X and Gamma Ray Electronics

XGLab S.R.L.
Via Conte Rosso 23, I-20134 Milano (Italy)
ph: +39 02 49660460
VAT and Tax Payer #: 06557660963
Capitale Sociale 50.000 euro I.V., R.E.A. MI-1899304
Direction and co-ordination: Bruker Italia S.r.l.

# Table of Contents

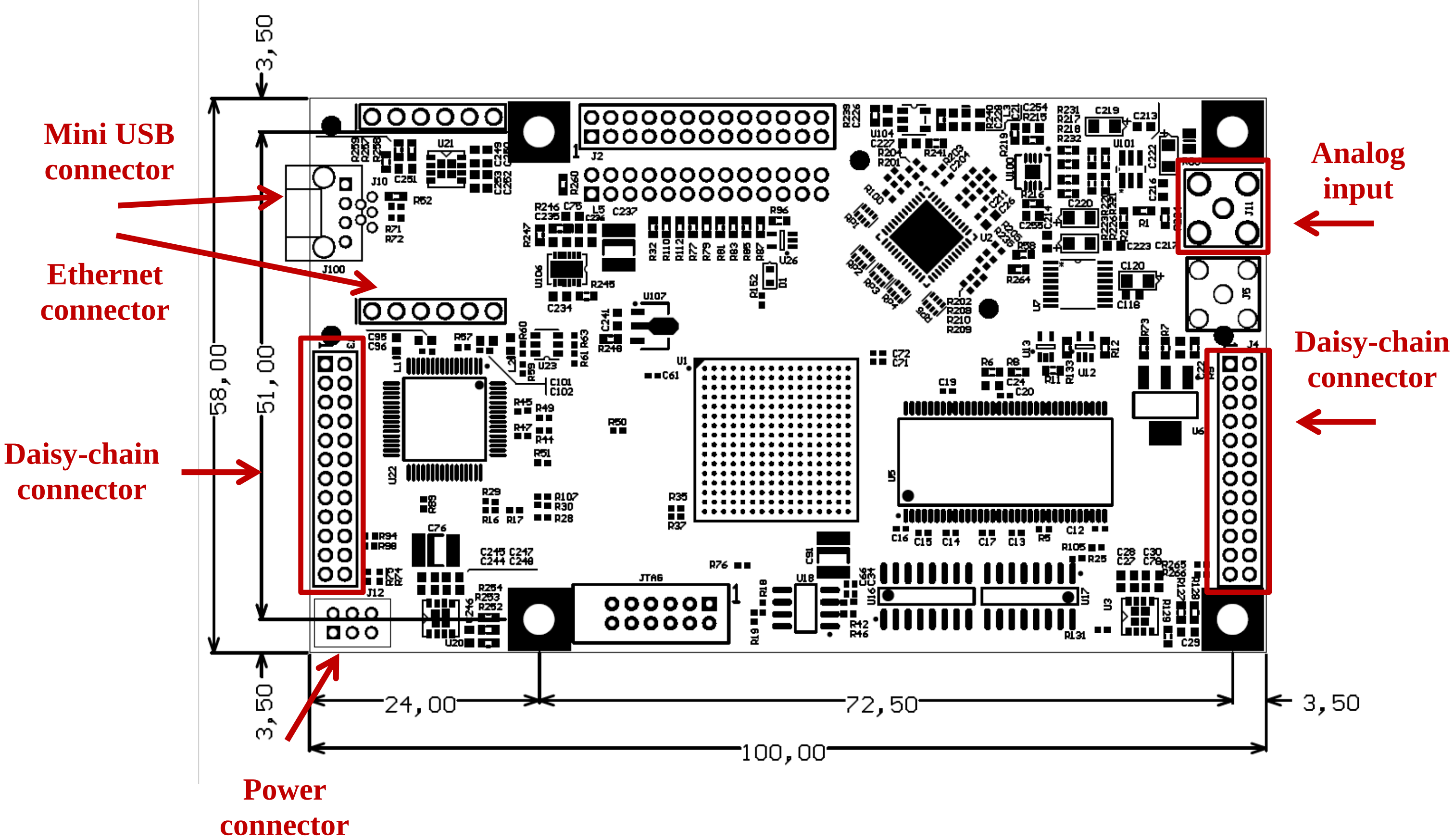
# DANTE Overview

The following document contains the description and practical information of the **DPP_4552** board. A separate manual describes the DLL library functions.

The **DPP_4552** board is designed to perform digital filtering of pulses and spectra acquisitions from an external charge preamplifier operating in pulsed reset mode. The filter parameters are completely configurable with specific DLL calls (detailed in the separate DLL manual). The board is equipped with a high performance 16-bit ADC which runs at 125 MSPS. The system can also acquire the analog waveform of the input signal that can be used as an excellent diagnostic tool.

The **DPP_4552** board receives command and transmits data to the PC over a USB 2.0 connection or an Ethernet cable using the TCP-IP protocol.

# Dimensions and connectors




The board dimensions are 100 mm x 58 mm.
Four mount holes for 3M screws are provided. With the coordinate reference on the bottom left (X, Y = 0, 0). The holes are placed at:

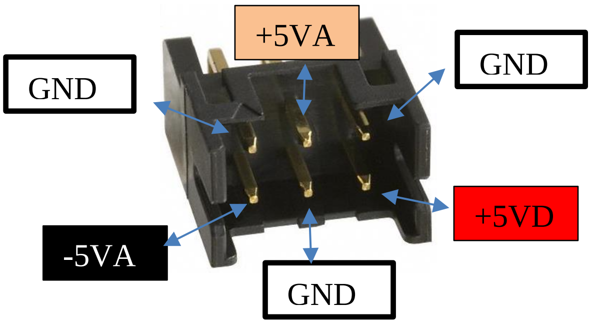
X= 24mm and X= 96.5mm
Y= 3.5mm and Y= 54.5mm

# DANTE DPP-4552 OEM Board

## POWER connector

The DANTE board requires a +5V digital supply (+5VD) and a bipolar +5V/-5V analog power supply (+5VA, -5VA): 400 mA are drown from +5VD and 50 mA from the +-5VA. Those directly powers some parts of the analog stage therefore, in order to achieve best resolutions, it is important that the power is stable and introduce less possible noise. The tolerances on power supplies are +-4%. Absolute maximum voltages are +-6V.

The mating part number for the power connector it's DF11-6DS-2C by Hirose. The pinout it's the following:




## Analog input

This is the connector for the input signal coming from external charge preamplifier. The **DPP_4552** board accepts a single-ended input from a coaxial LEMO connector. The maximum input span is 3Vpp with an high input impedance buffer.

Depending on the amount of offset applied (see 'Offset Configuration' part inside the User Interface section) the effective absolute input range varies. In particular:

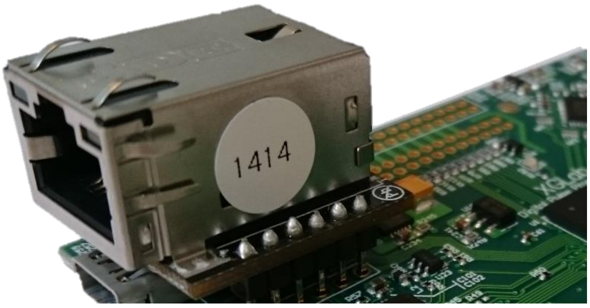
| | Offset setting | Input range |
|---|---|---|
| No offset applied | 128 | -1.5V ÷ 1.5V |
| Maximum negative offset | 0 | 0V ÷ 3V |
| Maximum positive offset | 255 | -3V ÷ 0V |

*Although the LEMO cable intrinsically carries a ground connection, depending on the setup it might be necessary (to achieve best resolutions) to connect the DPP_4552 and*
*the detector with a separate ground cable. In this regard on the DPP we suggest you to use the LEMO connector shield, which is grounded.*

### Mini USB connector

In order to receive commands and transmit data the **DPP_4552** board is equipped with a Mini USB connector that implements the USB 2.0 standard.

### Ethernet connector (Optional)




*Ethernet module mounted on the DPP_4552 board.*

The Ethernet connector can be used to entirely control the **DPP_4552** over a network using the TCP-IP protocol as an alternative to the USB connection.

### Daisy-chain connectors

The DPP_4552 boards can be easily connected in series using connectors in a daisy-chain fashion. In this mode only one board need to be connected to the PC (either via USB or within the LAN using the Ethernet connector) and that board will act as the master of the chain: it configures all the other boards depending on the commands received and receives the spectrum data from them. Thus, multiple channel can be paralleled needing only one connection.
After daisy chain connection, the boards need 2-3 seconds to initialize the communication and then a red LED on all the boards will indicate the successful initialization. After this moment, the slaves will be available and recognized by the DLL or the GUI.

# DANTE DPP-4552 Single Box


| Performance summary | |
|---|---|
| Box size and weight | 6.5cm x 11cm x 4cm, 0.4Kg |
| Power dissipation | ~2.5W |
| Analog Input | 8 analog input ( 2Vpp dynamic) *(coaxial SMA)* |
| Auxiliary IO | Auxiliary input for gating aquisition *(coaxial SMA)*, 2x10 GPIO connector |
| Interface | USB 2.0 \| TCP/IP 100BaseTX |

### POWER connector

The DANTE Single channel Box requires a +12V supply from which are drown maximum 250mA. The Box it's provided together with the AC Power supply adapter.

### Analog input

Same considerations stand for OEM board. (coaxial LEMO)

### Daisy-chain connectors

Daisy chain of multiple Single channel Boxes is not possible.

### Mini USB and Ethernet connector

8ch DANTE DPP box is equipped with both a Mini USB and Ethernet TCP/IP connector (100Mbit/s bandwidth). For the USB connection, the USB 2.0 standard is used.

### AUX GATING

The DANTE DPP is equipped with digital input labelled as AUX in the front panel which can be used as gating for the acquisitions. GATE function supports CMOS3.3 level. See the related section for further details.

# DANTE DPP-4552 8Ch Box


Following table summarize the main features of the 8ch DANTE DPP.

| Performance summary | |
|---|---|
| **Box size and weight** | 20 cm x 11 cm x 11 cm, 1.7Kg |
| **Power dissipation** | ~25W |
| **Analog Input** | 8 analog input ( 2Vpp dynamic) *(coaxial SMA)* |
| **Auxiliary IO** | Auxiliary input for gating aquisition *(coaxial SMA)* |
| **Interface** | USB 2.0 \| TCP/IP 100BaseTX |
| **Scalability** | Daisy-chain connection |

### POWER connector

The 8ch DANTE DPP requires a 25V-1A power supply (XP Power AFM45US24). In order to achieve a solid mechanical power connection, the box is equipped with the 3-pole KFV 30 connector. A custom cable with a SV30 connector is supplied together with the box. SV30 connector can be directly fasten into its KFV 30 housing.
The current AC/DC converter used for the power supply is the 36W (25V-1.5A max) AED36US24 from XP Power.

### Mini USB and Ethernet connector

8ch DANTE DPP box is equipped with both a Mini USB and Ethernet TCP/IP connector (100Mbit/s bandwidth). For the USB connection, the USB 2.0 standard is used.

### Analog input

The 8ch DANTE DPP accepts 8 single-ended inputs from a coaxial SMA connector. The maximum input span is ~3Vpp with a high input impedance buffer.

*Although the SMA cable intrinsically carries a ground connection, depending on the setup it might be necessary (to achieve best resolutions) to connect the DANTE DPP and the detector with a separate ground cable. In this regard on the 8ch DANTE DPP we suggest you to use the banana plug which is connected to the ground of the DANTE system.*

### AUX GATING

The DANTE DPP is equipped with digital input labelled as AUX in the front panel which can be used as gating for the acquisitions. GATE function supports CMOS3.3 level. See the related section for further details.

### Daisy-chain connection

The DANTE DPP box is intrinsically made up of different DANTE DPP-4552 board connected in a daisy-chain fashion.

Furthermore, the DANTE DPP box can be connected in daisy-chain itself by stacking different boxes using the top and bottom panel DSUB25 connectors. In this mode only one box need to be connected to the PC (either via USB or within the LAN using the Ethernet connector) and that box will act as the master of the chain: it configures all the other boxes depending on the commands received and receives the spectrum data from them. After a daisy chain connection, the system need about 5 seconds to initialize the communication and recognized the new 8 channel DANTE DPP by the DLL or the GUI.

NOTE 1

*In case of high data throughput (e.g. map or waveform acquisition), it can be useful to use a separate USB 2.0 (or ethernet) connection for each box even if they are connected in daisy-chain. The daisy-chain connection still remains useful if the time-stamp synchronization is required.*

NOTE 2

*In order to update the firmware, the user need to remove the daisy-chain connection between boxes. The firmware update should be done individually for each box.*

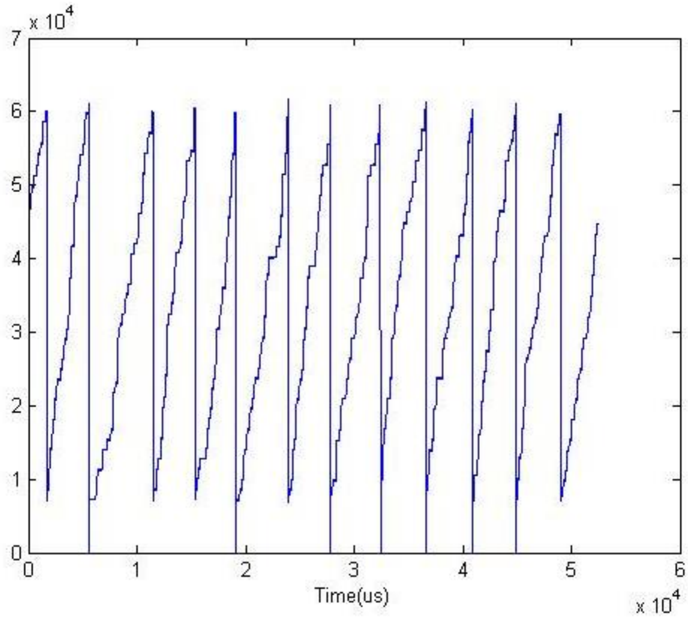
# Usage Guidelines

In this section are described the various steps the user should follow to correctly configure the DPP parameters via USB. First, initialize the DLL as described in the **XGL-DLL - DANTE library** manual, then:

1. Place the Fe55 source near to the detector in order to have a slightly high input count rate and launch some waveform acquisitions. Use the maximum decimation of samples and MSB mode to easily detect a reset event and check that the ADC do not saturates (ADC output values are from 0 to 65535). To do that call this functions:

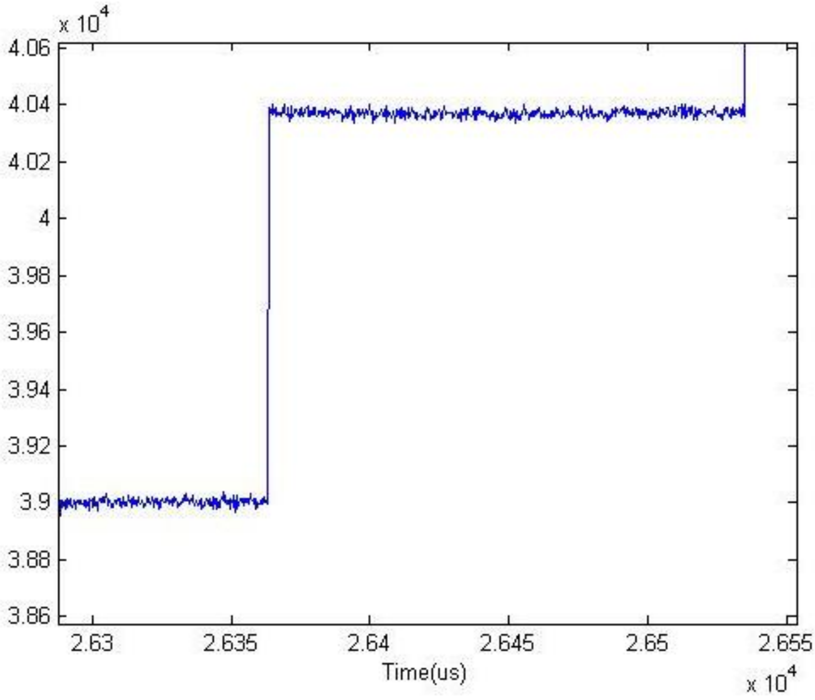
```
start_waveform(serial, 0, 32, 1, 30000, 1, 10);
double values[4096];
unsigned int spectra_size = 4096;
getData(serial, spectra_size, 0);
```

   Note that with this decimation the actual sampling rate is 62.5 MHz divided by 32. The data acquired should look like this:




   Then in this same acquisition focus on the instants right after the reset event to choose the proper setting of the **reset recovery time**. It should be chosen large enough to have the reset transient finished before this time elapses.

2. Focus on a event in the same waveform acquired previously:




From his height and choosing the energy dynamic (remembering that the spectra have 4096 bins) you can retrieve the desired the **digital gain** setting. For example, in upper plot the height of the highlighted event is about 1380 LSBs and it is assumed as an X-ray of 6 keV (Mn-K$\alpha$).

A typical choice is to set this peak at the 600th bin in the spectra, so that the energy calibration is approximately 10 eV/bin. To achieve that, the user should set the digital gain at approximately 600/1380 = 0.43 (BIN/LSB).

However, note that changing the digital gain will have no impact on resolution.

3. Select the desired peaking time and flattop of the main filter and the peaking time of the edge detection filter. The **flattop** (or gap) is the time duration to allocate the rise-time of the signal so it has to be chosen longer than the maximum rise-time of the signal: if it's set too low the spectra will have worst resolution at shorter main filter peaking times and also if set too high the resolution will (more gradually) get worse. The **peaking time of the fast filter** has to be shallower than the **main filter peaking time**. We recommend to use at least a fast filter peaking time of 2 and not excessively faster than the main filter peaking time. Also the **flat top of the fast filter** can be set to 1, and may be increased if the user is interested to very low energy elements. Disable pileup rejection setting the max risetime to 0 and choose a proper value for the detection threshold (in spectrum bin unit). Along with the digital gain (0.43 bin/LSB) and reset recovery time (3us) chosen at the previous points, you can now call the function `configure`:

```
configuration cfg;
cfg.fast_filter_thr = 20;
```

```
cfg.max_risetime = 0;
cfg.gain = 0.43;
cfg.peaking_time = 25;
cfg.flat_top = 5;
cfg.edge_peaking_time = 1;
cfg.edge_flat_top = 1;
cfg.zero_peak_freq = 1;
cfg.baseline_samples = 64;
cfg.inverted_input = false;
cfg.reset_recovery_time = 375;
};

configure(serial, board, cfg);
```

Then start the spectrum acquisition and retrieve data:

```
start(serial, 1, 4096);
unsigned long long int values[4096];
unsigned long int id;
statistics stats;
unsigned int spectra_size = 4096;
unsigned short acq_status;
getData(serial,0,values,id,stats,spectra_size,acq_status);
```

4. **Fast filter threshold.** Note that if it is set too low random noise spikes are wrongly counted as events and the typical noise peak would appear. Instead, if the detection threshold is set too high, real events may be discarded and baseline measurement could result inaccurate.

   The proper procedure to choose the threshold is to reduce it gradually and stop just when the noise peak appears (of course deactivating first the zero peak). With the digital gain and edge detection peaking time chosen, a typical value of the threshold is 20 bins that correspond to about 200 eV in this example.

5. **Maximum risetime.** This setting, expressed in 8 ns samples, it's used for the pileup rejection. It has to be set accordingly to the maximum risetime of the detector. If the precise risetime is not precisely known we recommend to be conservative and choose a larger number, although if it is set too high pileup rejection would be suboptimal.

   Inversely, if the maximum risetime is set too low most of the events are discarded as falsely recognized affected by pileup. This can be noted at low input count rates at which pileup are unlikely to occur.

   If the maximum risetime is set to 0, the pileup rejection on the fast filter is disabled. The pileup rejection on the main filter (energy filter) instead cannot be currently disabled (that is when two events seen by the fast filter are too near to each other with respect of the energy filter length).

6. **Baseline samples.** This is a setting for the baseline correction. In previous firmware version this was in a total automated way with no input required by the user. Now, the algorithms have been improved and to achieve optimal performances with different detectors a setting from the user is required.

   It has to be chosen among this values: 16,32,64,128,256,512 (32 ns units). The best setting is equal to the optimal energy filter peaking time. This defines the length of the portion where the baseline is calculated.

   The calculation of the baseline is directly applied on the output of the energy filter in order to correct shifts of the energies due to temperature or count rate. So the precision of the baseline value subtracted is critical to achieve best performances, as noise on this signal would directly degrade the resolution.

   The noise of this calculation is similar to the one of the trapezoidal filtering done by the energy filter: if the length is set too low the calculation is very noisy mainly due to white noise contribution, if set too high it again become noisy because of the low frequency noise (1/f). So, as for the energy filter, there is an optimum and it comes out that the optimum is when the baseline length is similar to the optimal peaking time of the energy filter.

   For debugging purposes, it is possible to disable the baseline correction by setting baseline samples to 0. This usually lead to optimal energy resolution performances (because nothing is subtracted from the energy filter output, so no noise is added to it) but of course the spectrum will be subject to energies shifts due to temperature or input count rate changes.

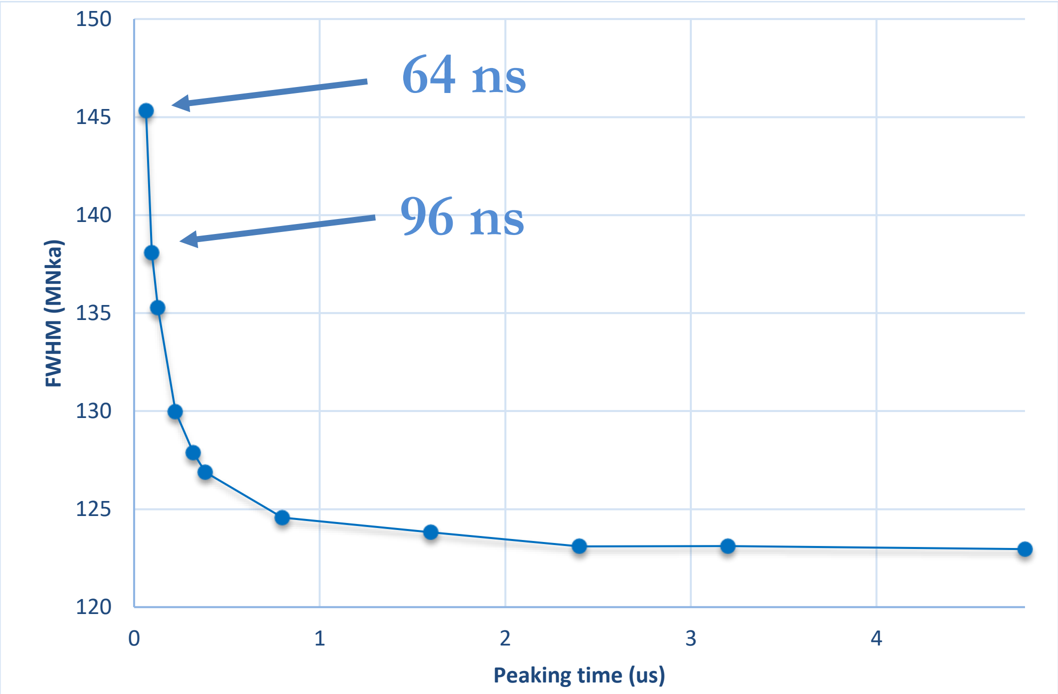
7. **Zero peak rate.** This setting enables the zero peak in the spectrum at bin #96. This peak is needed in the user interface for peak fitting and may be useful in general for precisely calibrate each recorded spectrum.

# Performances Example 1

In this section are presented achievable performances in terms of resolution. These measures have been done with the following setup and settings:

- **Source**: Fe55;
- **Detector**: 7 mm$^2$ Ketek's VITUS SDD;
- **Preamplifier**: Cube (~12mV/keV);
- **Temperature**: -38°C;
- **Flat top**: 2 (64 ns);
- **Fast filter threshold**: 70 spectrum bins;
- **Digital gain**: 0.43;
- **Reset recovery time**: 200 (1.6 µs);
- **Fast filter peaking time**: 1 (8 ns).
- **Fast filter flat top**: 1 (8 ns).

The first graph shows the MnKa resolution at various peaking times. The ICR was about 8 kcps.

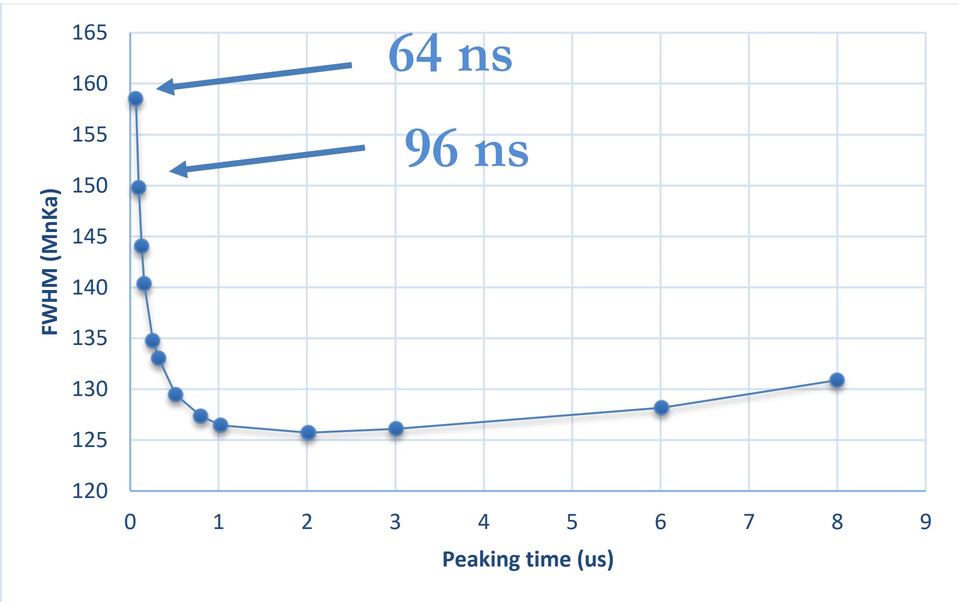


# Performances Example 2

In this section are presented achievable performances in terms of resolution. These measures have been done with the following setup and settings:

- **Source**: Fe55;
- **Detector**: 20 mm$^2$ Ketek's VITUS SDD;
- **Preamplifier**: Cube (5mV/keV);
- **Temperature**: -35°C;
- **Flat top**: 9 (288 ns);
- **Fast filter threshold**: 25 spectrum bins;
- **Digital gain**: 1;
- **Reset recovery time**: 200 (1.6 μs);
- **Fast filter peaking time**: 5 (40 ns).
- **Fast filter flat top**: 1 (8 ns).

The first graph shows the MnKa resolution at various peaking times. The ICR was about 20 kcps.

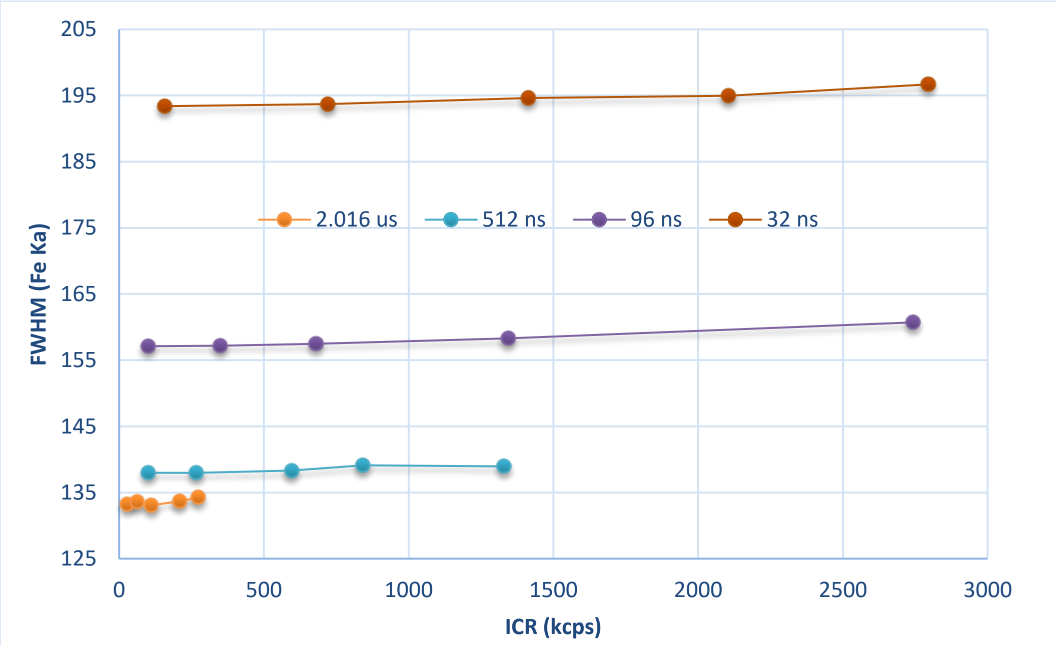


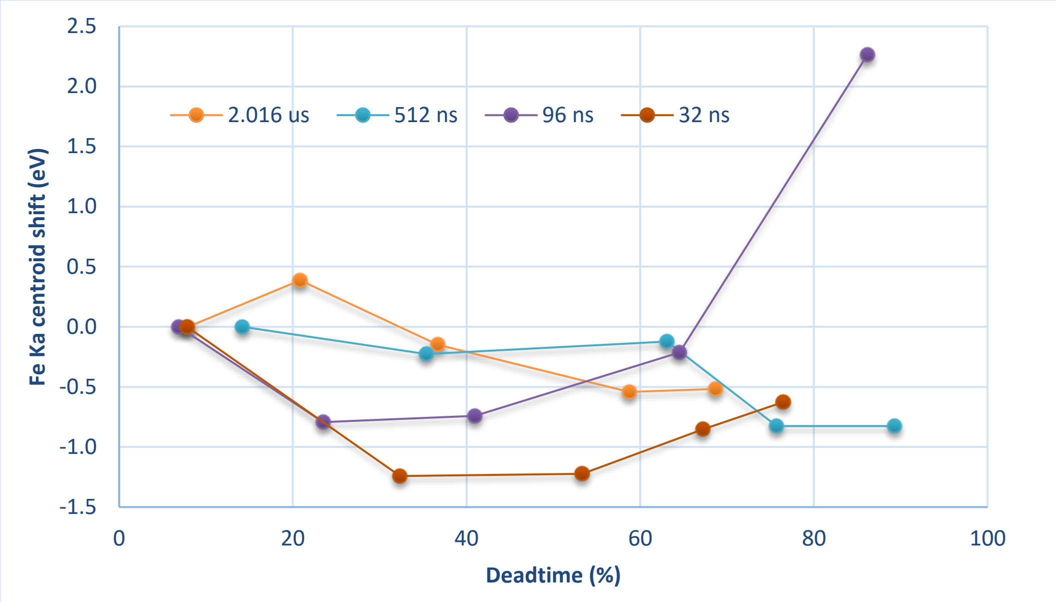

# Performances Example 3

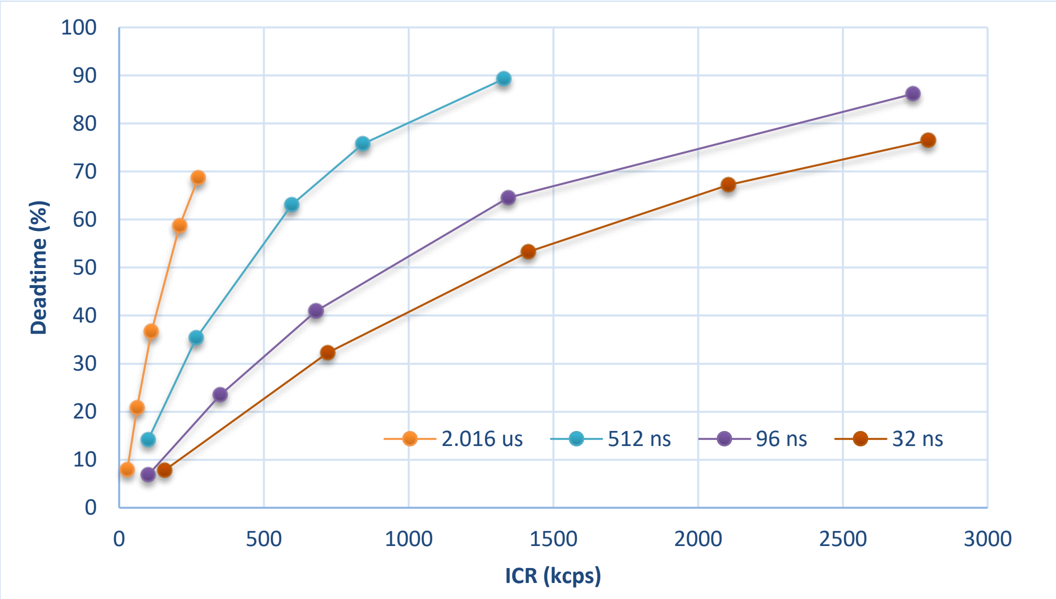
In this section are presented achievable performances at high count rates. To obtain the following data, it has been done an XRF measure with an X-Ray tube with a Fe target (not Fe55).

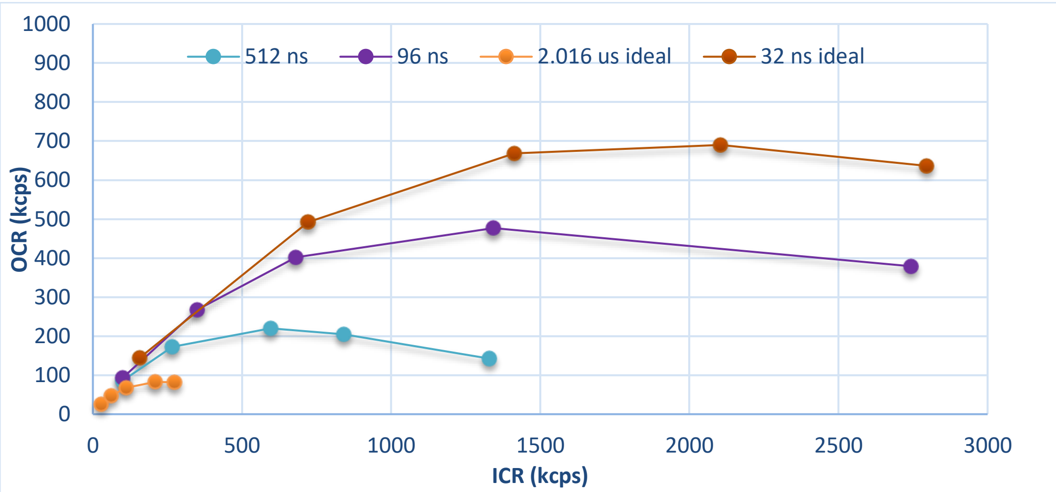
- **Sources**: X-Ray tube with Fe target (6.405 keV);
- **Detector**: 20 $mm^2$ Ketek's VITUS SDD;
- **Preamplifier**: Cube (5mV/keV);
- **Temperature**: -38°C;
- **Flat top**: 7 (224 ns);
- **Digital gain**: 1;
- **Reset recovery time**: 200 (8 ns samples);
- **Fast filter peaking time**: 31 (for energy filter PT of 2.016 us), 3 (for energy PT of 512 and 96 ns), 2 (for energy PT of 32 ns);
- **Fast filter flat top**: 10 (for energy filter PT of 2.016 us), 1 (for others);
- **Fast filter threshold**: 10 (for energy filter PT of 2.016 us), 40 (for energy PT of 512 and 96 ns), 50 (for energy PT of 32 ns);
- **Max risetime**: 30 (8 ns samples).

In the second plot are compared the obtained output count rates with respect of ideal curves from theory taking into account the deadtime due to reset recovery time.




2.016 us
512 ns
96 ns
32 ns
Fe Ka centroid shift (eV)
2.5
2.0
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
0
20
40
60
80
100
Deadtime (%)

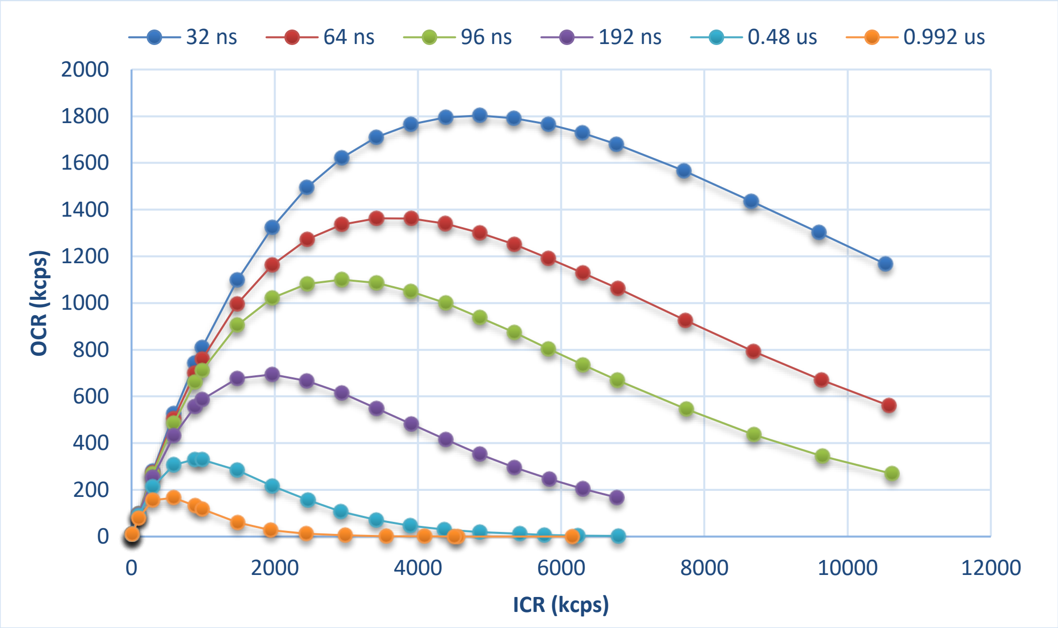

Deadtime (%)
100
90
80
70
60
50
40
30
20
10
0
2.016 us
512 ns
96 ns
32 ns
0
500
1000
1500
2000
2500
3000
ICR (kcps)


OCR (kcps)
1000
900
800
700
600
500
400
300
200
100
0
512 ns
96 ns
2.016 us ideal
32 ns ideal
0
500
1000
1500
2000
2500
3000
ICR (kcps)

# Performances Example 4

In this section are presented achievable performances at very high count rates performed with emulated randomly distributed monoenergetic events.

- **Flat top**: 2 (64 ns);
- **Fast filter threshold**: 100 spectrum bins;
- **Digital gain**: 1;
- **Reset recovery time**: 0.48 μs;
- **Fast filter peaking time**: 1 (8 ns).



# Matlab based GUI

## Introduction

In order to quickly evaluate the DANTE DPP performances, it is provided a standalone graphical user interface created in a MATLAB environment. The application does not need to be installed but the MATLAB Compiled Runtime (MCR) version 8.3 (R2014a – x86) is required. This component is available for free download at:

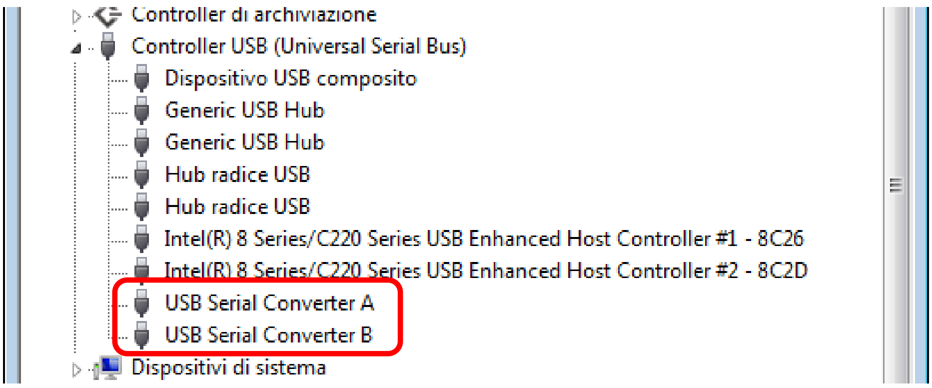
http://it.mathworks.com/supportfiles/downloads/R2014a/deployment_files/R2014a/installers/win32/MCR_R2014a_win32_installer.exe

In order to use the GUI and the DLL you need to install the Visual C++ Runtime 2015 (x86). You will find the installer (vc_redist.x86.exe) in the provided package.

The minimum system requirements are:

- Operative system: Windows Vista or later x86 or x64 (it could not work properly in Virtual Machines environments).
- An USB 2.0 port.
- Ethernet link supporting TCP-IP 100BaseTX or higher.

If an Internet connection is available, the USB drivers will be automatically found and installed by Windows Update the first time the board is plugged to the PC. Otherwise you can also find the driver in the provided package (CDM21214_Setup.exe) and install it manually before connecting the board. In some OS you may need to install this driver twice. If the board is correctly detected by the system you should see under the Device Manager two devices named USB Serial Converter A and B:



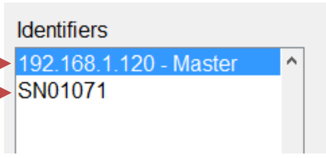
# Initialization

The library is automatically initialized when DANTE.exe is opened and if a DANTE board is connected it will display its identifier in the upper left corner. The identifier should appear in the form SN00XXX (the serial number) for USB connected devices or the IP of the TCP-IP devices. Otherwise, if the board is not connected or a communication problem has occurred, it will be displayed 'Searching DPPs…'. The boards can be connected at any time.

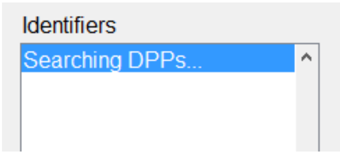
> **Warning**
> The interface can be launched even if no DPP is connected but it needs the USB drivers installed first, otherwise there will be errors in initialization. To do so you can connect the DANTE DPP board and let Windows to locate the drivers from Internet (preferred) otherwise manually install the package provided in the Drivers folder.

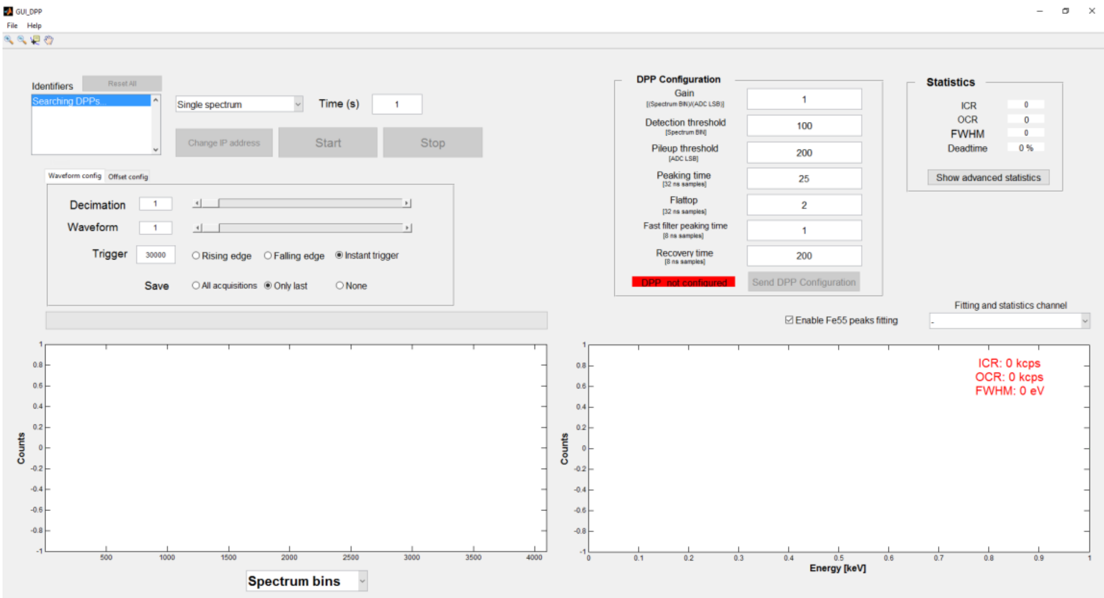
Boards connected

TCP-IP device
USB device



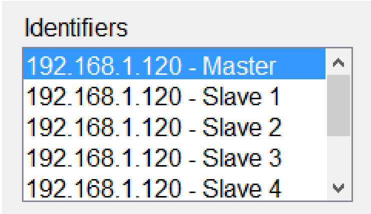

No board connected



## Daisy-chain

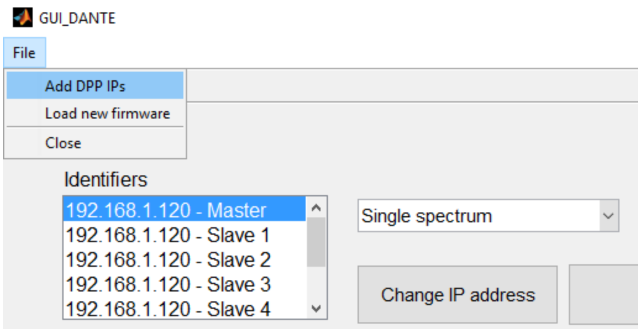
If a TCP-IP or USB device is connected to other boards using the daisy chain connectors and cables, also the slaves will be shown in the GUI. This will happen only after the slave communication has been initialized, which is signaled thought red LEDs on the boards.



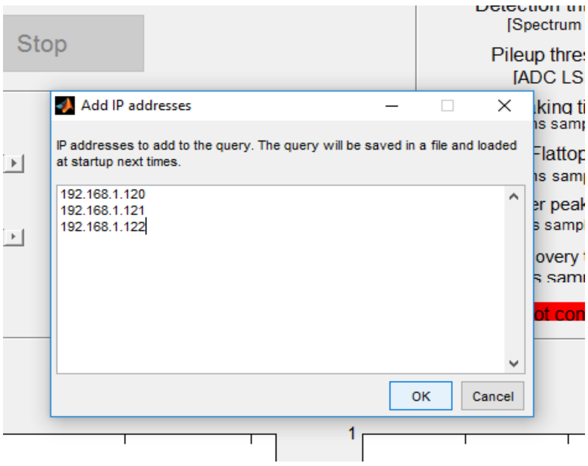

## TCP-IP Query

USB devices are automatically recognized by the GUI when connected. Regarding Ethernet devices instead the GUI need to know their IP first. The user can add IPs to a list, named query, and then the GUI will try connecting to them. If the connection succeeds, you will see the corresponding board IP in the Identifiers list.

To add IPs to the query use the 'Add DPP IPs' function in the drop down menu:



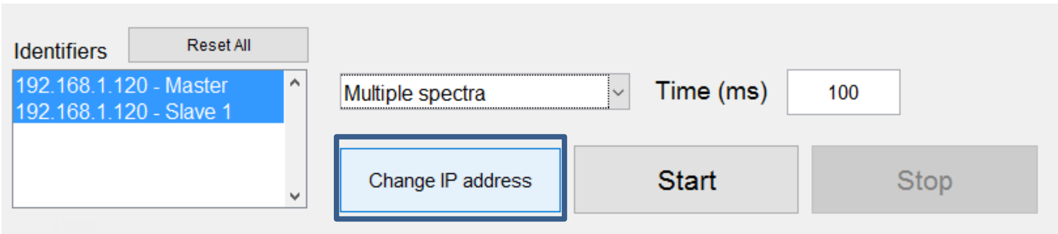

Then you will be asked to insert the desired IPs:

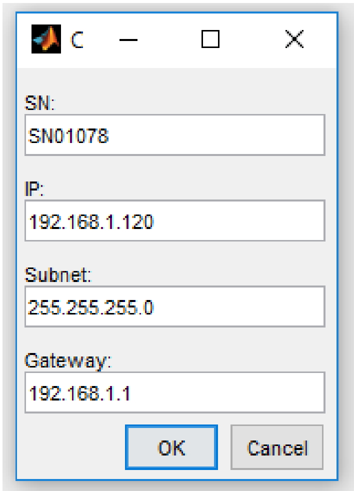


After the query has been updated it will be saved in the main directory a file named *query.txt* which contains the inserted IPs. The next times the GUI will be launched it will automatically add to the query the IPs contained in *query.txt*. So you can also modify this file prior to launch the GUI if the desired IPs changed.

## Change IP settings

Each board will initially have by default the IP 192.168.1.120. In other to change it, together with the subnet mask and the gateway you have to connect the DPP thought USB. Then selecting the board's identifier you can push the 'Change IP address' button and modify the IP settings in the popup window.

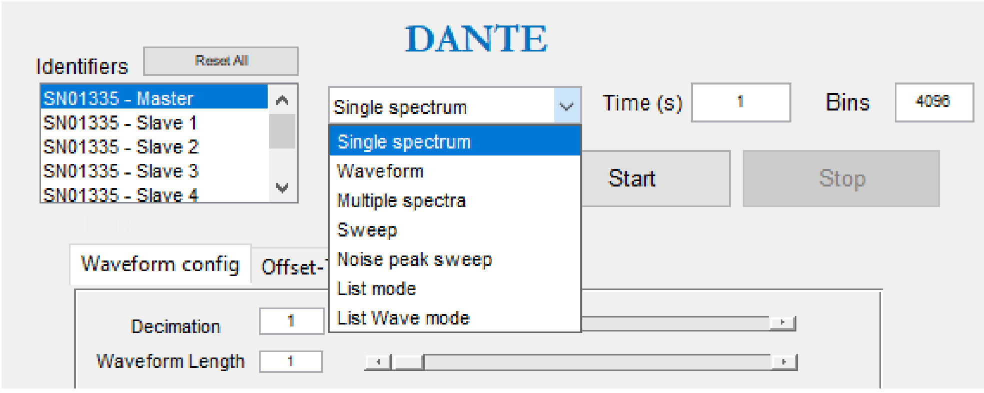



**Warning**
The IP will be not be set immediately but the change will take effect after a board power cycle.

## Select acquisition mode

To start using the DPP choose an acquisition mode:

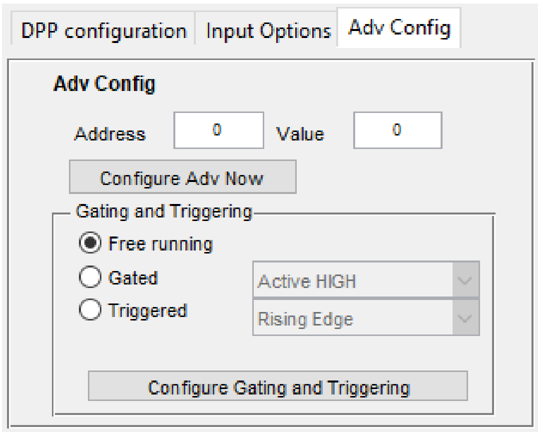



The available acquisition modes are: Single spectrum, Waveform, Multiple spectra, List mode, Sweep, Noise peak sweep, List Wave mode.

<u>The interface can control multiple DANTE DPP at the same time</u> and save acquisition data separately for each board recognized. To select multiple DPPs just press CTRL+Left click on the desired boards and then launch the acquisitions.

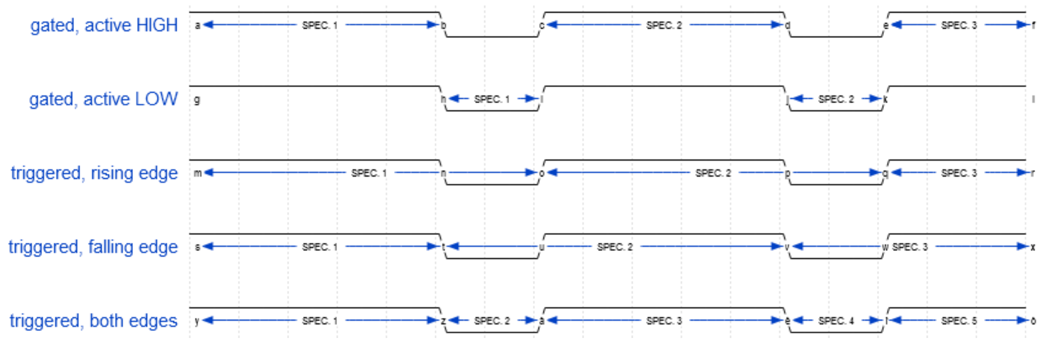
## GATE/TRIGGERED acquisition

DANTE DPP can be configured to acquire accordingly to an external logic (BNC) input as advancing pixel method. Next figure shows the gating setting panel, accessible from the "Advanced option" panel of the GUI, and only affects *multiple spectra* acquisitions (see below).




Three possible gating modes can be selected for map acquisitions: "free running", "gated" and "triggered". In "free running" acquisitions, the BNC input is ignored by the DPP, and a new spectrum is acquired every time the user-defined timeout elapses. When the "gated" mode is selected, a new spectrum is acquired every time the BNC input is either low or high (depending on "Active LOW/Active HIGH" setting), and the DPP is kept inoperative while the input is inactive. Finally, in "triggered" mode the DPP is always active, and a new spectrum is initialized every time an edge on the input

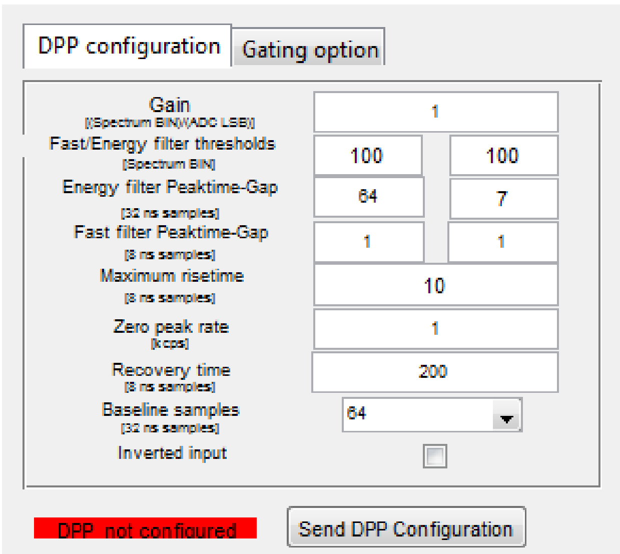
signal occurs; depending on the corresponding setting, the new spectrum can either be initialized on rising edges, on falling edges, or on both rising and falling edges.
Each gated/triggered operating mode is schematized in the following figure, where the active period for each spectrum is highlighted with blue arrows.




# DPP Configuration

All the DPP parameters are located in the DPP Configuration box along with the corresponding units. Push the button Send DPP Configuration to configure the DPP.

## Default DPP Mode

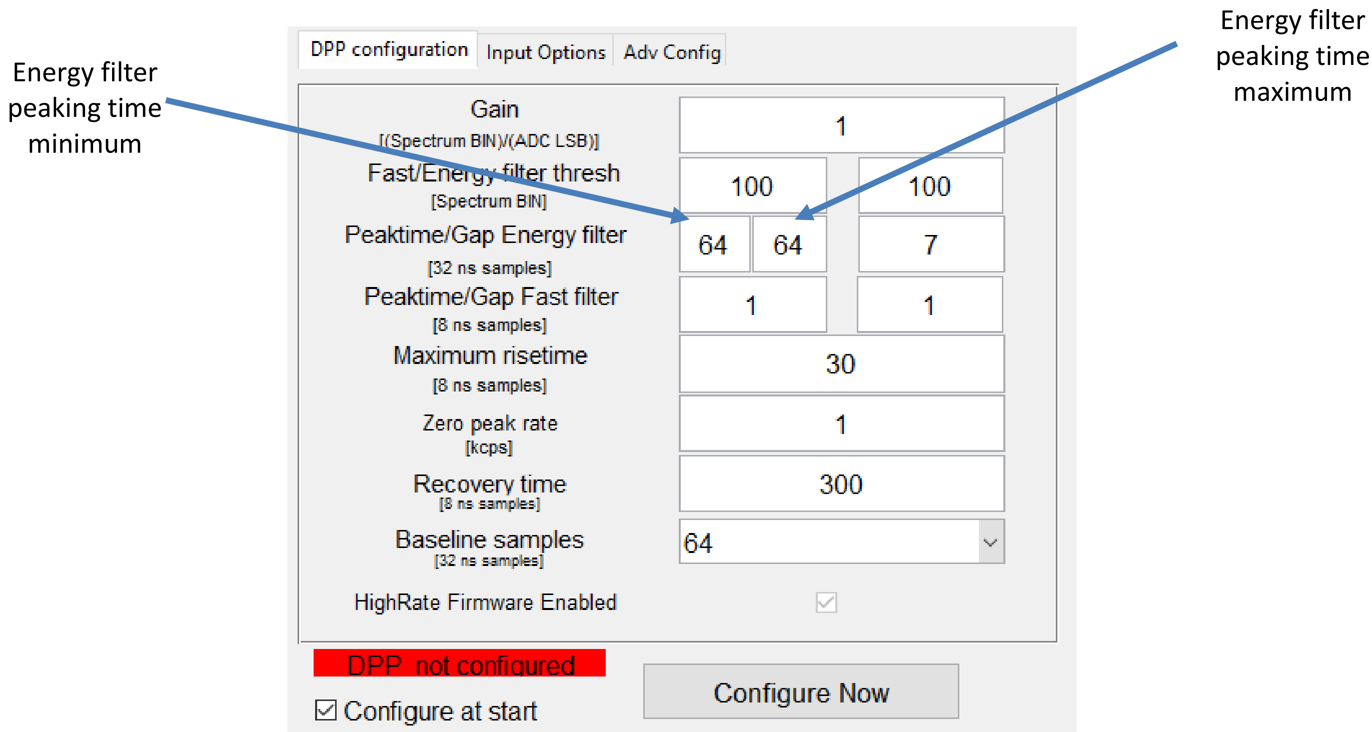



Following table summarize the selectable range for the DPP parameters.

| Parameter | Unit | Range |
|---|---|---|
| Gain | Spectrum_BIN/ADC_LSB | 0.01 to Peaking_time*2-0.01 |
| Fast Filter Threshold | Spectrum_BIN | 0 to 4096 |
| Energy Filter Threshold | Spectrum_BIN | 0 to 4096 |

| | | |
|---|---|---|
| Maximum risetime | 8 ns sample | 0 to 127 |
| Energy filter Peaking time | 32 ns sample | 1 to 511 |
| Energy filter Flat top | 32 ns sample | 1 to 15 |
| Fast Filter Peaking time | 8 ns sample | 1 to 31 |
| Fast Filter Flat Top | 8 ns sample | 1 to 31 |
| Baseline samples | 32 ns sample | 0,8,16,32,64,128,256 or 512 |
| Zero Peak Rate | kcps | 1 to 501 |
| Recovery Time | 8ns sample | 0 to 2^24-1 |
| Inverted Input | - | [0:1] |

## HighRate DPP Mode




Following table summarize the selectable range for the DPP parameters in case of high rate mode.

| Parameter | Unit | Range |
|---|---|---|
| Gain | Spectrum_BIN/ADC_LSB | 0.01 to Peaking_time*2-0.01 |
| Fast Filter Threshold | Spectrum_BIN | 0 to 4096 |
| Energy Filter Threshold | Spectrum_BIN | Not implemented |
| Maximum risetime | 8 ns sample | 0 to 127 |
| Energy filter Peaking time minimum | 32 ns sample | 1 to (128 - Energy Flattop) |
| Energy filter Peaking time maximum | 32 ns sample | 1 to (128 - Energy Flattop) |
| Energy filter Flat top | 32 ns sample | 1 to 15 |
| Fast Filter Peaking time | 8 ns sample | 1 to 31 |
| Fast Filter Flat Top | 8 ns sample | 1 to 31 |
| Baseline samples | 32 ns sample | 0,8,16,32,64,128,256 or 512 |
| Zero Peak Rate | kcps | 1 to 501 |

Please note that:

- FlatTop must be an odd number
- Following formula must be followed:

$$PeakingTime_{MAX} < \frac{FlatTop}{2} + 1$$

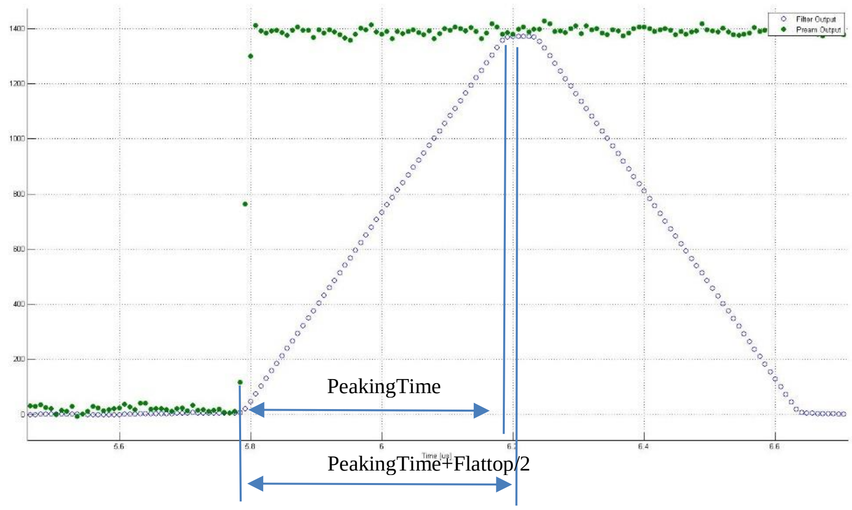
## Parameters description

### Gain

Digital gain applied to the ADC samples. Effect of gain value is to change the dynamic range of the spectrum. Does not improve energy resolution of the spectrum.

### Energy filter Peaking-time

Filtering of the digitalized waveform is done through a trapezoidal filter as shown in the following picture. Both the peaking-time and the gap of the energy filter are expressed in multiple of 32 ns. The Energy filter threshold is currently not implemented.

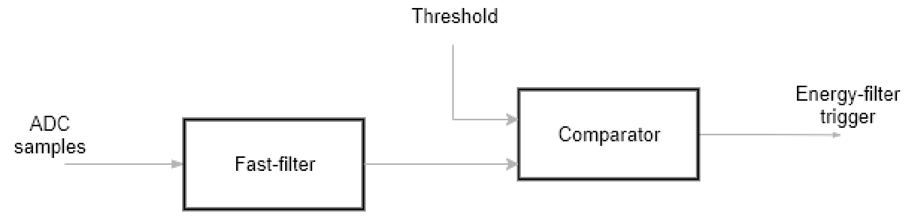



*Trapezoidal filtering with peaking time = 50 and flattop = 6*

### Fast-Filter threshold and Fast-Filter Peaking-time

Detection of events is done by a combination of the Fast-filter and the fast filter threshold. Fast filter output is compared with the set threshold: if higher, the event will be then processed by the Energy filter, otherwise the event is discarded. Like the energy filter, both peaking time and gap can be selected for the fast filter.

Since the fast filter threshold is not simply a cut of the reconstructed spectrum, it can happen that noise peaks whose energy (expressed in bin) is less than the threshold, appear in the spectrum. This is true for noise or interference. Since they are not real steps in the ramp, a short fast filter can have an output which is higher than the detection threshold; then the longer energy filter, filters out the spikes reconstructing an energy which can be less than the threshold.




### Zero peak rate

Enabling the zero peak rate, the DPP will acquire the filter output periodically when there a no input events. This feature can be disabled setting the rate to 0.

### Maximum risetime

This setting, expressed in 8 ns samples, it's used for the pileup rejection. It has to be set accordingly to the maximum risetime of the detector. If the precise risetime is not precisely known we recommend to be conservative and choose a larger number, although if it is set too high pileup rejection would be suboptimal.

Inversely, if the maximum risetime is set too low most of the events are discarded as falsely recognized affected by pileup. This can be noted at low input count rates at which pileup are unlikely to occur.

If the maximum risetime is set to 0, the pileup rejection on the fast filter is disabled. The pileup rejection on the main filter (energy filter) instead cannot be currently disabled (that is when two events seen by the fast filter are too near to each other with respect of the energy filter length).

### Baseline samples

This is a setting for the baseline correction. In previous firmware version this was in a total automated way with no input required by the user. Now, the algorithms have been improved and to achieve optimal performances with different detectors a setting from the user is required.

It has to be chosen among this values: 16,32,64,128,256,512 (32 ns units). The best setting is equal to the optimal energy filter peaking time. This defines the length of the portion where the baseline is calculated.

The calculation of the baseline is directly applied on the output of the energy filter in order to correct shifts of the energies due to temperature or count rate. So the precision of the baseline value subtracted is critical to achieve best performances, as noise on this signal would directly degrade the resolution.

The noise of this calculation is similar to the one of the trapezoidal filtering done by the energy filter: if the length is set too low the calculation is very noisy mainly due to white noise contribution, if set too high it again become noisy because of the low frequency noise (1/f). So, as for the energy filter, there is an optimum and it comes out that the optimum is when the baseline length is similar to the optimal peaking time of the energy filter.

For debugging purposes, it is possible to disable the baseline correction by setting baseline samples to 0. This usually lead to optimal energy resolution performances (because nothing is subtracted from the energy filter output, so no noise is added to it) but of course the spectrum will be subject to energies shifts due to temperature or input count rate changes.

### Recovery time

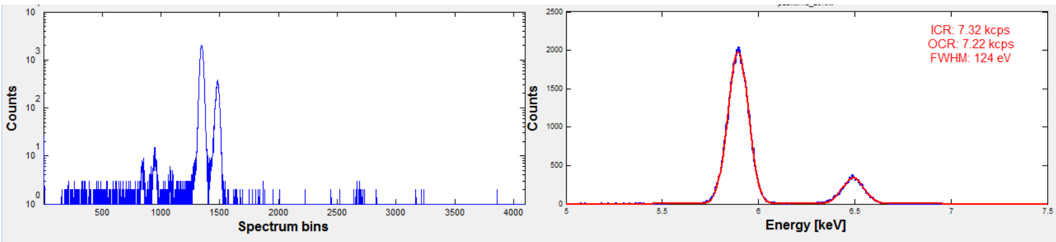
After the detection of the reset, following events are ignored for a time set by the recovery time. Recovery time is expressed in multiple of 8 ns

### Inverted input

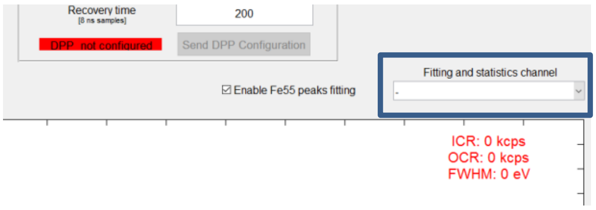
To be enabled in case of negative output ramp from the detector.

# Single spectrum

In this mode the user can acquire a single spectrum specifying the acquisition time and the DANTE DPP parameters.
In the interface, there is also the possibility to automatically fit the Fe55 peaks and calculate the FWHM.




The statistics of the acquisition are located in the upper right part of the GUI.

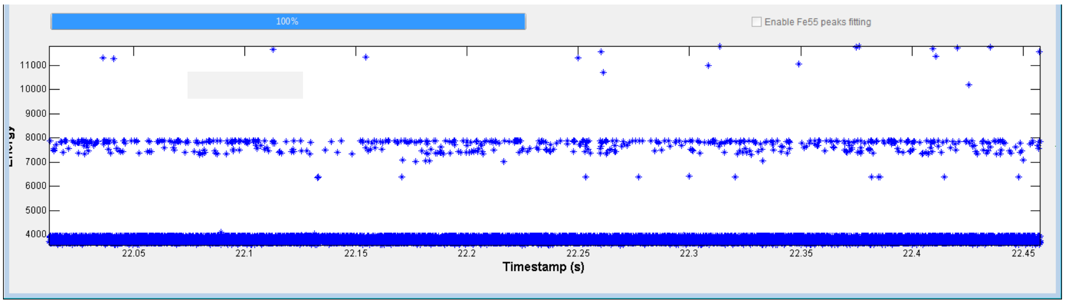



In case the Fe55 peaks fitting is enable, the GUI will show the fitting of just one channel. Anyway, at the end of the single spectrum acquisition, the user can change which fitting plot to show. Changing the fitting plot will automatically change also all the statistics information shown.

The spectrum is automatically saved inside a folder named 'measures' that will be created in the path of the executable.

# List mode

In this mode, the user can launch single spectra and the DPP will output all the recorded energies together with the timestamp of their arrival (in 8 ns samples). The setup of this mode is the same as in Single spectrum mode and, once started, the interface will begin to save the recorded events in the *measures* folder in different .mat files together with the most updated statistics for each packet.




The number of events contained in each file is at least the number in the *Events per file* textbox (except for the last ones which will contain all the events available). Then the user can choose the plot refresh rate and the maximum number of events to plot each time.

The USB connection can handle OCR up to 2.5 Mcps while the Ethernet connection up to 1.2 Mcps. If this rates are exceeded, events will likely be lost, and this will be indicated in the Lost counts (in the Advanced statistics tab) and the error code 82 (hardware memories full) will probably appear too.

**Warning**
Currently the timestamp counter is 44 bit wide and so will overflow after about 1.6 days. This timestamp is reset after every configuration of the DPP and then goes in free-running, even if no acquisition is in progress. This counter is also independent from board to board.

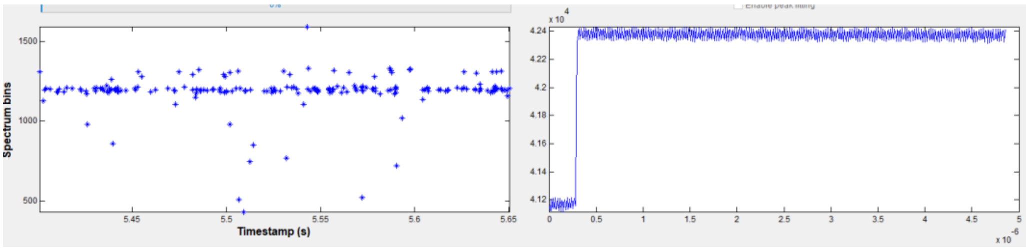
The energy has 16 bit resolution instead of the standard 12 bits used for the spectrum. So are added 2 LSBs and 2 MSBs to the energy information. Both for plot and data saving the actual energy received from the DPP is divided by 4 to be directly comparable with spectrum acquisition. In this way the energy information expressed in bin will likely have decimals.

The main purpose of this GUI mode is to save all the data as fast as possible, in order to be able to handle very high OCRs. For this reason not all the counts can be displayed otherwise the process would become very slow. When 1/plot_refresh_rate has elapsed from the last plot the GUI will display a bunch of max_events_in_plot events.

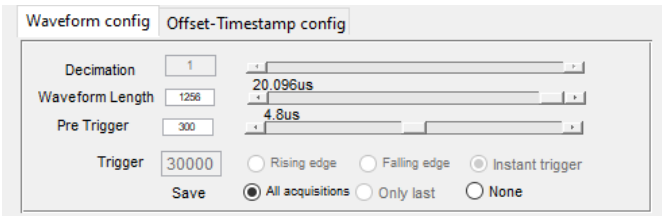
Timestamp of each DPP connected in the chain is synchronized with a precision of ± 8ns.

# List Wave mode

List Wave acquisition mode allows to record, together with energies and timestamps, the digitalized waveform associated to the event. The following picture shows an example for a single channel DANTE DPP.



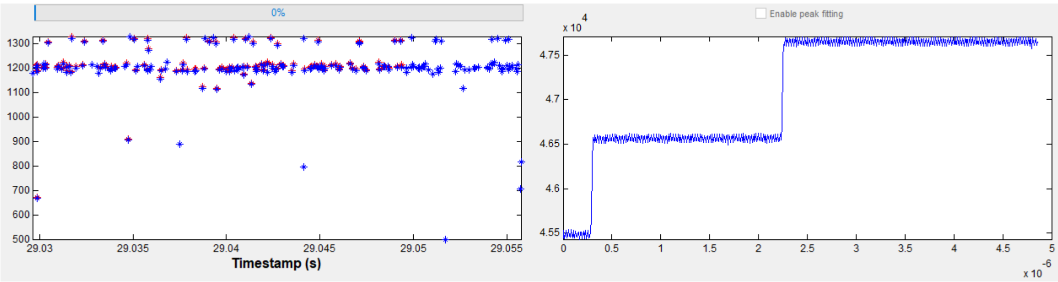

The left plot shows the recorded energies with a 16 bit precision like the list mode read-out. The right plot shows the waveforms associated to detected events. The "Waveform panel" of the GUI allows to customize the waveform acquisition using the "*waveform length*" and "*pre trigger*" sliders.



| Parameter | Unit | Range | Range[us] |
|---|---|---|---|
| Waveform length | 16 ns sample | 8 to 1256 | 0.128 to 20us |
| Pre Trigger | 16 ns sample | 0 to 600 | 0 to 9.6 |

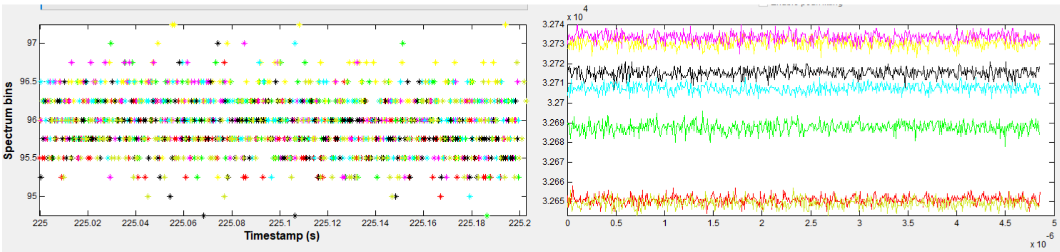
The *"Pre Trigger"* slider can be used for shifting the time-window around the edge of the triggering signal. A default value of 4.8us allows to see the edge of the recorded event.

It can happen that inside the time-window of the processed event, another event is detected as shown in the following picture



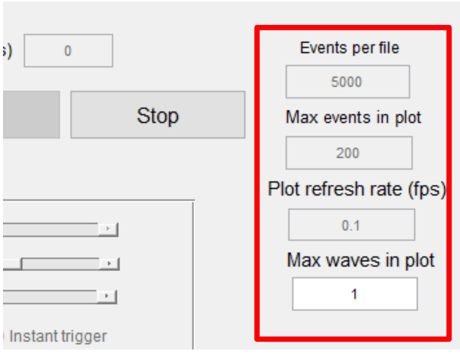

In this case the second event is discarded and won't be processed by the DPP.

In case of daisy-chain connection, events and waveforms of different channels are superimposed in the same two plots (see next figure)



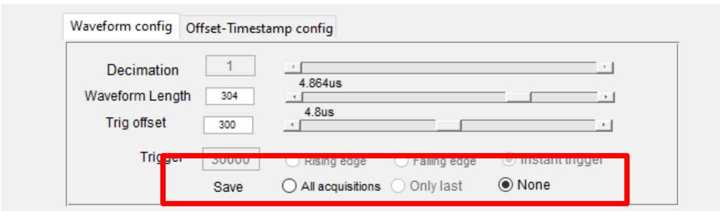

## Plot option and Saving

Waveforms and energies plot can be configured using the following dedicated panel:




- *Event per file*: Number of events and waveforms contained in a single *.mat* file.
- *Max events in plot*: Number of events (per channel) contained in the energy/timestamp plot.
- *Plot refresh rate (fps)*: Refresh rate for both the energies/timestamps and waveform plot. Using a higher value can significantly slow-down the GUI.
- *Max waves in plot*: Number of waveforms (per channel) contained in the energies/timestamps plot. The waveform will be plotted starting from the first detected event of the left plot.

Furthermore, the GUI allows to enable/disable the saving of both energies and waveforms data from the *Waveform config* panel.




Based on the *Events per file* value, the GUI generates *.mat* file containing the following data:

- *Energies*: column vector containing the energies (expressed in bin) of the detected events
- *Timestamps*: column vector containing the timestamps value (expressed in number of 8ns clock cycle) for each detected event
- *Waves*: matrix containing all the recorded waveform. Each column contains a single waveform.
- *Statistics*: statistics information of the acquisition.

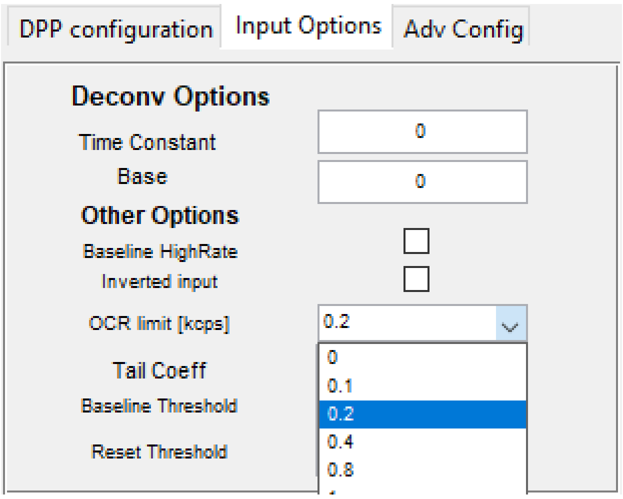
## Throughput

List-wave mode is very demanding in terms of bandwidth and can sustain relative low count rate. Given a maximum throughput of 80Mbit/s (TCP/IP connection), consider the following formula for a rough estimation of the maximum input count rate.

$$OCR_{max} \approx \frac{80Mbit/s}{N_{ch} \cdot 64bit \cdot wave\ length}$$

In case the limit is overcame, the hardware memory may become full during the acquisition.

Based on the throughput limitation, the possibility to internally limit the rate of processed events (independently to the ICR) has been implemented. As shown in the next picture, from the "*Input options*" panel the user can set the maximum rate of events to be processed by the DPP. Further events will be discarded. In this way, the user can stand with long waveform acquisition and high count rate sources.

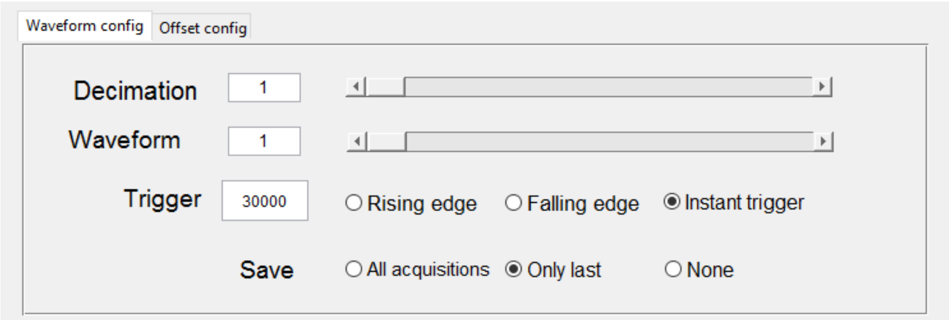



Accepted values are:

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 (disabled) | 100 Hz | 200 Hz | 400 Hz | 800 Hz | 1 KHz | 2 KHz | 4 KHz | 8 KHz | 10 KHz | 20 KHz | 40 KHz | 80 KHz | 100 KHz | 200 KHz |

# Waveform mode

In this acquisition mode, the user can acquire the input signal waveform. In the Waveform Configuration box there are some parameters the user can adjust before or during the acquisition.




The Decimation parameter indicates the decimation of the acquired samples, from 1 to 32. If set to 1 the waveform is sampled at 62.5 MHz, if set to N one sample every N samples is acquired.

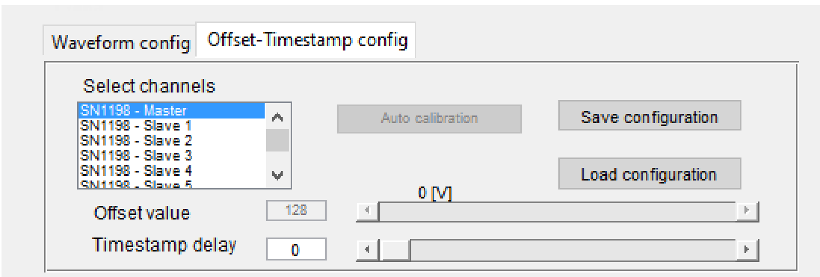
The Waveform Length specify the desired length of the waveform and is expressed in 16384 samples unit. For example if length is set to 1, 16384 samples are acquired; if set to 2, 32768 sample are acquired; etc.

Then the user can specify a trigger for the acquisition that can be an Instant trigger or on rising/falling edge of the level in ADC units (from 0 to $2^{16}$-1) inserted in the adjacent text box. A combination of the various triggers can be selected and the acquisition will be triggered by the first event that happens. In this mode the Time specification is the time the system waits for a triggering event.

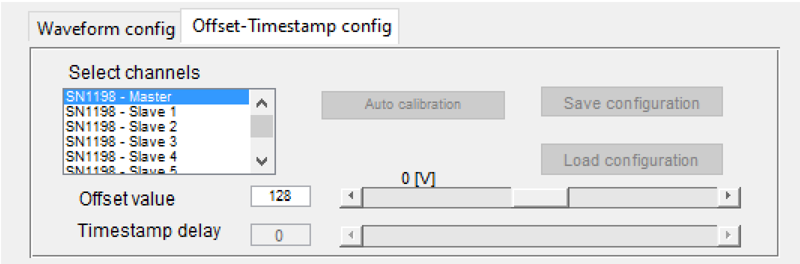
The acquisition is started pressing the Start acquisition button and is stopped pressing it again. During the acquisition the waveform is acquired continuously and the parameters can be changed in the meantime.

## Offset configuration

The GUI allows to offset the analog input voltage of the connected DPP boards. By clicking on the offset-timestamp config tab the user can change the offset parameter of the boards.




In order to manually change the offset, the waveform acquisition must be started. After the start of the waveform, the offset slider will be enabled. The user can change both a single board or multiple board from the menu and change their offset through the slider.




**Warning**
Due to a low pass filter, the offset will reach its target value with a time constant of the order of second.

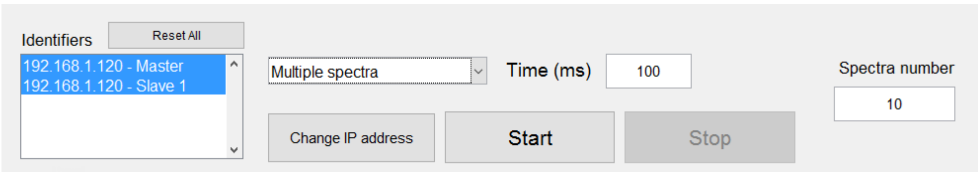
Looking at the waveform plot the user can see the offset effect on the input signal.

In case the user wants to store a particular configuration, it can be done by the "save configuration" button. In the same way, the user can load any previously saved configuration.

In addition to the manual load and save feature, the GUI automatically save the current configuration for the boards in order to load it at the start-up. After the connection of a new board, the GUI will search for an offset configuration related to the serial number of the new connected device. And automatically load it.

# Multiple spectra

In this mode the user can launch fast acquisitions one after the other with no deadtime in between.



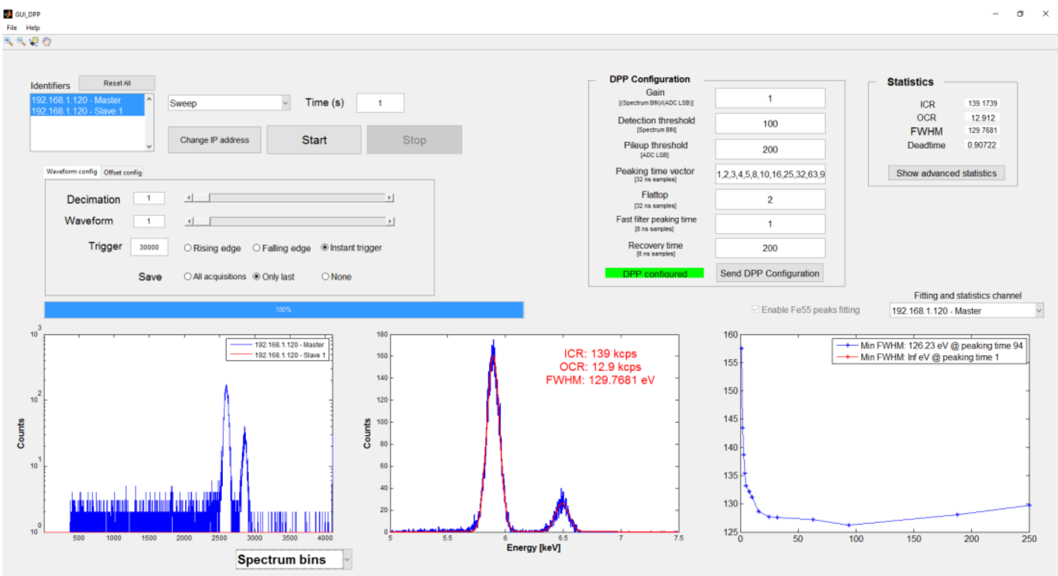

The configuration of this acquisition mode is simple: set the time per spectrum in milliseconds, the number of spectra and start the acquisition. On the GUI will be displayed the latest spectrum acquired once a second while all the data will be saved in one or more mat file in the *measures* folder. If the spectra number is set to zero the acquisition will be free running, meaning that it runs indefinitely until the user stops it.

The saved .mat file will have a name similar to this:
*map_modeX_SN00YYY_ZZZZ.mat*
Where X represent the number of the measure, YYY the serial number of the board, ZZZZ the number of the packet within the same measure. Every file generated will contain a number roughly equal to 1000 spectra with the corresponding statistics.
The saved file will contain four variables: *spectra* which contains a matrix with all the spectra acquired; *stats* which contains the base statistics (in this order: realtime in microseconds, livetime in microseconds, ICR in kcps, OCR in kcps); *adv_stats* which contains the advanced acquisition statistics for each measure; *id* that is a vector with the identifiers of each spectrum (always incrementing).
The data gets split in more file to be easily managed afterwards for elaboration.
Column 15, 16, 17 and 18 of *adv_stats* contains all the gating sampled values (if enabled by the GUI). Namely, columns 15 and 16 indicate that a rising or falling edge occurred on the trigger signal, respectively, during the spectrum acquisition, whereas columns 17 and 18 indicate that the gating signal was low or high during the spectrum acquisition, respectively.

# Sweep

In this mode the application can launch several acquisition one after another varying a DANTE DPP parameter. The parameters that can be varied are: Peaking time, Flattop, Pileup threshold, Detection threshold, Fast filter peaking time. The vector of the values must be specified in this way: 'value1,value2,value3,value4'. There are no limits in length. When selecting this acquisition mode, as an example, a Peaking time vector is inserted in the corresponding box.

A recommended practice before looking at the DPP performances at various peaking times, is to sweep the flattop parameter using this mode looking for the best value in term of resolution (which will result just longer than the maximum rise-time of the signal).



During the sweep acquisition, the user can choose which channel to show both the fitting plot and the statistics information.

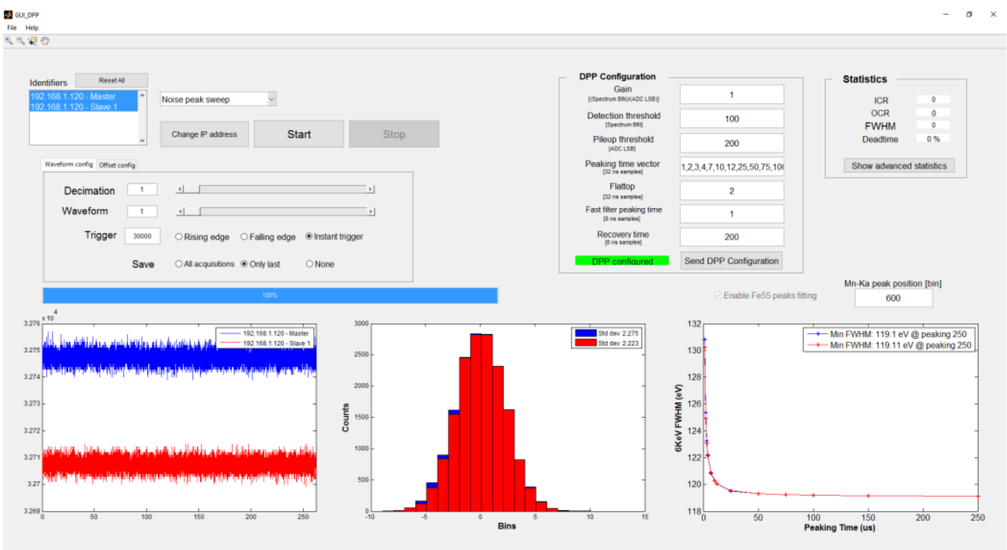
Also in the sweep mode, the user can calibrate the x-axis showing the energy of the events instead of using the bins.

All the spectra and the statistics are automatically saved inside the 'measures' folder. The acquisition can be stopped at any time pressing the stop button.

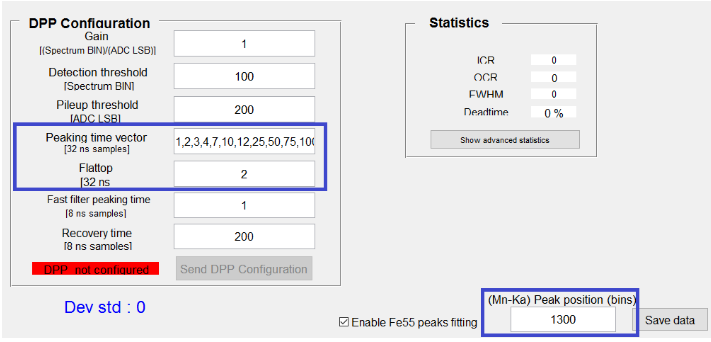
# Noise peak sweep

In this mode the application shows the noise contribution of the system for the Mn-Kα peak for a sweep of peaking times. This is a theoretical analysis of the noise contribution of the setup. No source is required. The results are related to the measuring setup (i.e. in the figure it is represented the noise contribution due to DANTE DPP, no sensor is attached). The application acquires a waveform of the setup in a continuous mode and calculates the noise distribution histogram (LSB). The standard deviation of the histogram is given for each waveform. The noise contribution is represented in the third graphic, giving the minimum FWHM value and the corresponding peaking time.

The data can be saved pressing the Save data button as .mat file and as image.

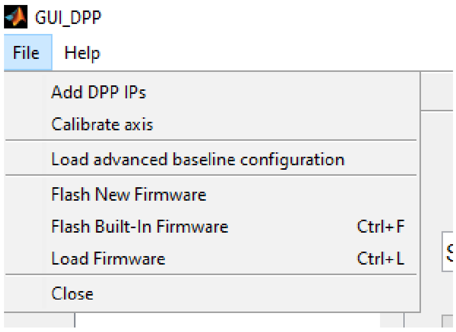



This mode allows only the configuration of three parameters, the peaking time vector, the flattop value and the Peak position. The peak position parameter is needed for the calibration of the noise. It is the bin of the spectrum corresponding to the 5.9keV peak position (centroid of Mn-kα lines).




# Load new firmware

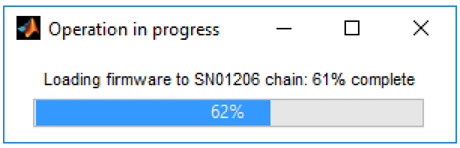
If a newer version of the firmware is released by XGLab you can upload it using the GUI with USB connection.



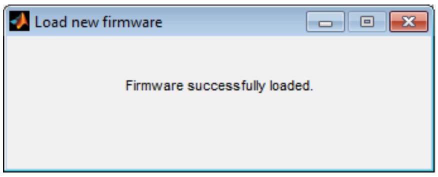

In *File* toolbar the user can select the following actions:

- *Flash New Firmware*: Used for storing a new firmware into the DANTE DPP.
- *Flash Built-In Firmware*: Used for automatically store the default firmware provided by the API package.
- *Load Firmware*: Temporary load a new firmware into the DANTE FPGA. After the power cycle of the device the firmware upgrade will be lost.

By choosing the Flash New Firmware option,  the user is asked to select the board to upgrade. You can at this point select the firmware .bitc file to upload. This operation lasts up to 45 seconds for the single channel unit and this message will be displayed in the meantime:

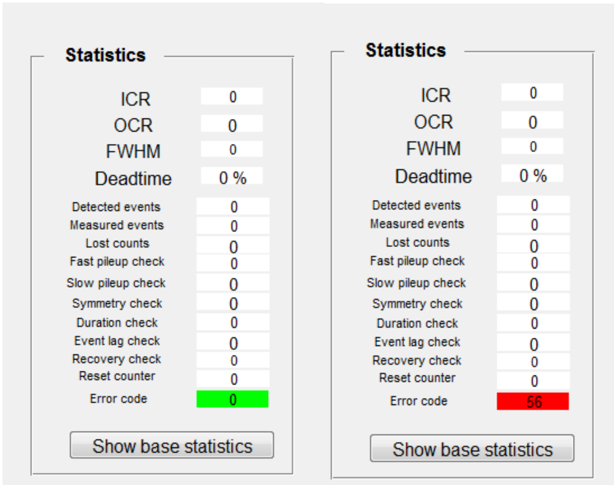



Please do not send configurations or acquire with the DPP during this time. If the firmware loading succeeded it will be displayed this message:




## Debugging functionalities

The DANTE GUI will help the user in understanding more about errors if they occurs. The current DLL error code is constantly displayed under the 'Advanced statistics' tab: a number different from 0 means that some error occurred. You can than check the DLL manual to get more about it or signal it to us.




In special cases, it is also possible to make the DLL produce logging files that will help solving eventual problems. To enable this feature just place a file named 'enable_log.xgl' (.xgl is the extension) with a number from 0 to 5 inside. 0 will produce a very detailed report while 5 will record only errors (2 is usually enough). We don't recommend using logging levels 0 and 1 for long sessions or when the DPP throughput is pushed to the limit, as this will slow down a bit the software. Please share us this log files when needed.

# Throughput optimization

The maximum throughput of the DPP when using the TCP-IP protocol is 70 Mbps. In multiple spectra mode this allows to successfully transmit 1 spectrum per ms without being limited by the connection. For boards connected in daisy chain of course the limit is higher, for example in a two boards daisy chain the limit is reached when acquiring one spectra every 2 ms, for four boards one spectra every 4 ms.

Same considerations can be done when using the USB connection, although the maximum tested speed is 140 Mbps. This means for example that 1 ms spectra can be done in a two DPPs chain without reaching the limit.

However, with some slower PCs or network adapter; it could happen that the maximum speed is 60 Mbps, so when doing 1 ms per spectra with one board the throughput is limited by the connection.

In order to reach this throughput is highly recommended to disable the WiFi adapter, if present, in order not to overload the network adapter.

**Warning**
From firmware version 1.5.20 it is not needed anymore to modify Windows registry entries about the Tcp protocol. Instead we recommend to delete the registry entries TcpAckFrequency and TcpDelAckTicks, if created for previously firmware versions.

Anyway DPP will not block if the desired transmission speed exceed the maximum achievable throughput: it will just start to drop some spectrum if not capable of retrieving them all in time. If this occurs, it will be signaled by the GUI displaying the MISSED_SPECTRA error code (which can be seen in the DLL header or in the DLL manual).

However each DPP mounts a RAM module that can contain up to 7500 spectra, so one can successfully acquire a spectra burst lower than 7500 whichever it is the spectrum time.

# Manual revisions

**Revision 2.9**:
Revisioned triggering and gating features.

**Revision 2.8**:
Added weight.

**Revision 2.7**:
Firmware loading section updated.

**Revision 2.6**:
Parameters range for high-rate mode.

**Revision 2.5**:
Added high-rate mode

**Revision 2.4**:
Inserted absolute input signal ranges depending on the offset applied.

**Revision 2.3**:
Detailed more the pileup threshold and baseline samples settings in Usage guidelines and User Interface sections.

**Revision 2.2**:
Added OCR limitation for list wave mode.

**Revision 2.1**:
Added List Wave mode

**Revision 2.0**:
Added support for new 3.0.x firmware versions with new configuration parameters.

**Revision 1.9**:
Added description of DPP configuration panel. Added location of gating signal in adv_stats.

**Revision 1.8**:
Added gating operation.

**Revision 1.7**:
Added zero peak rate and programmable edge detection filter flat top.

**Revision 1.6**:
Added power input connector pinout.

**Revision 1.5**:
Added 8ch DANTE box description.

**Revision 1.4**:
Added time-stamp delay configuration.

**Revision 1.3**:
Added box dimensions.

**Revision 1.2**:
Added VC++ Runtime as a requirement installation for the GUI and DLL.

**Revision 1.1**:
Modified the Throughput Optimizations section to reflect the latest changes for firmwares from 1.5.20.

**Revision 1.0**:
Initial release.