PETIROC2A

SOFTWARE & TEST BOARD USER GUIDE

Version: 15 February 2023

Abstract

PETIROC2A, standing for *PET Integrated Read Out Chip*, is a 32-channel front end ASIC dedicated to read-out SiPM detectors.

This guide explains how to install and use the test board for PETIROC2A and how to operate with the associated software.

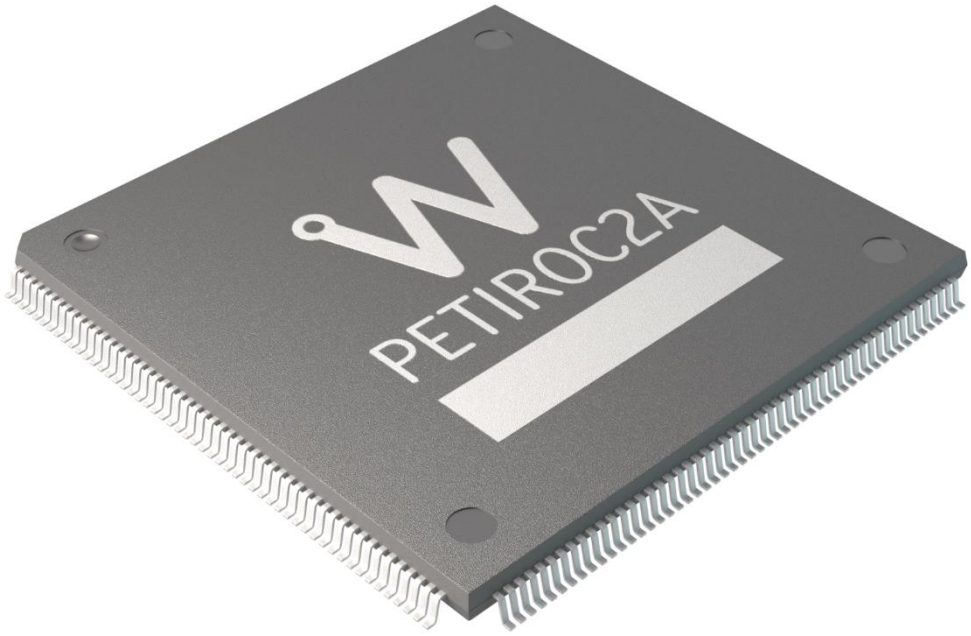


Table of content

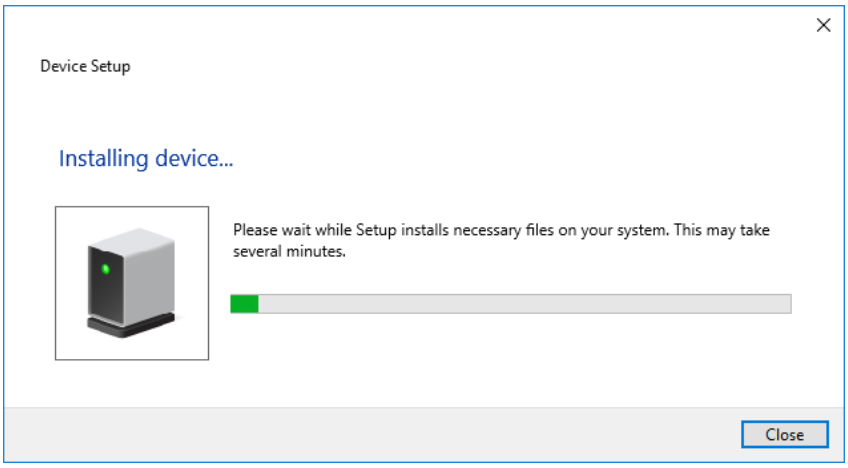
1 Installation & Test of the Test Board

1.1 Pre requisites

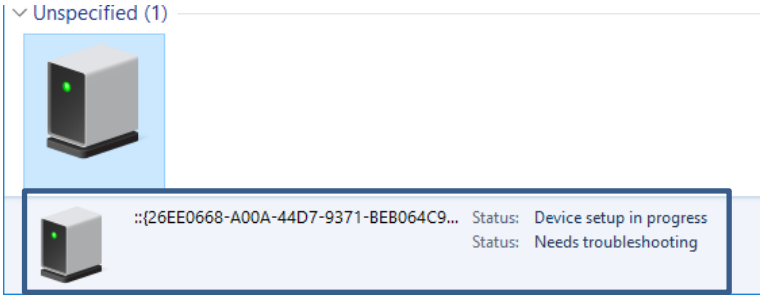
The use of this PCB requires:

- A computer (windows OS) with USB connection
- A USB-A to mini-USB cable
- Optional : A positive output power supply (delivering 500mA)

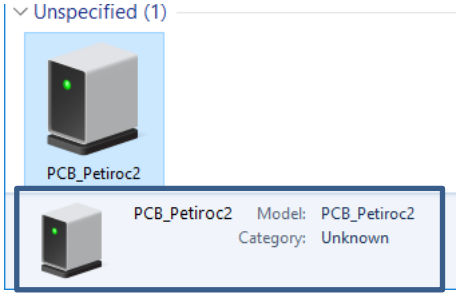
The first time a Weeroc testboard is plugged, the following message should prompt.



In order to verify that the drivers are correctly installed, go in the control panel under the "Devices and Printers" window. the PCB_Petiroc2 device should have gone from



to



1.2 Installation guide

Before running the software for the first time, please verify that the PCB is correctly identified in the "Devices and Printers" window under the control panel. The release of the Petiroc2 user interface can be found in the Weeroc download center on the website http://www.weeroc.com .

2 Evaluation board description

The evaluation board has mainly been developed to allow characterization and debug of the ASIC PETIROC2. Some features were added on the board or in the firmware/software to allow versatility and its use with real detectors or within an experiment. The schematics of the evaluation board, the firmware and software sources are provided on the WEEROC Website, users can modify anything they need to fit their own requirements.

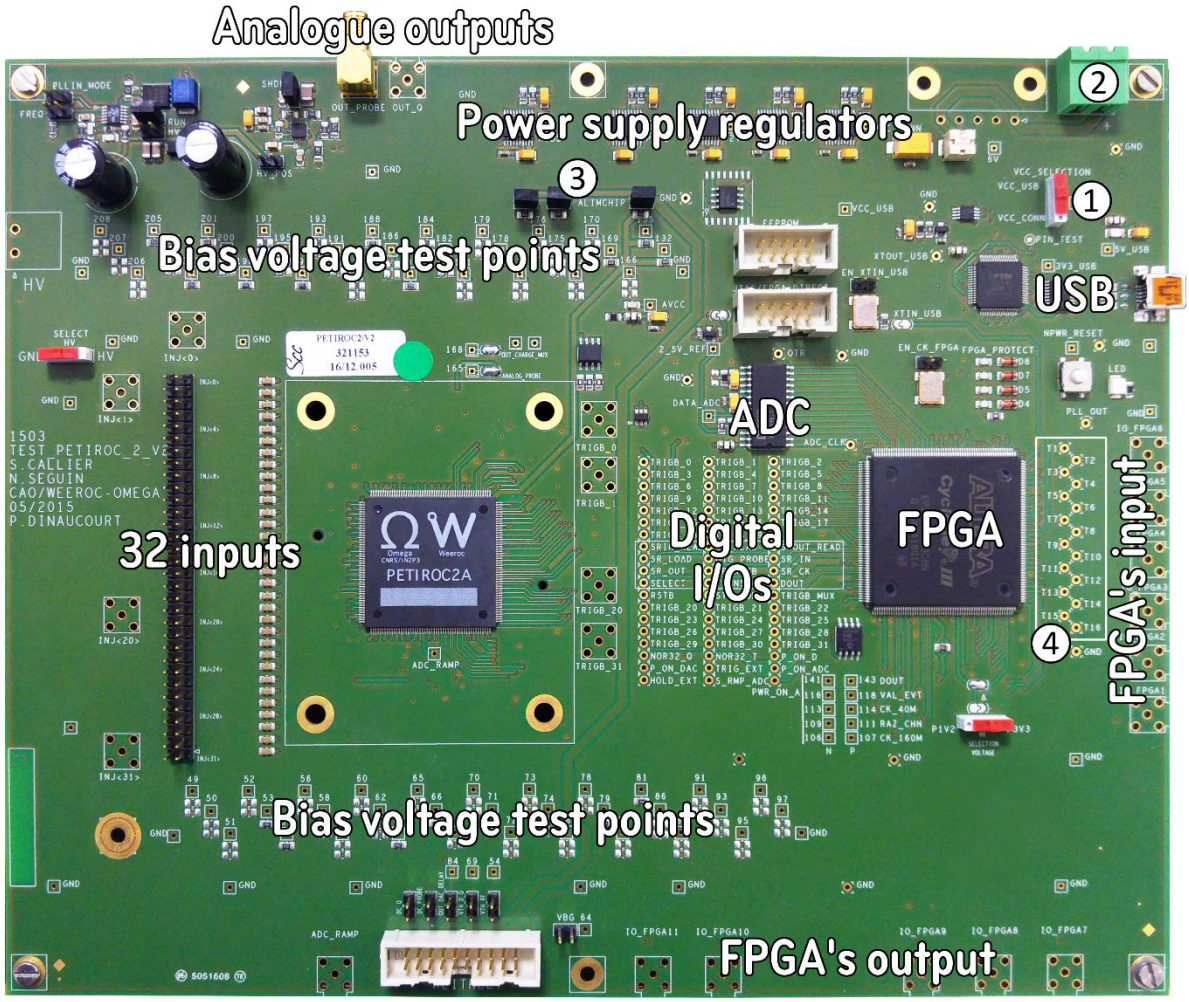


Figure 1 – Top view

- A "VCC_SELECTION" jumper ① allow to choose whether the board is powered through the USB connection or using an external power supply plugged to the top right connector of the board (~6V, ②). The power consumption of the chip can be checked with a voltmeter connected to the dedicated test point

"ALIMCHIP" ③. On the Figure 1 the "VCC_SELECTION" is set on "VCC_USB" so there is no need to plug an external power supply.

- The PCB allows easy access to each PETIROC2's pin, as all analogue pins are connected to bias voltage test points (the corresponding pin number is written on the silkscreen layer) and all the digital I/Os are connected to test points between the ASIC and the FPGA.
- Many test points are also connected to the FPGA ("T1" to "T16", ④), allowing outputting digital internal nodes.
- OR32_Time (OR32_T), OR32_charge (OR32_Q) and Multiplexed trigger output (TRIG_MUX) outputs are available on SMA connectors (respectively FPGA11, 10 and 9)
- 2 analogue buffers provide ASIC's analogue outputs on 50 ohm load on SMA connectors ("OUT_PROBE", "OUT_Q").
- Petiroc2 integrates a 10-bit ADC. Besides, 1 external ADC is on-board, allowing ASIC data acquisitions while the ASIC works in analogue only.
- A 2x32-pin connector allows the use of this board with SiPMs or matrixes. The HV must be provided externally.
- The 32 trigger outputs (trigb<0:31>) are available on the testboard as digital I/Os. The letter "b" in the name signal (trig**b**) means that the signal is active low.

3 Software description

The software has been written in the C# language with the help of visual studio (free version). The source code is available to help comprehend the functioning of the software. This is especially useful if users aim to develop their own DAQ system.

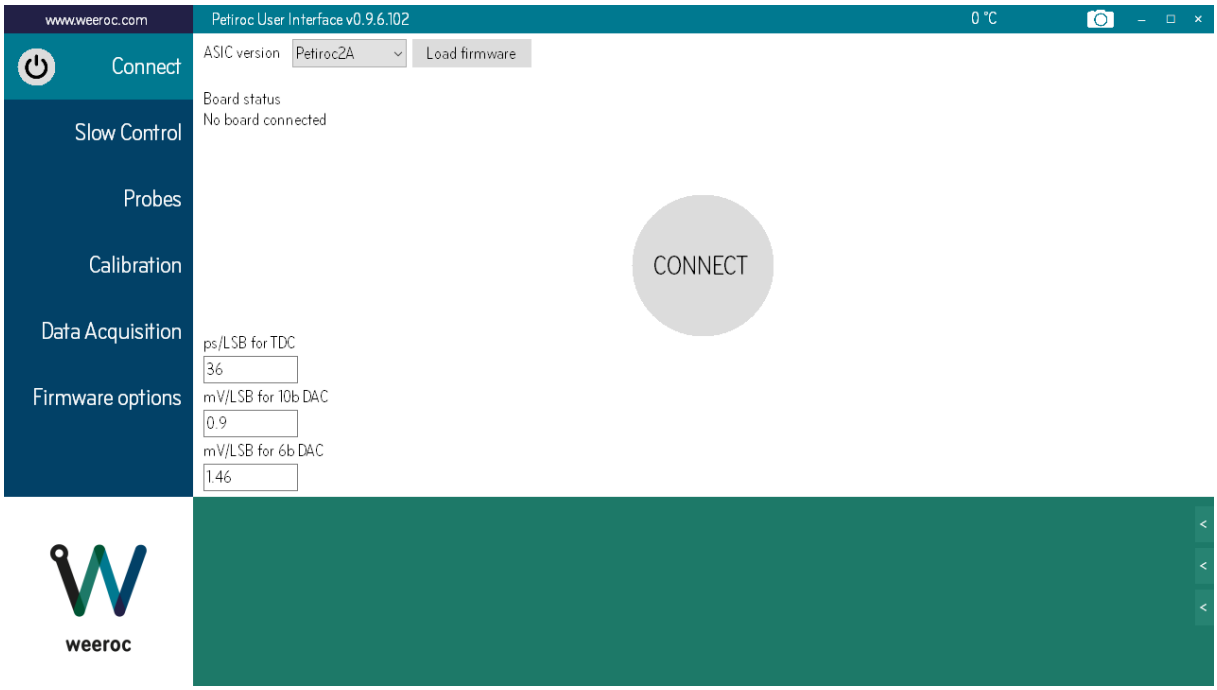


Figure 2 – Petiroc software connect tab.

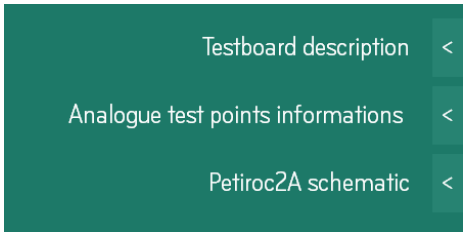
To start the test board you need to:

1. Connect the USB cable from the PCB to your computer.
2. Provide power supply to the PCB if not provided by the USB (the power supply can be switched between external and USB with the jumper on the top right of the evaluation board).
3. Start the software and click on the "Connect" round button.

When connecting the evaluation board, drivers for the USB device should install automatically. If it is not the case, the drivers can be found on the FTDI website (http://www.ftdichip.com/Drivers/D2XX.htm).

When this software is launched and the "Connect" button clicked, no error should occur, meaning that the installation has been done successfully and all the drivers and dll have been found. If a crash occurs or if you need assistance for any other issue with the software, contact the Weeroc support by opening a new support ticket at the address http://www.weeroc.com/my-weeroc/support.

While this user guide will help the user to use the software and evaluation board, it should be noted that there is an embedded help in the software. By hovering controls with the mouse, the green bottom part will be filled with information on the object being hovered. Moreover information about the board and the ASIC can be found by clicking the < buttons on the bottom right.



The firmware version is automatically detected by the software. If the firmware version doesn't correspond to the expected version, the software will propose to load the correct firmware version. The expected

firmware version corresponds to the last number in the software version. The firmware version is 101 on Figure 3, while the software version is 0.9.6.102. Make sure that the correct ASIC version is selected when loading the firmware.

Legacy users : When using the first version of the ASIC, make sure the "Petiroc2" ASIC version is chosen in the combo box before loading the firmware. The pinout of the FPGA has been changed between the board versions for Petiroc2 and Petiroc2A.

On the title bar is also displayed the temperature of the board given by the TMP275 component on the board.

"

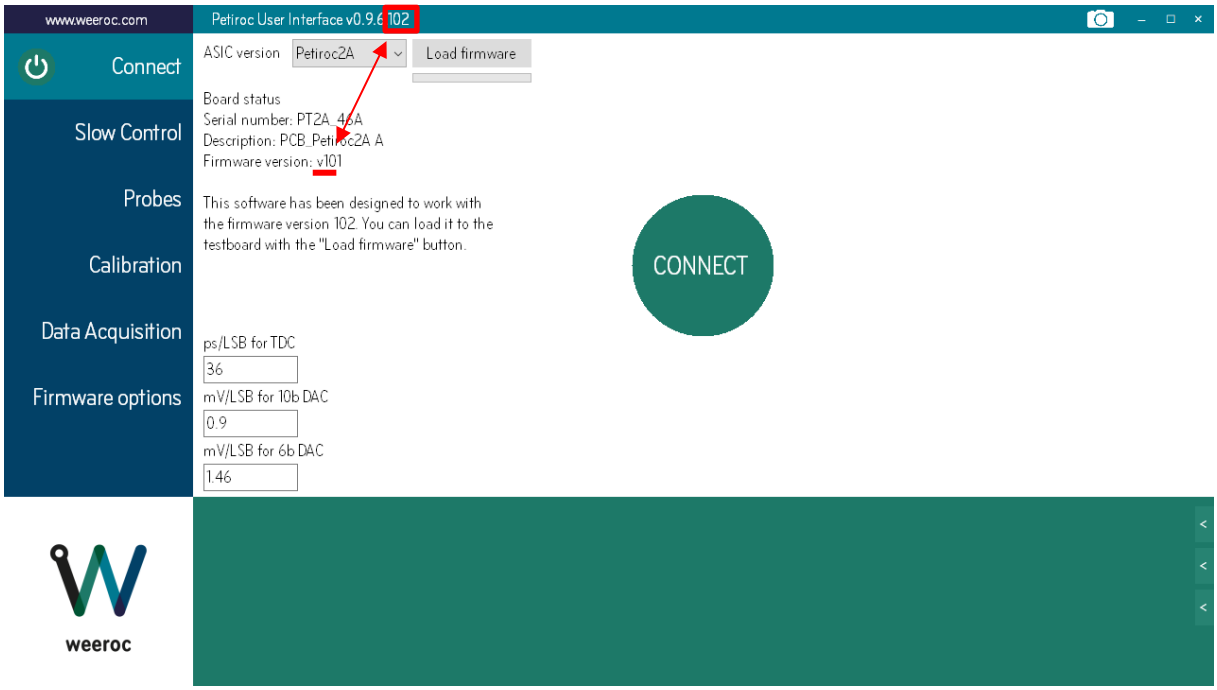


Figure 3 – Detection of the firmware version.

3.1 The slow control page

All the slow control parameters of the PETIROC2 are displayed on this tab control, allowing tuning & tests for different settings.

To program the PETIROC2, users must set the Slow Control parameters and click on the "Send SC" button on the bottom right of the page. The slow control parameters are loaded twice in the ASIC and a checksum is done to verify that the slow control has been correctly loaded in the ASIC. The "Send SC" button turns green if the load succeeded. If it turns red, then there is an error in the slow control bits loading. Verify the board is plugged and powered correctly.

The settings can be saved in a text format file thanks to "Save SC" button, and reloaded from this file by the "Load SC" command or by drag and dropping the file on the software window. This is very helpful as

there are 640 bits of slow control. Alternatively it is possible to export and import slow control files for the LabView software of Petiroc2 with the "Export to LV" and "Import LV file" buttons.

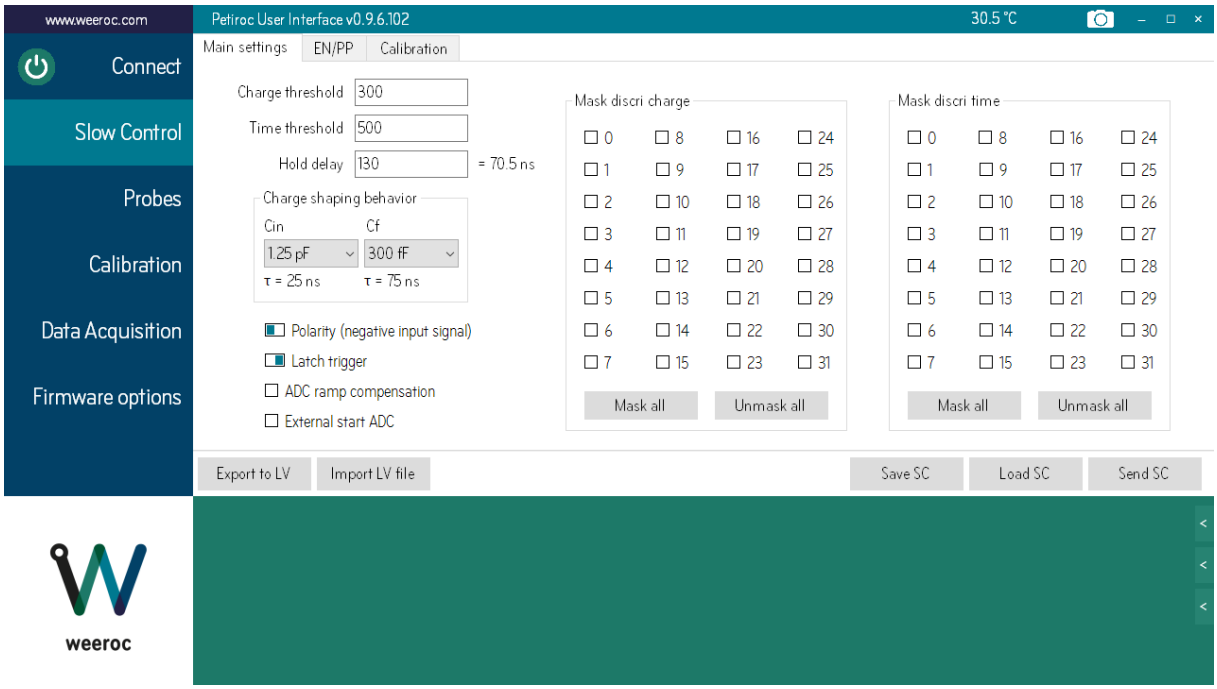


Figure 4 – Slow control main settings tab.

The main slow control page brings together slow control parameters such as the common thresholds for time and charge triggers or the shaping behavior of the charge shaper. Most of the options available here are self-explained (input signal polarity, trigger latch, etc.).

The mask group boxes allow masking triggers from the "time" discriminator and the "charge" discriminator. With the mask, "hit" information will not be registered on these channels and a trigger will not initiate data acquisition. The number next to the checkbox corresponds to the channel number. A check means the channel is masked.

Figure 5 – Slow control enable and power management tab.

The EN/PP tab stands for "Enable" and "Power Pulsing". On this tab it is possible to disable certain parts of the chip in order to reduce the power consumption by disabling unused blocks or for test purpose. Some of these blocks can also be power managed in continuous power mode (Ctn) or in power pulsing mode[1] (PP).

[1] In power pulsing mode, various components in the ASIC could be shut down or powered on according to the sequences provided by users.

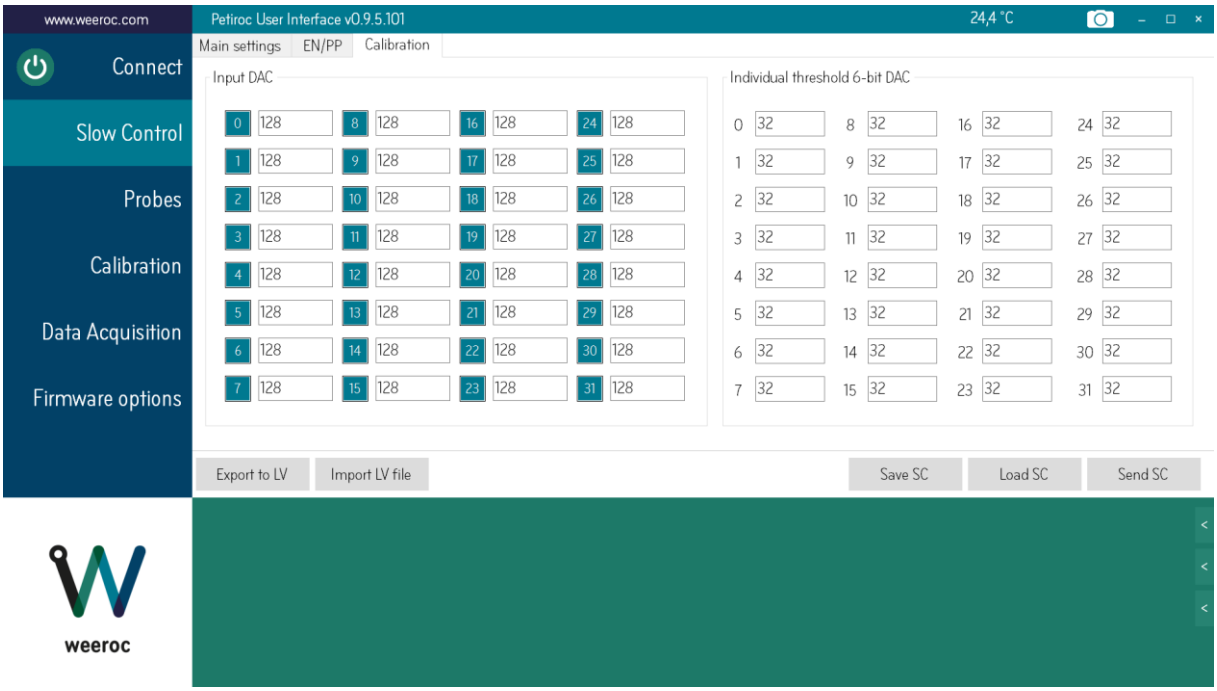


Figure 6 – Slow control calibration tab.

The calibration tab is used for the input DAC and 6-bit DAC values.

The input DAC slow control group box allows user to tune the DC value of the ASIC inputs. When using Petiroc2 with SiPM arrays, the input DAC provides the possibility to user of correcting detector gain dispersion by adjusting individually the high voltage biasing of each input channel. The number next to the textbox corresponds to the channel number. These numbers may differ from the SiPM channel number allocation. The checkbox containing the channel number on the left of the textbox allows to disable input DAC by channel. When the input DAC is disabled no acquisition can be done on the corresponding channel.

The purpose of 6-bit DAC group box almost similar to input DAC except this adjustment is applicable to "time" trigger threshold. This threshold is provided by a 10-bit DAC (available in "Main settings" tab – Figure 4) and it is common to the 32 channels. In order to correct the trigger dispersion, which can be superior to several photoelectrons, a 6-bit DAC for trimming the "time" threshold has been added for each channel.

3.2 The probe page

An internal ASIC shift register allows to monitor each channel and to display them on oscilloscope. User can chose to monitor the following signal from analogue probe register :

- the preamplifier output
- the charge shaper output
- the TDC ramp output
- the threshold set for the time discriminators.

The analogue signal (available after a click on "Send probes" command) is outputted on pin 165 and buffered to the "OUT_PROBE" SMA connector onboard. This probe system is usually used for debugging. In

typical ASIC operation, it is advised to turn off or set the analogue probe to "No analogue probe" in order to avoid analogue performances degradation.

For the internal digital signal, the following output can be observed on the evaluation board "IO_FPGA8" SMA connector :

- The holdb signal from the time trigger delay box output ;
- The start flag of the Wilkinson ADC ramp (and data conversion) ;
- Charge discriminator trigger for each channel.

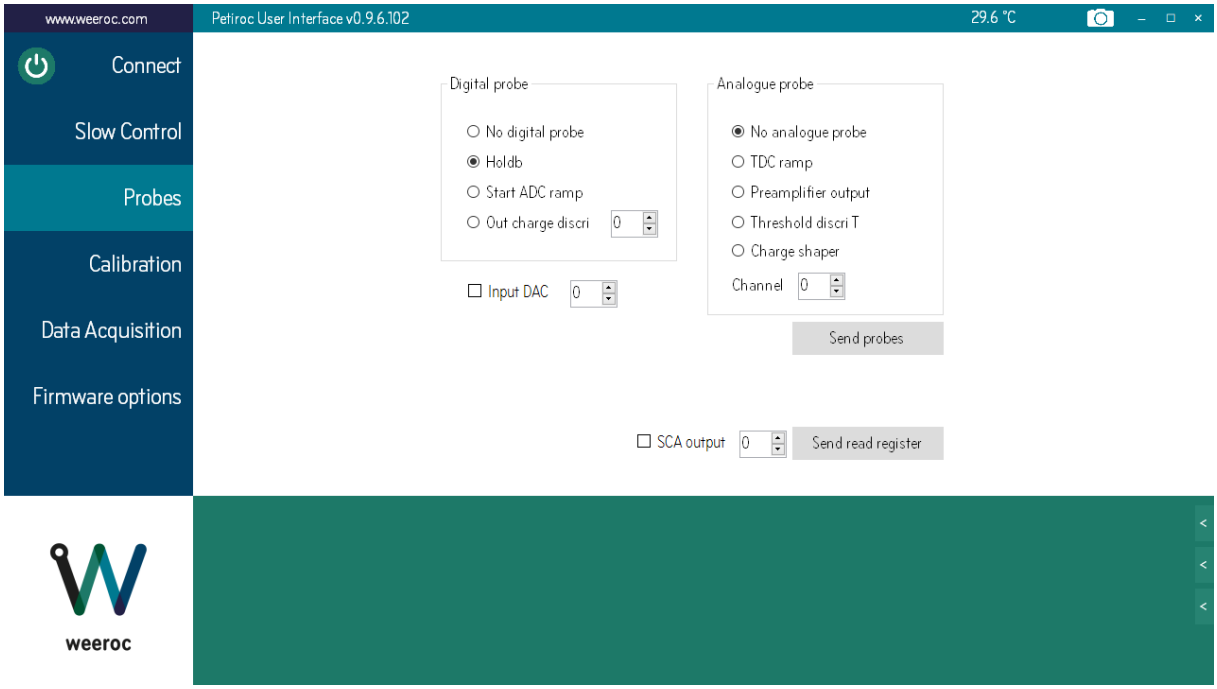


Figure 7 – Probes register page

The SCA output signal is sent on the "OUT_Q" SMA connector on the testboard thanks to the read register.

3.3 The calibration page

3.3.1 S-curves

The first tab in this page is called "S-curves" and its aim is to perform trigger efficiency test or S-curves. Its secondary function is to calibrate the 6-bit DAC setting of each channel so that the channel-by-channel trigger dispersion will be reduced. The user has the choice of simply plotting the S-curves with the current slow control configuration by clicking on the "Start S-curves" button. The S-curves can be plotted on the pedestal or on the signal thanks to a switchbox. To plot the S-curves on a signal you will need to inject signal as shown in section 3.7.1. The clock speed can be adjusted from 1 kHz to 100 kHz. Slower clock will make the acquisition of the S-curves slower but the S-curves will be cleaner (for more information on S-curves, see appendix).

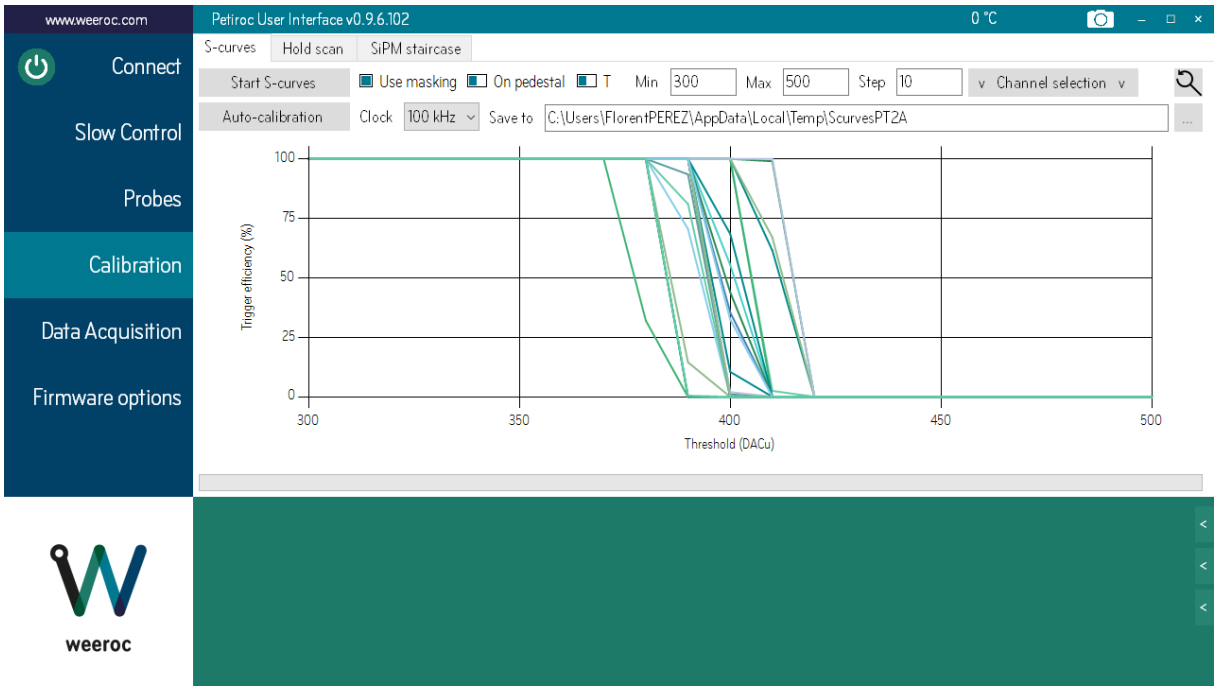


Figure 8 – S-curves tab.

Without any calibration, the dispersion of the S-curves positions is quite large. When clicking on the "Auto-calibration" button, the software starts a 3-step calibration of the S-curves. A message box will be prompted asking the user to define a slow control output file in order to save the calibrated slow control parameters in this file.

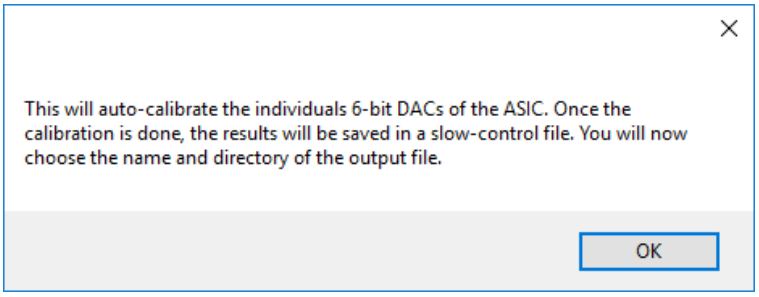


Figure 9 – Message prompt when clicking the "Auto-calibration" button.

The slow control parameters of the 6-bit DAC tab (Figure 6) are automatically updated once the operation is finished. After calibrating the 6-bit DAC, the S-curves should look like on the Figure 10, with a very narrow dispersion. The x-axis represents the 10-bit time threshold DAC code. This plot should allow users to choose a suitable time threshold for triggering.

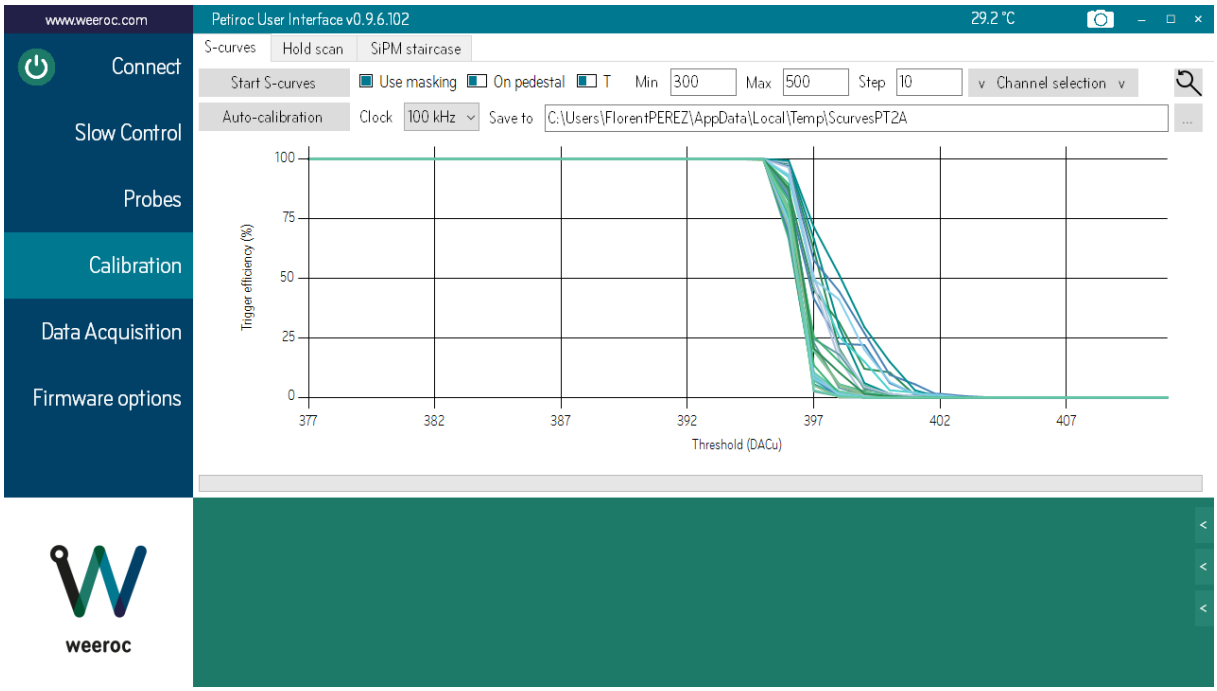


Figure 10 – Calibrated S-curves.

The "Use masking" checkbox is automatically checked during the auto-calibration. This means that non-measured channels are masked (not triggering) during S-curves acquisition to avoid disturbances and unwanted signal couplings. Once the S-curves are correctly calibrated, plotting them without trigger masking ("Start S-curves" button) will result with something similar to the plot on the Figure 11.

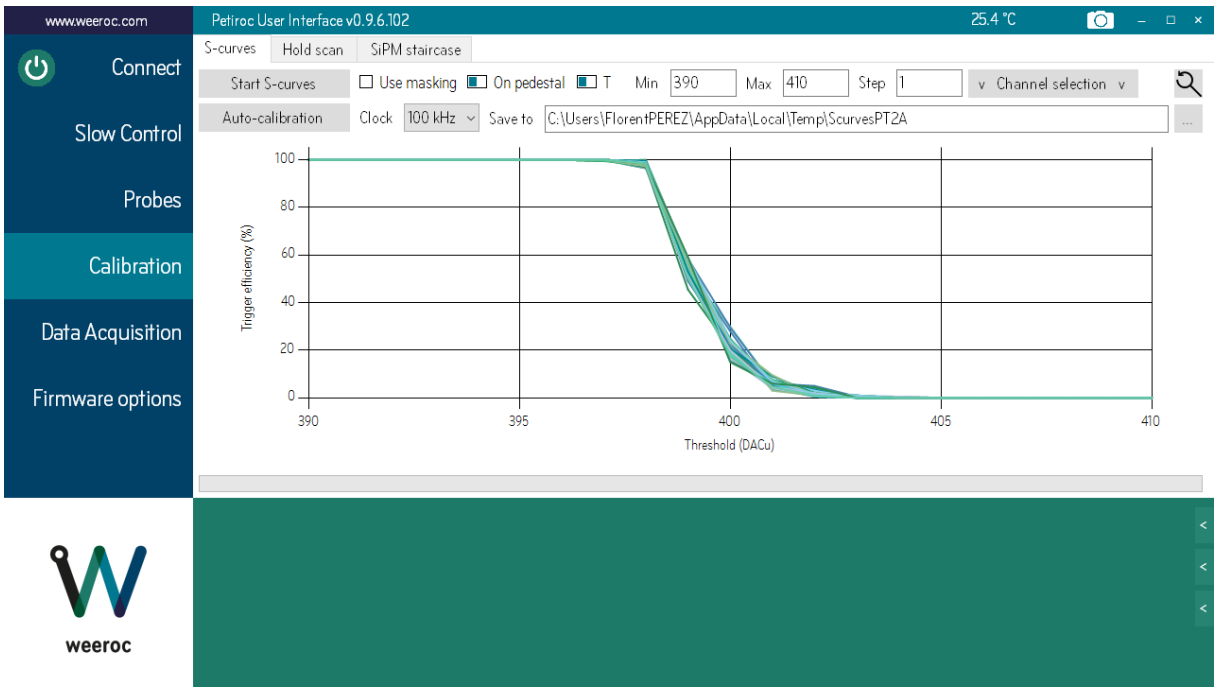


Figure 11 – S-curves without trigger masking.

What is happening here is that the noisiest channel brings all the other ones to its level of noise by crosstalk. The S-curves all seem perfectly aligned.

The S-curves can be plot on the signal instead of the pedestal to verify the positioning of the 50 % trigger efficiency of the injected signal by checking the checkbox "On pedestal". The text next to the box will then become "On signal". When plotting S-curves on signal the user have to make sure that the injection is made synchronously with the S-curves clock. The S-curves clock is generated in the FPGA and is outputted on IO_FPGA2. The clock rising edge must trigger the injection step (see section 3.7.1 on page 20).

It is also possible to plot only selected channels in the channel selection.

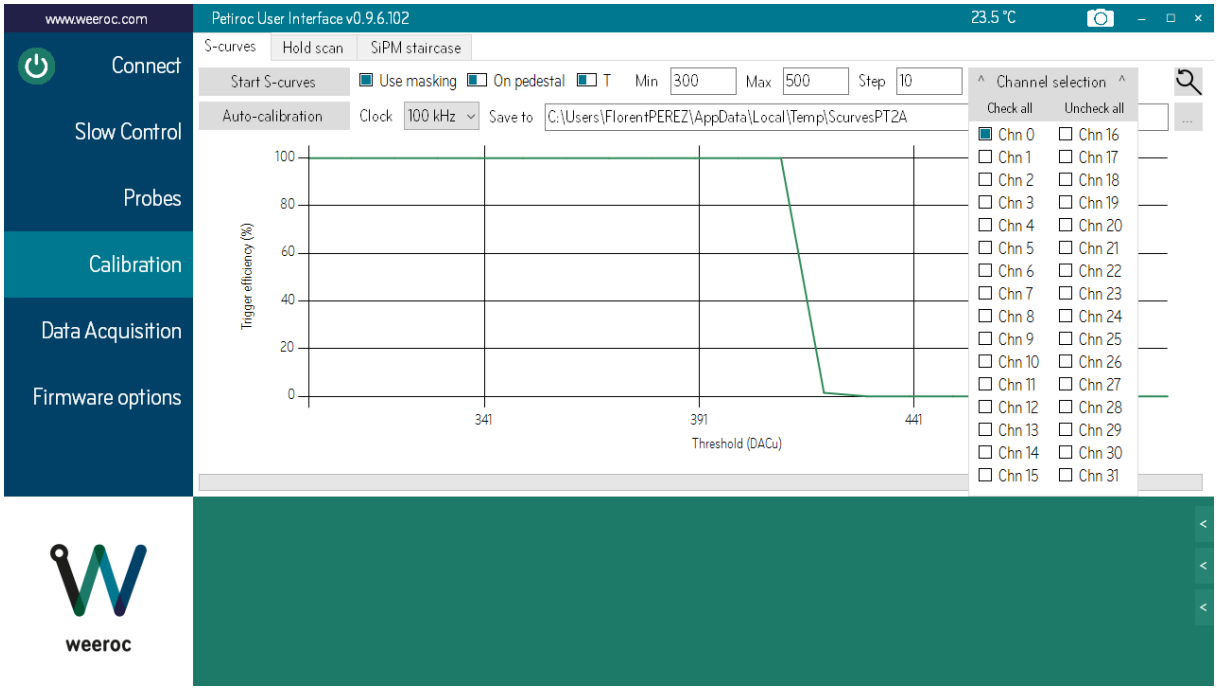


Figure 12 –S-curves of channel 0.

3.3.2 Hold scan

The second tab under the calibration page is the "hold scan". What it does is tracing the charge shaper output signal by varying the "hold" delay value. In order to have it working properly a signal should be injected in the measured channel and the time discriminator must trigger on the signal. A fit on the result of the scan is performed and the x-axis value corresponding to the maximum of the fit is automatically extracted. The slow control delay parameter (in Main settings tab – Figure 4) is updated with the extracted value. The following plot has been done by using the default software configuration and by injecting a negative step of 100 mV @ 1 kHz.

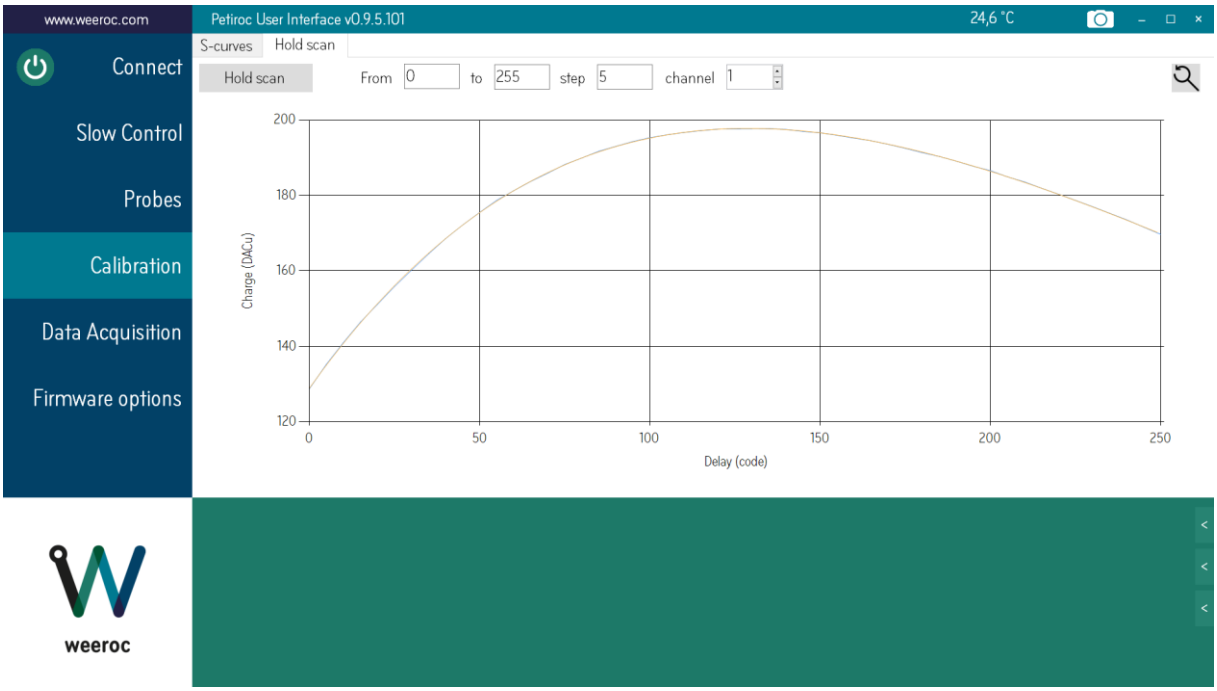


Figure 13 – Hold scan tab.

It is advised that the hold scan should be redone if the charge shaping parameters (Cin and Cf) are changed through the slow control parameters because the peaking time of the shaper will vary. Once the proper delay parameter corresponding to required shaper peaking time is selected, data acquisition can be effectively done.

3.4 The data acquisition page

To start an acquisition, simply click on the "Start acquisition" button. On the top of the page can be set the number of acquisitions, the acquisition trigger mode and the name of the output file, where the data will be saved. A high number of acquisitions between each graphics refresh allows for faster acquisitions, the max being 100 (limited by the FIFO size in the FPGA). The different acquisition trigger modes are :

1. Time trigger
2. Time trigger with Charge trigger validation
3. Coincidence between two time triggers of different channels
4. Coincidence + Charge trigger validation

The Coincidence test is made between two groups of triggers, T<0:15> and T<16:31>. The window for the coincidence is 10 ns.
For the charge Validation, an analog reset is sent to the ASIC if there is no Trigger Charge in the 200 ns following a Trigger T. For Coincidence + Charge Validation, it needs a 10 ns coincidence and a Trigger Charge within 200ns.

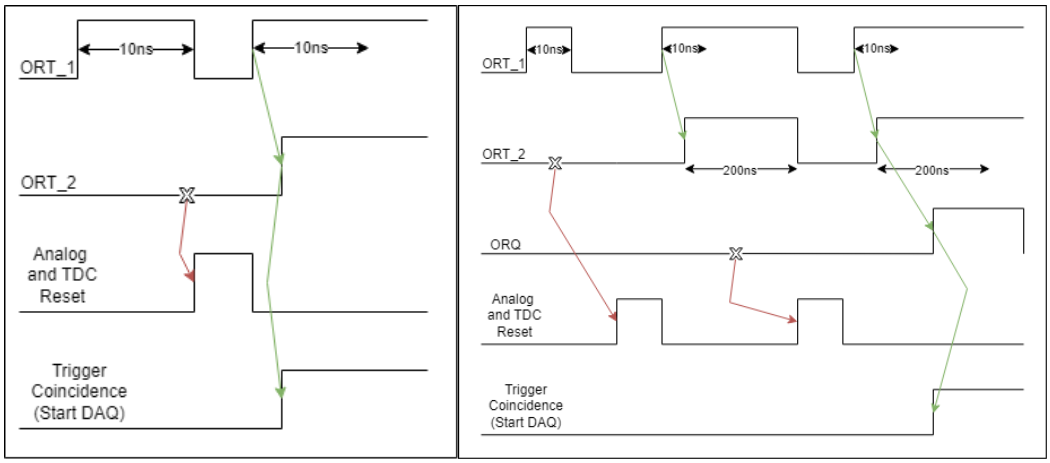


Figure 14 – Coincidence and Charge + Coincidence Chronogram for DAQ in the FPGA

Output data can be visualized in four different ways. If the number of acquisitions is very large (> 10 Go) the data won't be open in soft as this software is not developed to handle such large data. Large data set, depending on the performances of the computer, can take some time to be displayed. The output data file is constructed as shown on Figure 15.

FineTime0	Charge0	FineTime1	Charge1	...	FineTime31	Charge31	CoarseTime0	Hit0	CoarseTime1	Hit1	...	CoarseTime31	Hit31
4	33	473	159	...	4	33	201	0	258	1	...	201	0
4	33	483	159	...	4	33	201	0	188	1	...	201	0
4	32	495	159	...	4	34	201	0	183	1	...	201	0
4	34	508	159	...	4	34	201	0	177	1	...	201	0
4	33	521	159	...	4	33	201	0	171	1	...	201	0

Figure 15 – Output data format.

The column are self-explanatory. "FineTime0" is the TDC measurement of time of channel 0, expressed in 10-bit TDC units (1 TDCu ≈ 36 ps). "Charge0" is the ADC measurement of the charge of channel 0, expressed in 10-bit ADC units (1 ADCu ≈ 2 mV). "CoarseTime0" is the coarse time measurement of channel 0, the LSB is 25 ns and the value is coded on 9 bits. The "Hit" flag tells if a trigger occurred during the acquisition. Time measurement on a channel is expressed as

$$(CoarseTime + 1) \times 25\,ns - FineTime \times 1\,TDCu$$

Each row corresponds to 1 acquisition. On the Figure 15, 5 acquisitions are represented.

The first way to visualize data in the software is "per channel". It displays the histograms of charge and time measurements for the whole bunch of acquisitions. The number of registered hit in the currently displayed channel is given on a label over the charts (Figure 16). By right-clicking the chart, the user has access to a context menu with a fit option (Gaussian only) to fit the peak. Chart data can also be exported to various output format (.txt, .xls, .csv or .xml). Note that the "load data" button only works with the output data format shown on Figure 15 and is used to load previous acquisitions data set in the software. Exports of the charts data will have to be opened in third-party software.

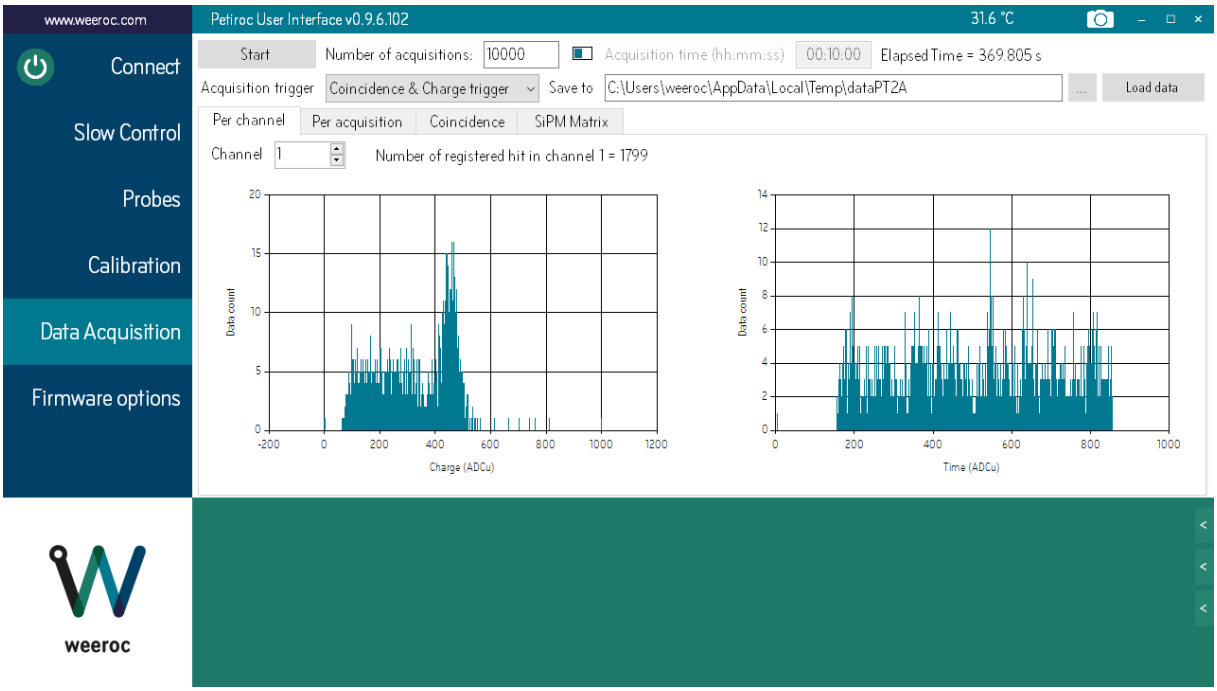


Figure 16 – Per channel data visualization. 511 keV peak of Na22 is shown.

The visualization can also be done per acquisition (Figure 17). The charge and time measurements are displayed for all the channels during a single acquisition. The acquisition number can be changed to browse through all the acquisitions. The red channels on the charge measurement means a "hit" has been registered in this channel.

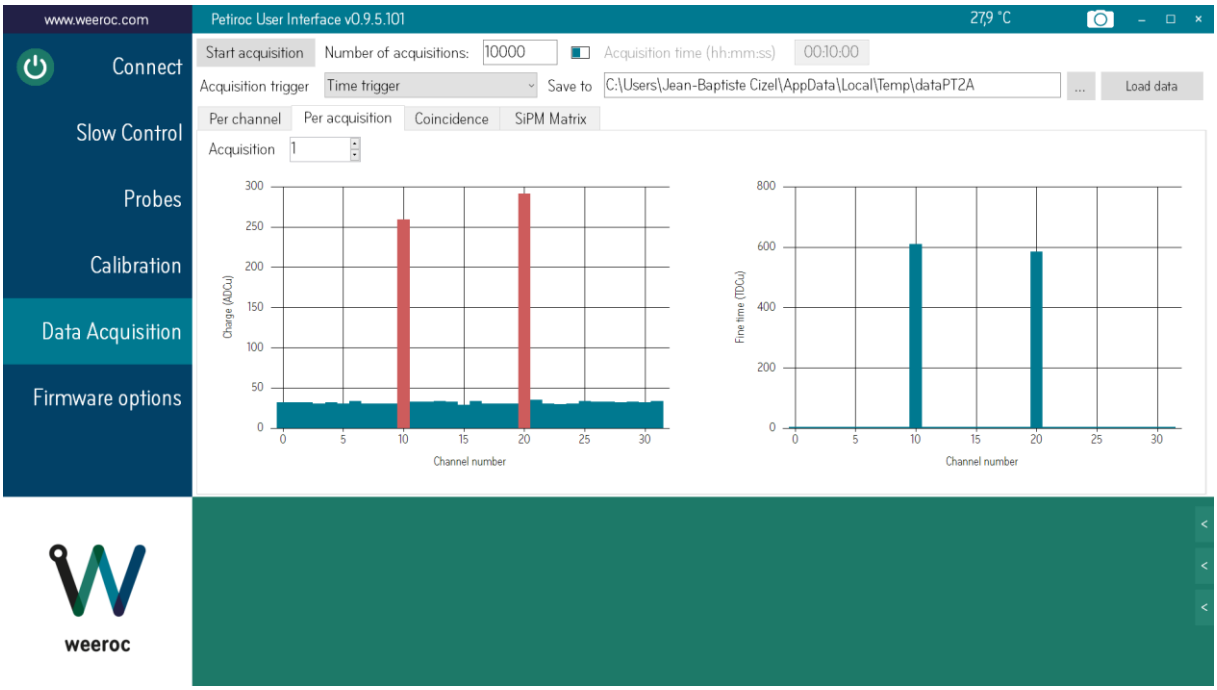


Figure 17 – Per acquisition data visualization.

The next data visualization is for coincidence analysis only (Figure 18). Two channels are chosen by the user to be studied. If the chosen channels have a "hit" tag in the same acquisition, the software will compare their timing and plot the results in a histogram. A Gaussian fit is automatically done on the whole data set but the fit can be refined by the user afterward by right clicking the chart area. An energy filter can be applied to take into consideration only the events of sufficient energy. As an example on the shown figure, the energy has been filtered to only take into consideration the energy of the photopeak between 650 and 750 ADCu.

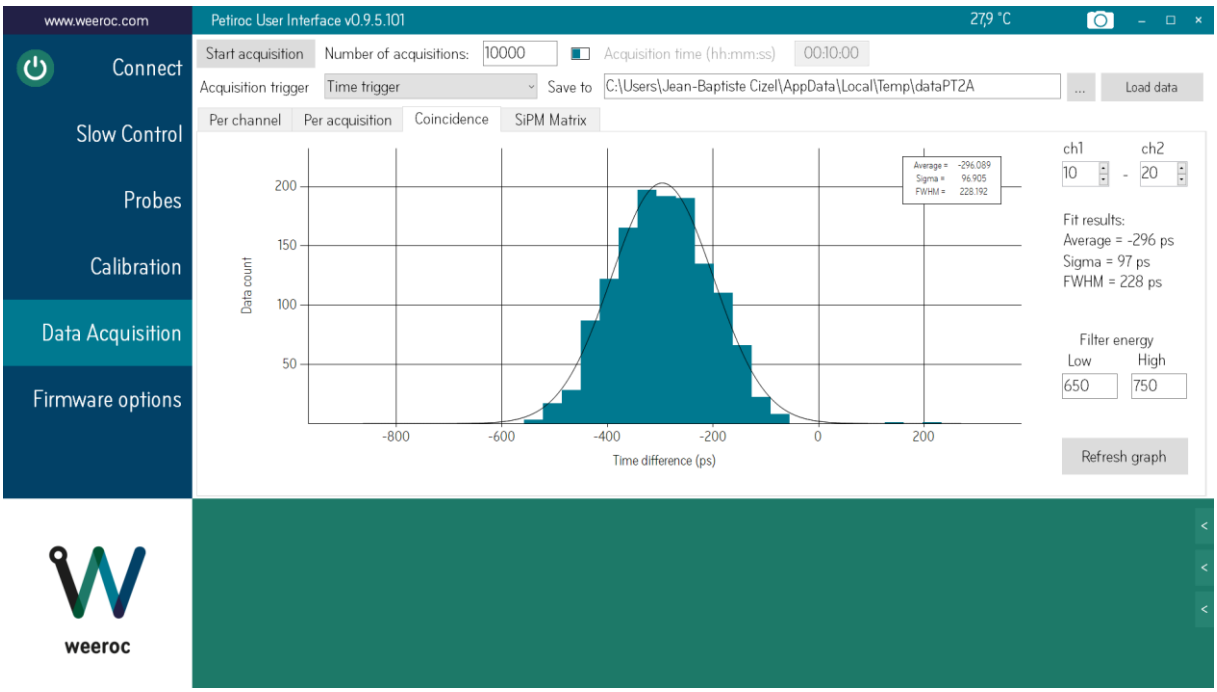


Figure 18 – Coincidence data visualization.

The last data visualization is similar to the per acquisition visualization but the results are displayed with the physical layout of the SiPM matrix. The channel mapping can be changed thanks to the "Show mapping" checkbox. On the left is displayed the charge measurement and on the right the fine time measurement with gradient color display. On the right matrix is also written on the pixels the coarse time without the color information. If no hit are registered, the pixels corresponding to those channels stay plain. The pixel is only drawn when a hit has occurred.

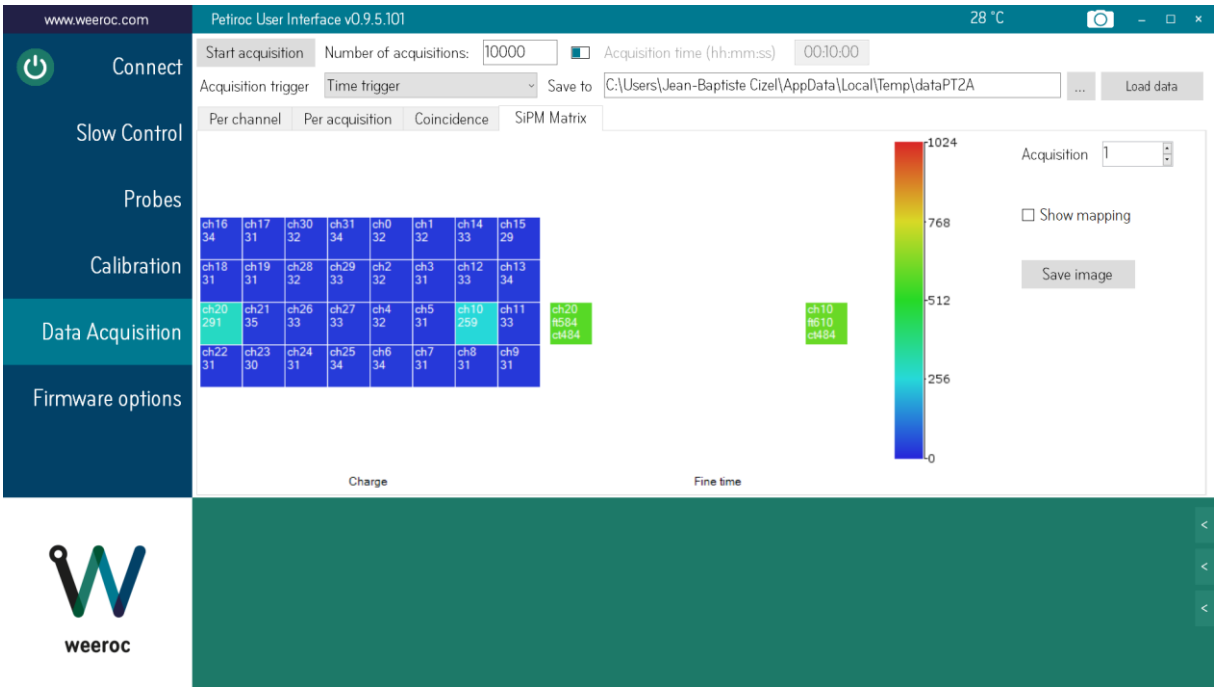


Figure 19 – Sipm Matrix visualization

3.5 The firmware options page

The settings in this page are reserved for advanced user.

3.6 Bias voltages

When the testboard is powered for the first time or when the board is not working properly, the user must check with a voltmeter the bias voltages on the test points available on the testboard. The approximate expected values are listed below. The deviation from these values should be inferior to ± 50/100 mV with a 3.3 V power supply. Before doing this test, the SC parameters must be sent using the software.

Pin #	Name	Expected value
53, 55 …	vdd	3.2 or 3.3V
64	vbg	2.5V (no PP), bandgap output
49	vref_time	2.2V
50	vcasc_time	2.2V
51	vcasc_discri	2.3V
52	vslope_tdc	350mV
54	vth_Time	Depends on the 10b DAC code of the time threshold
56	vref_inpdac	1V
58	vref_10bdac	vdc out dummy, ~900mV to 1V
59	iref_10bdac	2.25V
60	vcasc1_tdc	1.2V
62	ib_otabg	
65	vcasc2_tdc	1.7V
67	vcasc_time_pad	2.2V
69	vth_discri_charge	Depends on the 10b DAC code of the charge threshold

70	vref_tdc	900 mV
71	vref_time_pad	2.2V
72	vslope _delay	250 mV
73	ib_adc	2.27V
74	vcasc1_tdc_pad	1.2V
75	adc_ramp	Ramp of the Wilkinson ADC.
76	vref_adc	980 mV
78	vslope_adc	300 mV
80	ibi_cs_adc	2.5V
84		pad to set an external value
85	hold_ext	OR between hold ext and OR32 delayed
91	ibi_delay	2.4V
93	ibi_rx	2.1V
95	ibo_rx	1.2V
96	ibi_tx	600mV
97	vcm_tx	1.2V
166	vref_charge	1V
168	out_charge_mux	
170	ibo_discri_adc	2.3V
174	ibo_discri_charge	2.3V
175	ib_sca	855 mV
176	iref_inpdac	410 mV
177	ibi_discri_charge	780 mV
179	ibo_charge	810mV
181	ibi_charge	800 mV
182	ibo_inpdac	800mV
184	ibm_inpdac	530mV
186	ibi_inpdac	2.5V
189	ibi_tdc	2.4V
191	ibo_tdc	800mV
193	ibo_cs_tdc	2.3V
195	ibm_cs_tdc	2.3V
197	ibi_cs_tdc	2.5V
200	ib_6bdac	600 mV
	ibm_discri	620 mV
204	ibo_discri	1.4 V
206	ibi_discri	785 mV
207	ibo_time	1.3V
208	ibi_time	1.3V

Table 1 - Analogue test points information.

3.7 Setup to inject signals

3.7.1 Injection of a voltage step

First tests to be more familiar with the board and the software should be done using the following setup. A voltage step can be injected in one channel (a 100 nF capacitor in series with the signal as well as a

resistor for the cable adaptation are already soldered on the testboard). A specific cable with a female "BERG" connector must be "homemade" to inject in each channel as the input connector is a male HE13 - 2x32pins. Otherwise, SMA connectors (INJ<i>) are available to inject in channels 0, 1, 20, 31. SMA connectors are available also for the trigger signals corresponding to these channels next to the digital I/Os for easy testing.

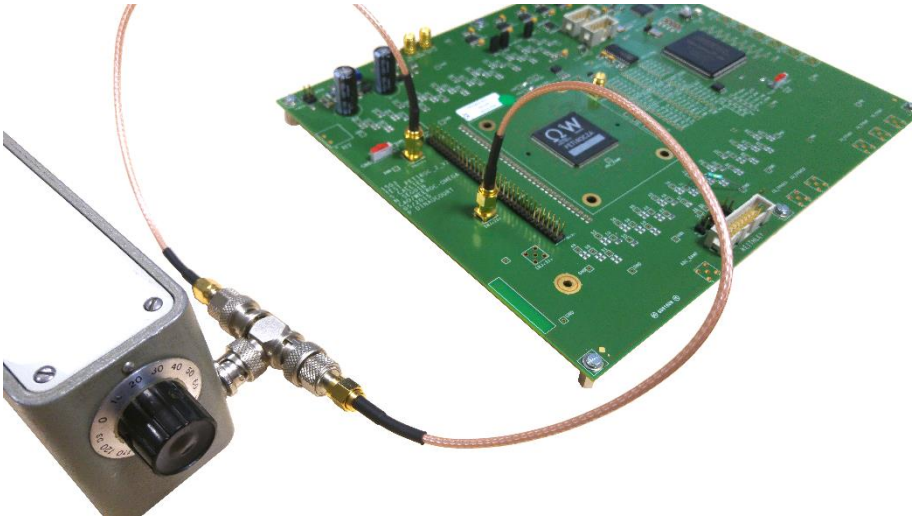


Figure 20 – Injection in channel 1 and 20.

The waveform of the injected signal is displayed Figure 16. Input polarity must be set to negative in the slow control parameters "Main Settings" tab (Figure 4) as the step is a negative one in this case. A slow positive ramp allows to inject no significant signal in the ASIC before the next step. A voltage attenuator is needed to diminish the noise from the waveform generator. A 1 mV voltage step in a channel corresponds more or less to 1 photoelectron.

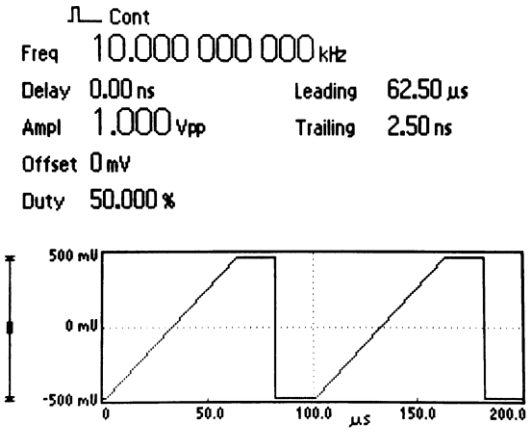


Figure 21 – Injection in each channel

You can then probe the output of the shaper (OUT_SSH) using the "probes" page and check the waveform on the scope. The waveform displayed below was obtained by injecting a voltage step of 1V@20dB.

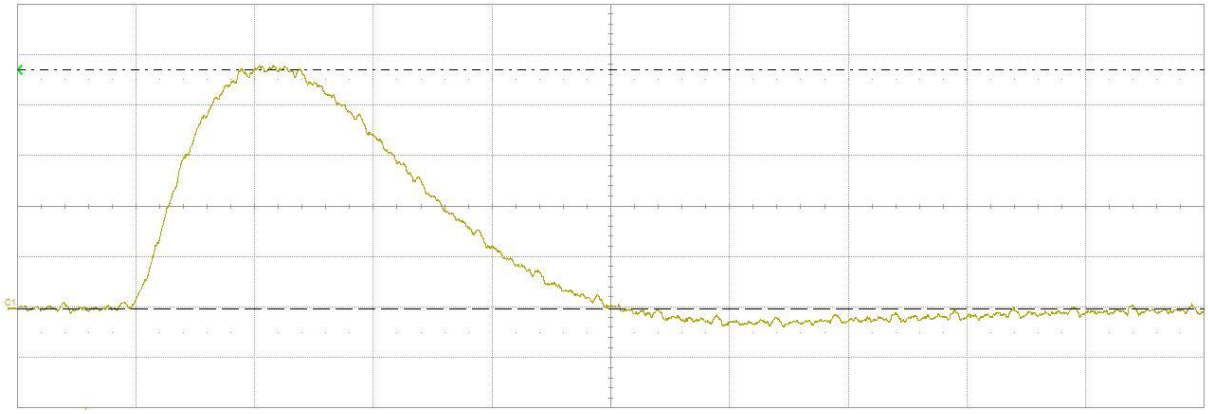


Figure 22 – Slow shaper output through the probe. Y-axis : 50 mV/div, X-axis : 50 ns/div.

3.7.2 Setup for SiPM connected to the PCB

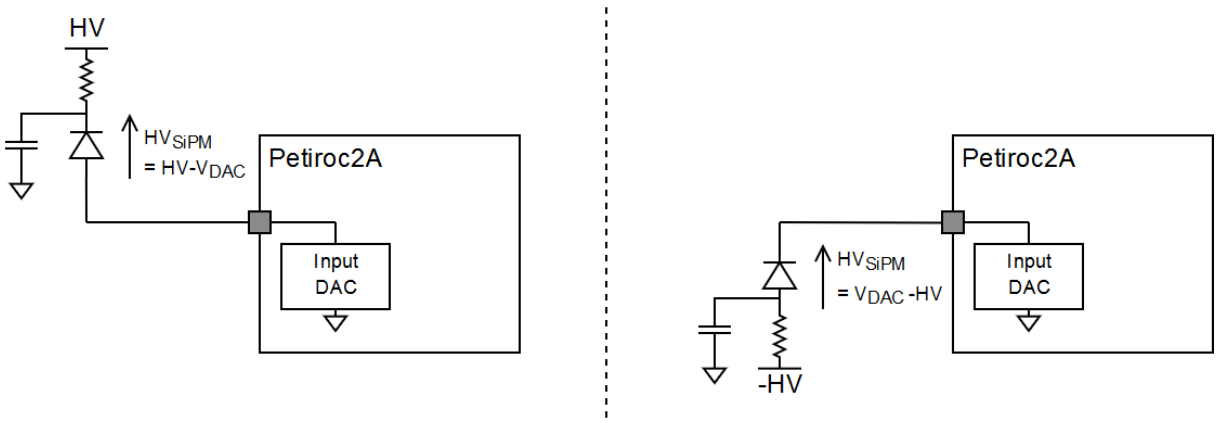


Figure 23 – Left : Negative SiPM injection. Right : Positive SiPM injection.

CAUTION: when the High Voltage is applied to the board/chip, check that the High Voltage has a slow ramp-up to avoid input-DAC destruction.

4 Appendix

4.1 S-curves

Two Gaussians are plotted on Figure 24. The red Gaussian stands for the noise probability density function around the pedestal of the signal with the average set at x = 0 and the standard deviation σ = 1. In the case of a perfect Gaussian distribution, the RMS noise corresponds to the standard deviation σ so here is represented a signal with a RMS noise of 1. The blue Gaussian represent a signal with a RMS noise of 1 and the signal over noise ratio is 10.

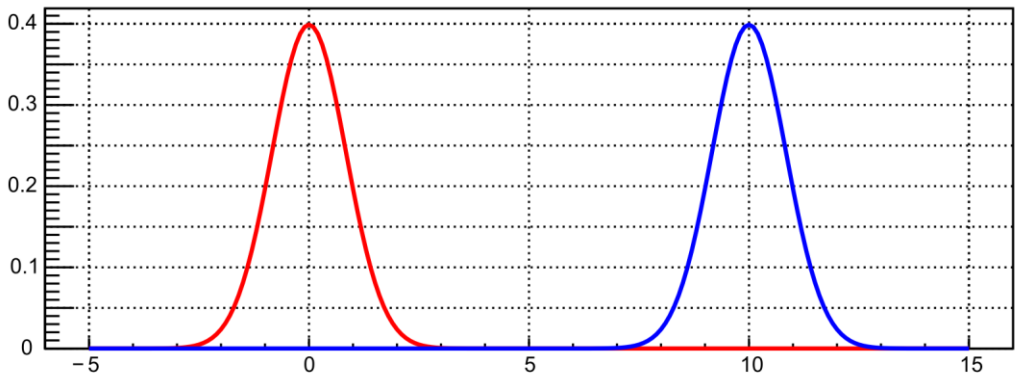


Figure 24 – Red : Normalized Gaussian with average = 0 and **σ** = 1 (arbitrary units). Blue : Normalized Gaussian with average = 10 and **σ** = 1.

By setting a threshold at x = 5 and by checking the position of the signals noise at a random time, the probability for the noise of the pedestal to be over the threshold is

$$0{,}5 \times \mathrm{erfc}\left(\frac{5}{\sqrt{2}}\right) = 2{,}8665 \cdot 10^{-7} \#(1)$$

With *erfc()* the complementary error function. This result corresponds to one chance in 3.5 million. On the other hand the probability for the noise on the signal to be under the threshold is also 1/3.5M. The above equation is the complementary cumulative distribution function of the normal distribution evaluated at x = 5, μ = 0 and σ = 1. This function is written

$$S(x) = 0{,}5 \times \mathrm{erfc}\left(\frac{x-\mu}{\sigma\sqrt{2}}\right) \#(2)$$

With μ the mean (average of the signal or pedestal here) and σ the standard deviation (equivalent to the RMS noise). This function allows to evaluate the probability for an event to happen over a set threshold. The graphical representation of this function is shown on Figure 25.

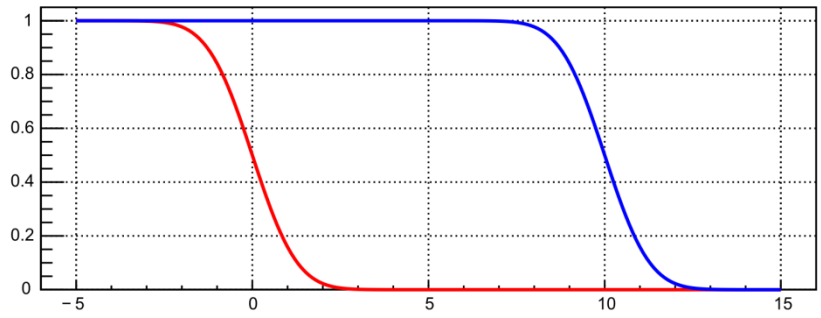


Figure 25 – S-curve representation of the signals shown on Figure 24.

This describes the probability to trigger on a discrete signal but concerning the pedestal it is a bit more complex because the trigger probability depends on the acquisition time as demonstrated on Figure 26 with a threshold at 5 whose trigger happens at the beginning and a threshold at 10 which triggers at 900 (arbitrary units).

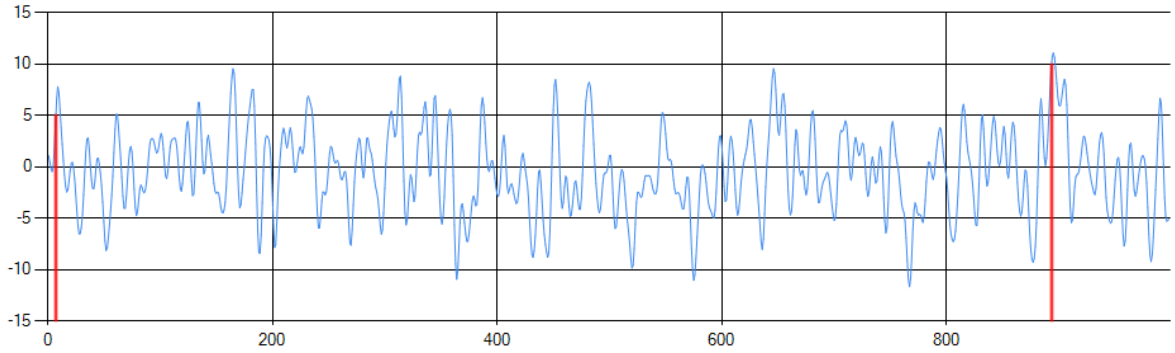


Figure 26

A study shown in *Mathematical Analysis of Random Noise* (Rice) give a zero-crossing frequency as

$$f_0 = 2\left[\frac{1}{3}\frac{f_b^3 - f_a^3}{f_b - f_a}\right]^{\frac{1}{2}} \#(3)$$

With f_b and f_a high cut and low cut frequencies. In the case of an ideal CRRC bandpass filter the zero frequency is near 3 x f_c. Noise cross zero in both ascending and descending ways so the number of triggers on zero would be half the number of crossing. With a shaper central frequency of 5.3 MHz (shaping time of 30 ns), the number of triggers per seconds with a threshold at 0σ is N_0 = 1,5 x f_c = 7,95 M.

From P. Da Silva, «Élaboration d'un banc de tests pour l'électronique front-end du détecteur de particules MICROMEGAS pour l'expérience COMPASS.» (2000) it is known that the number of triggers on a threshold x during a time interval T is

$$N = N_0 \times \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \#(4)$$

With N_0 = T x f_0 and μ the pedestal position. Supposing that this crossing number is a Poisson process, the probability to have k occurrences within T is as

$$P(k) = \frac{N^k e^{-N}}{k!} \#(5)$$

Probability of having 0 events within T is

$$P(0) = e^{-N} \#(6)$$

So the probability of having at least one event during T is

$$P(k \geq 1) = 1 - P(0) \#(7)$$

$$P(k \geq 1) = 1 - e^{-N_0 \times \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)} \#(8)$$

The noise trigger probability on a certain threshold within T is then

$$P(x) = 1 - e^{-N_0 \times \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)} \#(9)$$

This is valid reasoning only in the case of a Gaussian distribution of noise. This study allows to plot the trigger efficiencies on pedestal for various acquisition window lengths (10 µs, 100 µs, 1 ms).

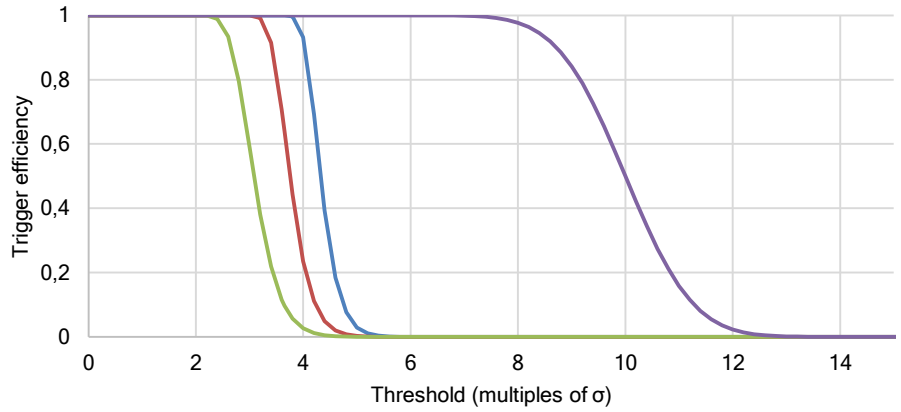


Figure 27 – Trigger efficiency on pedestal noise depending on threshold value for an acquisition window of 10 µs (green) ; 100 µs (red) and 1 ms (blue). Purple : S-curve of a signal with μ = 10 and **σ** = 1 (SNR = 10). The pedestal position is 0.

Hence having a wider acquisition window will shift the trigger efficiency along the x-axis in the case of the pedestal trigger efficiency acquisition.