Product Document

Published by
ams OSRAM Group

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

Application note for CMV sensors

Sensor Calibration

Change record

Issue	Date	Modification
v1	28/01/2013	Origination
V2	07/04/2014	Added Vramp register overview

Table of Contents

1 SENSOR OVERVIEW

Below you can see a block diagram of how the CMV sensors are composed. All columns have their own pre-amplifier and ADC. Per channel there is one LVDS driver. The logic and decoders are controlled via the sequencer which can be programmed with the SPI registers.

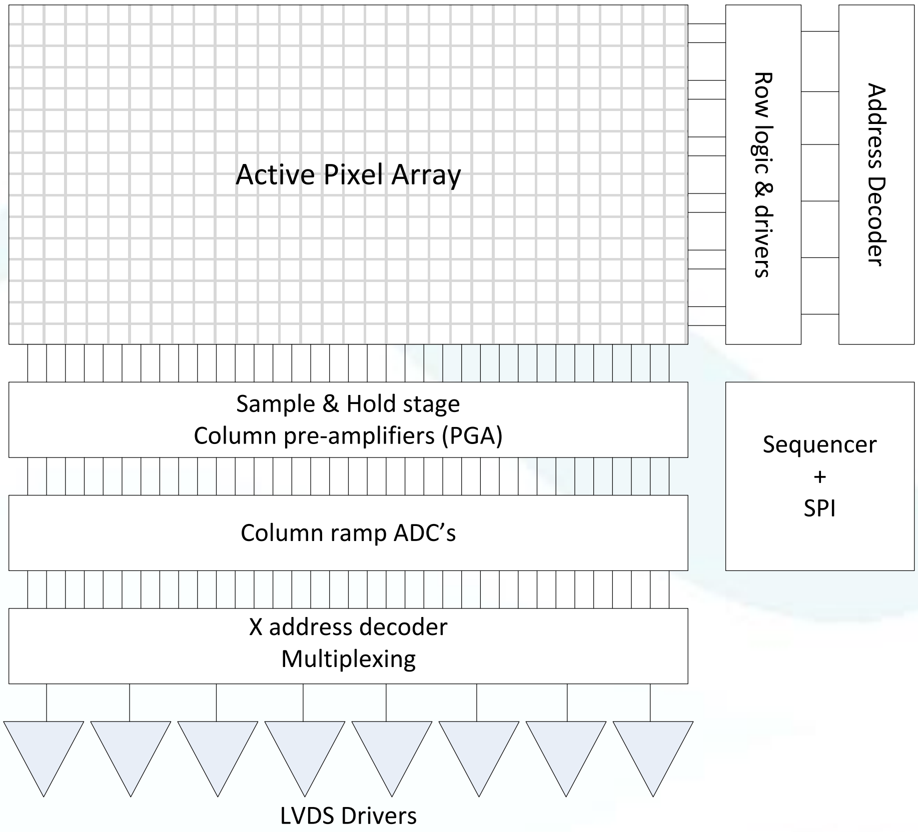


FIGURE 1: CMV BLOACK DIAGRAM

Below you can see the path of the incoming light to the LVDS data going out of the sensor.

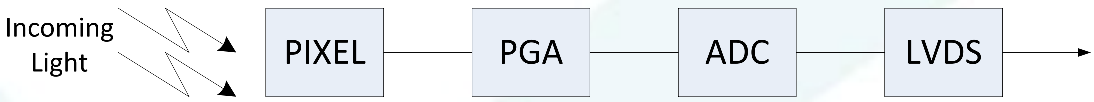


FIGURE 2: LIGHT INPUT TO DATA OUTPUT PATH

1.1 PIXEL DETAIL

In Figure 3 you can see the inner workings of the 8T pixel used in the CMV sensors. During FOT the reset switch is opened and the reset signal is sampled on C2. Then the diode charge is transferred and the signal level is sampled on C1. During the read out time, the reset level is read out. Then the C2 charge is divided over C1 and C2 and the signal level is read out. Because of this CDS (Correlated Double Sampling) the pixel FPN and noise is very low.

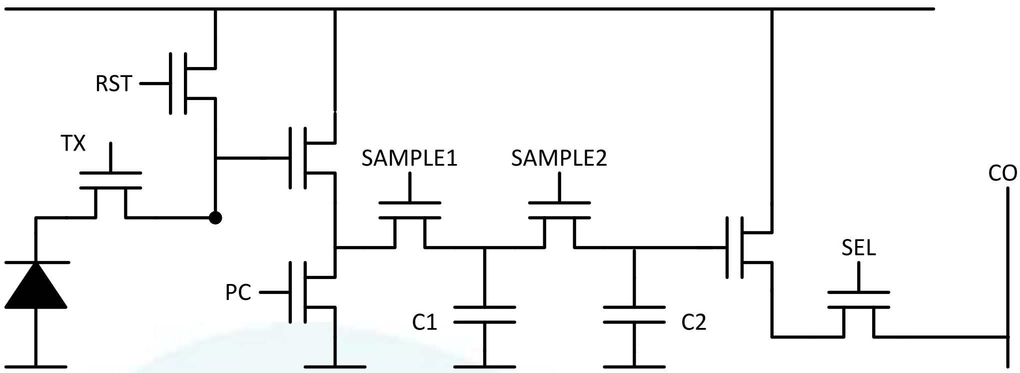


FIGURE 3: 8T PIXEL

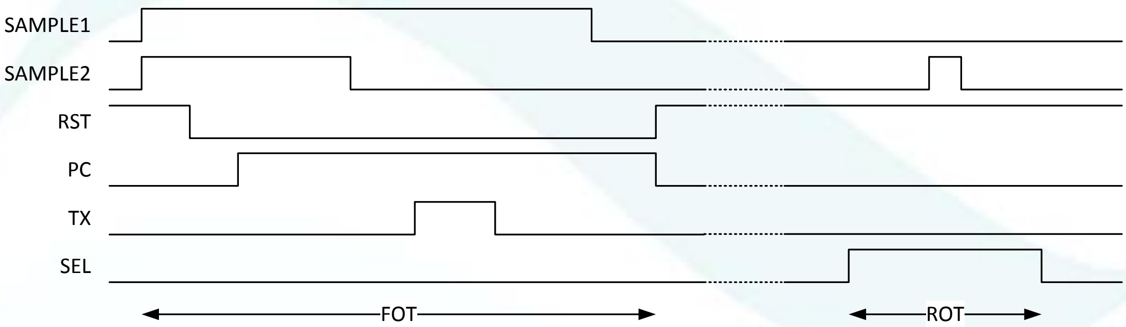


FIGURE 4: 8T PIXEL TIMING

1.2 PGA & ADC STAGES

The pixel voltage on the column bus will pass through the analog amplifier (PGA – Programmable Gain Amplifier). The output of the PGA is then used as ADC input voltage. Therefor it is important to calibrate the sensor so the useable PGA output voltage swing will match with the ADC input range.

The pixel voltage response is non-linear close to saturation. Depending on how much non-linearity is tolerated in the camera system, only a certain percentage of the pixel output range can be used to have a linear response. This linear region should then be matched to the ADC input range.

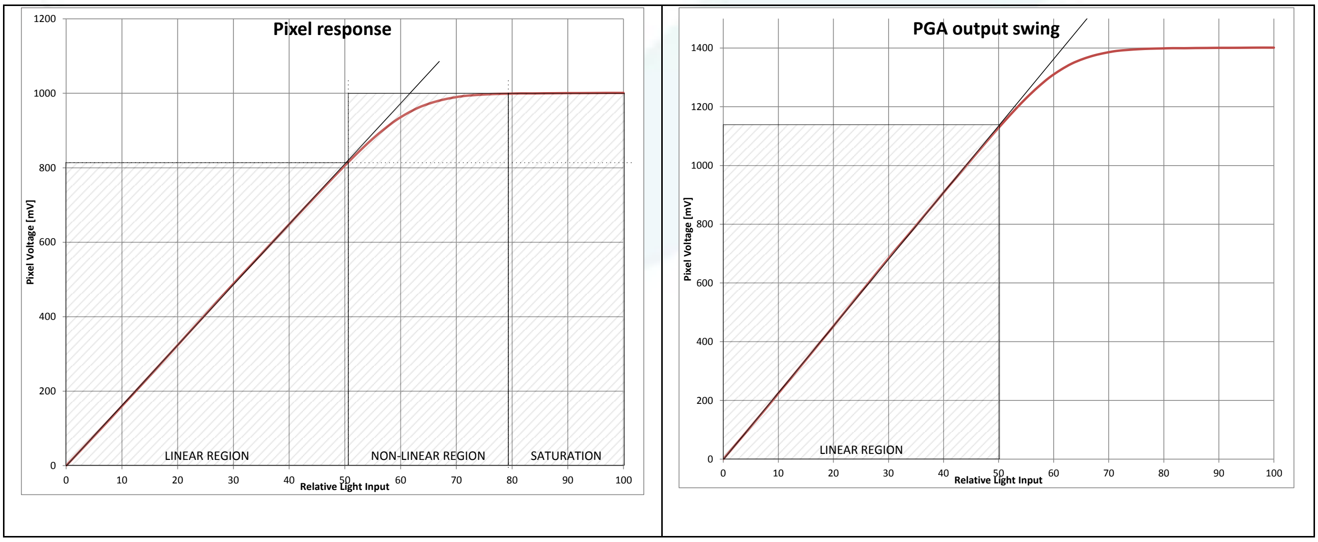
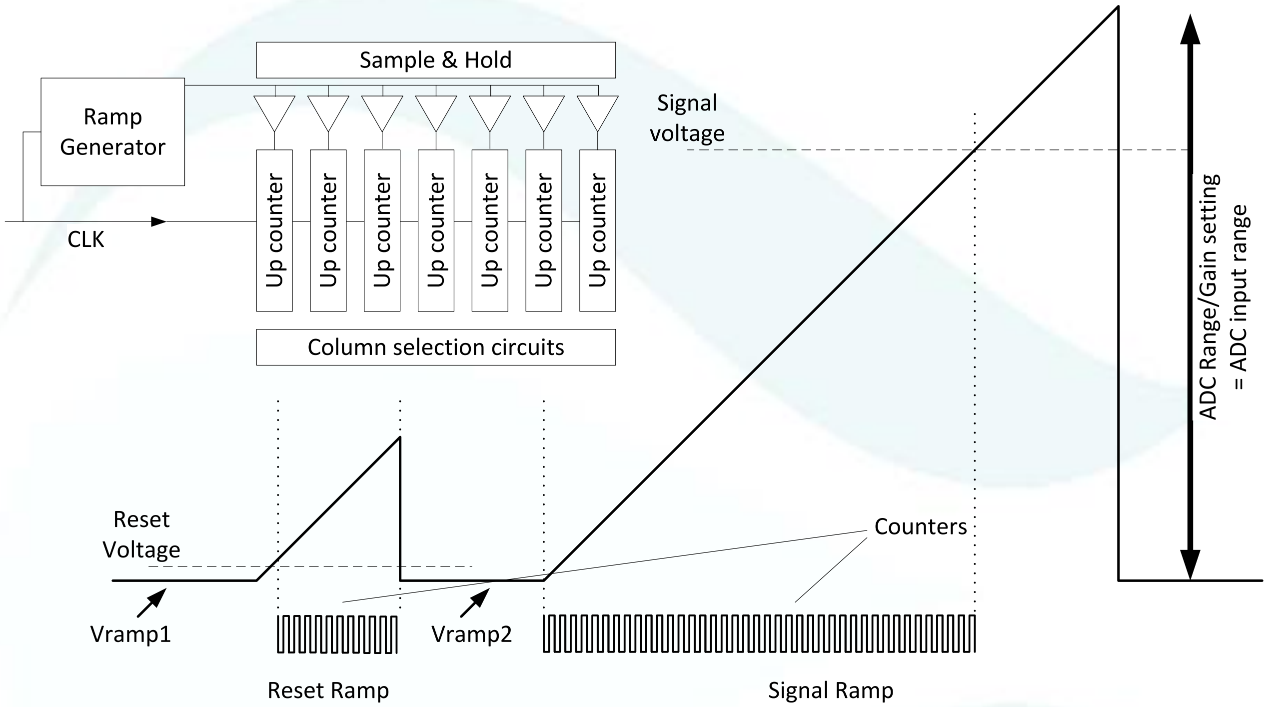


FIGURE 5: PIXEL AND PGA OUTPUT VOLTAGE SWING

In the example above we see that the pixel voltage swing is 1000mV (from dark to 100% relative light). With a PGA setting of x1.4, this becomes 1400mV. Therefor the ADC input range (=ADC gain setting), should be matched to this swing, so that a pixel voltage of 0V will correspond to 0DN and 810mV will be 1023DN.

The ADC's used in CMV sensors are ramp ADC's. The reset and signal voltage are both converted on their own ramp. The ramp generator increases its voltage until it matches the reset/signal voltage. The counters will count during that period, after which the counted value for the reset and the signal will be subtracted, giving the actual pixel swing in DN. This value is then transferred to the LVDS drivers.



As you can see in the diagram above, Vramp1/2 and ADC gain/range set the ADC input range. Because of small differences between sensors, it is recommended that these settings are calibrated to get the same response for every sensor.

Vramp1/2 sets the start voltages of the reset and signal ramp. When they are set too critical it can be that some column's PGA lowest voltage (at reset/dark) is just below the ADC start voltage. This will result in column FPN. If a lot of column FPN is visible in a fully dark image, please tweak (increase or decrease) both Vramp settings until the column FPN is gone. Normally Vramp1 and Vramp2 should equal. When they are not equal, this will result in an ADC offset (which can be corrected with the offset register).

ADC gain/range sets the height of the ADC ramps. When different sensors have a different response, this setting should be used to correct them to the same response. A higher register value will result in a lower ramp height. When setting the ramp height too high, the non-linear part of the pixel will now also be converted. Setting it too high will result in saturation on pixel level rather than ADC level, causing the image to saturate below 1023 (10bit). In almost every application this is not desired. When lowering the clock speed, the ADC setting should be increased to compensate.

Device	Vramp1	Vramp2
CMV300	109[6:0]	110[6:0]
CMV2000/4000 v2/v3	98[6:0]	99[6:0]
CMV12000 v1/v2	109[6:0]	109[13:7]
CMV20000	117[6:0]	116[6:0]

1.3 READ OUT

The CMV sensors use a pipelined architecture. This means that the exposure of the next frame can start during the read out of the current frame. The read out time consists of 3 pipelines: Pixel access & sampling, ADC conversion, LVDS read out. This means that when row N is read out to the LVDS drivers, row N-1 is at the ADC stage and row N-2 is at the access and sampling stage. So there is a period of 2 line times between the FOT and FVAL on the control channel or Tdig pins. This two line time periods are already in the formula of the FOT in the datasheets.

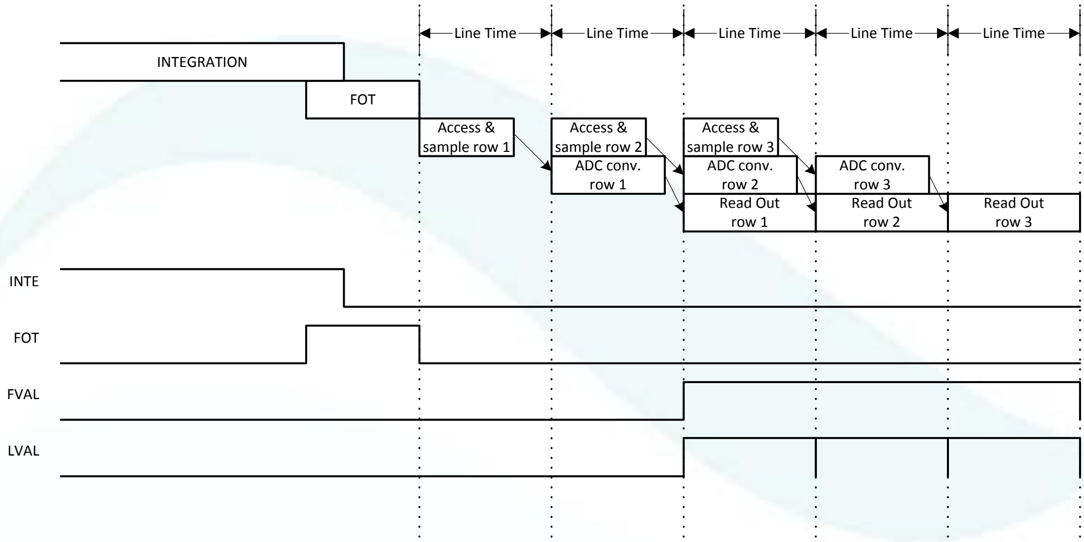


FIGURE 6: READ OUT TIMING

2 OFFSET, FPN AND PRNU

The pixel, PGA and ADC will add FPN and PRNU to the final image. FPN (Fixed Pattern Noise) is considered to be a fixed offset difference between pixels independent on the incoming light. FPN can be measured when taking many images in the dark with a short exposure time and averaging them to get rid of the noise. The FPN is then considered the standard deviation in that averaged image.

PRNU (Photo Response Non Uniformity) on the other hand is the light dependent 'FPN'. Different pixels will have different sensitivity to light, thus having a different response curve. This can be measured by taking many equally lighted grey images (around 50% of the sensor swing) but with rather short exposure time (so dark current has little to no influence) and averaging them down. Subtract the FPN image from this grey one. The result is an image containing the PRNU (standard deviation of that image).

Due to the CDS used in the 8T pixel FPN and PRNU are low.

Offset is set in the ADC and is digital of nature. The offset register will add/subtract the same number of DN to/from every pixel. This is used to set the dark level of the sensor.

In the plots below, you can see the values of 10 pixels. In the first chart the raw pixel data from the averaged dark and grey image is plotted. You can see little detail in this. When we zoom in on the start, we can see that all pixels have an offset (>0) and that all pixels have a different starting value (=FPN). In the next chart, the FPN image is subtracted from the grey one (so all pixels start at (0, 0)). The resulting value spread at 50% of the image swing is the PRNU (caused by different slopes). You can see that pixel 8 is more sensitive to light than pixel 4, so it will reach a higher grey value at the same lighting condition.

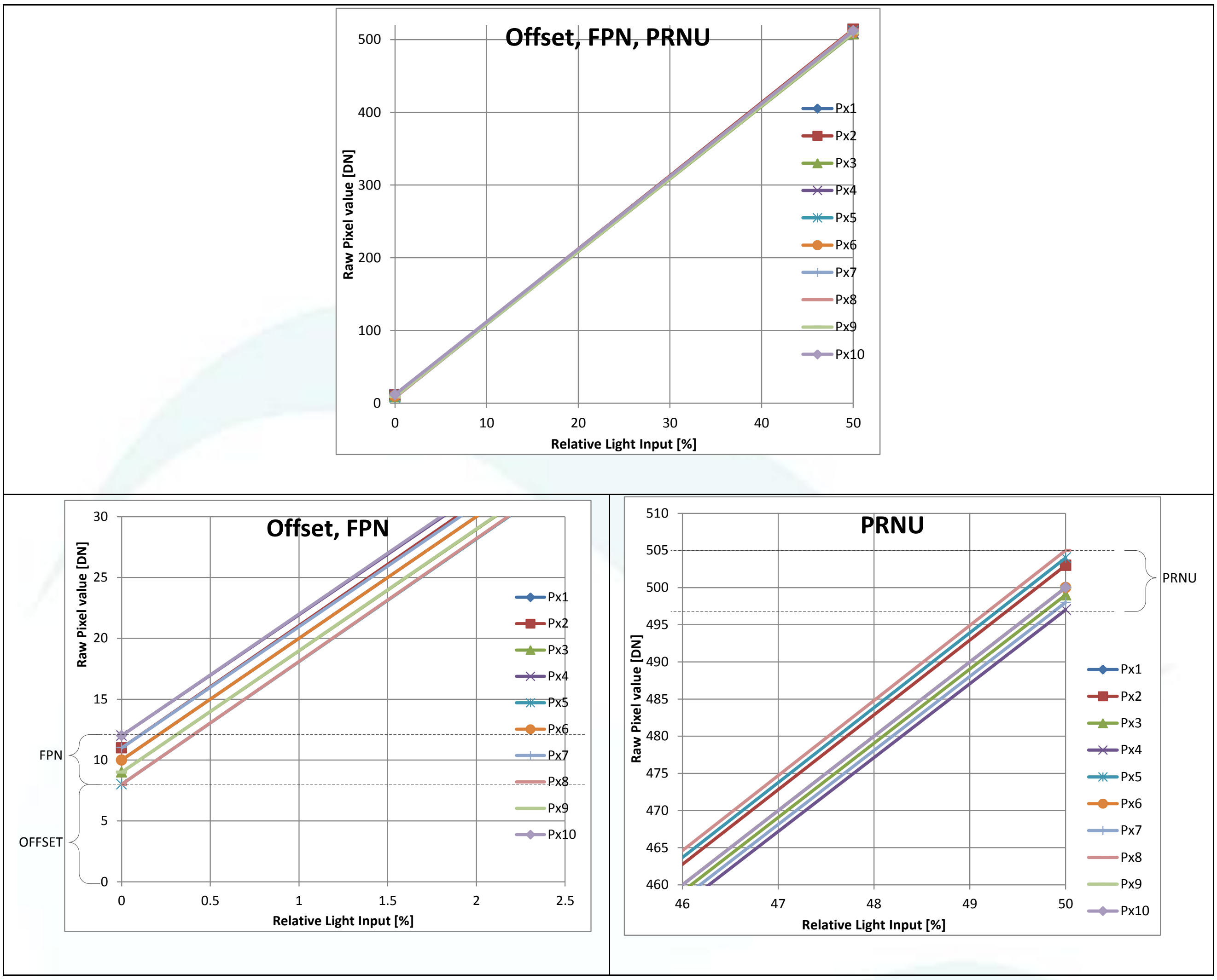


FIGURE 7: OFFSET, FPN AND PRNU

2.1 OFFSET, FPN AND PRNU CORRECTION

To correct all pixels so that they will have the same response as an ideal pixel a correction is needed. Depending on the camera/application you can choose which corrections you need to achieve good results.

So first you will have to subtract an offset from every pixel so that the darkest pixel in the image is at 0DN. Setting the offset lower, will result in clipping in black and therefor you will lose information. After that the FPN should be subtracted per pixel. Finally the gain (= sensitivity) of each pixel should be corrected so all pixels achieve the same grey value when the same amount of light is falling on them. The formula for the corrected value of a pixel is:

$$U_{corr} = \left(U_{raw} - U_{offset} - U_{FPN}\right) * g$$

U_{corr} = corrected pixel data [DN]
U_{raw} = the raw uncorrected pixel data [DN]
U_{offset} = the offset correction value [DN]. This can be set with the offset register.
U_{FPN} = the FPN correction value [DN]. An ideal pixel would have $U_{FPN} = 0$.
g = gain correction value [DN/DN]. An ideal pixel will have g = 0

Let's take the above ten pixels as an example. We want all pixels in dark to have a value of 0DN. At 50% illumination, we want all pixels to have a value of 512DN. The offset value is the value of the darkest pixel. The FPN value is the raw dark pixel value minus the offset value. The gain is calculated from [512 / (RAW grey – offset – FPN)]. So we get the following overview:

Pixel	RAW dark	RAW grey	U_{OFFSET}	U_{FPN}	g	$U_{CORR,\ DARK}$	$U_{CORR,\ GREY}$
1	10	510	8	2	0.024	0	512
2	11	514	8	3	0.0179	0	512
3	9	508	8	1	0.0261	0	512
4	12	509	8	4	0.0302	0	512
5	8	512	8	0	0.0159	0	512
6	10	510	8	2	0.024	0	512
7	11	509	8	3	0.0281	0	512
8	8	513	8	0	0.0139	0	512
9	9	509	8	1	0.024	0	512
10	12	512	8	4	0.024	0	512

If we look at the plots, we can still see a slight slope difference when only correcting offset and FPN.

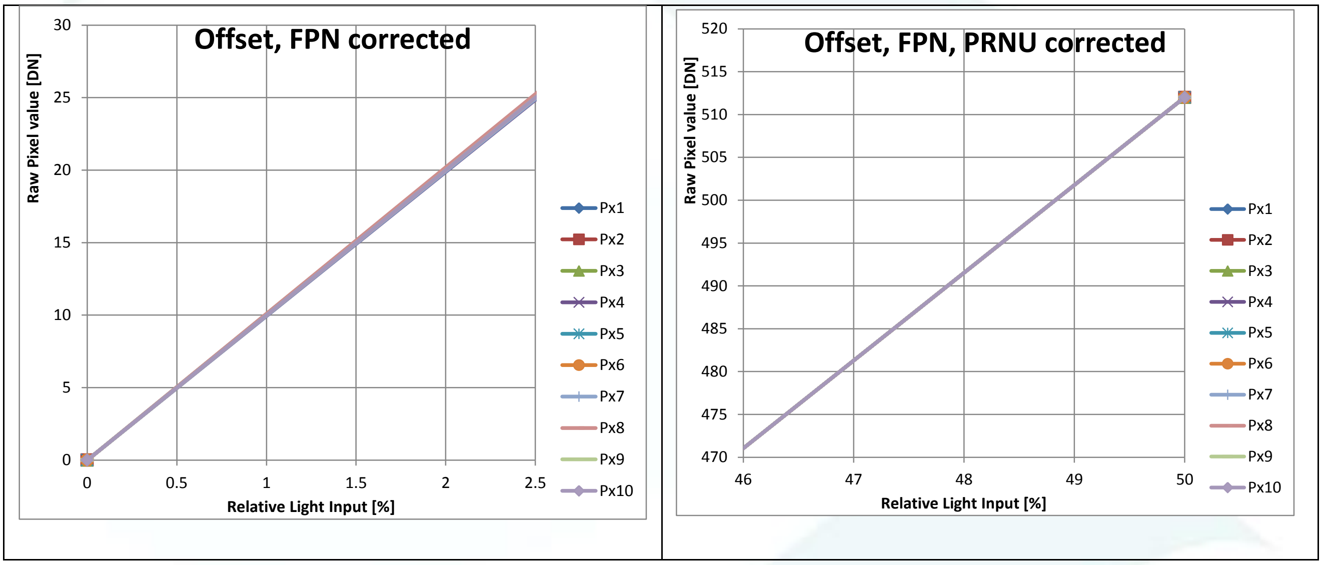


FIGURE 8: OFFSET, FPN AND PRNU CORRECTED PLOTS

Doing an offset, FPN and PRNU correction per pixel is also called flat-filed correction. If done with a lens on the sensor you can also correct for lens vignetting. While a flat field correction will yield the best results, you quite some processing power to do the calculations and memory to store the correction values per pixel.

You can choose to do the PRNU correction per column and/or row instead of every pixel separately. Due to the CDS done in the pixel, the pixel-to-pixel FPN/PRNU is quite small. More noticeable will be the column FPN/PRNU caused by the amplifiers. As every column has its own PGA and ADC, very small differences between them will cause FPN/PRNU visible between whole columns rather than pixels themselves.

When doing column PRNU correction, you will only have to save a 1D array with correction values per column instead of a full resolution 2D array. The same applies to row PRNU correction.

3 Dark current correction

Besides correcting for gain depending on the incoming light, also dark current correction can be done. Dark current will generate charge in the pixels even when no light is falling on the sensor. This dark current is dependent on the exposure time and temperature of the sensor and differs per pixel. A typical figure for the dark current would be 125e/s at 25°C die temperature. For example when exposing a sensor at 25°C for 1 second and with a conversion gain of 0.075DN/e, this will create an additional offset of 9.375DN in that pixel. As the dark current will differ per pixel so for a full correction a full resolution 2-D array of dark current correction values should be used. To get these values you should take two images in the dark; one with a very short exposure time (so little to no dark current influence) and an averaged frame with a long exposure time (so dark current will be noticeable). The difference between the two images for every pixel will determine the dark current of that pixel.

As the dark current is also heavily dependent on the sensor temperature, you can either choose to keep the sensor at a stable temperature so the DC stays the same, or you could monitor the sensor temperature and compensate for the DC. The dark current will double with every 6-7°C increase. So when running the sensor at 51°C, the dark current will be 2000e/s, but at 5.5°C, it will only be around 15.6e/s.

So there are a few options to correct dark current. First of all is to keep the sensor's temperature as low as possible so the dark current influence is negligible and no corrections are needed.

If the camera's temperature is stable during operation, you can choose to use the average datasheet dark current values (125e/s; doubling per 6.5°C increase) to calculate the overall dark current of the sensor depending on the exposure time and subtract that value from every pixel. A variation would be that you monitor the sensor's temperature and accordingly change the dark current calculation.

You can also measure the dark current per pixel at a specific temperature and use that DC value and the general rule of doubling per 6.5°C increase to calculate the offset caused by the DC per pixel and per exposure time and per temperature.

If you want to go crazy, you could measure a full dark current profile per pixel, per exposure time and temperature to have a fully accurate dark current profile per pixel, so you can fully correct each pixel individually. Although this seems to be overkill for most applications.

4 CALIBRATION OVERVIEW

This will give a general overview of how to calibrate each sensor so they will have the same behavior:

1. Start by programming all registers with the recommended values from the datasheet.
2. Take fully dark images with short exposure and calibrate the offset register so no pixel clips in black (< 0DN).
3. Now take images with light and normal exposure. If the image isn't saturated increase the light or the exposure time until all pixels reach a constant value. If not all pixels saturate at 1023 (meaning that the non-linear part of the pixel voltage is in the ADC input range), increase the ADC gain/range setting until they do. The PGA amplifier can also be used at this stage.
4. The dark offset level may have shifted when doing ADC calibration, so repeat step 1.
5. To compensate gain differences between sensors, choose a fixed light setting or exposure time at which the sensor shows a grey image about 50% of its swing (512 at 10bit). Now tweak the ADC setting per sensor so that all sensors will have the same average grey value of about 512. This way all sensors will behave about the same to the same amount of light.
6. Do FPN, PRNU, dark current correction to have all pixels of all sensors to behave exactly the same to the same amount of light.