FASTER QT2T MnM USER MANUAL

The QT2T MnM (**M**easurement **n**umerical **M**odule) is a module particularly suitable for precise half-life measurement, or any other counting experiment where the energy deposited in the detector is variable from pulse to pulse.

Cathy Fontbonne

Summary

I Theory of operation

The QT2T is a QDC that computes, among other things, the threshold-to-threshold charge: i.e., for each electronic pulse, the integration of all signal samples whose amplitude is over threshold.

One of the advantages appears on the histograms of the charges deposited in the detector. At high counting rates, a piled charge is the sum of two distinct charges with a QT2T gate, not a value truncated by a fixed charge gate, as shown on Figure 1 below:

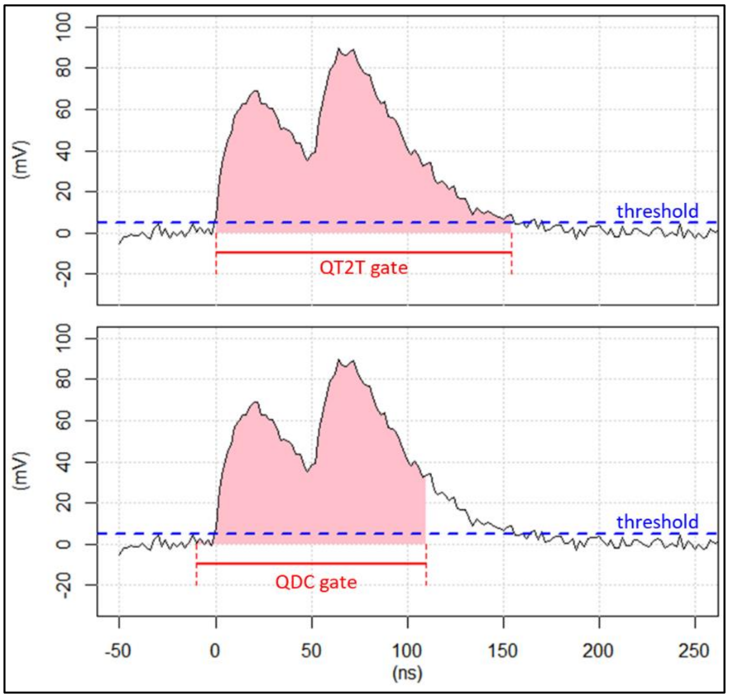

Figure 1: Difference between the QT2T gate and the QDC gate : the QT2T gate expands according to the time width of the pulse

The Qt2t MnM provides other information about the pulse:

- its amplitude,
- its duration,
- the position of the maximum,
- and a charge whose gate is chosen by the user before the pulse. This is used to check the baseline level.

If the user does not choose to set a specific dead time, the dead time of the Qt2t MnM is reduced to the pulse duration + 2 ns, as long as this pulse lasts more than 6 ns.

II Preamble

II.A FASTER launching

To take benefit of the QT2T MnM of the FASTER acquisition, the experimenter can build his experiment interface with the command `faster_setup_gui`.

The interface, shown on Figure 2, is launched. To use the FASTER QT2T MnM, the experimenter must select the "QT2T" type, by right-clicking on the daughterboard line.

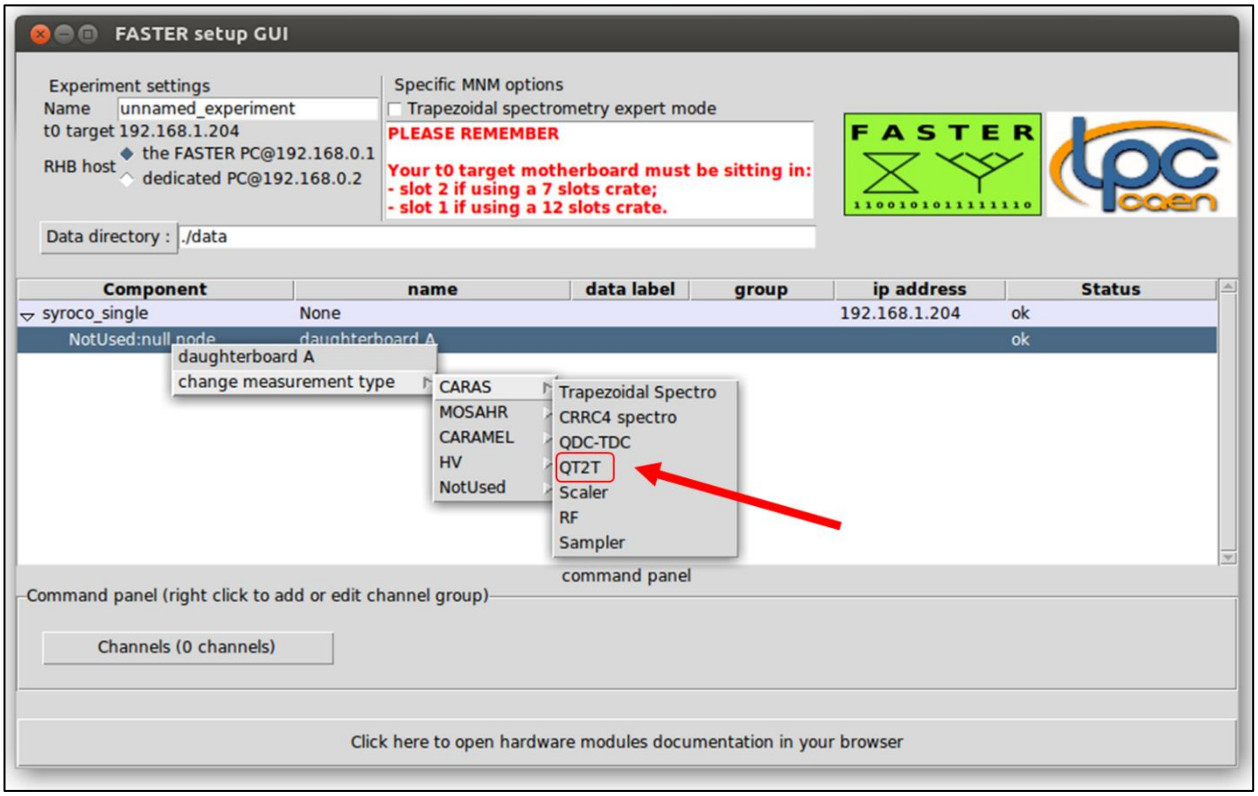

Figure 2: experiment building with "faster_setup_gui".

Before exiting the application form, the user can choose the names of the experiment, of the daughterboard and of each channel of the "daughterboard".

With the command line `faster_gui`, the experimenter accesses to the FASTER acquisition display, shown on Figure 3.

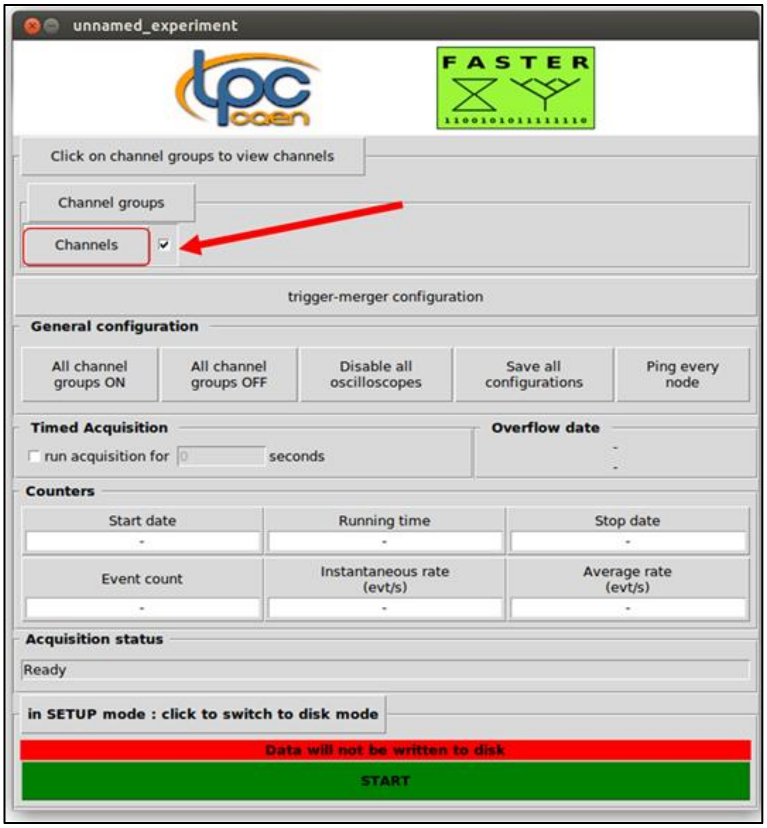

Figure 3: faster_gui interface.

By clicking on "Channels" (cf. Figure 3), the MnM interface is displayed, allowing the user to adjust the different module parameters (Figure 4).

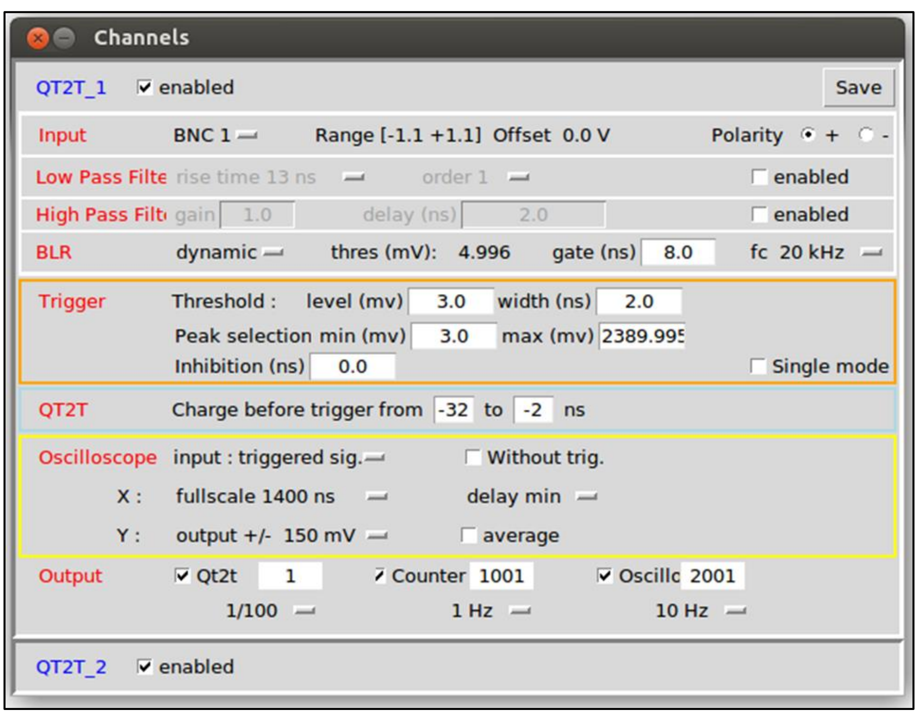

Figure 4: QT2T interface

II.B RHB launching

To take benefit of RHB displays, the experimenter shall execute the command `faster_rhb_demo_qt2t_copy` in the working directory, in order to have an example of RHB files adapted to the FASTER QT2T MnM.

A new repertory `QT2T_RHB_Demo` is created. The user shall go in the full_config repertory contained in the repertory `QT2T_RHB_Demo`, before launching RHB interface.

```
cd QT2T_RHB_Demo
cd full_config
RHB -r
```

III Description of FASTER QT2T MnM

The interface includes several tuning modules, synthetically described in Table 1 and Figure 5. Each module has some parameters that are adjustable by the user to achieve the best performance.

module	Border color	Purpose
Input module	white	It defines the dynamic range, the polarity of the input signal, and provides filters to clean it.
Trigger module	orange	It timestamps the particle arrival time and wakes up the other modules except BLR module.
QT2T module	green	It allows the user to define the charge gate before the event pulse.
Oscilloscope module	yellow	It allows the user to visualize several module signals
Data output module	no color	It allows the user to choose output data (label, rate)

Table 1: Description of the QT2T MnM

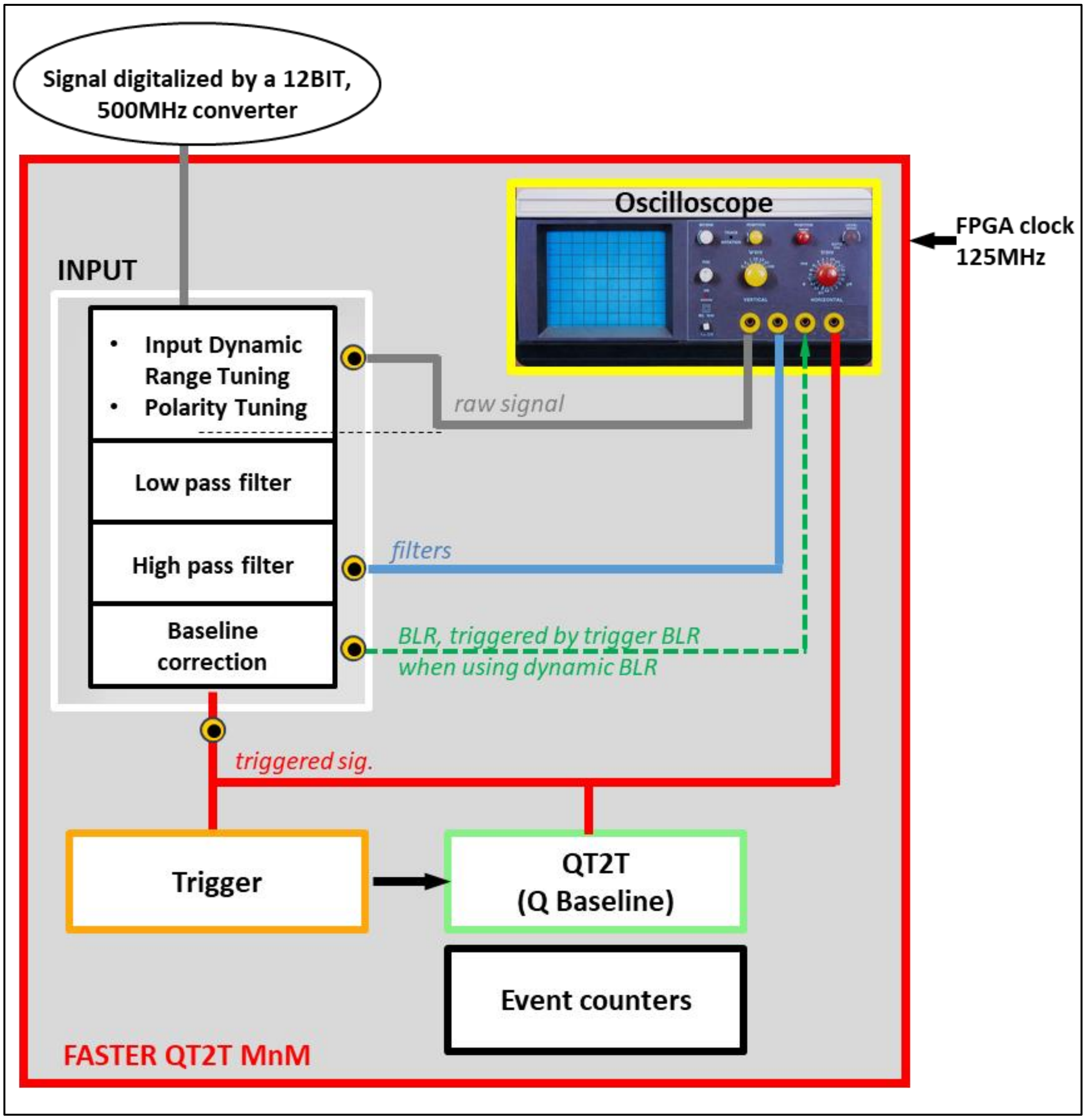

Figure 5: Diagram of FASTER QT2T MnM. The border colour of each box is as the one of the MnM interface.

The following chapters describe these different modules and explain how to select all the FASTER QT2T MnM parameters to achieve the best charge histogram. We recommend to use RHB facilities to display signal shapes and deposited energy spectra. The signal shapes are observable thanks to the Oscilloscope module. This one is therefore the first to be described.

III.A The Oscilloscope module

The signal, which is displayed by the oscilloscope module, is defined by parameter ① (Figure 6). Each ◉ on Figure 5 indicates the location of an Oscilloscope module input. The use case of this signal is described in Table 2.

Input signal	Use case
raw signal	To adjust Raw signal module parameters
filters	To adjust filters parameters before BLR
BLR	To adjust BLR module parameters
triggered sig.	To adjust trigger module parameters.

Table 2: Use case of the Input signal of the Oscilloscope module

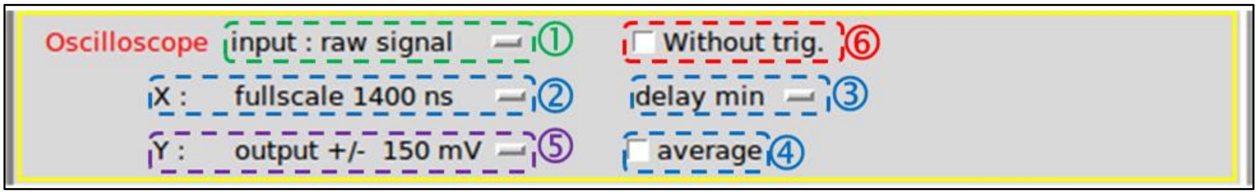

Figure 6: Oscilloscope module.

The user can select the time fullscale range with parameter ②. Oscilloscope data contains up to 700 points separated by a time step. As the time step is a power of 2 ns, the proposed dynamic ranges vary from 1400 ns to 5.2 µs.

As each ion-induced pulse, triggered by the Trigger module or the BLR Trigger module, is identified by the relative time "0 ns" in an oscilloscope frame, the user can select where to place the sample "0 ns" ("min", "25%", "50%", "75%"), with the shift parameter ③.

If the “AVERAGE” button ④ is checked, every sample of the oscilloscope display is the average of many samples (2 for 2800 ns range, 4 for 5600 ns range, 8 for 11 µs range …). Therefore, oscilloscope display seems to be less noisy than it is in reality. When “AVERAGE” is unchecked, one sample every 2 or 4 or 8 or 16 (etc.), depending on time-range value of 2800 ns, 32 ns, 5600 ns or 11 µs (etc…) forms the oscilloscope display. The sample, which triggered the Trigger module, is not necessarily apparent.

The Y-fullscale ⑤ depends on CARAS daughter board characteristics. It is always 0 centered, and the experimenter can zoom the signal display between +/- 150 mV, +/- 300 mV, +/- 600 mV, +/- 1200 mV or +/- 2400 mV.

If “without trig” is checked in ⑥ (Figure 6), Oscilloscope module does not use the Trigger module or BLR Trigger to start storing an oscilloscope frame, but it uses a clock, whose frequency is defined in the “Data output” module (cf. Figure 7).

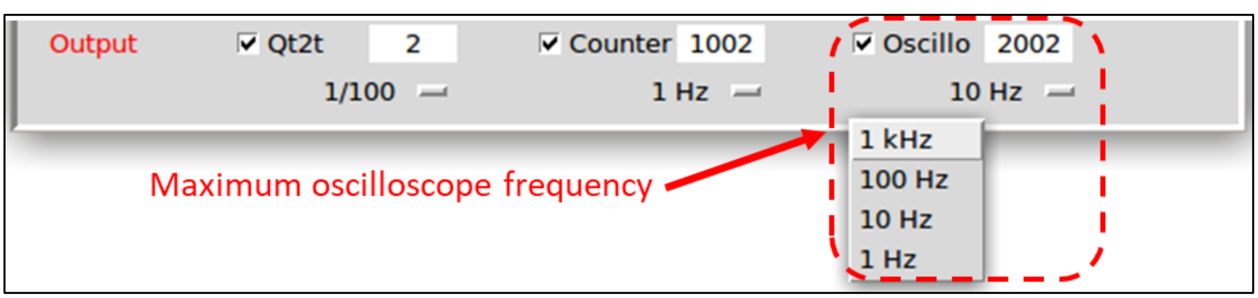

Figure 7: Maximum frequency of the oscilloscope frame definition

If “without trig” is unchecked in ⑥ (Figure 6), the Oscilloscope module is woken either by the BLR trigger if the displayed signal is “trigger BLR”, otherwise by the Trigger module. In all case, the maximum oscilloscope frequency is defined by the Data output module.

III.B The Input module

In this module, the user has to define the raw signal (cf. Figure 8). That means:

- its link with one of the two daughter board input channel ①,
- its ranges ②,
- its polarity③.

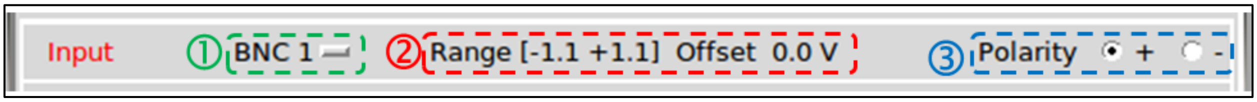

Figure 8: The raw signal parameters.

III.B.1 The CARAS Data Channel Selection

The CARAS daughter board has two input channel called SMB 1 and SMB 2. Each FASTER QT2T MnM in the FPGA can be linked with any SMB, as shown Figure 9 and Figure 10.

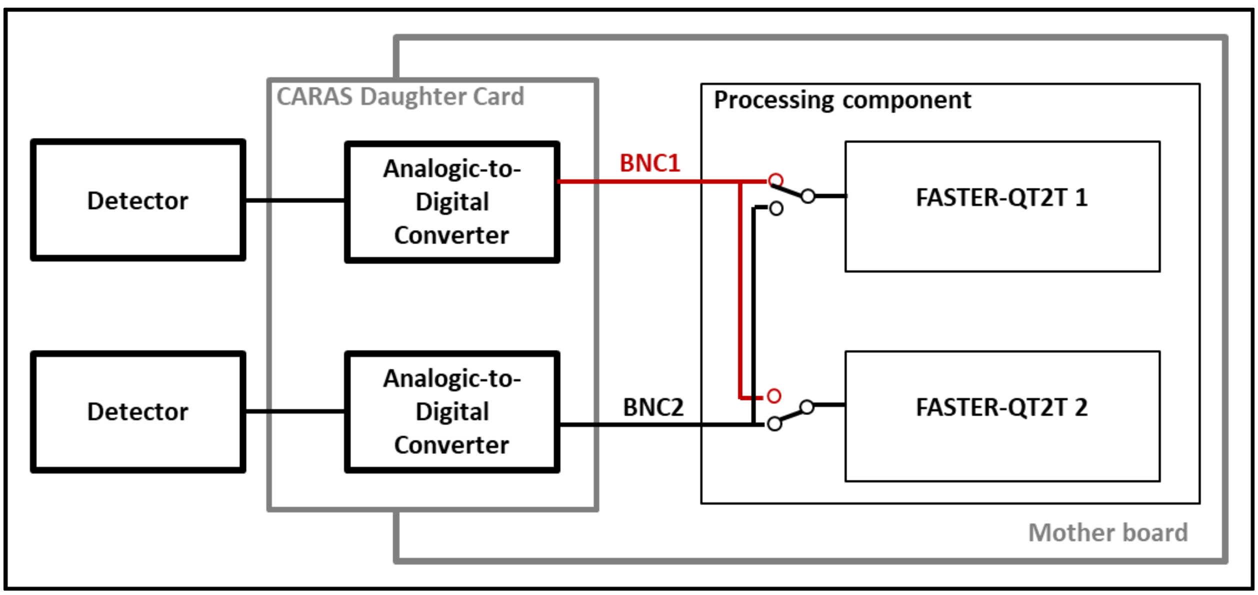

Figure 9: Multiplexers to select which analog channel to be processed.

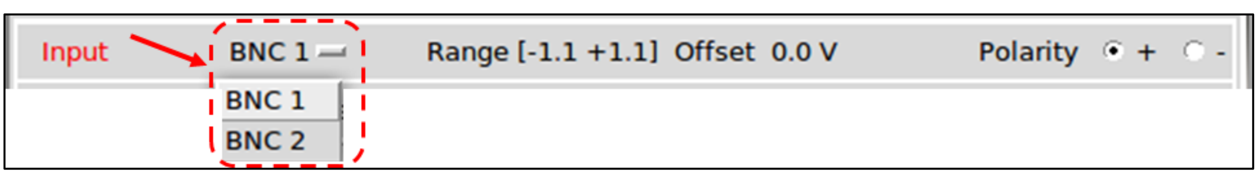

Figure 10: Input Channel selection.

III.B.2 Range Tuning

Each SMB channel can have a different dynamic input range from [0.0 +2.2] V to [-2.2 0.0] V. This dynamic range is configured by the user, by clicking on “Range” switch ②. A new interface is launched (cf. Figure 11): the user can add a voltage source (called “Offset”) between -1.1 V and 1.1 V with 0.1 V-step.

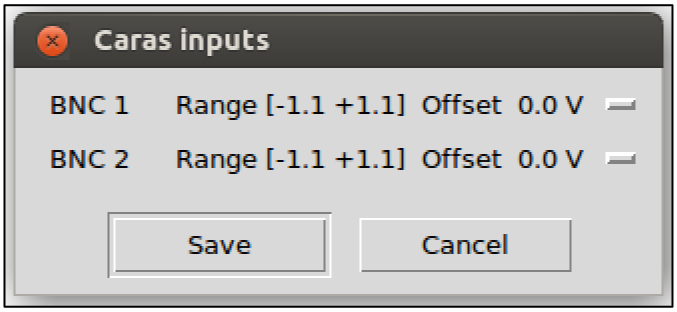

Figure 11: fullscale range interface for CARAS daughter board.

III.B.3 Polarity Tuning

The Oscilloscope module allows the user to display the raw signal (Figure 12):

The FASTER-SAMPLER process uses positive values. If the detector provides a negative signal, it is essential to convert it into a positive signal, with selection check button ③ (Figure 8).

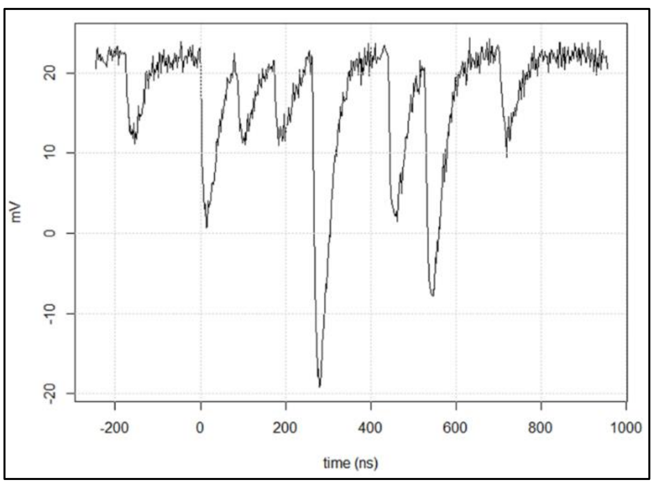

Figure 12: Raw signal example. As the pulse direction is downward, the polarity ”-“ must be selected.

III.C The low-pass filter module

If the experimenter wants to benefit from a low-pass filter, he must check the "enable" box ① (Figure 13).

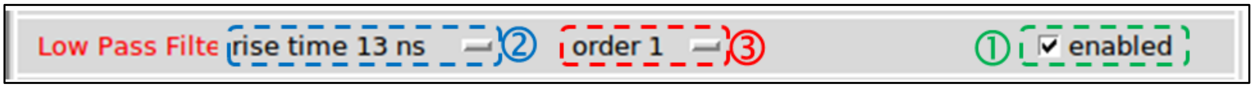

Figure 13: Low-pass filter parameters.

The user can select with button ② (Figure 13) one of the four available filters: either he chooses a moving average of 4 samples, or he chooses a filter defined by its rise time: 13 ns, 27 ns or 57 ns.

He can select the filtering order ③ (Figure 13) : "order 1" means that there is only one low-pass filter, "order 2" means that there are two successive low-pass filters, "order 3" means that there are three successive low-pass filters. Figure 14 shows the different effects of low-pass filters on one electronic pulse.

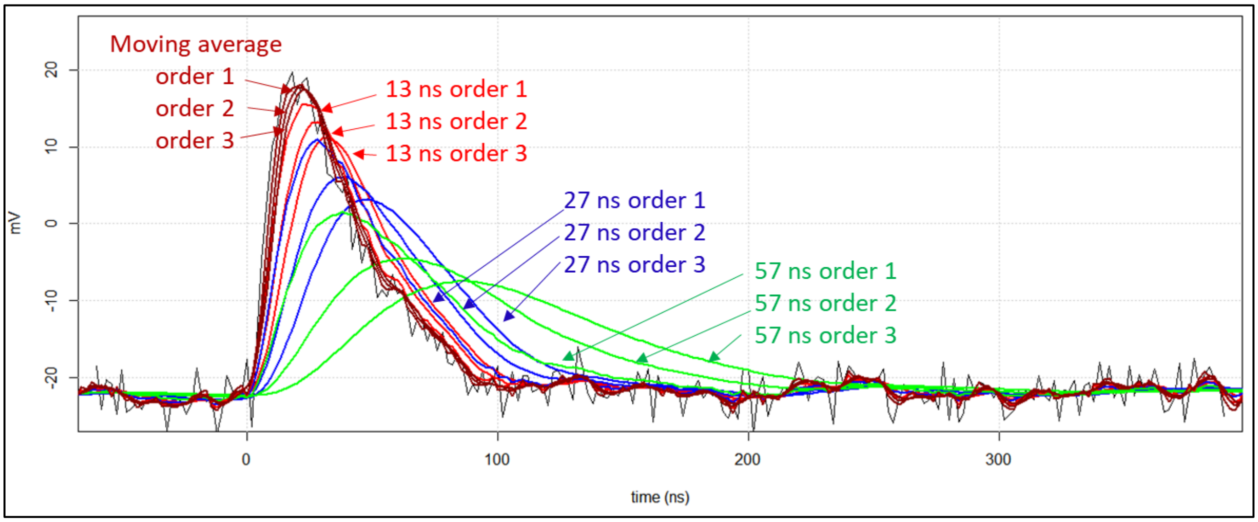

Figure 14: effects of different low-pass filters

III.D The High-pass filter module

If the experimenter wants to benefit from a high-pass filter, he must check the "enable" box ① (Figure 15).

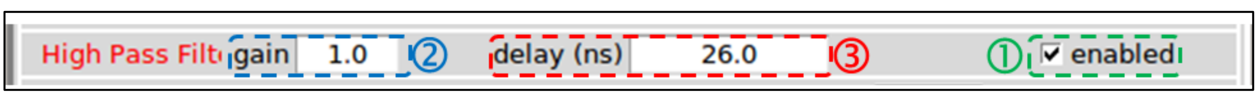

Figure 15: High-pass filter parameters

This high-pass filter is a delay-line pass-filter. The experimenter has to tune the gain from 0 to 1 ② (Figure 15), and the delay from 2 ns to 56 ns ③ (Figure 15).

The implemented operation is:

$$S_{out}(t) = S_{in}(t) - gain \cdot S_{in}(t - delay)$$

If the gain is equal to 1, the resulting signal is bipolar, with a baseline centered on zero, as shown on Figure 16.

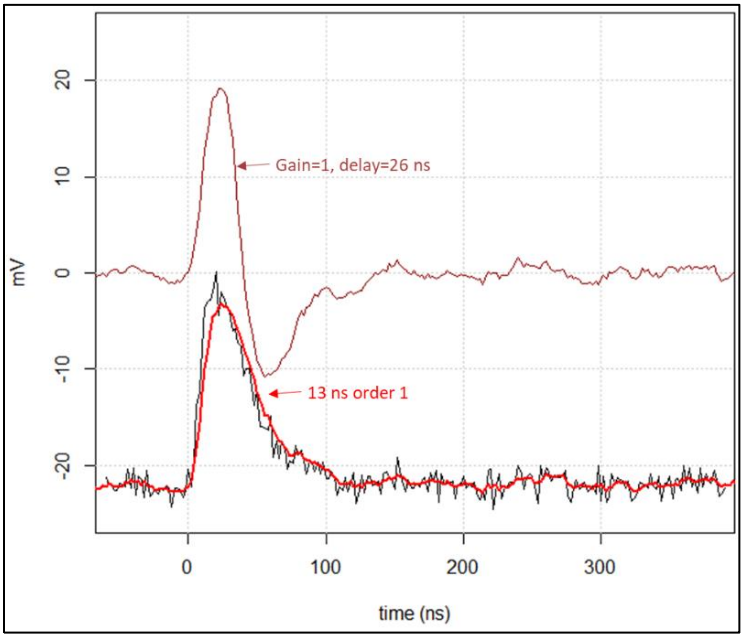

Figure 16: Effect of the delay-line high-pass filter (gain=1, delay=26 ns) located after a low-pass filter (t_{rise}=13 ns, order 1)

III.E Baseline Restorer Filter module

The role of the BLR module is to put the baseline at 0 mV. The BLR module operates on the filtered signal if filters are enabled. The user can choose between two kinds of baseline restorer: a constant one and a dynamic one.

III.E.1 Constant BLR

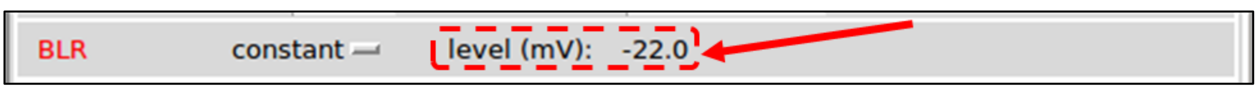

Figure 17: Constant BLR level

This constant BLR simply subtracts a constant level.

III.E.2 Dynamic BLR

Figure 18 explains the dynamic BLR concept. The dynamic BLR filter follows the slow variations of the baseline as long as it does not detect an event. If it detects one, it maintains the last constant level during all the event duration. The resulting signal is therefore the baseline, which is then subtracted from the input signal.

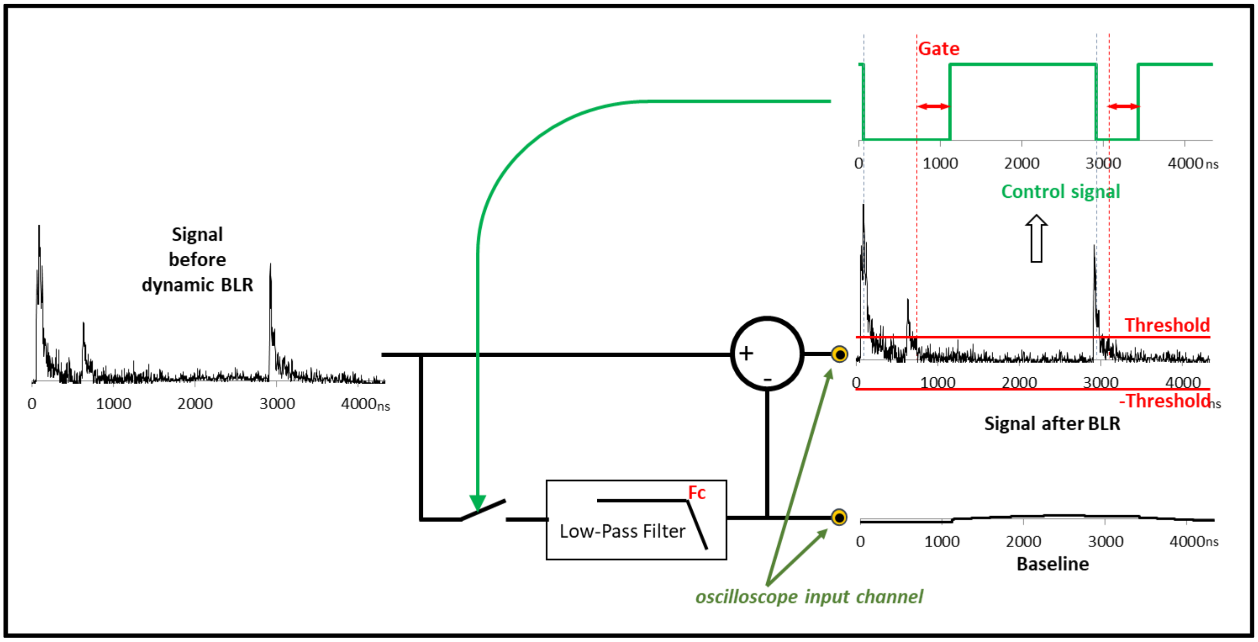

Figure 18: BLR concept

The signal from the previous filter module is filtered by a new low-pass filter. This tracking filter lets through any slow variations of the signal. The **cut-off frequency "fc"** ① (Figure 19) of this low-pass filter is 20 kHz, 160 kHz, 640 kHz or 6600 kHz. It is a first order low-pass filter.

The baseline tracking by the low-pass filter is stopped when an event is detected. That means when the cleaned **signal after BLR exceeds a "± threshold" parameter value** ②(Figure 19). Then the baseline tracking stops while the value of signal samples is above the threshold and after a **duration defined in the "gate" parameter** ③(Figure 19). Thus, the signal tracking is stopped during all the signal duration. When the baseline tracking is stopped, the computed baseline value is set to the baseline value just before the detected event. In case of pile up, the computed baseline value is equal to the baseline value just before the first detected event.

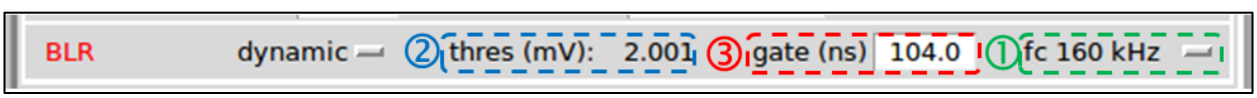

Figure 19: Dynamic BLR parameters.

To adjust BLR threshold, the experimenter must display **"Input: BLR"** with the Oscilloscope module. It is very important that the BLR adjustment is minutely performed by observing this oscilloscope display:

- The BLR threshold should be as small as possible, and selected just above the electronic noise. If this threshold level is too low, the BLR will never have the opportunity to follow the baseline. If it is too high, it will take physical events as baseline disturbances.
- "gate" values duration, where tracking is disabled, must be large enough to encompass the entire input signal of the BLR.
- The low-pass filter frequency Fc is preferentially chosen at 20 kHz, in order to improve the signal-to-noise ratio. If pulses rate is high, this frequency must be increased.

III.F The Trigger module

The role of the trigger module is to detect signal pulses. The trigger module features a threshold discriminator. The detected event is timestamped with a 2 ns accuracy clock.

To adjust Trigger module parameters, the experimenter must display **"Input: triggered sig."** with the Oscilloscope module.

All input pulses of the Trigger module, whose signal has crossed the threshold "Level" (Figure 20 ①) on rising edge and remains above this level during at least the duration "width" (Figure 20 ②), are detected. But only pulses with an amplitude between [min; max] during the first 64 nanoseconds ("Peak selection" parameter in Figure 20 ③), are retained.

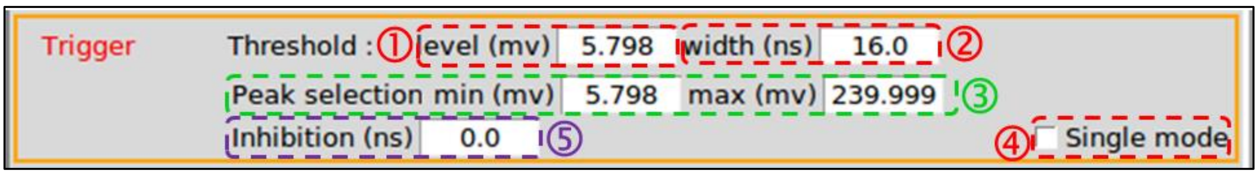

Figure 20: parameters of the threshold discriminator.

On Figure 21, we can see that a judicious value of "width" parameter can discriminate physical event from noise. "width" parameter is a multiple of 2 ns, between 2 ns and 64 ns. When displaying the "triggered sig." signal with an 2 ns step, the sample at time "0 ns" is the first sample above the threshold level.

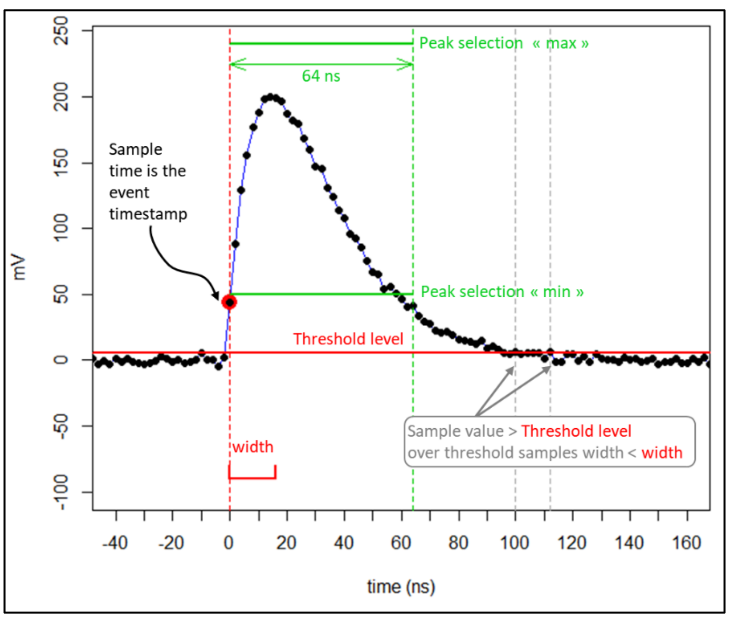

Figure 21: The first pulse, which has crossed the threshold level is detected. The following ones are not, thanks to a judicious value of the "width" parameter.

By checking "Single mode" (Figure 20 ④), the "Single Arm" button appears (Figure 22). A cursor click on this button will trigger a single pulse detection.

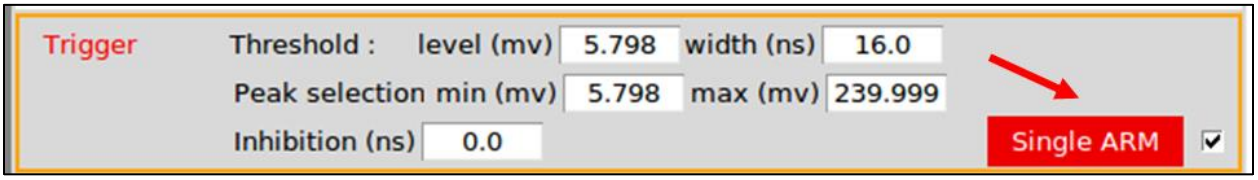

Figure 22: "Single Arm" button

Finally, the experimenter can adjust the minimum duration between two detections with the "Inhibition" parameter (Figure 20 ④), in order to ensure dead time duration (cf. § III.I). This duration must be wider than the pulse width, otherwise it has no impact. It is a multiple of 8 ns between 0 ns and 17 s.

III.G QT2T

The "QT2T" module is used to define the charge gate before the triggered event pulse. The user can choose a charge gate with 1 to 16 samples between -64 ns and -2 ns, as shown on Figure 23 and Figure 24.

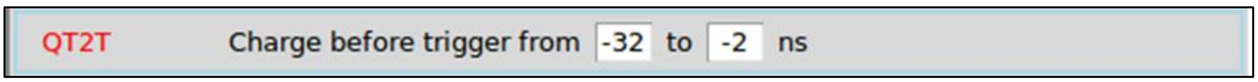

Figure 23: Definition of charge gate before triggered event.

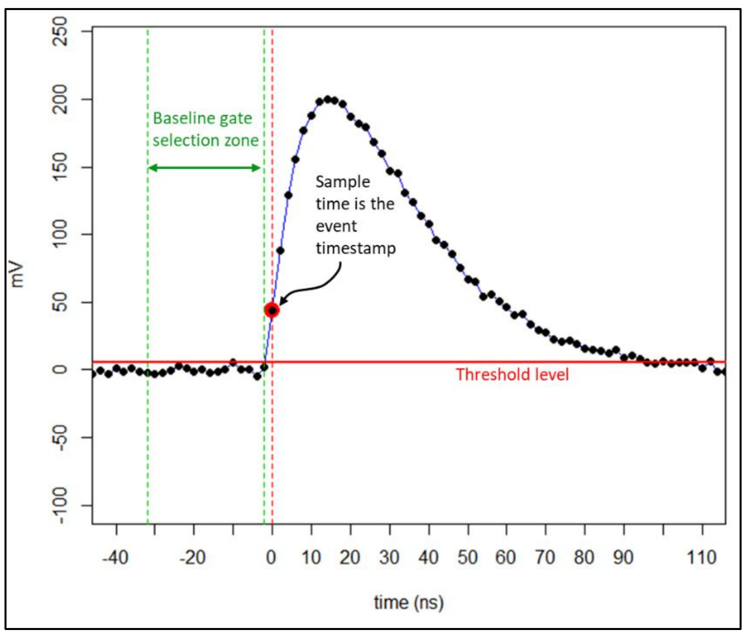

Figure 24: Selection zone of the baseline gate.

The study of the load before the triggered event is useful to check the stability of the baseline.

III.H Data output module

The Data output module (Figure 25) defines the output data sent to the computer by checking buttons ①. These data are:

- the data processed by the QT2T module ②,
- the counting data ③,
- the data from the Oscilloscope module ④.

To distinguish these data in the stored file, they have different labels ⑤. The user can moreover control the rate of output data.

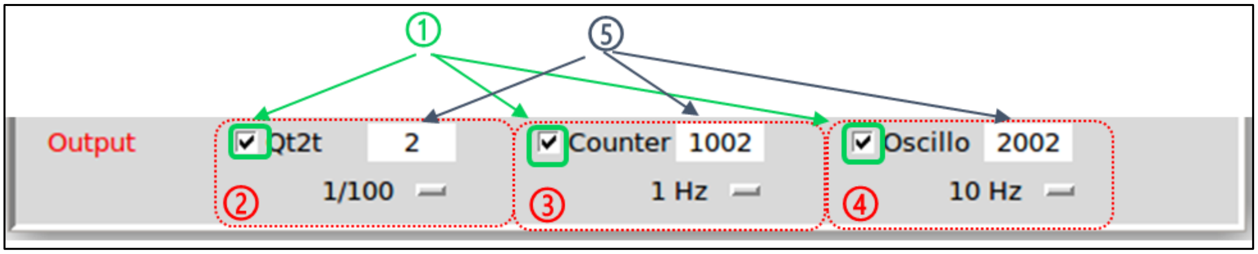

Figure 25: parameters of the Data output module

When the data comes out the FASTER QT2T MnM, they are temporally stored in some buffers of the FPGA, waiting for being sent to the computer by the Gigabit Ethernet communication.

Here are described the content of each data. If the experimenter uses RHB facilities, data from FASTER QT2T MnM are called QT2T in the RHB file "*.pid". In this file, the prefix label of each channel of the FASTER acquisition is defined. For example, if there is

1: QT2T: myChannel

that means that all data from the first channel of FASTER acquisition must have the prefix "**myChannel_**" in the configuration file of RHB (i.e. "*.facqConf" file). The "RHB –r" command, made in the directory including these files, starts RHB interface.

III.H.1 "QT2T" data

The experimenter can store or display all "QT2T" data processed by the QT2T module by enabling data output (cf. Figure 26 ①) and selecting 1/1 (cf. Figure 26 ②). But, when the event rate is too high and saturates the FASTER acquisition, the user can choose to store a percentage of this rate: one event data every 100 event data is sent, or every 10^4 event data, or one event data every 10^5 event data. With this decimation parameter, the acquisition efficiency can be controlled.

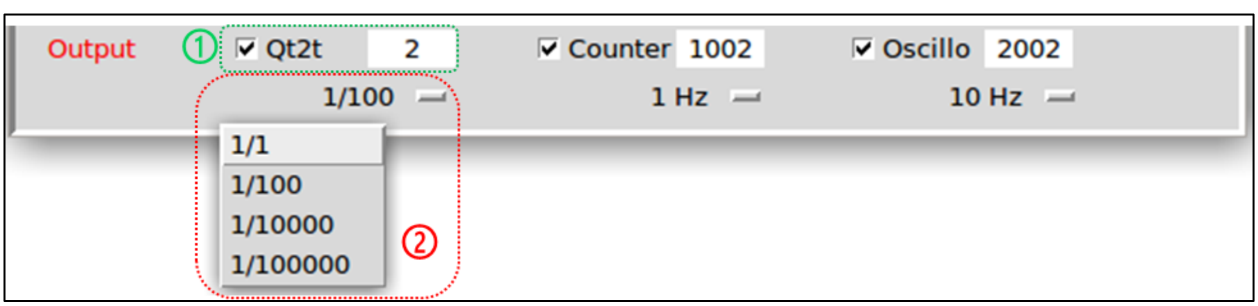

Figure 26: decimation of QT2T data

In “QT2T” data, the user has, for each event, the following information, shown on Figure 27.

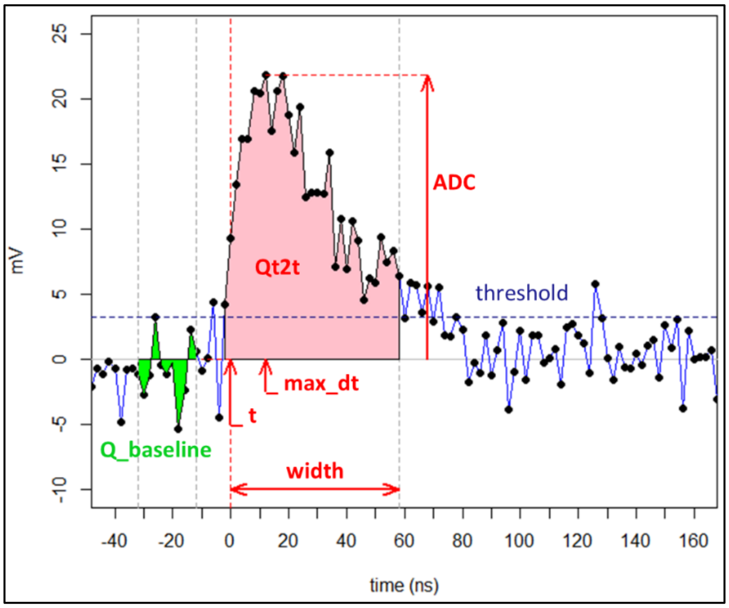

Figure 27: Information from QT2T data. The “saturated” flag is not represented.

- **myChannel_Qt2t**: this is the charge over threshold: it is the sum of samples from the first sample to the last sample over the threshold in the event. It is a 31-BIT signed integer. For information only, the measured charge (in Coulomb) is:

$$\boldsymbol{Q} = \mathrm{Qt2t} \cdot \frac{2390.\,10^{-3}}{2^{13}} \cdot\ 2 \cdot 10^{-9} \cdot \frac{1}{50}\ (C)$$

  i.e

$$\boldsymbol{Q} = \mathrm{Qt2t} \cdot \frac{239}{2^{12}} \cdot \frac{1}{5}\ (pC)$$

- **myChannel_Q_baseline:** this is the charge before the event. Its gate is defined in §**Erreur ! Source du renvoi introuvable.**. It is a 18-BIT signed integer.
- **myChannel_ADC**: the amplitude of the “triggered sig.”. It is a 20-BIT signed integer. To convert this value to millivolt, it must be multiplied by $\mathbf{2390/2^{13}}$.
- **myChannel_width**: the width of the pulse over threshold in nanosecond.
- **myChannel_max_dt**: the location in nanosecond of the maximum amplitude of the pulse.
- **myChannel_saturated**: the saturation flag. The value 1 informs that the raw signal, that has generated the “triggered sig.”, is outside the input range of the daughter card.
- **myChannel_t**: the time-stamp (in nanosecond) of the event, with a 2 ns accuracy.

III.H.2 “Counters” data

By enabling the “Counters” output (cf. Figure 28 ①), the experimenter can store and display the counting data, called “Counters” with a selected frequency of 1 Hz, 10 Hz, 100 Hz and 1 kHz (cf. Figure 28 ②).

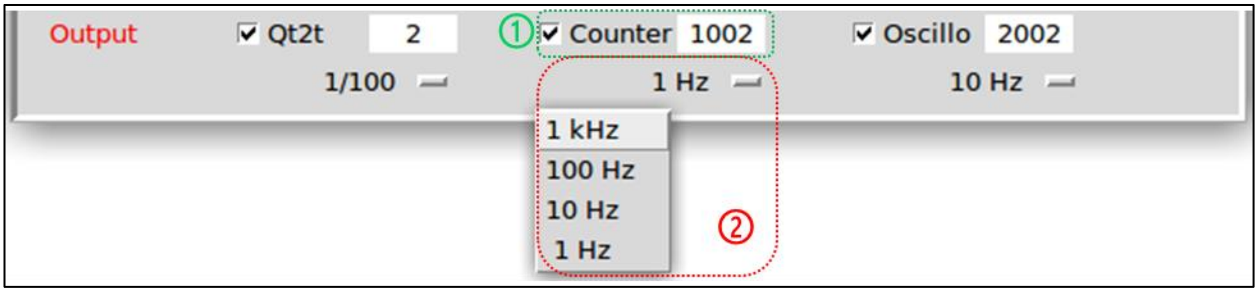

Figure 28: out frequency of the "Counters" data

Each "Counters" data has:

- **myChannel_COUNT_t:** the time-stamp of the "Counters" data
- **myChannel_TRIG**: the number of events discriminated by the Trigger module of the MnM.
- **myChannel_CALC**: depends on the decimation parameter (cf. Figure 26).

$$\text{myChannel\_CALC} = \text{decimation} \cdot \text{myChannel\_TRIG}$$

- **myChannel_SENT**: the number of data from QT2T module, temporally stored in the FPGA buffers, and waiting for being sent to the computer.

The following inequality is always verified: myChannel_SENT≤ myChannel_CALC ≤ myChannel_TRIG.

These myChannel_TRIG, myChannel_CALC and myChannel_SENT are reset each time any module parameters is changing.

III.H.3 "Oscillo" data

The "Oscillo" data, already explained in chapter III.A , are defined in RHB by the variables:

- **myChannel_OSC_t**: the time stamp of the "0 ns" sample of the oscilloscope frame.
- **myChannel_OSC**: the Oscilloscope frame.

III.I Dead time

Depending on the considered data, the dead time is not the same.

III.I.1 Dead time of the counting data

III.I.1.a myChannel_TRIG

The dead time of the event discriminated by the Trigger module depends on the spreading of the input signal of the Trigger module and on the "inhibition" parameter:

$$DT_{trig} = \max(\,duration\ of\ the\ trigger\ module\ signal + 2\ ns, inhibition) \qquad \textbf{Equation III-1}$$

This dead time DT_{trig} is either expandable if the duration of the Trigger module is predominant, either non-expandable if the inhibit duration is greater than the signal duration of the Trigger module.

III.I.1.b myChannel_CALC

In decimation 1/1, the dead time of the event processed by the QT2T module is the same as triggered event:

$$DT_{calc} = DT_{trig}$$ **Equation III-2**

III.I.1.c myChannel_SENT

The dead time of data sent to computer depends of the data rate and the depth of FPGA buffers.

If the data rate is too high for FASTER acquisition, typically above 500 000 events per second, the dead time is not constant. It is equal to DT_{calc} at the beginning, (typically the first 10 s if the rate is under 3.10^6 events per second). As soon as the FIFO are full, all the data of all the channel belonging to the same µTCA carrier are sent with the rate of the Gigabit Ethernet communication. The FIFO are emptied in turn and a constant dead time is not guaranteed. The experimenter can increase the "QT2T" data decimation in order to reduce the rate of the sent events. He must likewise disable the sending of the oscilloscope data, or reduce the number of samples of the oscilloscope data frames.

III.I.2 Dead time of the "QT2T" data

If QT2T data decimation is 1/1 (Figure 26), the dead time of the "QT2T" data is equal to

$$DT_{calc} = \max(\, duration\ of\ the\ trigger\ module\ signal + 2\ ns, inhibit\ duration)$$ **Equation III-3**

If the data rate is greater than 500 000 events per second, some data can be lost. The user must take care to buffer depth of the computer. He can then check the match between the stored data number with myChannel_CALC and myChannel_SENT.

III.I.3 Dead time of the "Oscillo" data

The dead time of an oscilloscope frame is

$$DT = \max(oscilloscope\ frame\ duration + 88\ ns, oscilloscope\ module\ period)$$ **Equation III-4**

The oscilloscope duration is equal to 700 x time step and defined in Figure 6 ②.

The oscilloscope module period is defined in Figure 7.