1/3-Inch CMOS Digital Image Sensor

AR0130 Data Sheet

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

Features

- Superior low-light performance both in VGA mode and HD mode
- Excellent Near IR performance
- HD video (720p60)
- On-chip AE and statistics engine
- Auto black level calibration
- Context switching
- Progressive Scan
- Supports 2:1 scaling
- Internal master clock generated by on-chip phase locked loop (PLL) oscillator.
- Parallel output

Applications

- Gaming systems
- Video surveillance
- 720p60 video applications

General Description

Aptina's AR0130 is a 1/3-inch CMOS digital image sensor with an active-pixel array of 1280H x 960V. It captures images with a rolling-shutter readout. It includes sophisticated camera functions such as auto exposure control, windowing, and both video and single frame modes. It is programmable through a simple two-wire serial interface. The AR0130 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 gaming systems, surveillance, and HD video.

Table 1: Key Parameters

Parameter	Typical Value
Optical format	1/3-inch (6 mm)
Active pixels	1280 x 960 = 1.2 Mp
Pixel size	3.75µm
Color filter array	Monochrome, RGB Bayer
Shutter type	Electronic rolling shutter
Input clock range	6 – 50 MHz
Output clock maximum	74.25 MHz

Table 1: Key Parameters (continued) (continued)

Parameter		Typical Value
Output	Parallel	12-bit
Max. Frame rates	1.2MP (full FOV)	45 fps
	720pHD (reduced FOV)	60 fps
	VGA (full FOV)	45 fps
	VGA (reduced FOV)	60 fps
	800x800 (reduced FOV)	60 fps
Responsivity at 550 nm		5.5 V/lux-sec
SNR_{MAX}		44 dB
Dynamic range		83.5 dB
Supply voltage	I/O	1.8 or 2.8V
	Digital	1.8 V
	Analog	2.8 V
Power consumption		270mW (1280x720 60fps)
Operating temperature (junction) -T_J		–30°C to + 70° C
Package option		Bare die, iLCC, PLCC

Ordering Information

Table 2: Available Part Numbers

Part Number	Description
AR0130CSSM00SUD20	Monochrome die
AR0130CSSC00SUD20	RGB Bayer die
AR0130CSSM00SPCA0	Monochrome 48-Pin iLCC
AR0130CSSC00SPCA0	RGB Bayer 48-Pin iLCC
AR0130CSSM00SPBA0	Monochrome 48-Pin PLCC
AR0130CSSC00SPBA0	RGB Bayer 48-Pin PLCC
AR0130CSSM00SPCAH	Monochrome headboard iLCC
AR0130CSSC00SPCAH	RGB Bayer headboard iLCC
AR0130CSSM00SPBAH	Monochrome headboard PLCC
AR0130CSSC00SPBAH	RGB Bayer headboard PLCC
AR0130CSSM00SPCAD	Monochrome demo kit iLCC
AR0130CSSC00SPCAD	RGB Bayer demo kit iLCC
AR0130CSSM00SPBAD	Monochrome demo kit PLCC
AR0130CSSC00SPBAD	RGB Bayer demo kit PLCC

Table of Contents

List of Figures

List of Tables

General Description

The Aptina™ AR0130 can be operated in its default mode or programmed for frame size, exposure, gain, and other parameters. The default mode output is a 960p-resolution image at 45 frames per second (fps). It outputs 12-bit raw data over the parallel port. The device may be operated in video (master) mode or in single frame trigger mode.

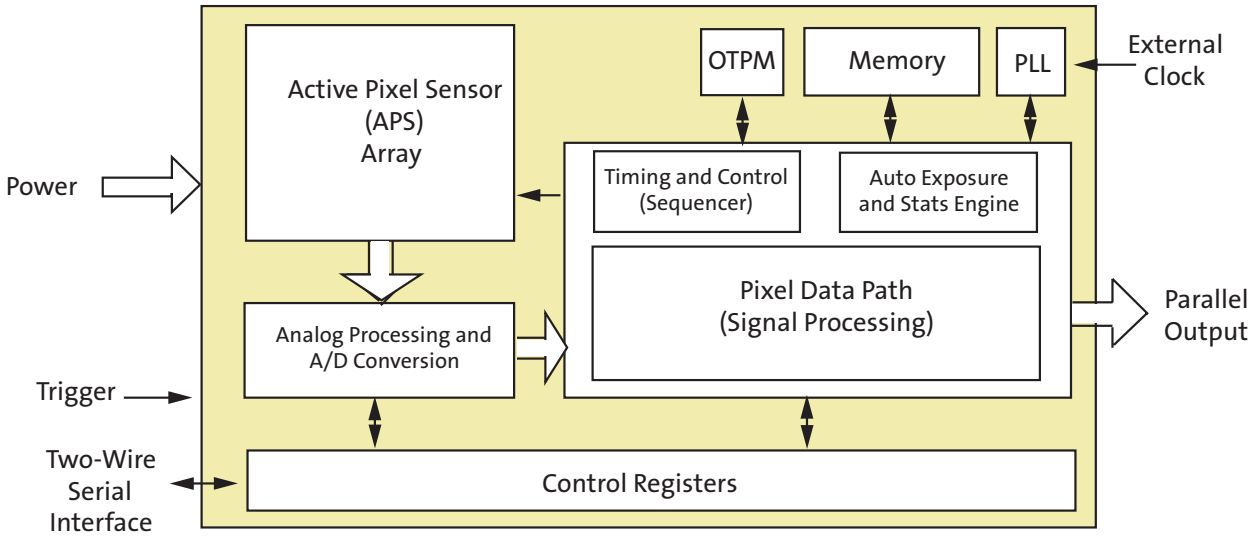
FRAME_VALID and LINE_VALID signals are output on dedicated pins, along with a synchronized pixel clock in parallel mode.

The AR0130 includes additional features to allow application-specific tuning: windowing and offset, adjustable auto-exposure control and auto black level correction. Optional register information and histogram statistic information can be embedded in first and last 2 lines of the image frame.

Functional Overview

The AR0130 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 74.25 Mp/s, corresponding to a clock rate of 74.25 MHz. Figure 1 shows a block diagram of the sensor.

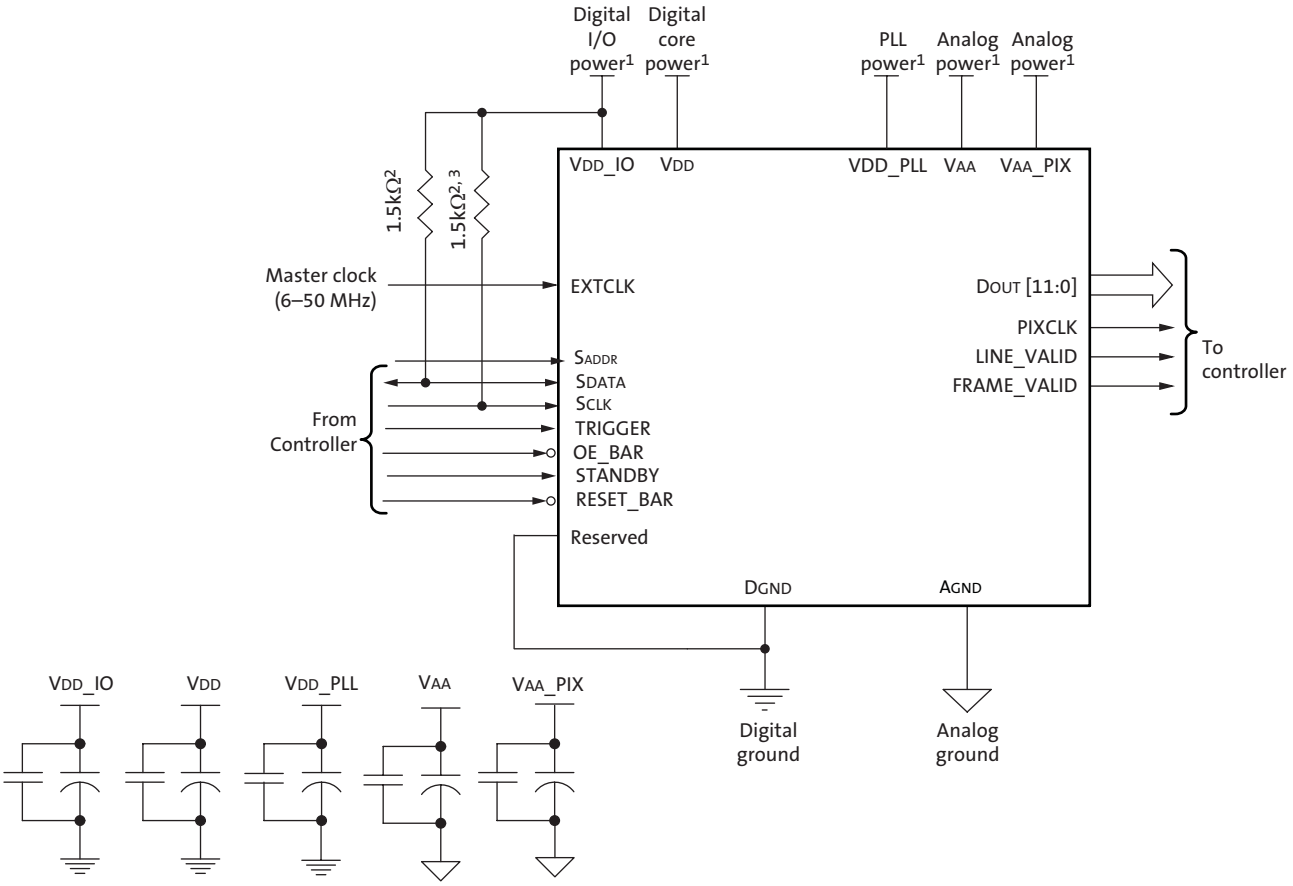
Figure 1: Block Diagram



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 1.2 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). The pixel data are output at a rate of up to 74.25 Mp/s, in parallel to frame and line synchronization signals.

Figure 2: Typical Configuration: Parallel Pixel Data Interface



Notes:
1. All power supplies must be adequately decoupled.
2. Aptina recommends a resistor value of 1.5kΩ, but a greater value may be used for slower two-wire speed.
3. This pull-up resistor is not required if the controller drives a valid logic level on SCLK at all times.
4. Aptina recommends that VDD_SLVS pad (only available in bare die) is left unconnected.
5. Aptina recommends that 0.1µF and 10µF 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 AR0130 demo headboard schematics for circuit recommendations.
6. Aptina recommends that analog power planes are placed in a manner such that coupling with the digital power planes is minimized.
7. I/O signals voltage must be configured to match VDD_IO voltage to minimize any leakage currents.

Table 3: Pad Descriptions

Name	Type	Description
STANDBY	Input	Standby-mode enable pin (active HIGH).
VDD_PLL	Power	PLL power.
VAA	Power	Analog power.
EXTCLK	Input	External input clock.
VDD_SLVS	Power	Digital power (do not connect).
DGND	Power	Digital ground.
VDD	Power	Digital power.
AGND	Power	Analog ground.
SADDR	Input	Two-Wire Serial Interface address select.
SCLK	Input	Two-Wire Serial Interface lcock input.
SDATA	I/O	Two-Wire Serial Interface data I/O.
VAA_PIX	Power	Pixel power.
LINE_VALID	Output	Asserted when DOUT line data is valid.
FRAME_VALID	Output	Asserted when DOUT frame data is valid.
PIXCLK	Output	Pixel clock out. DOUT is valid on rising edge of this clock.
VDD_IO	Power	I/O supply power.
DOUT8	Output	Parallel pixel data output.
DOUT9	Output	Parallel pixel data output.
DOUT10	Output	Parallel pixel data output.
DOUT11	Output	Parallel pixel data output (MSB)
Reserved	Input	Connect to DGND.
DOUT4	Output	Parallel pixel data output.
DOUT5	Output	Parallel pixel data output.
DOUT6	Output	Parallel pixel data output.
DOUT7	Output	Parallel pixel data output.
TRIGGER	Input	Exposure synchronization input.
OE_BAR	Input	Output enable (active LOW).
DOUT0	Output	Parallel pixel data output (LSB)
DOUT1	Output	Parallel pixel data output.
DOUT2	Output	Parallel pixel data output.
DOUT3	Output	Parallel pixel data output.
RESET_BAR	Input	Asynchronous reset (active LOW). All settings are restored to factory default.
FLASH	Output	Flash control output.
NC	Input	Do not connect.

Figure 3: 48-Pin iLCC Pinout Diagram

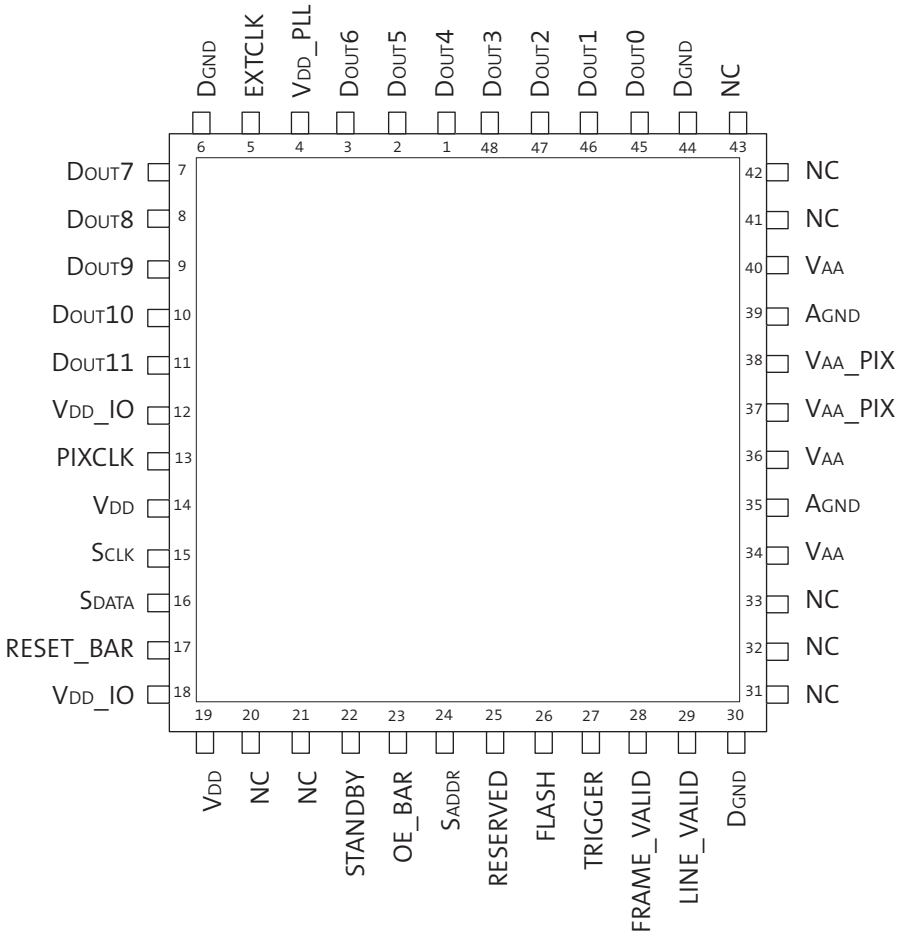
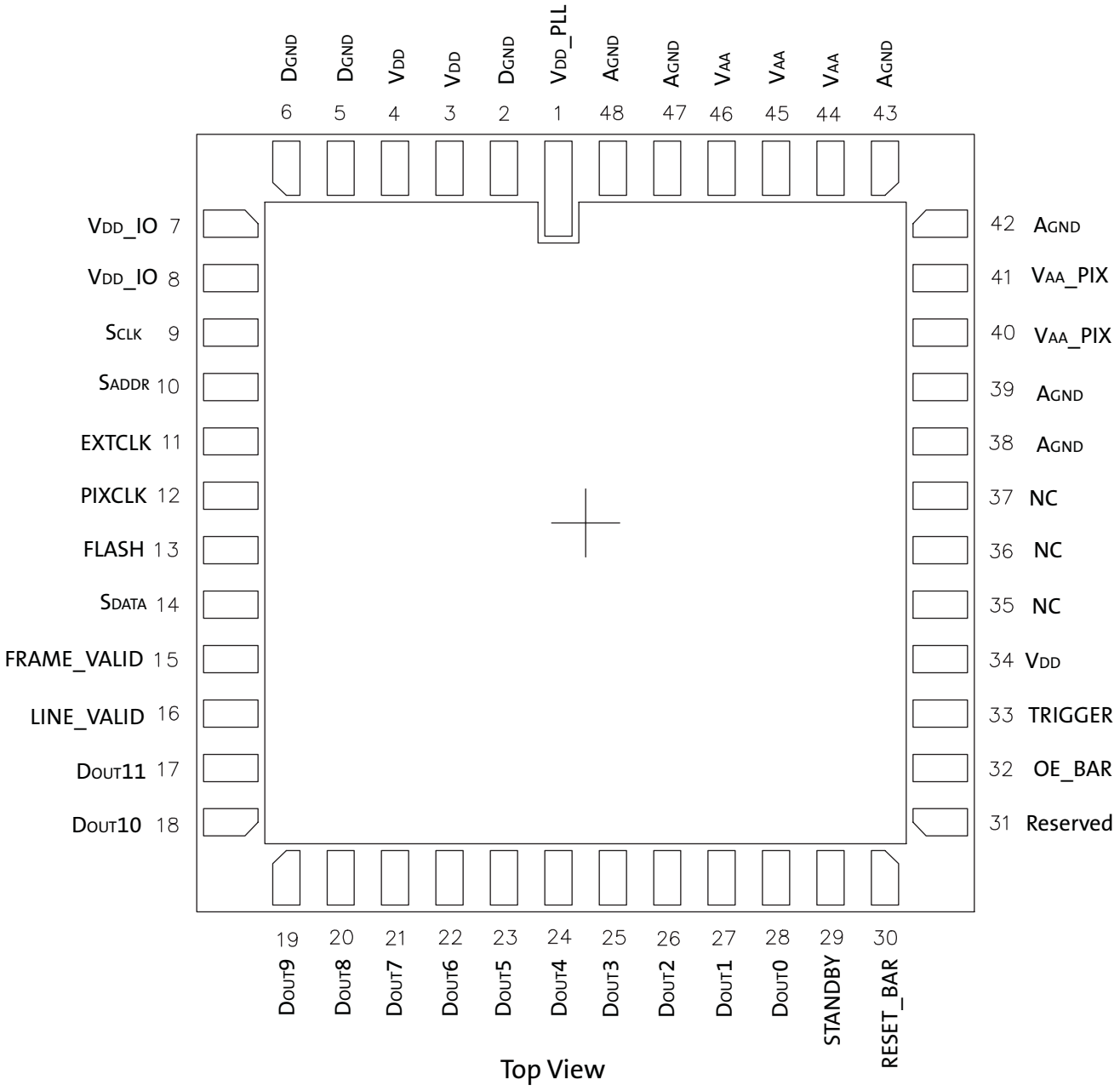


Figure 4: 48-Pin PLCC Pinout Diagram



Pixel Data Format

Pixel Array Structure

The AR0130 pixel array is configured as 1412 columns by 1028 rows, (see Figure 5). The dark pixels are optically black and are used internally to monitor black level. Of the right 108 columns, 64 are dark pixels used for row noise correction. Of the top 24 rows of pixels, 12 of the dark rows are used for black level correction. There are 1296 columns by 976 rows of optically active pixels. While the sensor's format is 1280 x 960, 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 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

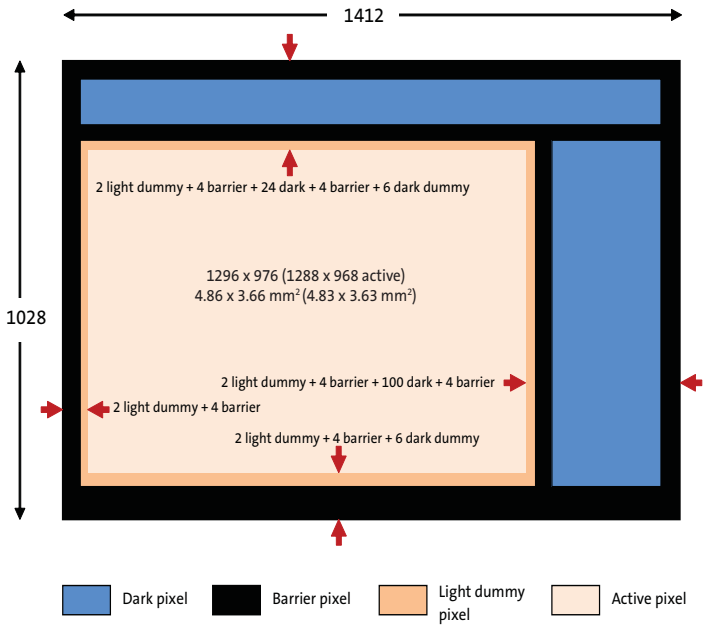
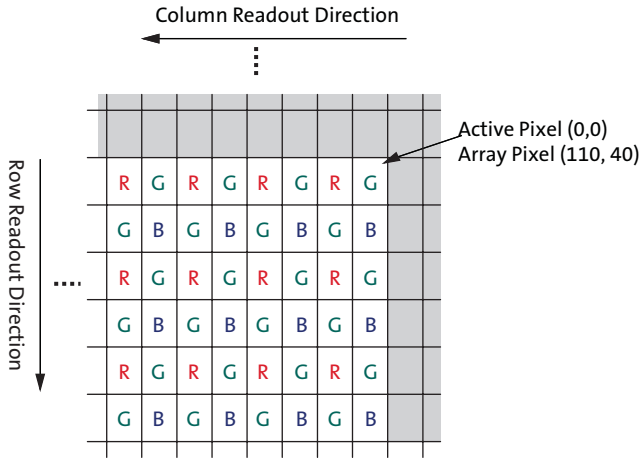


Figure 6: Pixel Color Pattern Detail (Top Right Corner)

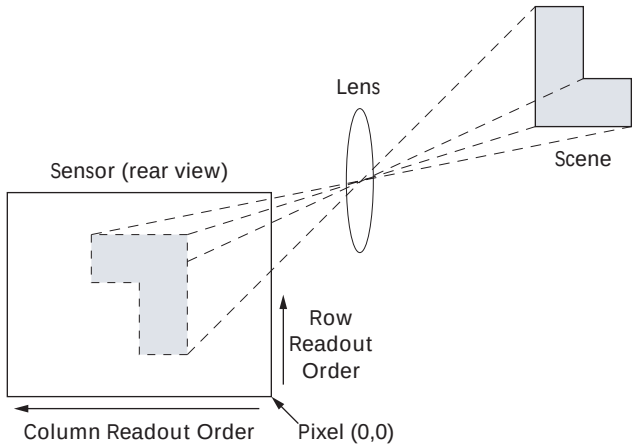


Default Readout Order

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 (110, 40).

When the sensor is imaging, the active surface of the sensor faces the scene as shown in Figure 7. 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 7 on page 12.

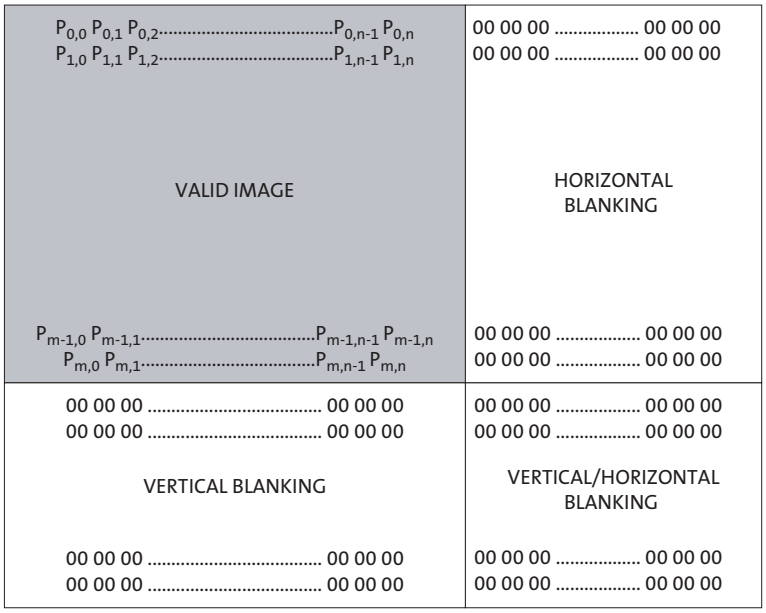
Figure 7: Imaging a Scene



Output Data Format

The AR0130 image data is read out in a progressive scan. Valid image data is surrounded by horizontal and vertical blanking (see Figure 8). The amount of horizontal row time (in clocks) is programmable through R0x300C. The amount of vertical frame time (in rows) is programmable through R0x300A. Line_Valid (LV) is HIGH during the shaded region of Figure 8. Optional Embedded Register setup information and Histogram statistic information are available in first 2 and last row of image data.

Figure 8: Spatial Illustration of Image Readout



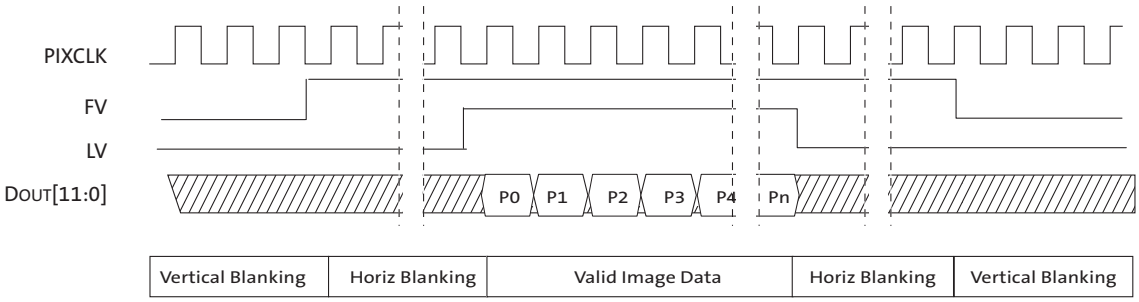
Readout Sequence

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

Parallel Output Data Timing

The output images are divided into frames, which are further divided into lines. By default, the sensor produces 968 rows of 1288 columns each. The FV and LV signals indicate the boundaries between frames and lines, respectively. PIXCLK can be used as a clock to latch the data. For each PIXCLK cycle, with respect to the falling edge, one 12-bit pixel datum outputs on the DOUT pins. When both FV and LV are asserted, the pixel is valid. PIXCLK cycles that occur when FV is de-asserted are called vertical blanking. PIXCLK cycles that occur when only LV is de-asserted are called horizontal blanking.

Figure 9: Default Pixel Output Timing



LV and FV

The timing of the FV and LV outputs is closely related to the row time and the frame time.

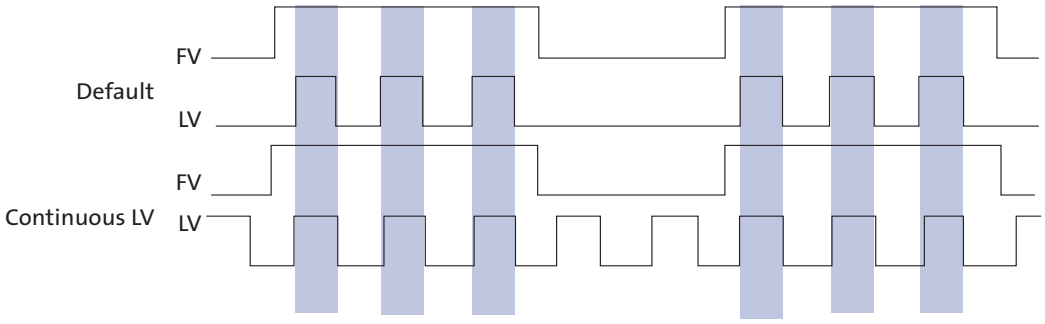
FV will be asserted for an integral number of row times, which will normally be equal to the height of the output image.

LV will be asserted during the valid pixels of each row. The leading edge of LV will be offset from the leading edge of FV by 6 PIXCLKs. Normally, LV will only be asserted if FV is asserted; this is configurable as described below.

LV Format Options

The default situation is for LV to be de-asserted when FV is de-asserted. By configuring R0x306E[1:0], the LV signal can take two different output formats. The formats for reading out four lines and two vertical blanking lines are shown in Figure 10.

Figure 10: LV Format Options



The timing of an entire frame is shown in Figure 11: “Line Timing and FRAME_VALID/LINE_VALID Signals,” on page 15.

Frame Time

The pixel clock (PIXCLK) represents the time needed to sample 1 pixel from the array. The sensor outputs data at the maximum rate of 1 pixel per PIXCLK. One row time (t_{ROW}) is the period from the first pixel output in a row to the first pixel output in the next row. The row time and frame time are defined by equations in Table 4.

Figure 11: Line Timing and FRAME_VALID/LINE_VALID Signals

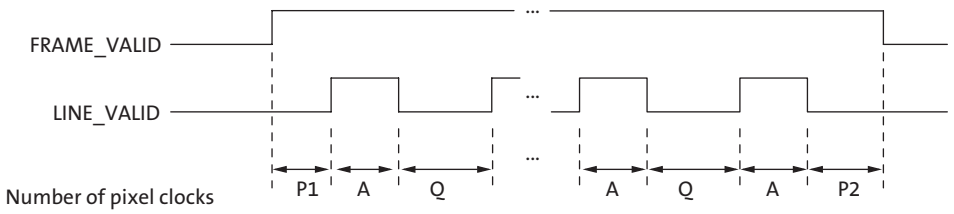


Table 4: Frame Time (Example Based on 1280 x 960, 45 Frames Per Second)

Parameter	Name	Equation	Timing at 74.25 MHz
A	Active data time	Context A: R0x3008 - R0x3004 + 1 Context B: R0x308E - R0x308A + 1	1280 pixel clocks = 17.23µs
P1	Frame start blanking	6 (fixed)	6 pixel clocks = 0.08µs
P2	Frame end blanking	6 (fixed)	6 pixel clocks = 0.08µs
Q	Horizontal blanking	R0x300C - A	370 pixel clocks = 4.98µs
A+Q (tROW)	Line (Row) time	R0x300C	1650 pixel clocks = 22.22µs
V	Vertical blanking	Context A: (R0x300A-(R0x3006-R0x3002+1)) x (A + Q) Context B: ((R0x30AA-(R0x3090-R0x308C+1)) x (A + Q)	49,500 pixel clocks = 666.66µs
Nrows x (t_{ROW})	Frame valid time	Context A: ((R0x3006-R0x3002+1)*(A+Q))-Q+P1+P2 Context B: ((R0x3090-R0x308C+1)*(A+Q))-Q+P1+P2	1,583,642 pixel clocks = 21.33ms
F	Total frame time	V + (Nrows x (A + Q))	1,633,500 pixel clocks = 22.22ms

Sensor timing is shown in terms of pixel clock cycles (see Figure 8 on page 13). The recommended pixel clock frequency is 74.25 MHz. The vertical blanking and the total frame time equations assume that the integration time (coarse integration time plus fine integration time) is less than the number of active lines plus the blanking lines:

Window Height + Vertical Blanking *(EQ 1)*

If this is not the case, the number of integration lines must be used instead to determine the frame time, (see Table 5). In this example, it is assumed that the coarse integration time control is programmed with 2000 rows and the fine shutter width total is zero.

For Master mode, if the integration time registers exceed the total readout time, then the vertical blanking time is internally extended automatically to adjust for the additional integration time required. This extended value is **not** written back to the frame_length_lines register. The frame_length_lines register can be used to adjust frame-to-frame readout time. This register does not affect the exposure time but it may extend the readout time.

Table 5: Frame Time: Long Integration Time

Parameter	Name	Equation (Number of Pixel Clock Cycles)	Default Timing at 74.25 MHz
F'	Total frame time (long integration time)	Context A: (R0x3012 x (A + Q)) + R0x3014 + P1 + P2 Context B: (R0x3016 x (A + Q)) + V R0x3018 + P1 + P2	3,300,012 pixel clocks = 44.44ms

Note: The AR0130 uses column parallel analog-digital converters; thus short line timing is not possible. The minimum total line time is 1430 columns (horizontal width + horizontal blanking). The minimum horizontal blanking is 150.

Exposure

Total integration time is the result of Coarse_Integration_Time and Fine_Integration_Time registers, and depends also on whether manual or automatic exposure is selected.

The actual total integration time, t_{INT} is defined as:

$$t_{INT} = t_{INTCoarse} + t_{INTFine} \quad \text{(EQ 2)}$$

= (number of lines of integration x line time) + (number of pixels of integration x pixel time)

where:

- Number of Lines of Integration (Auto Exposure Control: Enabled)
  When automatic exposure control (AEC) is enabled, the number of lines of integration may vary from frame to frame, with the limits controlled by R0x311E (minimum auto exposure time) and R0x311C (maximum auto exposure time).
- Number of Lines of Integration (Auto Exposure Control: Disabled)
  If AEC is disabled, the number of lines of integration equals the value in R0x3012 (context A) or R0x3016 (context B).
- Number of Pixels of Integration
  The number of fine shutter width pixels is independent of AEC mode (enabled or disabled):
  - Context A: the number of pixels of integration equals the value in R0x3014.
  - Context B: the number of pixels of integration equals the value in R0x3018.

Typically, the value of the Coarse_Integration_Time register is limited to the number of lines per frame (which includes vertical blanking lines), such that the frame rate is not affected by the integration time. For more information on coarse and fine integration time settings limits, please refer to the Register Reference document.

Real-Time Context Switching

In the AR0130, the user may switch between two full register sets (listed in Table 6) by writing to a context switch change bit in R0x30B0[13]. This context switch will change all registers (no shadowing) at the frame start time and have the new values apply to the immediate next exposure and readout time.

Table 6: Real-Time Context-Switchable Registers

Register Description	Register Number	
	Context A	Context B
Y_Addr_Start	R0x3002	R0x308C
X_Addr_Start	R0x3004	R0x308A
Y_Addr_End	R0x3006	R0x3090
X_Addr_End	R0x3008	R0x308E
Coarse_Integration_Time	R0x3012	R0x3016
Fine_Integration_Time	R0x3014	R0x3018
Y_Odd_Inc	R0x30A6	R0x30A8
Green1_Gain (GreenR)	R0x3056	R0x30BC
Blue_Gain	R0x3058	R0x30BE
Red_Gain	R0x305A	R0x30C0
Green2_Gain (GreenB)	R0x305C	R0x30C2
Global_Gain	R0x305E	R0x30C4
Analog Gain	R0x30B0[5:4]	R0x30B0[9:8]
Frame_Length_Lines	R0x300A	R0x30AA
Digital_Binning	R0x3032[1:0]	R0x3032[5:4]

Features

See the AR0130 Register Reference for additional details.

Reset

The AR0130 may be reset by using RESET_BAR (active LOW) or the reset register.

Hard Reset of Logic

The RESET_BAR pin can be connected to an external RC circuit for simplicity. The recommended RC circuit uses a 10kΩ resistor and a 0.1µF capacitor. The rise time for the RC circuit is 1µs maximum.

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 while preserving the existing two-wire serial interface configuration. 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

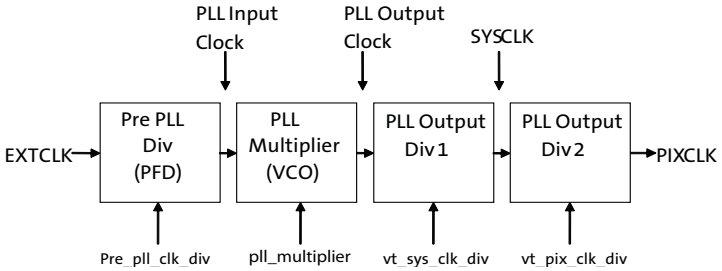
The AR0130 requires one clock input (EXTCLK).

PLL-Generated Master Clock

The PLL contains a prescaler to divide the input clock applied on EXTCLK, a VCO to multiply the prescaler output, and two divider stages to generate the output clock. The clocking structure is shown in Figure 12. PLL control registers can be programmed to generate desired master clock frequency.

Note: The PLL control registers must be programmed while the sensor is in the software Standby state. The effect of programming the PLL divisors while the sensor is in the streaming state is undefined.

Figure 12: PLL-Generated Master Clock PLL Setup



The PLL is enabled by default on the AR0130.

To configure and use the PLL:

1. Bring the AR0130 up as normal; make sure that ^{f}EXTCLK is between 6 and 50MHz and ensure the sensor is in software standby (R0x301A[2]= 0). PLL control registers must be set in software standby.
2. Set pll_multiplier, pre_pll_clk_div, vt_sys_clk_div, and vt_pix_clk_div based on the desired input (f_{EXTCLK}) and output (f_{PIXCLK}) frequencies. Determine the M, N, P1, and P2 values to achieve the desired f_{PIXCLK} using this formula:

   $f_{PIXCLK} = (f_{EXTCLK} \times M) / (N \times P1 \times P2)$
   where
   M = PLL_Multiplier (R0x3030)
   N = Pre_PLL_Clk_Div (R0x302E)
   P1 = Vt_Sys_Clk_Div (R0x302C)
   P2 = Vt_PIX_Clk_Div (R0x302A)
3. Wait 1ms to ensure that the VCO has locked.
4. Set R0x301A[2]=1 to enable streaming and to switch from EXTCLK to the PLL-generated clock.

Notes:

1. The PLL can be bypassed at any time (sensor will run directly off EXTCLK) by setting R0x30B0[15]=1. However, only the parallel data interface is supported with the PLL bypassed. The PLL is always bypassed in software standby mode. To disable the PLL, the sensor must be in standby mode (R0x301A[2] = 0)
2. The following restrictions apply to the PLL tuning parameters:

$$32 \le M \le 384$$

$$1 \le N \le 64$$

$$1 \le P1 \le 16$$

$4 \leq P2 \leq 16$

Additionally, the VCO frequency, defined as $f_{VCO} = f_{EXTCLK} \times M/N$ must be within 384-768MHz.

The user can utilize the Register Wizard tool accompanying DevWare to generate PLL settings given a supplied input clock and desired output frequency.

Spread-Spectrum Clocking

To facilitate improved EMI performance, the external clock input allows for spread spectrum sources, with no impact on image quality. Limits of the spread spectrum input clock are:

- 5% maximum clock modulation
- 35 KHz maximum modulation frequency
- Accepts triangle wave modulation, as well as sine or modified triangle modulations.

Stream/Standby Control

The sensor supports two standby modes: Hard Standby and Soft Standby. In both modes, external clock can be optionally disabled to further minimize power consumption. If this is done, then the “Power-Up Sequence” on page 44 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 (default behavior) or after the completion of the current row readout. When the sensor comes back from Soft Standby, previously written register settings are still maintained.

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

Entering Soft Standby:

1. R0x301A[12] = 1 if serial mode was used
2. Set R0x301A[2] = 0
3. External clock can be turned off to further minimize power consumption (Optional)

Exiting Soft Standby:

1. Enable external clock if it was turned off
2. R0x301A[2] = 1
3. R0x301A[12] = 0 if serial mode is used

Hard Standby

Hard Standby puts the sensor in lower power state but will not maintain register settings. When the sensor comes back from Hard Standby, all register settings are set back to default values.

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

Entering Hard Standby:

1. R0x301A[8] = 1
2. R0x301A[12] = 1 if serial mode was used
3. Assert STANDBY pin
4. External clock can be turned off to further minimize power consumption (Optional)

Exiting Hard Standby:

1. Enable external clock if it was turned off
2. De-assert STANDBY pin
3. Set R0x301A[8] = 0
4. Write all required register settings before entering streaming

Window Control

Registers x_addr_start, x_addr_end, y_addr_start, and y_addr_end control the size and starting coordinates of the image window.

The exact window height and width out of the sensor is determined by the difference between the Y address start and end registers or the X address start and end registers, respectively.

The AR0130 allows different window sizes for context A and context B.

Blanking Control

Horizontal blank and vertical blank times are controlled by the line_length_pck and frame_length_lines registers, respectively.

- Horizontal blanking is specified in terms of pixel clocks. It is calculated by subtracting the X window size from the line_length_pck register. The minimum horizontal blanking is 150 pixel clocks.
- Vertical blanking is specified in terms of numbers of lines. It is calculated by subtracting the Y window size from the frame_length_lines register. The minimum vertical blanking is 26 lines.

The actual imager timing can be calculated using Table 4 on page 15 and Table 5 on page 16, which describe the Line Timing and FV/LV signals.

Readout Modes

Digital Binning

By default, the resolution of the output image is the full width and height of the FOV as defined above. The output resolution can be reduced by digital binning. For RGB and monochrome mode, this is set by the register R0x3032. For Context A, use bits [1:0], for Context B, use bits [5:4]. Available settings are:

00 = No binning

01 = Horizontal binning

10 = Horizontal and vertical binning

Binning gives the advantage of reducing noise at the cost of reduced resolution. When both horizontal and vertical binning are used, a 2x improvement in SNR is achieved therefore improving low light performance

Bayer Space Resampling

All of the pixels in the FOV contribute to the output image in digital binning mode. This can result in a more pleasing output image with reduced subsampling artifacts. It also improves low-light performance. For RGB and monochrome mode, the digital binning factor is set by the register DIGITAL_BINNING (R0x3032). For Context A, use bits [1:0],

for Context B, use bits [5:4]. Available settings are: 00 = No binning; 01 = Horizontal binning; 10 = Horizontal and vertical binning. For RGB mode, resampling can be enabled by setting of register 0x306E[4] = 1.

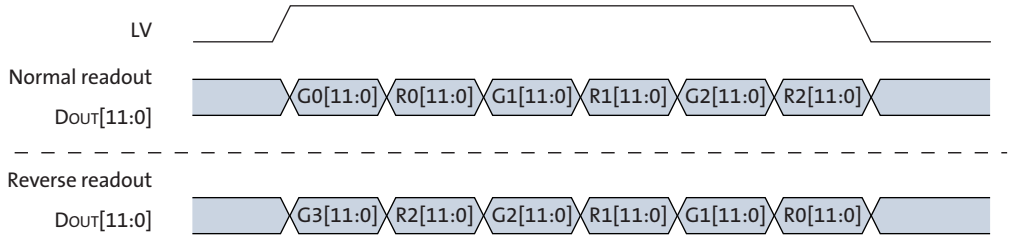
Mirror

Column Mirror Image

By setting R0x3040[14] = 1, the readout order of the columns is reversed, as shown in Figure 13. The starting color, and therefore the Bayer pattern, is preserved when mirroring the columns.

When using horizontal mirror mode, the user must retrigger column correction. Please refer to the column correction section to see the procedure for column correction retriggering. Bayer resampling must be enabled, by setting bit 4 of register 0 x 306E[4] = 1.

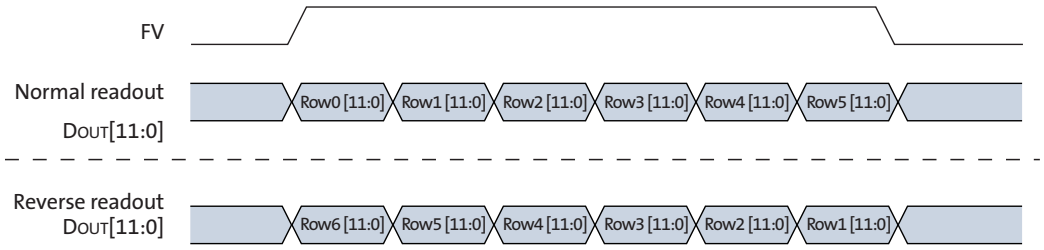
Figure 13: Six Pixels in Normal and Column Mirror Readout Modes



Row Mirror Image

By setting R0x3040[15] = 1, the readout order of the rows is reversed as shown in Figure 14. The starting Bayer color pixel is maintained in this mode by a 1-pixel shift in the imaging array. When using horizontal mirror mode, the user must retrigger column correction. Please refer to the column correction section to see the procedure for column correction retriggering.

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



Maintaining a Constant Frame Rate

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

By default, the following register fields cause a "bubble" in the output rate (that is, the vertical blank increases for one frame) if they are written in video mode, even if the new value would not change the resulting frame rate. The following list shows only a few examples of such registers; a full listing can be seen in the AR0130 Register Reference.

- x_addr_start
- x_addr_end
- y_addr_start
- y_addr_end
- frame_length_lines
- line_length_pclk
- coarse_integration_time
- fine_integration_time
- read_mode

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

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

Synchronizing Register Writes to Frame Boundaries

Changes to most register fields that affect the size or brightness of an image take effect on the frame after the one during which they are written. These fields are noted as "synchronized to frame boundaries" in the AR0130 Register Reference. To ensure that a register update takes effect on the next frame, the write operation must be completed after the leading edge of FV and before the trailing edge of FV.

As a special case, in single frame mode, register writes that occur after FV but before the next trigger will take effect immediately on the next frame, as if there had been a Restart. However, if the trigger for the next frame occurs during FV, register writes take effect as with video mode.

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

Restart

To restart the AR0130 at any time during the operation of the sensor, write a "1" to the Restart register (R0x301A[1] = 1). This has two effects: first, the current frame is interrupted immediately. Second, any writes to frame-synchronized registers and the shutter width registers take effect immediately, and a new frame starts (in video mode). The current row completes before the new frame is started, so the time between issuing the Restart and the beginning of the next frame can vary by about ^{t}ROW.

Image Acquisition Modes

The AR0130 supports two image acquisition modes: video and single frame.

Video

The video mode takes pictures by scanning the rows of the sensor twice. On the first scan, each row is released from reset, starting the exposure. On the second scan, the row is sampled, processed, and returned to the reset state. The exposure for any row is therefore the time between the first and second scans. Each row is exposed for the same duration, but at slightly different point in time, which can cause a shear in moving subjects as is typical with electronic rolling shutter sensors.

Single Frame

The single-frame mode operates similar to the video mode. It also scans the rows of the sensor twice, first to reset the rows and second to read the rows. Unlike video mode where a continuous stream of images are output from the image sensor, the single-frame mode outputs a single frame in response to a high state placed on the TRIGGER input pin. As long as the TRIGGER pin is held in a high state, new images will be read out. After the TRIGGER pin is returned to a low state, the image sensor will not output any new images and will wait for the next high state on the TRIGGER pin.

The TRIGGER pin state is detected during the vertical blanking period (i.e. the FV signal is low). The pin is level sensitive rather than edge sensitive. As such, image integration will only begin when the sensor detects that the TRIGGER pin has been held high for 3 consecutive clock cycles. During integration time of single-frame mode and video mode, the FLASH output pin is at high.

Stereo Timing Synchronization

In certain applications, multiple sensors need to have their video streams synchronized (E.g. surround view or panorama view applications). The TRIGGER pin can also be used to synchronize output of multiple image sensors together and still get a video stream. This is called continuous trigger mode. Continuous trigger is enabled by holding the TRIGGER pin high. Alternatively, the TRIGGER pin can be held high until the stream bit is enabled (R0x301A[2]=1) then can be released for continuous synchronized video streaming.

If the TRIGGER pins for all connected AR0130 sensors are connected to the same control signal, all sensors will receive the trigger pulse at the same time. If they are configured to have the same frame timing, then the usage of the TRIGGER pin guarantees that all sensors will be synchronized within 1 PIXCLK cycle if PLL is disabled, or 2 PIXCLK cycles if PLL is enabled.

With continuous trigger mode, the application can now make use of the video streaming mode while guaranteeing that all sensor outputs are synchronized. As long as the initial trigger for the sensors takes place at the same time, all subsequent video streams will be synchronous.

Automatic Exposure Control

The integrated automatic exposure control (AEC) is responsible for ensuring that optimal settings of exposure and gain are computed and updated every other frame. AEC can be enabled or disabled by R0x3100[0].

When AEC is disabled (R0x3100[0] = 0), the sensor uses the manual exposure value in coarse and fine shutter width registers and the manual gain value in the gain registers.

When AEC is enabled (R0x3100[0]=1), the target luma value is set by R0x3102. For the AR0130 this target luma has a default value of 0x0800 or about half scale.

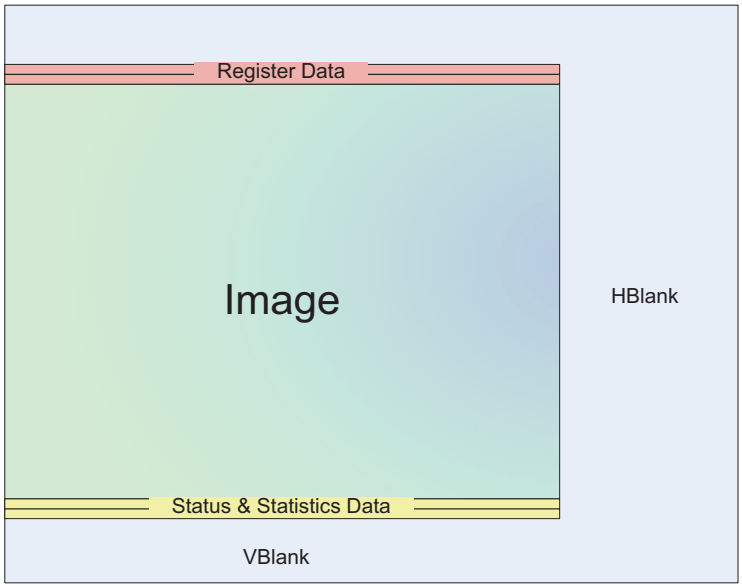
The exposure control measures current scene luminosity by accumulating a histogram of pixel values while reading out a frame. It then compares the current luminosity to the desired output luminosity. Finally, the appropriate adjustments are made to the exposure time and gain. All pixels are used, regardless of color or mono mode.

Embedded Data and Statistic

The AR0130 has the capability to output image data and statistics embedded within the frame timing. There are 2 types of information embedded within the frame readout:

1. Embedded Data: If enabled, these are displayed on the 2 rows immediately before the first active pixel row is displayed.
2. Embedded Statistics: If enabled, these are displayed on the 2 rows immediately after the last active pixel row is displayed.

Figure 15: 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 non-defined registers will have a value of 0.

The format of the embedded register data transmission is defined per the embedded data section of the SMIA Function Specification.

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 8msb and 8lsb. The alignment of the 8bit data will be on the 8MSB bits of the 12-bit pixel word. For example, if a register value of 0x1234 is to be transmitted, it will be transmitted over 2, 12-bit pixels as follows: 0x120, 0x340.

Embedded Statistics

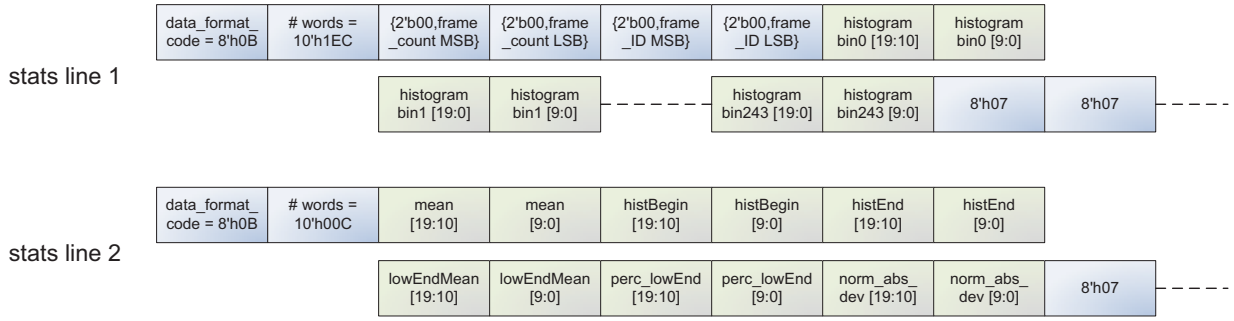
The embedded statistics contain frame identifiers and histogram information of the image in the frame. This can be used by downstream auto-exposure algorithm blocks to make decisions about exposure adjustment.

This histogram is divided into 244 bins with a bin spacing of 64 evenly spaced bins for digital code values 0 to 2^{12}. Only bins 0 through 63 contain histogram information. Bins 64 through 243 contain a zero value.

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.

The figure below summarizes how the embedded statistics transmission looks like. It should be noted that data, as shown in Figure 16, is aligned to the msb of each word:

Figure 16: 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 (TAG)
- Mean
- Histogram Begin
- Histogram End
- Low End Histogram Mean
- Percentage of Pixels Below Low End Mean
- Normal Absolute Deviation

Gain

Digital Gain

Digital gain can be controlled globally by R0x305E (Context A) or R0x30C4 (Context B). There are also registers that allow individual control over each Bayer color (GreenR, GreenB, Red, Blue).

The format for digital gain setting is *xxx.yyyyy* where 0b00100000 represents a 1x gain setting and 0b00110000 represents a 1.5x gain setting. The step size for yyyyy is 0.03125 while the step size for xxx is 1. Therefore to set a gain of 2.09375 one would set digital gain to 01000011.

Analog Gain

The AR0130 has a column parallel architecture and therefore has an Analog gain stage per column.

There are 2 stages of analog gain, the first stage can be set to 1x, 2x, 4x or 8x. This is can be set in R0x30B0[5:4](Context A) or R0x30B0[9:8] (Context B). The second stage is capable of setting an additional 1x or 1.25x gain which can be set in R0x3EE4[9:8].

This allows the maximum possible analog gain to be set to 10x.

Black Level Correction

Black level correction is handled automatically by the image sensor. No adjustments are provided except to enable or disable this feature. Setting R0x30EA[15] disables the automatic black level correction. Default setting is for automatic black level calibration to be enabled.

The automatic black level correction measures the average value of pixels from a set of optically black lines in the image sensor. The pixels are averaged as if they were light-sensitive and passed through the appropriate gain. This line average is then digitally low-pass filtered over many frames to remove temporal noise and random instabilities associated with this measurement. The new filtered average is then compared to a minimum acceptable level, low threshold, and a maximum acceptable level, high threshold. If the average is lower than the minimum acceptable level, the offset correction value is increased by a predetermined amount. If it is above the maximum level, the offset correction value is decreased by a predetermined amount. The high and low thresholds have been calculated to avoid oscillation of the black level from below to above the targeted black level.

Row-wise Noise Correction

Row (Line)-wise Noise Correction is handled automatically by the image sensor. No adjustments are provided except to enable or disable this feature. Clearing R0x3044[10] disables the row noise correction. Default setting is for row noise correction to be enabled.

Row-wise noise correction is performed by calculating an average from a set of optically black pixels at the start of each line and then applying each average to all the active pixels of the line.

Column Correction

The AR0130 uses column parallel readout architecture to achieve fast frame rate. Without any corrections, the consequence of this architecture is that different column signal paths have slightly different offsets that might show up on the final image as structured fixed pattern noise.

AR0130 has column correction circuitry that measures this offset and removes it from the image before output. This is done by sampling dark rows containing tied pixels and measuring an offset coefficient per column to be corrected later in the signal path.

Column correction can be enabled/disabled via R0x30D4[15]. Additionally, the number of rows used for this offset coefficient measurement is set in R0x30D4[3:0]. By default this register is set to 0x7, which means that 8 rows are used. This is the recommended value. Other control features regarding column correction can be viewed in the AR0130 Register reference. Any changes to column correction settings need to be done when the sensor streaming is disabled and the appropriate triggering sequence must be followed as described below.

Column Correction Triggering

Column correction requires a special procedure to trigger depending on which state the sensor is in.

Column Triggering on Startup

When streaming the sensor for the first time after powerup, a special sequence needs to be followed to make sure that the column correction coefficients are internally calculated properly.

1. Follow proper power up sequence for power supplies and clocks
2. Apply sequencer settings if needed
3. Apply frame timing and PLL settings as required by application
4. Set analog gain to 1x and low conversion gain
5. Enable column correction and settings
6. Enable streaming (R0x301A[2]=1)
7. Wait 8 frames to settle
8. Disable streaming (R0x301A[2]=0)

After this, the sensor has calculated the proper column correction coefficients and the sensor is ready for streaming. Any other settings (including gain, integration time and conversion gain etc.) can be done afterwards without affecting column correction.

Column Correction Retriggering due to Mode Change

Since column offsets is sensitive to changes in the analog signal path, such changes require column correction circuitry to be retriggered for the new path. Examples of such mode changes include: horizontal mirror, vertical mirror, changes to column correction settings.

When such changes take place, the following sequence needs to take place:

1. Disable streaming (R0x301A[2]=0)
2. Disable column correction (R0x30D4[15]=0)
3. Enable streaming (R0x301A[2]=1)
4. Wait one frame time, or more
5. Disable streaming (R0x301A[2]=0)
6. Enable column correction (R0D4[15]=0)

7. Enable Streaming (R0x301A[2]=1)

Note: The above steps are not needed if the sensor is being reset (soft or hard reset) upon the mode change.

Test Patterns

The AR0130 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 7. 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.

Note: Turn off black level calibration (BLC) when Test Pattern is enabled.

Table 7: Test Pattern Modes

Test_Pattern_Mode	Test Pattern Output
0	No test pattern (normal operation)
1	Solid color test pattern
2	100% color bar test pattern
3	Fade-to-gray color bar test pattern
256	Walking 1s test pattern (12-bit)

Color Field

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 AR0130. This interface is designed to be compatible with the electrical characteristics and transfer protocols of the two-wire serial interface specification.

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 SCLK LOW; the AR0130 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 AR0130 are 0x20 (write address) and 0x21 (read

address) in accordance with the specification. Alternate slave addresses of 0x30 (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.

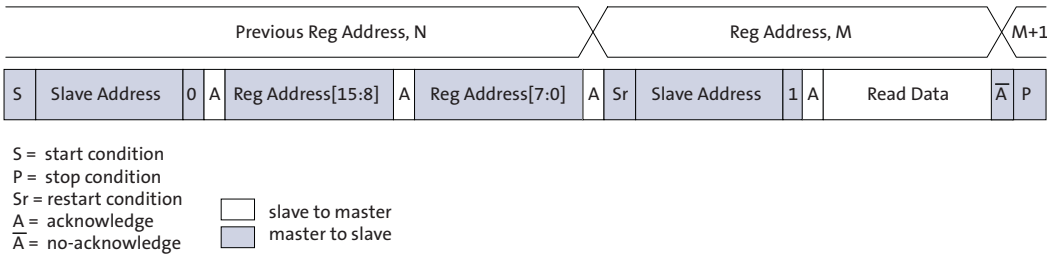
If the request was a WRITE, the master then transfers the 16-bit register address to which the WRITE should take place. This transfer takes place as two 8-bit sequences and the slave sends an acknowledge bit after each sequence to indicate that the byte has been received. The master then transfers the 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, eight 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 17) 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 17 shows how the internal register address maintained by the AR0130 is loaded and incremented as the sequence proceeds.

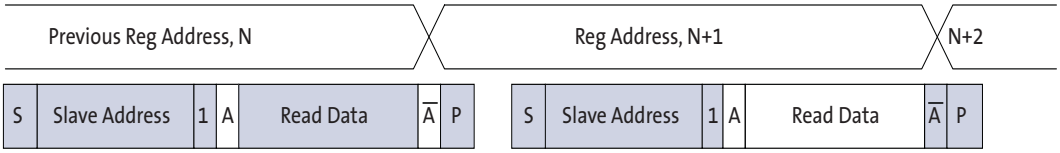
Figure 17: Single READ from Random Location



Single READ from Current Location

This sequence (Figure 18) performs a read using the current value of the AR0130 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.

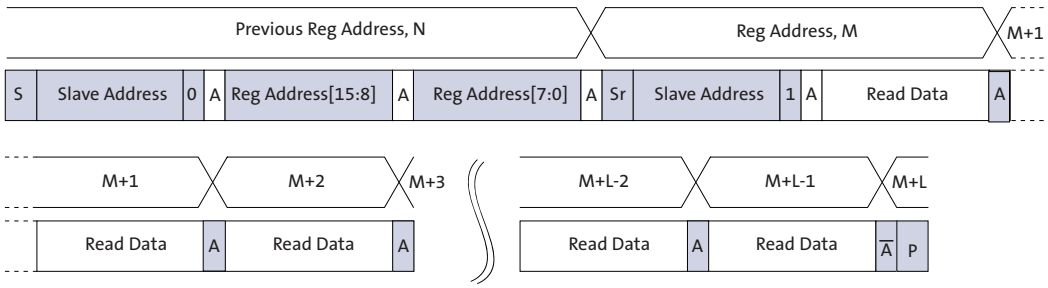
Figure 18: Single READ from Current Location



Sequential READ, Start from Random Location

This sequence (Figure 19) starts in the same way as the single READ from random location (Figure 17). 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.

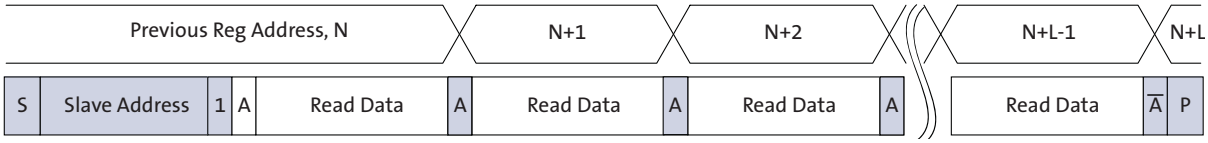
Figure 19: Sequential READ, Start from Random Location



Sequential READ, Start from Current Location

This sequence (Figure 20) starts in the same way as the single READ from current location (Figure 18). 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.

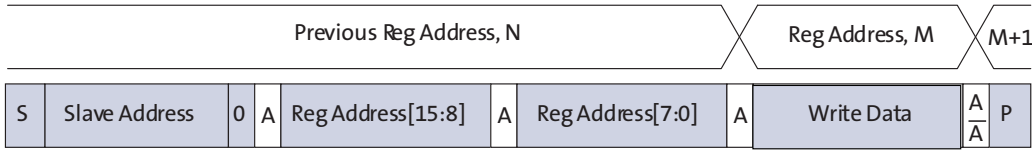
Figure 20: Sequential READ, Start from Current Location



Single WRITE to Random Location

This sequence (Figure 21) 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.

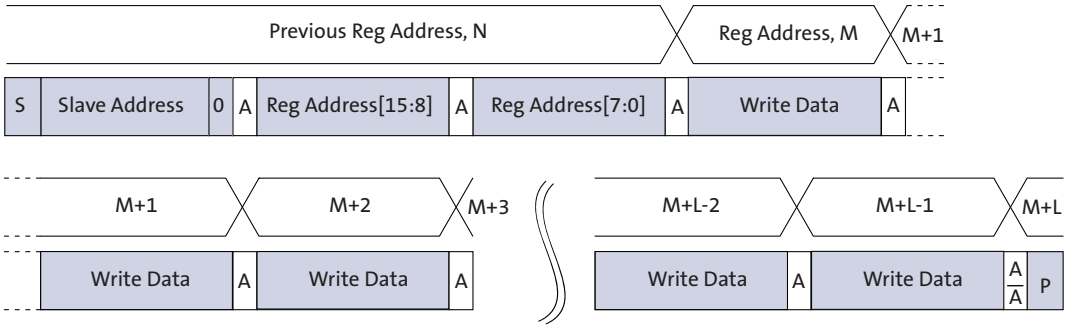
Figure 21: Single WRITE to Random Location



Sequential WRITE, Start at Random Location

This sequence (Figure 22) starts in the same way as the single WRITE to random location (Figure 21). 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 22: Sequential WRITE, Start at Random Location



Spectral Characteristics

Figure 23: Quantum Efficiency – Monochrome Sensor

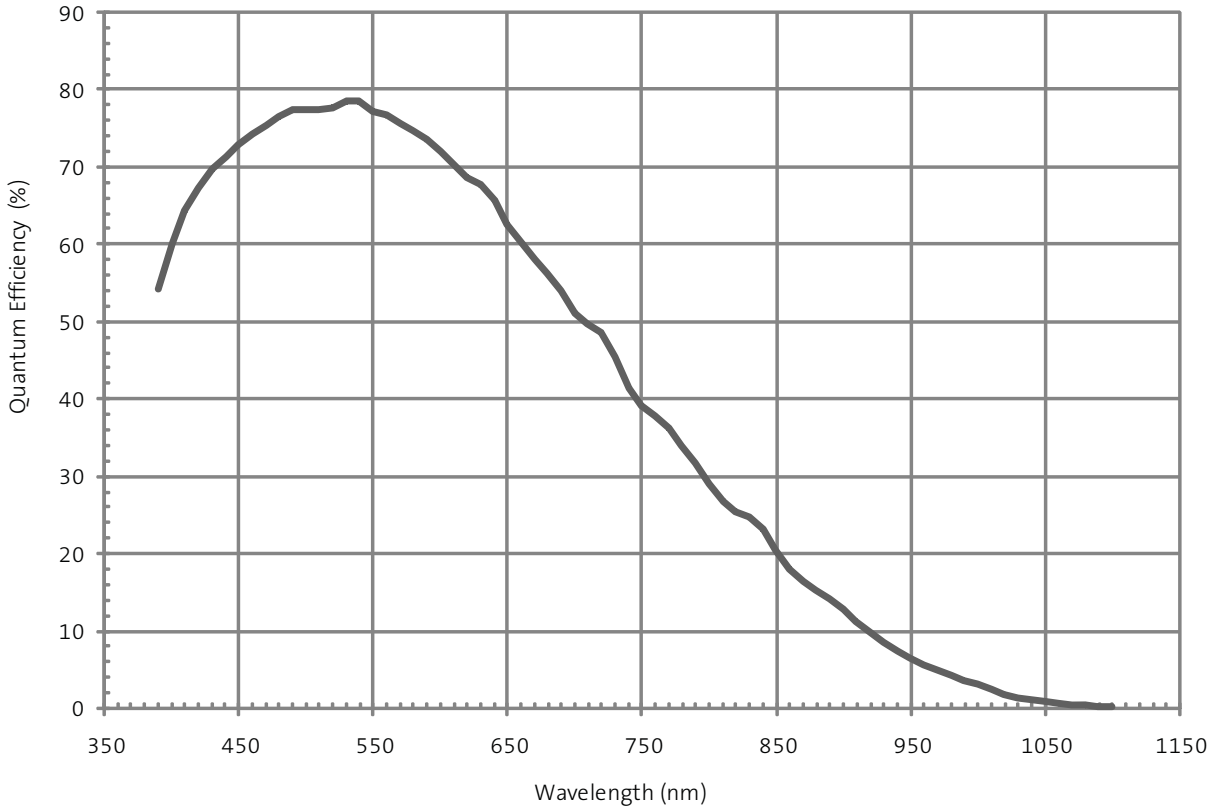
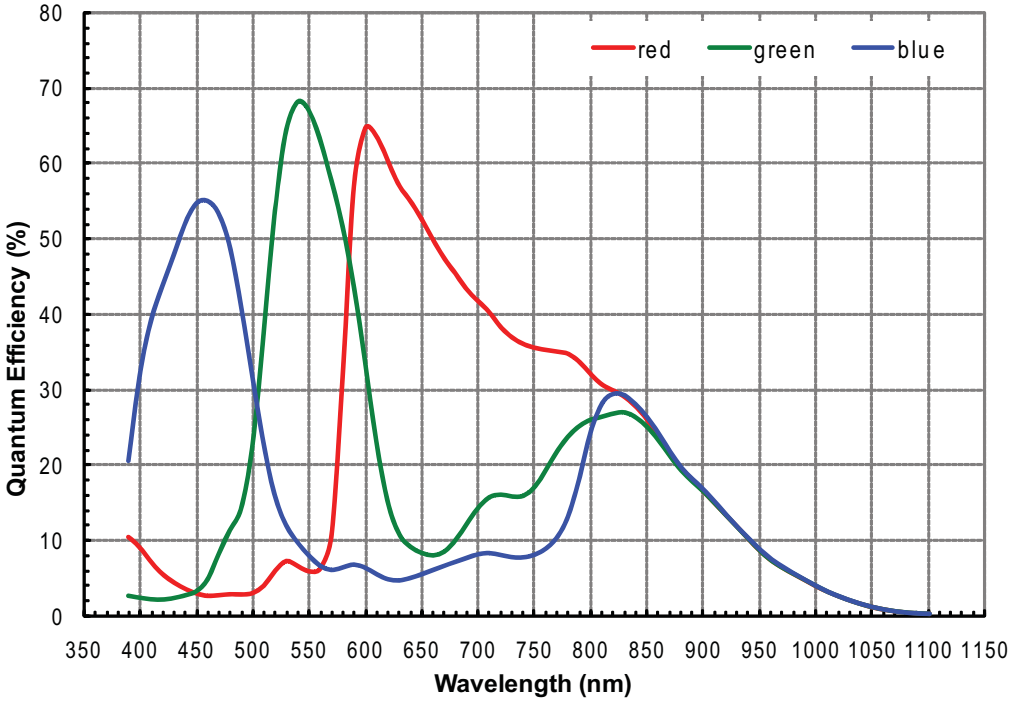
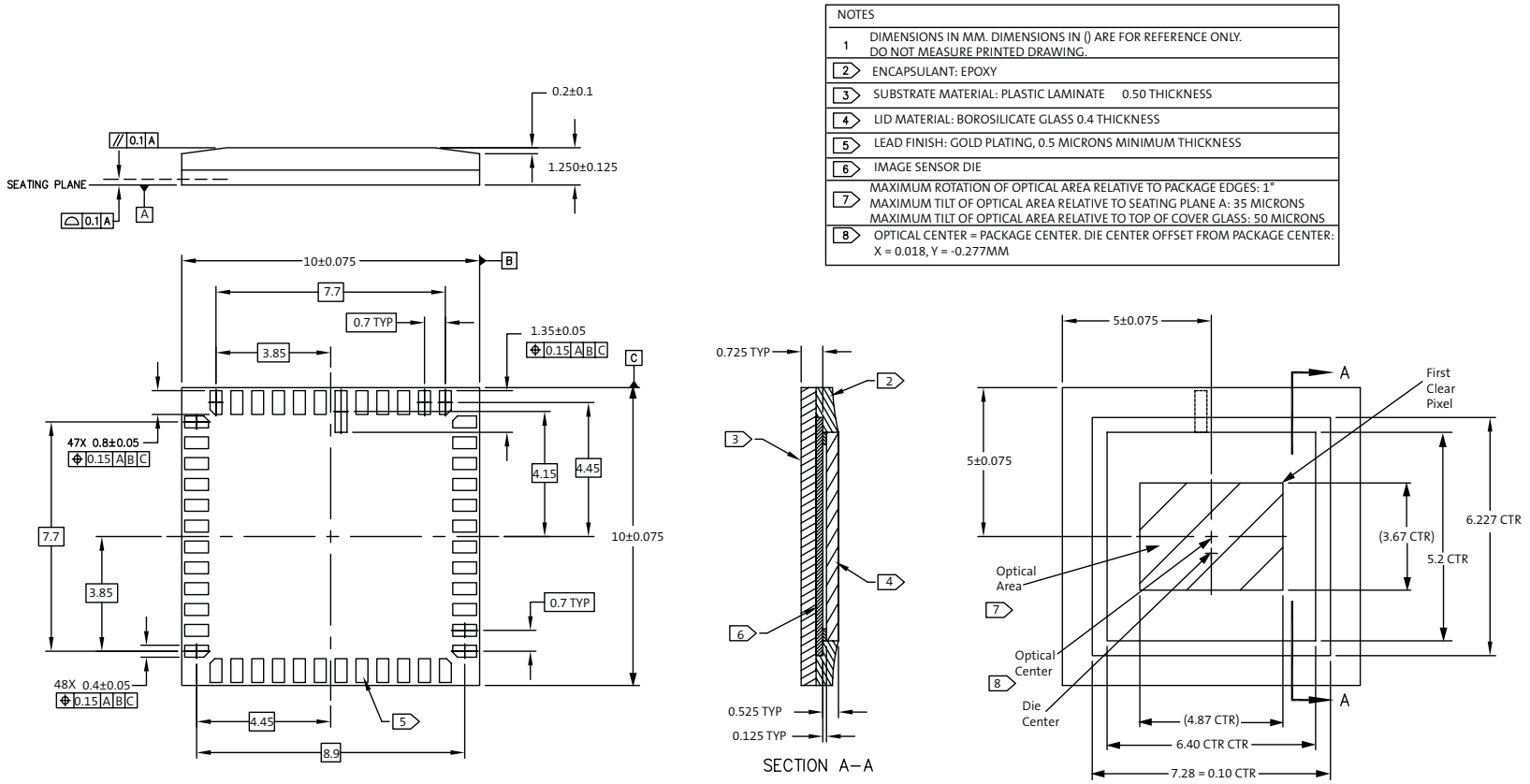


Figure 24: Quantum Efficiency – Color Sensor



Package Dimensions

Figure 25: 48 iLCC Package Outline Drawing



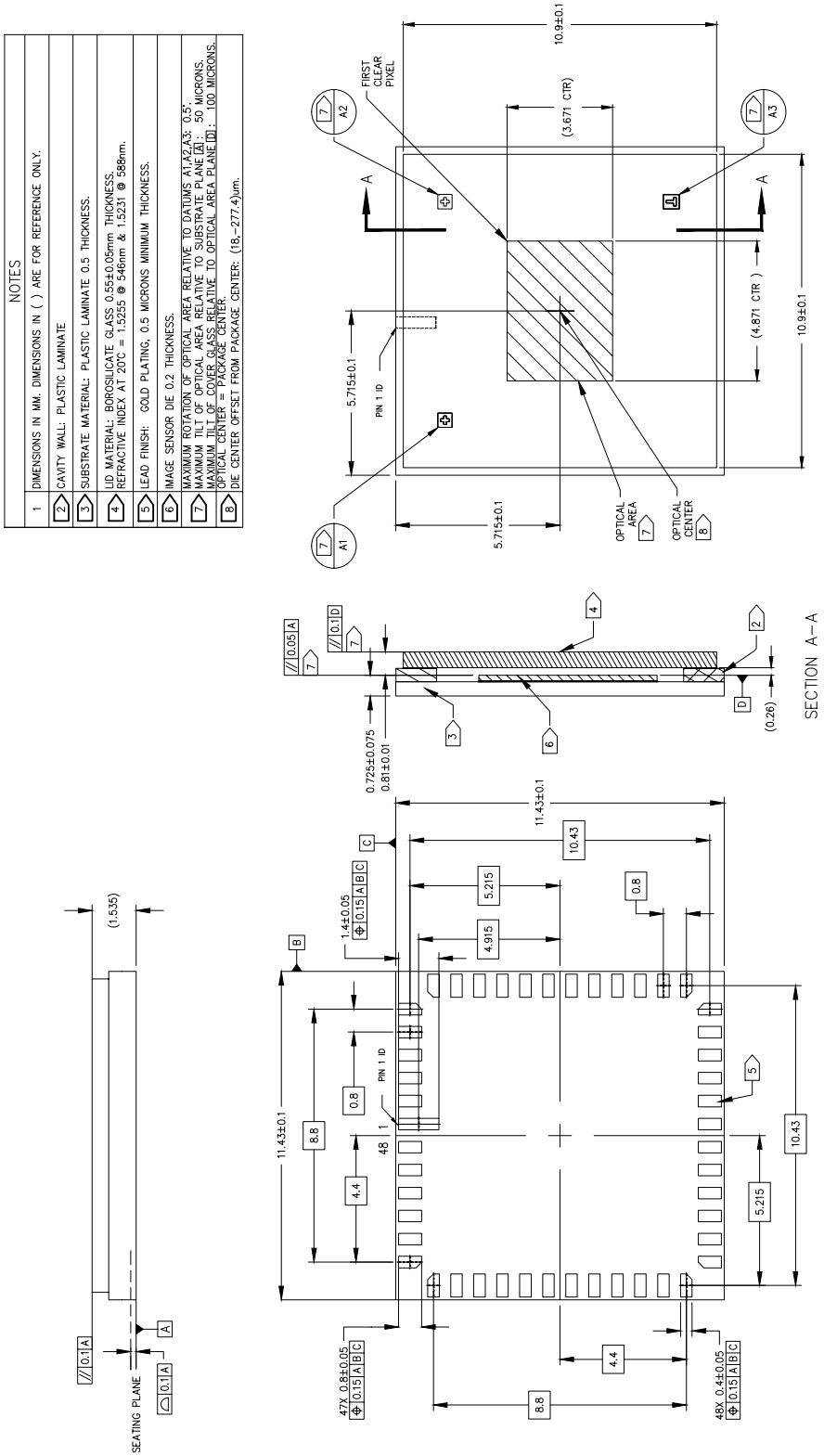


Figure 26: 48 PLCC Package Outline Drawing

Electrical Specifications

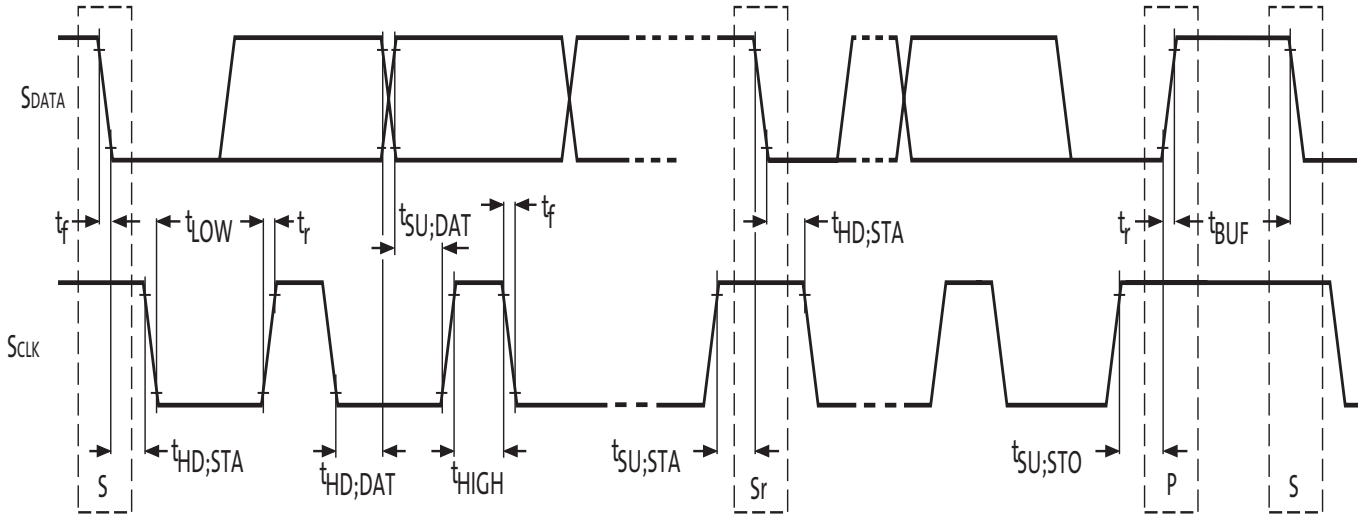
Unless otherwise stated, the following specifications apply to the following conditions:

V_{DD} = 1.8V – 0.10/+0.15; V_{DD_IO} = V_{DD_PLL} = V_{AA} = V_{AA_PIX} = 2.8V ± 0.3V; V_{DD_SLVS} = 0.4V – 0.1/+0.2; T_A = -30°C to +70°C; output load = 10pF; frequency = 74.25 MHz.

Two-Wire Serial Register Interface

The electrical characteristics of the two-wire serial register interface (SCLK, SDATA) are shown in Figure 27 and Table 8.

Figure 27: Two-Wire Serial Bus Timing Parameters



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

Table 8: Two-Wire Serial Bus Characteristics

[f]EXTCLK = 27 MHz; V_{DD} = 1.8V; V_{DD_IO} = 2.8V; V_{AA} = 2.8V; V_{AA_PIX} = 2.8V; V_{DD_PLL} = 2.8V; 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
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[4]	3.45[5]	0[6]	0.9[5]	µs
Data set-up time	[t]SU;DAT	250	-	100[6]	-	ns
Rise time of both SDATA and SCLK signals	[t]r	-	1000	20 + 0.1Cb[7]	300	ns
Fall time of both SDATA and SCLK signals	[t]f	-	300	20 + 0.1Cb[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	Cb	-	400	-	400	pF
Serial interface input pin capacitance	CIN_SI	-	3.3	-	3.3	pF
SDATA max load capacitance	CLOAD_SD	-	30	-	30	pF
SDATA pull-up resistor	RSD	1.5	4.7	1.5	4.7	KΩ

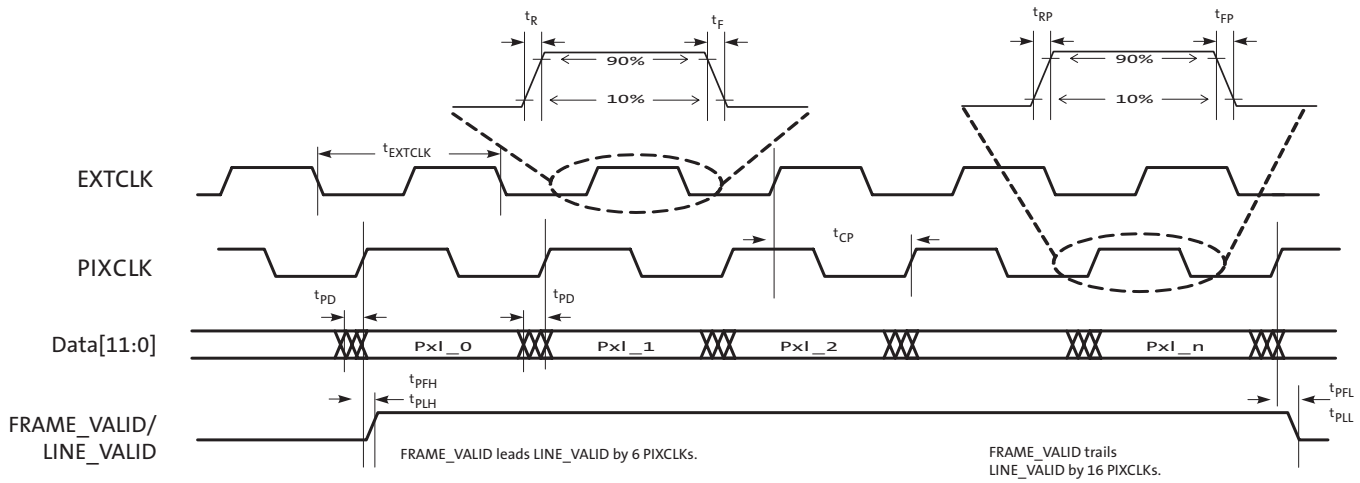
Notes:
1. This table is based on I^2C standard (v2.1 January 2000). Philips Semiconductor.
2. Two-wire control is I^2C-compatible.
3. All values referred to V_{IHmin} = 0.9 V_{DD} and V_{ILmax} = 0.1V_{DD} levels. Sensor EXTCLK = 27 MHz.
4. A device must internally provide a hold time of at least 300 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^2C-bus device can be used in a Standard-mode I^2C-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^2C-bus specification) before the SCLK line is released.
7. Cb = total capacitance of one bus line in pF.

I/O Timing

By default, the AR0130 launches pixel data, FV, and LV with the rising edge of PIXCLK. The expectation is that the user captures DOUT[11:0], FV, and LV using the falling edge of PIXCLK.

See Figure 28 and Table 9 on page 40 for I/O timing (AC) characteristics.

Figure 28: I/O Timing Diagram



*PLL disabled for t_{CP}

Table 9: I/O Timing Characteristics

Symbol	Definition	Condition	Min	Typ	Max	Unit
$f_{EXTCLK1}$	Input clock frequency	PLL enabled; slew setting = 3	6	–	50	MHz
$t_{EXTCLK1}$	Input clock period	PLL enabled; slew setting = 3	20	–	166	ns
$f_{EXTCLK2}$	Input clock frequency	PLL disabled; slew setting = 3	6	–	74.25	MHz
$t_{EXTCLK2}$	Input clock period	PLL disabled; slew setting = 3	13.4	–	166	ns
t_R	Input clock rise time		–	3	–	ns
t_F	Input clock fall time		–	3	–	ns
t_{RP}	Pixclk rise time		2	–	7	ns
t_{FP}	Pixclk fall time		3	–	8	ns
	Clock duty cycle		45	50	55	%
$t_{(PIX\ JITTER)}$	Jitter on PIXCLK		–	1		ns
$t_{JITTER2}$	Input clock jitter 74.25 MHz		–	600	–	ps
t_{CP}	EXTCLK to PIXCLK propagation delay	Nominal voltages, PLL Disabled; slew setting = 3	12		18	ns
f_{PIXCLK}	PIXCLK frequency	Default, Nominal Voltages; slew setting = 3	6		74.25	MHz

Table 9: I/O Timing Characteristics (continued)

Symbol	Definition	Condition	Min	Typ	Max	Unit
^tPD	PIXCLK to data valid	Default, Nominal Voltages; slew setting = 3	–2		4	ns
^tPFH	PIXCLK to FV HIGH	Default, Nominal Voltages; slew setting = 3	–2		4	ns
^tPLH	PIXCLK to LV HIGH	Default, Nominal Voltages; slew setting = 3	–3		4	ns
^tPFL	PIXCLK to FV LOW	Default, Nominal Voltages; slew setting = 3	–3		4	ns
^tPLL	PIXCLK to LV LOW	Default, Nominal Voltages; slew setting = 3	–3		4	ns
C_{LOAD}	Output load capacitance		–	<10	–	pF
C_{IN}	Input pin capacitance		–	2.5	–	pF

DC Electrical Characteristics

The DC electrical characteristics are shown in the tables below.

Table 10: DC Electrical Characteristics

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}	Digital supply voltage	Do not connect.	–	–	–	V
V_{IH}	Input HIGH voltage		V_{DD_IO}*0.7	–	–	V
V_{IL}	Input LOW voltage		–	–	V_{DD_IO}*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

Caution **Stresses greater than those listed in Table 11 may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.**

Table 11: Absolute Maximum Ratings

Symbol	Parameter	Minimum	Maximum	Unit
V_{SUPPLY}	Power supply voltage (V_{DD} and V_{AA} supplies)	–0.3	4.5	V
I_{SUPPLY}	Total power supply current	–	200	mA
I_{GND}	Total ground current	–	200	mA
V_{IN}	DC input voltage	–0.3	V_{DD_IO} + 0.3	V
V_{OUT}	DC output voltage	–0.3	V_{DD_IO} + 0.3	V
T_{STG}[1]	Storage temperature	–40	+85	°C

Notes: 1. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

2. To keep dark current and shot noise artifacts from impacting image quality, keep operating temperature at a minimum.

Table 12: Operating Current Consumption in Parallel Output

Definition	Condition	Symbol	Min	Typ	Max	Unit
Digital operating current	Streaming, 1280x960 45fps	$I_{DD}1$	–	40	65	mA
I/O digital operating current	Streaming, 1280x960 45fps	I_{DD_IO}	–	35	–	mA
Analog operating current	Streaming, 1280x960 45fps	I_{AA}	–	30	55	mA
Pixel supply current	Streaming, 1280x960 45fps	I_{AA_PIX}	–	10	15	mA
PLL supply current	Streaming, 1280x960 45fps	I_{DD_PLL}	–	7	–	mA
Digital operating current	Streaming, 720p 60fps	$I_{DD}1$	–	40	–	mA
I/O digital operating current	Streaming, 720p 60fps	I_{DD_IO}	-	35	–	mA
Analog operating current	Streaming, 720p 60fps	I_{AA}	–	30	–	mA
Pixel supply current	Streaming, 720p 60fps	I_{AA_PIX}	–	10	15	mA
PLL supply current	Streaming, 720p 60fps	I_{DD_PLL}	–	7	–	mA

Notes: 1. Operating currents are measured at the following conditions:
V_{AA}=V_{AA_PIX}=V_{DD_IO}=V_{DD_PLL}=2.8V
V_{DD}=1.8V
PLL Enabled and PIXCLK=74.25MHz
T_A = 25°C

Table 13: Standby Current Consumption

Definition	Condition	Symbol	Min	Typ	Max	Unit
Hard standby (clock off)	Analog, 2.8V	-	–	70	200	µA
	Digital, 1.8V	-	–	640	900	µA
Hard standby (clock on)	Analog, 2.8V	-	–	275	–	µA
	Digital, 1.8V	-	–	1.55	–	mA
Soft standby (clock off)	Analog, 2.8V	-	–	70	200	µA
	Digital, 1.8V	-	–	640	900	µA
Soft standby (clock on)	Analog, 2.8V	-	–	275	–	µA
	Digital, 1.8V	-	–	1.55	–	mA

Notes: 1. Analog – V_{AA} + V_{AA_PIX} + V_{DD_PLL}
2. Digital – V_{DD} + V_{DD_IO} + V_{DD_SLVS}

Power-On Reset and Standby Timing

Power-Up Sequence

The recommended power-up sequence for the AR0130 is shown in Figure 29. 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 0–10µs, turn on VAA and VAA_PIX power supply.
3. After 0–10µs, turn on VDD power supply.
4. After 0–10µs, turn on VDD_IO power supply.
5. After the last power supply is stable, enable EXTCLK.
6. Assert RESET_BAR for at least 1ms.
7. Wait 150000 EXTCLKs (for internal initialization into software standby.
8. Configure PLL, output, and image settings to desired values.
9. Wait 1ms for the PLL to lock.
10. Set streaming mode (R0x301a[2] = 1).

Figure 29: Power Up

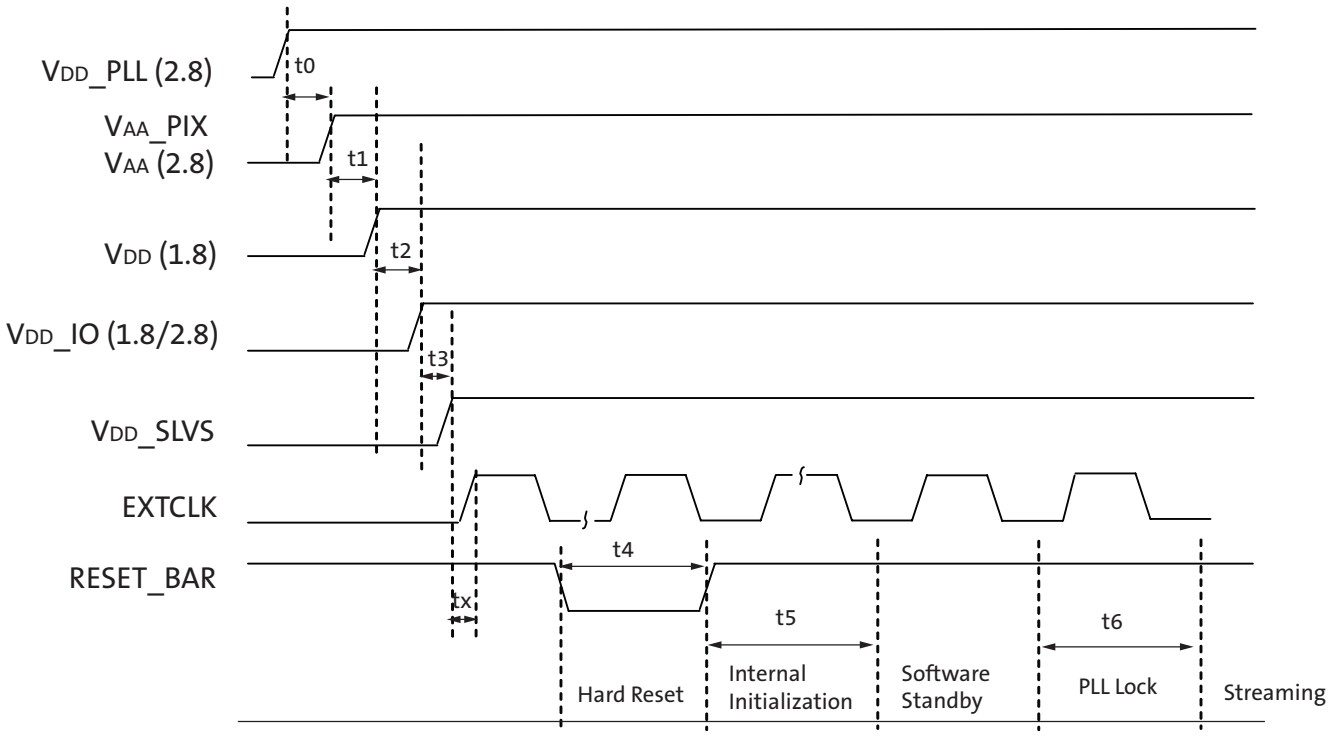


Table 14: Power-Up Sequence

Definition	Symbol	Minimum	Typical	Maximum	Unit
V_{DD}_PLL to V_{AA}/V_{AA}_PIX[3]	t0	0	10	–	µs
V_{AA}/V_{AA}_PIX to V_{DD}	t1	0	10	–	µs
V_{DD} to V_{DD}_IO	t2	0[4]	10	–	µs
V_{DD}_IO to V_{DD}_SLVS	t3	0	10	–	µs
Xtal settle time	tx	–	30[1]	–	ms
Hard Reset	t4	1[2]	–	–	ms
Internal Initialization	t5	150000	–	–	EXTCLKs
PLL Lock Time	t6	1	–	–	ms

Notes:
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 V_{DD}_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 V_{DD}_PLL is powered after other supplies then sensor may have functionality issues and will experience high current draw on this supply.
4. For the case where V_{DD}_IO is 2.8V and V_{DD} is 1.8V, it is recommended that the minimum time be 5µs.

Power-Down Sequence

The recommended power-down sequence for the AR0130 is shown in Figure 30. 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, if used.
4. Turn off VDD_IO.
5. Turn off VDD.
6. Turn off VAA/VAA_PIX.
7. Turn off VDD_PLL.

Figure 30: Power Down

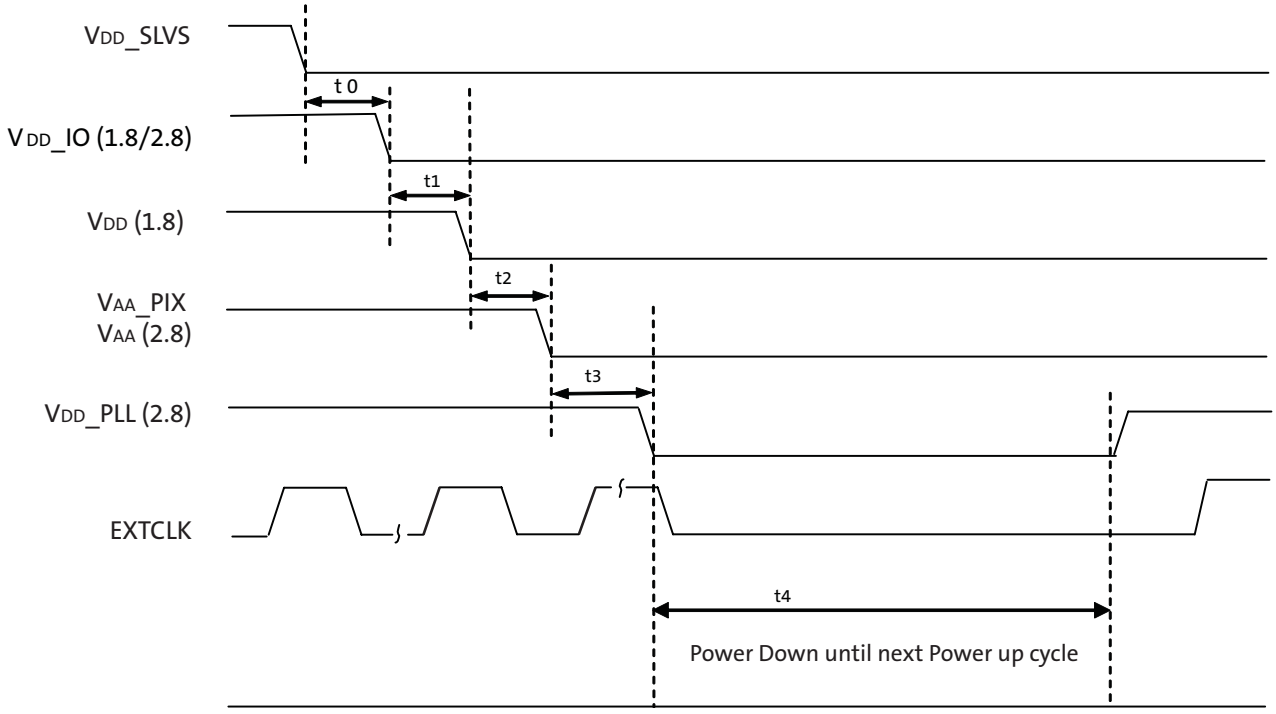


Table 15: Power-Down Sequence

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

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

Revision History

Rev. E 6/5/12

- Updated Table 2, “Available Part Numbers,” on page 1
- Updated “General Description” on page 6

Rev. D 3/19/12

- Updated to Production
- Removed section “Defective Pixel Correction
- Updated Table 9, “I/O Timing Characteristics,” on page 40
- Updated Table 12, “Operating Current Consumption in Parallel Output ,” on page 43
- Updated Table 13, “Standby Current Consumption ,” on page 43

Rev. C 2/16/12

- Updated “Features” on page 17
- Updated Table 1, “Key Parameters,” on page 1
- Updated Table 2, “Available Part Numbers,” on page 1
- Updated “General Description” on page 6
- Updated Figure 1: “Block Diagram,” on page 6
- Deleted Figure 2: “Typical Configuration: Serial Four-Lane HiSPi Interface” from p.6 and renumbered remaining figures.
- Updated notes for Figure 2: “Typical Configuration: Parallel Pixel Data Interface,” on page 7
- Updated Table 3, “Pad Descriptions,” on page 8
- Updated Figure 3: “48-Pin iLCC Pinout Diagram,” on page 9
- Added Figure 4: “48-Pin PLCC Pinout Diagram,” on page 10
- Deleted section “Serial Output Data Timing”
- Updated title of Figure 23: “Quantum Efficiency – Monochrome Sensor,” on page 34
- Updated Figure 24: “Quantum Efficiency – Color Sensor,” on page 35
- Added Figure 26: “48 PLCC Package Outline Drawing,” on page 37
- Updated “Electrical Specifications” on page 38
- Updated Table 8, “Two-Wire Serial Bus Characteristics,” on page 39
- Updated Table 10, “DC Electrical Characteristics,” on page 41
- Updated Table 11, “Absolute Maximum Ratings,” on page 41
- Deleted Table 13: Operating Currents in HiSPi Output and renumbered succeeding tables
- Deleted section “HiSPi Electrical Specifications”
- Updated Figure 29: “Power Up,” on page 44
- Updated “Power-Down Sequence” on page 46
- Updated Figure 30: “Power Down,” on page 46

Rev. B 9/12/11

- Updated “Features” on page 17
- Updated Table 1, “Key Parameters,” on page 1
- Updated Table 2, “Available Part Numbers,” on page 1
- Added note to Figure 2: “Typical Configuration: Parallel Pixel Data Interface,” on page 7
- Updated note 4 in Figure 2: “Typical Configuration: Parallel Pixel Data Interface,” on page 7

- Added Figure 3: “48-Pin iLCC Pinout Diagram,” on page 9
- Updated “Parallel Output Data Timing” on page 14
- Updated Figure 12: “Timing Diagram,” on page 15
- Updated Table 4, “Frame Time (Example Based on 1280 x 960, 45 Frames Per Second),” on page 15
- Updated Table 6, “Real-Time Context-Switchable Registers,” on page 17
- Updated “To configure and use the PLL:” on page 18
- Updated “Entering Soft Standby:” on page 19
- Updated “Single Frame” on page 23
- Deleted “Temperature Sensor”
- Updated “Embedded Statistics” on page 26
- Updated “Column Correction Retriggering due to Mode Change” on page 28
- Added Figure 25: “48 iLCC Package Outline Drawing,” on page 36
- Updated Figure 27: “Two-Wire Serial Bus Timing Parameters,” on page 38
- Updated Table 8, “Two-Wire Serial Bus Characteristics,” on page 39
- Updated note 1 in Table 13, “Standby Current Consumption ,” on page 43
- Updated “Power-Up Sequence” on page 44
- Updated Figure 29: “Power Up,” on page 44
- Updated Table 14, “Power-Up Sequence,” on page 45
- Updated “Power-Down Sequence” on page 46
- Updated Figure 30: “Power Down,” on page 46
- Updated Table 15, “Power-Down Sequence,” on page 46

Rev. A 3/1/11

- Initial release

10 Eunos Road 8 13-40, Singapore Post Center, Singapore 408600 prodmktg@aptina.com www.aptina.com