, a typical color bar pattern will be sent through the digital pipeline.

Walking 1s

When the walking 1s mode is selected, a walking 1s pattern will be sent through the digital pipeline. The first value in each row is 1.

TWO−WIRE SERIAL REGISTER INTERFACE

The two−wire serial interface bus enables read/write access to control and status registers within the AR0141CS.

The interface protocol uses a master/slave model in which a master controls one or more slave devices. The sensor acts as a slave device. The master generates a clock (SCLK) that is an input to the sensor and is used to synchronize transfers. Data is transferred between the master and the slave on a bidirectional signal (SDATA). SDATA is pulled up to VDD_IO off−chip by a 1.5kΩ resistor. Either the slave or master device can drive SDATA LOW—the interface protocol determines which device is allowed to drive SDATA at any given time.

The protocols described in the two−wire serial interface specification allow the slave device to drive SCLKLOW; the AR0141CS uses SCLK as an input only and therefore never drives it LOW.

Protocol

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

1. a (repeated) start condition
2. a slave address/data direction byte
3. an (a no) acknowledge bit
4. a message byte
5. 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.

Start Condition

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.

Stop Condition

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

Data Transfer

Data is transferred serially, 8 bits at a time, with the 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/Data Direction Byte

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. The default slave addresses used by the AR0141CS are 0x20 (write address) and 0x21 (read address) in accordance with the specification. Alternate slave addresses of0x30 (write address) and 0x31 (read address) can be selected by enabling and asserting the SADDR input.

An alternate slave address can also be programmed through R0x31FC.

Message Byte

Message bytes are used for sending register addresses and register write data to the slave device and for retrieving register read data.

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.

Typical Sequence

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 the 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 an acknowledge bit at the end of the sequence. 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, the same way as with a 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.

Single READ from Random Location

This sequence (Figure 34) 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 34 shows how the internal register address maintained by the AR0141CS is loaded and incremented as the sequence proceeds.

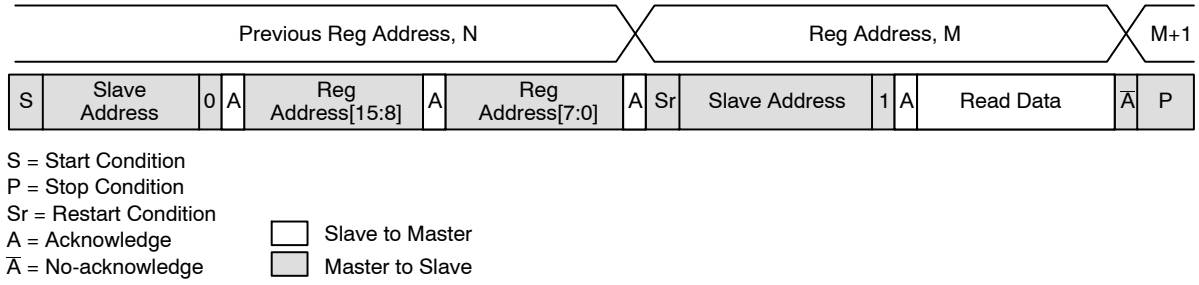


Figure 34. Single READ from Random Location

Single READ from Current Location

This sequence (Figure 35) performs a read using the current value of the AR0141CS 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.

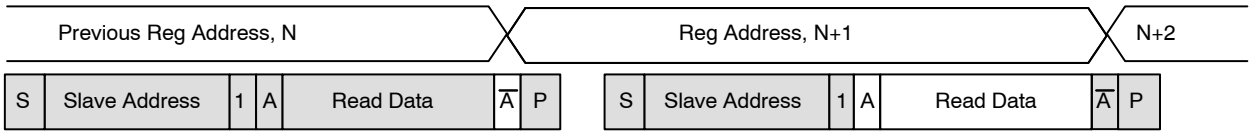


Figure 35. Single READ from Current Location

Sequential READ, Start from Random Location

This sequence (Figure 36) starts in the same way as the single READ from random location (Figure 34). 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.

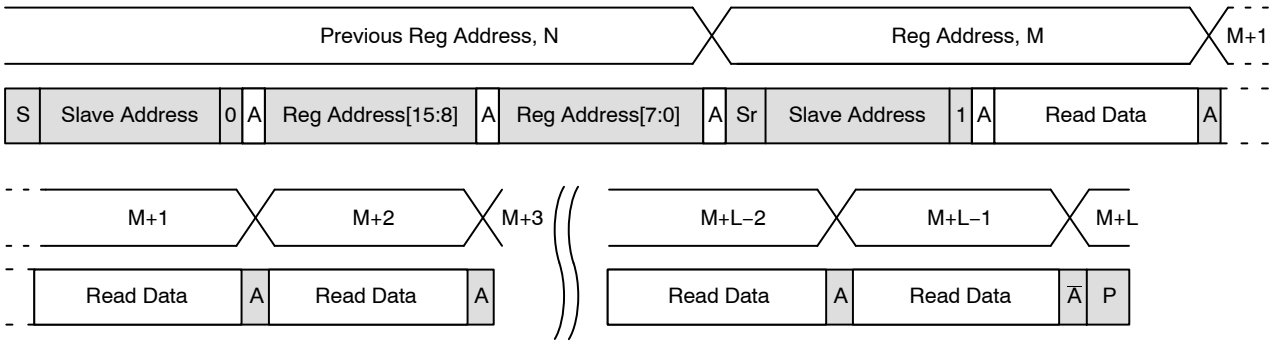


Figure 36. Sequential READ, Start from Random Location

Sequential READ, Start from Current Location

This sequence (Figure 37) starts in the same way as the single READ from current location (Figure 35). 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 37. Sequential READ, Start from Current Location

Single WRITE to Random Location

This sequence (Figure 38) 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.

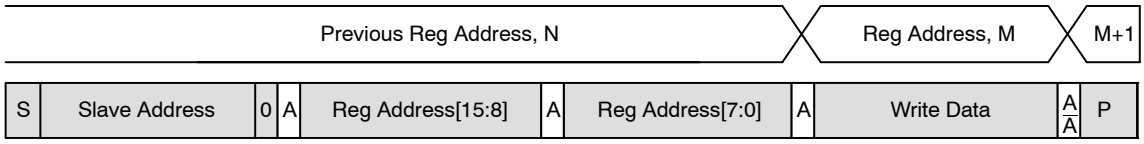


Figure 38. Single WRITE to Random Location

Sequential WRITE, Start at Random Location

This sequence (Figure 39) starts in the same way as the single WRITE to random location (Figure 38). 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.

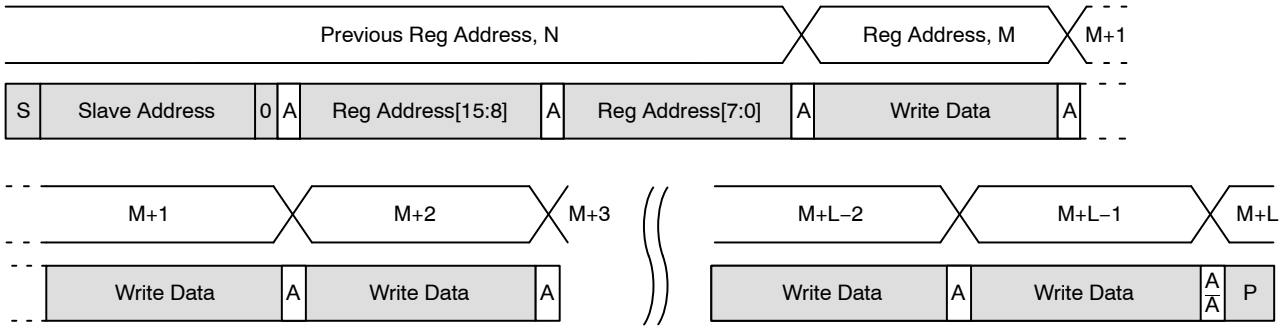


Figure 39. Sequential WRITE, Start at Random Location

SPECTRAL CHARACTERISTICS

Figure 40 specifies the quantum efficiency of the RGB Bayer sensor.

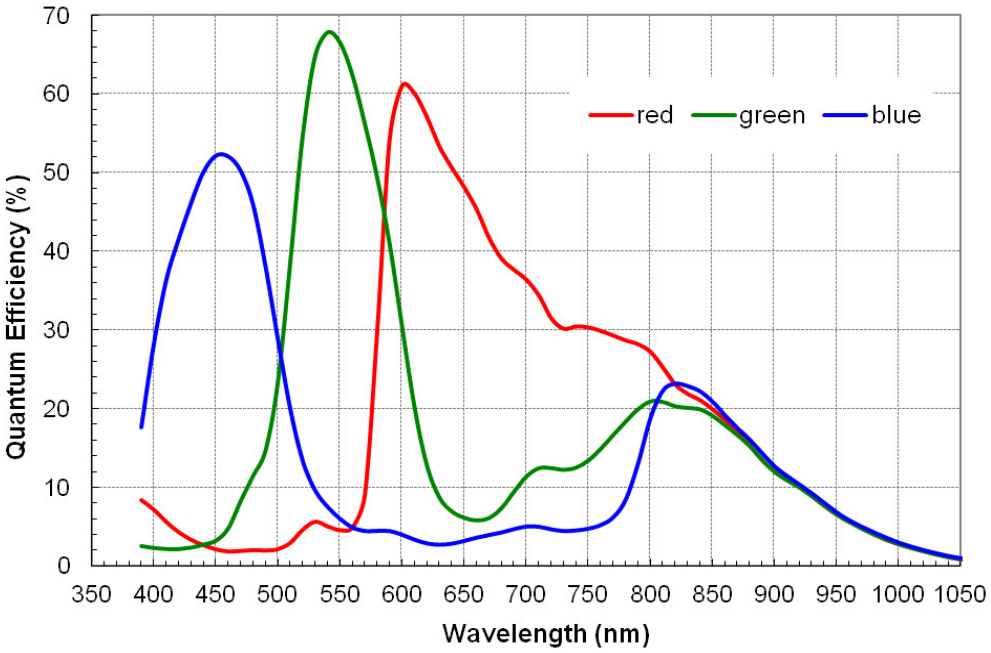


Figure 40. Quantum Efficiency – Color Sensor

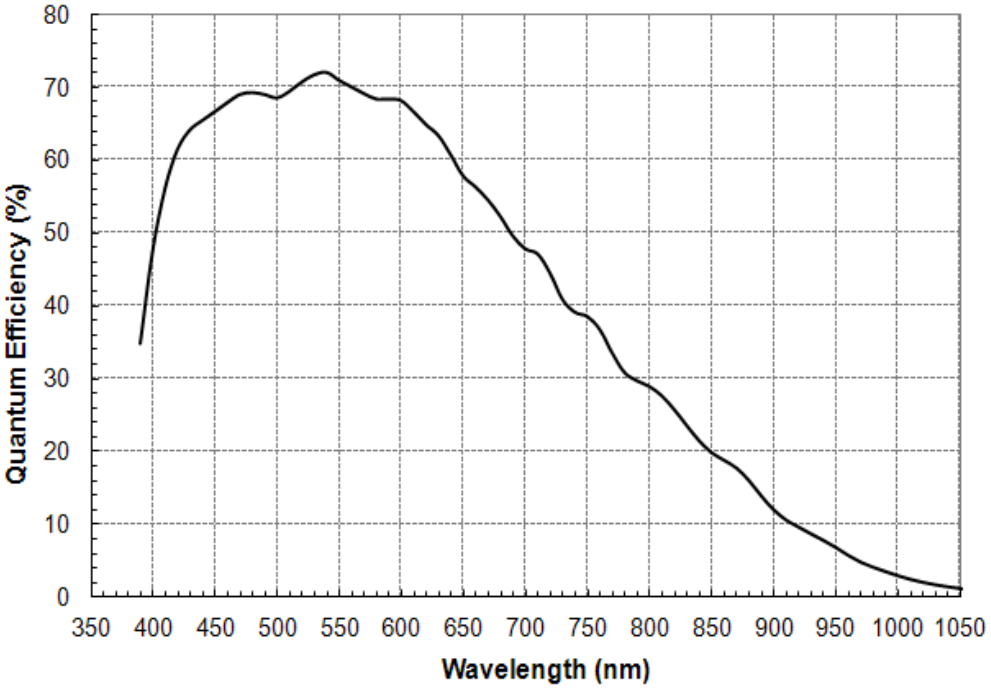


Figure 41. Quantum Efficiency – Monochrome Sensor

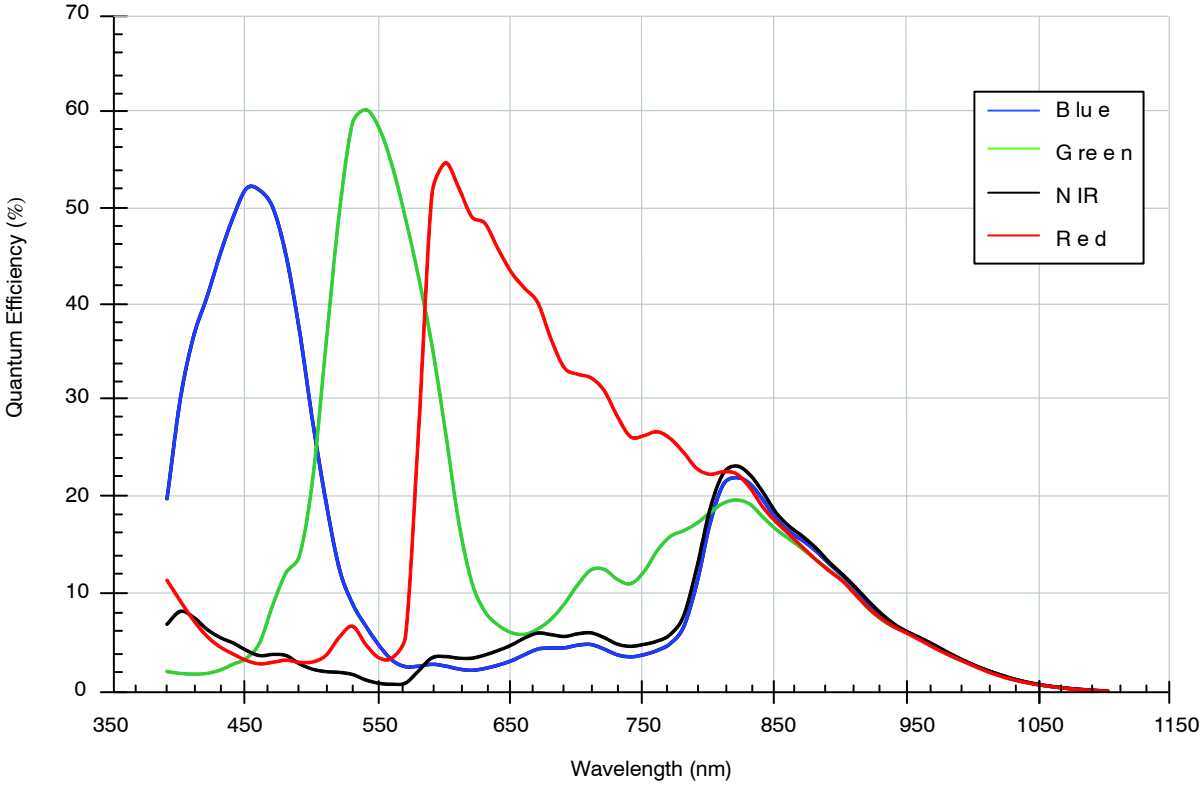


Figure 42. RGB−NIR Quantum Efficiency

CHIEF RAY ANGLE – 21 deg

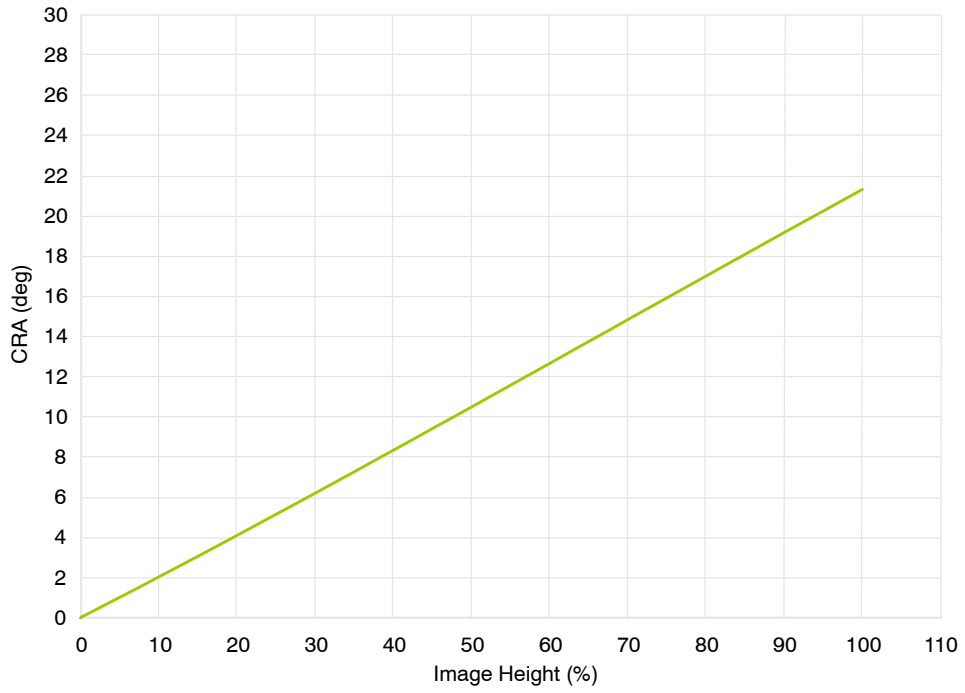


Figure 43. Chief Ray Angle – 21 deg

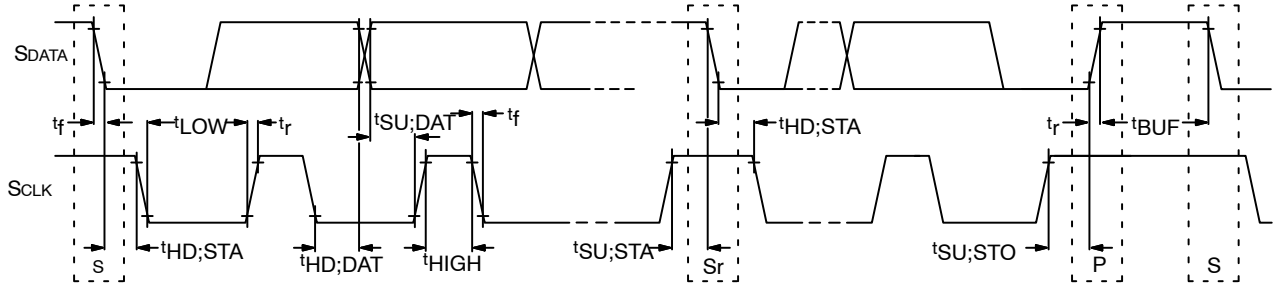
Image Height (%)	Image Height (mm)	CRA (deg)
0	0	0
5	0.113	1.01
10	0.226	2.03
15	0.340	3.07
20	0.453	4.11
25	0.566	5.17
30	0.679	6.23
35	0.792	7.30
40	0.906	8.38
45	1.019	9.46
50	1.132	10.54
55	1.245	11.63
60	1.358	12.73
65	1.472	13.82
70	1.585	14.92
75	1.698	16.01
80	1.811	17.10
85	1.925	18.19
90	2.038	19.28
95	2.151	20.36
100	2.264	21.43

ELECTRICAL SPECIFICATIONS

Unless otherwise stated, the following specifications apply under the following conditions:
V_{DD} = 1.8 V – 0.10 / +0.15; V_{DD}_IO = V_{DD}_PLL = V_{AA} = V_{AA}_PIX = 2.8 V ± 0.3 V;
V_{DD}_SLVS = 0.4 V – 0.1/+0.2; T_A = −30°C to +85°C;
output load = 10pF; frequency = 74.25 MHz; HiSPi off.

Two−Wire Serial Register Interface

The electrical characteristics of the two−wire serial register interface (SCLK, SDATA) are shown in Figure 44 and Table 18.



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

Figure 44. Two-Wire Serial Bus Timing Parameters

Table 18. TWO−WIRE SERIAL BUS CHARACTERISTICS

(^{f}EXTCLK = 27 MHz; V_{DD} = 1.8 V; V_{DD}_IO = 2.8 V; V_{AA} = 2.8 V; V_{AA}_PIX = 2.8 V; V_{DD}_PLL = 2.8 V; V_{DD}_DAC = 2.8 V; T_A = 25°C)

Parameter	Symbol	Standard Mode		Fast Mode		Unit
		Min	Max	Min	Max	
SCLK Clock Frequency	^{f}SCL	0	100	0	400	KHz
Hold Time (Repeated) START Condition						
After this Period, the First Clock Pulse is Generated	$t_{HD;STA}$	4.0	–	0.6	–	μs
LOW Period of the SCLK Clock	t_{LOW}	4.7	–	1.3	–	μs
HIGH Period of the SCLK Clock	t_{HIGH}	4.0	–	0.6	–	μs
Set−up Time for a Repeated START Condition	$t_{SU;STA}$	4.7	–	0.6	–	μS
Data Hold Time	$t_{HD;DAT}$	0 (Note 4)	3.45 (Note 5)	0 (Note 6)	0.9 (Note 5)	μs
Data Set−up Time	$t_{SU;DAT}$	250	–	100 (Note 6)	–	ns
Rise Time of both SDATA and SCLK Signals	t_r	–	1000	20 + 0.1Cb (Note 7)	300	ns
Fall Time of both SDATA and SCLK Signals	t_f	–	300	20 + 0.1Cb (Note 7)	300	ns
Set−up Time for STOP Condition	$t_{SU;STO}$	4.0	–	0.6	–	μs
Bus Free Time between a STOP and START Condition	t_{BUF}	4.7	–	1.3	–	μs
Capacitive Load for Each Bus Line	C_b	–	400	–	400	pF
Serial Interface Input Pin Capacitance	C_{IN_SI}	–	3.3	–	3.3	pF
SDATA Max Load Capacitance	C_{LOAD_SD}	–	30	–	30	pF
SDATA Pull−up Resistor	RSD	1.5	4.7	1.5	4.7	kΩ

1. This table is based on I^{2}C standard (v2.1 January 2000). ON Semiconductor.
2. Two−wire control is I^{2}C−compatible.
3. All values referred to V_{IHmin} = 0.9 VDD and V_{ILmax} = 0.1VDD levels. Sensor EXCLK = 27 MHz.
4. A device must internally provide a hold time of at least 300 ns for the SDATA signal to bridge the undefined region of the falling edge of SCLK.
5. The maximum $t_{HD;DAT}$ has only to be met if the device does not stretch the LOW period (^{t}LOW) of the SCLK signal.
6. A Fast−mode I^{2}C−bus device can be used in a Standard−mode I^{2}C−bus system, but the requirement $t_{SU;DAT}$ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCLK signal. If such a device does stretch the LOW period of the SCLK signal, it must output the next data bit to the SDATA line t_r max + $t_{SU;DAT}$ = 1000 + 250 = 1250 ns (according to the Standard−mode I^{2}C−bus specification) before the SCLK line is released.
7. C_b = total capacitance of one bus line in pF.

I/O Timing

By default, the AR0141CS launches pixel data, FV, and LV with the falling edge of PIXCLK. The expectation is that the user captures DOUT[11:0], FV, and LV using the rising edge of PIXCLK.

See Figure 45 for I/O timing diagram.

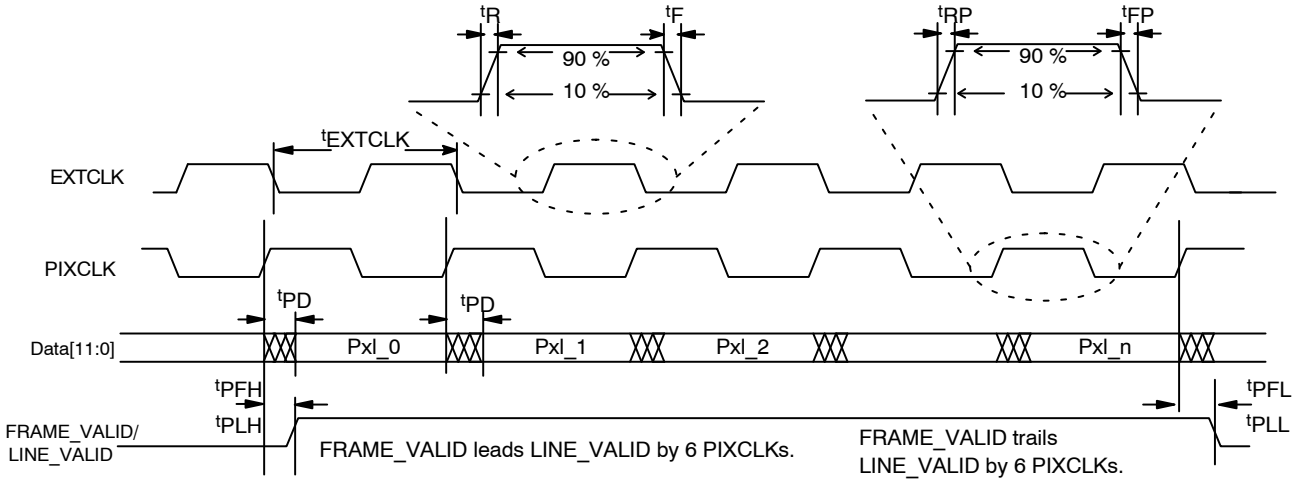


Figure 45. I/O Timing Diagram

Table 19. I/O TIMING CHARACTERISTICS (2.8 V V_{DD}_IO)

(Conditions: f_{PIXCLK} = 37.125 MHz (720P30fps; V_{DD}_IO = 2.8 V)

Symbol	Definition	Condition	Min	Typ	Max	Unit
$f_{EXTCLK1}$	Input clock frequency	PLL enabled	6	–	50	MHz
$t_{EXTCLK1}$	Input clock period	PLL enabled	20	–	166	ns
t_R	Input clock rise time		–	3	–	ns
t_F	Input clock fall time		–	3	–	ns
t_{RR}	PIXCLK rise time	PCLK slew rate setting = 2	2.0	3.5	6.4	ns
t_{FP}	PIXCLK fall time	PCLK slew rate setting = 2	1.9	3.3	6.2	ns
	Clock duty cycle		45	50	55	%
$t_{JITTER2}$	Input clock jitter at 27 MHz		–	–	600	ps
f_{PIXCLK}	PIXCLK frequency	default PLL configuration	6	37.125	74.25	MHz
t_{PD}	PIXCLK to Data[11:0]	PCLK slew rate setting = 2 parallel slew rate setting = 4	−2.0	–	5.9	ns
t_{PFH}	PIXCLK to FV high	PCLK slew rate setting = 2 parallel slew rate setting = 2	−0.9	–	4.4	ns
t_{PLH}	PIXCLK to LV high	PCLK slew rate setting = 2 parallel slew rate setting = 2	−0.8	–	4.6	ns
t_{PFL}	PIXCLK to FV low	PCLK slew rate setting = 2 parallel slew rate setting = 2	−1.5	–	3.1	ns
t_{PLL}	PIXCLK to FV low	PCLK slew rate setting = 2 parallel slew rate setting = 2	−1.5	–	3.3	ns
C_{LOAD}	Output load capacitance		–	30	–	pF
C_{IN}	Input pin capacitance		–	2.5	–	pF

1. Slew rate setting = 2 for PIXCLK
   Slew rate setting = 2 for parallel ports

Table 20. I/O TIMING CHARACTERISTICS (1.8 V V_{DD}_IO)

(Conditions: f_{PIXCLK} = 37.125 MHz (720P30fps; V_{DD}_IO = 1.8 V)

Symbol	Definition	Condition	Min	Typ	Max	Unit
$f_{EXTCLK1}$	Input clock frequency	PLL enabled	6	–	50	MHz
$f_{EXTCLK1}$	Input clock frequency	PLL enabled	6	–	50	MHz
$t_{EXTCLK1}$	Input clock period	PLL enabled	20	–	166.6666667	ns
t_R	Input clock rise time		–	3	–	ns
t_F	Input clock fall time		–	3	–	ns
t_{RR}	PIXCLK rise time	PCLK slew rate setting = 2	3.2	5.6	9.5	ns
t_{FP}	PIXCLK fall time	PCLK slew rate setting = 2	2.9	5.0	8.8	ns
	Clock duty cycle		45	50	55	%
$t_{JITTER2}$	Input clock jitter at 27 MHz		–	–	600	ps
f_{PIXCLK}	PIXCLK frequency	Default PLL configuration	6	37.125	74.25	MHz
t_{PD}	PIXCLK to Data[11:0]	PCLK slew rate setting = 2 Parallel slew rate setting = 2	–2.2	–	5.9	ns
t_{PFH}	PIXCLK to FV high	PCLK slew rate setting = 2 Parallel slew rate setting = 2	–0.9	–	4.5	ns
t_{PLH}	PIXCLK to LV high	PCLK slew rate setting = 2 Parallel slew rate setting = 2	–0.9	–	4.6	ns
t_{PFL}	PIXCLK to FV low	PCLK slew rate setting = 2 Parallel slew rate setting = 2	–1.7	–	3.1	ns
t_{PLL}	PIXCLK to FV low	PCLK slew rate setting = 2 Parallel slew rate setting = 2	–1.6	–	3.4	ns
C_{LOAD}	Output load capacitance		–	30	–	pF
C_{IN}	Input pin capacitance		–	2.5	–	pF

1. Slew rate setting = 2 for PIXCLK
   Slew rate setting = 2 for parallel ports

Table 21. I/O RISE SLEW RATE (2.8 V V_{DD}_IO)

Parallel Slew Rate (R0x306E[15:13])	Conditions	Min	Typ	Max	Units
7	Default	0.83	1.38	2.1	V/ns
6	Default	0.71	1.2	1.84	V/ns
5	Default	0.64	1.07	1.65	V/ns
4	Default	0.56	0.94	1.44	V/ns
3	Default	0.47	0.79	1.21	V/ns
2	Default	0.39	0.64	0.98	V/ns
1	Default	0.29	0.48	0.74	V/ns
0	Default	0.2	0.32	0.49	V/ns

1. 30pf loads at nominal voltages.

Table 22. I/O FALL SLEW RATE (2.8 V V_{DD}_IO)

Parallel Slew Rate (R0x306E[15:13])	Conditions	Min	Typ	Max	Units
7	Default	0.76	1.25	1.85	V/ns
6	Default	0.67	1.12	1.68	V/ns
5	Default	0.61	1.04	1.56	V/ns
4	Default	0.55	0.93	1.41	V/ns

Table 22. I/O FALL SLEW RATE (2.8 V V_{DD}_IO) (continued)

Parallel Slew Rate (R0x306E[15:13])	Conditions	Min	Typ	Max	Units
3	Default	0.48	0.81	1.23	V/ns
2	Default	0.4	0.67	1.03	V/ns
1	Default	0.31	0.52	0.79	V/ns
0	Default	0.21	0.35	0.54	V/ns

1. 30pf loads at nominal voltages.

Table 23. I/O RISE SLEW RATE (1.8 V V_{DD}_IO)

Parallel Slew Rate (R0x306E[15:13])	Conditions	Min	Typ	Max	Units
7	Default	0.32	0.51	0.85	V/ns
6	Default	0.28	0.44	0.75	V/ns
5	Default	0.25	0.4	0.68	V/ns
4	Default	0.23	0.36	0.6	V/ns
3	Default	0.2	0.31	0.51	V/ns
2	Default	0.17	0.26	0.41	V/ns
1	Default	0.13	0.2	0.32	V/ns
0	Default	0.09	0.13	0.21	V/ns

1. 30pf loads at nominal voltages.

Table 24. I/O FALL SLEW RATE (1.8 V V_{DD}_IO)

Parallel Slew Rate (R0x306E[15:13])	Conditions	Min	Typ	Max	Units
7	Default	0.32	0.53	0.87	V/ns
6	Default	0.28	0.47	0.77	V/ns
5	Default	0.26	0.43	0.71	V/ns
4	Default	0.24	0.39	0.64	V/ns
3	Default	0.21	0.34	0.56	V/ns
2	Default	0.18	0.29	0.47	V/ns
1	Default	0.14	0.22	0.36	V/ns
0	Default	0.1	0.16	0.25	V/ns

2. 30pf loads at nominal voltages.

DC Electrical Characteristics

The DC electrical characteristics are shown in the tables below.

Table 25. DC ELECTRICAL CHARACTERISTIC

Symbol	Definition	Condition	Min	Typ	Max	Unit
V_{DD}	Core digital voltage		1.7	1.8	1.95	V
V_{DD_IO}	I/O digital voltage		1.7/2.5	1.8/2.8	1.9/3.1	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
V_{DD_PLL}	PLL supply voltage		2.5	2.8	3.1	V
V_{DD_SLVS}	HiSPi supply voltage		0.3	0.4	0.6	V
V_{IH}	Input HIGH voltage		$V_{DD\_IO} \times 0.7$	–	–	V
V_{IL}	Input LOW voltage		–	–	$V_{DD\_IO} \times 0.3$	V
I_{IN}	Input leakage current	No pull−up resistor; V_{IN} = V_{DD_IO} or D_{GND}	20	–	–	μA
V_{OH}	Output HIGH voltage		V_{DD_IO} – 0.3	–	–	V
V_{OL}	Output LOW voltage		–	–	0.4	V
I_{OH}	Output HIGH current	At specified V_{OH}	−22	–	–	mA
I_{OL}	Output LOW current	At specified V_{OL}	–	–	22	mA

Product parametric performance is indicated in the Electrical Characteristics for the listed test conditions, unless otherwise noted. Product performance may not be indicated by the Electrical Characteristics if operated under different conditions.

Table 26. ABSOLUTE MAXIMUM RATINGS

Symbol	Definition	Condition	Typ	Max	Unit
V_{DD_MAX}	Core digital voltage		−0.3	2.4	V
$V_{DD\_IO\_MAX}$	I/O digital voltage		−0.3	4	V
V_{AA_MAX}	Analog voltage		−0.3	4	V
V_{AA_PIX}	Pixel supply voltage		−0.3	4	V
V_{DD_PLL}	PLL supply voltage		−0.3	4	V
VDD_SLVS_MAX	HiSPi I/O digital voltage		−0.3	2.4	V
t_{ST}	Storage temperature		−40	150	°C

Stresses exceeding those listed in the Maximum Ratings table may damage the device. If any of these limits are exceeded, device functionality should not be assumed, damage may occur and reliability may be affected.

Table 27. OPERATING CURRENT CONSUMPTION IN PARALLEL OUTPUT AND LINEAR MODE

Definition	Condition	Symbol	Min	Typ	Max	Unit
Digital Operating Current	Streaming,1280x720 60 fps	I_{DD1}	–	137	160	mA
I/O Digital Operating Current	Streaming,1280x720 60 fps	I_{DD_IO}	–	15	25	mA
Analog Operating Current	Streaming,1280x720 60 fps	I_{AA}	–	20	30	mA
Pixel Supply Current	Streaming,1280x720 60 fps	I_{AA_PIX}	–	1.5	3	mA
PLL Supply Current	Streaming,1280x720 60 fps	I_{DD_PLL}	–	4	8	mA

1. Operating currents are measured at the following conditions:
   V_{AA} = V_{AA_PIX} = V_{DD_PLL} = 2.8 V
   V_{DD} = V_{DD_IO} = 1.8 V; C_{LOAD} = 68pF
   PLL Enabled and PIXCLK = 74.25 MHz
   1x analog gain, 0.36 ms integration time, 60 fps, dark conditions
   T_J = 25°C

Table 28. OPERATING CURRENT IN HiSPi OUTPUT AND LINEAR MODE

Definition	Condition	Symbol	Min	Typ	Max	Unit
Digital Operating Current	Streaming,1280x720 60 fps	I_{DD}	–	147	170	mA
Analog operating current	Streaming,1280x720 60 fps	I_{AA}	–	20	30	mA
Pixel Supply Current	Streaming,1280x720 60 fps	I_{AA_PIX}	–	1.5	3	mA
PLL Supply Current	Streaming,1280x720 60 fps	I_{DD_PLL}	–	5	9	mA
SLVS Supply Current	Streaming,1280x720 60 fps	I_{DD_SLVS}	–	8	15	mA
HiVCM Supply Current	Streaming,1280x720 60 fps	I_{DD}	–	22	25	mA

1. V_{AA} = V_{AA_PIX} = V_{DD_PLL} = 2.8 V
   V_{DD} = V_{DD_IO} = 1.8 V
   V_{DD_SLVS} = 1.8 V for HiVCM and = 0.4 V for SLVS
   PLL Enabled and PIXCLK = 74.25 MHz
   1x analog gain, 0.36 ms integration time, 60 fps, dark conditions
   T_J = 25°C

Table 29. STANDBY CURRENT CONSUMPTION

Definition	Condition	Symbol	Min	Typ	Max	Unit
Soft Standby (Clock Off)	Analog, 2.8 V	–	–	0	0.1	mA
	Digital, 1.8 V	–	–	0.1	0.25	mA
Soft standby (Clock On)	Analog, 2.8 V	–	–	0.01	0.2	mA
	Digital, 1.8 V	–	–	26	30	mA

1. Analog = V_{AA} + V_{AA_PIX} + V_{DD_PLL}
2. Digital = V_{DD_IO} + V_{DD_SLVS}

HiSPi Electrical Specifications

NOTE: Refer to “High−Speed Serial Pixel Interface Physical Layer Specification v2.00.00” for further explanation of the HiSPi transmitter specification. The electrical specifications below supersede those given in the HiSPi Physical Layer Specification.

Table 30. SLVS POWER SUPPLY AND OPERATING TEMPERATURE

Parameter	Symbol	Min	Typ	Max	Unit
SLVS Current Consumption (Note 1, 2)	I_{DD_TX}			18	mA
HiSPi PHY Current Consumption (Note 1, 2)	I_{DD_HiSPi}			45	mA
Operating Temperature	T_A	–30		70	°C

1. Temperature of 25°C
2. Up to 600 Mbps

Table 31. SLVS ELECTRICAL DC SPECIFICATION

Parameter	Symbol	Min	Typ	Max	Unit
SLVS DC Mean Common Mode Voltage	V_{CM}	0.45 × V_{DD_TX}	0.5 × V_{DD_TX}	0.55 × V_{DD_TX}	V
SLVS DC Mean Differential Output Voltage	$\|V_{OD}\|$	0.36 × V_{DD_TX}	0.5 × V_{DD_TX}	0.64 × V_{DD_TX}	V
Change in V_{CM} between Logic 1 and 0	ΔV_{CM}			25	mV
Change in $\|V_{OD}\|$ between Logic 1 and 0	$\|V_{OD}\|$			25	mV
V_{OD} noise margin	NM			±30	%
Difference in V_{CM} between any Two Channels	$\|\Delta V_{CM}\|$			50	mV
Difference in V_{OD} between any Two Channels	$\|\Delta V_{OD}\|$			100	mV

Table 31. SLVS ELECTRICAL DC SPECIFICATION (continued)

Parameter	Symbol	Min	Typ	Max	Unit
Common−mode AC Voltage (pk) without VCM cap Termination	VCM_AC			50	mV
Common−mode AC Voltage (pk) with VCM Cap Termination	VCM_AC			30	mV
Maximum Overshoot Peak |VOD|	VOD_AC			1.3 × |VOD|	V
Maximum Overshoot Vdiff pk−pk	V_{diff_pkpk}			2.6 × OD	V
Single−ended Output Impedance	RO	35	50	70	Ω
Output Impedance Mismatch	ΔRO			20	%

Table 32. SLVS ELECTRICAL TIMING SPECIFICATION

Parameter	Symbol	Min	Max	Unit
Data Rate (Note 1)	1/UI	280	600	Mbps
Bitrate Period (Note 1)	tPW	1.43	3.57	ns
Max Setup Time from Transmitter (Note 1, 2)	tPRE	0.3		UI
Max Hold Time from Transmitter (Note 1, 2)	tPOST	0.3		UI
Eye Width (Note 1, 2)	tEYE		0.6	UI
Data Total Jitter (pk−pk) @1e−9 (Note 1, 2)	tTOTALJIT		0.2	UI
Clock Period Jitter (RMS) (Note 2)	tCKJIT		50	ps
Clock Cycle−to−Cycle Jitter (RMS) (Note 2)	tCYCJIT		100	ps
Rise Time (20% − 80%) (Note 3)	tR	150ps	0.25	UI
Fall Time (20% − 80%) (Note 3)	tF	150ps	0.25	UI
Clock Duty Cycle (Note 2)	DCYC	45	55	%
Mean Clock to Data Skew (Note 1, 4)	tCHSKEW	−0.1	0.1	UI
PHY−to−PHY Skew (Note 1, 5)	tPHYSKEW		2.1	UI
Mean Differential Skew (Note 6)	tDIFFSKEW	−100	100	ps

1. One UI is defined as the normalized mean time between one edge and the following edge of the clock.
2. Taken from the 0V crossing point with the DLL off.
3. Also defined with a maximum loading capacitance of 10 pF on any pin. The loading capacitance may also need to be less for higher bitrates so the rise and fall times do not exceed the maximum 0.3 UI.
4. The absolute mean skew between the Clock lane and any Data Lane in the same PHY between any edges.
5. The absolute skew between any Clock in one PHY and any Data lane in any other PHY between any edges.
6. Differential skew is defined as the skew between complementary outputs. It is measured as the absolute time between the two complementary edges at mean VCM point. Note that differential skew also is related to the ?VCM_AC spec, which also must not be exceeded.

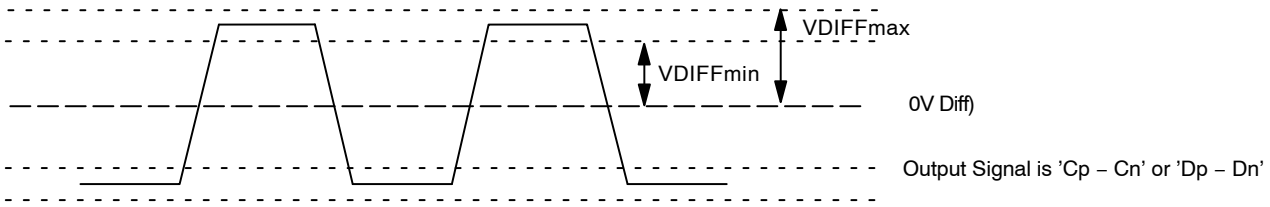


Figure 46. Differential Output Voltage for Clock or Data Pairs

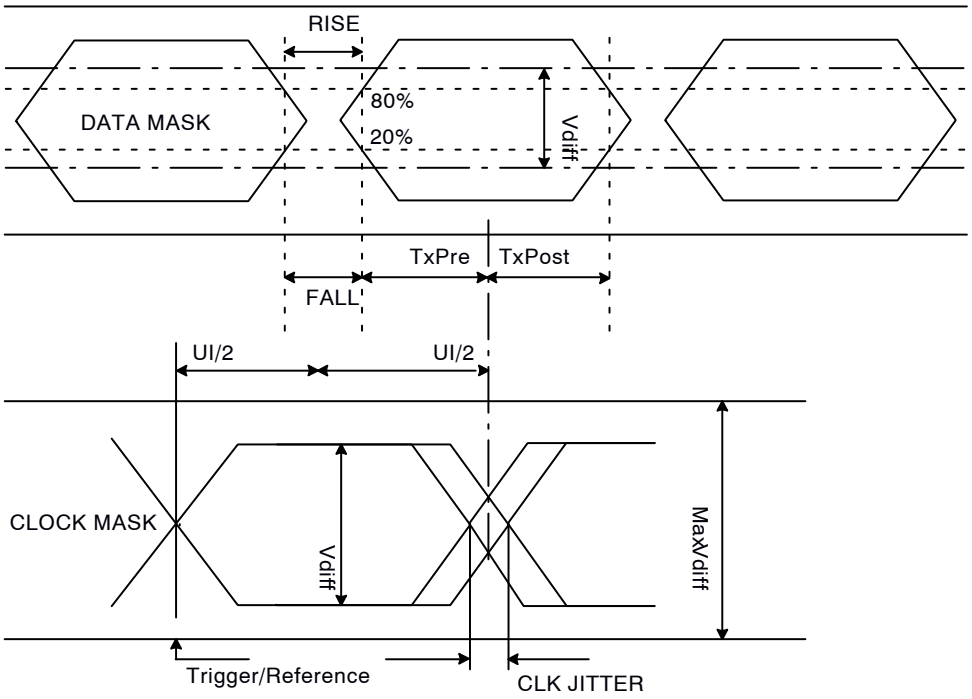


Figure 47. Eye Diagram for Clock and Data Signals

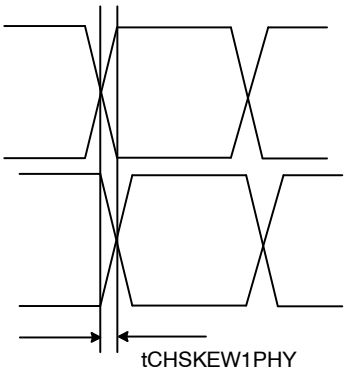


Figure 48. HiSPi Skew Between Data Signals Within the PHY

Table 33. CHANNEL, PHY, AND INTRA−PHY SKEW

(Measurement Conditions: V_{DD}_HiSPi = 1.8 V; V_{DD}_HiSPi_TX = 0.8 V; Data DLL set to 0)

Data Lane Skew in Reference to Clock	tCHSKEW1PHY	−150	ps

Table 34. CLOCK DLL STEPS

(Measurement Conditions: V_{DD}_HiSPi = 1.8 V; V_{DD}_HiSPi_TX = 0.8 V; Data DLL set to 0)

Clock DLL Step	1	2	3	4	5	Step
Delay at 660 Mbps	0.25	0.375	0.5	0.625	0.75	UI
Eye_opening at 660 Mbps	0.85	0.78	0.71	0.71	0.69	UI

1. The Clock DLL Steps 6 and 7 are not recommended by ON Semiconductor for the AR0141CS.

Table 35. DATA DLL STEPS

(Measurement Conditions: V_{DD}_HiSPi = 1.8 V; V_{DD}_HiSPi_TX = 0.8 V; Data DLL set to 0)

Data DLL Step	1	3	4	5	Step
Delay at 660 Mbps	0.25	0.375	0.625	0.875	UI
Eye opening at 660 Mbps	0.79	0.84	0.71	0.61	UI

1. The Data DLL Steps 3, 5, and 7 are not recommended by ON Semiconductor for the AR0141CS.

Power−Up Sequence

The recommended power−up sequence for the AR0141CS is shown in Figure 49. The available power supplies (VDD_IO, VDD, VDD_SLVS, VDD_PLL, VAA, VAA_PIX) must have the separation specified below.

1. Turn on VDD_PLL power supply
2. After 100 µs, turn on VAA and VAA_PIX power supply
3. After 100 µs, turn on VDD_IO power supply
4. After 100 µs, turn on VDD power supply
5. After 100 µs, turn on VDD_SLVS power supply
6. After the last power supply is stable, enable EXTCLK
7. Assert RESET_BAR for at least 1 ms. The parallel interface will be tri−stated during this time
8. Wait 1800 EXTCLKs for internal initialization into software standby
9. Initiate load of OTPM data by setting R0x304A = 0x0010
10. Wait for 185135 EXTCLKs for a full OTPM loading
11. Configure PLL, output, and image settings to desired values
12. Wait 1ms for the PLL to lock
13. Set streaming mode (R0x301A[2] = 1)

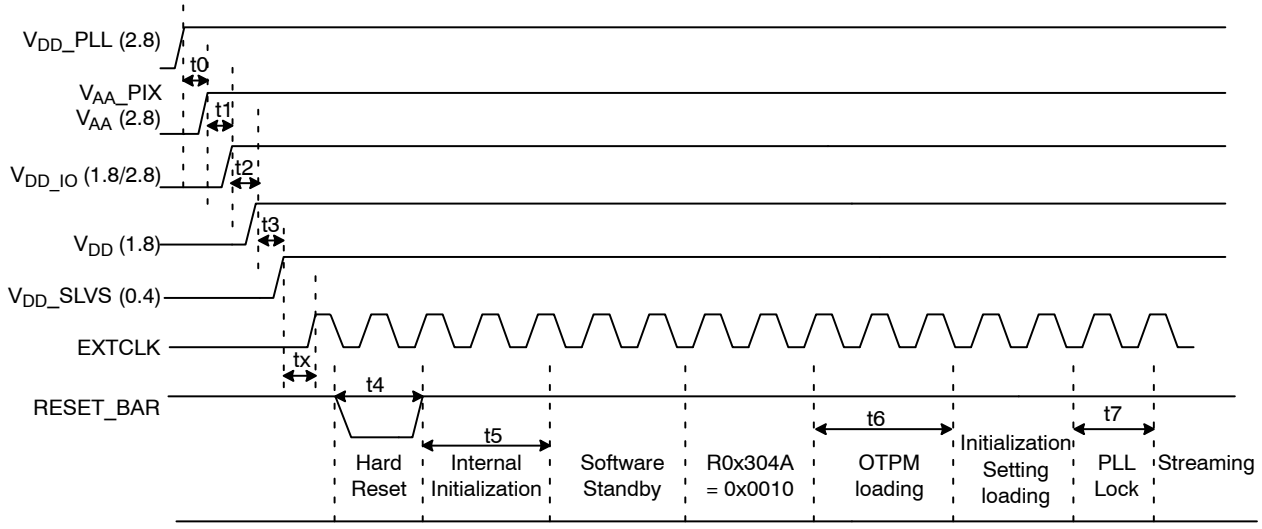


Figure 49. Power Up

Table 36. POWER−UP SEQUENCE

Definition	Symbol	Minimum	Typical	Maximum	Unit
VDD_PLL to VAA/VAA_PIX (Note 3)	t0	0	100	–	µs
VAA/VAA_PIX to VDD_IO	t1	0	100	–	µs
VDD_IO to VDD	t2	0	100	–	µs
VDD to VDD_SLVS	t3	0	100	–	µs
Xtal Settle Time	tx	–	30 (Note 1)	–	ms
Hard Reset	t4	1 (Note 2)	–	–	ms
Internal Initialization	t5	1800	–	–	EXTCLK
OTPM Loading	t6	185135	–	–	EXTCLK
PLL Lock Time	t7	1	–	–	ms

1. Xtal settling time is component−dependent, usually taking about 10 − 100 ms.
2. Hard reset time is the minimum time required after power rails are settled. In a circuit where Hard reset is held down by RC circuit, then the RC time must include the all power rail settle time and Xtal settle time.
3. It is critical that VDD_PLL is not powered up after the other power supplies. It must be powered before or at least at the same time as the others. If the case happens that VDD_PLL is powered after other supplies then sensor may have functionality issues and will experience high current draw on this supply.

Power−Down Sequence

The recommended power−down sequence for the AR0141CS is shown in Figure 50. The available power supplies (VDD_IO, VDD, VDD_SLVS, VDD_PLL, VAA, VAA_PIX) must have the separation specified below.

1. Disable streaming if output is active by setting standby R0x301a[2] = 0
2. The soft standby state is reached after the current row or frame, depending on configuration, has ended
3. Turn off VDD_SLVS
4. Turn off VDD
5. Turn off VDD_IO
6. Turn off VAA/VAA_PIX
7. Turn off VDD_PLL

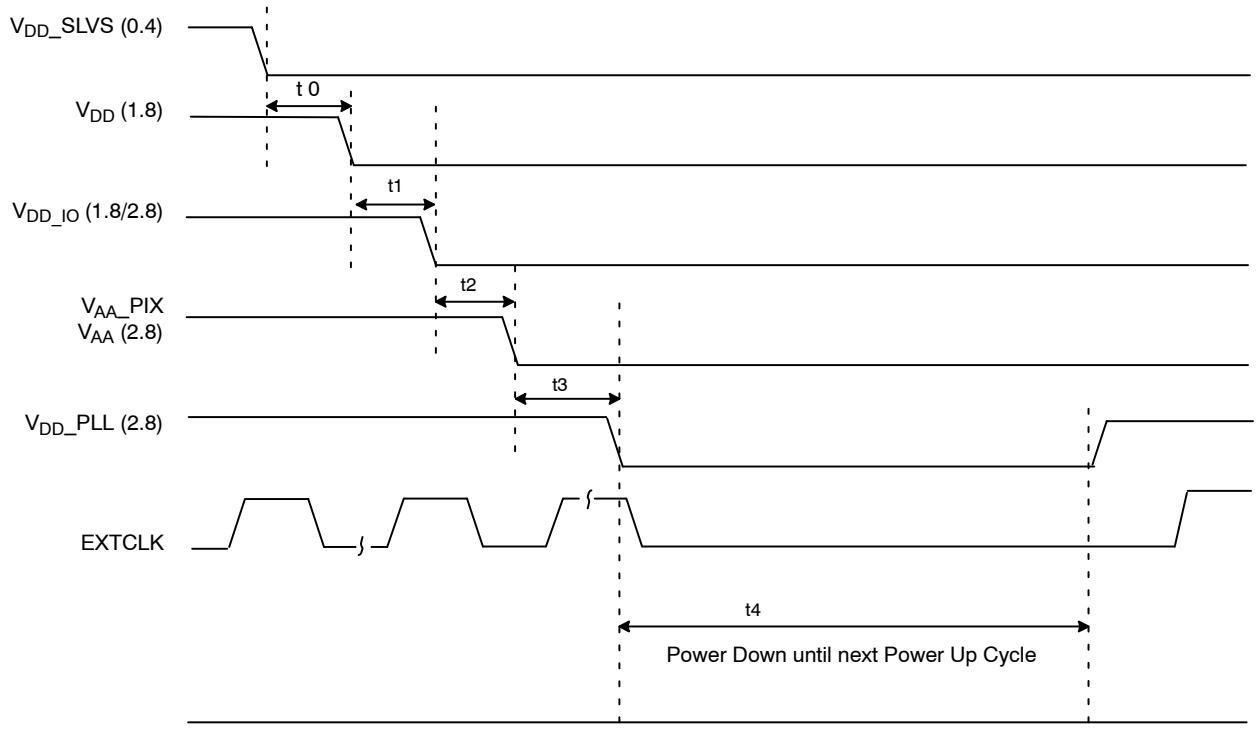


Figure 50. Power Down

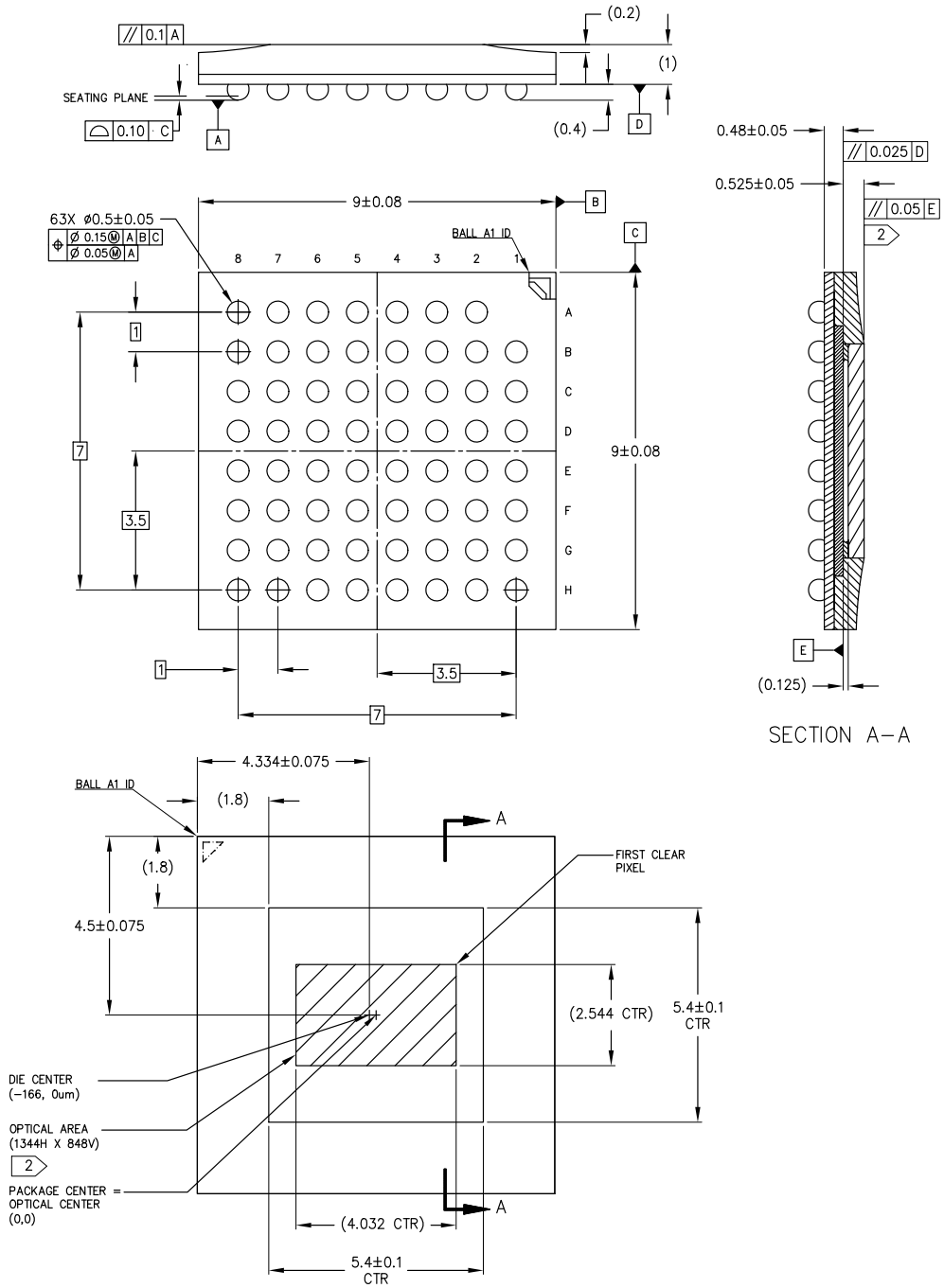
Table 37. POWER−DOWN SEQUENCE

Definition	Symbol	Minimum	Typical	Maximum	Unit
VDD_SLVS to VDD	t0	0	–	–	µs
VDD to VDD_IO	t1	0	–	–	µs
VDD_IO to VAA/VAA_PIX	t2	0	–	–	µs
VAA/VAA_PIX to VDD_PLL	t3	0	–	–	µs
PwrDn until Next PwrUp Time	t4	100	–	–	ms

1. t4 is required between power down and next power up time; all decoupling caps from regulators must be completely discharged.

PACKAGE DIMENSIONS

IBGA63 9x9
CASE 503AH
ISSUE O



NOTES	
1	DIMENSIONS IN MM. DIMENSIONS IN ( ) ARE FOR REFERENCE ONLY.
2	MAXIMUM ROTATION OF OPTICAL AREA RELATIVE TO PACKAGE EDGES: 1°. MAXIMUM TILT OF OPTICAL AREA RELATIVE TO SUBSTRATE PLANE [D]: 25 MICRONS MAXIMUM TILT OF COVER GLASS RELATIVE TO OPTICAL AREA PLANE [E]: 50 MICRONS.

PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:
Literature Distribution Center for ON Semiconductor
19521 E. 32nd Pkwy, Aurora, Colorado 80011 USA
Phone: 303−675−2175 or 800−344−3860 Toll Free USA/Canada
Fax: 303−675−2176 or 800−344−3867 Toll Free USA/Canada
Email: orderlit@onsemi.com

N. American Technical Support: 800−282−9855 Toll Free USA/Canada
Europe, Middle East and Africa Technical Support:
Phone: 421 33 790 2910
Japan Customer Focus Center
Phone: 81−3−5817−1050

ON Semiconductor Website: **www.onsemi.com**

Order Literature: http://www.onsemi.com/orderlit

For additional information, please contact your local Sales Representative

OCEAN CHIPS
Океан Электроники
Поставка электронных компонентов

Компания «Океан Электроники» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Поставка оригинальных импортных электронных компонентов напрямую с производств Америки, Европы и Азии, а так же с крупнейших складов мира;
- Широкая линейка поставок активных и пассивных импортных электронных компонентов (более 30 млн. наименований);
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Помощь Конструкторского Отдела и консультации квалифицированных инженеров;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Поставка электронных компонентов под контролем ВП;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- При необходимости вся продукция военного и аэрокосмического назначения проходит испытания и сертификацию в лаборатории (по согласованию с заказчиком);
- Поставка специализированных компонентов военного и аэрокосмического уровня качества (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Actel, Aeroflex, Peregrine, VPT, Syfer, Eurofarad, Texas Instruments, MS Kennedy, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Компания «Океан Электроники» является официальным дистрибьютором и эксклюзивным представителем в России одного из крупнейших производителей разъемов военного и аэрокосмического назначения «**JONHON**», а так же официальным дистрибьютором и эксклюзивным представителем в России производителя высокотехнологичных и надежных решений для передачи СВЧ сигналов «**FORSTAR**».

AVIC JONHON

«JONHON» (основан в 1970 г.)

Разъемы специального, военного и аэрокосмического назначения:

(Применяются в военной, авиационной, аэрокосмической, морской, железнодорожной, горно- и нефтедобывающей отраслях промышленности)

«FORSTAR» (основан в 1998 г.)

ВЧ соединители, коаксиальные кабели, кабельные сборки и микроволновые компоненты:

(Применяются в телекоммуникациях гражданского и специального назначения, в средствах связи, РЛС, а так же военной, авиационной и аэрокосмической отраслях промышленности).

FORSTAR®

Телефон: 8 (812) 309-75-97 (многоканальный)
Факс: 8 (812) 320-03-32
Электронная почта: ocean@oceanchips.ru
Web: http://oceanchips.ru/
Адрес: 198099, г. Санкт-Петербург, ул. Калинина, д. 2, корп. 4, лит. А