PSIROC

SOFTWARE & EVALUATION BOARD USER GUIDE

Version: 16 May 2025

PSIROC is a 64-channel front-end ASIC designed to read-out low gain detectors such as PIN diodes, silicon strips, GEMs, LGADs, etc. PSIROC allows triggering down to 0.5 fC and provides dual-gain energy measurement with excellent Signal-to-Noise Ratio on the high gain (ENC down to 400 e- RMS) and large dynamic range on the low gain.

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

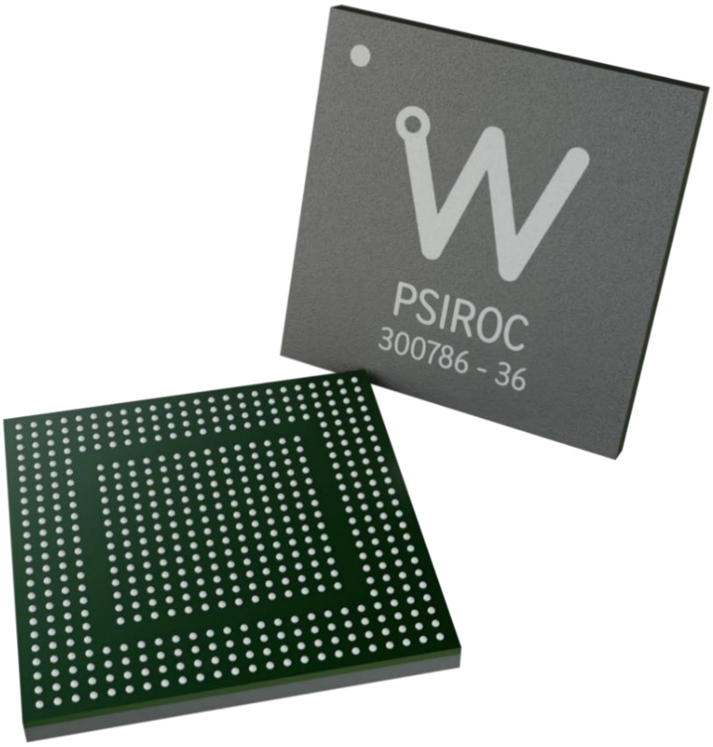


Table of content

Table of figures

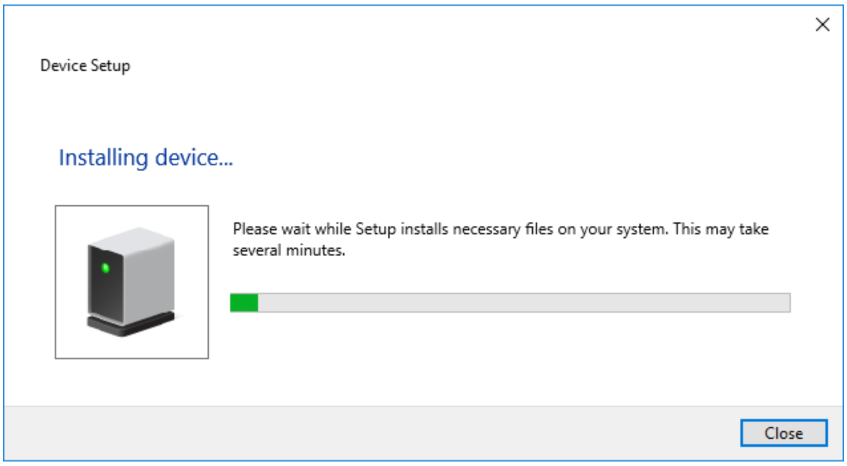
1 Driver and interface installation

1.1 Prerequisites

The use of this evaluation board requires the following:

- A computer (Microsoft Windows 10 or later) with USB connection
- A USB-A to mini-USB cable
- A positive output power supply (delivering 5V and 2A)

The first time a Weeroc evaluation board is plugged, the following message should prompt:

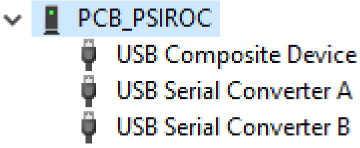


Device Setup

Installing device...

Please wait while Setup installs necessary files on your system. This may take several minutes.

In order to verify that the drivers are correctly installed, go to the control panel under the "Devices and Printers" window. the PCB_PSIROC device should appear as follow:



PCB_PSIROC
- USB Composite Device
- USB Serial Converter A
- USB Serial Converter B

1.2 Installation guide

Before running the software for the first time, please verify that the evaluation board is correctly identified in the "Devices and Printers" window under the control panel. The latest release of the PSIROC user interface can be found in the Weeroc download center on the website http://www.weeroc.com. If assistance is required, please contact the Weeroc support by sending an e-mail at support@weeroc.com.

2 Evaluation board presentation

The evaluation board has mainly been developed to allow characterization and debug of the PSIROC ASIC. 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 can be provided, users can modify anything they need to fit their own requirements.

- This board provides easy access to each PSIROC pin, as all analog and digital pins are connected to through-hole test points.
- Many test points are also connected to the FPGA output signals.
- 1 analogue buffer provides ASIC's analogue probe output on an SMA connectors (Analog_Probe).
- The board needs to be externally powered by a 5V/2A supply.

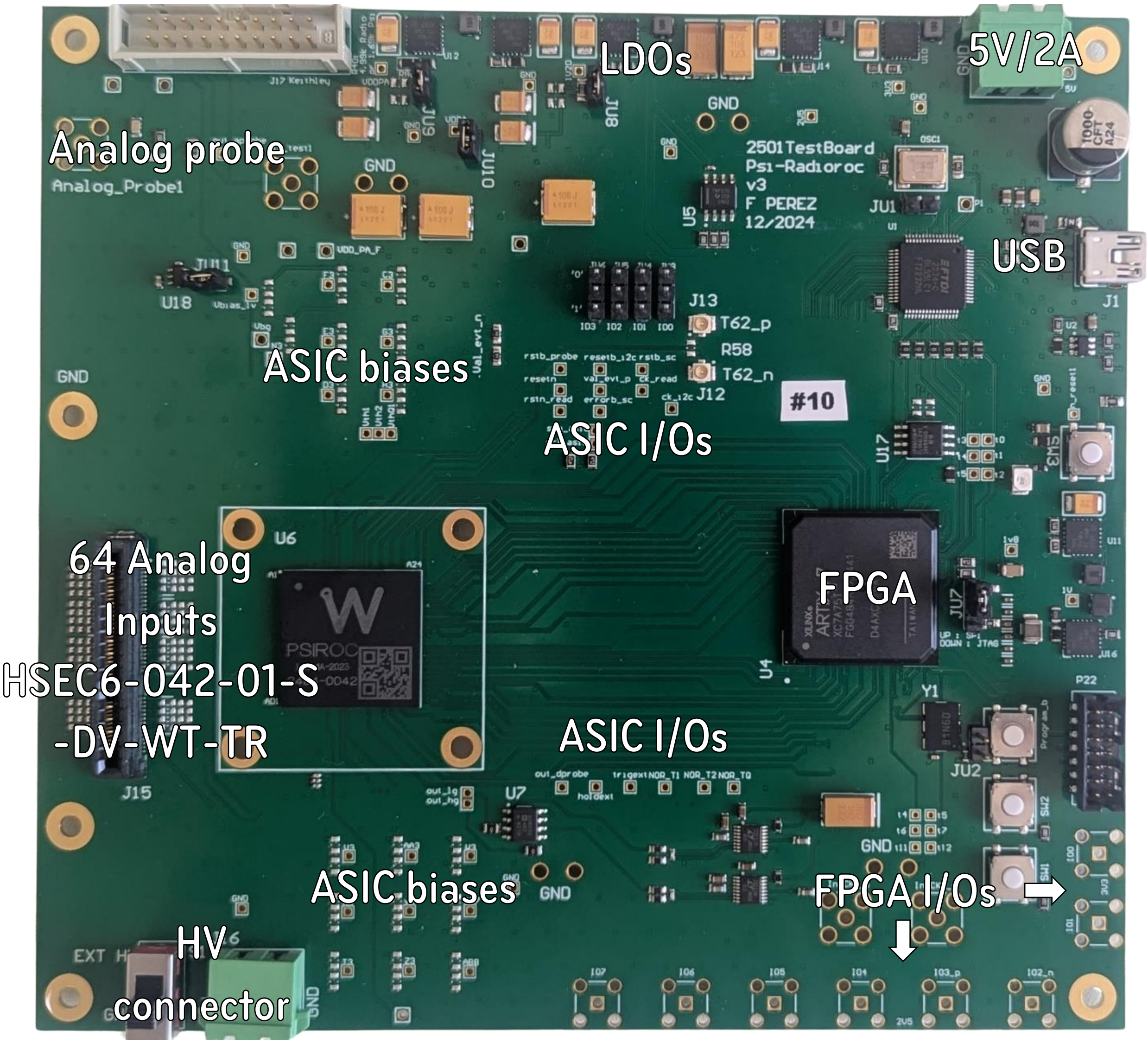


Figure 1 – PSIROC evaluation board

3 User Interface description

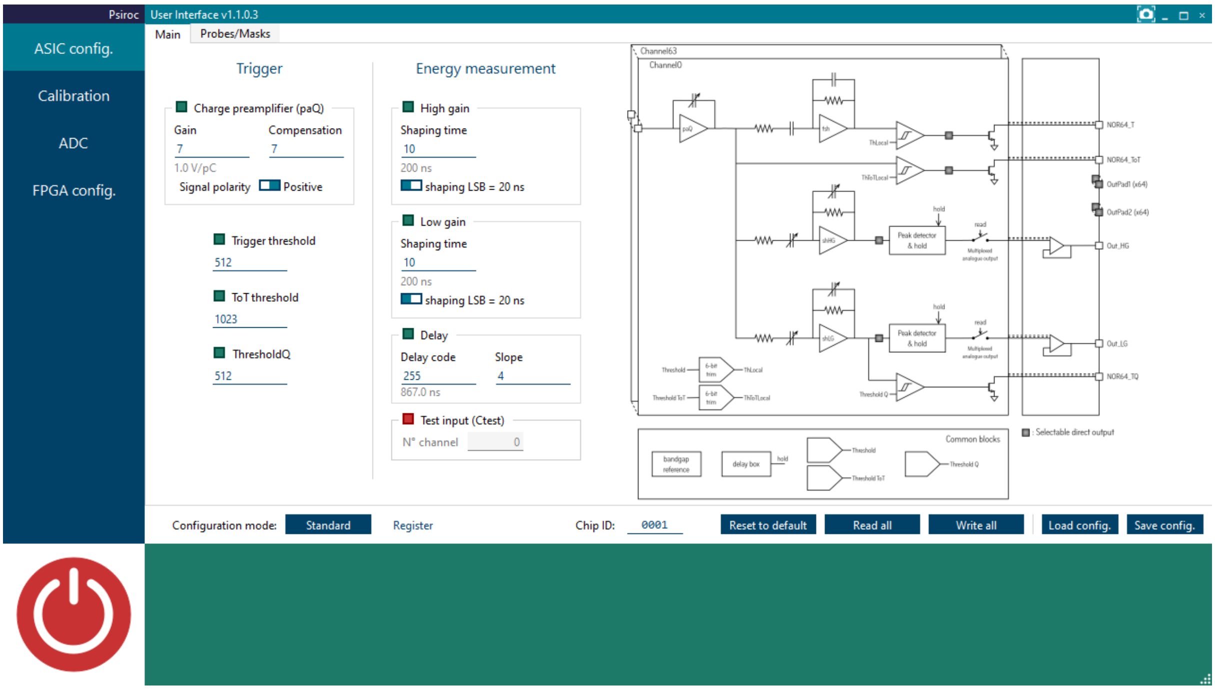
The software has been written in python language. The source code is available on demand in order to help users to comprehend the functionality of the software. This is especially useful if users aim to develop their own data acquisition (DAQ) system.

To start the evaluation board, users need to:

1. Connect the USB cable from the evaluation board to your computer.
2. Provide power supply to the evaluation board (top-right connector) using an external Power Supply (5V-2A).
3. Start the software and click on the ⏻ round button in the bottom left

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 the software is started, the user should be prompted by the following screen.



Charge preamplifier (paQ)
Gain: 7
Compensation: 7
1.0 V/pC
Signal polarity: Positive

Trigger threshold: 512
ToT threshold: 1023
ThresholdQ: 512

High gain
Shaping time: 10
200 ns
shaping LSB = 20 ns

Low gain
Shaping time: 10
200 ns
shaping LSB = 20 ns

Delay
Delay code: 255
Slope: 4
867.0 ns

Test input (Ctest)
N° channel: 0

Configuration mode: Standard / Register
Chip ID: 0001

Figure 2 – Software main page.

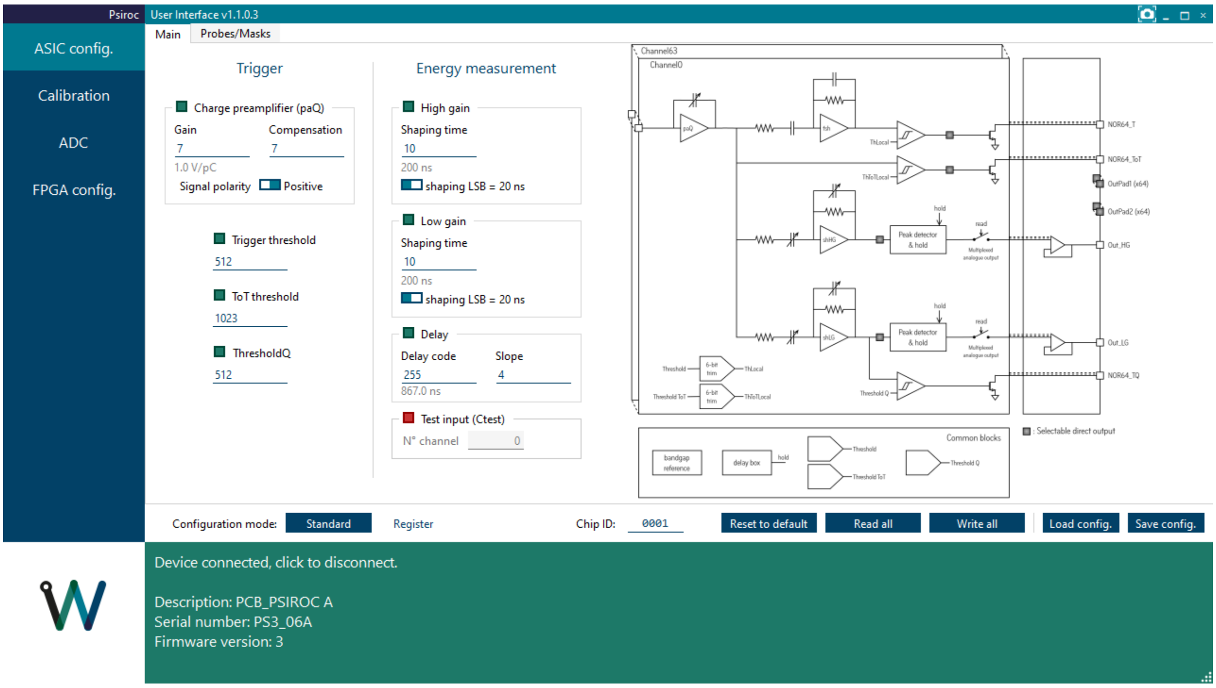
When the software is running and the ⏻ button clicked, no error should occur, meaning that the installation has been done successfully and all the drivers have been found.

While this user guide will help the user to with 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 bottom part will be filled with information on the object being hovered.

The firmware version is automatically detected by the software and is written to the interface. The expected firmware version should correspond to the last number in the software version. The latest firmware version is 3, while the software version is 1.1.0.3.

When the software is started, the user should be prompted by the following screen.

If the evaluation board is plugged and powered, the USB connection is automatically performed and the prompted screen would be as shown on Figure 3.



Charge preamplifier (paQ)
Gain: 7
Compensation: 7
1.0 V/pC
Signal polarity: Positive

Trigger threshold: 512
ToT threshold: 1023
ThresholdQ: 512

High gain
Shaping time: 10
200 ns
shaping LSB = 20 ns

Low gain
Shaping time: 10
200 ns
shaping LSB = 20 ns

Delay
Delay code: 255
Slope: 4
867.0 ns

Test input (Ctest)
N° channel: 0

Configuration mode: Standard
Chip ID: 0001

Device connected, click to disconnect.

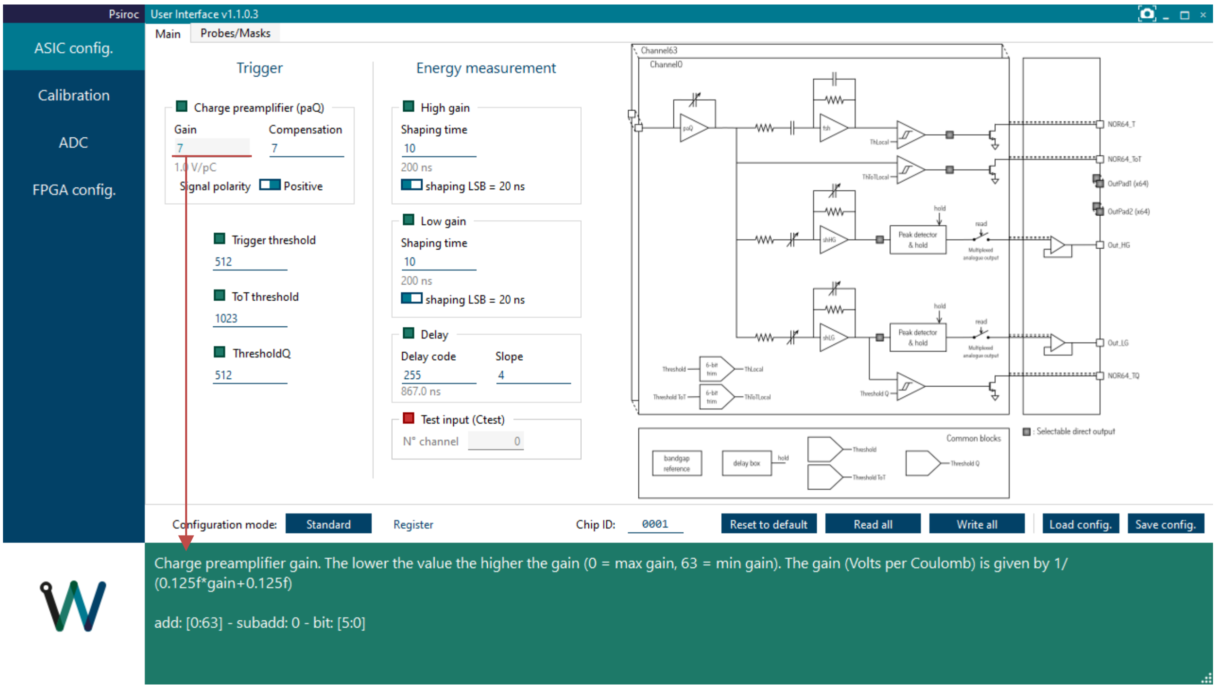
Description: PCB_PSIROC A
Serial number: PS3_06A
Firmware version: 3

Figure 3 – Connection status

The button becomes the Weeroc logo and information about the board are displayed in the embedded help. To show that information at any moment, just hover the Weeroc logo in the lower left screen.

3.1 ASIC configuration (Commonly referred as slow control/I^{2}C configuration)

The home page of the software is the ASIC configuration tab. In this tab users can modify the ASIC parameters to adjust its behavior to their needs (pre-amplifier gain, shapers shaping time, trigger threshold, etc.). The embedded help is particularly handy here because the description of parameters is given to the user when hovering the corresponding control as shown on Figure 4.



Trigger

☑ Charge preamplifier (paQ)
Gain: 7
Compensation: 7
1.0 V/pC
Signal polarity: Positive

☑ Trigger threshold: 512

☑ ToT threshold: 1023

☑ ThresholdQ: 512

Energy measurement

☑ High gain
Shaping time: 10
200 ns
shaping LSB = 20 ns

☑ Low gain
Shaping time: 10
200 ns
shaping LSB = 20 ns

☑ Delay
Delay code: 255
Slope: 4
867.0 ns

☒ Test input (Ctest)
N° channel: 0

Configuration mode: Standard | Register

Chip ID: 0001

Charge preamplifier gain. The lower the value the higher the gain (0 = max gain, 63 = min gain). The gain (Volts per Coulomb) is given by 1/(0.125f*gain+0.125f)

add: [0:63] - subadd: 0 - bit: [5:0]

Figure 4 – Embedded help for ASIC configuration.

On the left side of the Main page the parameters are organized between "Trigger" and "Energy measurement" and on the right side the schematic of the ASIC is displayed for quick reference. More specific parameters can be found on other tabs for calibration purposes or signal probing for instance.

ASIC configurations can be saved and loaded back using the "Load config." And "Save config." Buttons in the bottom right. Alternatively, the configuration files can be drag and dropped in the UI. To quickly reset the I^{2}C registers to their default state the user can click the "Reset to default" button without having to manually load the default configuration file.

The "Read all" and "Write all" are here for debug purposes as in nominal operations the displayed I^{2}C should always match the ASIC registers. The parameters are sent to the ASIC and read back for validation at each interaction (If the board is plugged and powered).

TextBox being edited:

8

TextBox after validation (Enter/Return pressed or TextBox lost focus):

8

The displayed data is the one read back from the ASIC, not the one inputted by the user, although they should match. It shall be noted that while the board isn't connected the configuration can be edited freely and the edited configuration will be loaded in the ASIC when connecting the board.

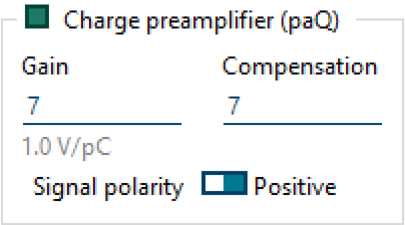
There are two configuration modes, either the "Standard" one where only the most relevant parameters are exposed to the user and layout in a user-friendly way, or the "Register" mode where the user can modify directly the binary content of every

I^2C register in the ASIC. To switch from one to the other the user has to click the bottom left buttons on the tab. In this mode the embedded help shows the content of the data in the register.

3.1.1 Main parameters

The standard mode slow control page brings together many useful ASIC parameters.

3.1.1.1 Charge preamplifier



Charge preamplifier (paQ)

Gain: 7

Compensation: 7

1.0 V/pC

Signal polarity: Positive

Figure 5 – Charge preamplifier parameters

Gain and compensation can be changed for the charge preamplifier. The output of the pre-amplifier is a voltage step with a slow return to baseline. The pre-amplifier has a variable gain adjustment accessible through Slow Control bits:

- *paGain* (Address: 0-63; Subadd: 0; Bits: 5-0). Cf: 125 fF – 8 pF.

The Slow Control bits will change the value of feedback capacitor Cf of the pre-amplifier. Compensation must also be set for pre-amplifier stability. At first order the *paComp* value can be kept the same as the *paGain*, tuning it further is at the user's discretion.

- *paComp* (Address: 0-63; Subadd: 1; Bits: 5-0). Ccomp: 0 F – 7.875 pF.

Depending on the polarity of the input signal the corresponding slow control bit must be set:

- *polarity* (Address: 0-63; Subadd: 0; Bits: 6). 0: negative input – 1: positive input.

The feedback is done either with a feedback OTA or a 4 MOhm resistor. The OTA is mainly used to bias the output higher than the input. The input is biased at V_{th} of the input Nmos (~500 mV) and in the case of positive input signal, the output is negative hence having a higher output DC biasing (> 2 V if VDD_PA = 2.5 V) gives more dynamic range. When used for negative input signal the pre-amplifier output is inverted and series noise is added to the signal because of the inverter resistors. This is more noticeable for larger C_f values (lower gains) because for large gains the input Nmos thermal noise is the major noise contributor (see Figure 9).

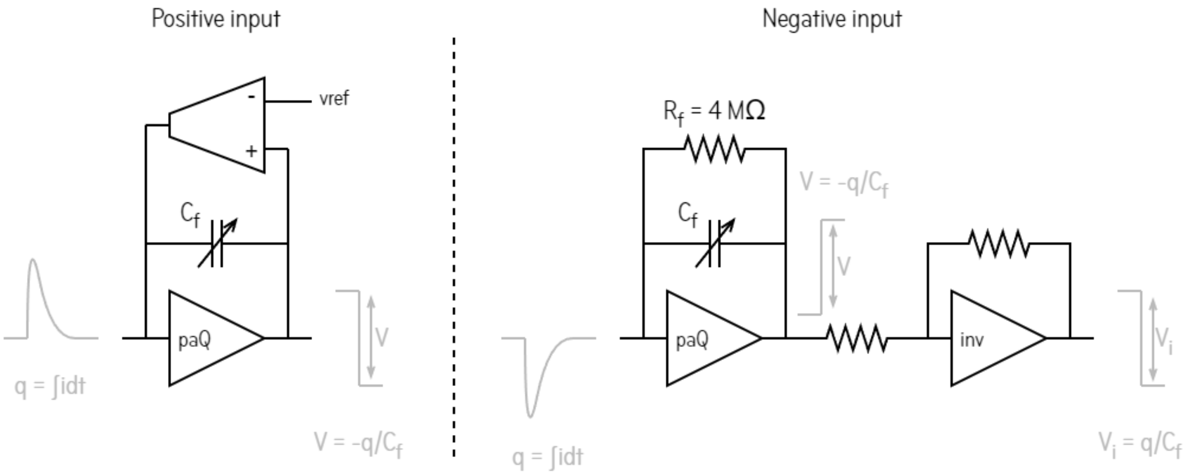
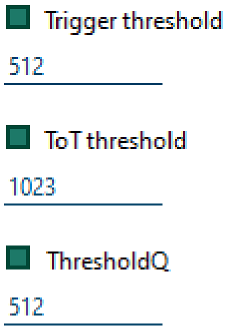


Figure 6 – Pre-amplifier block diagram for positive (left) and negative (right) input signals.

3.1.1.2 Thresholds



☑ Trigger threshold

512

☑ ToT threshold

1023

☑ ThresholdQ

512

Figure 7 – Thresholds parameters

The first discriminator is preceded by a fast shaper to significantly increase Signal to Noise Ratio. Its threshold can be adjusted by the *DACT* slow control bits.

- (*DACT*, Address: 65; Subadd: 2; Bits: 1-0 & Subadd: 1; Bits: 7-0). Voltage span: 390 mV ~ 1,15 V. Step = 0.75 mV

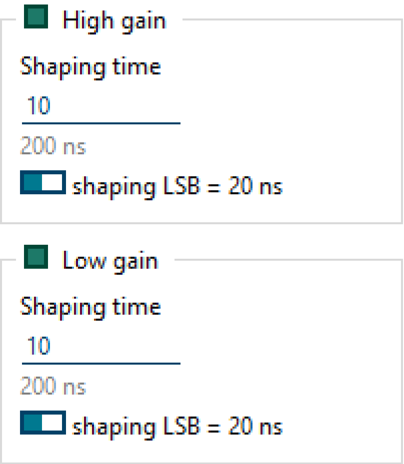
The second discriminator is plugged directly at paQ output and can be used for ToT measurements. Because the paQ discharge is very slow the ToT can be as large as few hundreds of microseconds for signal charges of a few tens of pico-coulombs. Its threshold is adjusted via *DACTOT*.

- (*DACTOT*, Address: 65; Subadd: 3; Bits: 3-0 & Subadd: 2; Bits: 7-2). Voltage span: 390 mV ~ 1,15 V. Step = 0.75 mV

A third discriminator at the Low Gain shaper output can be used to generate a "charge" Trigger (TQ). The threshold can be set through the following Slow Control bits:

- *DACQ* (Address: 65; Subadd: 4; Bits: 5-0 & Subadd: 3; Bits: 7-4). Voltage span: 75 mV ~ 1,2 V. Step = 1 mV

3.1.1.3 Shapers



■ High gain

Shaping time: 10

200 ns

shaping LSB = 20 ns

■ Low gain

Shaping time: 10

200 ns

shaping LSB = 20 ns

Figure 8 – Shapers parameters

This ASIC also embeds two shapers (High Gain & Low Gain). Both shapers have a variable shaping time. The shaping time for both shapers can be adjusted using the Slow Control bits *tauHG* (Address: 0-63; Subadd: 3; Bits: 7-4) for the High Gain shaper and *tauLG* (Address: 0-63; Subadd: 3; Bits: 3-0) for the Low Gain shaper. The Slow Control bits adjusting the shaping time are:

- *tauHG* (Address: 0-63; Subadd: 3; Bits: 7-4). Shaping time span: 20 ns ~300 ns. Step = 20 ns
- *tauLG* (Address: 0-63; Subadd: 3; Bits: 3-0). Shaping time span: 20 ns ~300 ns. Step = 20 ns

If the slow control bits *slowShapingLG* (Address: 0-63; subadd: 7; bit: 7) or *slowShapingHG* (Address: 0-63; subadd: 7; bit: 6) are set to '1', the adjustment becomes 200 ns ~ 3 µs with a step of 200 ns.

Switching this bit actually multiplies the shaper resistors by 10 so it will add a copious amount of series noise (from 14.7 kΩ ⟶ 15.6 nV/√Hz to 147 kΩ ⟶ 49.3 nV/√Hz). Albeit it is second stage noise it has a very noticeable effect as shown on Figure 9. Thus, unless long integration time are needed and ENC is acceptable it is better to stick with LSB = 20 ns.

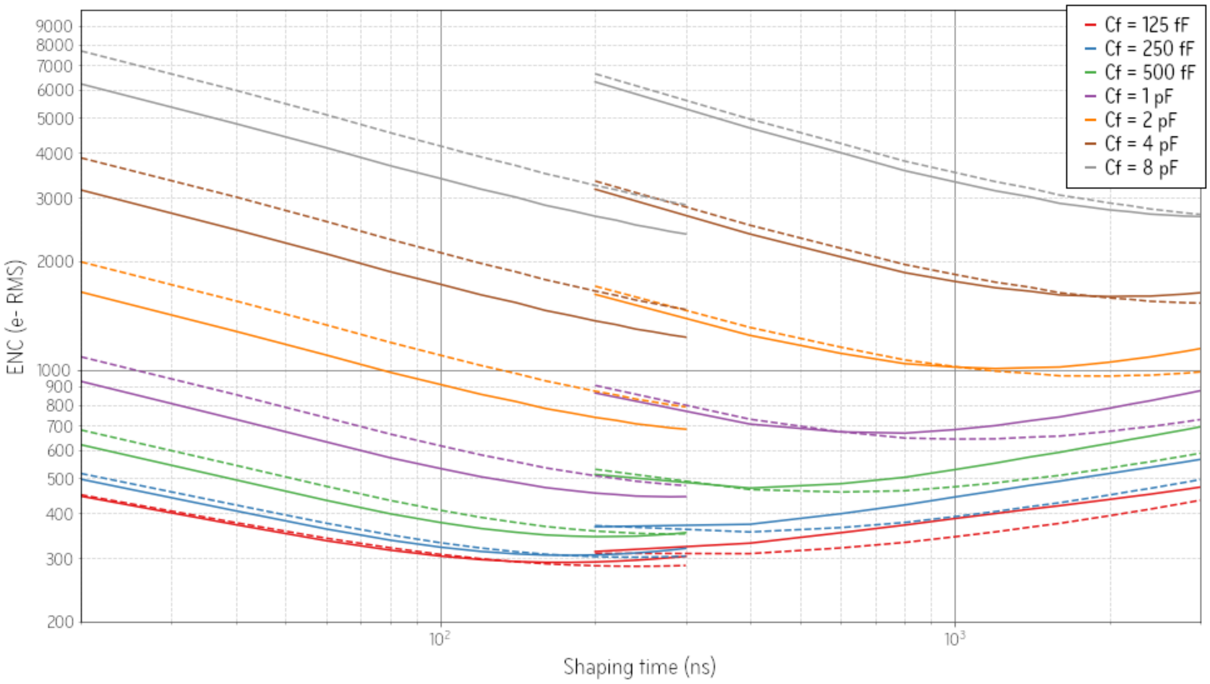
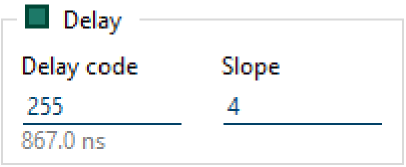


Figure 9 – ENC as a function of shaping time with Cd = 20 pF (simulation results). Solid line: positive input; dotted line: negative input. The discontinuity is due to the shaping time LSB toggling from 20 ns to 200 ns.

Each of the shaper output will be sent to a peak detector cell which will be connected to an analog multiplexer for external amplitude (charge) measurements.

3.1.1.4 Delay box



☑ Delay

Delay code: 255

867.0 ns

Slope: 4

Figure 10 – Delay parameters

A delay cell is used to generate the "hold" signal (delay started by the acquisition trigger, see Figure 11) latching the output value of the peak detector before sending it to the analog multiplexer output for digitization.

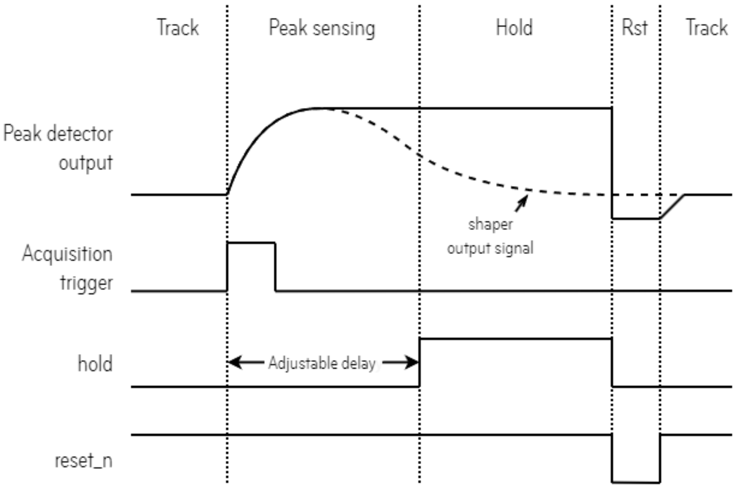


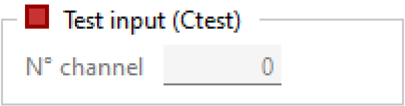
Figure 11 – Illustration of peak detection in Psiroc.

This delay can be configured using this option:

- Delay(8-bit): Delay span: 2 ns~2.6 µs. Step size depends on Delay Slope Trimming
- Delay Slope Trimming (4-bit): Range = 60 ns (0) - 2.6 µs (15).

It is possible to toggle the hold signal by using the external pin (AC9, Active high) instead of relying on the internal delay box.

3.1.1.5 Test input (Ctest)



🟥 Test input (Ctest)

N° channel 0

Figure 12 – Test input parameters.

It is possible to enable the test input for a specified channel. The input signal will be sent through a 1.5 pF injection capacitor with 50 Ohm impedance adaptation on the specified channel. On Figure 13 is an example schematic of the Ctest injection in channel 1.

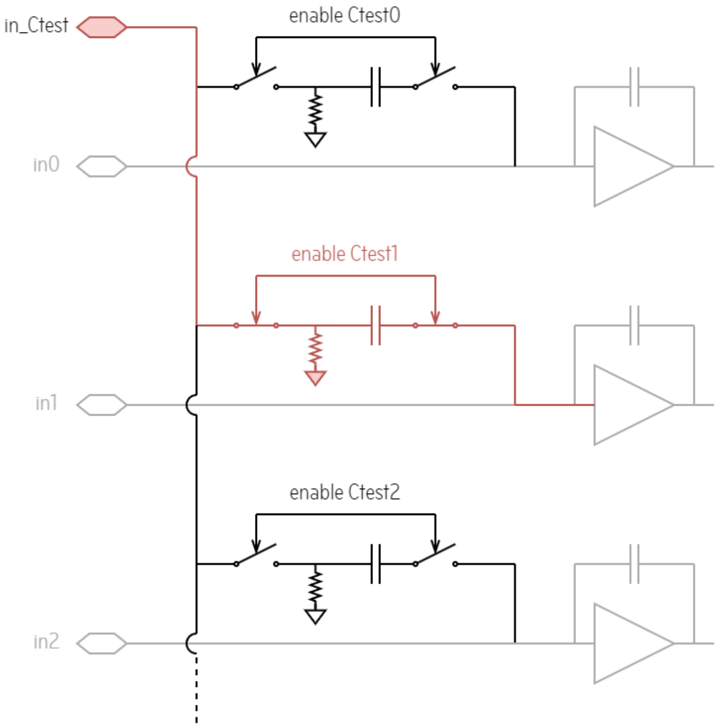
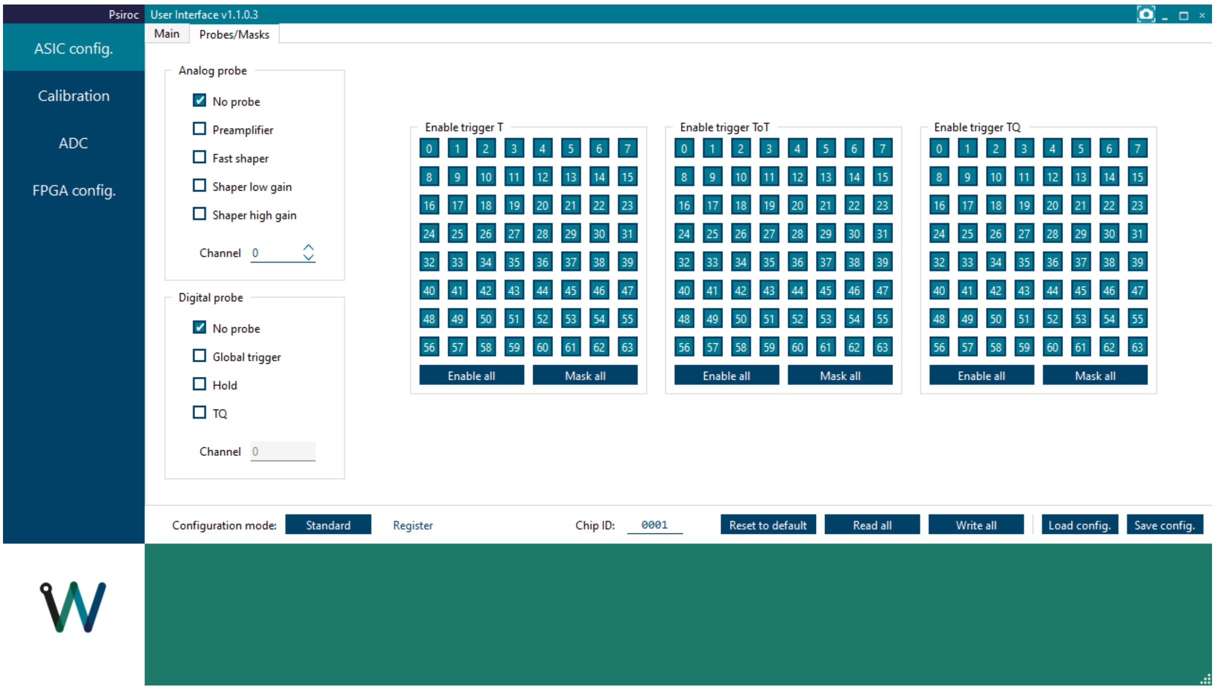


Figure 13 – Schematic of Ctest injection in channel 1.

3.1.2 *Probes/Masks*

This tab allows to enable/mask channel-wise triggers for T1, T2 and TQ.



Analog probe

☑ No probe
☐ Preamplifier
☐ Fast shaper
☐ Shaper low gain
☐ Shaper high gain

Channel 0

Digital probe

☑ No probe
☐ Global trigger
☐ Hold
☐ TQ

Channel 0

Configuration mode: Standard

Chip ID: 0001

Figure 14 – Slow control standard mode – Probes/Masks

An internal ASIC probing multiplexer allows to monitor each channel pre-amp and shaper output signals and to display them on the oscilloscope. User can choose to monitor the output of the low gain and high gain preamplifier output as well as the low gain and high gain shaper output for a specified channel.

In typical ASIC operation, it is advised to turn off the probe register ("No probe" checked) to avoid performance degradation.

3.1.3 Register mode

By toggling the "Configuration Mode" switch in the bottom-left corner, the interface will change into "Register Mode" where each I^{2}C register is directly accessible. The large textbox on the right displays the hexadecimal values of all the registers in the ASIC. On the right side of the register tab the full register content is displayed by address number in a compact hexadecimal way. Sub-addresses are organized by ascending order from left to right e.g. for address 0:

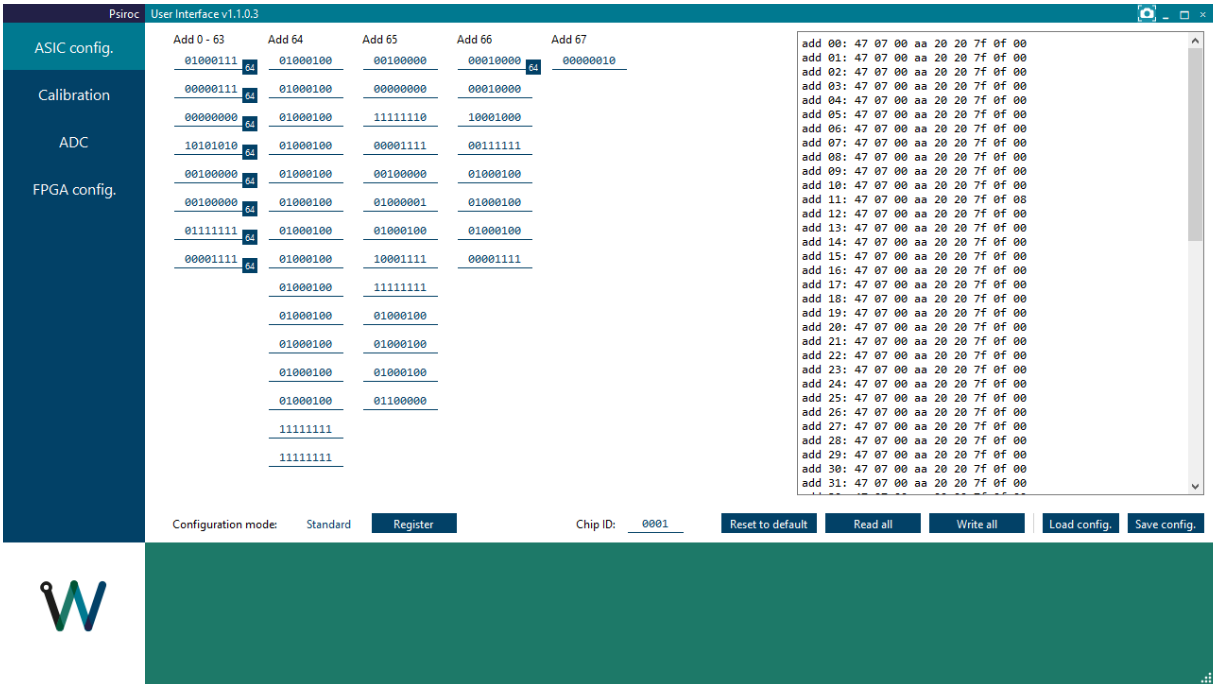
```
add 00: 48 08 00 aa 20 20 7f 0f 00
```

The sub-addresses (refer to datasheet for sub-adresses values) and their respective data are:

- 0 : 0x48
- 1 : 0x08
- 2 : 0x00
- 3 : 0xaa
- 4 : 0x20
- 5 : 0x20
- 6 : 0x7f
- 7 : 0x0f
- 66 : 0x00

The sub-address 66 isn't part of the editable textboxes because this register sub-address is reserved for probes and only one bit per probe (analog and digital) of all registers should be '1' to avoid contentions. Having '1' on more than one bit would shunt amplifiers outputs (analog probe) or buffers outputs (digital probe) in the ASIC. For probe operations it is recommended to use the dedicated tab in the standard configuration mode.

Refer to the datasheet for a description of the register parameters and addresses.



Add 0 - 63	Add 64	Add 65	Add 66	Add 67
01000111	01000100	00100000	00010000	00000010
00000111	01000100	00000000	00010000	
00000000	01000100	11111110	10001000	
10101010	01000100	00001111	00111111	
00100000	01000100	00100000	01000100	
00100000	01000100	01000001	01000100	
01111111	01000100	01000100	01000100	
00001111	01000100	10001111	00001111	
	01000100	11111111		
	01000100	01000100		
	01000100	01000100		
	01000100	01000100		
	01000100	01100000		
	11111111			
	11111111			

```
add 00: 47 07 00 aa 20 20 7f 0f 00
add 01: 47 07 00 aa 20 20 7f 0f 00
add 02: 47 07 00 aa 20 20 7f 0f 00
add 03: 47 07 00 aa 20 20 7f 0f 00
add 04: 47 07 00 aa 20 20 7f 0f 00
add 05: 47 07 00 aa 20 20 7f 0f 00
add 06: 47 07 00 aa 20 20 7f 0f 00
add 07: 47 07 00 aa 20 20 7f 0f 00
add 08: 47 07 00 aa 20 20 7f 0f 00
add 09: 47 07 00 aa 20 20 7f 0f 00
add 10: 47 07 00 aa 20 20 7f 0f 00
add 11: 47 07 00 aa 20 20 7f 0f 08
add 12: 47 07 00 aa 20 20 7f 0f 00
add 13: 47 07 00 aa 20 20 7f 0f 00
add 14: 47 07 00 aa 20 20 7f 0f 00
add 15: 47 07 00 aa 20 20 7f 0f 00
add 16: 47 07 00 aa 20 20 7f 0f 00
add 17: 47 07 00 aa 20 20 7f 0f 00
add 18: 47 07 00 aa 20 20 7f 0f 00
add 19: 47 07 00 aa 20 20 7f 0f 00
add 20: 47 07 00 aa 20 20 7f 0f 00
add 21: 47 07 00 aa 20 20 7f 0f 00
add 22: 47 07 00 aa 20 20 7f 0f 00
add 23: 47 07 00 aa 20 20 7f 0f 00
add 24: 47 07 00 aa 20 20 7f 0f 00
add 25: 47 07 00 aa 20 20 7f 0f 00
add 26: 47 07 00 aa 20 20 7f 0f 00
add 27: 47 07 00 aa 20 20 7f 0f 00
add 28: 47 07 00 aa 20 20 7f 0f 00
add 29: 47 07 00 aa 20 20 7f 0f 00
add 30: 47 07 00 aa 20 20 7f 0f 00
add 31: 47 07 00 aa 20 20 7f 0f 00
```

Configuration mode: Standard

Chip ID: 0001

Figure 15 – Direct I^{2}C register editing.

3.2 Calibration

3.2.1 S-curves

The first tab in this page is called "S-curves" and its purpose is to perform a trigger efficiency test (named S-curves). It is done by scanning through several values of threshold and measuring if a trigger happens. The acquisitions are windowed by a clock. If a trigger happens outside the clock window, it will not be taken into account.

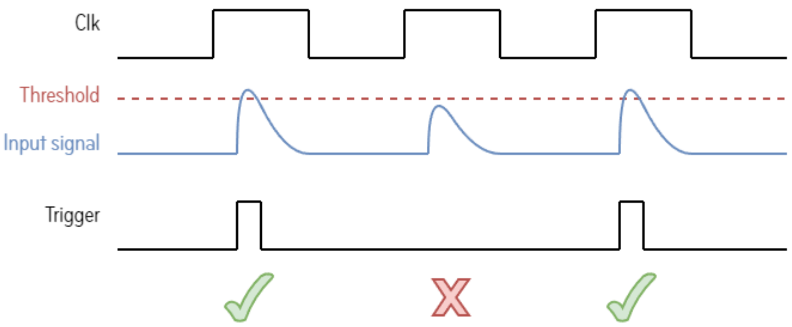
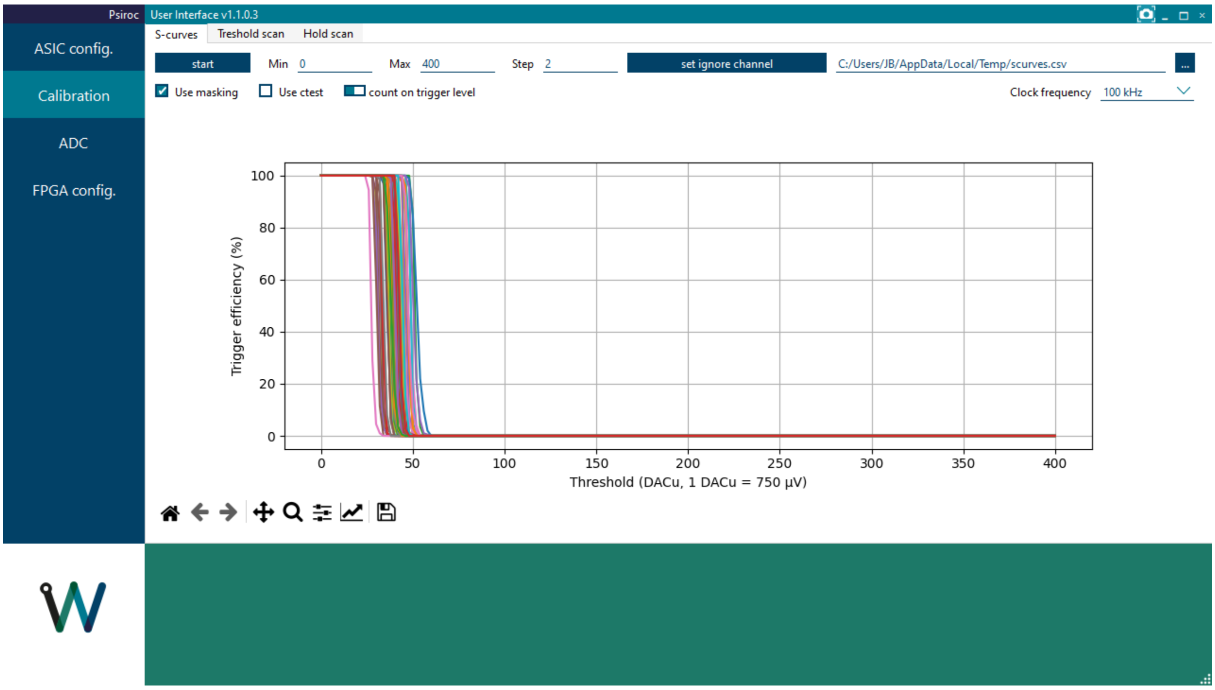


Figure 16 - S-curves acquisition chronogram

When the threshold is way below the signal amplitude, the trigger efficiency will be 100%. With a threshold way over the signal, trigger efficiency drops to 0%. On Figure 16, the threshold is set near the tip of the signal so the ASIC will have a ~50% chance of triggering depending on the noise. Following the Figure 16, the ASIC would trigger 2 times in 3 acquisition windows, resulting with a 67 % trigger efficiency.

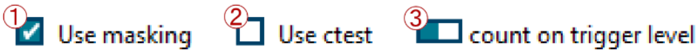
The acquisition is started by clicking on the "start" button. With default settings it will give the result shown on Figure 17.



Min 0
Max 400
Step 2
C:/Users/JB/AppData/Local/Temp/scurves.csv
☑ Use masking
☐ Use ctest
☐ count on trigger level
Clock frequency 100 kHz

Figure 17 – S-curves.

There are 3 checkboxes that allow to adjust the behavior of the S-curves measurement as shown on Figure 18.



① ☑ Use masking ② ☐ Use ctest ③ ☐ count on trigger level

Figure 18 – Behaviour checkboxes for S-curves.

By default, all but the measured channel triggers are masked to avoid seeing the effect of crosstalk. This can be changed by unchecking the "Use masking" checkbox ①. With this checkbox unchecked the masking will be kept as it is in the ASIC configuration so masking channels in the ASIC config. Tab and doing this test will not unmask all the channels.

To plot the S-curves using an input signal by checking the "use ctest" checkbox ② the user will need to inject signal through the "in_Ctest" connector. The injected signal must be synchronized with the clock signal available on the FPGA I/Os (default IO0, see FPGA config. tab) to send the signal in each acquisition window as shown on Figure 16. Note that to inject signal in a single channel there is no need to check this box. Checking the box will inject signal in ALL channels by enabling their respective Ctest one by one as measurement goes on. To inject in a single channel with Ctest, enabling it from the ASIC config. main page is enough (or inject in the channel input directly).

Finally, checkbox ③ allows to perform the measurement on trigger level or trigger edge. The difference is only visible when below the pedestal. If the measurement is done on trigger level the trigger efficiency will be 100 % when the threshold is below the pedestal. If done on trigger level the trigger efficiency will drop to 0 when the threshold is below pedestal. The difference is shown on Figure 19.

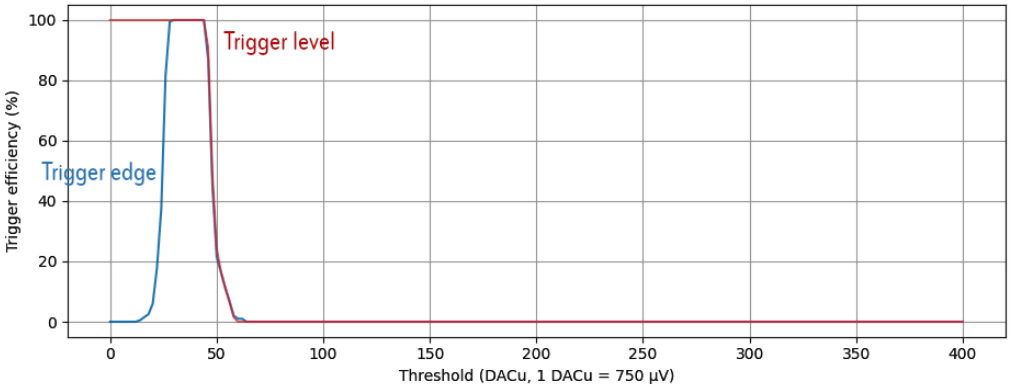


Figure 19 – S-curves on trigger level versus trigger edge on the pedestal.

The "Clock frequency" combo box allows to modify the clock frequency used for the S-curves measurement. Refer to appendix on S-curves page 30 for more information about the effects of the clock frequency. When doing the S-curves with signal injection it can be useful to have a slower clock to avoid signal pile-up. Clock period should be longer than the time it takes for the preamp to be back to baseline.

It is also possible to plot only chosen channels using "set ignore channel". Plotting the S-curve on a single channel will be 64 times faster than doing it on the 64 channels because each channel is measured sequentially.

3.2.2 Threshold scan

The threshold scan is similar to the S-curves measurement but here the total number of triggers happening in a time window is counted and the trigger frequency for each threshold value is calculated. This gives the trigger frequency as a function of the threshold value. The working principle is shown on Figure 20.

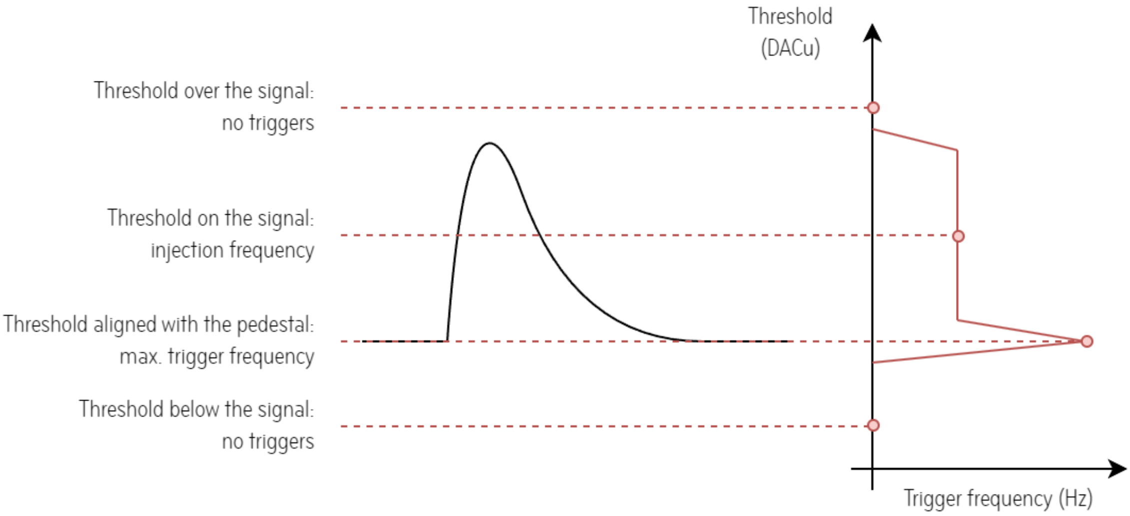
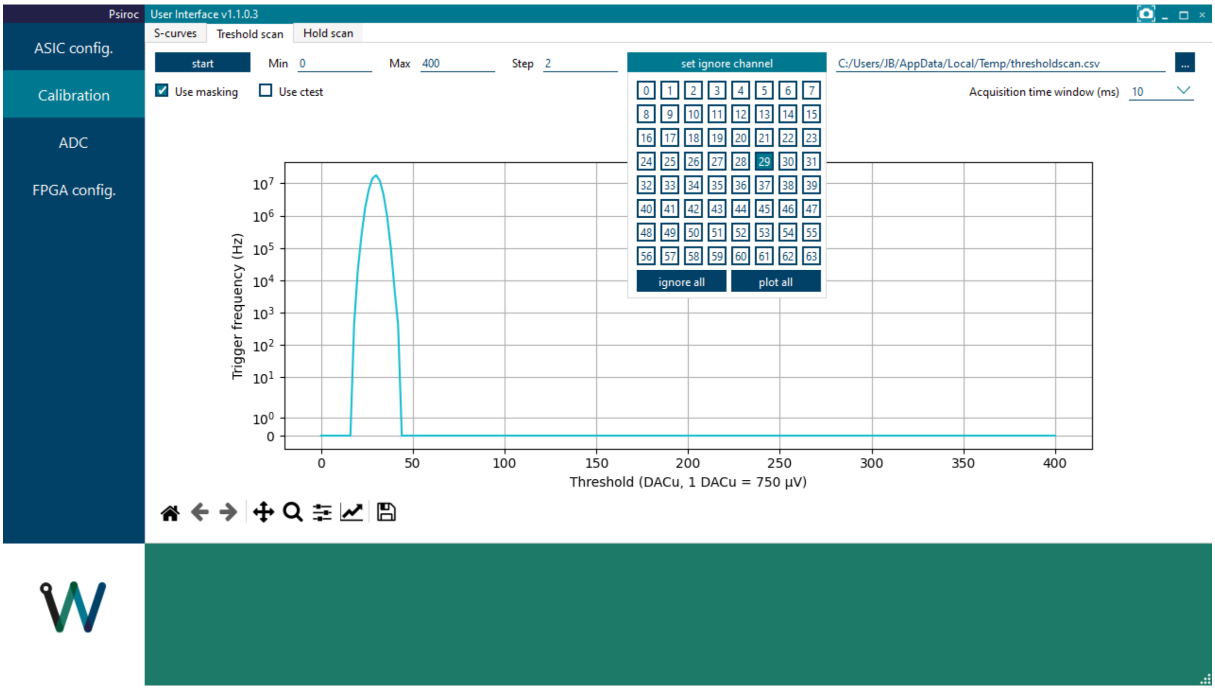


Figure 20 – Threshold scan working principle.

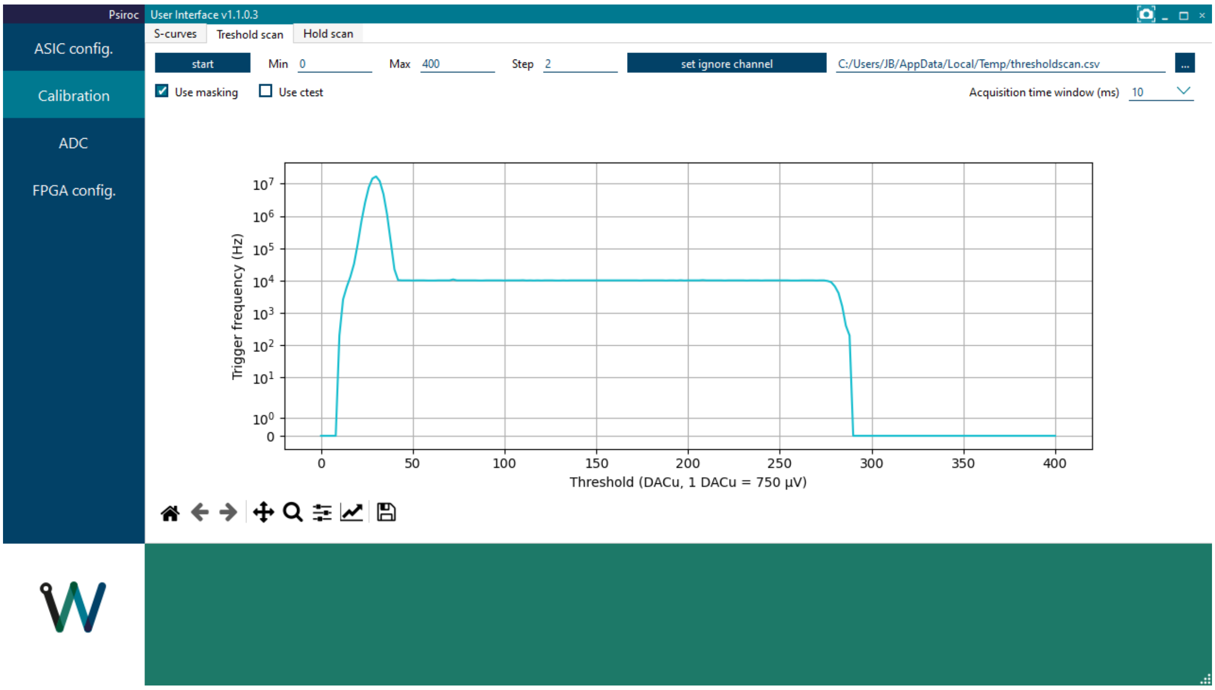


Min 0
Max 400
Step 2
C:/Users/JB/AppData/Local/Temp/thresholdscan.csv
☑ Use masking
☐ Use ctest
Acquisition time window (ms) 10

Figure 21 – Threshold scan ignore channels on the pedestal.

The checkboxes "Use masking" and "Use ctest" behave the same as in the S-curves measurement. The acquisition time window can be modified, there is a trade-off between the acquisition duration and the data statistics (longer window = more statistics).

When injecting a signal, the pedestal noise gaussian is followed by a plateau of the injection's frequency.

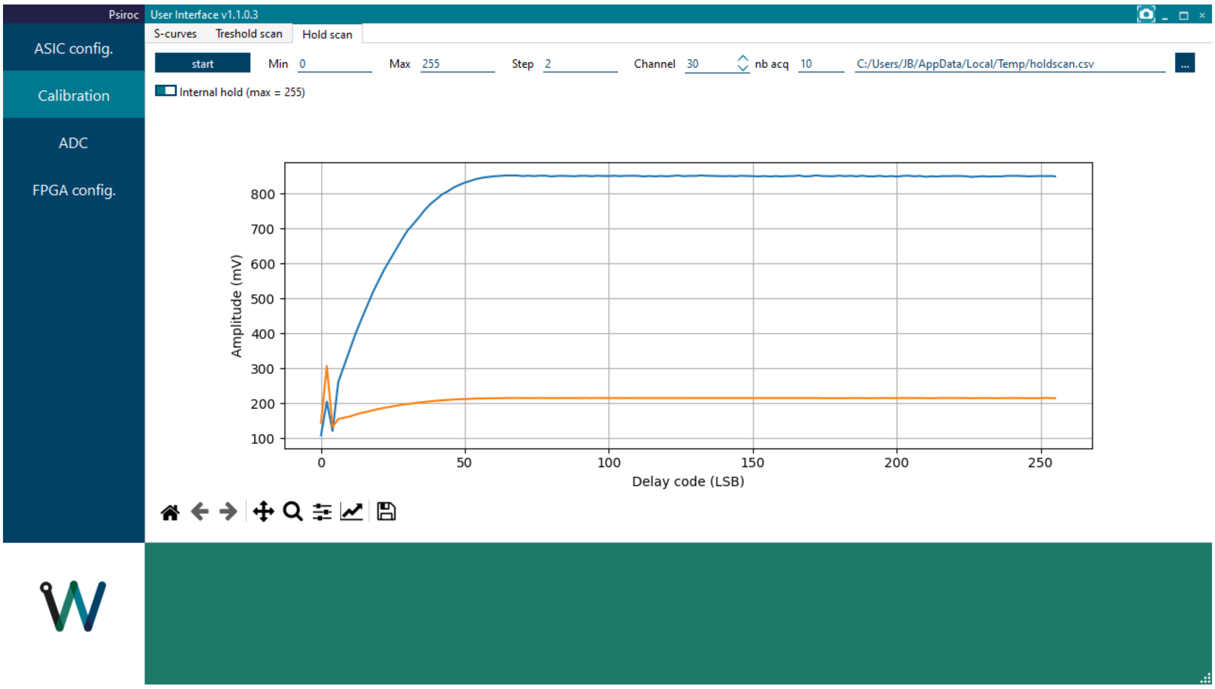


Min 0
Max 400
Step 2
C:/Users/JB/AppData/Local/Temp/thresholdscan.csv
☑ Use masking
☐ Use ctest
Acquisition time window (ms) 10

Figure 22 – Threshold scan with 10 kHz signal injection.

3.2.3 Hold scan

Hold scan is a test done by scanning through the hold delay value (delay between a trigger and the "hold" toggle, see Figure 11). It can be done with the internal delay box to reconstruct the signal at peak detector output as seen by the DAQ (Figure 23) or with an external hold to avoid triggering the peak detector and reconstruct the shaper output signal (Figure 24).Note that the first values for the internal hold plot are too low and thus invalid hence why it gives incoherent results.



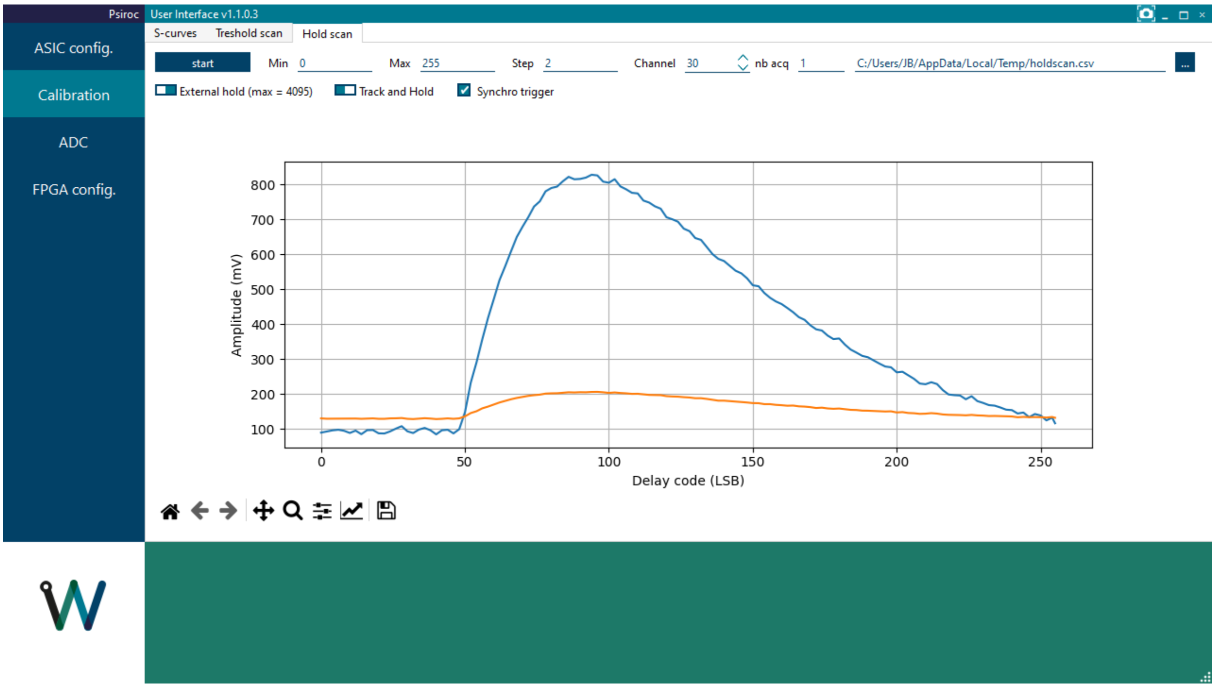
Min 0
Max 255
Step 2
Channel 30
nb acq 10
C:/Users/JB/AppData/Local/Temp/holdscan.csv
☑ Internal hold (max = 255)

Figure 23 – Internal hold (use embedded delay cell)

The number of acquisitions per hold value can be adjusted and the resulting plot will show the average of the signal as seen on Figure 25.

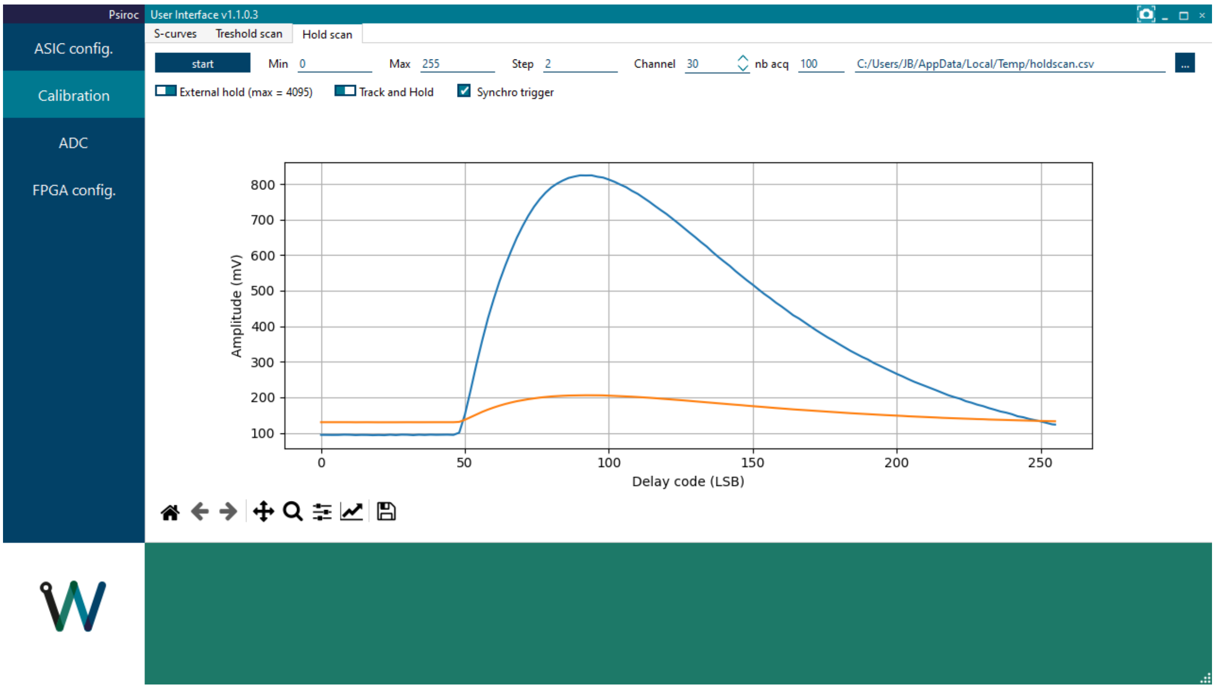
This test is performed to find the ideal hold value for the delay. In the case of using the peak detector the hold must be superior to the peaking time of the signal. If the cell is used as a track and hold the hold must be chosen to be precisely the peaking time to sample the maximal value of the shaper output. When using the internal hold, the LSB shall be calculated using the slow control parameter of the delay slope (adjustable LSB). When using the external hold, the LSB is 5 ns.

When doing the hold scan test with external hold, a synchronization trigger must be sent to the pulser. See the FPGA configuration section to find which I/O to connect to the pulser.


Min 0 Max 255 Step 2 Channel 30 nb acq 1 C:/Users/JB/AppData/Local/Temp/holdscan.csv

External hold (max = 4095) Track and Hold ☑ Synchro trigger

Figure 24 – External hold (hold signal generated by FPGA). 1 acquisition/point

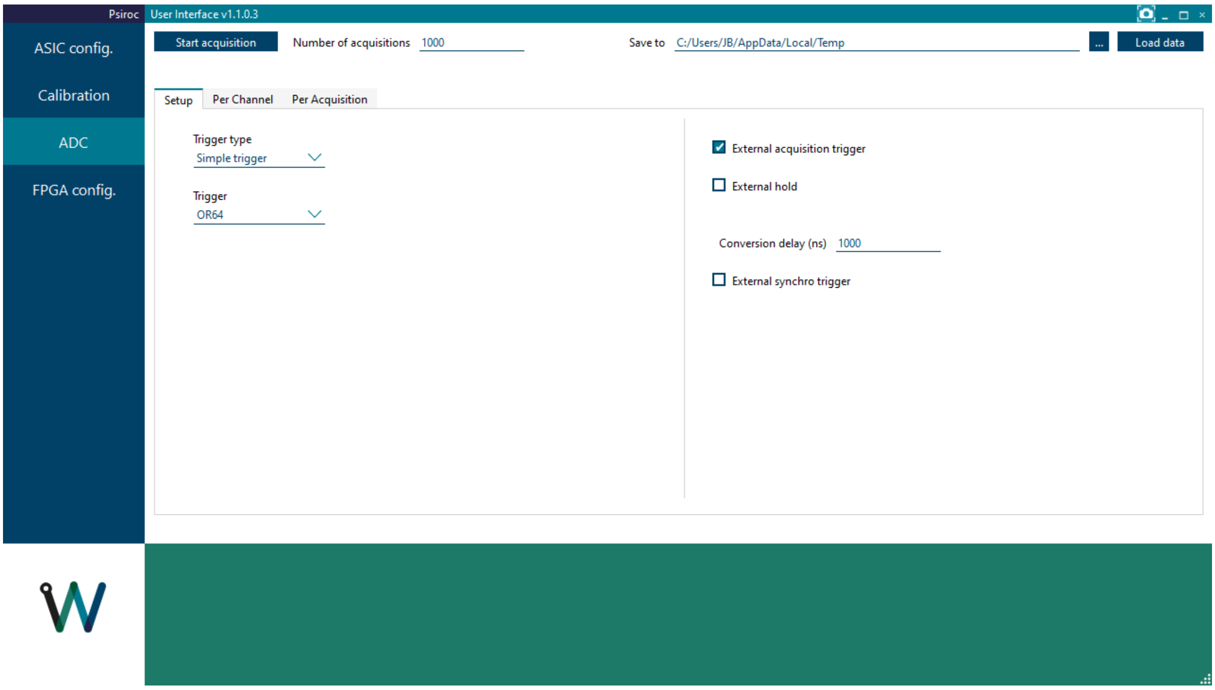

Min 0 Max 255 Step 2 Channel 30 nb acq 100 C:/Users/JB/AppData/Local/Temp/holdscan.csv

External hold (max = 4095) Track and Hold ☑ Synchro trigger

Figure 25 – External hold, 100 acquisitions/point. The signal appears less noisy.

3.3 Data acquisition

The ADC DAQ tab is prompted as shown on Figure 26. On the left panel of the setup tab, the DAQ trigger signal can be chosen to be a single trigger, a coincidence between two selected triggers, or a number of triggers in a time window (see Figure 27).



Number of acquisitions 1000

Save to C:/Users/JB/AppData/Local/Temp

Trigger type: Simple trigger

Trigger: OR64

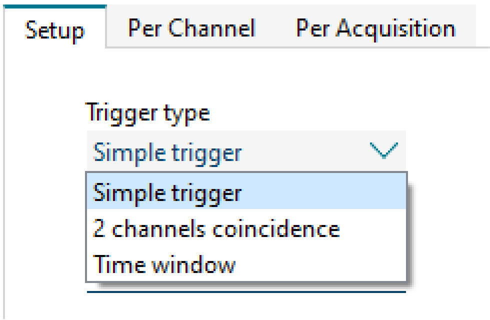
☑ External acquisition trigger

☐ External hold

Conversion delay (ns) 1000

☐ External synchro trigger

Figure 26 – ADC DAQ tab.

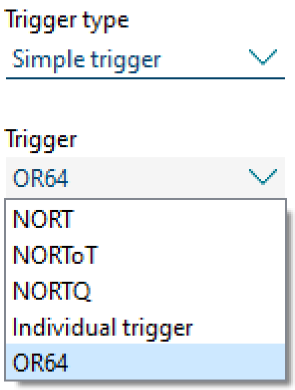


Trigger type: Simple trigger

- Simple trigger
- 2 channels coincidence
- Time window

Figure 27 – Trigger type combo box.

The trigger used to start the DAQ can be of different types. Note that a bug prevents using the NORT trigger. Alternatively, the OR64 trigger does the same thing as it takes the 64 trigger T and the OR is generated inside the FPGA instead of inside the ASIC.

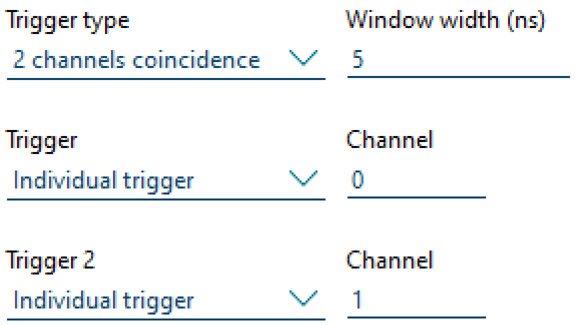


Trigger type
Simple trigger

Trigger
OR64

Figure 28 – Trigger used to start the DAQ.

Figure 29 shows a configuration for the trigger coincidence. Here the coincidence is between channel 0 and channel 1 triggers with a coincidence window of 5 ns.



Trigger type
2 channels coincidence

Window width (ns)
5

Trigger
Individual trigger

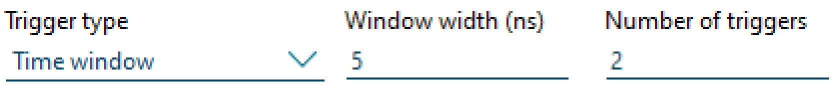
Channel
0

Trigger 2
Individual trigger

Channel
1

Figure 29 – Trigger coincidence example.

Figure 30 shows the default setting of the time window trigger. Here the DAQ is triggered by having N triggers in a given time window.



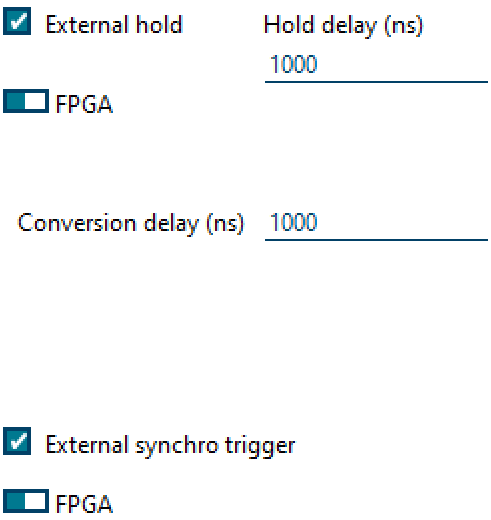
Trigger type
Time window

Window width (ns)
5

Number of triggers
2

Figure 30 – Time window default settings.

On the right panel of the setup tab, some DAQ options are available to the user (see Figure 31). Here the user can choose to have the ASIC acquisition trigger to be internal or external. The hold signal for the peak detector also can ben internal or external. The conversion delay is the delay between the acquisition trigger and the ADC conversion (if using internal hold) or the delay between the external hold and the conversion. If using the internal hold this acquisition delay has to take into account the internal delay of the ASIC (i.e. if the internal hold delay is 800 ns and the conversion delay is 100 ns the conversion will start before the hold is asserted). The external synchro trigger is an external signal sent both to the ASIC and an FPGA I/O, and can be used to synchronize a pulser with the DAQ or to force triggers in the ASIC for pedestal acquisition. External Hold and External Synchro are respectively routed to SMA connectors IO_FPGA7 and IO_FPGA6, allowing the DAQ to be controlled via a 2.5V TTL signal.



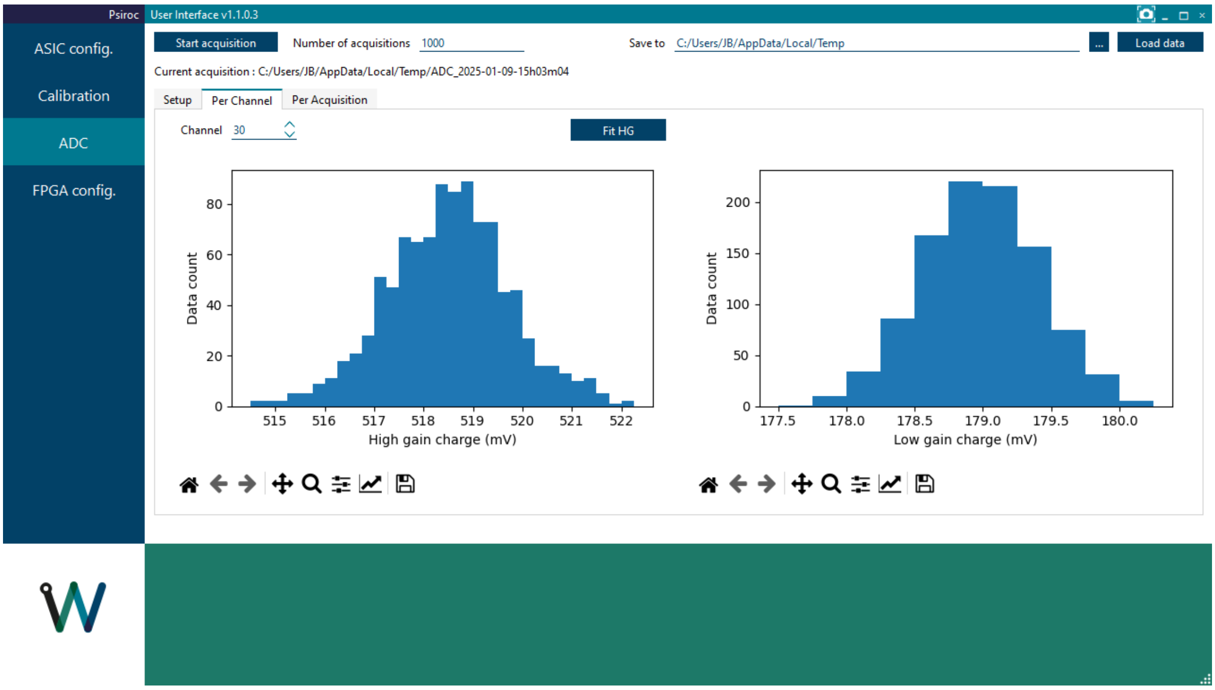
☑ External hold
Hold delay (ns) 1000
FPGA

Conversion delay (ns) 1000

☑ External synchro trigger
FPGA

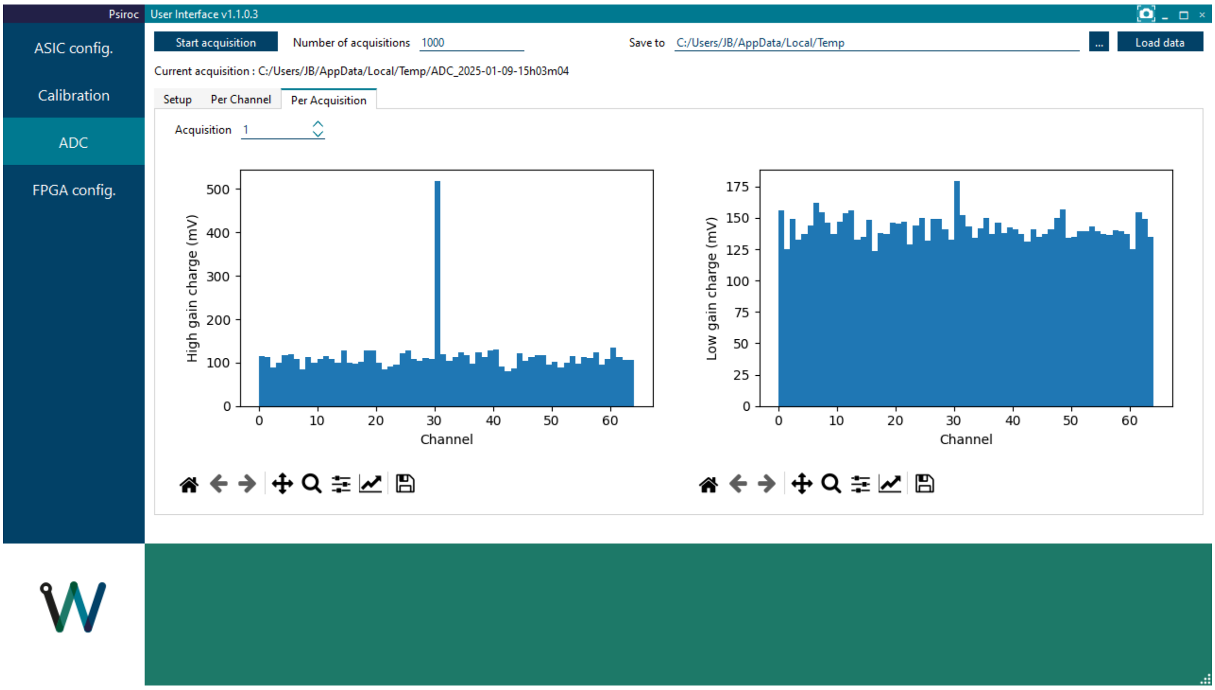
Figure 31 – Acquisition behavior options.

The following acquisitions have been performed by injecting in channel 30 using the SMA connector with a 10 pF injection capacitance and injecting 50 fC (500 mV pulse with 40 dB attenuation → 5 mV × 10 pF = 50 fC) with default ASIC configuration.



Number of acquisitions 1000
Save to C:/Users/JB/AppData/Local/Temp
Current acquisition : C:/Users/JB/AppData/Local/Temp/ADC_2025-01-09-15h03m04
Channel 30

Figure 32 – DAQ result per channel (charge spectrum histogram).



Number of acquisitions 1000

Save to C:/Users/JB/AppData/Local/Temp

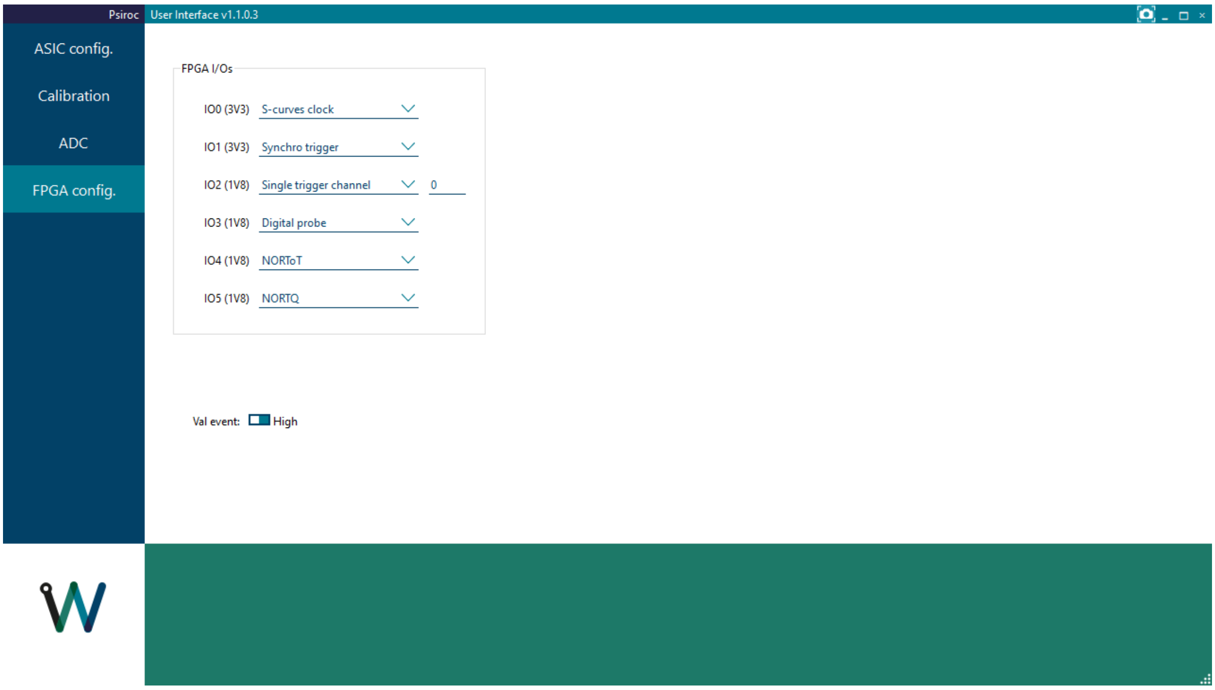
Current acquisition : C:/Users/JB/AppData/Local/Temp/ADC_2025-01-09-15h03m04

Acquisition 1

Figure 33 – DAQ result per acquisition.

3.4 FPGA configuration

FPGA I/Os are located in the bottom-right corner of the board. Output signals can be selected using combo boxes. The IO0 by default is the synchronization clock to send to the pulser for the S-curves test using Ctest. Synchro trigger on IO1 is the synchronization trigger for the external hold scan.



IO0 (3V3) S-curves clock

IO1 (3V3) Synchro trigger

IO2 (1V8) Single trigger channel 0

IO3 (1V8) Digital probe

IO4 (1V8) NORToT

IO5 (1V8) NORTQ

Val event: High

Figure 34 – FPGA configuration.

4 First measurements

4.1 Injection

The injection setup provided here is an example. Injection parameters aren't set in stone but the following is a good way to inject signal if the user is unfamiliar with the electronics.

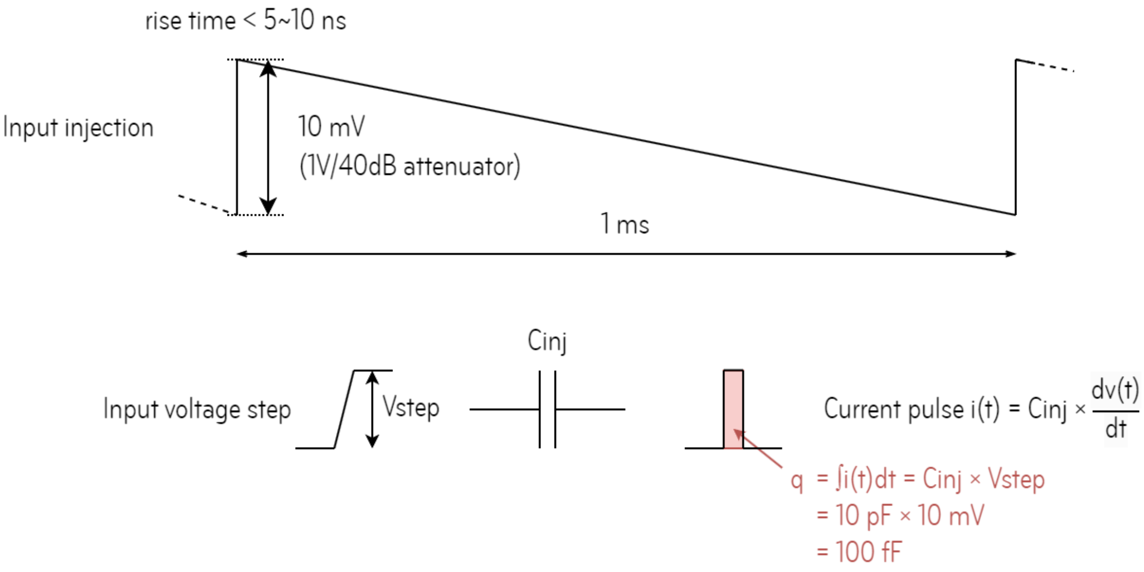


Figure 35 – Basic injection setup example.

The injection capacitance can be the ASIC embedded Ctest (1.5 pF) or an external one. SMA connectors are available on the testboard for channels 30, 34, 46, 53 and 62, next to the in_Ctest connector. On the board bottom layer next to the SMA footprint are soldered 10 pF injection capacitors and a 50 Ohm resistor to the ground for impedance adaptation. It is possible to change this injection capacitor by soldering another one on its place (0805 footprint).

Figure 36 - SMA input injectors and Analog_Probe output (top-left corner of the board).

Note that because of the added PCB trace length those channels are noisier than the others (see Figure 37). The added noise from the input capacitance is only visible for higher preamp gains, as for lower gains second stage noise becomes dominant.

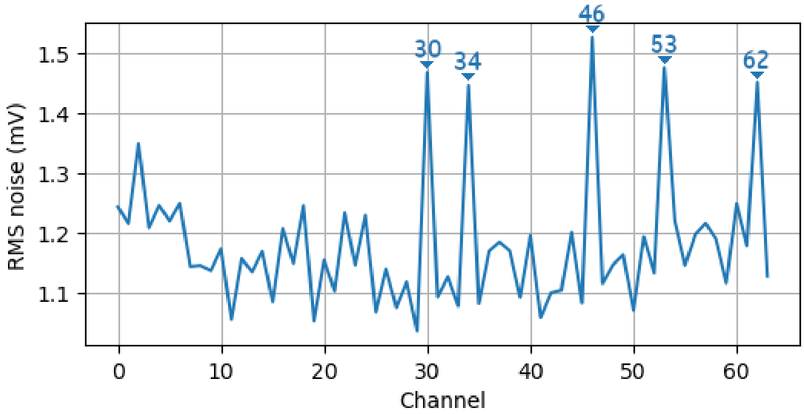


Figure 37 – Noise w.r.t. channel number measured at ADC output. Channels 30, 34, 46, 53 and 62 are noisier because of the added parasitics of the PCB trace. Channel 29 is the least noisy because its PCB trace is the shortest.

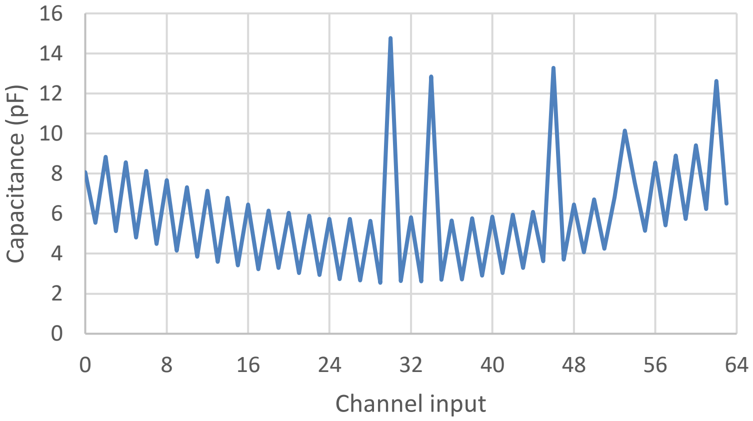


Figure 38 – PCB trace parasitic capacitance extracted from Altium designer.

A critical part of the injection is the grounding as any noise on the ground is amplified as if it were input signal. It is best to have ground braids (or desoldering braids) between the pulser and the on-board connector, preferably minimizing the distance between them as shown on Figure 39.

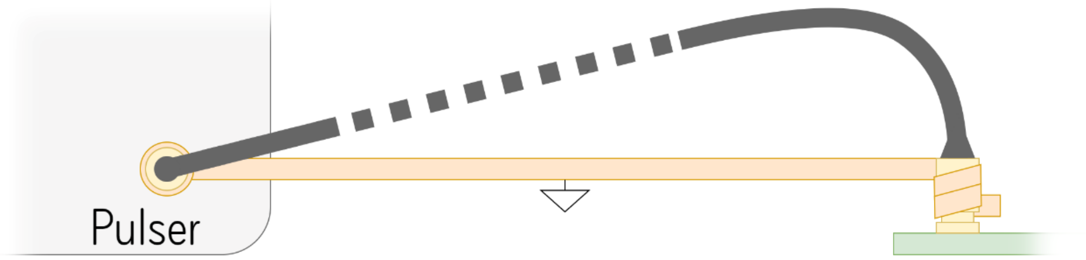


Figure 39 – Injection grounding.

4.2 Probes output

When injecting as shown on Figure 35 in a channel and by looking at the probe output on this channel, it is possible to observe the resulting signal on an oscilloscope. The transient signal at preamp and shaper outputs should look like Figure 40

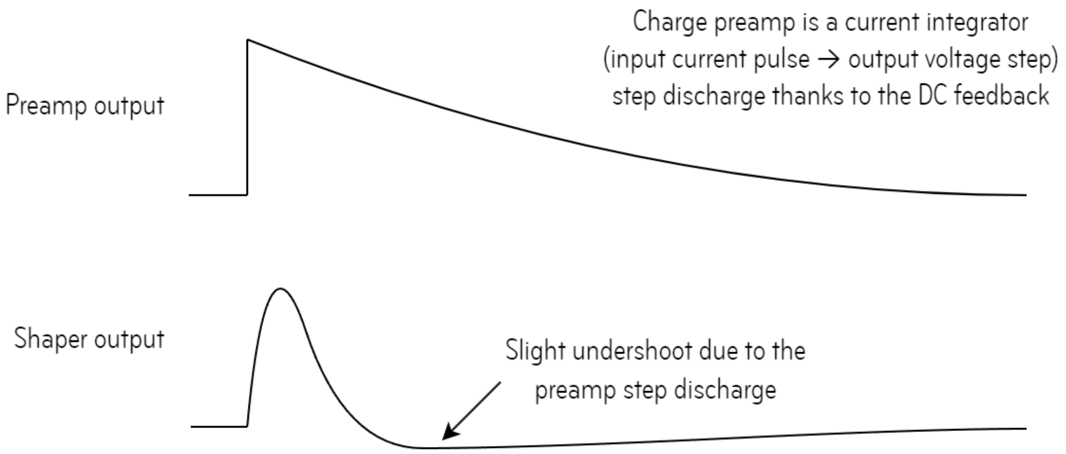


Figure 40 – Transient signal at preamp and shaper output.

From there the injection setup is valid and further measurements can be done as described on this user guide (S-curves, threshold scan, hold scan, ADC DAQ, etc.).

5 Appendix

5.1 S-curves

Two Gaussians are shown on Figure 41. 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 an RMS noise of 1. The blue Gaussian represents a signal with an RMS noise of 1 and the signal over noise ratio is 10.

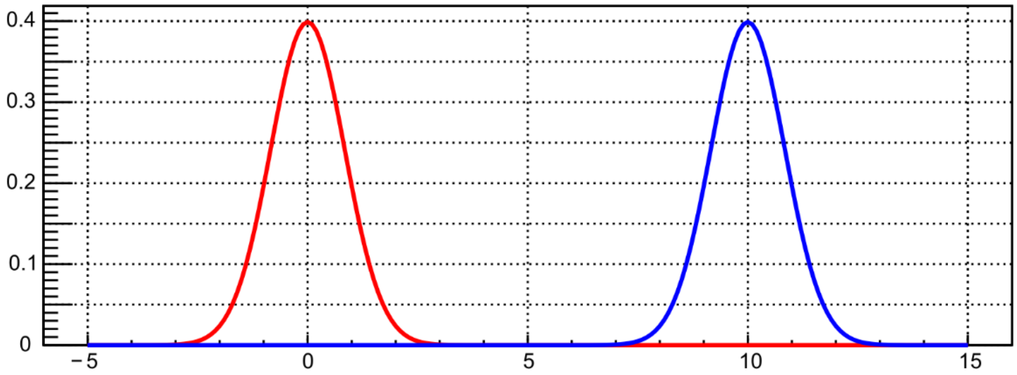


Figure 41 – Red: Normalized Gaussian with average = 0 and σ = 1 (arbitrary units). Blue: Normalized Gaussian with average = 10 and σ = 1.

By setting a threshold at x = 5, the probability for the noise of the pedestal to be over the threshold at any given time is

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

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

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

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 42.

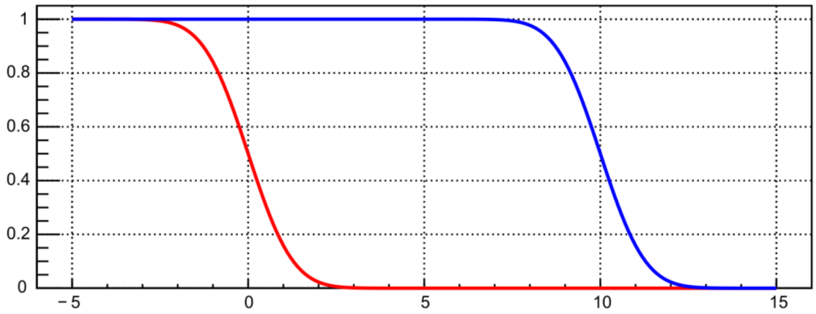


Figure 42 – S-curve representation of the signals shown on Figure 41.

This describes the probability to trigger on a discrete signal and if the acquisition time window used to sample the trigger was infinitely small the above representation would be valid for pedestal. But reality is a bit more complex because the trigger probability depends on the acquisition time as demonstrated on Figure 43. With a threshold set at y=5 (arb. unit), a trigger happens right at the beginning and with a threshold at 10 the trigger happens at t=900 (arb. unit). So, depending on the duration of the acquisition window on the pedestal the system can give a trigger or not.

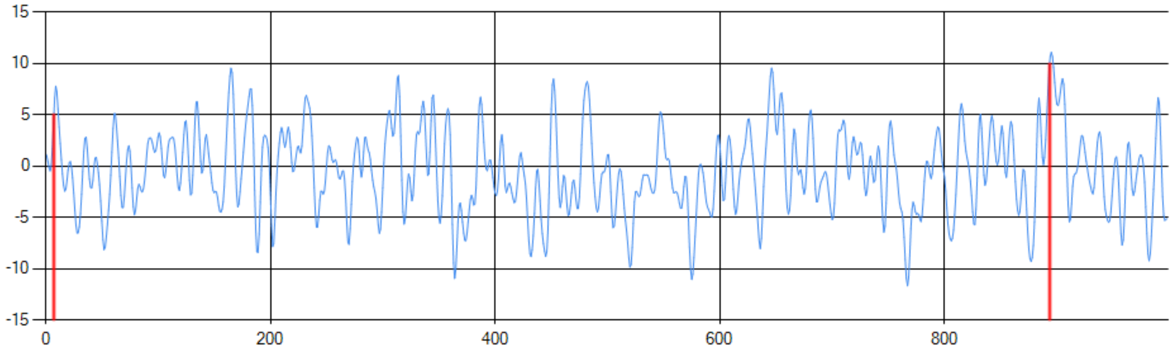


Figure 43 – Influence of acquisition time window on S-curves for pedestal noise.

A study shown in *Mathematical Analysis of Random Noise* by S.O. Rice gives a zero-crossing frequency of the noise for a bandpass as

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

With f_b and f_a being respectively the high-cut and low-cut frequencies. In the case of an ideal CRRC bandpass filter the zero frequency is near 3 times the center frequency f_c = 1/(2πRC). Noise crosses zero in both ascending and descending ways so the number of triggers on zero N_0 would be half the number of crossings 0.5 × f_0 × T = 1.5 × f_c × T with T the time interval.

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

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

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

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

Probability of having 0 events within T is

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

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

$$P(k \geq 1) = 1 - P(0) \tag{7}$$

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

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

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

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).

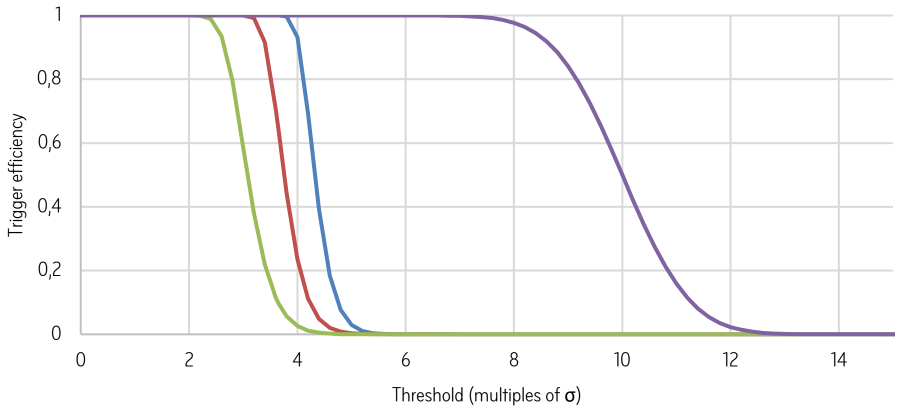


Figure 44 – Trigger efficiency on pedestal noise depending on threshold value for an acquisition window of 10 µs (green); 100 µs (red) and 1 ms (blue). Purple: S-curve of a signal with µ = 10 and σ = 1 (SNR = 10). The pedestal position is 0.

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

5.2 Shapers

Shapers are bandpass filters. They are used to improve the Signal to Noise Ratio (SNR). Figure 45 show the standard schematic of an inverting CRRC filter.

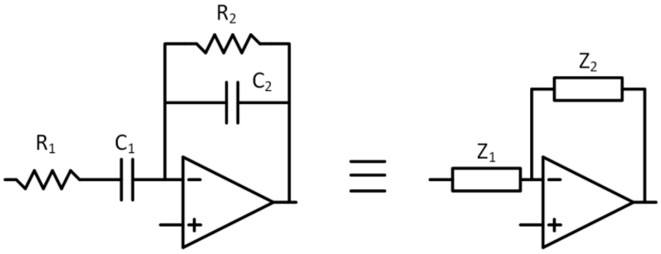


Figure 45 – Schematic of an inverting CRRC shaper.

From Figure 45 the transfer function is

$$H_{CRRC}(s) = -\frac{Z_2}{Z_1} \tag{10}$$

With $Z_1 = R_1 + \frac{1}{j\omega C_1}$ and $Z_2 = R_2 || \frac{1}{j\omega C_2} = \frac{R_2}{1+j\omega C_2 R_2}$

Defining i$\omega = s$

$$H_{CRRC}(s) = -\frac{\frac{R_2}{1+C_2R_2s}}{R_1+\frac{1}{C_1s}} \tag{11}$$

$$H_{CRRC}(s) = -\frac{R_2}{R_1}\frac{R_1C_1s}{(1+R_1C_1s)(1+R_2C_2s)} \tag{12}$$

To obtain a good bandpass filter, we choose $C_1R_1 = C_2R_2 = \tau$ giving the following transfer function

$$H_{CRRC}(s) = -\frac{R_2}{R_1}\frac{\tau s}{(1+\tau s)(1+\tau s)} \tag{13}$$

The ideal step response is given by

$$V(t) = \mathcal{L}^{-1}\left[F_{step}(s) \times H_{CRRC}(s)\right] \tag{14}$$

$$= \mathcal{L}^{-1}\left[\frac{1}{s} \times H_{CRRC}(s)\right] \tag{15}$$

$$= \mathcal{L}^{-1}\left[-\frac{R_2}{R_1}\frac{\tau}{(1+\tau s)^2}\right] \tag{16}$$

$$= -\frac{1}{\tau}\frac{R_2}{R_1}\mathcal{L}^{-1}\left[\frac{1}{\left({}^{1}/_{\tau}+s\right)^2}\right] \tag{17}$$

$$= -\frac{t}{\tau}\frac{R_2}{R_1} \times e^{-t/\tau} \times u(t) \tag{18}$$

With $u(t)$ the Heaviside step function given as

$$\forall t \in \mathbb{R}, u(t) = \begin{cases} 0 & si\ t \leq 0 \\ 1 & si\ t > 0 \end{cases}$$

The shaper gain is adjustable with the R_2/R_1 ratio. With an ideal step of amplitude -1 and τ = 200 ns, considering $R_1 = R_2$ the transient response is shown on Figure 46.

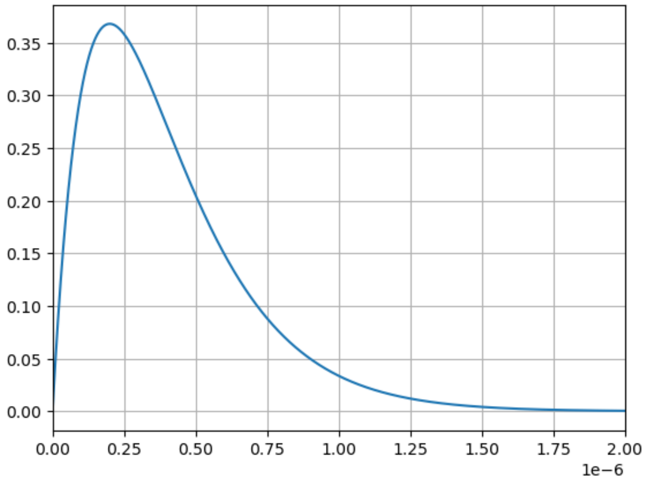


Figure 46 – Transient response to an ideal input step of amplitude -1 with τ = 200 ns.

This transient equation is valid only in the case of an ideal input step. In reality it's never the case and to obtain a realistic signal it is necessary to convolve the impulse response of the shaper with the input signal. Impulse shaper response is given by

$$V_{sh}(t) = -\frac{R_2}{R_1}\left(1 - \frac{t}{\tau_{sh}}\right) \times exp\left(-\frac{t}{\tau_{sh}}\right) \tag{19}$$

A first improvement on the previous equation is taking into account the preamplifier step discharge, with the preamplifier output signal being a simple exponential discharge

$$V_{pa}(t) = -exp\left(-\frac{t}{\tau_{pa}}\right) \tag{20}$$

By calculating

$$(V_{pa} * V_{sh})(t) = -exp\left(-\frac{t}{\tau_{pa}}\right) * \left(-\frac{R_2}{R_1} \times \left(1 - \frac{t}{\tau_{sh}}\right) \times exp\left(-\frac{t}{\tau_{sh}}\right)\right) \tag{21}$$

It is obtained

$$(V_{pa} * V_{sh})(t) = \frac{R_2}{R_1} \times \frac{\tau}{\tau_{sh}} e^{\frac{-t}{\tau_{pa}}}\left(1 - e^{\frac{-t}{\tau}} + \frac{te^{\frac{-t}{\tau}} + \tau e^{\frac{-t}{\tau}} - \tau}{\tau_{sh}}\right) \tag{22}$$

With

$$\tau = \frac{\tau_{pa} \times \tau_{sh}}{\tau_{pa} - \tau_{sh}}$$

The effect of the preamplifier discharge can then be observed at shaper output on Figure 47 as it creates an undershoot after the signal and the peak amplitude and peaking time are lessened.

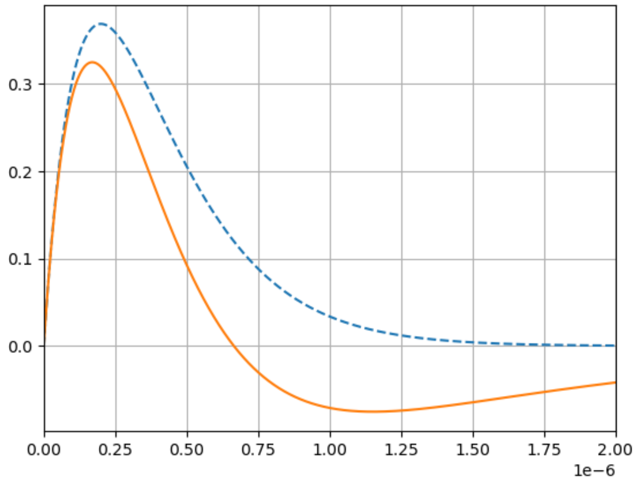


Figure 47 – Orange: Transient response to a step discharge of amplitude -1, τ_{pa} = 1 µs, R_2/R_1 = 1, τ_{sh} = 200 ns.
Blue dashed: ideal step response ($\tau_{pa} \rightarrow \infty$).

Further improvements of the model can be brought by adding bandwidth limitations in the preamplifier and shaper creating ballistic deficit etc.