

AM256: 256 Channel Linear Multiplexer for Photoconductive Detectors

FEATURES

- 256 Channels per die
- Per channel dark current subtraction
8 Bit DAC correction
Low Noise Subtractor
- Expansion to 2 or 4 die arrays
- High linearity design: Typ. < 0.1%
- Single 5 Volt power supply
- Low power design Typ. 5mW
- Advanced Readout System
 - Reversible readout direction
 - Partial Readout - Windowing
 - Fast - typ 4MHz max readout
 - Read while integrate or
 - Read after integrate
- Globally selectable charge well size
Typ. 1pF to 20pF in 8 steps
- Serial control interface
- TTL compatible digital inputs
- Detector input pads on 50 μ centers
- Excellent for Lead Salt applications
- Interface Electronics available
 - USB Interface
 - Integrated TE Controller
 - Designed for 28 pin tub pkg.
 - Windows base software available

PHYSICAL

- Die Size Typ. 12.90mm x 3.04mm
- Die Thickness Typ. 0.7mm
- Detector Bond Pads
 - 70 μ x 200 μ pad opening
 - 50 μ center to center spacing on two staggered rows
- Power and Ground Bond Pads
 - 165 μ x 200 μ Pad Opening
- IO Bond Pads
 - 100 μ x 200 μ Pad Opening

ABSOLUTE MAXIMUM RATINGS

- VDD 5.5V
- Any Pin VDD+0.5V, VSS-0.5V
- Storage Temperature 50K to 150C
- Operating Temperature 77K to 50C
- All typical ESD protection measures should be observed when handling bare die.

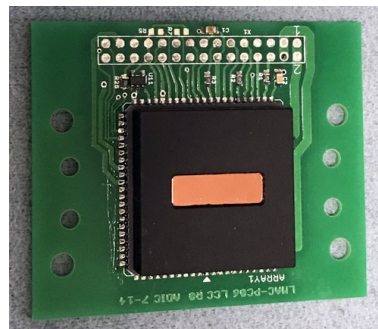
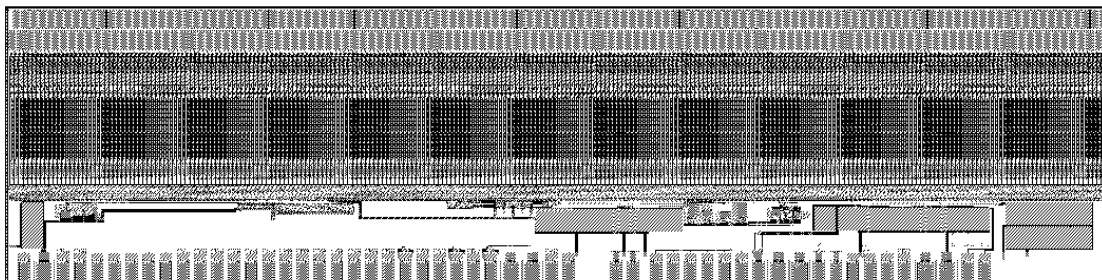


Figure 2: An LCC array



Functional Description

The AM256 is a state of the art, low noise, high performance readout integrated circuit (ROIC), designed for use with photoconductive infrared detectors e.g. PbS and PbSe. (See figure 3) Each of the 256 channels (pixels) is designed to provide current skimming so that the majority of the dark current can be removed while integrating and amplifying the photo current. The detector attaches to a buffered director injector stage which feeds the charge well and the two dark current skimming circuits. The first skimming circuit is a global current skimmer, which subtracts the same amount of dark current from each of the 256 channels. The second current skimmer is a per pixel skimmer that is adjustable over a user controlled range through the use on an internal, per pixel, 8 bit digital memory. The per pixel skimmer allows the user to reduce the detector dark current non uniformity by a

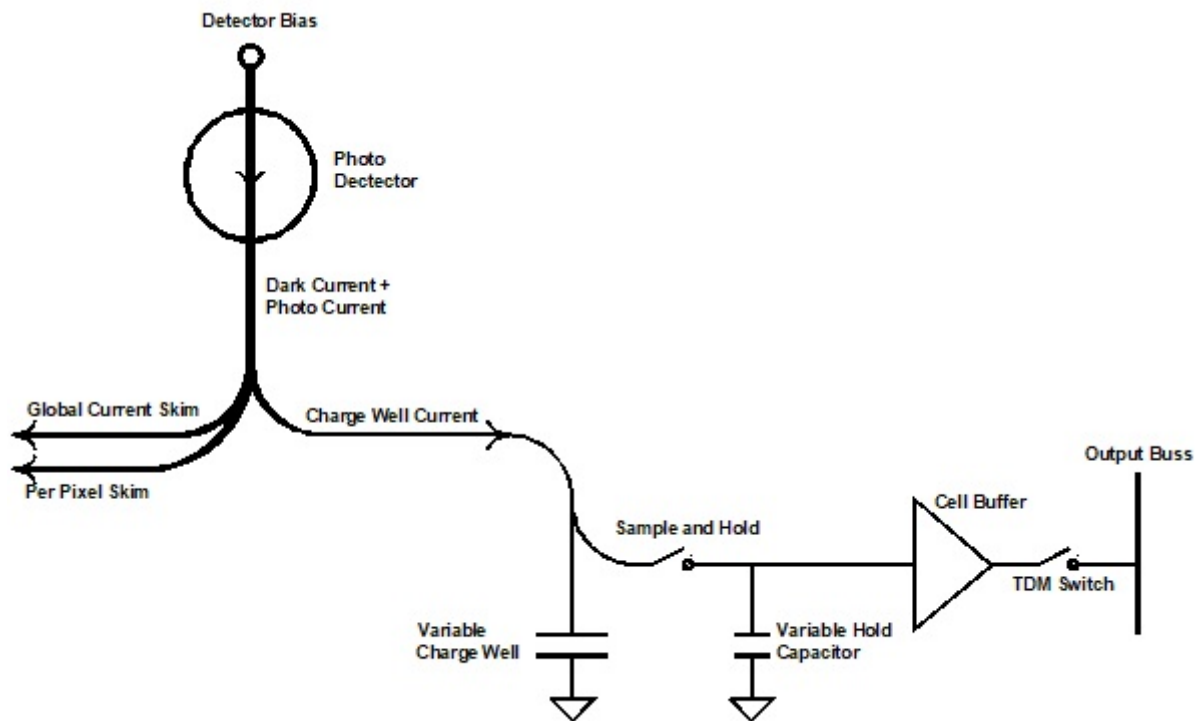


Figure 3: Multiplexer channel diagram

factor of up to 256 to 1. The remaining current coming out of the BDI stage, after skimming, is accumulated by a variable depth charge well. The charge well circuit features globally selectable size from 20 million to 400 million electrons. The accumulated charge is sampled via a charge sharing sample and hold capacitor. This sampled value is then read out through a cell buffer that connects to the internal output buss via a time division multiplexing (TDM) readout switch. The internal output buss is buffered and driven off the die by an internal output opamp.

Accumulation of charge onto the charge well is controlled by an integration clock input (INT_CLK - pad 3). See Figure 4. The charge well detail shown in Figure 4 shows how the INT_CLK input controls a reset switch that eliminates the stored charge on the well and precharges the well to the voltage present on RST_N_BIAS (Pad 35). This eliminates the memory of the previous integration cycles. The RST_N_BIAS voltage is internally generated or may be externally supplied. Figure 4 also shows the selections available for the charge well size. Detailed integration timing diagrams are provided in a later section.

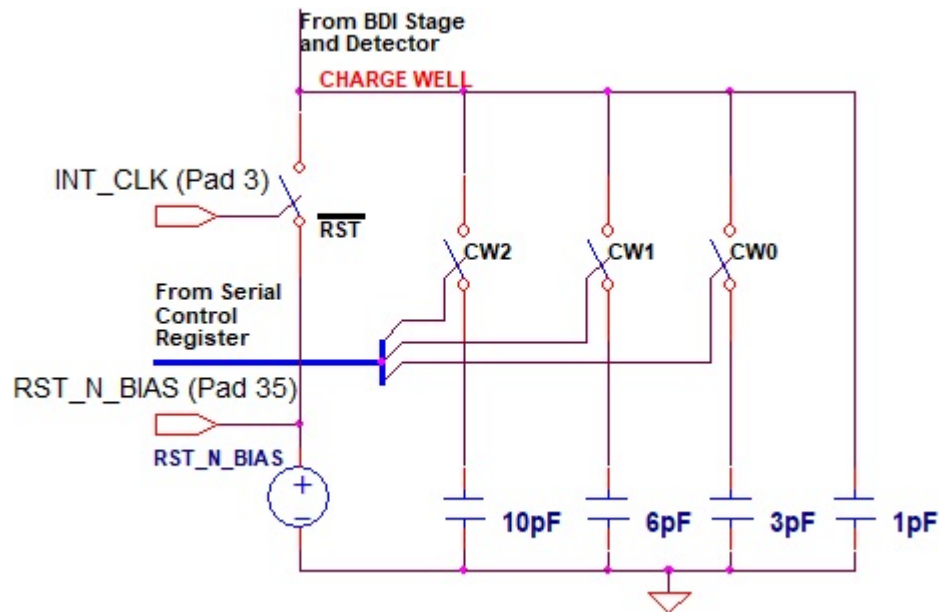


Figure 4: Charge Well Detail

Readout of the array is accomplished by a bidirectional, dual edge, serial shift register, with user assigned start and stop points that provide a limited form of windowing. (See Figure 5). Timing diagrams and a more detailed discussion of readout and windowing are provided later in this document.

Control of the charge well size, hold capacitor size, per pixel skimmer, and other features is provided via a serial interface that consists of two serial register - a control register, 24 bits in length, and a skimmer register that is 8 bits in length. (See figure 6). The two serial registers share the same serial data and clock lines, but have their own data latch clock input. For timing requirements see the later section on serial control operation.

AM256: 256 Channel Linear Multiplexer for Photoconductive Detectors

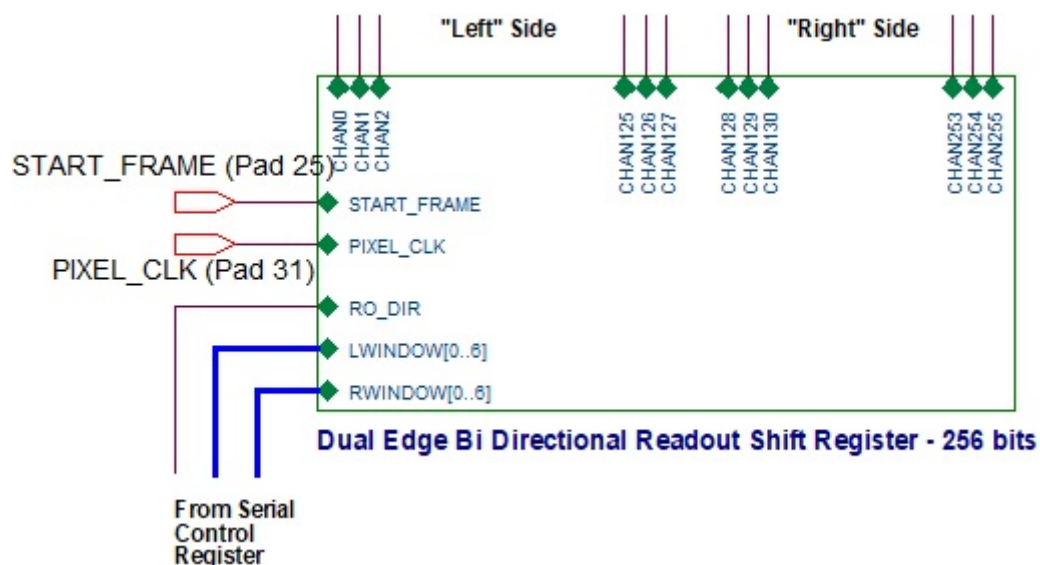


Figure 5: Readout Shift Register

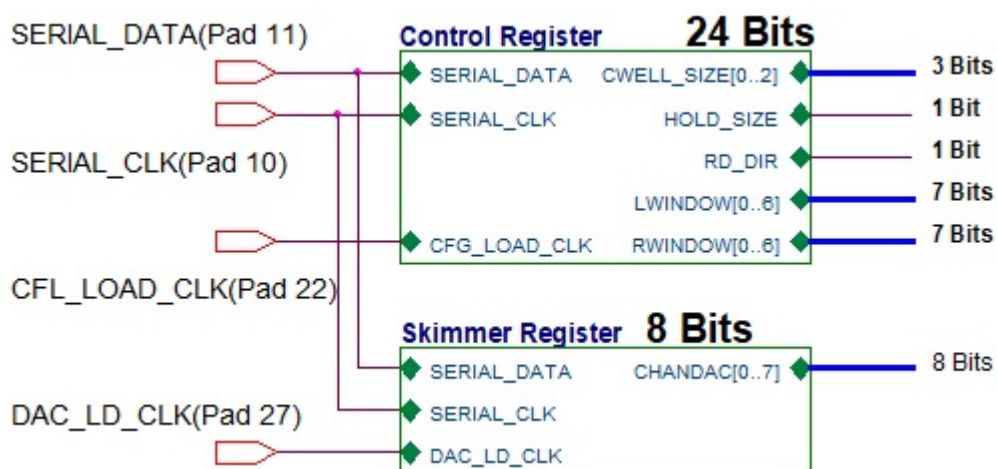


Figure 6: Serial Control Registers

Operation of the dark skimming circuits require a set of low noise voltage inputs, see figure 7. The global current subtraction is controlled via the GLOBAL SKIM (Pad 41) input. The per pixel skim circuit requires two voltages to set the end points on the internal resistive DACs inside each channel. It is important to note that with 256 resistive DACs in parallel on the die, the input impedance of DAC VH and VL is low, approx 1.2Kohms. The driving source needs to be low impedance, and well filtered. Each of these reference inputs have a useful range of approx 0.6VDC to VDD. Both DAC VH and VL should be adjustable, preferably under computer control. A twelve to sixteen bit control for these voltages should be sufficient. For best noise performance the global skim current subtractor should be turned off. This is because the FET used for global skimming is much smaller than the per pixel FET and therefore, injects more noise. To turn off the global skim FET, either drive or connect GLOBAL SKIM to VSS (ground). More detailed information on the current subtraction circuits is contained later in this document.

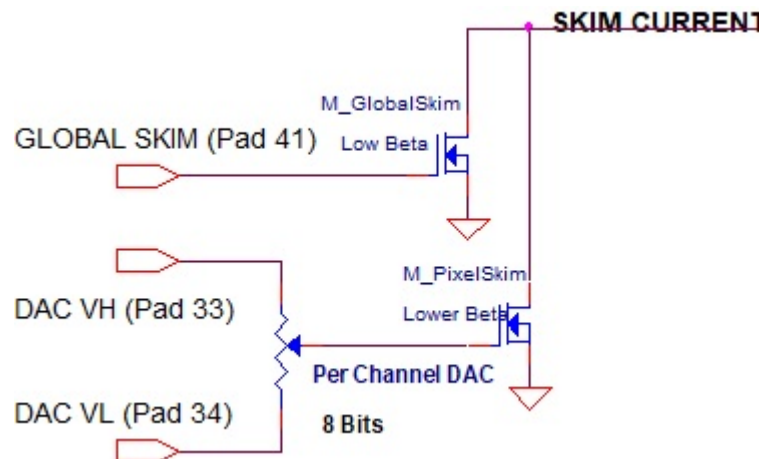


Figure 7: Current Skimming Voltage Inputs

DETECTOR BIAS is another required reference voltage into the array. This is the most critical input to get quiet. Noise on detector bias directly creates a noise current in the detector that experiences the same preamp gain as photo current. The value of the detector bias is dependent on many operational and detector parameters. In general the higher the applied detector bias, the more signal current generated, but also more noise current. Once above a certain amount of bias, the signal to noise ratio of the detector remains constant with increasing bias. Increasing bias can not increase the inherent signal to noise ratio of the detector, but it can raise the noise of the detector above the system

AM256: 256 Channel Linear Multiplexer for Photoconductive Detectors

noise, and this can be useful. Of course if the dark current from the applied bias exceeds the ability of the multiplexer to subtract it, then the system will not work at all. For computations of actual applied detector bias, it can be assumed that the side of the detector attached to the multiplexer is at a voltage approx 1 volt below the value of VDD. In the companion board that interfaces to the array package, absolute bias voltages of up to 12 VDC are possible, this equates to an actual detector bias of approx 8 VDC.

The output of the array is on pad 44 of the die. This output is from an on die CMOS opamp. It is designed to drive into as small a capacitive load as possible, it is recommended that a high impedance, low capacitance, low noise buffer amp be placed as close to the array as possible (see figure 8). Note: the output of the multiplexer can NOT drive a low value resistive load, such as in the Analog Devices Pulsar family of A/D converters. A unity gain configuration is recommended over an inverting configuration due to the reduced noise gain for the unit gain topology. If the VCC and VEE of the buffer are the same supply as the multiplexer, then a modern RRIO opamp should be used.

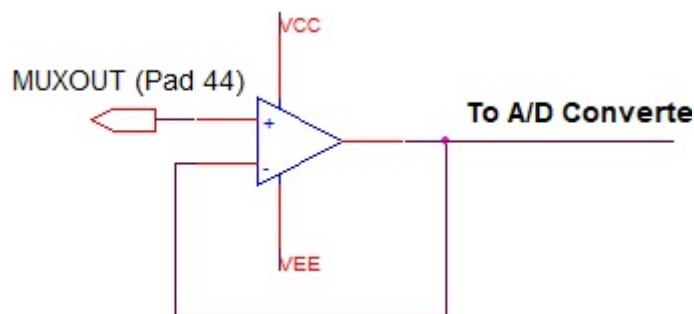


Figure 8: Output Buffer (typical)

Power to the array is supplied via the VDD (Pads 29, 43, 50) and VSS (Pads 2, 30, 42, 51) pads. The IC is manufactured in a 0.5 micron CMOS process which is rated for 5VDC operation. Adequate supply bypassing is essential for best performance, see Figure 9 for recommended supply decoupling.

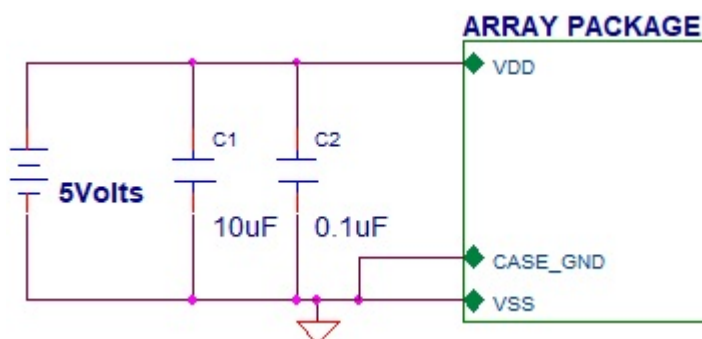


Figure 9: Power Supply Detail

Detailed Description

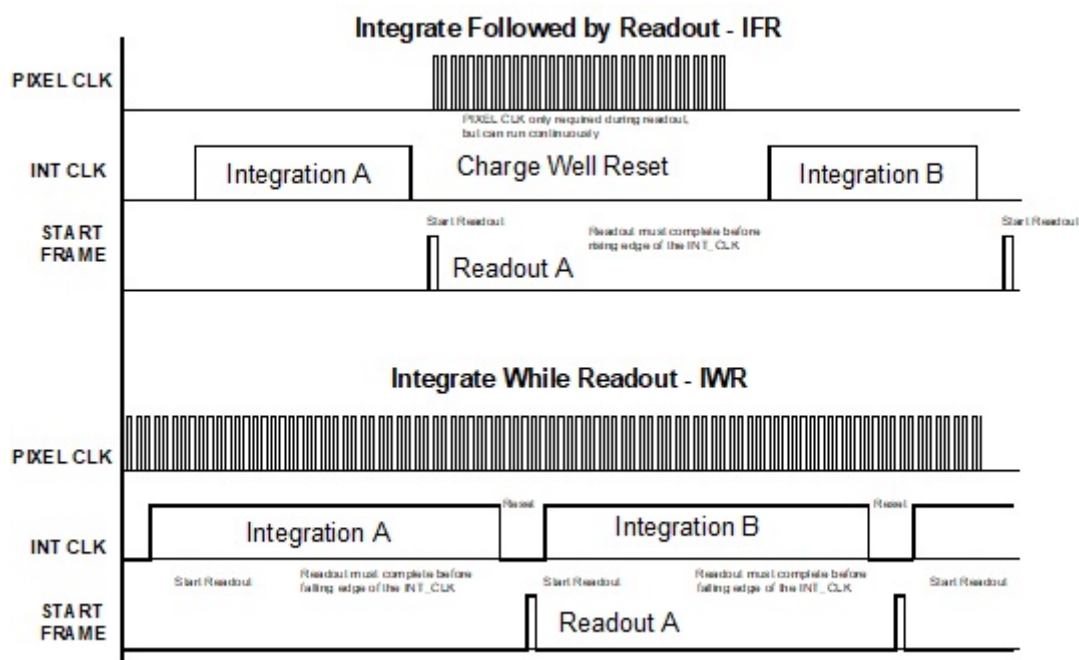
Shown in Figure 10 is a list of the die pads that are typically required to come out to any external control system. In addition there would certainly be thermoelectric cooler signals needed for an actual array package.

Die Pad#	Name	Type	Description
29, 43, 50	VDD	Power	Typically +5 VDC
2, 30, 42, 51	VSS (GND)	Power	Ground
44	MUXOUT	Analog OUT	ROIC analog output, used for single die setups or for multi die / multi output setups
33	DAC VH	Analog IN	Analog voltage required for per pixel skimmer circuit.
34	DAC VL	Analog IN	Analog voltage required for per pixel dark current skim circuit.
41	GLOBAL SKIM	Analog IN	Analog voltage required for the global dark current skim circuit
35	RST N BIAS	Analog IN	Charge well reset bias level. Generated on die, or supplied externally through this pad
3	INT_CLK	Dig IN	When high, the ROIC will integrate charge on the charge well. During the lowtime, the charge well will be held in reset.
25	START FRAME	Dig IN	Initiates pixel readout. In master mode, pixel channel 1 is output on the second rising edge of Pixel Clock, following the rising edge of Start Frame. In slave mode, pixel channel 1 is output on the first rising edge of pixel clock, following the rising edge of Start Frame
31	PIXEL CLK	Dig IN	Pixels are output on each edge (rising and falling) of pixel clock during readout.
10	SERIAL CLK	Dig IN	Rising edge clocks data present on SERIAL DATA, into the serial control registers.
11	SERIAL DATA	Dig IN	Data on this line is clocked into serial config registers on rising edge of SERIAL CLK.
22	CFG LOAD CLK	Dig IN	Latches serial data into the serial configuration register on it's rising edge
27	DAC LD CLK	Dig IN	Latches serial data into per pixel skimmer.

Figure 10: Table of typically required die pads

Integration Timing:

The relationship between the time for charge integration and the time for readout can be setup in two distinct modes, integrate followed by readout (IFR) and integrate while readout (IWR). (See Figure 11). The choice of which mode to run in depends on the system requirements and capabilities. IFR mode does not constrain the readout speed, it can be read out with a wide range of clock speeds, but there is large amounts of time when the charge well is not integrating input current. IWR give the highest percentage of time for actually integrating charge, but you must finish the readout of the previous integration, before the end of the next integration cycle. This generally places a limit on how short an integration time can be achieved, while requiring the fastest possible readout speed.

**Figure 11:** Basic Integration Timing Modes

IFR Details: Figures 12 and 13 show the details regarding integration and readout timing for the IFR integration mode. Figure 12 shows the requirements for starting integration, which are very simple. The only requirement is that the previous readout cycle must be completed prior to starting the next integration cycle. The other requirement shown is on the frequency of the PIXEL_CLK, which is suggested to be a maximum of 2 MHz, which results in a 4 mega sample pixel output rate, due to the dual edge readout shift register.

Figure 13 shows the end of integration and the beginning of readout, which again, has very few requirements. The most important is that the clocking of the start of frame input be a minimum of 5 μ s after the falling edge of the integration clock. Please note, that although the readout shift register is dual edge clocked, the START_FRAME input is only clocked in on the rising edge of PIXEL_CLK.

With integration followed by read (IFR) operation some timing limits are clearly evident. Assuming a full 256 element array is readout at the maximum possible rate (4 msmtps), the time required for readout is $250\text{ns} \times 256 = 64\mu\text{s}$. Adding 5 μ s to this, gives a

AM256: 256 Channel Linear Multiplexer for Photoconductive Detectors

minimum time between integration cycles of 69 μ s. This is time that the charge well will be in reset and not integrating charge. For many applications this is not a problem, and the benefits of IFR operation are - timing simplicity, reduced speed requirements or

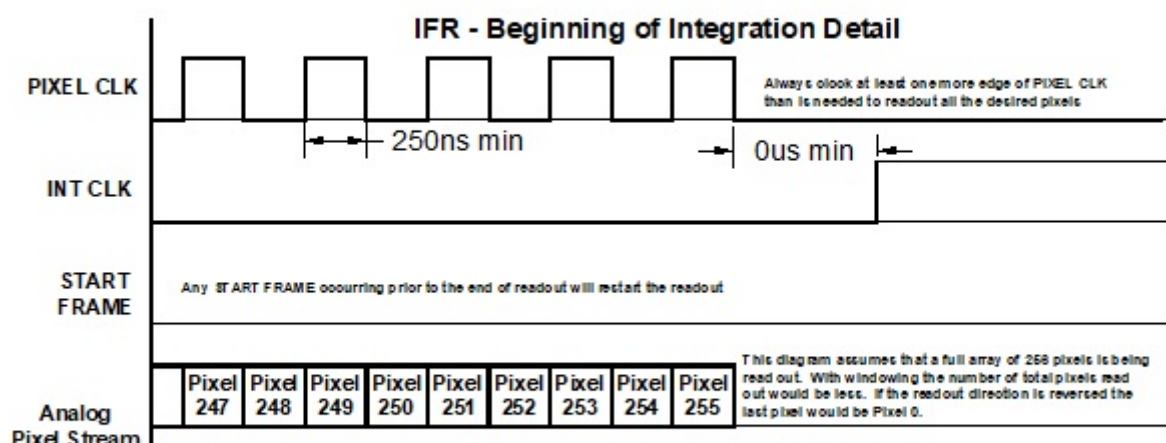


Figure 12: IFR Mode - Beginning of Integration

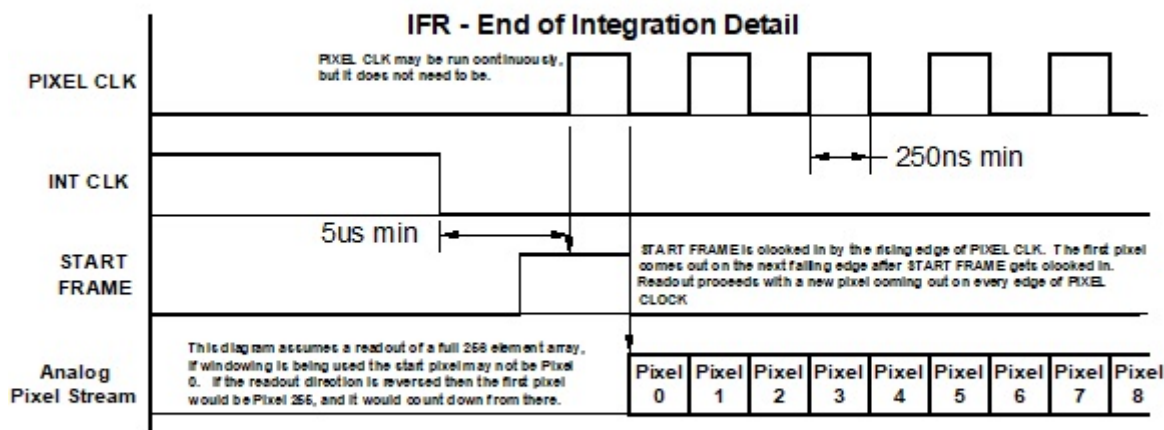


Figure 13: IFR Mode - End of Integration and Start of Readout

IWR Details: Figure 14 shows the timing details for the IWR mode of operation. The beginning of readout is similar to IFR operation, START_FRAME must be clocked in at least 5 μ s after the falling edge of INT_CLK. The major difference between IFR and IWR is that a new integration cycle can begin as little as 5 μ s after the end of the previous integration. To make use of this, the readout of the array must be complete before the next falling edge of INT_CLK. Once again this sets a timing limit, assuming a full array read out at maximum rate (250ns per pixel) the minimum allowable integration time would be 64 μ s. In general longer integration times are not an issue with a mux with adjustable charge well size, but for some high flux applications this timing constraint must be kept in mind.

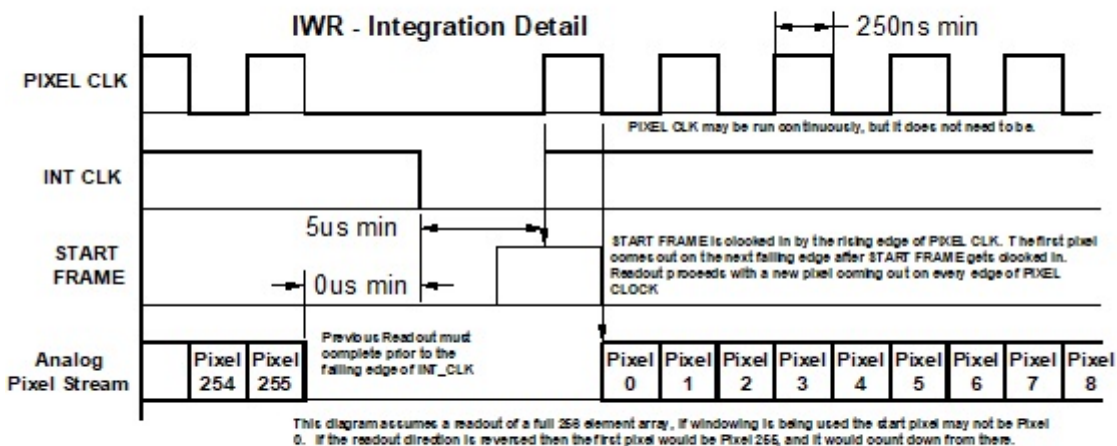


Figure 14: IWR Integration details

Serial Control Registers:

Control Register: Figure 6 shows a 24 bit serial control register, which consists of a 24 bit, rising edge clocked, shift register, followed by a 24 bit wide, rising edge D latch. The control register allows for configuration of the multiplexer operation - including charge well size, hold capacitor size, readout direction, and readout windowing limits. There are several other features controlled by this register, but they must be set to a particular value for the multiplexer to operate properly. Which means, and this is very important, **to operate the multiplexer the serial control register MUST be properly loaded every time the array is powered.** The onboard D latch is volatile and all information will be lost on power down. On power up the values in the serial register and D latch will be random, this will not damage the mux in any way, but it will not operate as expected until the serial control register has been loaded.

Figures 15 and 16 , below, lists the bits contained in the control register, their function and provide timing information on loading the register..

Bit#	Value	Name	Description
23	0	cfgbit23	First bit clocked in
22	x	RWINDOW6	Right side stop/start address
21	x	RWINDOW5	
20	x	RWINDOW4	
19	x	RWINDOW3	
18	x	RWINDOW2	
17	x	RWINDOW1	
16	x	RWINDOW0	
15	x	LWINDOW6	Left side start/stop address
14	x	LWINDOW5	
13	x	LWINDOW4	
12	x	LWINDOW3	
11	x	LWINDOW2	
10	x	LWINDOW1	
9	x	LWINDOW0	
8	0	cfgbit8	
7	x	RO_DIR	Read out direction
6	0	cfgbit6	
5	x	CWELL_SIZE0	Charge well 3pF
4	x	CWELL_SIZE1	Charge well 6pF
3	x	CWELL_SIZE2	Charge well 10pF
2	x	HOLD_CAP_SEL	Hold cap size
1	1	cfgbit1	
0	0	cfgbit0	Last bit clocked in

Figure 15: Serial Control Register Bits

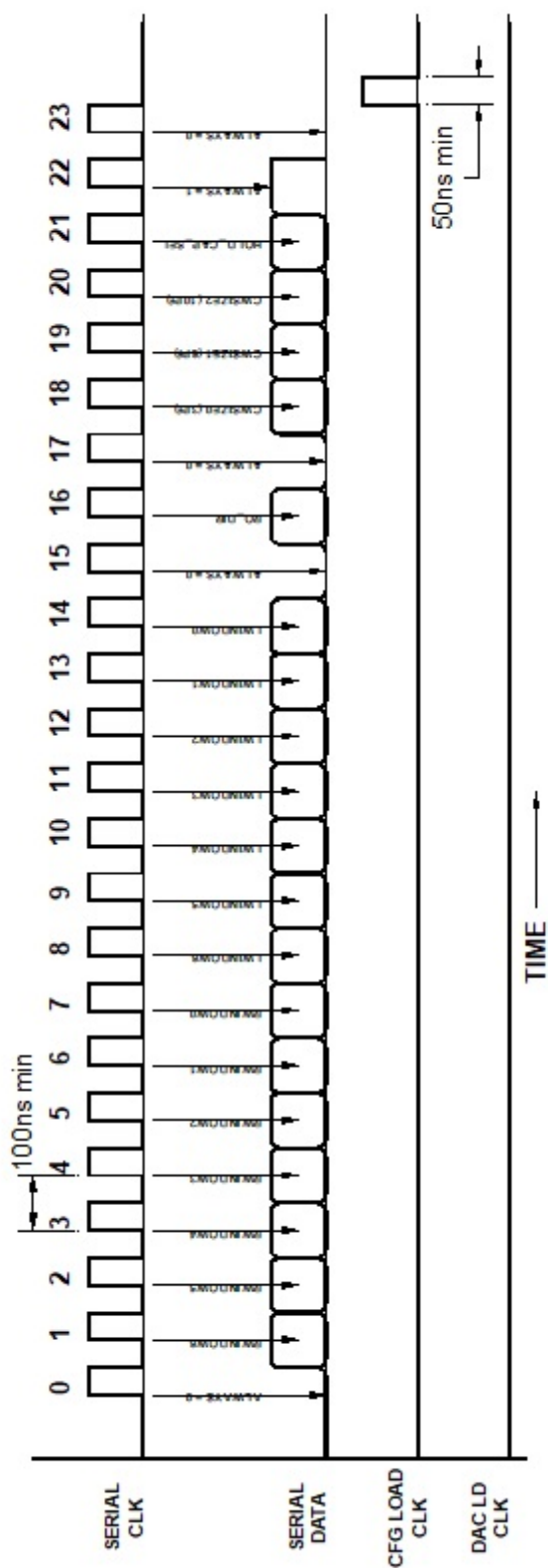


Figure 16: Serial Control Register Timing Diagram

Bit Functions - Charge Well Size:

Serial configuration register bits, 3, 4, and 5 control the size of the charge well. See figures 4, 15, and 16. When the charge well bit is a logic low, its corresponding well capacitor is off. Therefore, when all three charge well bits are a logic low, the charge well is at the minimum value possible, which is 1pF. When all three bits are a logic high, the charge well is at the maximum value of 20pF.

Bit Functions - Hold Cap Size

Serial configuration bit 2 controls the size of the hold capacitor. See figures 15, 16, and 17. The larger the hold capacitor, the less noise it makes (kTC noise) and the less it will be effected by leakage currents. The down side to a larger hold capacitor is the gain loss due to the charge sharing sample structure. Each integration cycle the charge well and hold capacitor are reset to a known value, then, after integration is complete, the charge on the well is shared with the hold capacitor. This results in a gain loss related to the ratio of the charge well size to hold capacitor size. For these reasons we recommend using the small hold cap size with charge wells less than 7pF and the large hold cap size with charge wells 7pF and larger. The large hold capacitor is engage by loading a logic high into bit 2 of the serial configuration register.

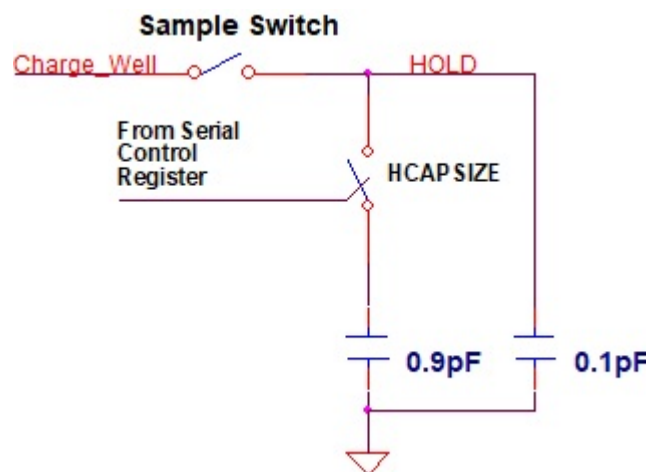


Figure 17: Hold Cap Size Detail

Bit Functions - Readout Direction Control:

Bit 7 of the serial configuration register controls the direction of the readout. When bit 7 is a logic low, the array will readout in a left to right direction. The reference for direction is to orient the die with the detector bond pads at the top. When the direction control bit is loaded with a logic high the readout goes from right to left.

Note: When using windowing, the window values do not swap sides with direction control, when reading out left to right the left window value will define the start point and the right window value the end point. However, when reading out right to left it is the right window value that defines the readout start point and the left window value sets the end point - see the windowing bits description below for more information.

Bit Functions - Readout Windowing Control:

Serial configuration bits 9 through 22 control the readout windowing of the array. See figure 18. With the windowing control the start and stop point of the readout can be selected, with one condition. For a left to right readout direction, the start point must be within the leftmost 128 pixels, and the stop point must be within the right most 128 pixels, that is why the windowing is referred to as "limited windowing". (see section of Readout Direction Control for array orientation definition). The two start/stop pixels selected via the serial configuration, will act as either the start pixel or stop pixel for the array readout, depending on the state of the readout direction control bit. Assuming the readout direction control bit is low, the array readout will start at the pixel defined by the value of the Left Window control value - set by bits 9 through 15 of the serial configuration register. Bit 9 represents the Lsb of the left window value. A left window value of zero will define the first pixel as the left most pixel of the array. The readout will progress from the start point, in a left to right fashion, across the array until it hits the stop point defined by bits 16 to 22 of the serial configuration register, which is the right window value (bit 16 being the lsb of the right window value). The right window value is interpreted differently than the left window value, instead of defining the actual pixel number, it defines the number of pixels from the right most edge of the array. In this way, if at start up, all the windowing bits, 9 through 22 are loaded with logic lows (0), then the array readout will be the entire array. The smallest readout that can be defined is if the left and right window values are both 7Fhex, which would be a readout of only 2 pixels, exactly in the center of the array (pixels 127 and 128 to be precise)

Skimmer Register: Figure 7 shows that each pixel has a current skim FET that is controlled via an 8 bit DAC. The 8 Bit DAC is a monotonic resistive DAC that applies a voltage to the gate of the skim FET that is between the two input DAC voltages, DAC VH and DAC VL. Loading the pixel skim register with all zeros will apply the DAC VL to the gate of the skim FET, which will result in the smallest amount of skim current being removed from the input current. Conversely, loading all ones will apply the DAC VH to the gate, resulting in the maximum amount of current being removed from the input current. The amount of current that can be skimmed depends on the levels of DAC VH and DAC VL, Figure 20 shows a Pspice simulation of the skim current versus gate voltage for both the global skim FET and the pixel skim FET.

To load each of the pixel skim DACs with a unique value the skim DAC register loading utilizes the readout register to point to the DAC register that should be updated. See figure 18. The process of updating the DAC registers involves significant clocking of lines inside each pixel, this can impact the noise for any readout that is in process. For that reason it is recommended that a separate readout cycle be devoted to updating the DAC registers. Once updated they will retain their value until the power is removed from the array.

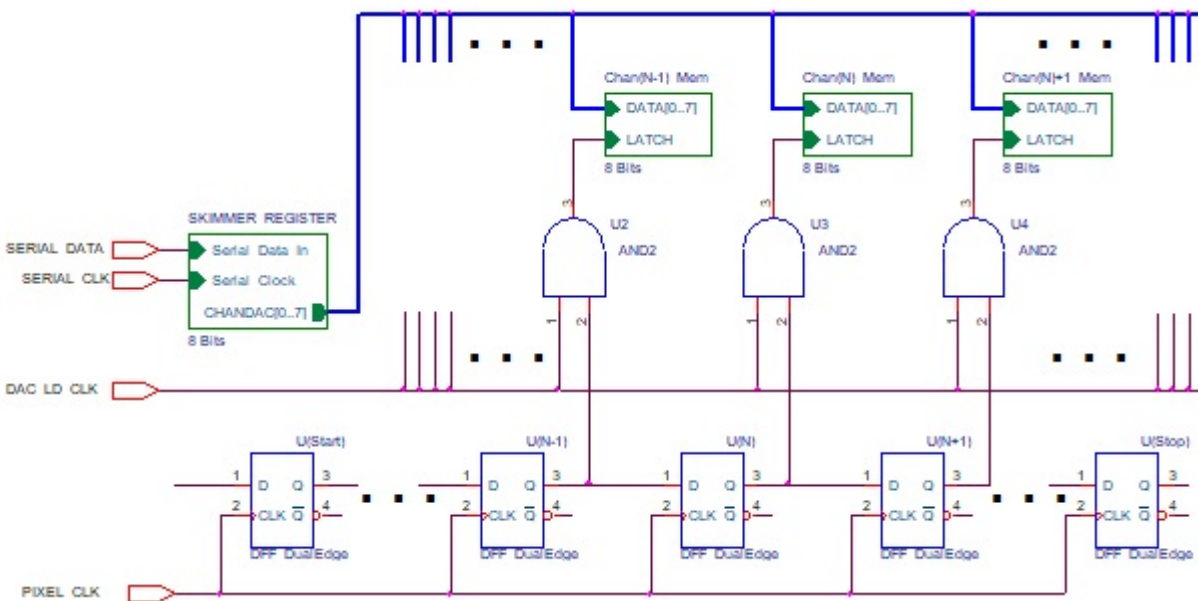


Figure 18: Pixel Skim Serial Data Loading Structure Diagram

Figure 18 shows that the DAC register being updated by the rising edge of DAC LD CLK is the current pixel being read out, this is regardless of any windowing or direction control settings - the pixel being updated will always be the pixel being read out. For this reason it is very important to keep track of any windowing or direction control settings while operating the array, so that the pixel digital DAC settings do not become mixed up by the external controller and its software.

AM256: 256 Channel Linear Multiplexer for Photoconductive Detectors

Figure 19 shows an example timing diagram for loading of three successive pixels - pixel N-1, pixel N, and pixel N+1. The readout is clocked from one pixel to the other by any edge of pixel clock (remember that the readout shift register is comprised of dual edge flip flops). The DAC skimmer register requires 8 clocks to load a new value, with the LSb of the DAC being loaded first and the MSb last. The DAC skimmer register feeds an 8 bit buss that goes across the entire array, connected to this buss each pixel has an 8 bit D latch. To latch the 8 bit DAC data into the pixel data latch, the DAC LD CLK must have a rising edge while the pixel readout shift register is pointing to the particular pixel. The figure below shows the DAC LD CLK being a pulse at the end of the pixel time, the minimum width of this pulse is 50ns. The DAC LD CLK does not need to be a narrow pulse, it could easily be a clock running at 2 x the pixel clock rate, and shifted so that it's rising edge occurs before the edge of PIXEL CLK. The minimum timing requirements are shown in the figure below.

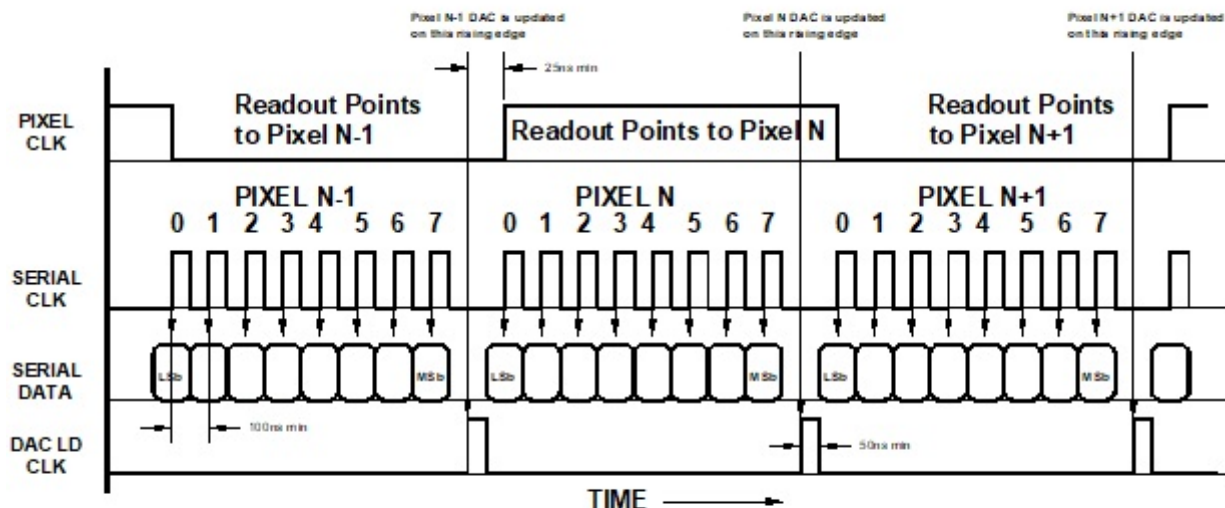


Figure 19: DAC Skimmer Register load timing diagram

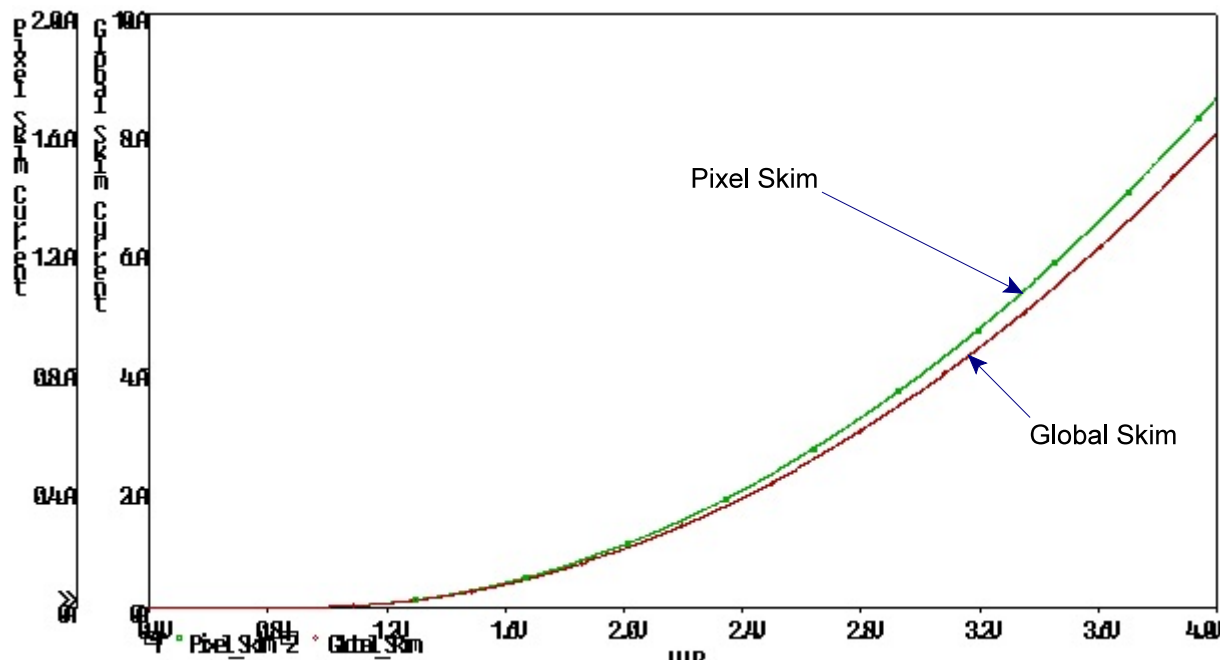


Figure 20: Pixel and Global Skim FET Current simulation

Analog Bias Voltages:

Global Skim: Each pixel of the array contains a transistor that is used to subtract dark current from the incoming detector current, see figures 3 and 7. This “global” current skim is based on a specially designed low beta FET whose gate node is connected to the external analog input voltage, GLOBAL SKIM (package pin 12). Figure 20 shows the amount of current skim versus the global skim input voltage. The drive requirements for global skim are based on the high impedance nature of FET gates. Because the only parts attached to global skim are the FETs of the skim transistors, there is no DC current into or out of the GLOBAL SKIM pin. This type of input can be very well filtered by using a modest impedance source with a large filter capacitor, the only down side is that the time constant used to filter the signal will slow the rate at which the voltage can be varied. For example, a 10 Kohm source impedance and a 1uF capacitor would yield a 10ms time constant, any change to the GLOBAL SKIM value would require on the order of 100ms to stabilize before accurate measurements could be made. For those systems that require the use of global skim, a trade off between noise and speed of operation will need to be examined. For a great many applications the global skim is not actually required, the detector dark

current is low enough that all of the skimming can be done by the per pixel skimmer circuit. This has a few advantages, first - no GLOBAL SKIM input required. Connecting GLOBAL SKIM to ground will effectively turn off all the global skim FETs. Second - there is reduced noise when not using global skim, this is because the global skim FET inherently has higher noise than the per pixel skim FET.

DAC VH and DAC VL: Driving the 8 bit resistive DAC inside each pixel channel, are the DAC VH and DAC VL voltages. These voltages set the upper and lower bounds of the per pixel current skimming (see figures 3, 7, and 20) circuit. Driving 256 resistive DACs in parallel means that the load impedance between DAC VH and DAC VL is low, on the order of 1200 ohms. This requires a drive with low output impedance and reasonable current capability. The recommended drive is a precision low noise opamp that can drive reasonable capacitance loads, then add a series 10 or 20 ohm source resistance and a 1uF to 10uF filter cap to get the noise bandwidth down.

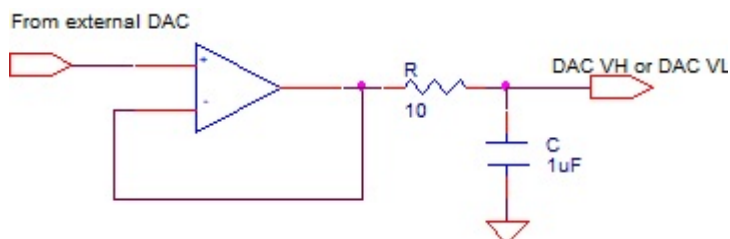


Figure 21: Example drive circuit for DAC VH/VL

The generation of the DAC VH and VL values is also important for array performance. The quality of the uniformity correction utilizing the pixel DACs is dependent on the selection of an appropriate VH and VL level. It is highly recommended that any calibration routine be based on a multi step closed loop process that utilizes digitally variable sources for the DAC VH and VL levels. In practice a 12 bit converter is generally sufficient for the DAC voltage generation.

Detector Bias: As mentioned earlier, the detector bias is the most important bias to get quiet. Any noise on detector bias will result in a noise current in the detector that will experience the same gain as the signal current. In addition to the bias being quiet, it needs to be modestly stiff. A single detector may be 40 megohms in impedance when cold, but put 256 of them in parallel and you have 156Kohms as a load. If you run square detectors, that number can drop by a factor of 10 or more. This sets up a difficult trade off, on the one hand it would be nice to have a large series resistance in the detector bias line to achieve a very low corner frequency for the filter, making the bias very quiet. On the other hand, having a low output impedance eliminates pixel

AM256: 256 Channel Linear Multiplexer for Photoconductive Detectors

output variation due to changes in detector impedance. The circuit shown in figure 22 strikes a compromise (one of many possible) between the two conditions. The choice of opamp is not trivial, it should have good temperature performance, low noise, low bandwidth, and the ability to drive lightly decoupled, large capacitive loads without oscillating. If your detector bias circuit is not up to the task, it will show up as noise that effects all pixels the same. If you view the pixel output in real time, it will look like the entire array is bouncing up and down, this type of common mode noise is typically from detector bias. It can be confirmed by doubling or halving the integration time, if the amplitude of the noise varies $1=1$ with integration time, then it is coming in the front end, from the pixels themselves. There are only two likely suspects at that point, either detector bias, or the temperature of the detectors is varying. Usually temperature variation is at a much lower frequency than the detector bias noise.

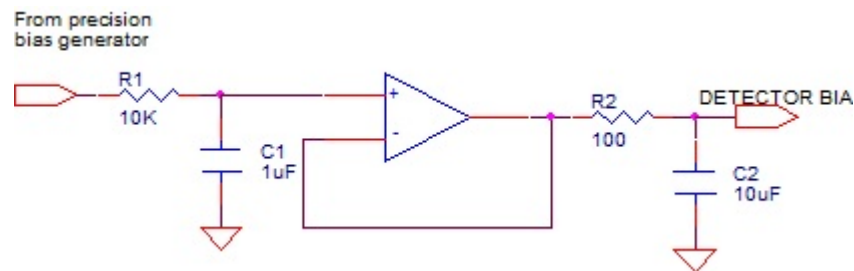


Figure 22: Detector Bias drive circuit