CMOSIS / AWAIBA

is now

Member of the ams Group

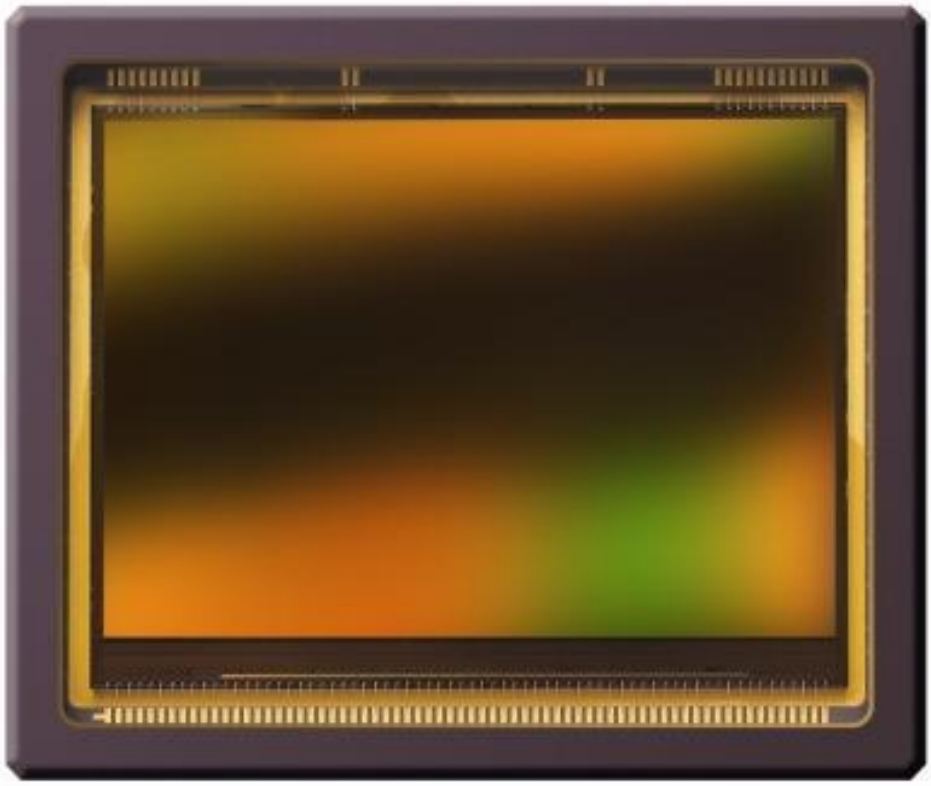
The technical content of this CMOSIS / AWAIBA document is still valid.

Contact information:

Headquarters:
ams AG
Tobelbaderstrasse 30
8141 Premstaetten, Austria
Tel: +43 (0) 3136 500 0
e-Mail: ams_sales@ams.com

Please visit our website at www.ams.com

CHR70M Datasheet



CMOSIS High Resolution 70 Megapixels CMOS Image Sensor

Change record

Issue	Date	Modification
1	1/10/2010	Original document
2	30/8/2011	- Added ordering information - Added glass lid transfer curve and info - Added color filter placement and info - Added latest pixel specs
3	18/11/2011	First draft for standard product
4	02/05/2012	Full datasheet revision
5	23/08/2013	Added: - SPI read-out delay Updated: - Ordering information: p/n changed - VDD_ana current to 80mA - Total power consumption to 435mW - Assembly drawing pin orientation - Frame and line timing diagrams Removed: - Confidential annotation
6	05/06/2014	Updated: - Pin Descriptions - QE and spectral response of mono/color devices with microlenses - VDD_AB can be tied to ground Added: - Angular response
7	12/12/2014	Added: - Micro lens shift for a 22.3° CRA - Output buffers current (reg96) details (in register overview) - ES (Engineering Sample) to part number. - Part number for sensor with removable glass - SPI I/O's not tri-stated
8	19/06/2015	Updated: - Power supplies and decoupling recommendation - Recommended value reg 39: 4 → 14 - Recommended value reg 41: 58 → 46 - Recommended value reg 43: 0 → 1 - Recommended value reg 44: 0 → 1 - Data output timing in external timing mode, ch 4.2 Added: - Excessive light precaution - Chief Ray Angle (CRA);10 below ch10 - reg97: Delay of output sample clock details
9	14/01/2016	Updated: - Register overview; changes from v8 datasheet - ROT diagram made clearer; no timing changes. Added: - High Grade variant Removed: - No-micro-lenses variant

Disclaimer

Because the CHR70M started its design life as a custom imager for biometric applications, the sale of the imager for these applications is excluded.

CMOSIS reserves the right to change the product, specification and other information contained in this document without notice. Although CMOSIS does its best efforts to provide correct information, this is not warranted.

Table of Contents

1 INTRODUCTION

1.1 OVERVIEW

This document is the datasheet for the CHR70M sensor. The CHR70M is a high resolution CMOS image sensor with 10000 by 7096 pixels. The image array consists of 3.1µm x 3.1µm pinned diode pixels which share a number of transistors (2 pixels sharing). The image sensor has 8 analog outputs, each running at 30MHz. The image sensor also integrates a programmable gain amplifier and offset regulation. This results in a frame rate of 3fps at full resolution. Higher frame rates can be achieved in windowing mode or subsampling mode. These and other settings are all programmable using the SPI interface. All internal exposure and read out timings are generated by a programmable on-board sequencer. External triggering and exposure programming is also possible.

1.2 FEATURES

- 10000 * 7096 active pixels on a 3.1um pitch
- frame rate 3 Frames/sec
- Windowing capability
- Moving window functionality
- Master clock 30MHz
- 8 analog outputs @30MHz
- On chip timing generation
- SPI-control
- Ceramic PGA package (65 pins)
- 3.3V signaling

1.3 SPECIFICATIONS

- Full well charge: >13Ke$^-$
- Sensitivity: 0.15 A/W (@ 555nm)
- Dark noise: 7e$^-$ (measured in high gain mode)
- Conversion factor: ~63 µV/e$^-$
- Dynamic range: 63 dB
- Dark current: 3.2 e$^-$/s @ room temperature
- Fixed pattern noise: 0.09 (% of full swing)
- Power consumption: 435mW

1.4 CONNECTION DIAGRAM

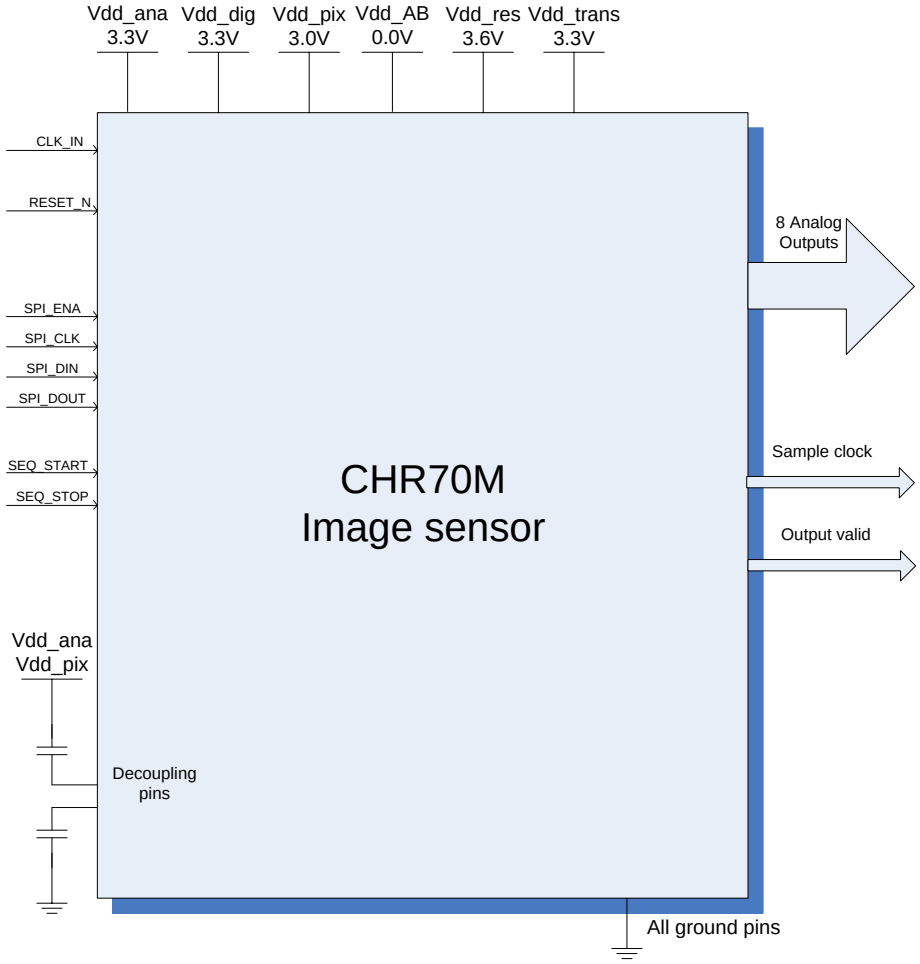


Figure 1: Connection diagram for the CMV4000 image sensor

Please see the pin list for a detailed description of all pins and their proper connections. Some optional pins are not displayed on the figure above. The exact pin numbers can be found in the pin list and on the package drawing.

2 SENSOR ARCHITECTURE

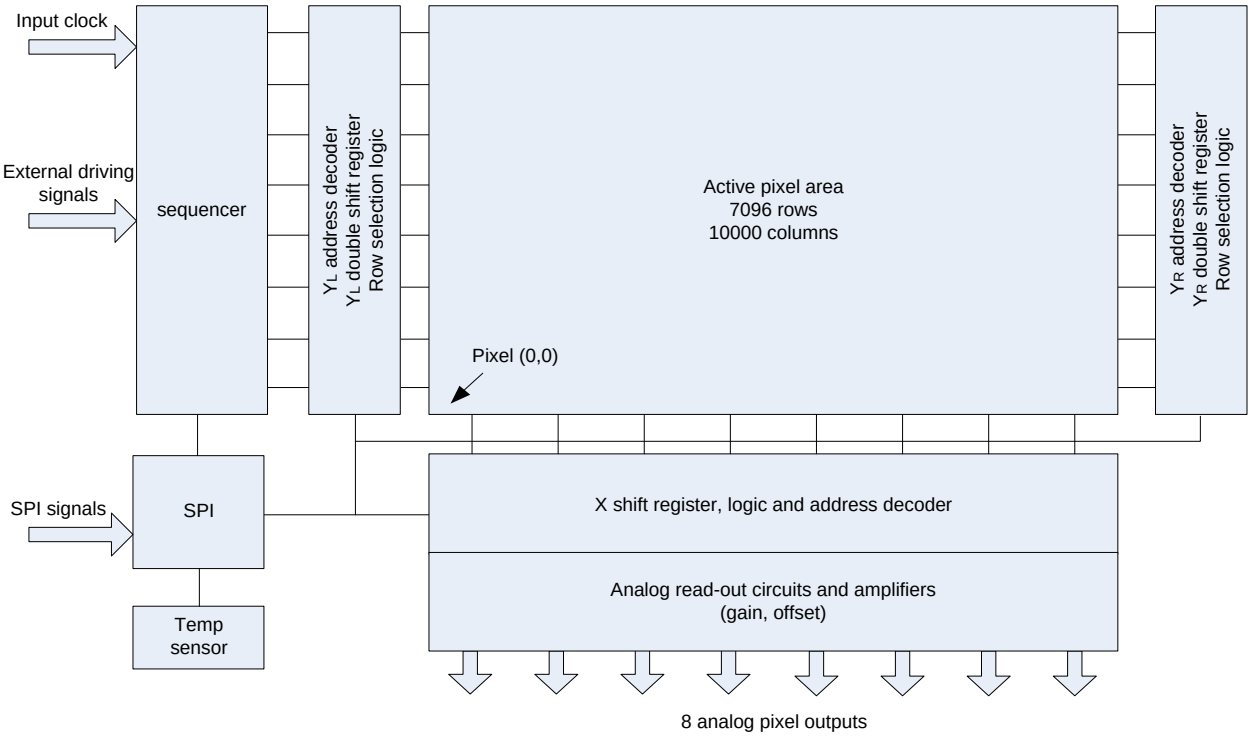


Figure 2: Sensor block diagram

Figure 1 shows the general sensor architecture. It basically consists of the pixel array, addressable shift registers in X and Y direction (for pixel read out) and column amplifiers of which the signals are multiplexed to the sensor analog outputs. The pixel array has 7096 rows and 10000 columns (dummy lines and columns not included) and is read out in landscape mode. The readout sequence begins by starting an integration period at the first row of the window. The address decoder is used to load pointers into the integration time shift register at the start address (writeable over SPI). The shift registers shifts its value into the next row at each clk_Y pulse (internally or externally generated). After the required integration time, the readout of the window is started. This is done by loading a pointer into the read shift register. This pointer is also shifted to the next row at each clk_Y pulse (see also 3.5.2.1). The row selection logic is equipped at both sides to speed up the row access time.

When a particular row is selected, its pixel values appear on the pixel column bus at the input of the column amplifiers. These pixel values are sampled into the sample and hold capacitors located in the column amplifier block. The column amplifier first samples the reset level of the pixel, and then, after a transfer line pulse, the photo-induced signal of the pixels is sampled.

The X shift register is then activated. Both signals are sequentially sampled over 16 multiplexed bus lines to 8 parallel output channels. Subsampling in Y can be achieved by programming the corresponding register, while 1-out-of-8 subsampling in the X direction is achieved by just sampling data from a single output channel (ignoring data from other outputs).

An SPI interface is provided to program different on-chip registers. Amongst these registers are the start and stop addresses of the window, bias settings for amplifiers, gain and offset registers of the output channels, standby of output channels, etc.

It is provided that the sensor can be fully operated by external control signals. This is a guarantee for the largest flexibility in mode of operation and possibility for full timing optimization of the different blocks. However, we have

also put a sequencer on-chip that will generate all required control signals to operate the sensor from only a few external control clocks. The timing of the signals from the sequencer is based on best operation mode simulations but still quite allow some programmability. It supports full frame readout with programmable integration time, number of frames, as well as subsampled and windowed operation with possibility to move the window from frame to frame.

The default start-up condition is with external control signals. The sequencer can be activated through the SPI interface and settings of an on-chip register. At that moment, most of the external control signals are ignored and are generated by the sequencer instead.

2.1 PIXEL ARRAY

The image array consists of 3.1µm x 3.1µm pinned diode pixels which share a number of transistors (2 pixels sharing) and a rolling shutter. The default product has micro lenses on top of the pixels. These lenses are shifted linearly towards the edges for a CRA of 22.3°.

2.2 SEQUENCER

It is provided that the sensor can be fully operated by external control signals (default start up condition). This is a guarantee for the largest flexibility in mode of operation and possibility for full timing optimization of the different blocks. However, we have also put a sequencer on-chip that will generate all required control signals to operate the sensor from only a few external control clocks. This sequencer can be activated through the SPI interface.

2.3 SPI INTERFACE

The SPI interface is used to load the sequencer registers with data. The data in these registers is used by the sequencer while driving and reading out the image sensor. Features like windowing, subsampling, gain and offset are programmed using this interface. The data in the on-chip registers can also be read back for test and debug of the surrounding system. Chapter 3.7 contains more details on register programming and SPI timing.

2.4 TEMPERATURE SENSOR

The on-chip temperature sensor can be read out by toggling the reg_latch_temp_data bit (register 125) from 1 to 0. When toggling this register like this, the temperature sensor data will be latched in registers 126-127. The temperature can then be readout by retrieving the data from the reg_temperature registers 126 and 127.

A calibration of the temperature sensor is needed for absolute temperature measurements as the offset between devices will vary. The temperature sensor requires a running input clock (CLK_IN), the other functions of the image sensor can be operational or in standby mode. The output value of the sensor is dependent on the input clock. A typical temperature sensor output vs. temperature curve at 30MHz can be found below. This results in response of about 77 DN/°C.

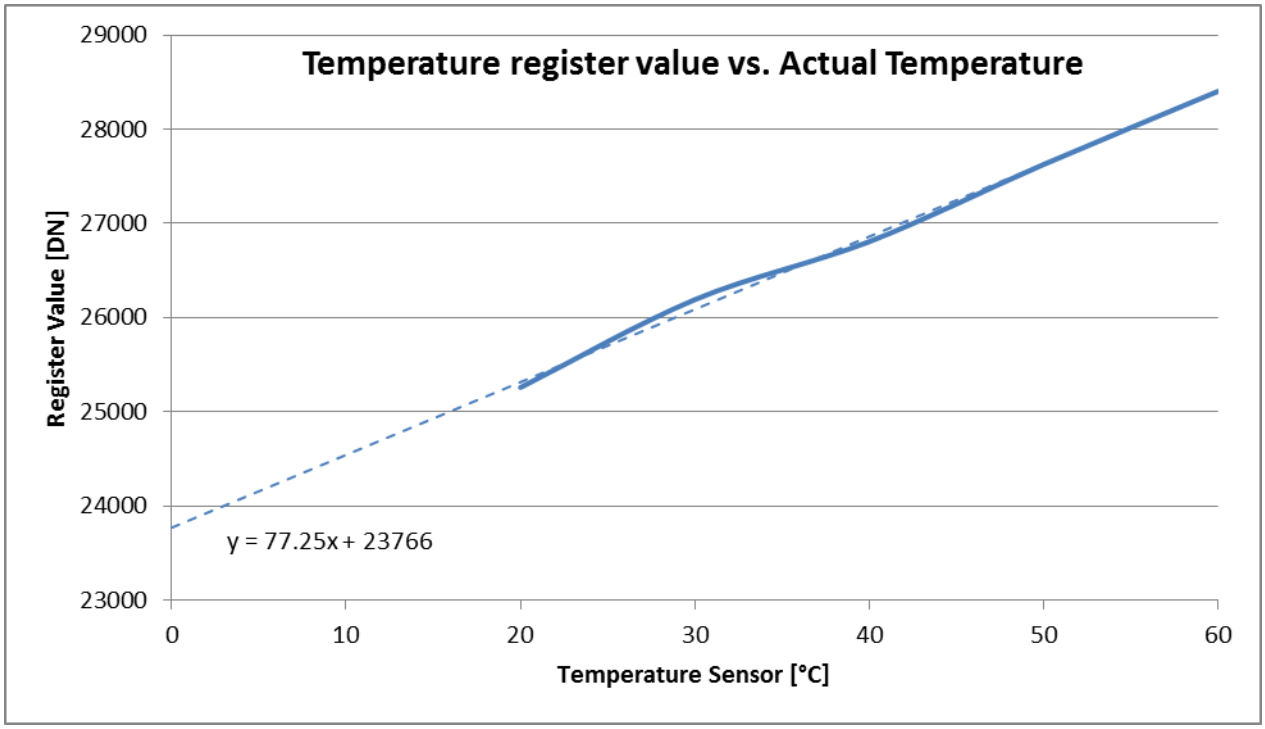


Figure 3: Typical output of the temperature sensor of the CHR70M

3 DRIVING THE CHR70M

The following paragraphs describe how the CHR70M sensor can be controlled. Controlling the sensor is done in two ways:

- By applying correct timing signals to the digital input pins
- By setting the correct programmability options.

Programmability is supported by a number of on-chip registers that can be loaded with user data. The contents of these registers define the way the different blocks on the chip operate.

A digital logic block on the sensor (called 'sequencer') can be used to control the chip timing. It generates the timing pulses for the different blocks on the chip to work properly, based on settings loaded in a number of registers. It is possible to bypass the sequencer and use external pulses (digital input pins). The selection between internal or external timing can be made by register upload (see further in this chapter). The default case is external timing.

3.1 POWER SUPPLIES AND SETTINGS

The CHR70M image sensor needs six separate power supplies.

Supply name	Usage	Typical value	Current typ.	Current peak
Vdd_ana	Analog read-out circuit, output amplifier	3.3V	100mA	140mA
Vdd_dig	SPI, sequencer, row/column address logic	3.3V	20mA	270mA
Vdd_pix	Pixel array supply	3.0V	35mA	170mA
Vdd_AB	Anti-blooming	0.0V (GND)	-	-
Vdd_res	Pixel reset signal	3.6V	10mA	70mA
Vdd_trans	Pixel transfer supply	3.3V	10mA	60mA

See pin list for exact pin numbers for every supply.

The typical current is what a typical sensor draws during idle and read out. The peak currents drawn are mostly during ROT and should be countered with enough decoupling capacitors. We recommend one 100µF bulk capacitor at the regulator side and one 100nF local capacitor per supply pin (not plane!), close to the sensor pins.

3.2 BIASING

For optimal performance, some pins need to be decoupled to ground or a Vdd. Please refer to the pin list for a detailed description for every pin and the appropriate decoupling if applicable.

3.3 DIGITAL INPUT PINS

The table below gives an overview of the external I/O and power pins used to drive the sensor.

Logic pins		
Use	***Name***	***Description***
PWR	VDDD	Digital power
GND	VSSD	Digital ground
I	CLK_IN	Sensor master clock input
I	RESET_N	Global asynchronous active-low reset
I	SEQ_START	Start sequencer operation
I	SEQ_STOP	Stop sequencer operation
SIGNALS FOR SEQUENCER BYPASS		
I	SYNC_X	Sets x shift register to start address
I	CLK_Y	Clock for y shift register

Logic pins		
Use	***Name***	***Description***
I	SYNC_Y_READ	Sets y read shift register to start address
I	SYNC_Y_INTE	Shifts input of y inte shift register at start address
I	PIX_TRANSFER	Transfer signal for pixels
I	PIX_RESET	Reset signal for pixels
I	PIX_SELECT	Select signal for pixels
I	COL_SAMPLE	Sample signal for columns
I	COL_SAMPLE_R	Sample reset value of pixels in columns
I	COL_SAMPLE_S	Sample signal value of pixels in columns
I	COL_INIT	Initiation signal for column amplifier
I	COL_PRECHARGE	Precharge signal for columns
I	COL_ENABLE	Enable signal for column amplifier
ON-CHIP GENERATED SIGNALS		
O	OUT_LVAL	Indicates when outputs are valid
O	OUT_CLK_SMP	Sample clock (indicates when outputs are best sampled)
SPI INTERFACE SIGNALS		
I	SPI_CLK	SPI Clock input
I	SPI_ENA	SPI enable signal (data transfer only valid when high)
I	SPI_DIN	SPI data input (data from master to slave)
O	SPI_DOUT	SPI data output (data from slave to master)

The OUT_LVAL and OUT_CLK_SMP have a driving strength to drive 30 MHz signals with a capacitive load of 20pF.

3.4 ELECTRICAL I/O SPECIFICATIONS

3.4.1 DIGITAL I/O DC SPECIFICATIONS (SEE PIN LIST FOR SPECIFIC PINS)

Parameter	***Description***	***Conditions***	***min***	***max***	***Units***
V_{IH}	High level input voltage		2.0	Vdd_dig	V
V_{IL}	Low level input voltage		GND	0.8	V
V_{OH}	High level output voltage	Vdd_dig=3.3V I_{OH}=-2mA	2.4		V
V_{OL}	Low level output voltage	Vdd_dig=3.3V I_{OL}=2mA		0.4	V

3.4.2 ANALOG OUTPUT (OUT_X) SPECIFICATIONS

Parameter	***Description***	***Conditions***	***min***	***typ***	***max***	***Unit***
V_{OH}	Output voltage high *	With gain		3.0	3.2	V
V_{OL}	Output voltage low *	Depending on the offset	0.45		1.85	V
f					CLK_IN	MHz
C_{LOAD}	Capacitive load drive				20	pF

* See more details in chapter 4.3

3.5 SENSOR TIMING

3.5.1 INPUT CLOCK AND RESET

The maximum frequency of the master clock input (CLK_IN) is 30MHz. There are no specific duty-cycle requirements. The chip has 8 analog outputs. The frame rate is dependent of the number of lines, columns, clock frequency and the Row Overhead Time (ROT).

$$line\ time = T_{clk} * \left(ROT + \left(\frac{\#columns}{8} \right) \right) \qquad frame\ time = line\ time * \#lines$$

$$frame\ rate = \frac{1}{frame\ time} = \frac{f_{clock}}{\#lines(ROT + \frac{\#columns}{8})}\ [fps]$$

Where ROT is expressed in number of master clock periods (see chapter 3.5.4 for more details). So for a full image with a 30MHz clock and a ROT of 168 clock cycles (default), you can achieve a 2.98fps frame rate.

The global active-low reset (RESET_N) resets all digital sequential cells (flip-flops) in the sensor. A falling edge on the RESET_N input (enter reset state) is fed to all sequential cells asynchronously. A rising edge on the RESET_N input (exit reset state) is synchronized internally to the rising edge of CLK_IN (the 3th one after the rising edge of the reset signal) before it goes to the flip-flops. This ensures that all flip-flops exit the reset state during the same clock period. The reset synchronization circuit also ensures that the minimum width of an active pulse on RESET_N is 2 clock cycles on the internal reset_n. This means that every glitch on the RESET_N input will generate a complete reset of the device. Figure 4 shows the timing of the reset synchronization circuit.

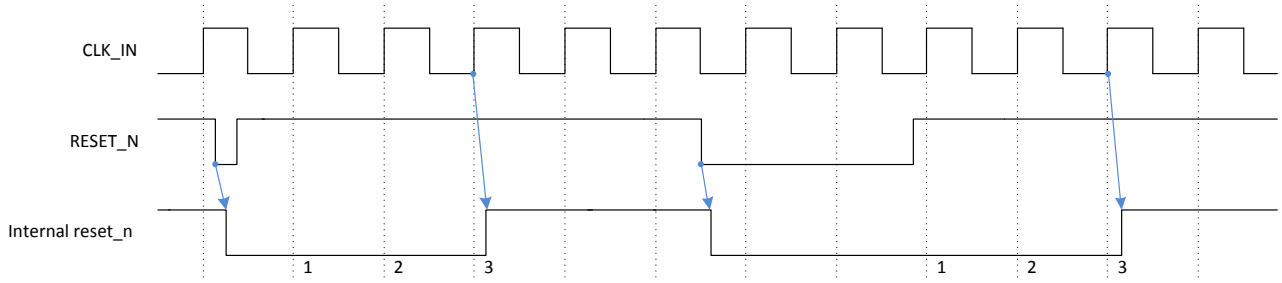


Figure 4: LVDS clock delay versus master clock

3.5.2 Frame and Line Timing

A frame is a collection of lines, which in its turn is a collection of pixels. One line-readout consists of a sampling period and a readout period. During the sampling period, all blocks on the sensor are prepared for readout of the line: the column readout structures are initialized and the pixel values (reset and signal) are copied into the columns. This operation is generally called Row Overhead Time or ROT. When the ROT has finished, the pixel values that are stored in the columns are read out sequentially. This sequence is repeated for every line in a frame.

Before the readout of a frame can start, an integration period must have passed. During this period, light integrates in the pixels. The integration time is specified as a number of line times. Figure 5 shows the timing of reading out 1 frame of 4 lines with an integration time of 2 line times.

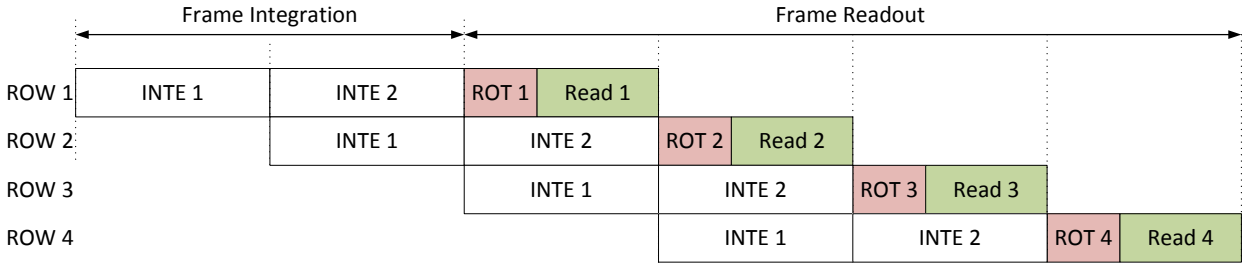


Figure 5: Schematic representation of integration and readout

Because of the rolling shutter operation of the sensor, the integration of frame n+1 can happen during the readout of frame n. Figure 6 shows the timing of reading out 2 frames of 4 lines with an integration time of 2 line times.

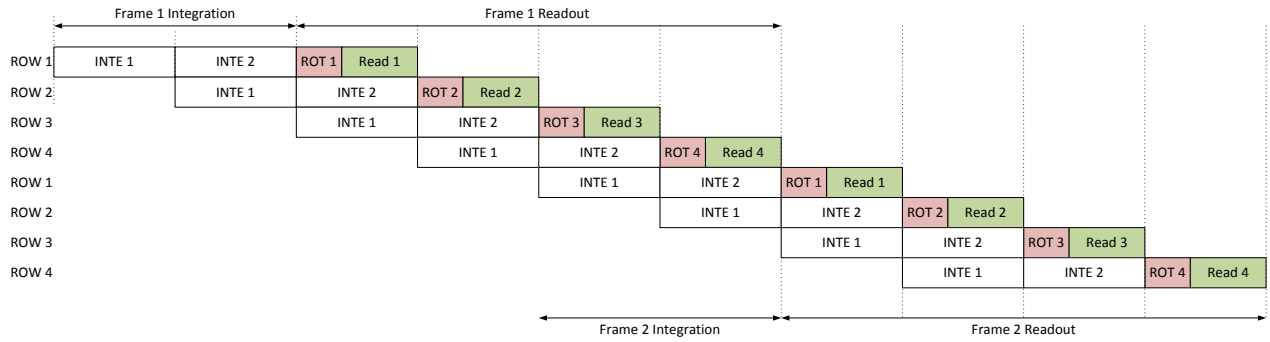


Figure 6: Example of readout of 2 frames (4 lines) with 2 lines integration time

If the integration time (in number of lines) is longer than the number of lines in a frame, the start of a frame is delayed until the integration time has finished. Figure 7 shows the timing of reading out 2 frames of 2 lines, with an integration time of 4 line times.

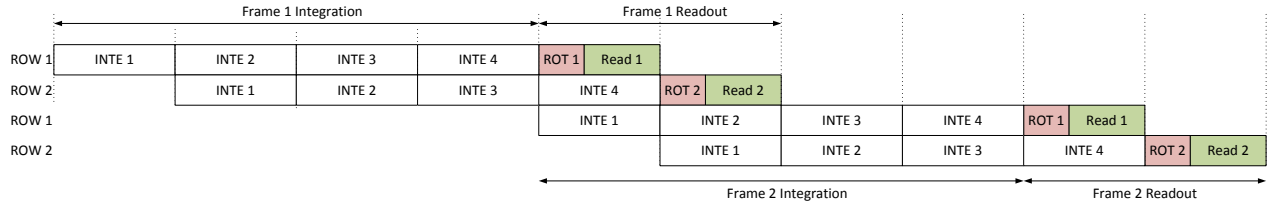


Figure 7: Example of readout of 2 frames (2 lines) with 4 lines integration time

3.5.2.1 USING EXTERNAL TIMING

The frame timing (integration and row readout) is controlled with the signals CLK_Y, SYNC_Y_READ and SYNC_Y_INTE. A new line time starts at every rising edge on CLK_Y. A line time can have three functions. The function of a line is defined by the state of SYNC_Y_READ and SYNC_Y_INTE at the time of the rising edge on CLK_Y. The three functions of a line can be:

SYNC_Y_READ	***SYNC_Y_INTE***	***Function***
1	1	First line of a frame read-out
0	1	Integration
0	0	Reset. A line is in reset if it is not integrating or being read-out.
1	0	Illegal

As an example, the figure showing the timing of reading out 2 frames of 4 lines with an integration time of 2 lines is repeated, but with the timing of the three input pulses included (Figure 8).

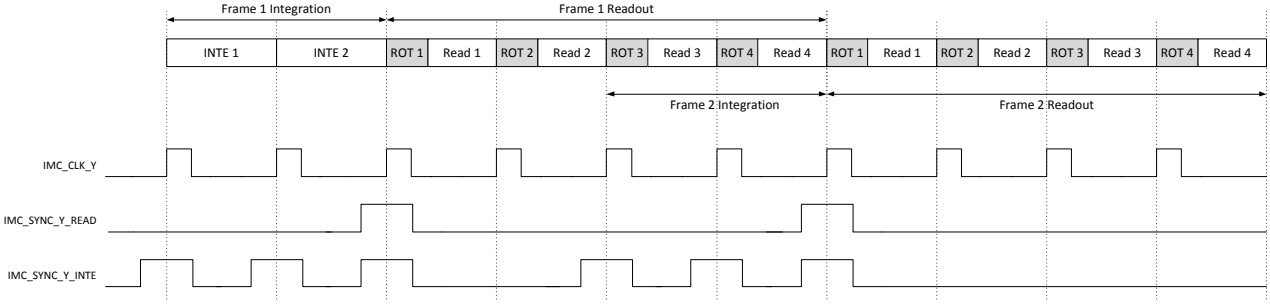


Figure 8: Timing with external signals

The exact timing of these three signals and their relation to the timing of the other digital inputs is specified in more detail in the section “Row overhead timing” (chapter 3.5.4).

Because of the 4T shared pixel architecture, a number of extra rules apply to the timing of CLK_Y, SYNC_Y_READ and SYNC_Y_INTE:

- The integration on a line can only start 2 line times after that line has been read out. This means that if the address settings are unchanged between two frames, there must be two line times with function 'Reset' after the line 'First line' before a line with function 'Integration' can start.
- The y-start address can only change to a new value 2 line times after the line addressed by that new value has been read out.

3.5.2.2 USING THE INTERNAL SEQUENCER

When using the internal sequencer, all image core signals are generated on the chip. The only digital inputs that need to be driven when using the sequencer are CLK_IN, RESET_N, SEQ_START and SEQ_STOP. The SEQ_START is pulsed to initiate a frame transfer and SEQ_STOP can be used to stop a frame transfer before it is finished. The rest of the timing is controlled through register upload. The internal sequencer is enabled by writing a '1' to the register reg_seq_enable.

The most important properties that are controlled through register upload are:

- The number of frames that are grabbed after a pulse on SEQ_START. (Set by register reg_nrof_frames[15:0]. The range is 1 to 65535. If a value of 0 is uploaded, the sensor sends out frames continuously until stopped by a pulse on SEQ_STOP).
- The integration time (expressed as number of line times). (Set by register reg_inte_time[23:0]. The range is 1 to 2^{24}-1. A value of 0 is invalid).
- The window and sub-sampling settings. (Set with registers reg_addr_x_start[9:0], reg_addr_y_start[9:0], reg_addr_x_end[9:0], reg_addr_y_end[9:0] and reg_sub_y. The range of this registers has been discussed before).

The timing generated by the sequencer follows the timing as specified in previous section, with a setting of 60 clock cycles for sample length and 30 clock cycles for transfer length respectively.

3.5.3 ROW OVERHEAD TIMING

During the row overhead time (ROT), the digital chip inputs need to be pulsed when the internal sequencer is not used. Figure 9 shows the required timing of the pulses and their relation to the master clock input.

The timing for the two signals imc_sync_y_read and imc_sync_y_inte is combined into one signal in the figure (imc_sync_y_*). The ROT length is defined in first order by the sample length and the transfer length. Next table gives the details of the programmable timing. The last column details the number of clock cycles (30MHz) for the duration of each state.

#	*Name*	*From*	*To*	*Clock Cycles*
T1		r-edge sync_y_*	r-edge clk_y	3
T2		r-edge clk_y	f-edge clk_y f-edge sync_y_* f-edge pix_reset r-edge col_enable	1
T3		f-edge pix_reset	r-edge col_precharge	0
T4	Precharge length	r-edge col_precharge	f-edge col_precharge	2
T5		f-edge col_precharge	r-edge pix_select r-edge col-init r-edge col_sample r-edge col-sample_r	0
T6	Sample length	r-edge col_sample	f-edge col_sample	60

#	Name	From	To	Clock Cycles
T7		f-edge col_init	f-edge col_sample	14
T8		f-edge col_sample	f-edge col_sample_r/s f-edge pix_select	1
T9		f-edge pix_select	r-edge pix_transfer	1
T10	Transfer length	r-edge pix-transfer	f-edge pix_transfer	30
T11		f-edge pix_transfer	r-edge col_precharge	1
T12		f-edge pix_select	r-edge pix_reset	1
T13		r-edge pix_reset	r-edge sync_x	1
T14		f-edge col_precharge	f-edge col_sample	46

If a value of '0' is entered for one of the registers, the state will be skipped.

Except for states T6 and T7 all states are sequential. If the value entered for T7 is larger than the value of T6, the col_init signal will not go to '1' during the ROT.

IMC_COL_PRECHARGE is not going low after T4, instead it overlaps with IMC_COL_SAMPLE.

The length of the ROT is (T1+T2+T3+2*(T4+T5+T6+T8)+T9+T10+T11+T12+T13+4). With the default values, the ROT length is 168 clock cycles, or 5.56 us. The read out will start after this ROT (4 clock pulses after sync_x rising edge).

The ROT timing can be configured further with the following registers:

- With the register reg_pol, the polarity of all outputs can be set. To achieve the timing as in previous figure, the reg_pol bit for every signal should be '1'. The signal can be inverted by changing the register bit to '0'. Next table shows which register bit maps to which signal.

Reg_pol	output
[0]	imc_sync_x
[1]	imc_clk_y
[2]	imc_sync_y_read
[3]	imc_sync_y_inte
[4]	imc_pix_transfer
[5]	imc_pix_reset
[6]	imc_pix_select
[7]	imc_col_sample
[8]	imc_col_sample_r
[9]	imc_col_sample_s
[10]	imc_col_init
[11]	imc_col_precharge
[12]	imc_col_enable

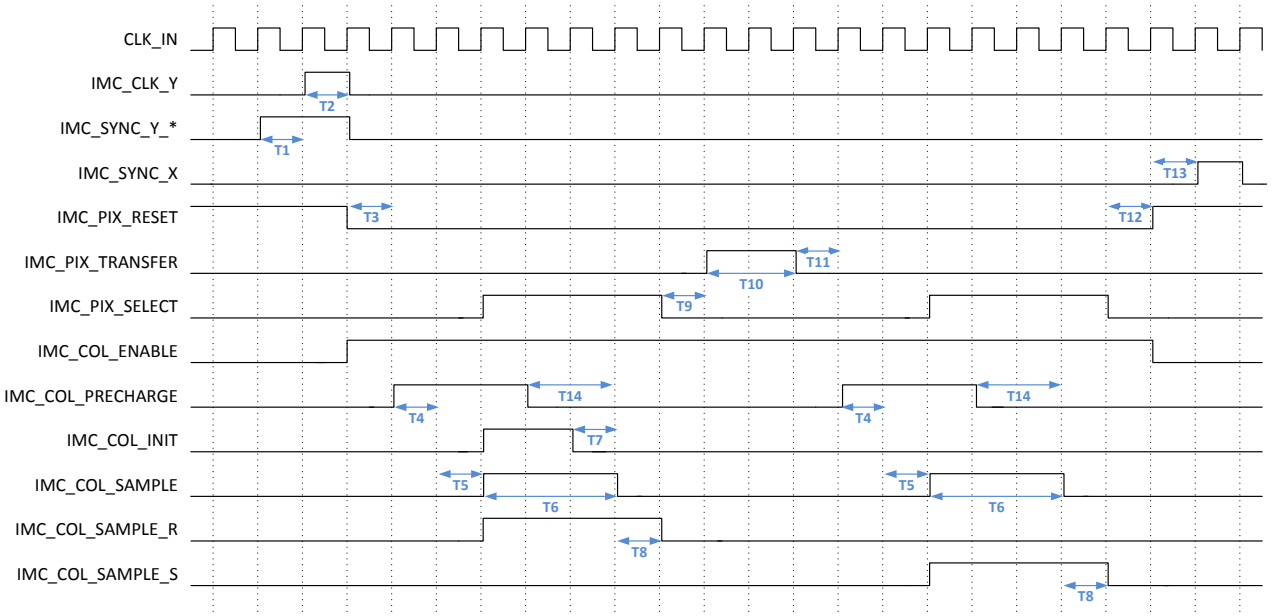


Figure 9: Timing diagram

3.6 WINDOWING & SUBSAMPLING

3.6.1 WINDOWING

The pixel array resolution is 10000 pixels in x direction and 7096 pixels in y direction. On each side of the pixel array, there are a number of dummy pixels (to improve the optical behavior of the pixels at the edges of the active pixel array). For windowing purposes, the rows and columns of the pixel array have an address. When reading out a frame, the start and end positions of the frame in x- and y-direction need to be specified. To reduce the number of available addresses, only every 16th column and every 8th row have an address. This means that the size of a window is always a multiple of 16(x) by 8(y) pixels.

The dummy pixels on each side of the pixel array also have an address (y,x): (0,y), (626,y), (x,0) and (x,888). Figure 10 shows a representation of the pixel array with the addresses in x and y direction.

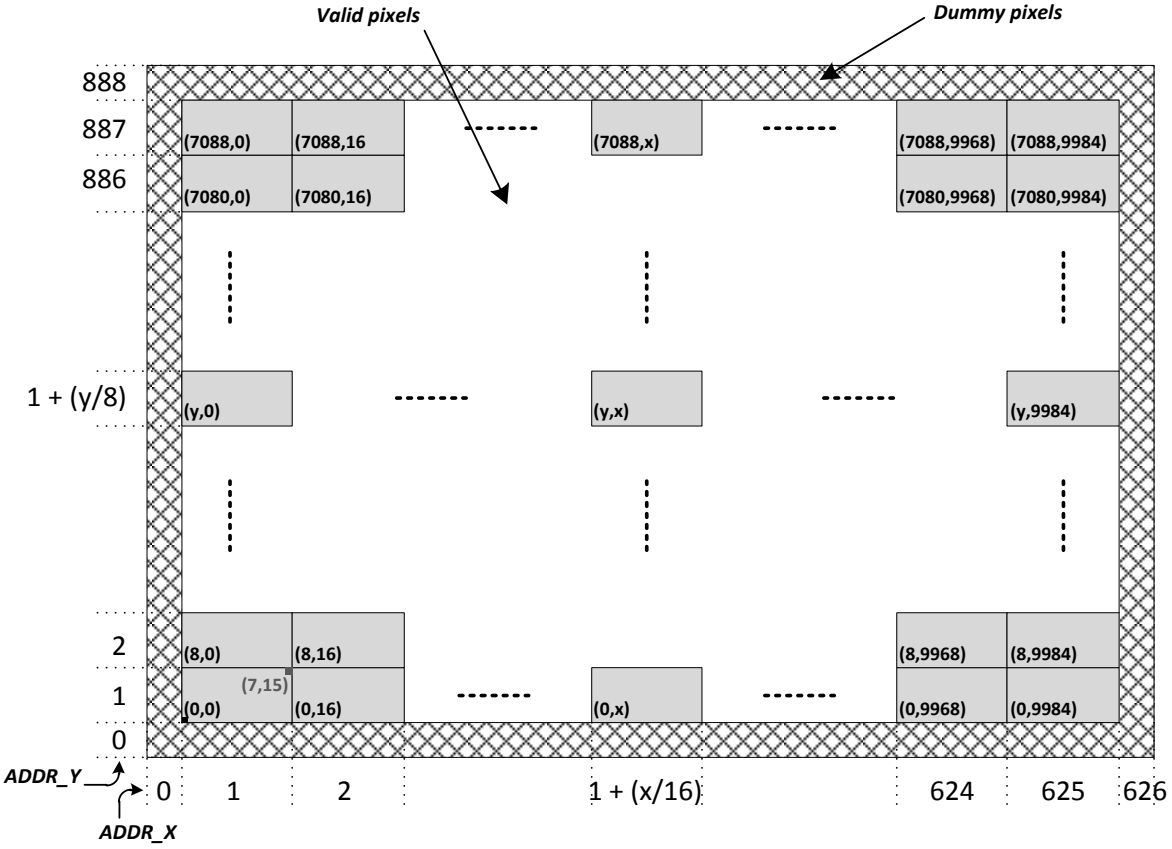


Figure 10: Representation of the pixel array

The position of a pixel is specified as a two dimensional coordinate, written as (y,x). Pixel (0,0) is the left-bottom pixel of the valid pixel array. The valid array extends from pixel (0,0) to pixel (7095, 9999). Pixels (0,0) to (8,15) are addressed with address (1,1). From then on, the y-address increases by 1 every 8 rows and the x-address by 1 every 16 columns. This means that the address of a window with position (y,x) can be written as (1+(y/8) , 1+(x/16)).

The window that is read out is defined by the four 10-bit registers reg_addr_x_start, reg_addr_x_end, reg_addr_y_start and reg_addr_y_end (see later for more details on programmability). The registers reg_addr_x_start and reg_addr_y_start point at the first column and row of the window. The registers reg_addr_x_end and reg_addr_y_end point at the first column and row that are outside the window at its right-top corner.

Example: To read out a 128(y)*64(x) window starting from pixel (24,32):

- The window is between pixels (24,32) and (151,95)

- The address of the first pixel of the window (24,32) maps to address (4,3)
- The address of the first pixel outside of the window (152,96) maps to address (20,7)
- Upload the following values:
  - reg_addr_x_start = 3
  - reg_addr_x_end = 7
  - reg_addr_y_start = 4
  - reg_addr_y_end = 20

A full frame readout (pixel (0,0) to pixel (7095,9999)) requires the following values:

- reg_addr_x_start = 1
- reg_addr_x_end = 626
- reg_addr_y_start = 1
- reg_addr_y_end = 888

3.6.2 SUBSAMPLING

Next to windowing, the sensor also supports sub-sampling in y. Sub-sampling in y can be enabled with a register upload (it's off by default). When sub-sampling is enabled, only every 8^{th} row is read out. The window size in y is divided by 8. Sub-sampling is not possible in the x-direction because of the parallel output structure (8 columns read out in parallel). Sub-sampling in x-direction is actually achieved by ignoring data on some outputs (these outputs can be put in standby in such case).

(Note that the integration time is specified as a number of line times. If the window size in x is changed, the time it takes to read out a line also changes. This means that changing the window size in x also changes the actual integration time.)

3.6.3 MOVING WINDOW

It is possible to change the position of the window every subsequent frame without intermediate register uploads. The window can be moved by setting the registers reg_addr_x_step and reg_addr_y_step. If they are both set to 0, the window does not move. If one of the registers as set to a non-zero value, the window moves in that direction. If both are set, the window moves diagonally.

Window stepping is only allowed if the number of frames is constrained. If the reg_nrof_frames register is set to 0 (continuous frames), the value in the 'step' registers is ignored.

The reg_addr_x/y_step registers are 10 bit wide and represent a signed integer. This allows stepping in both positive and negative directions (the window can move up-left, left, down-left, down, down-right, right, up-right and up). These registers have the same behavior as the windowing registers (steps of 8 and 16).

10 bit	*int*	*Signed int*
00 0000 0000	0	0
00 0000 0001	1	1
…		
01 1111 1111	511	511
10 0000 0000	512	-512
10 0000 0001	513	-511
…		
11 1111 1111	1023	-1

If invalid frame settings are set in moving window mode (addresses out of range), frame grabbing is aborted at the start of the next frame and the sensor will return to idle mode.

3.6.4 INVALID FRAMES

A frame is not started if it has invalid settings. Invalid settings are:

- Start address is bigger than end address in y direction.
- Start address is bigger than end address in x direction.
- Start address and/ or end address out of range in y direction.
- Start address and/ or end address out of range in x direction.
- Integration time setting is 0

If invalid settings are set through a register upload, the next frame is not started and the sensor returns to idle state.

3.7 SPI PROGRAMMING

The register block contains an SPI register interface and the array of registers that are used for the chip programmability. Each register is 8 bit wide and can be individually addressed for read and write.

Data is always captured on the rising edge of SPI_CLK. When reading data from the sensor, the sensor launches data after the rising edge of SPI_CLK. It shall be captured on the next rising edge of SPI_CLK.

Because the SPI clock is internally sampled by the master clock, it should be 4x slower than the input master clock. This means that for a 30 MHz master clock, the SPI clock frequency is 7.5MHz. The edges of the SPI clock should coincide with the falling edges of the master clock.

Also note that the master clock input should always be running when doing a SPI transfer.

The SPI I/O's will not be tri-stated when not active.

3.7.1 SPI WRITE

Data bits are written at each rising edge of the SPI clock (SPI_CLK). The control bit that is clocked in first on the SPI_DIN line is used to indicate a write or read access cycle.

In case this control bit is '1', the data bits are written into the on-chip registers as described above.

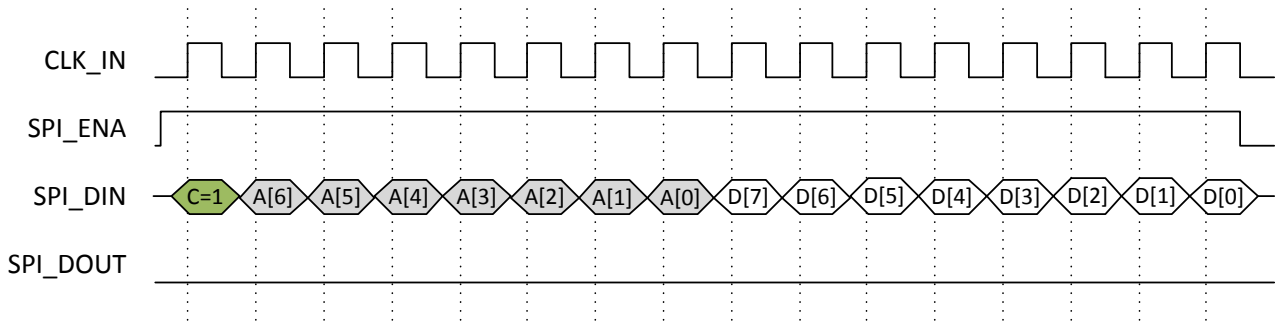


Figure 11: SPI Write timing

3.7.2 SPI READ

When the control bit is '0', the address bits are clocked in. The data bits on the SPI_DIN bus are ignored. The data bits in the on-chip register indicated by the address bits are clocked out (SPI_DOUT) at each falling edge of the clock with a

fixed delay of 75ns from the last falling SPI clock edge of the address. Therefor it is recommended to sample the read out bytes in at the falling SPI clock edge.

If SPI_ENA is made low, the SPI logic immediately resets to an idle state. All transferred bits during the active transfer will be lost. When writing and/or reading multiple registers sequentially, it is allowed to keep SPI_ENA high in-between two transfers. Doing so allows you to continuously read/write data, without a time gap between transfers.

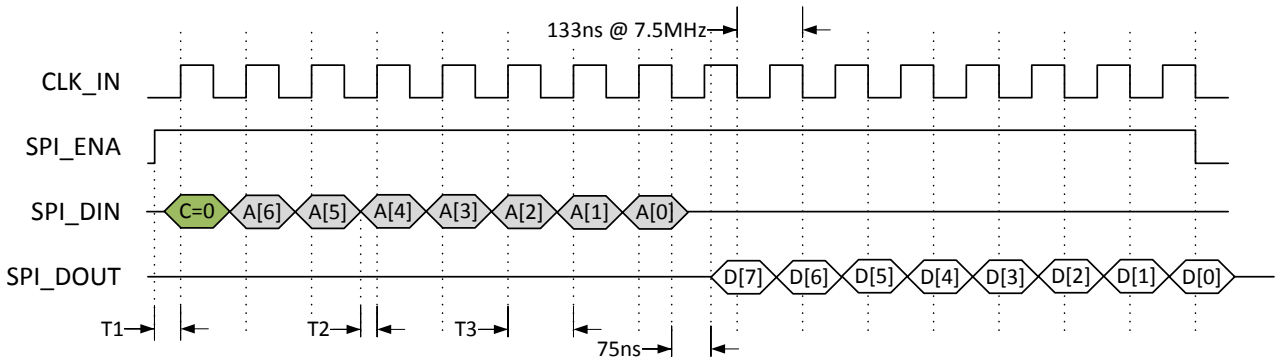


Figure 12: SPI Read timing

#	Description	Requirement
T1	spi_ena setup time before rising edge of spi_clk	Min 0.5x spi_clk period
T2	spi_din setup time before rising edge of spi_clk	Min 0.25x spi_clk period
T3	SPI clock period	Min 4x master clock period

4 READING OUT THE SENSOR

4.1 ANALOG DATA OUTPUTS

The CHR70M sensor has 16 internal output channels which are multiplexed to 8 parallel outputs (OUT_0 to OUT_7)

4.2 TIMING OF CHIP OUTPUTS

There are two digital outputs available that assist frame grabbing. The output OUT_LVAL indicates when valid pixel information is available at the outputs (see Figure 13). The output OUT_CLK_SMP is a sampling clock with the same frequency as the master input clock. This signal can be used to drive external ADC's to ensure correct signal sampling.

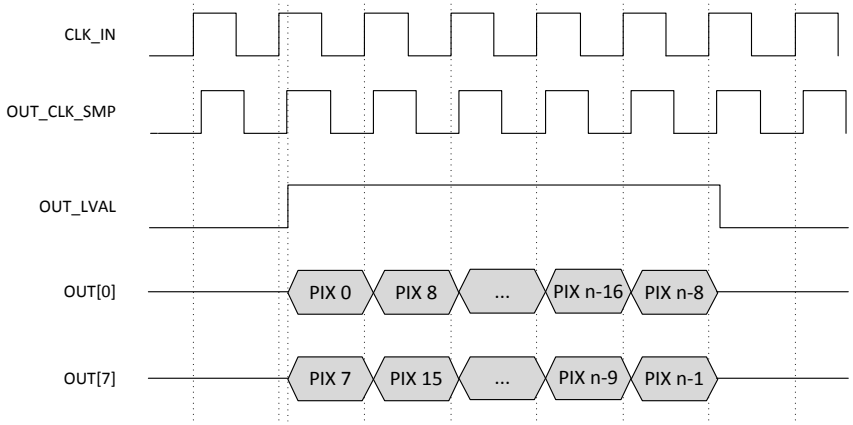


Figure 13: Timing of OUT_CLK_SMP and OUT_LVAL signals

The OUT_LVAL signal is only pulsed when using the internal sequencer. If the sequencer is disabled, it will be '0' continuously.

The output signal OUT_LVAL can be delayed in steps of 1 master clock cycle. This could be used to incorporate the delay of the data path external to the sensor in the timing of OUT_LVAL. The register for this delay value is register 17 (reg_del_lval).

When using external timing, you can use the CLK_Y, SYNC_Y_READ and the SYNC_X pulses to know when data is being output. The data read out starts 4 clock pulses after SYNC_X.

4.3 OUTPUT GAIN AND OFFSET

The gain of the on-chip output amplifier is selectable through the SPI resister settings. The gain varies linearly between 1 and 3.5 in 16 steps (4 bits). Up to gain setting 8, the gain increases by approximately 10-12 % for successive settings. From setting 9 onwards, the gain increases by about 20-22 % for successive settings. Figure 14 shows the gain values for the different gain register settings.

Besides the gain, also the offset level of each channel can be programmed. This allows balancing the random offsets of the different internal channels. Internally, the CHR70M uses 16 analog channels (multiplexed to 8 outputs). That is why 16 offset registers are programmable over SPI (registers 64 – 87). The output buffer has been designed to nominally drive a 20 pF capacitive load at nominal 30 Mpixels/s readout rate (per output). The polarity of the output signal is positive video (dark pixel signals low, bright pixel signals high).

So the analog output swing depends on the gain and offset settings. U can see the different levels and swing in Figure 15.

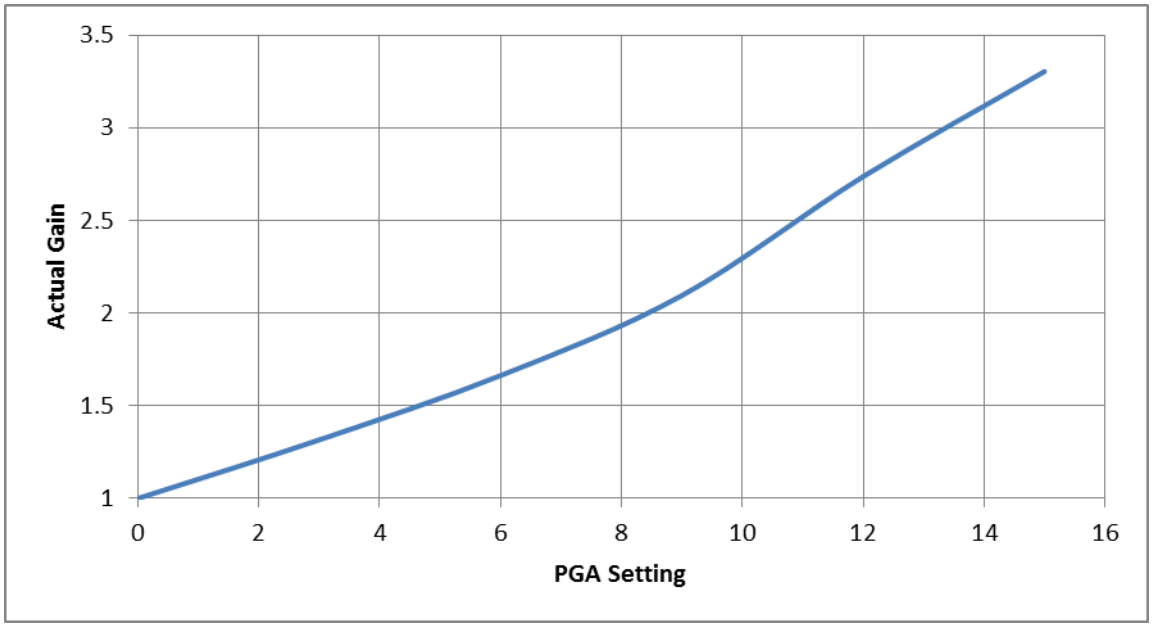


Figure 14: PGA setting vs actual PGA gain

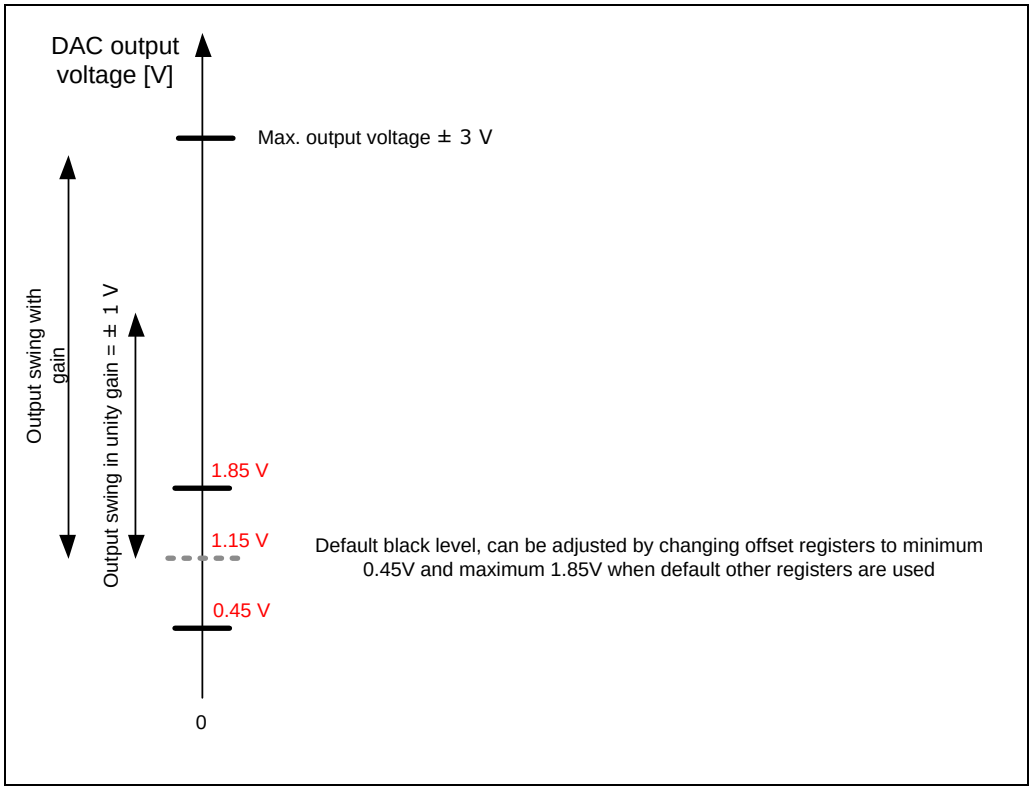


Figure 15: Analog output swing

5 REGISTER OVERVIEW

The next table shows the register map of the sensor.

The first column of the table gives the address that needs to be uploaded during an SPI transfer to interface with the register. Every register address represents 8 register bits. The second column of the table lists which of the 8 bits of the address are used by the register. The last column contains the value the register should be set to after boot up or reset. They can differ from the default startup value of the register.

A number of registers is reserved for internal use only.

Addr	Bits	Name	Description	Default	Recommended
0	[7:0]	reg_addr_x_start[7:0]	Start address in x direction	1	1
1	[7:0]	reg_addr_x_end[7:0]	End address in x direction	114	114
2	[7:0]	reg_addr_x_step[7:0]	Step of moving window in x direction	0	0
3	[1:0]	reg_addr_x_start[9:8]	MSB of address setting	0	0
	[3:2]	reg_addr_x_end[9:8]	MSB of address setting	2	2
	[5:4]	reg_addr_x_step[9:8]	MSB of address setting	0	0
4	[7:0]	reg_addr_y_start[7:0]	Start address in y direction	1	1
5	[7:0]	reg_addr_y_end[7:0]	End address in y direction	120	120
6	[7:0]	reg_addr_y_step[7:0]	Step of moving window in y direction	0	0
7	[1:0]	reg_addr_y_start[9:8]	MSB of address setting	0	0
	[3:2]	reg_addr_y_end[9:8]	MSB of address setting	3	3
	[5:4]	reg_addr_y_step[9:8]	MSB of address setting	0	0
8	[7:0]	reg_inte_time[7:0]	Set integration time	181	181
9	[7:0]	reg_inte_time[15:8]	MSB of integration time	27	27
10	[7:0]	reg_inte_time[23:16]	MSB of integration time	0	0
11	[7:0]	reg_nrof_frames[7:0]	Set number of frames in burst	1	1
12	[7:0]	reg_nrof_frames[15:8]	MSB of number of frames in burst	0	0
13	[0]	reg_seq_enable	Enable internal sequencer	0	0
	[1]	reserved	/	0	0
	[2]	reserved	/	0	0
	[3]	reserved	/	1	1
	[4]	reg_sub_y	Enable sub-sampling in y direction	0	0
	[5]	reserved	/	0	0
	[6]	reserved	/	0	0
14	[7:0]	reg_rot_sample[7:0]		60	60
15	[7:0]	reg_rot_transfer[7:0]		30	30
16	[7:0]	reg_rot_prech[7:0]		2	2
17	[3:0]	reg_del_lval	Delay pulse on OUT_LVAL	1	1
18	[3:0]	reg_pga_gain[3:0]	Gain settings for amplifiers	0	0
18	[4]	reg_pga_gain_uni		0	0
18	[5]	reg_pga_gain_evn		0	0
32	[7:0]	reg_pol[7:0]		255	255
33	[4:0]	reg_pol[12:8]		31	31
33	[5]	reg_pol_amp_phase		1	1
33	[6]	reg_pol_amp_rot		1	1
33	[7]	reg_pol_x_left		1	1
34	[7:0]	reg_rot_clk_y_1[7:0]		3	3
35	[7:0]	reg_rot_clk_y_2[7:0]		1	1
36	[7:0]	reg_rot_prech_nov_1[7:0]		0	0
37	[7:0]	reg_rot_prech_nov_2[7:0]		0	0
38	[7:0]	reg_rot_prech_nov_3[7:0]		1	1
39	[7:0]	reg_rot_sample_nov_1[7:0]		4	**14**
40	[7:0]	reg_rot_sample_nov_2[7:0]		1	1
41	[7:0]	reg_rot_sample_nov_3[7:0]		58	**46**

Addr	Bits	Name	Description	Default	Recommended
42	[7:0]	reg_rot_transfer_nov_1[7:0]		1	1
43	[7:0]	reg_rot_reset_nov_1[7:0]		0	**1**
44	[7:0]	reg_rot_sync_x_nov_1[7:0]		0	**1**
45	[1:0]	reg_bist_start[1:0]		0	0
45	[3:2]	reg_bist_done[1:0]		/	/
46	[4:0]	reg_bist_error[4:0]		/	/
47	[0]	reg_pol_rst_x		1	1
63	[7:0]	reg_revision[7:0]	Revision number of chip	1	1
64	[7:0]	reg_offset_0[7:0]	DAC offset of even pixels in channel 0	0	0
65	[7:0]	reg_offset_1[7:0]	DAC offset of odd pixels in channel 0	0	0
66	[3:0]	reg_offset_0[11:8]	MSB of DAC offset	8	8
66	[7:4]	reg_offset_1[11:8]	MSB of DAC offset	8	8
67	[7:0]	reg_offset_2[7:0]	DAC offset of even pixels in channel 1	0	0
68	[7:0]	reg_offset_3[7:0]	DAC offset of odd pixels in channel 1	0	0
69	[3:0]	reg_offset_2[11:8]	MSB of DAC offset	8	8
69	[7:4]	reg_offset_3[11:8]	MSB of DAC offset	8	8
70	[7:0]	reg_offset_4[7:0]	DAC offset of even pixels in channel 2	0	0
71	[7:0]	reg_offset_5[7:0]	DAC offset of odd pixels in channel 2	0	0
72	[3:0]	reg_offset_4[11:8]	MSB of DAC offset	8	8
72	[7:4]	reg_offset_5[11:8]	MSB of DAC offset	8	8
73	[7:0]	reg_offset_6[7:0]	DAC offset of even pixels in channel 3	0	0
74	[7:0]	reg_offset_7[7:0]	DAC offset of odd pixels in channel 3	0	0
75	[3:0]	reg_offset_6[11:8]	MSB of DAC offset	8	8
75	[7:4]	reg_offset_7[11:8]	MSB of DAC offset	8	8
76	[7:0]	reg_offset_8[7:0]	DAC offset of even pixels in channel 4	0	0
77	[7:0]	reg_offset_9[7:0]	DAC offset of odd pixels in channel 4	0	0
78	[3:0]	reg_offset_8[11:8]	MSB of DAC offset	8	8
78	[7:4]	reg_offset_9[11:8]	MSB of DAC offset	8	8
79	[7:0]	reg_offset_10[7:0]	DAC offset of even pixels in channel 5	0	0
80	[7:0]	reg_offset_11[7:0]	DAC offset of odd pixels in channel 5	0	0
81	[3:0]	reg_offset_10[11:8]	MSB of DAC offset	8	8
81	[7:4]	reg_offset_11[11:8]	MSB of DAC offset	8	8
82	[7:0]	reg_offset_12[7:0]	DAC offset of even pixels in channel 6	0	0
83	[7:0]	reg_offset_13[7:0]	DAC offset of odd pixels in channel 6	0	0
84	[3:0]	reg_offset_12[11:8]	MSB of DAC offset	8	8
84	[7:4]	reg_offset_13[11:8]	MSB of DAC offset	8	8
85	[7:0]	reg_offset_14[7:0]	DAC offset of even pixels in channel 7	0	0
86	[7:0]	reg_offset_15[7:0]	DAC offset of odd pixels in channel 7	0	0
87	[3:0]	reg_offset_14[11:8]	MSB of DAC offset	8	8
87	[7:4]	reg_offset_15[11:8]	MSB of DAC offset	8	8
88	[5:0]	reg_dac_high[5:0]	Internal DAC high setting	48	**52**
89	[5:0]	reg_dac_low[5:0]	Internal DAC low setting	16	**12**
90	[5:0]	reg_rblank_ref[5:0]	Internal reference voltage setting	32	**26**
91	[5:0]	reg_clamp[5:0]	Clamping circuit in output stage	63	63
92	[7:0]	reg_stby[7:0]	Standby register	0	0
93	[3:0]	reg_clk_delay[3:0]	Clock delay for sample signal output stage	5	**0**
	[4]	reg_prebuf_en	Mux buffer before output stage	1	**0**
	[5]	reg_bgap_on	Internal bandgap reference	1	1
	[6]	reg_separate_mux	Separate multiplexer bus lines	1	1
94	[3:0]	reg_pw_ctr_coldriv[3:0]	Current bias column drivers	8	**14**
	[7:4]	reg_pw_ctr_pixcds[3:0]	Current bias column amp CDS stage	8	**14**
95	[3:0]	reg_pw_ctr_colbias[3:0]	Current bias column bus (ctu)	8	**0**
	[7:4]	reg_pw_ctr_colpc[3:0]	Current bias column bus (precharge)	10	**15**
96	[3:0]	reg_pw_ctr_driv[3:0]	Current of the output buffers	8	**1***
	[7:4]	reg_pw_ctr_pga[3:0]	Current bias output amp CDS/PGA stage	8	**8****
97	[3:0]	reg_clk_delay_clk_smp[3:0]	Delay of clk_smp		***

Addr	Bits	Name	Description	Default	Recommended
125	[0]	reg_latch_temp_data	Temperature sensor latch		
126	[7:0]	reg_temperature[7:0]	Temperature sensor read out	0	0
127	[7:0]	reg_temperature[15:8]	Temperature sensor read out	0	0

* System dependent: this depends on the output load of the sensor in the camera, the higher the load, the higher this value should be at the expense of power consumption. Increasing the current of the output buffers will increase the slew rate of the outputs. This can be used to adapt the sensor outputs to the receiver systems input capacitance.

** Testing was done with setting 14, setting 1 shows no visible degradation.

*** This register can be used to fine-tune the output clock delay for your specific receiver (ADC) to the data. The delay can be set between 0 - 180°.

The registers reg_revision[7:0] and reg_temperature[15:0] are read-only registers. A write operation to one of these registers will not change their contents.

All addresses that are not mentioned in this list are reserved and may not be overwritten.

6 SPECIFICATIONS

Specification	value	Comment
Geometry		
Effective pixels	10000 x 7096	10032 x 7112 including dummy pixels
Pixel pitch	3.1 x 3.1 μm^2	2-shared pixel
Pixel		
Full well charge	>13 Ke^-	Pinned photodiode pixel, two shared pixel
Signal swing	1 V	@ sensor output in lowest gain mode
Conversion gain	63 $\mu V/e^-$	@ sensor output in lowest gain
Sensitivity	0.15 A/W	@550nm
Temporal noise	10 e^-	@ sensor output in lowest gain mode
Dynamic range	63 dB	Full well charge / temporal noise
Shutter type	Rolling shutter	With integration time control
Micro lenses	Possible	If required. 40 % gain in QE x FF expected.
Fill Factor	50 %	See comment microlenses
QE x FF	28 %	@ 550 nm, w/o microlenses
Dark current signal	3.2 e^-/s	@ Room temperature
DSNU	6 e^-/s	@ Room temperature
Fixed Pattern Noise	0.09%	% of full swing (RMS)
PRNU	1.5%	RMS
Image lag	<0.1%	
Readout		
Output channels	8	8 output channels at 30 MHz
Frame rate	3 fps	@ full resolution; using a 30 MHz pixel clock
Functionality and system aspects		
Timing generation	External (default) On-chip	Off-chip allows more flexibility but on-chip can be activated with some extent of programmability
Variable gain amplifier	x 1-4 in output stage	Programmable through SPI 16 settings
Programmable Registers	Sensor parameters	Window coordinates, timing parameters, gain & offset per channel, exposure times Via SPI interface.
Interface	Analog outputs	
I/O logic levels	CMOS 3.3 V	
Supply voltage	3.3 V	Possibly separate voltages for optimal operation
Color filters	RGB	Bayer pattern
Power	435 mW	
Package	Ceramic package	Custom PGA (65 pins)
Cover glass		Plain AR coated glass or with IR filter.
Technology	0.18 µm CMOS	
Operating range	0°C to +60°C Ambient	Dark current and noise performance will degrade at higher tempertures
ESD	Class 1A HBM Class 4C CDM	
RoHS	Compliant	Directive 2002/95/EC

7 MECHANICAL SPECIFICATIONS

7.1 PACKAGE

The sensor is packaged in a custom ceramic PGA. Figure 15 shows as the package drawing. It has no mounting holes. In the current design 65 pins are provided. There is a single line of 13 pins on one side of the package and 4 lines of 13 pins on the other side. The 4 corner pins have stand-off rings.

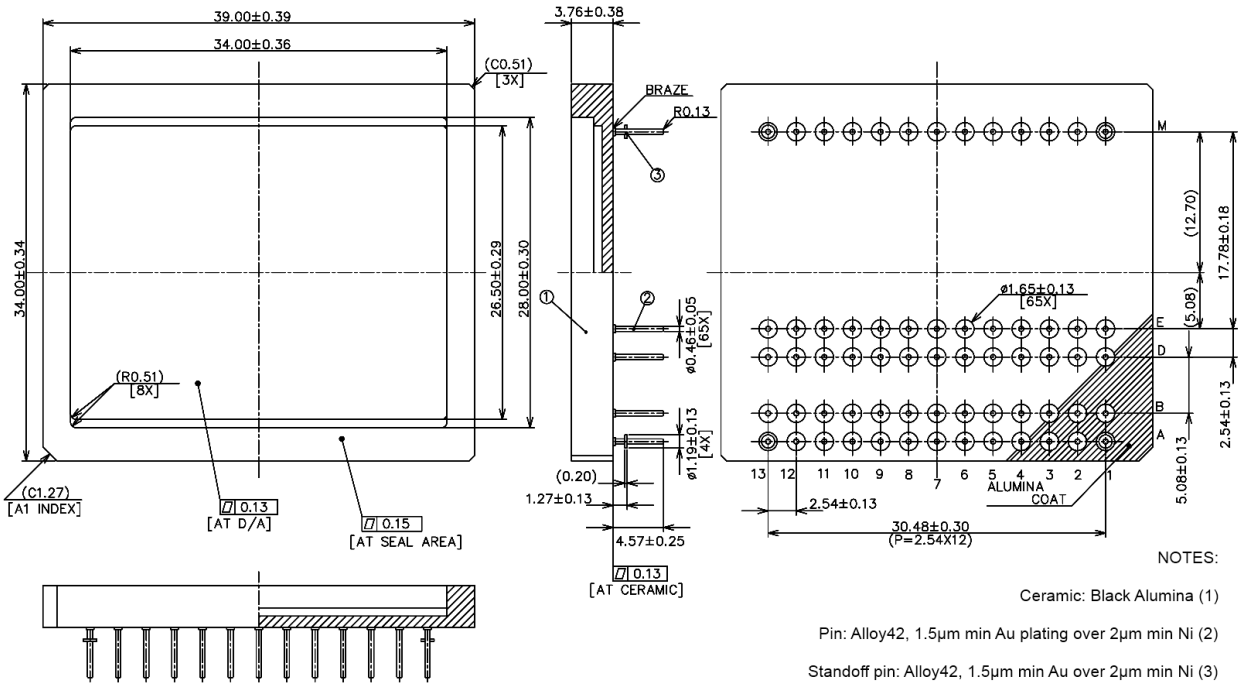


Figure 16: Package drawing of CHR70M

7.2 SENSOR FLOOR PLAN

Figure 16 shows the sensor floor plan. The pixel array is centered in the horizontal direction. Pixel (0, 0) is at the bottom left position. Bond pad 1 is also situated at the bottom left. The numbering of the bond pads is counter clock wise.

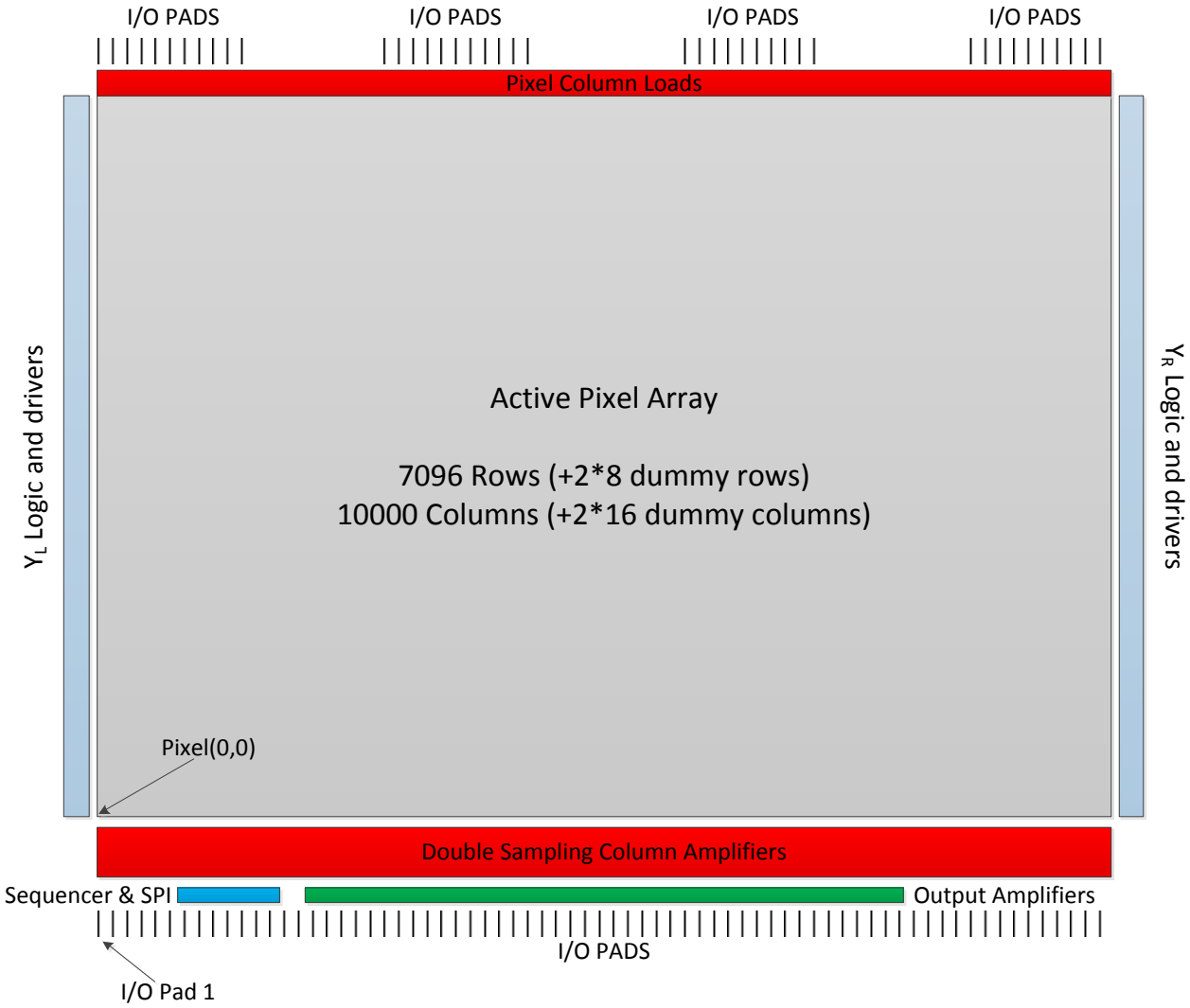


Figure 15. Floor plan of the sensor.

7.3 Assembly Drawing

The figure below shows the assembly of the CHR70M image sensor die inside the ceramic package.

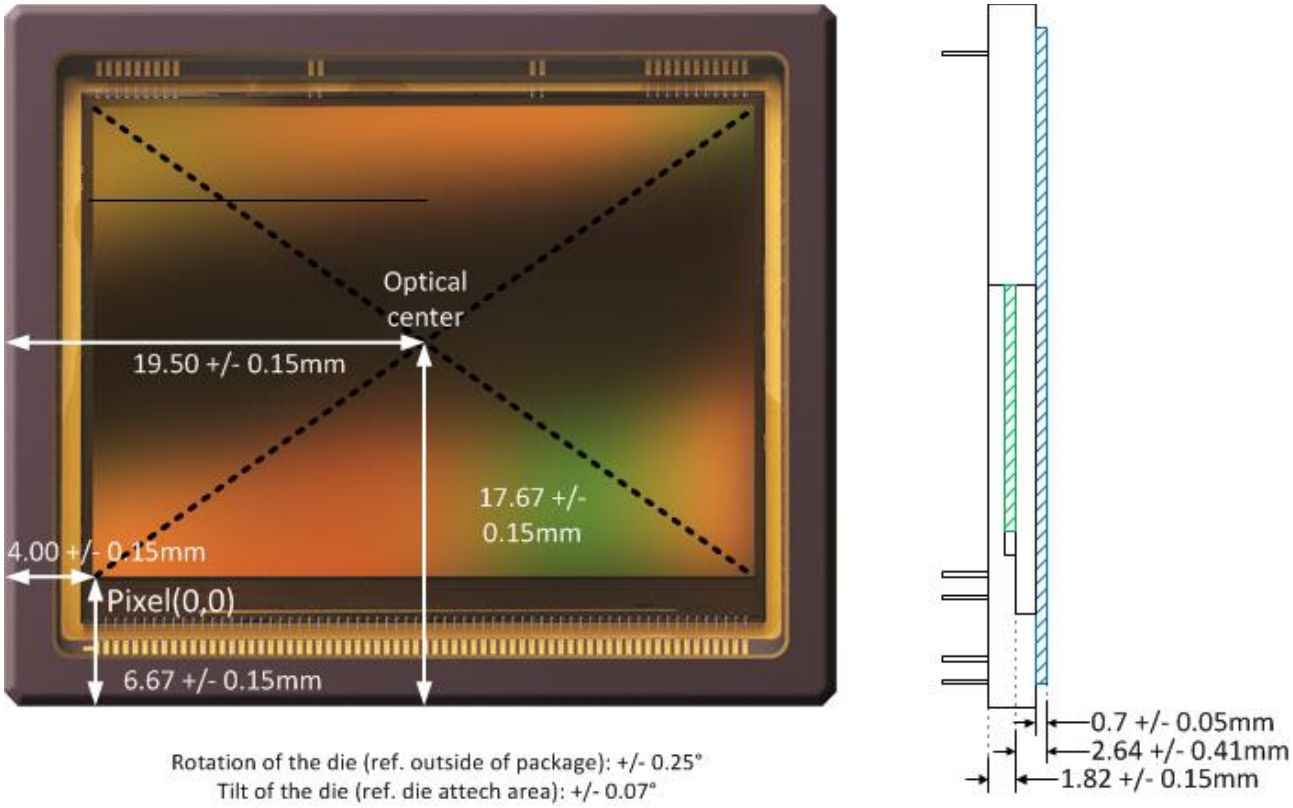


Figure 17: Assembly drawing

7.4 Cover glass

The CHR70M image sensor is available with a regular D263 glass lid. See the transmittance curve of this glass below.

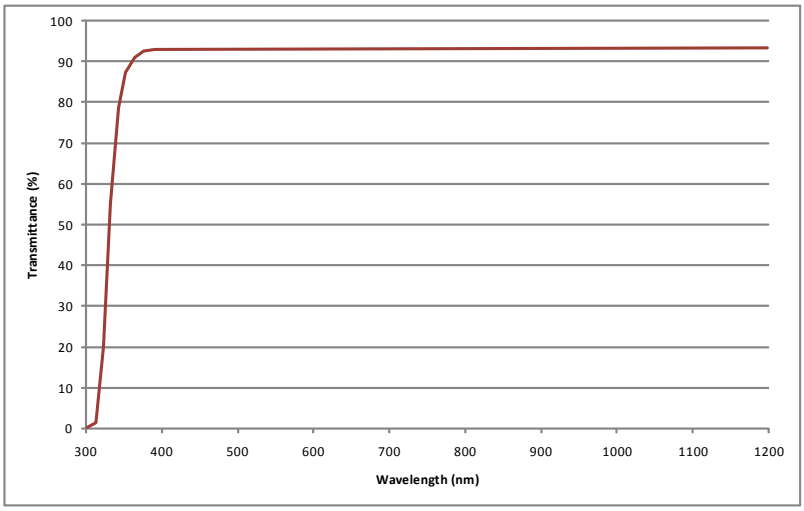


Figure 18: D263 transmittance curve

Both types of glass have an AR coating with a reflectance of 0.5% at 500nm

7.5 COLOR FILTER

A RGB Bayer pattern is used on the CHR70M image sensor. The order of the RGB filter can be found in the drawing below.

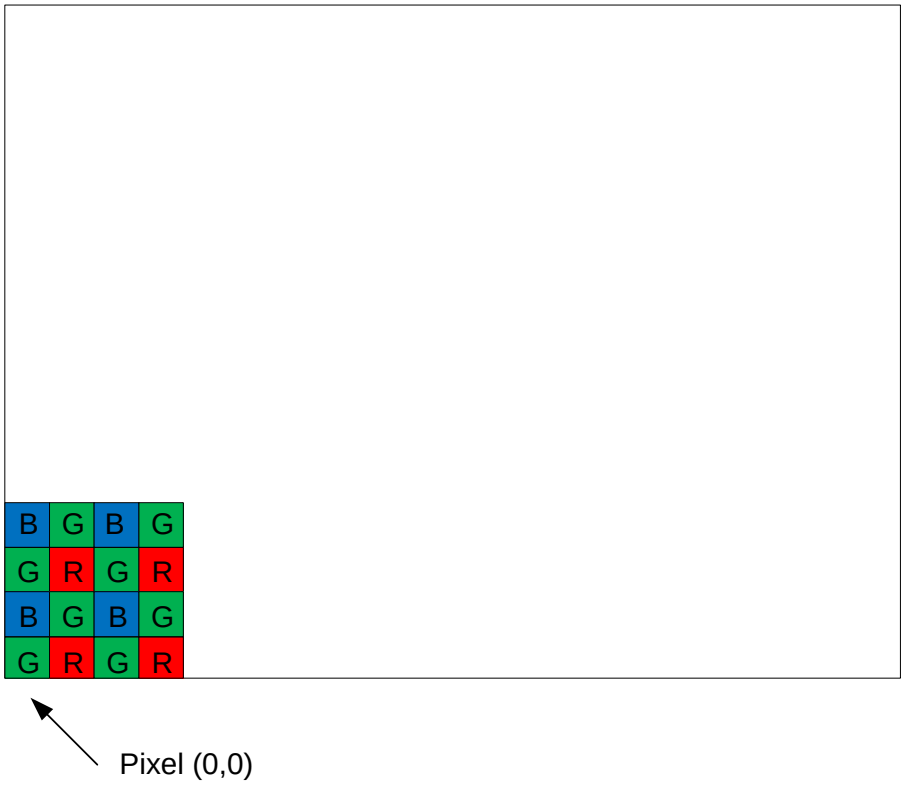


Figure 19: RGB Bayer pattern order

8 RESPONSE

Figure 2 shows the typical response curve of the CHR70M pixel.

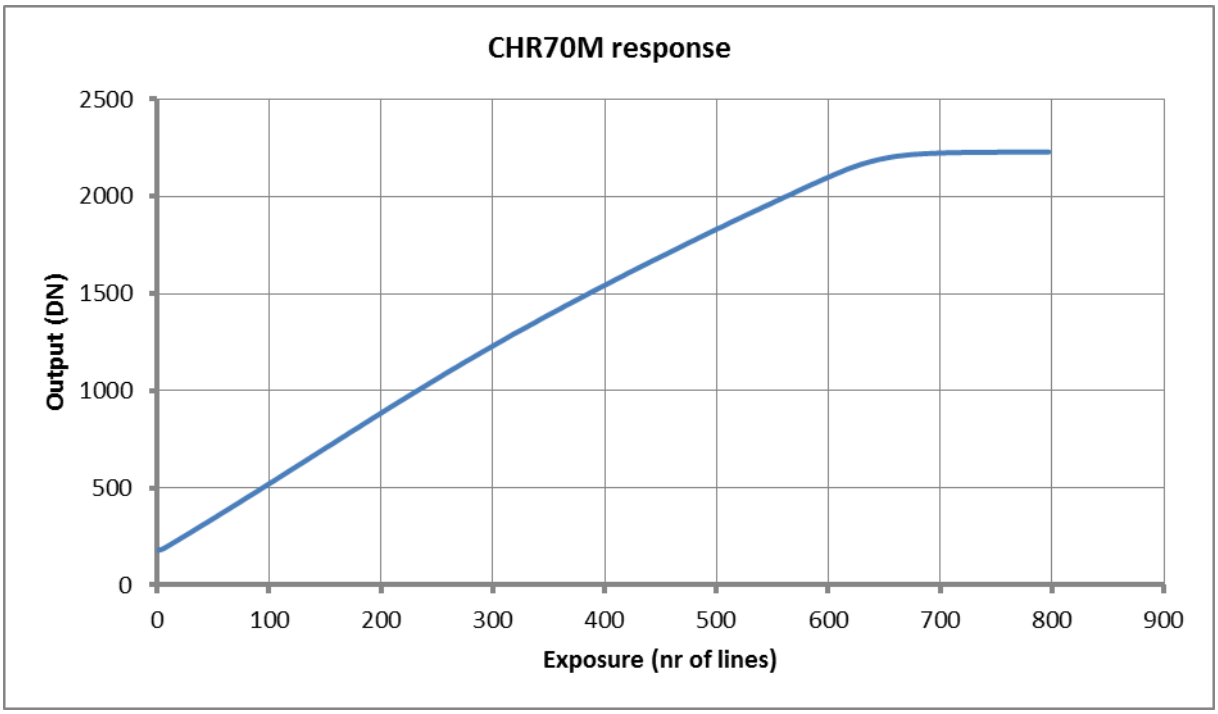


Figure 20: Response curve of CHR70M image sensor

A full well charge of 13Ke$^-$ with a conversion gain of 63µV/e$^-$ is achieved.

9 SPECTRAL RESPONSE

The figures below show the spectral response and QE curves of the mono/color CHR70M image sensor with and without micro-lenses and cover glass.

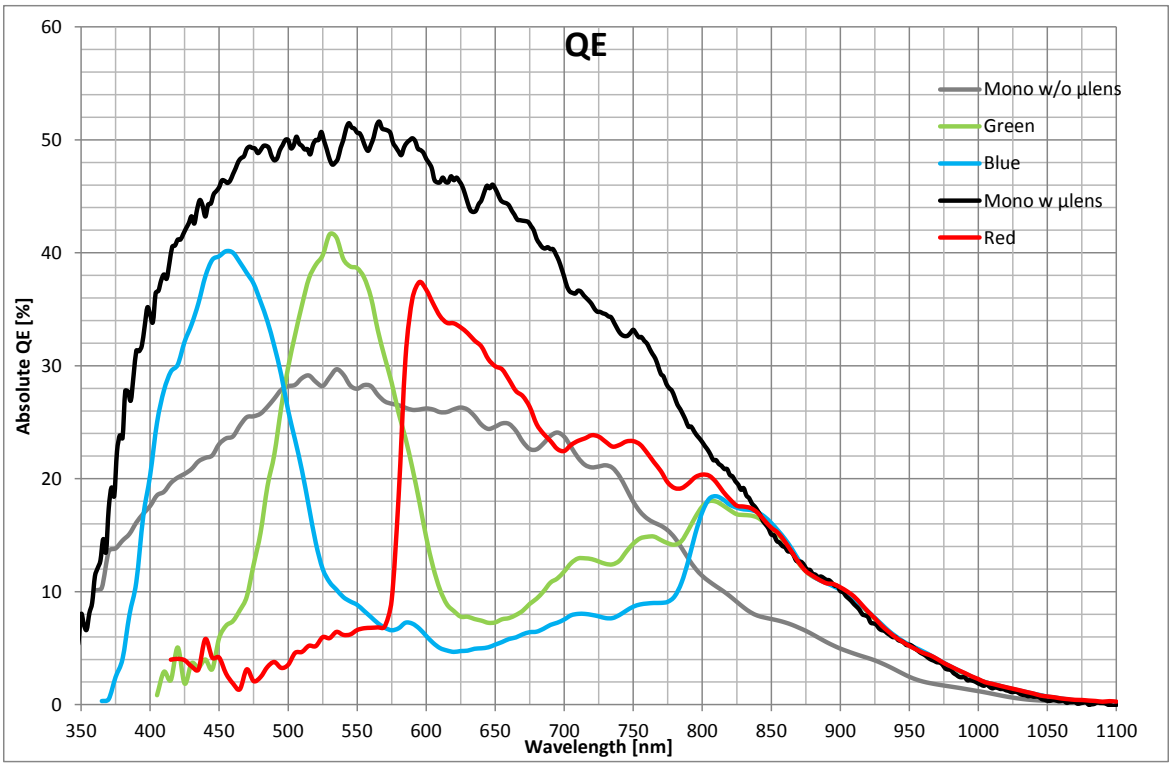


Figure 21: Typical QE of a CHR70M

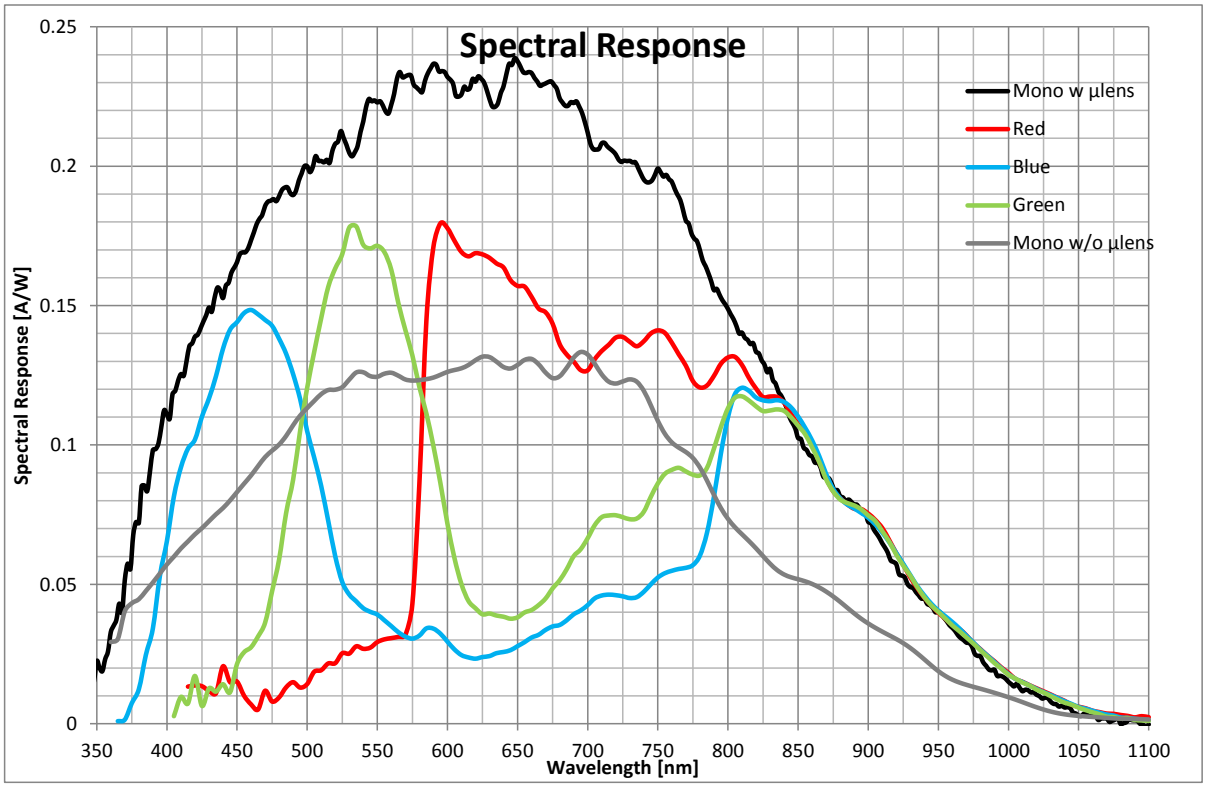


Figure 22: Typical spectral response of a CHR70M

10 CHIEF RAY ANGLE (CRA)

To improve light sensitivity in the pixels towards the edges of the pixel array, the micro lenses on top of the pixels are shifted linearly towards the corners/edges. The shift is designed so the CRA of the sensor is 22.3° in the corners.

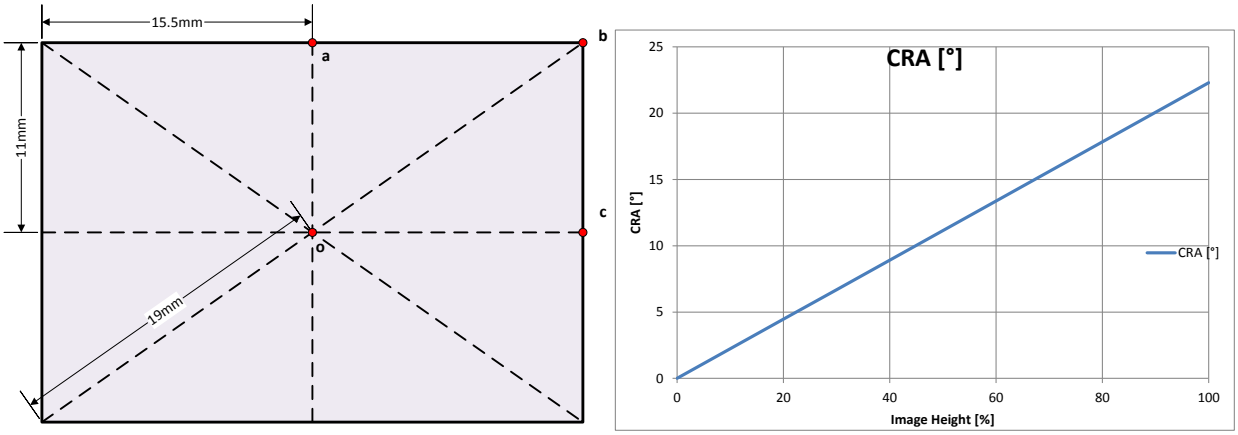


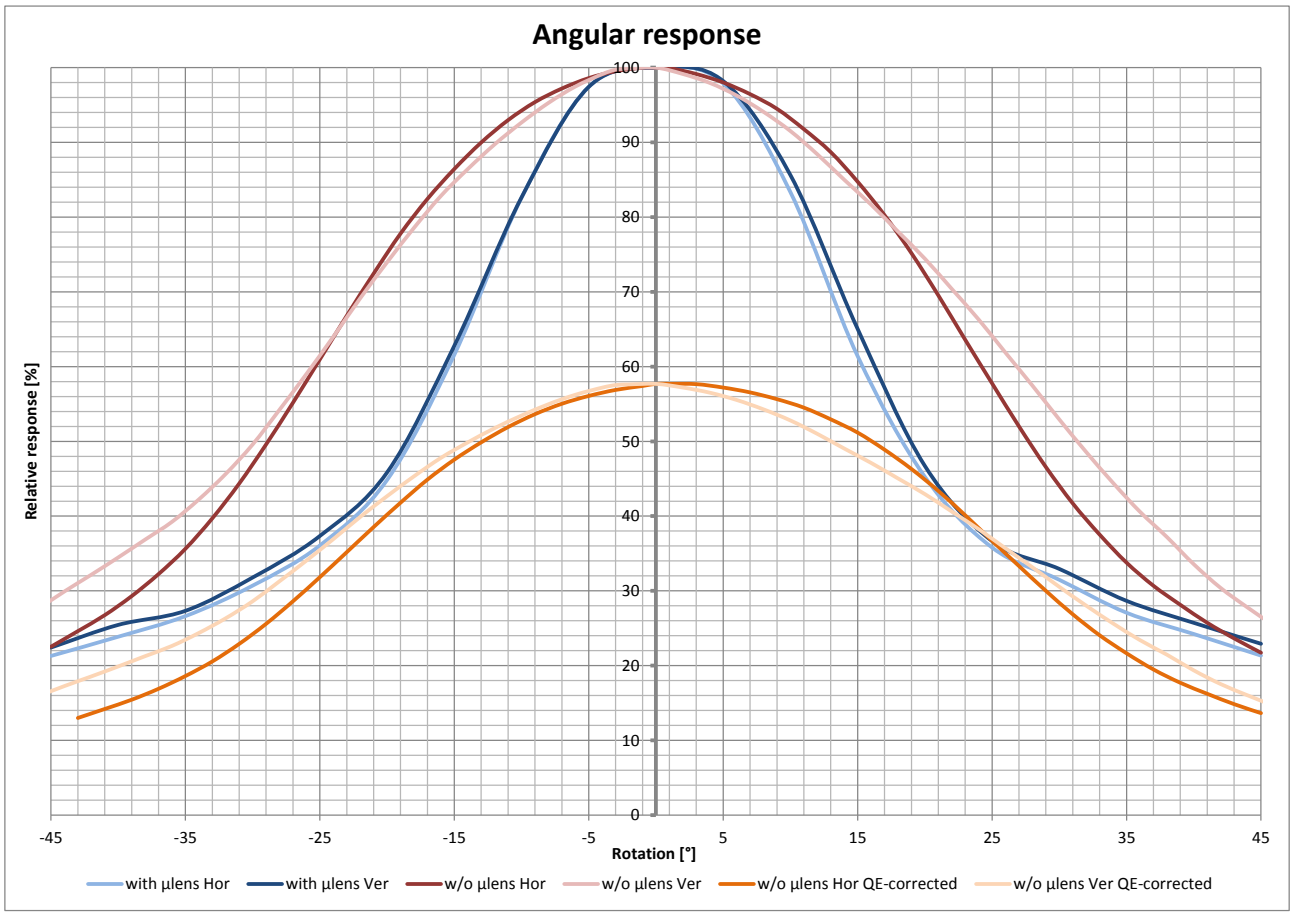
Figure 23: CRA vs. image height

Point	Image Height [%]	Image Height [mm]	CRA [°]
o	0	0	0
	10	1.9	2.23
	20	3.8	4.46
	30	5.7	6.69
	40	7.6	8.92
	50	9.5	11.15
a	57.9	11	12.91
	60	11.4	13.38
	70	13.3	15.61
	80	15.2	17.84
c	81.58	15.5	18.2
	90	17.1	20.07
b	100	19	22.3

11 ANGULAR RESPONSE

Below you can see the relative response of the CHR70M with and without micro lenses. The sensors without micro lenses have a lower QE, so with the same amount of light, they will have a lower response. This is also added in to the plot below (QE-corrected curves).

This angular response is measured on the center pixels of the array where the micro-lens shift is minimal.



12 PIN LIST

The pin list of the packaged sensor is given in the table below.

Pin	Name	Description	Remarks
A1	CLK_IN	Master Clock Input	
A2	SEQ_STOP	Stop sequencer operation	
A3	OUT_LVAL	Indicates if outputs when valid (pixel signal)	
A4	SYNC_Y_READ	Sets y read shift register to start address	external signal for sequencer bypass
A5	OUT_0	analog sensor output	max. 20pF
A6	OUT_2	analog sensor output	max. 20pF
A7	OUT_4	analog sensor output	max. 20pF
A8	OUT_6	analog sensor output	max. 20pF
A9	VDD_D	digital power supply	250mA peak current (all pins)
A10	SPI_DOUT	SPI data output (data from slave to master)	
A11	VDD_D	digital power supply	250mA peak current (all pins)
A12	EOS_X	EOS from shift register in x direction	Not used. Do not connect.
A13	VREF	internal bandgap reference voltage	connect to 100nF capacitor to GND (or force bias reference voltage**)
Pin	**Name**	**Description**	**Remarks**
B1	GND	0V reference	
B2	SYNC_X	Sets x shift register to start address	external signal for sequencer bypass
B3	OUT_CLK_SMP	Sample clock (indicates when outputs are best sampled)	
B4	SYNC_Y_INTE	Shifts input of y inte shift register at start address	external signal for sequencer bypass
B5	OUT_1	analog sensor output	max. 20pF
B6	OUT_3	analog sensor output	max. 20pF
B7	OUT_5	analog sensor output	max. 20pF
B8	OUT_7	analog sensor output	max. 20pF
B9	SPI_ENA	SPI enable signal (data transfer only valid when high)	
B10	SPI_DIN	SPI data input (data from master to slave)	
B11	GND	0V reference	
B12	PH2	test diode, full n-well diode	Not used. Do not connect.
B13	DAC_LOW	DAC low reference voltage	connect to 100nF capacitor to GND (or force bias voltage**)
Pin	**Name**	**Description**	**Remarks**
D1	RESET_N	Global asynchronous active low reset	
D2	CLK_Y	Clock for y shift register	external signal for sequencer bypass
D3	PIX_RESET	Reset signal for pixels	external signal for sequencer bypass
D4	COL_SAMPLE	Sample signal for columns	external signal for sequencer bypass
D5	VDD_A	analog power supply	65mA normal operation (all pins)
D6	GND	0V reference	
D7	GND	0V reference	
D8	VDD_A	analog power supply	65mA normal operation (all pins)
D9	COL_ENABLE	Enable signal for column amplifier	external signal for sequencer bypass
D10	SPI_CLK	SPI Clock input	
D11	GND	0V reference	
D12	PH1	test diode, pixel array	Not used. Do not connect.
D13	DAC_HIGH	DAC high reference voltage	connect to 100nF capacitor to GND (or force bias voltage**)
Pin	**Name**	**Description**	**Remarks**
E1	SEQ_START	Start sequencer operation	

Pin	Name	Description	Remarks
E2	PIX_TRANSFER	Transfer signal for pixels	external signal for sequencer bypass
E3	PIX_SELECT	Select signal for pixels	external signal for sequencer bypass
E4	COL_SAMPLE_R	Sample reset signal for columns	external signal for sequencer bypass
E5	COL_SAMPLE_S	Sample signal signal for columns	external signal for sequencer bypass
E6	GND	0V reference	
E7	VDD_A	analog power supply	65mA normal operation (all pins)
E8	COL_INIT	Init signal for column amplifier	external signal for sequencer bypass
E9	COL_PRECHARGE	Precharge signal for columns	external signal for sequencer bypass
E10	CMD_COL1	bias setting column amp 1	connect to 100nF capacitor to VDD_ANA (or force bias voltage**)
E11	CMD_COL2	bias setting column amp 2	connect to 100nF capacitor to VDD_ANA (or force bias voltage**)
E12	CMD_OUT1	bias setting output amp 1 (CDS/PGA)	connect to 100nF capacitor to GND (or force bias voltage**)
E13	CMD_OUT2	bias setting output amp 2 (output buffer)	connect to 100nF capacitor to GND (or force bias voltage**)
Pin	**Name**	**Description**	**Remarks**
M1	VDD_TRANSFER	power supply for the transfer pixel lines	peak current at start up*, 10mA peak normal operation (all pins)
M2	VDD_RESET	power supply for the reset pixel lines	peak current at start up*, 10mA peak normal operation (all pins)
M3	GNDAB	anti-blooming supply (reset and transfer low level)	Sink current, max. 10mA (all pins). Can be tied to analog ground.
M4	EOS_Y_L_READ	EOS from left read shift register in y direction	Not used. Do not connect.
M5	EOS_Y_L_INTE	EOS from left integration shift register in y direction	Not used. Do not connect.
M6	VDD_PIX	pixel array power supply	Has large peak currents during RBT
M7	GND	0V reference	
M8	EOS_Y_R_INTE	EOS from right integration shift register in y direction	Not used. Do not connect.
M9	EOS_Y_R_READ	EOS from right read shift register in y direction	Not used. Do not connect.
M10	CMD_COL3	bias setting column load 1 (ctu)	connect to 100nF capacitor to VDD_PIX (or force bias voltage**)
M11	CMD_COL4	bias setting column load 2 (precharge)	connect to 100nF capacitor to VDD_PIX (or force bias voltage**)
M12	VDD_PIX	pixel array power supply	peak currents during RBT
M13	GND	0V reference	

* Possibly large peak current at start-up; may be limited by resistance to pin or slower ramp-up

** Bias signals generated on-chip, forced outside if needed.

13 HANDLING AND SOLDERING

13.1 SOLDERING

13.1.1 MANUAL SOLDERING

Use partial heating method and use a soldering iron with temperature control. The soldering iron tip temperature is not to exceed 350°C with 270°C maximum pin temperature, 2 seconds maximum duration per pin. Avoid global heating of the ceramic package during soldering. Failure to do so may alter device performance and reliability.

13.1.2 WAVE SOLDERING

Wave soldering is possible but not recommended. Solder dipping can cause damage to the glass and harm the imaging capability of the device. See the figure below for the wave soldering profile.

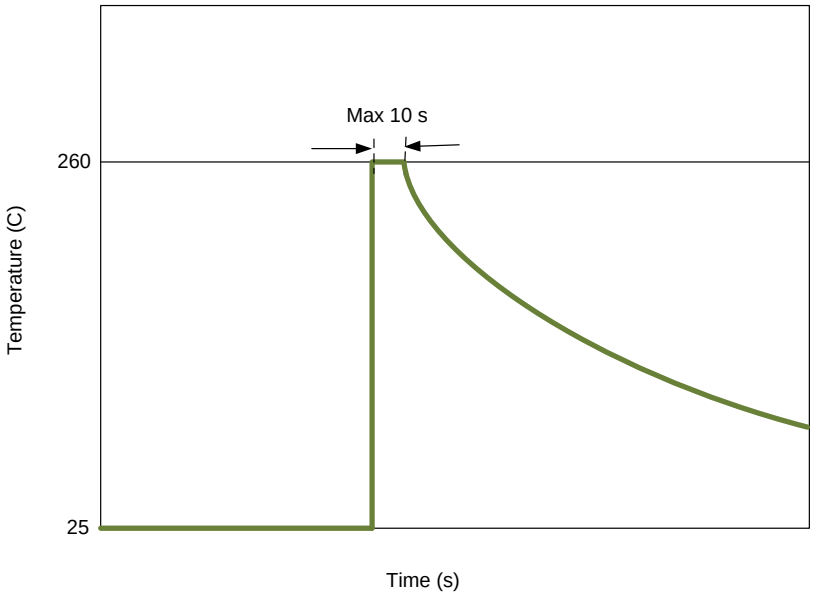


Figure 24: Wave solder profile

13.1.3 REFLOW SOLDERING

The figure below shows the maximum recommended thermal profile for a reflow soldering system. If the temperature/time profile exceeds these recommendations, damage to the image sensor can occur.

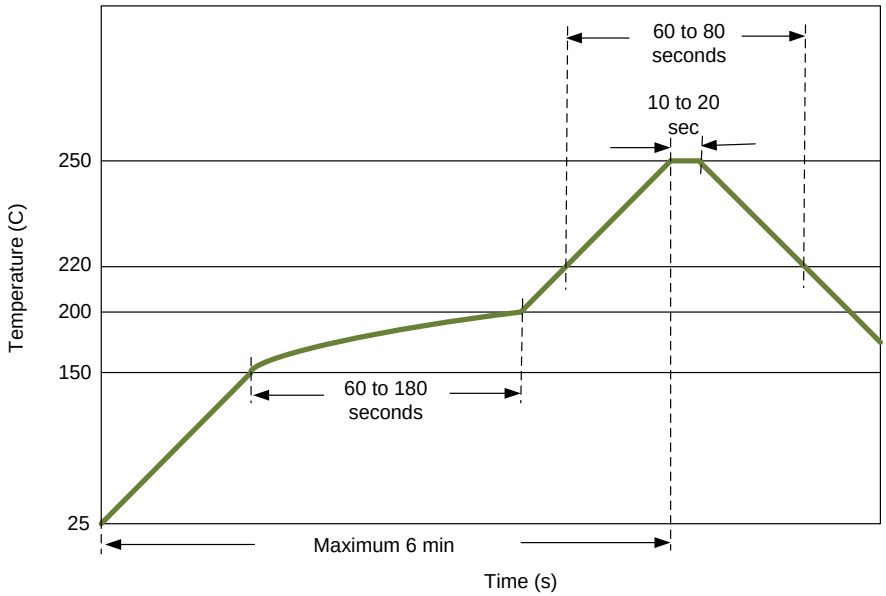


Figure 25: Reflow solder profile

13.1.4 Soldering Recommendations

Image sensors with filter arrays (CFA) and micro-lens are especially sensitive to high temperatures. Prolonged heating at elevated temperatures may result in deterioration of the performance of the sensor. Best solution will be flow soldering or manual soldering of a socket (through hole or BGA) and plug in the sensor at latest stage of the assembly/test process. The BGA solution allows more flexibility for the routing of the camera PCB.

13.2 Handling image sensors

13.2.1 ESD

The following are the recommended minimum ESD requirements when handling image sensors.

1. Ground workspace (tables, floors…)
2. Ground handling personnel (wrist straps, special footwear…)
3. Minimize static charging (control humidity, use ionized air, wear gloves…)

13.2.2 Glass cleaning

When cleaning of the cover glass is needed we recommend the following two methods.

1. Blowing off the particles with ionized nitrogen
2. Wipe clean using IPA (isopropyl alcohol) and ESD protective wipes.

13.2.3 Image sensor storing

Image sensors should be stored under the following conditions

1. Dust free
2. Temperature 20°C to 40°C
3. Humidity between 30% and 60%.
4. Avoid radiation, electromagnetic fields, ESD, mechanical stress

13.3 Excessive light

Excessive light falling on the sensor can cause heating up the micro lenses and color filters. This heat can cause deforming of the lenses and/or deterioration of the lenses and color filters by making them more opaque, increasing the heat up even more. Avoid shining high intensity light upon the sensors for extended periods of time. In case of lasers, they can cause heat up but can also damage the silicon die itself.

14 ORDERING INFORMATION

Part Number	Chroma	Microlenses	Package	Glass
CHR71000ES - 1E5C1PA	RGB Bayer	yes	Pin grid array	D263 AR coated
CHR71000ES - 1E5M1PA	mono	yes	Pin grid array	D263 AR coated
CHR71000ES - 1E5M1PN	mono	yes	Pin grid array	*Removable D263 AR coated
CHR71000HGES - 1E5M1PA **	mono	yes	Pin grid array	D263 AR coated

* Removable glass will be attached to the sensor with silicon glue which can be easily removed from the package.

** High Grade variant; has no defect rows or columns.

The ES (Engineering Sample) status will be removed from the part number once the qualification tests are done. There will be no physical changes between an ES and non-ES device.