Rev. 5 - Sept 6th, 2024

Digital Detector Emulator

Emulation of Digital Pulses from Radiation Detectors

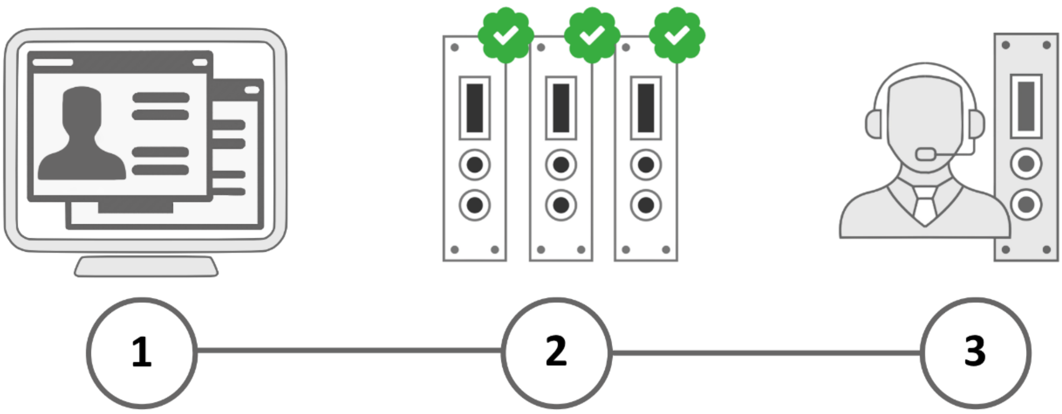
CAEN
Tools for Discovery

User Manual
UM5312

Register your device

Register your device to your **MyCAEN+** account and get access to our customer services, such as notification for new firmware or software upgrade, tracking service procedures or open a ticket for assistance. **MyCAEN+** accounts have a dedicated support service for their registered products. A set of basic information can be shared with the operator, speeding up the troubleshooting process and improving the efficiency of the support interactions.

MyCAEN+ dashboard is designed to offer you a direct access to all our after sales services. Registration is totally free, to create an account go to https://www.caen.it/become-mycaenplus-user and fill the registration form with your data.



create a MyCAEN+ account | **register your devices** | **get support and more!**

https://www.caen.it/become-mycaenplus-user/

Purpose of this Manual

This User Manual contains the full description of the Digital Detector Emulator firmware and software GUI. The description is compliant with Digital Detector Emulator firmware release **16.1_3.0.0.0,** and software release **2020.7.0.1.** For future release compatibility please check the firmware and software revision history files

Change Document Record

Date	Revision	Changes
December 21st, 2016	00	Initial release
February 26th, 2018	01	Added support to the DT5810B. DT5810D discontinued.
April 20th, 2020	02	Modified **Noise Emulation** section. SDK included in the installation folder.
November 22nd, 2021	03	Modified **Clock Manager** section
July 29th, 2024	04	Modified **Accepted File Formats** section. Added **Register your device, Packaging and compliancy, PID (Product Identifier), Cooling Management, Instructions for Cleaning, Device decommissioning, Disposal** Sections. Modified **First Page**, **Last Page**, **Technical Support** Chapter.
September 6th, 2024	05	Modified **Safety Notices**.

Symbols, abbreviated terms and notation

TDC	Time-to-Digital Converter
ADC	Analog-to-Digital Converter
DAQ	Data Acquisition
DPP	Digital Pulse Processing
MCA	Multi-Channel Analyzer
DDE	Digital Detector Emulator
QDC	Charge-to-Digital Converter
USB	Universal Serial Bus
DDE	Digital Detector Emulator

Reference Documents

[RD1] FPGA-Optimised Uniform Random Number Generators Using LUTs and Shift Registers, David B. Thomas and Wayne Luk]

[RD2] UM3182 – DPP-PHA and MC2 Analyzer User Manual

Manufacturer Contacts

CAEN S.p.A.
Via Vetraia, 11 55049 Viareggio (LU) - ITALY
Tel. +39.0584.388.398 Fax +39.0584.388.959
www.caen.it | info@caen.it

Limitation of Responsibility

If the warnings contained in this manual are not followed, CAEN will not be responsible for damage caused by improper use of the device. The manufacturer declines all responsibility for damage resulting from failure to comply with the instructions for use of the product. The equipment must be used as described in the user manual, with particular regard to the intended use, using only accessories as specified by the manufacturer. No modification or repair can be performed.

Disclaimer

The information contained herein has been carefully checked and is believed to be accurate; however, no responsibility is assumed for inaccuracies. CAEN spa reserves the right to modify its products specifications without giving any notice; for up to date information please visit www.caen.it.

Made in Italy

We remark that all our boards have been designed and assembled in Italy. In a challenging environment where a competitive edge is often obtained at the cost of lower wages and declining working conditions, we proudly acknowledge that all those who participated in the production and distribution process of our devices were reasonably paid and worked in a safe environment (this is true for the boards marked "MADE IN ITALY", while we cannot guarantee for third-party manufactures).

Index

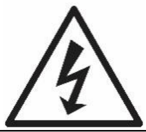
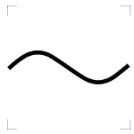
List of Figures

List of Tables

Safety Notices

N.B. Read carefully the "SAFETY, STORAGE AND SETUP INFORMATION, PRODUCT SUPPORT SERVICE AND REPAIR" document provided with the product before starting any operation.

The following HAZARD SYMBOLS may be reported on the unit:

	Caution, refer to product manual
	Caution, risk of electrical shock
	Protective conductor terminal
	Earth (Ground) Terminal
	Alternating Current
	Three-Phase Alternating Current

The following symbol may be reported in the present manual:

	General warning statement

The symbol could be followed by the following terms:

- **DANGER**: indicates a hazardous situation which, if not avoided, will result in serious injury or death.
- **WARNING**: indicates a hazardous situation which, if not avoided, could result in death or serious injury.
- **CAUTION**: indicates a situation or condition that, if not avoided, could cause physical injury or damage the product and / or its environment.

CAUTION: To avoid potential hazards

USE THE PRODUCT ONLY AS SPECIFIED.
ONLY QUALIFIED PERSONNEL SHOULD PERFORM SERVICE PROCEDURES

CAUTION: Avoid Electric Overload

TO AVOID ELECTRIC SHOCK OR FIRE HAZARD, DO NOT POWER A LOAD OUTSIDE OF ITS SPECIFIED RANGE

CAUTION: Avoid Electric Shock

TO AVOID INJURY OR LOSS OF LIFE, DO NOT CONNECT OR DISCONNECT CABLES WHILE THEY ARE CONNECTED TO A VOLTAGE SOURCE

CAUTION: Do Not Operate without Covers

TO AVOID ELECTRIC SHOCK OR FIRE HAZARD, DO NOT OPERATE THIS PRODUCT WITH COVERS OR PANELS REMOVED

CAUTION: Do Not Operate in Wet/Damp Conditions

TO AVOID ELECTRIC SHOCK, DO NOT OPERATE THIS PRODUCT IN WET OR DAMP CONDITIONS

CAUTION: Do Not Operate in an Explosive Atmosphere

TO AVOID INJURY OR FIRE HAZARD, DO NOT OPERATE THIS PRODUCT IN AN EXPLOSIVE ATMOSPHERE

The following operating limits must be respected:

Net class	Connector	Unit	Min	Max
Power	/	Voltage	90 V_{ac}	250 V_{ac}
OUT1/2	LEMO 00	Voltage	-2 V (50 Ohm)	2 V (50 Ohm)
		Voltage	-4 V (High-Z)	4 V (High-Z)
OUT1 HDR/OUT2 HDR	LEMO 00	Voltage	-8 V (High-Z)	8 V (High-Z)
GPO 1/2	LEMO 00	Voltage	LVCMOS	LVCMOS
GPI 1/2	LEMO 00	Voltage	LVCMOS	LVCMOS
Analog SE Input	LEMO 00	Voltage		1/2/4/8 V_{pp}

Table 0.1: operating limits for DT5810B connectors.

WARNING: the two digital front connector lines are directly connected to the FPGA I/Os. Violation in maximum absolute rating illustrated in this document will likely destroy the FPGA. There is no buffer or protection on this line. That is necessary because we want to preserve the possibility to operate at different voltages and with both single ended and differential signals: front Digital I/O can operate 2.5V and can be configured as LVDS.

THIS DEVICE SHOULD BE INSTALLED AND USED BY SKILLED TECHNICIAN ONLY OR UNDER HIS SUPERVISION

DO NOT OPERATE WITH SUSPECTED FAILURES.
IF YOU SUSPECT THIS PRODUCT TO BE DAMAGED, PLEASE CONTACT THE TECHNICAL SUPPORT

See **Chapt. 16** for the Technical Support contacts.

1 Introduction

The massive evolution of digital processors for radiation measurements has highlighted the extreme convenience to develop techniques for emulating detection and acquisition systems. The process of debugging systems like digital pulse processors, pulse discriminators, Time-to-Digital/Amplitude converters, etc., requires an ever-increasing effort of processing algorithms that are becoming more and more complex. The possibility to generate test vectors that are as similar as possible to the actual data, both in the software simulation and at the hardware level, can extremely reduce the R&D development time.

All of this can be summarized in the need to generate an electrical signal with completely controlled characteristics that is compliant to the real output of a radiation detection setup. Although the use of a source and a detector is always the best way to generate a reliable data set, it involves considerable disadvantages especially during preliminary feasibility studies. The use of the source inherently involves a risk for experimenters' health, and in addition it requires labs equipped in accordance with the regulations in term of use of radioactive substances.
Moreover, the emission spectrum depends on the nature of the source, e.g. the polarization of an X-ray tube or the process of decay. The statistical distribution of the events is Poissonian and usually the user can only control the rate while having no control on the statistics. The spectrum of noise, interferences and the pulse shape are issues on which the experimenter can hardly affect. Furthermore, the natural emission process is not repeatable and therefore it is not possible to evaluate the behaviour of different implementations of the processing system on a set of equal data.

It is common practice to use electronic instruments to generate analog signals with similar features to real experiment. There are instruments able to generate exponential signals with fixed amplitude and Poissonian temporal distribution, that can emulate effects such as stockpiling. However, they cannot modulate the amplitudes of generated signals according to a generic spectrum of emission.
To overcome this problem, hardware tools -called arbitrary waveform generators- have been developed. These instruments can generate long sequences of events at high rate (up to 1 GSPS), which have been synthetized off-line using simulation tools (e.g. MatLab). At present the most efficient tools have 1 Gword memory which means about a signal one second long before a repetition of about 10 millions of counts. Obviously this is not enough for good statistics of a complex spectrum.

Therefore, in collaboration with Nuclear Instruments srl, we developed a multichannel digital instrument for emulating radiation detection systems. The processor is initialized with a reference pulse shape, with statistic distribution of amplitude and time. According to this information, the device generates a stream of events that can be also selectively summed together simulating the pile-up phenomenon. At each pulse can be superimposed a baseline deviation and an arbitrarily-generated noise.

The Digital Detector Emulator is therefore the only synthesizer of random pulses that is also an emulator of radiation detector signals with the possibility to configure energy and time distribution. The stream of emulated signals becomes a statistical sequence of pulses, reflecting the programmed input features. When the emulation process is reset, the kernels of generators can be either re-initialized with new random data making the sequence always different, or they can be stored to reproduce the same sequence many times.
The Digital Detector Emulator is able to emulate two different radiation sources at a time on the two output channels and to provide them either with fully independent parameters, or with some of them correlated. For example, the events can be time-correlated (steps of 1 ns), or a subset of events can share the same energy spectrum. It is also possible to set the channels in a master/slave configuration, where the first channel works as a trigger for the second one.

The DT5810 is equipped with an analog input channel through which it is possible to add to a signal coming from a real measurement setup the signal generated by the emulator. In this way it is possible to emulate a source not actually present or hardly available.
Moreover, the analog input allows to characterize the detector by acquiring shape and spectrum of its output signal.

Main functionalities

The main features of the Digital Detector System are:

- Emulator/Pulser/Function Generator operation mode
- Energy spectrum emulation
- Time spectrum emulation
- Pile-up emulation
- Noise and periodic interference emulation
- Continuous and pulsed reset emulation
- Baseline drift
- Debug mode: predictable sequence generation with step-by-step pulse generation
- Windows software for full system management
- USB connection
- DLL for automation of emulation process
- Replay on analog channels of recorded or synthetized signals
- Generation of shifted copy of a signal with 1 ns step (i.e. correlated event emulation)
- Load / download in CSV format of parameters/shapes/spectra
- Analog Input

The description of this User Manual is compliant with the following products:

Board Models	Description	Product Code
DT5810D	DT5810D - Dual Channel Fast Desktop Digital Detector Emulator with channel correlation - DISCONTINUED	WDT5810DXAAA
DT5810B	DT5810B - Dual Channel Fast Desktop Digital Detector Emulator with channel correlation	WDT5810BXAAA

Tab. 1.1: Table of the supported Digital Detector Emulator models

2 Technical Specifications

Energy emulation features	• Single line (65535 selectable levels) • Spectrum emulation (16384 bins with 14-bit resolution) • ± 4 V output range, high impedance; ± 2 V, 50 Ω termination (high speed mode) – FAST OUTPUT • ± 8 V output range, high impedance; (high dynamic range) – HDR OUTPUT • 16-bit D/A converter
Time emulation features	• Constant rate emulation • Poisson distribution • Up to 30 MCPS, both in constant and statistical emulation • Integrator circuit emulation without pile-up limitation • Up to 16 pile-up events in the memory based algorithm • Programmable dead-time and emulation of parallelizable and non-parallelizable machines • 5 ns to 10 ms exponential decay time
Signal shape	• 4096 points to store waveforms • Arbitrarily programmable shapes • Shape length from 3 ns to 4 µs (w/o interpolation) / 4 ms (interp.) • Separated rising and falling edge interpolation • Up to four separate shapes mixed on the same channel with independent statistic
Noise emulation	• White noise emulation (BW 500 MHz) • Random noise • 1/f noise emulation • Random Walk (baseline drift) • Shot noise
Baseline	• Baseline drift programmable with arbitrary shape
Correlated events emulation	• Three operation modes: 1) Channel 1 (CH1) is the time shifted copy of Channel 2 (CH2) (1 ns step); 2) CH2 has its own statistics generator (i.e. different spectrum, different noise, etc.) but is triggered by CH1 (delayed by 1ns step); 3) A third emulator channel (with separate statistic properties) generates correlated pulses for both CH1 and CH2. In this way, only some events of the two channels are correlated • 1 ns step programmable delay (from 0ns to 32us)
Digital I/O	• 2-input and 2-output programmable • Trigger out, analog saturation warning, machine overload sensing • Trigger in, random number generator control (reset / play / pause), gating, baseline reset
RNG (random number generation)	• 8 independent LFSRs with 64 bits generate the base for the statistical emulation • Possibility to randomize the seeds of each Random Generator (RNG) independently • Possibility to initialize the RNG with fixed seeds to get repeatable sequences to test different processing architectures • Generation of finite length streams of pulses to debug the Device Under Test (DUT) step-by-step
Programmable sequence	• 4 kpoints of memory/CH to store a sequence of pairs (energy, time of occurrence) to generate long predictable and defined sequences of pulses
Waveform generator	• Function generation: sin, square, ramp, saw, pulse, sinc up to 10 MHz
Software and interface	• Windows-based user interface managing more than one emulator • USB interface
Analog Input	• 1 ADC input, 156 MSPS 14 bit • DC coupled input (dynamic 1/2/4/8 V) • AC coupled (gain from 2 to 110) • Real-time emulation signal mixing with analog source • Full MCA functionality to calculate the spectrum of the source • Fast shape averaging to extract a model of the input signal

3 Packaging and compliancy

The DT5810B is available as desktop module housed in an aluminium case - 165x50x227 mm^3 (WxHxD).

The unit is inspected by CAEN before the shipment, and it is guaranteed to leave the factory free of mechanical or electrical defects.

The content of the delivered package standardly consists of the part list shown in the table below (**Table 3.1**). All the official documentation, firmware updates, software tools, and accessories are available on *www.caen.it* at the product web page.

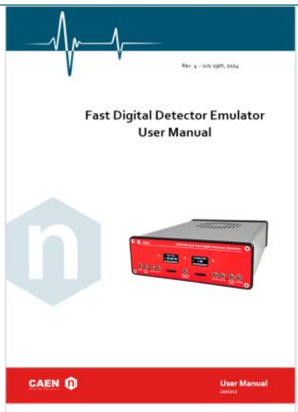
	Part	Description	Qty
	DT5810B	Dual Channel Desktop FAST Digital Detector Emulator with channel correlation	x1
	Power supply cable	Standard C13 power supply chord	x1
	AC/DC converter	12V-65W VEC65US12 AD/DC Converted	x1
	USB cable	USB 3.0 type A-MICRO-B I/O cable L=1.5MT	x1
	Adapter	LEMO-BNC ABF.00.250.CTA	x2
	User manual	UM5312 – Fast Digital Detector Emulator User Manual	x1

Table 3.1: delivered kit.

CAUTION: to manage the product, consult the operating instructions provided.

When receiving the unit, the user is strictly recommended to:

- Inspect containers for damage during shipment. Report any damage to the freight carrier for possible insurance claims.
- Check that all the components received match those listed on the enclosed packing list as in Tab. 4.1 .(CAEN cannot accept responsibility for missing items unless any discrepancy is promptly notified.)

- Open shipping containers; be careful not to damage contents.
- Inspect contents and report any damage. The inspection should confirm that there is no exterior damage to the unit such as broken knobs or connectors and that the front panel and display face are not scratched or cracked. Keep all packing material until the inspection has been completed.
- If damage is detected, file a claim with carrier immediately and notify CAEN service (see Chap. 17)..
- If equipment must be returned, carefully repack equipment in the original shipping container with original packing materials, if possible. Please contact CAEN service.
- If equipment is not installed when unpacked, place equipment in original shipping container and store in a safe place until ready to install

DO NOT SUBJECT THE ITEM TO UNDUE SHOCK OF VIBRATIONS

DO NOT BUMP, DROP OR SLIDE SHIPPING CONTAINERS

DO NOT LEAVE ITEMS OR SHIPPING CONTAINERS UNSUPERVISED IN AREAS WHERE UNTRAINED PERSONNEL MAY MISHANDLE THE ITEMS

USE ONLY ACCESSORIES WICH MEET THE MANUFACTURER SPECIFICATIONS

For a correct and safe use of the module, refer to **Chap. 5** and **6.**

4 PID (Product Identifier)

PID is the CAEN product identifier, an incremental number greater than 10000 that is unique for each product[1]. The PID is on a label affixed to the product (**Fig. 4.1**) and it is even stored in an on-board non-volatile memory readable. The PID information is also available through the Detector Emulator Control Cernter software.

Note: The serial number is still valid to identify older boards, where the PID label is not present.

Fig. 4.1:PID location taking a CAEN desktop unit as an example (the number in the picture and the device model are purely indicative)

[1] The PID substitutes the serial number previously identifying the boards.

5 Power Requirements

The module is powered by the external AC/DC stabilized power supply provided with the digitizer and included in the delivered kit. Please, use only the power supply shipped with this instrument and certified for the Country of use.

Input: 100-240 VAC, 50-60 Hz; Output: 12.0 V, 5.0 A.

Fig. 5.1: AC/DC power supply provided with the DT5810B kit

6 Cooling Management

The DT5810B board can operate in the temperature range -10 + 40 °C.

Air flow fans are installed onboard.

The user must take in care to provide a proper cooling to the board with external fan if the board is used in an enclosure or if the board is installed in a setup with poor air flow.
Excessive temperature will, in first instance, reduce the performance and the quality of the signal generation and can also damage the board.

Please do not close the read fan holes to avoid unit overheating. If in a single rack tower, multiple units are installed, please consider external fans or rack mounted air conditioning system.

If the board is stored in cold environmental, please check for water condensation before power on.

The board has not been tested for radiation hardness. High energy particle can be source of soft error and can damage the FPGA. If used in strong proton or neutron beams, arrange proper shielding or remote the sensor with a custom cable.

6.1 Cleaning the air vents

CAEN recommends to occasionally clean the air vents on all vented sides of the board. Lint, dust, and other foreign matter can block the vents and limit the airflow. Be sure to power off the board and disconnect it from the power by physically detach the power chord before cleaning the air vents and follow the general cleaning safety precautions.

IT IS UNDER THE RESPONSIBILITY OF THE CUSTOMER A NON-COMPLIANT USE OF THE PRODUCT

7 Installing the device

The device should be installed on a desktop by the following steps:

- Connect power supply chord
- Connect the USB cable

ONLY QUALIFIED PERSONNEL SHOULD PERFORM INSTALLATION, OPERATIONS

DO NOT INSTALL THE EQUIPMENT SO THAT IT IS DIFFICULT TO OPERATE THE DISCONNECTING DEVICE ON THE BACK PANEL

IT IS RECOMMENDED THAT THE SWITCH OR CIRCUIT-BREAKER IS NEAR THE EQUIPMENT

THE SAFETY OF ANY SYSTEM THAT INCORPORATES THE DEVICE IS UNDER THE RESPONSIBILITY OF THE ASSEMBLER OF THE SYSTEM

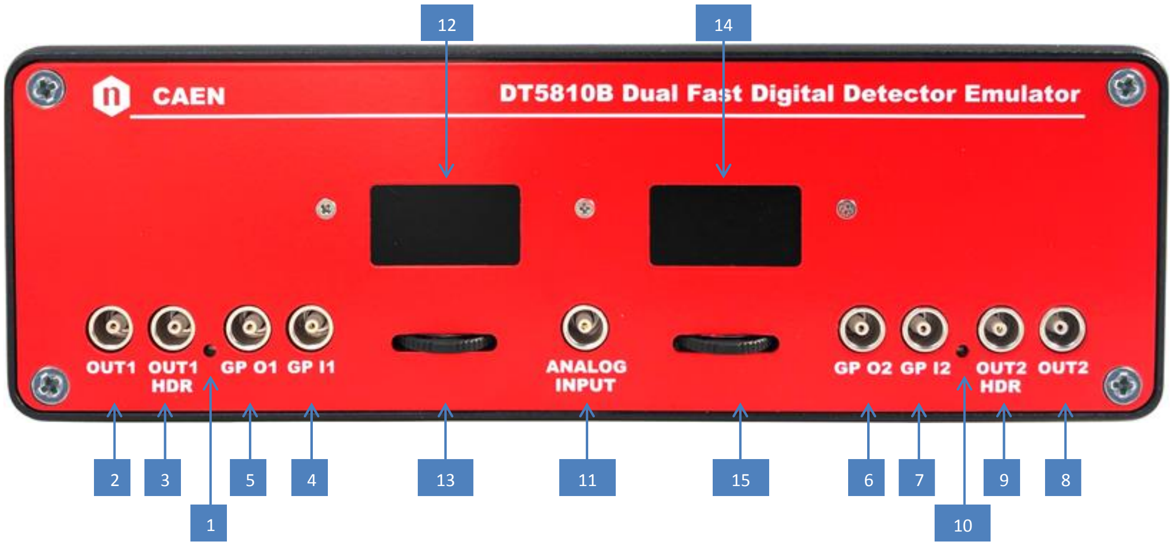
Do not use the device and contact technical support if one of these situations is verified:

- Enclosure integrity is compromised
- Insulation of HV chord is damaged (if present)
- The indication led or display is not performing as required (e.g. led not working, display with incorrect graphic)
- Fans are not working (if present)

8 Panel Description

Front panel device (DT5810B)

The DT5810B front panel is as shown in the following picture. Numbered labels are explained in the table.



Number	Description
1	Blue LED – Analog output CH1 power on status OFF: channel output disabled ON: channel output enabled
2	Fast Analog output CH1
3	High Dynamic Range (HDR) Analog output CH1
4	Digital input CH1
5	Digital output CH1
6	Digital output CH2
7	Digital input CH2
8	Fast Analog output CH2
9	High Dynamic Range (HDR) Analog output CH2
10	Blue LED – Analog output CH2 power on status OFF: channel output disabled ON: channel output enabled
11	Analog input
12	CH1 display
13	CH1 directional key controller
14	CH2 display
15	CH2 directional key controller

⚠ CAUTION. All I/O gates are LVCMOS compliant. The dynamic range of the analog outputs is 2.2 V @ 50 Ω output impedance and 4.4 V @ high impedance for the FAST Output and 12 V the HDR Output.

Display Information

The display allows the user to monitor the current status of each channel in the DT5810 device. The two channels have two separate displays so that it is always possible to independently monitor the information of interest.
Each display is organized in screens. To move between screens, the user can use the small jog dial located below each display, moving it to the right or to the left with your finger. Pressing the jog dial switches the channel on and off (toggle button). The user cannot change the status of the parameters. This can only be done from the PC.

When switched on, the device shows the following screen with the boot logo CAEN DT5810.

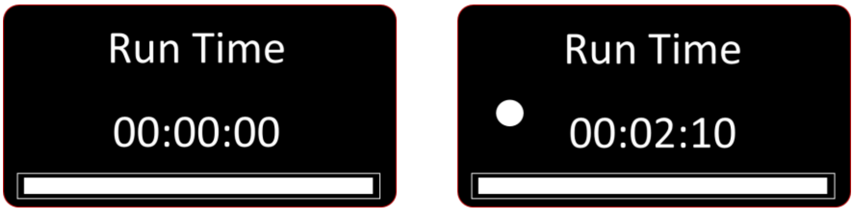
Bootstrap Screen

After booting, the instrument is not configured and needs to be programmed via the detector emulator software controller. The following screen will be shown.

Right now all the menus are locked. When the software is connected to the DT5810 and the device has been programmed, the displays are activated and the user can scroll through the menus.

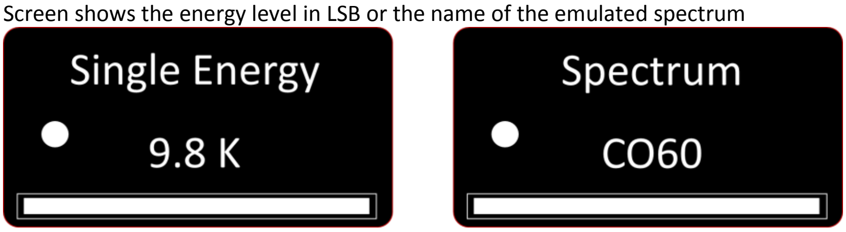
Runtime Screen

The homepage shows how long the emulation has been in progress. Every time the channel is turned on and off, the measuring time is resetted.



Energy Screen

Screen shows the energy level in LSB or the name of the emulated spectrum

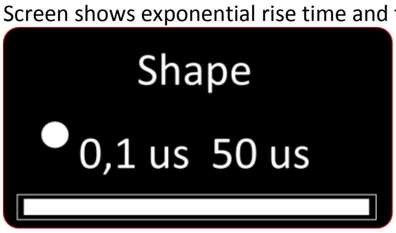


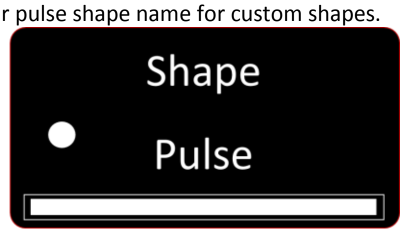
Emission Rate Screen

Screen shows the emission rate and the time base operation mode: constant rate/ Poisson distribution

Shape Screen

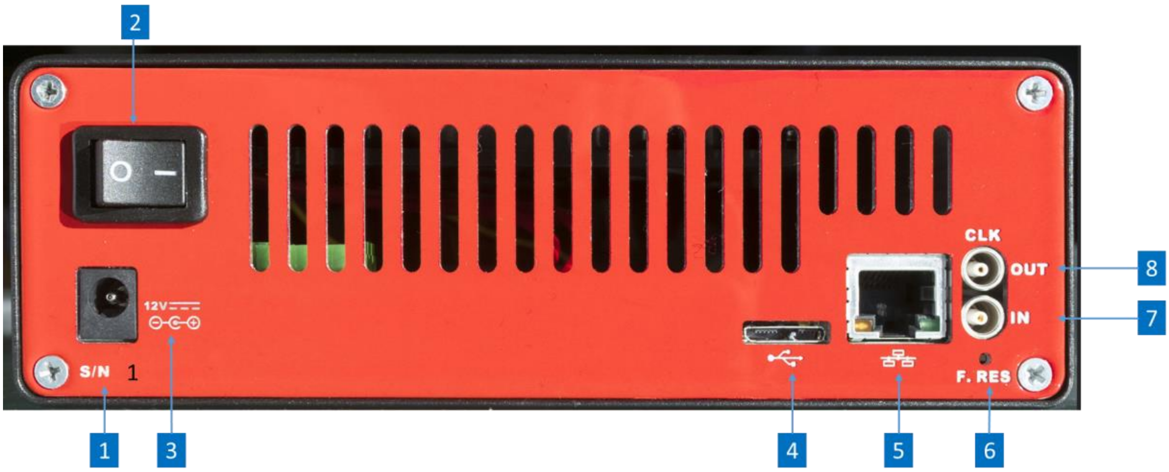
Screen shows exponential rise time and tau or pulse shape name for custom shapes.





Back panel device (DT5810B)

The DT5810B back panel is as shown in the following picture. Numbered labels are explained in the table.



Number	Description
1	Serial number
2	ON/OFF switcher
3	Power Supply Connector (+12 V central terminal positive)
4	USB Interface Connector
5	Ethernet Interface
6	Factory reset / Bootloader
7	25 MHz reference clock in
8	25 MHz reference clock out

9 Hardware architecture

Overview

The hardware structure of the emulator is shown in **Fig. 9.1**. The core of the system is a FPGA device Kintex 7 KC-160T which contains all the logic resources necessary for the emulation. The instrument does not have a hardware user GUI interface and requires a permanent connection via USB to the host PC, whose complex software allows to program all the operating settings of the instrument.

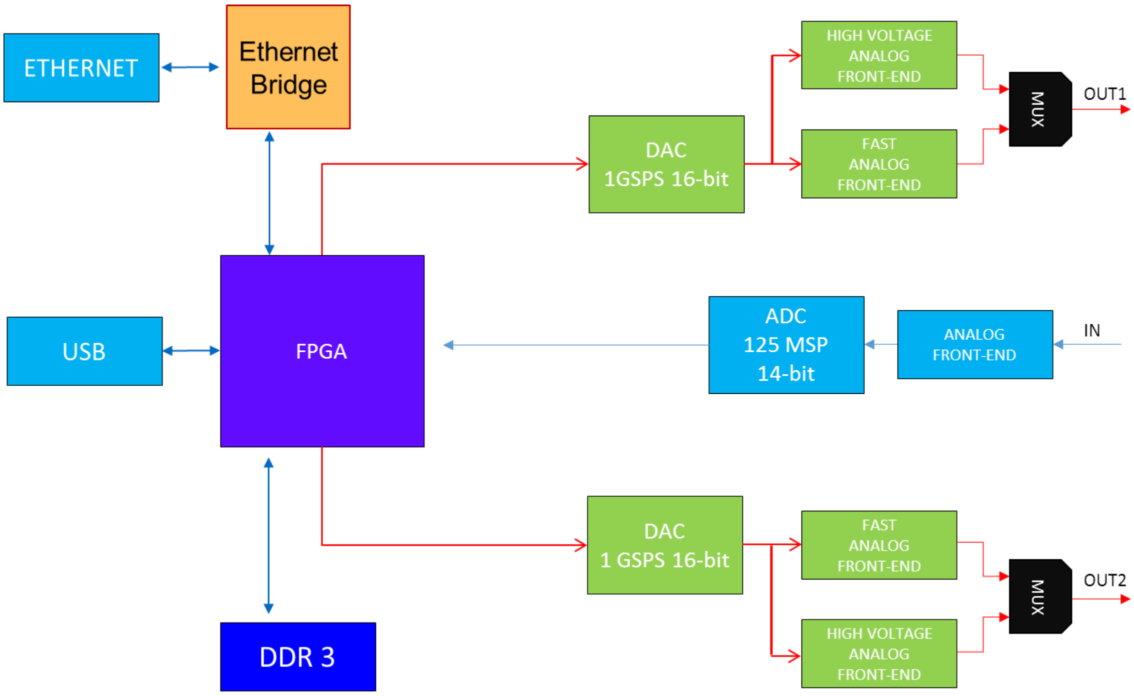


Fig. 9.1: The hardware architecture of the Digital Detector emulator

The USB connection is performed at the physical layer via the interface chip FX3 by Cypress. The FPGA is interconnected through two LVDS bus to a couple of 16-bit DACs, which generate the analog signals. Specific output stages convert the differential current signal from the DACs into single-ended voltage signals that are made available for the user. The FPGA is connected to two memories of 128 Mbyte for the storage of sequence. The power supply is a crucial point in the system. In fact, the digital circuits require high currents that are supplied by switching power architectures.
The analog section of the system requires pure power supplies with ripple below 5 mV. Moreover, the system is powered by a single +12 V source while the analog section needs both positive and negative voltage levels.

Analog Outputs

The DT5810 has two independent analog stages that can be used to simulate a large number of detectors. The low-voltage analog stage uses OPAMP CFA with over 2.5 GHz band in order to ensure fronts lower than 1 ns. The instrument is able to emulate both signals with a small voltage excursion and signals covering the whole dynamic of the instrument. It is important that the slew rate of the analog circuitry allows this excursion. Precisely, the output stage has to allow the op-amp output signal to increase from 0V to 4V in 1ns with a slew rate of 4000V/us. On the other hand, the high-voltage stage uses some more relaxed electronics and ensures rise time which are lower than 30 ns. As a result, the stage is able to provide a signal with a range of 24V (-12 to + 12V) on a HDR output at high impedance (1K), making it perfect for emulating signals from photomultipliers.
A filter for reducing the bandwidth to 30 MHz can be inserted in both output states to minimize the noise. The filter can be switched on and off via a software option.
The filter sequence in the analog output stage is shown in **Fig. 9.2**.

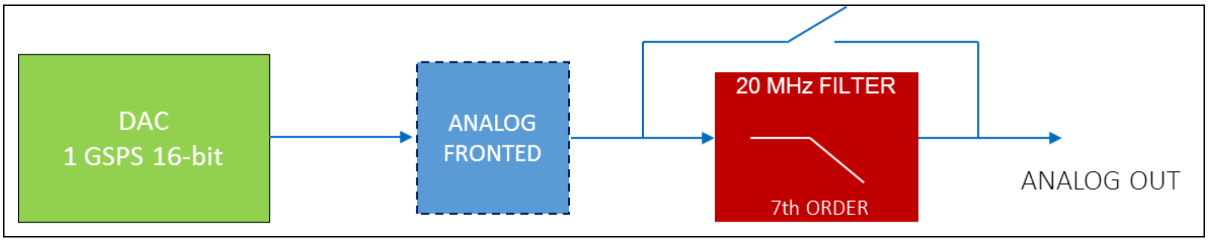


Fig. 9.2: Filter sequence on the Digital Detector analog output stage

Feature	Value		
Dynamic Range	± 2 V		
Linearity	10 ppm		
Rise Time	Configuration	Filter OFF	Filter ON
	High Speed	1 ns	25 ns
	High Voltage	25 ns	42 ns

Tab. 9.1: Analog output stage performance

The analog output is designed to generate a signal amplitude of ± 2 Vpp, with 50 Ω termination. It is possible to terminate it with high impedance, having a final amplitude of ± 4 Vpp. In this case, it is possible to have multiple signal reflections if the signal edge is sharp. It is strongly recommended to enable the filter when using high impedances. In addition, the ± 12 Vpp works with high impedance, and it has the same performance as the ± 4 Vpp option.

Digital I/O

The digital I/O functionalities can be set from the software GUI (refer to **Chapter Software Interface** for the complete list of functionalities).
The digital outputs provide LVCMOS 0-3.3 V signals, as the inputs can receive signals with amplitude 0-3.3V. Inputs and outputs are protected with anti-ESD diodes.

Correlation Block

It is possible to correlate the two analog outputs in three different ways.

Delay generation

The two channels of the DT5810 can be used in a master-slave configuration. The master channel has its own time base generator, as the slave channel generates a signal with a programmable time delay with respect to the master one. This resolution on the programmable delay is 1 ns. This is a quite peculiar feature, and even if higher temporal resolution pulser devices are available, none of them are able to generate at the same time analog signals with fully programmable shape, energy and temporal distributions. The high resolution of the settable delay makes the instrument particularly useful for designing and debugging measurements involving times of flight, as in high-energy physics experiments, or temporal correlation, as in PET applications, where the correlation defines a window of acceptance of the photons.

Three operating modes are available:

- In the *first mode* the slave channel generates the same event of the master channel, translated in time.
- In the *second mode* the two channels are totally independent and only temporally correlated. This means that different energy spectra, noise contributions, interference and shapes can be assigned to the two channels and the master channel only acts as a trigger for the slave one with a programmable temporal shift.

  The delay is obtained as shown in **Fig. 9.3** via a memory inside the FPGA. The digital signals which control the second DAC are delayed inside the device via a circular buffer whose length is programmed by the user. Each step of the time buffer is equivalent to a delay of 1 ns. The delay is programmable from 1ns to 4us.

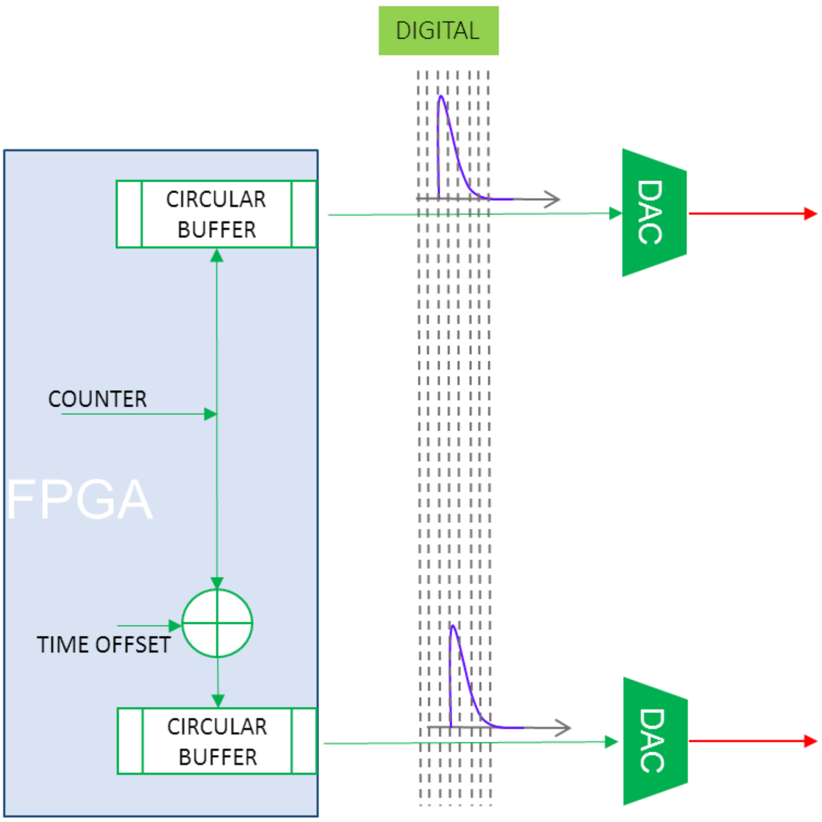


Fig. 9.3: Hardware and firmware generation of delay between the two output channels.

- In the *third operating mode* the emulator can be programmed to generate energy and time correlated signals. Besides the two physical channels there is a third channel inside the emulator that can be used to inject correlated pulses. The third channel has complete independent statistics, both in amplitude and time, and injects the same event in the two outputs (see **Fig. 9.4**).

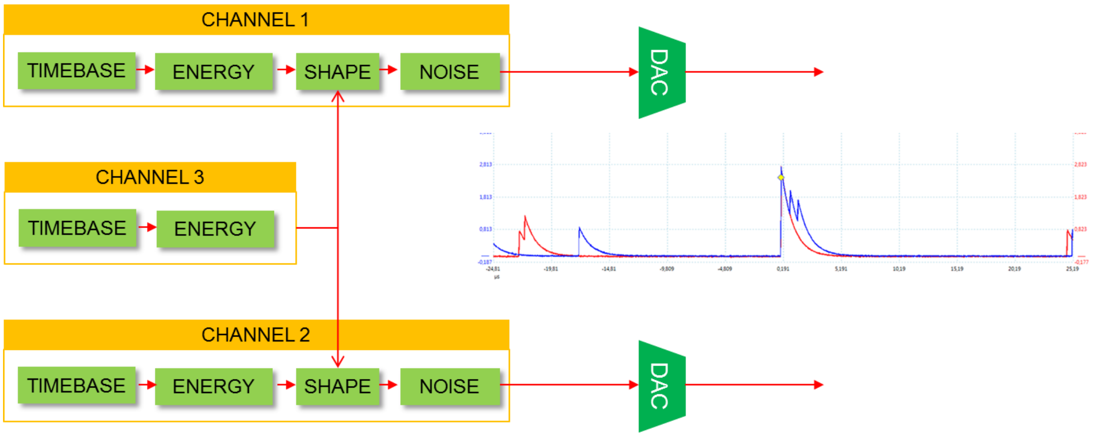


Fig. 9.4: Generation of correlated events with the internal third channel. In blue the channel 1 signal, in red the channel 2 signal. The event with the yellow bullet is generated by the third channel.

10 Firmware architecture

Overview

The hardware of the Digital Detector Emulator slightly differs from a classic arbitrary function generator. The great innovation is the ensemble of algorithms that allows the synthesis of digital signals in real-time with specific controlled characteristics. The number of operations that enables the real-time emulation of signal closest to reality is really huge. The emulator is able to generate each sample of the stream at the rate of 1.25 GS/s, taking into account the emission spectrum, the statistics of emission and therefore the pile-up between the events, the shape of the signal, the contributions of noise and interferences, the fluctuations and drift of the baseline, the shaping of the conditioning electronics.

The big difference between this model and the DT5800 is the firmware architecture used. The DT5800 used one long pipeline able to compose the emulated signal piece by piece. Such pipeline worked with a clock at 125 MHz. To achieve the generation rate of 1.25 GHz it was not sufficient to raise the system clock because no FPGA device is able to work at such a high speed. The solution has been to quadruple the core emulation: in this way the system works by simultaneously generating four samples at 250 MHz; the four samples are not only temporally staggered by T0 / 4 (T0 = 4ns) but they all belong to the same source, therefore they must respect output statistics and spectrum.

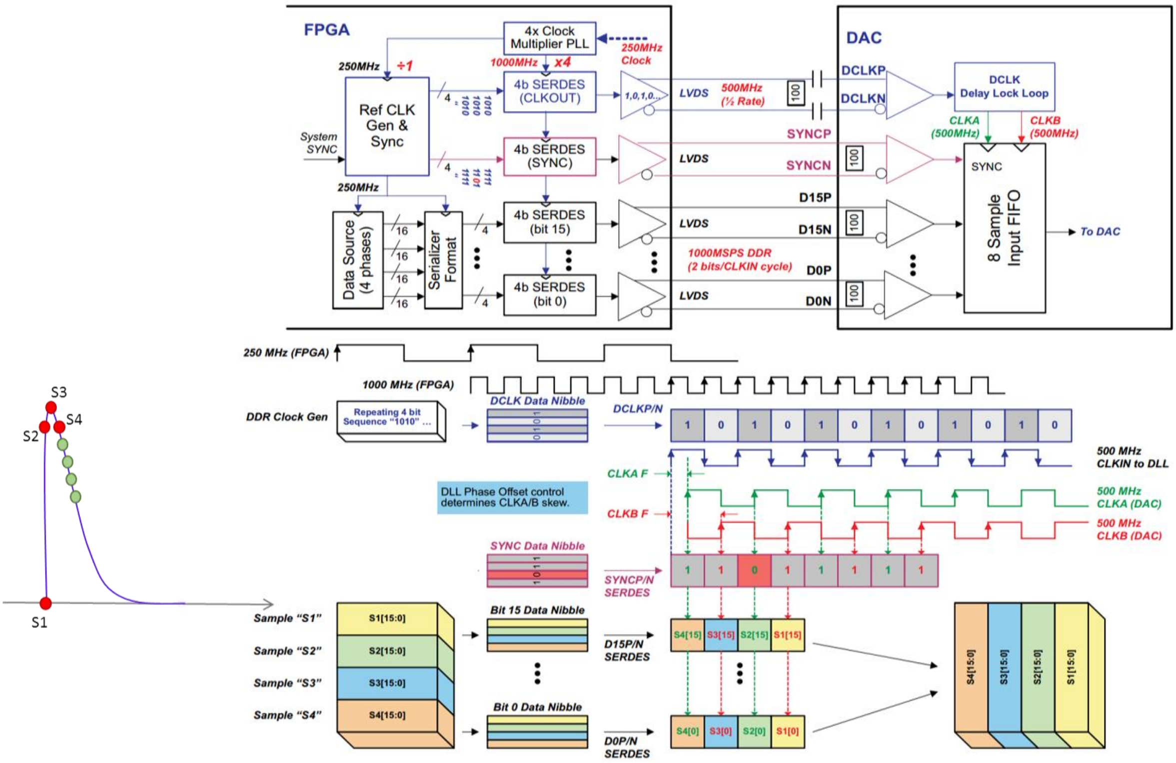


Fig. 10.1: Firmware architecture to achieve 1 GSPS emulation speed

The hardware is from 100 to 10,000 times faster than the equivalent software developed in C++ to characterize the system. To be able to accelerate the calculation as much as possible, the spatial distribution of the processing resources into the FPGA device has been deeply exploited.

Fig. 10.2 shows the structure of the firmware.

An algorithm for getting the first order statistical features of a random process from a histogram of events has been implemented. The algorithm returns the statistical distribution of amplitudes and occurrence times of the events. The generator of occurrence times acts as the trigger for the output of the signal shape, whose amplitude is scaled according to the sorted amplitude value. The user can load into the system memory the shape of the signal to be emulated. The stored shape (called *reference pulse*) is normalized to the unit maximum value. The algorithm then extracts a value of starting time and amplitude, and generates the reference shape rescaling its amplitude.

There are three independent data-paths for the pulse generation.

The first case is called “custom shape data-path”. The device provides 16 generator machines that can trigger independently with the extracted starting time. The shapes generated can be summed together to emulate the pile-up. Any kind of shape can be emulated with this data-path, even custom shapes which can be loaded through a file. The only limitation is the number of piled-up events that cannot exceed 16.
The instrument is able to simultaneously generate different shapes on the same channel (Multishape function). This feature is especially useful to simulate different processes of charge generation in the detector. The maximum number of different shapes is 4. Obviously the number of memory per shape is partitioned, further limiting the maximum number of manageable events (16 for one shape, 8 for two shapes, 4 for three to four shapes).

The second case is the “exponential data-path” that emulates a real analog pulser with no limits in the generation of piled-up events. It can only emulate exponential shapes.

The third case is the “pulsed reset data-path” that can emulate pulsed reset detectors.
In addition, the instrument implements the emulation of the baseline drift, noise, and analog output interferences. The drift of the baseline value is not statistical but it is introduced by the user as a deterministic shape, in order to allow the emulation of deterministic variations, like periodic interferences or couplings. At the rate of 1GHz, the system is able to emulate a drift profile up to 0.1s long.
White and 1/f sources of stationary noise and generic disturbances can be emulated as well. The range of frequencies of the 1/f noise extends from few Hz to more than 1 MHz. The power spectral density of the 1/f noise is obtained by properly shaping the power spectral density of a white noise generated by a LFSR. The shaping transfer function is obtained as the superposition of ten first-order low-pass filters with poles positioned at equal distance on a logarithmic scale in the range of frequencies where the 1/f shaping is desired.
The device can also emulate the random drift of the baseline.

The instrument has also an analog input channel with the additional function of MCA: to initialize the instrument from a real setup and mix in real time the emulated signal with the acquired signal. This allows to inject an artificial signal in the real signal without going to operate on the input test of the preamplifier. It is consequently possible to have a virtual source of signals to be added to the measured signal. The DT5810 has an ADC at 156 MSPS 14-bit. In order to use this signal in conjunction with the signal at 1 GSPS it is necessary a sample rate converter that works in upsampling. The signal at 1GSPS is then sent to the adder nodes of the two channels. The signal can be added to the channel 1/2 or to both of them. The user can set a gain / offset / polarity in the analog signal for each channel before it is mixed with the real signal.

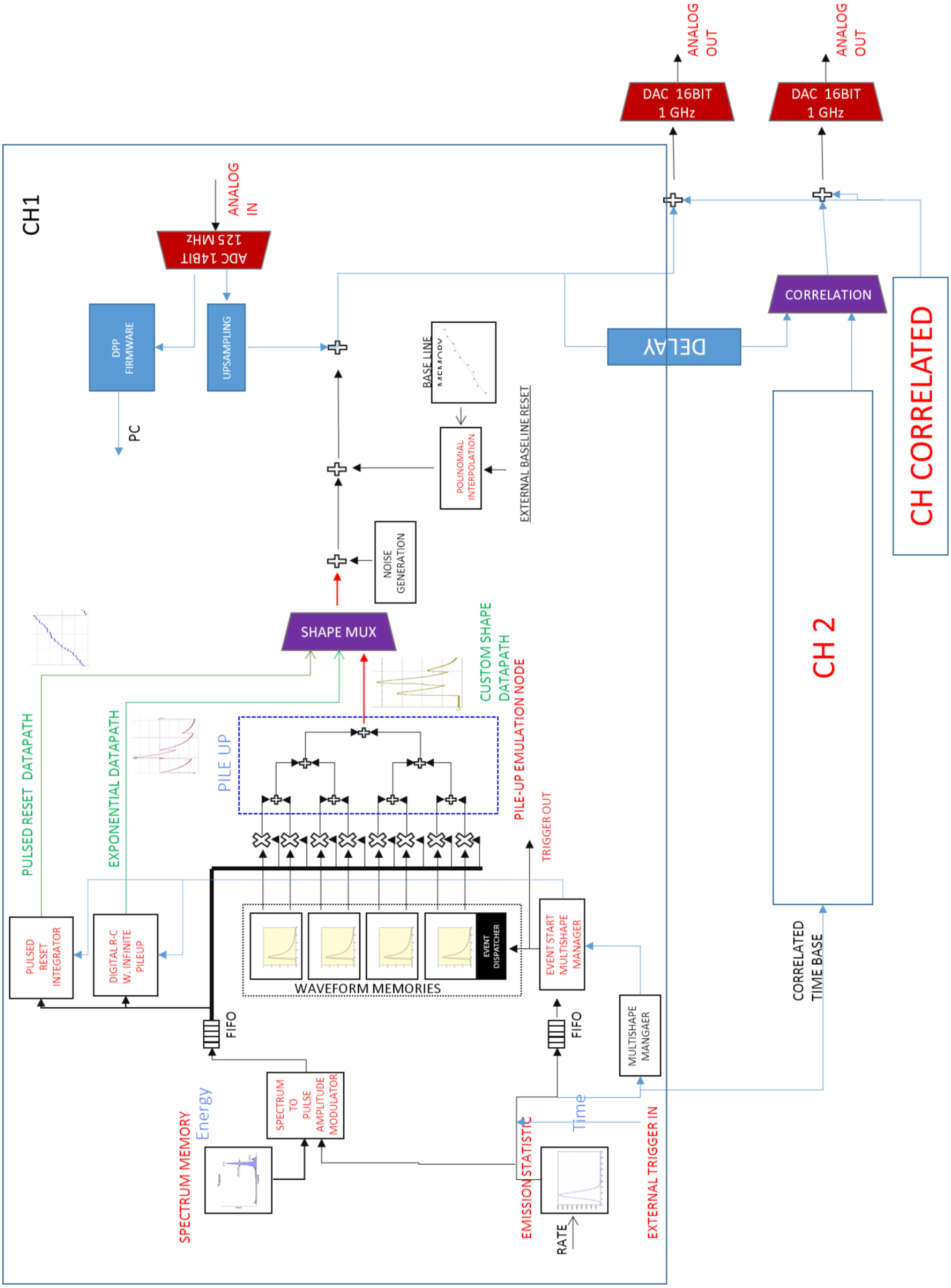
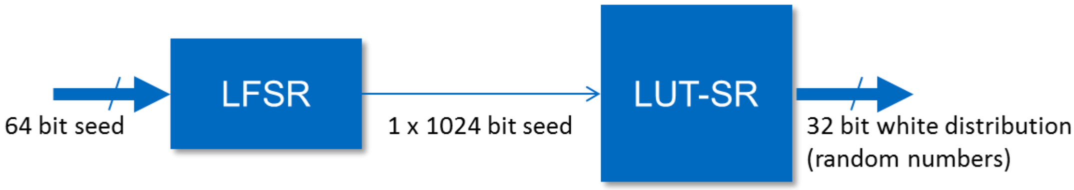


Fig. 10.2: Fast Digital Detector Emulator firmware architecture.

Random Number Generation

The emulation process is based on the generation of pseudo-random sequences that statistically reproduce the input programmed features. For example, the user can set the desired energy spectrum; then the device will convert the spectrum into a sequence of numbers representing the pulse amplitudes. A very good source of pseudo-random numbers is required to have very long sequences with neither pattern nor artefacts. This will ensure that the emulator output is as close as possible to a real output. Moreover, a pseudo-random generator allows to reproduce the same sequence many times, or to generate a statistically independent sequence any number of times. This is quite straightforward since the user has only to store the initial "seed" that enables the number generator. The seed is a 64-bit number, used as a starting point for the pseudo-random generator.



The use of a simple Linear-Feedback Shift Register (LFSR) allows to generate pseudo-random sequences with statistical properties that limit the good result of the emulation. For this reason, the LFSR are used to program a 32-bit LookUp Table-Shift Register (LUT-SR) generator that generates numbers with very small auto-correlation. The LUT-SR has a very long period, up to 2^{1024} clock cycles. Refer to **[RD1]** for further details about this method.
The structure of the number generation is shown in the figure above.

From custom distributions to a set of values

In emulating a radioactive source, a primary task is to generate the energy values following a user-defined energy spectrum and the Poisson distribution of the pulse occurrence times.
Those distributions have to be converted into a stream of values, whose probability of distribution follows the input spectra. Being a statistical variable x described by a density probability distribution f(x), it can be modelled by the cascade of a generator of uniformly distributed random numbers and the transform function F(x). This way, the quality of the generated statistic values depends only on the uniform number generator, which can be used for every emulated source, characterized only by F(x). Therefore from a white spectrum it is possible to get any kind of spectrum.
In order to explain how the algorithm works, let us consider that the energy spectrum is a histogram composed by 16 energy bins, from E0 up to E15, with a maximum dynamic range (DR) equal to 16. The bin width is the spectrum resolution, while the DR is the maximum height of each histogram column. The higher is the number of bins and the DR, the better is the represented spectrum. However, increasing the accuracy of the spectrum is simply a matter of the number of bits that can be used and this is not a problem with modern digital devices.
Each column of the histogram can be thought as composed by a number of small squares; if a generic bin x is twice higher than a bin y, it means that there is twice the probability for an event to have energy E_x rather than energy E_y. In fact, the height of the column of the bin x represents the density of probability that the event has energy between E_{x-1} and E_{x+1}. The product of the column value by the bin width returns the probability. The ratio of the probabilities that an event has energy in a certain interval rather than in another one is simply the ratio between the corresponding areas below the density probability curve.
With reference to **Fig. 10.3**, each square of each column is sequentially numbered. Consider the simplified case in which the total number of squares under the curve is a power of 2, e.g. 2^5=32. Using 5 bit in the random number generator, all the 32 numbers can be obtained with the same probability, i.e. the random numbers map completely the area under the spectrum curve. Every time a random number is generated, the algorithm searches the number in the spectrum area and, when it finds it, it delivers the bin number n thus indicating the corresponding energy value E_n. If we consider again a generic bin x two times higher than a bin y, it is easy to see that, since the random numbers with equal probability map all the squares, there is twice the probability that the random number picks up a square in x rather than in y, which means that generated pulses with E_x energy are twice those with E_y energy.
Basically, an array is loaded with the cumulative energy spectrum $H_c(E_x)$ that is computed from energy spectrum H(E).

$$H_C(E_X) = \int_0^{E_x} H(E)dE$$

This way, only one memory cell per bin is required. Using the cumulative spectrum, it is still possible to identify the bin that contains the generated random number by means of an extension of the described algorithm. For instance (see **Fig. 10.3**), if the random number is 18, it is easy to see that it belongs to the memory cell number 10: in fact, the cell number 10 contains a number that is higher than 18 (20), while the previous cell contains a number that is lower (16); this means that bin number 10 contains the squares that go from 16 to 19, exactly the range in which 18 belongs. So the output energy value correspondent to the random number 18 is 10.

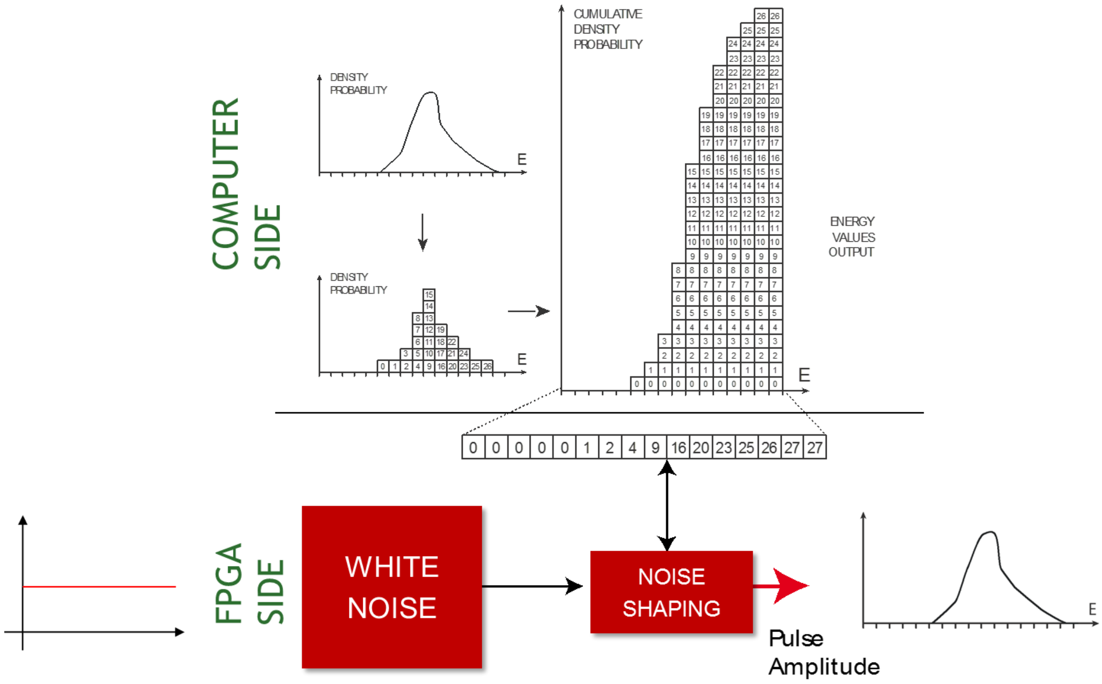


Fig. 10.3: Emulation of a spectrum.

Energy Datapath

The energy of a pulse represents the area under the pulse and for a constant shape it is proportional to the pulse amplitude itself.
The emulator has three ways to modulate the energy, as shown in the following figure.

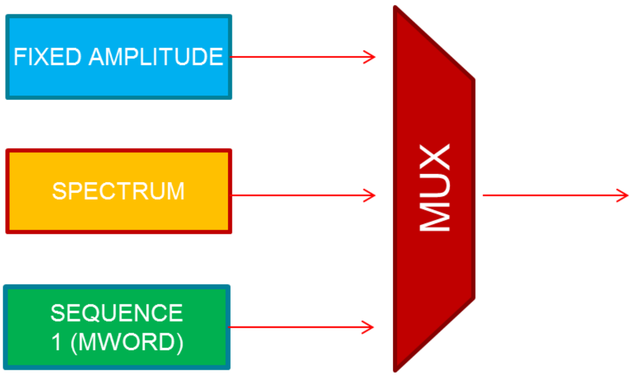


Fig. 10.4: The three ways to emulate the energy.

The “fixed energy” mode generates pulses with the same amplitude; “spectrum” generates pseudo-random sequences of values that follow the programmed input energy spectrum; “sequence” allows to load a file with a sequence of pre-defined energy values that the emulator will reproduce repetitively. The sequence is copied into the internal memory of the device. Therefore, the maximum length allowed is 1M values. The “spectrum” data-path uses the algorithm explained in the previous section. **Fig. 10.5** shows an example of generation of a set of amplitudes starting from an energy spectrum.

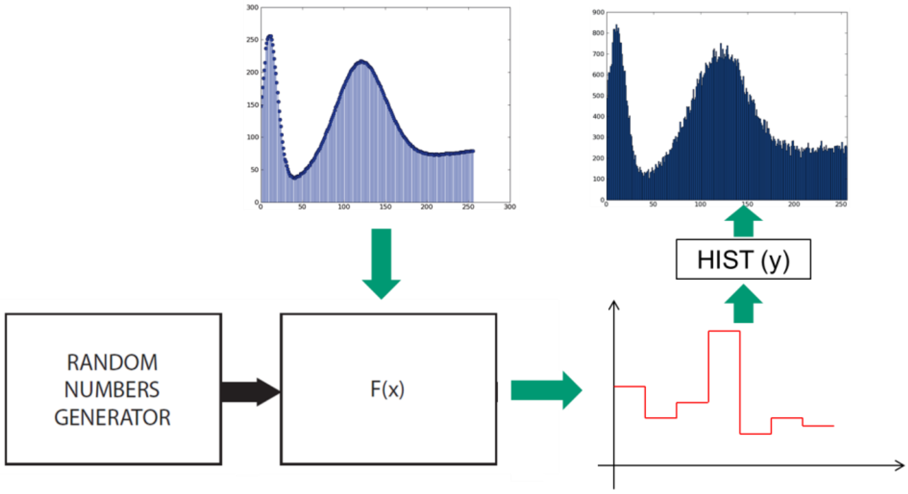


Fig. 10.5: Conversion of a programmed energy spectrum into an amplitude sequence. Evaluating the histogram on the output vector user gets the same input spectrum.

Timebase Datapath

It is possible to generate the time of occurrence of a pulse in three ways, as shown in the following figure: "fixed rate", "Poisson" distribution, thought a "custom" distribution, or a "sequence" of maximum 1M events.

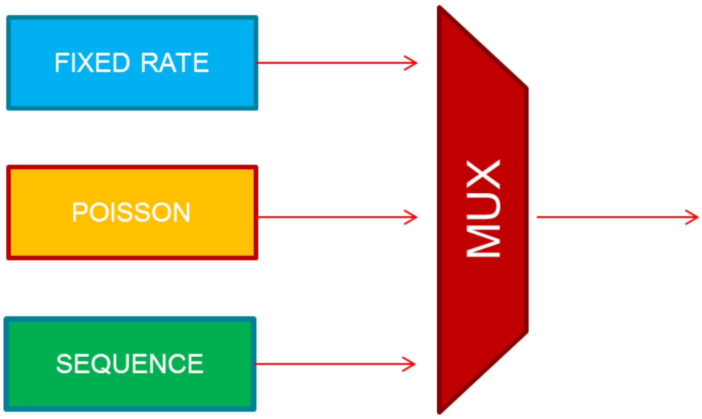


Fig. 10.6: The three ways to control the time base

The Poisson distribution is obtained through the Bernoulli Trials method. Repeated independent trials in which there can be only two outcomes are called Bernoulli trials. Bernoulli trials lead to the binomial distribution. If the number of trials is large, then the probability of k success in n trials can be approximated by the Poisson distribution.
The generation process of Poissonian events is shown in **Fig. 10.7**. For each clock cycle the device generates a pseudo-random number. If this number is smaller than $\alpha = rate/f_{clk}$, then the event is generated.

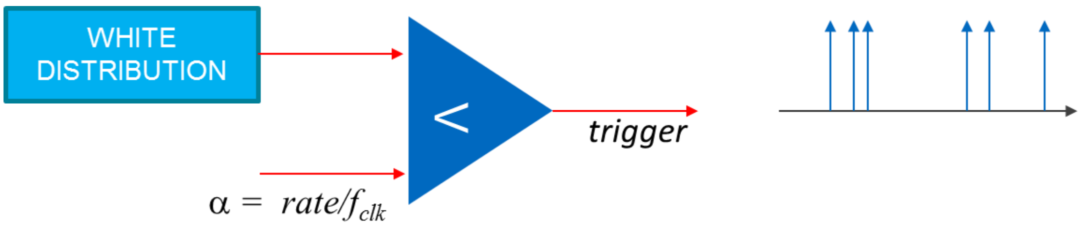


Fig. 10.7: Generation of Poisson distributed events using the Bernoulli Trails method.

For any selected data-path the emulator is able to generate two kinds of dead-time which are able to emulate detectors, or detection systems characterized by paralyzable or non-paralyzable dead-time. In a non-paralyzable detector, an event occurring during the dead time is simply lost, so that with an increasing event rate the detector will reach a saturation rate equal to the inverse of the dead time. In a paralyzable detector, an event occurring during the dead time will not just be missed, but will restart the dead time, so that with increasing rate the detector will reach a saturation point where it will be not able to record any event at all.

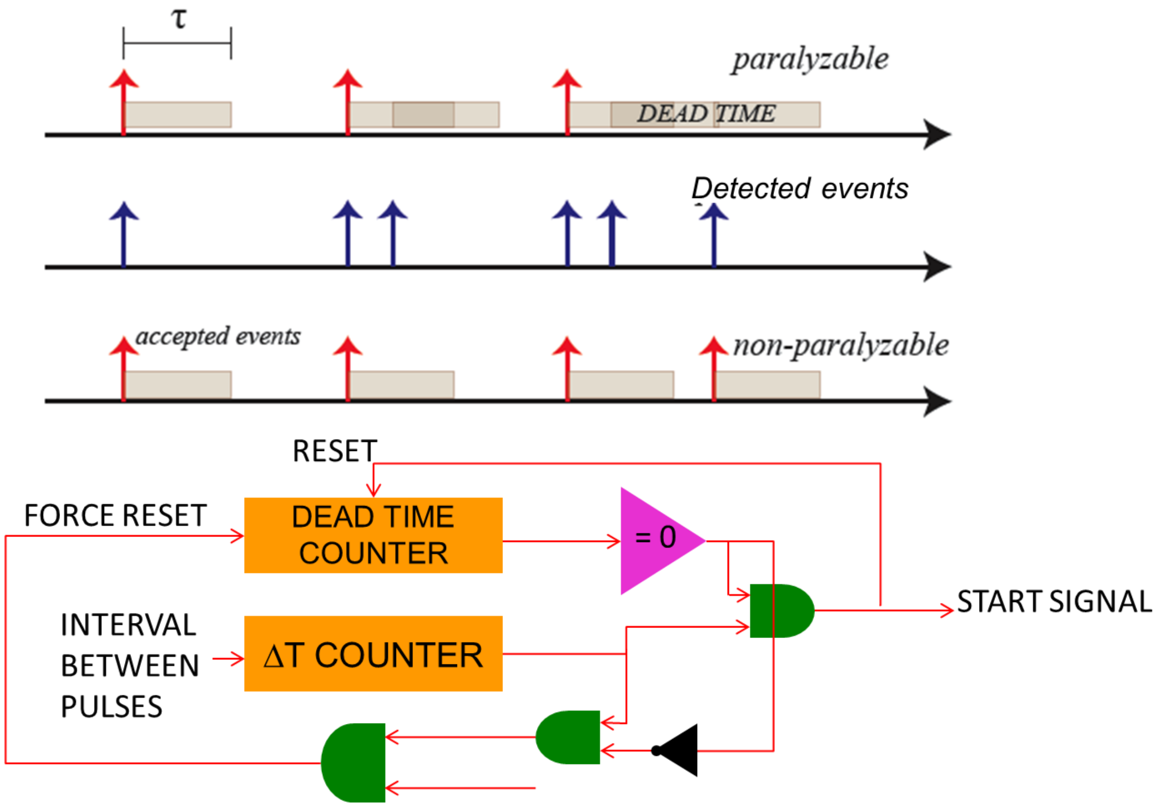


Fig. 10.8: Schematic diagram showing the principle of paralyzable and non-paralyzable dead-time on the top of the figure. On the bottom the emulation scheme of the dead-time**.**

The emulator can work also with an external trigger fed into the digital input connector. The external trigger is sampled with a precision of 4 ns and it then enables the event generation.

Shape Datapath

The emulator allows for three types of shape generation: “custom shape”, “digital RC”, and “pulsed reset”.

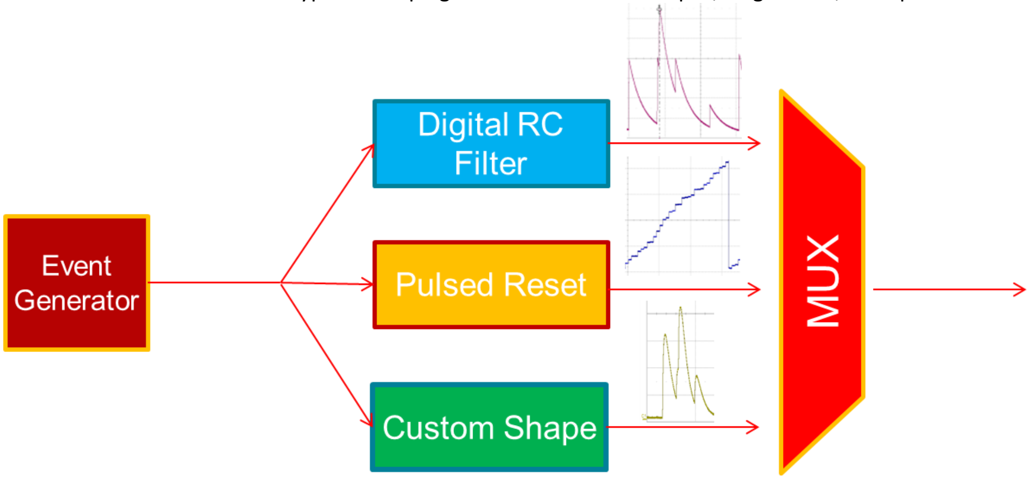


Fig. 10.9: Different datapath for shape emulation

Custom Shape – Memory Based data-path

The custom shape data-path allows to emulate any kind of shape the user can program. The control software fed a 4096 memory buffer with the points describing the shape. An interpolator is used to get shapes longer than 8 µs. Since the emulation is based on 16 memory machines, the maximum number of piled-up events is 16. If more than 16 piled-up events are requested, the exceeding events are discarded and the final statistic might not be Poissonian.

The shape generator is able to:

1. read a memory that contains the shape of the pulse
2. interpolate the output of the memory to achieve long pulse shapes using few kBytes of memory

Note: The “memory based” shape emulation is referred in the software GUI as “Exponential - Fast” shape.

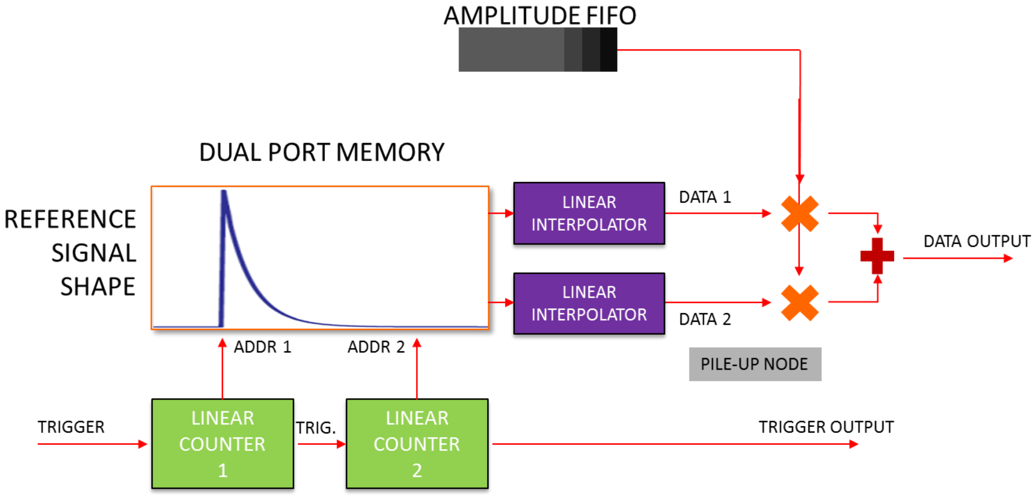


Fig. 10.10: Memory based shape generation.

A time-quantized vector that contains the shape of the pulse is stored in memory and read from the first to the last element any time a start pulse is fired from the time base generator. Reading the memory with a fixed time step dt and feeding the DAC with memory output, the emulator generates the desired waveform. The multiplier modulates the amplitude of the signal read from the memory to the value calculated according to the reference spectrum.

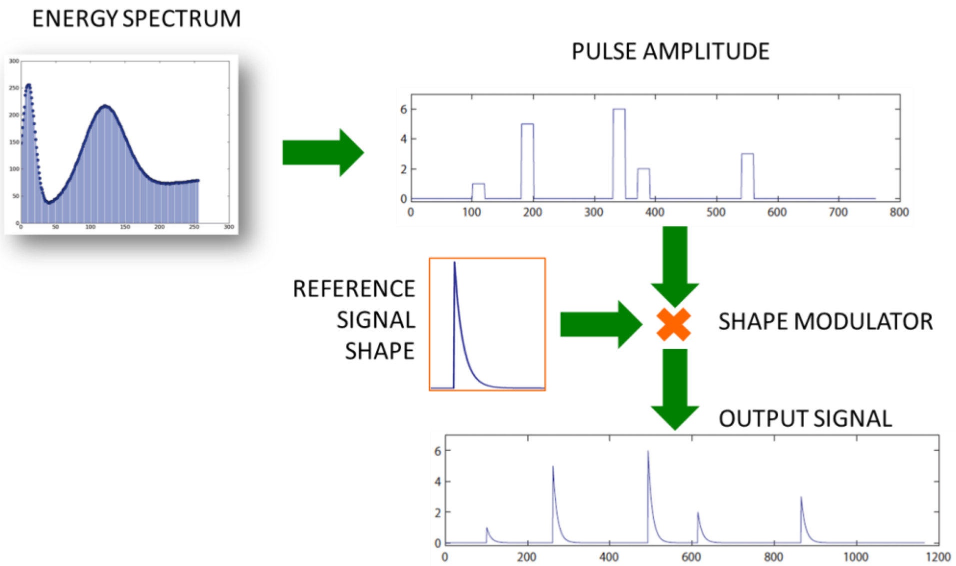


Fig. 10.11: Shape readout and amplitude modulation.

In case of exponential shape, its generation flow is as follows:

1. An ideal exponential shape is generated according to the definition
$$A * \begin{cases} e^{\frac{-(t-OFFSET)}{\tau}} & t \geq OFFSET \\ 0 & otherwise \end{cases}$$
2. The shape passes through a first-order IIR low-pass filter with bandwidth equal to $\frac{0.35}{t_{rise}}$
3. The system calculates the sample in correspondence to which the filtered shape at full-scale gets off below the LSB of the DAC
4. On this basis, the system calculates the factor of subsampling: 500 points are reserved to the rising edge, the remaining 3596 are for the tail
5. The subsampled shape is programmed and the factors for the linear interpolation of rising and falling edges are calculated

The value of time step dt between two samples depends on the interpolation function.
The interpolation allows the reduction of used memory. In order to explain how the interpolation operates, let's take as an example the output shape of a big gas-filled detector. The exponential queue associated to each pulse is typically a few milliseconds long. At the rate of 1 Gsamples/s, a signal 10 ms long requires 10 Mwords of memory to be stored that is too much for on-chip memory resources. An interpolator 1:8000 reduces the request of memory to 1.75 kwords. There are several kinds of interpolator functions that can be used to oversample the shape vector, e.g. polynomial, exponential, spline, and linear interpolation functions. Spline functions guarantee the minimum error, but require very high computational effort. Since for each piled-up event it is required one shape generator, and there are up to 16 piled-up events to be emulated, 16 shape generators are required (meaning 16 memories, 16 control logic units and 16 interpolator stages). Consequently, the complexity of the interpolator should be as smaller as possible and a linear interpolator has been implemented.
Moreover, detector signals can have sharp rise time and very long decay time. For this reason it is convenient to set two different interpolation ratios, one for the "rise time" region, and one for the "fall time" region. The algorithm is able to select a "corner" point, where the region on the left is the rise time region and the region on the right is the fall time region. The user can set the number of samples to be added between a real sample and the consecutive (see the bottom of **Fig. 10.12**).
The interpolator can work both with exponential shapes (Fast and Digital RC) and with any custom shape, even those taken from real setups.

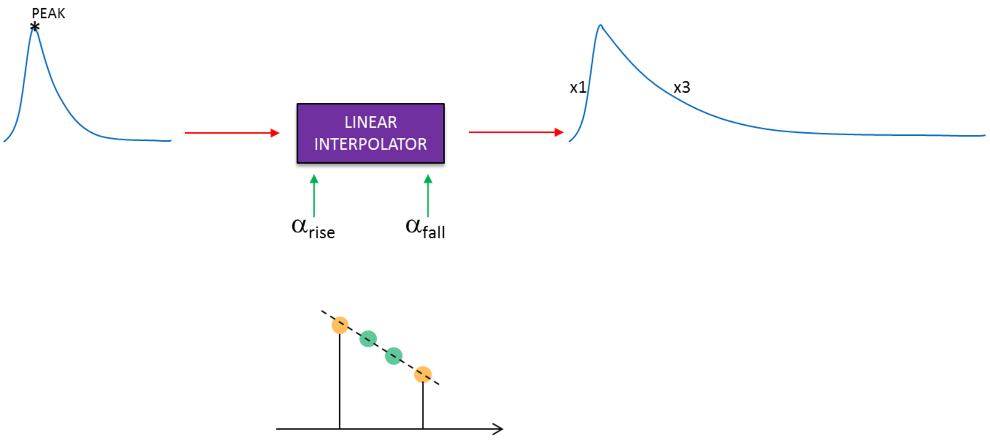


Fig. 10.12: Linear interpolation. It is possible to select a corner point that separates the "rise time" region from the "fall time" region. A different interpolation ratio can be set for the two regions (top). The user can set is the number of samples to be added between a real sample and the consecutive (bottom).

Since the time of arrival of pulses is random, it is inevitable that a pulse may arrive at a time when the previous one is not fully expired. In this case, the incoming pulse "piles up" on the tail of the previous one and appears to have a height different from its true value. Pulse pile-up depends on the counting rate. The pile-up emulation is obtained using a tree of adders that mixes the output of different shape generators. Each generator is started independently according to the START signal controlled by the time base generator. Sometimes it is useful to be able to limit the pile-up effect and simulate a detector where no pulses could be piled-up. If the user disables the pile-up emulation, a new pulse is fired only when the previous one is finished.

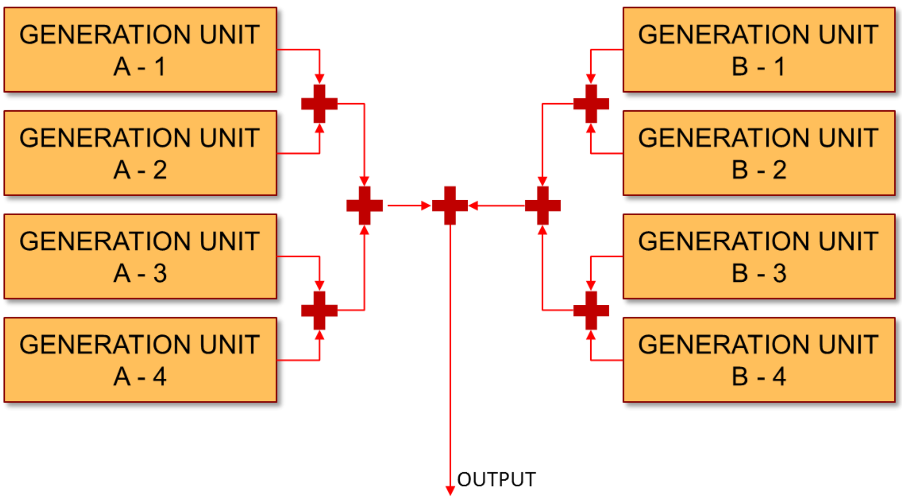


Fig. 10.13: Pile-up emulation.

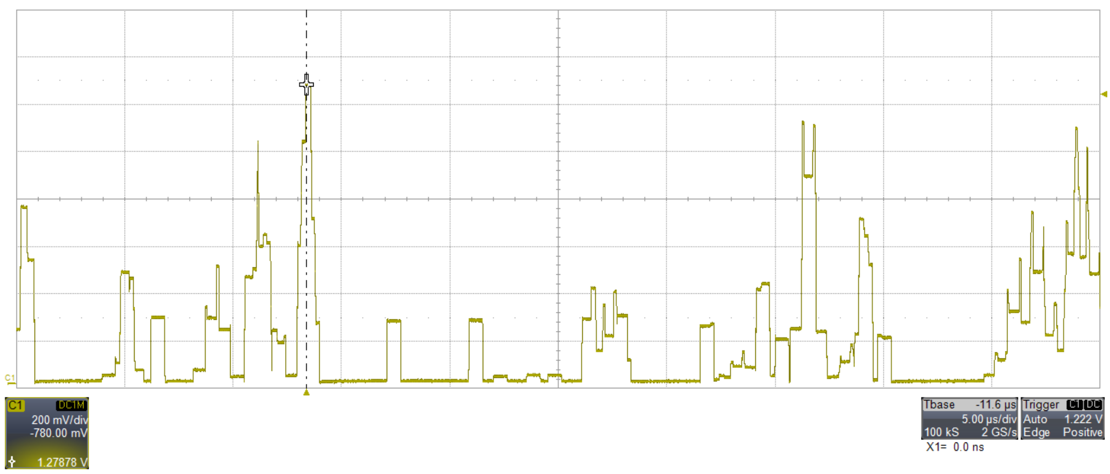


Fig. 10.14: Memory based data-path output signal.

Digital RC

In the "custom" shape option, the algorithm reads 16 memories where the samples of the shape are written. Therefore, this is a quite versatile option for emulating any kind of signals. The drawback is that in case of either high rate signals, or for very long shapes, the number of memories is not sufficient to guarantee the correct emulation.
The "Digital RC" option is able to emulate exponential-only shapes, with no limitation on the signal rate, and on the length of the exponential tails. The only limitation comes from the saturation of the analog stage; otherwise this method would give infinite piled-up events. It is based on two first order IIR (Infinite Impulse Response) filters that emulate the behaviour of a classical analog pulser.
The complexity of the IIR filters method allows the DAC to operate at 500 MSPS of frequency, half of its operating frequency. A linear interpolator is used to create the DAC missing samples.

Note: Do not use this method for pulses with decay time less than 20 ns.

In **Fig. 10.15** it is shown the implementation of the Digital RC option. Delta-like pulses are fed into the first IIR filter. The decay time of the first filter sets the pulse decay time. The second IIR filter then emulates the rise rime; it has a pole at the *rise_time/0.35* frequency.
Fig. 10.16 shows the typical analog output for the Digital RC option.

Note: Being the emulation of a real RC-RC, the output signal amplitude might depend on the rise and fall of time values too.

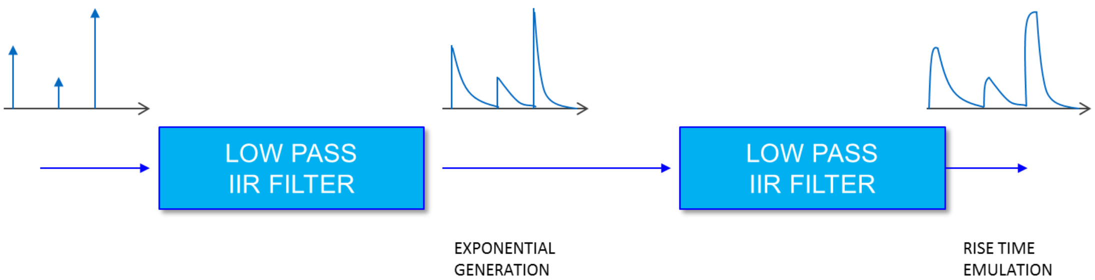


Fig. 10.15: Digital RC structure.

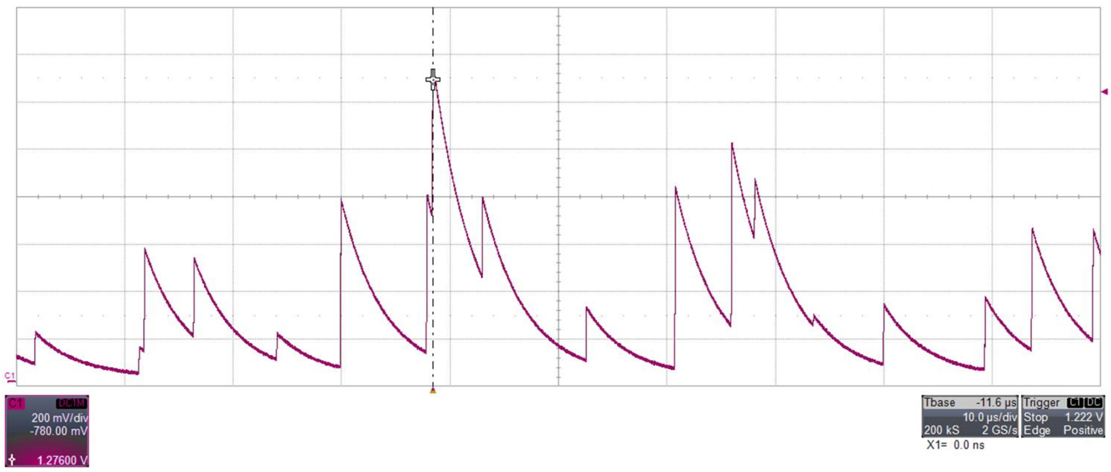


Fig. 10.16: Typical Digital RC analog output.

Pulsed Reset

The Digital Detector Emulator allows the basic emulation of a pulsed reset detector. The signal output has the typical staircase shape of a pulsed reset detector, whose amplitude and time distribution are consistent with the input energy and time distributions. In the basic option the user cannot set the shape of the signal nor the reset discharge.
This block is an integrator that sums the pulses up to a fixed threshold. Then it resets it to the programmed minimum value. The typical output of the pulsed reset basic function is shown in **Fig. 10.17**.

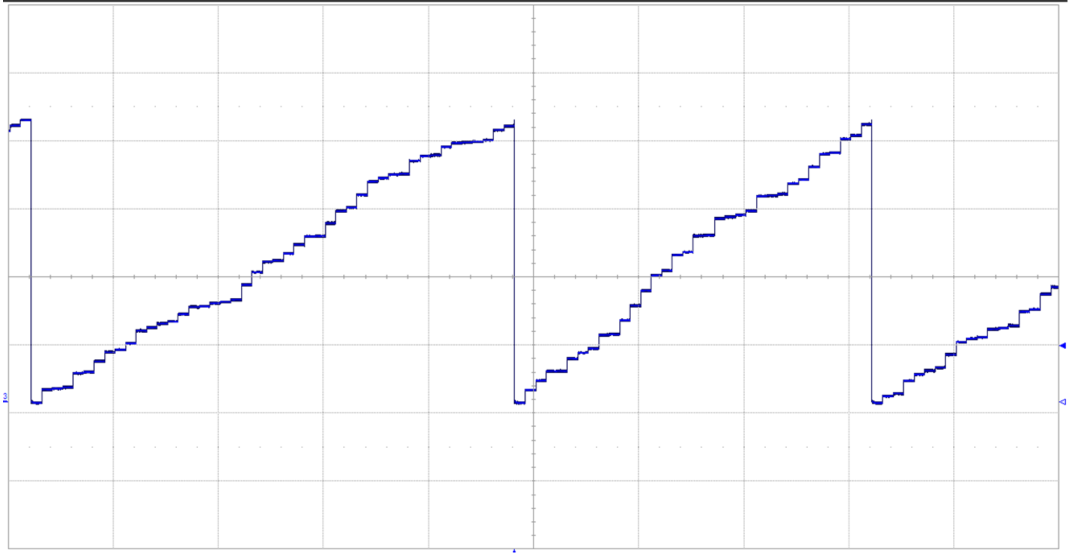


Fig. 10.17: Typical output of the basic pulsed reset option.

Noise Emulation

The Digital Detector Emulator allows the emulation of different types of noise, as shown in **Fig. 10.18**. The user can program the spectral features and the amplitude modulation independently.

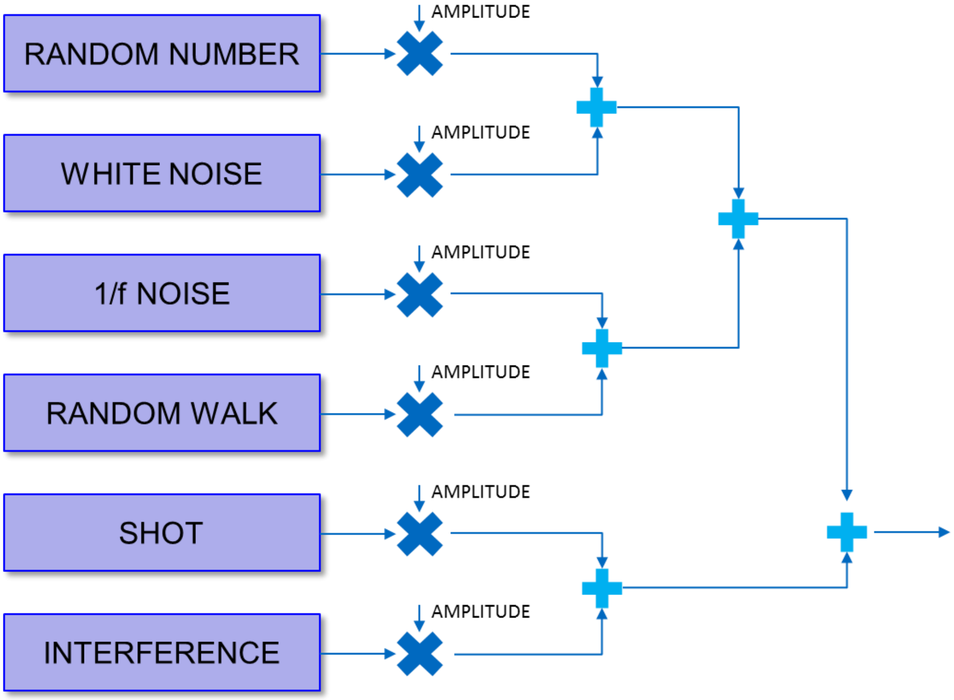
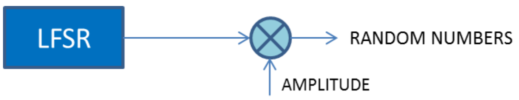


Fig. 10.18: Noise generation architecture.

Random Number

The Digital Detector Emulator generates pure random numbers. The output of the LFSR is modulated according to a constant amplitude spectrum.



White noise

The emulated white noise is a wideband noise with a constant spectral density (up to 200MHz) and a Gaussian distribution of amplitudes.

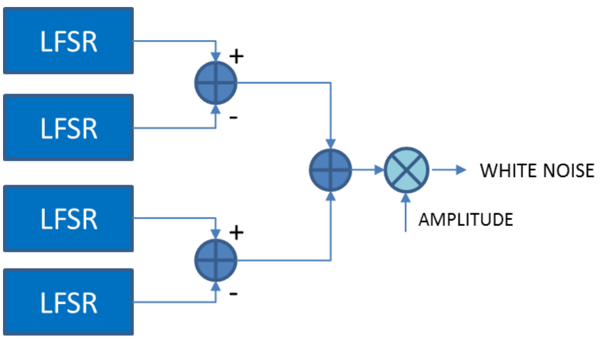


Fig. 10.19: White noise emulation process

The white noise is generated as RND1-RND2+RND3-RND4, where RND*n* is the random number generated by the LFSR generator *n*. The noise spectrum approximate well a real white noise up to 250 MHz (limitation due to the DAC frequency) if neither the anti-alias nor the output filter are applied. If they are applied, then the white noise band decreases up to 20 MHz.

Flicker noise

Flicker noise (-10 dB/dec) can be obtained by properly shaping a white noise according to the formula:

$$S_{pink} = S_{white} \cdot \left| T_{shaper} \right|^2$$

where the transfer function is

$$T_{shaper} = \sqrt{\frac{\alpha_{pink}}{f}}$$

A simple way to obtain the desired transfer function is to approximate the function with the sum of a given number of first order low pass filters with n singularities positioned at an equal distance on a logarithmic scale in the range of frequencies where we want to implement the 1/f noise shaping.

$$\log(f_{pole}(i+1)) - \log(f_{pole}(i)) = \left(\frac{\log(f_{max}) - \log(f_{min})}{n-1} \right)$$

with

$$\frac{f_{pole}(i+1)}{f_{pole}(i)} = \left(\frac{f_{max}}{f_{min}} \right)^{\frac{1}{n-1}}$$

where n is the number of low pass filters used (LPFs) and f_{min} and f_{max} are the minimum and maximum frequencies representing the limit of the pink noise shaping range. The gain of the LPFs - at low frequency- has to follow the following relation:

$$G_0(i) = \sqrt{\frac{f_{pole}(1^{st})}{f_{pole}(i)}}$$

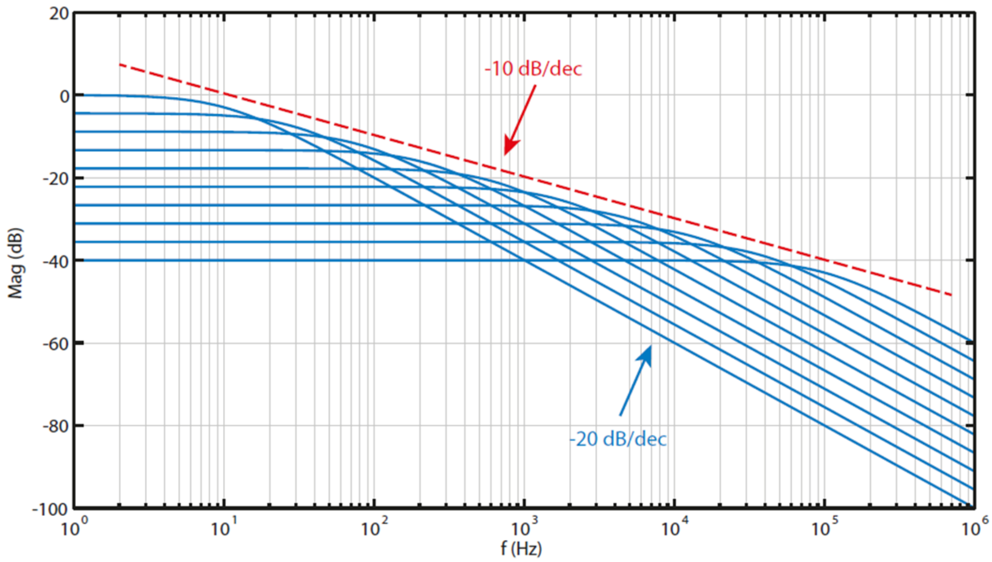
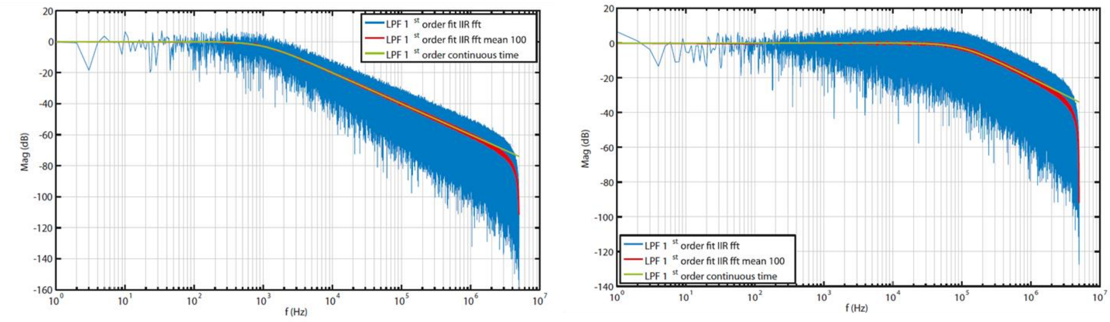


Fig. 10.20: Implementation of 1/f noise shaping filter: we can observe that the envelope of the frequency responses of the first order low pass filters (-20 dB/dec) has the same trend of the 1/f ideal noise (-10 dB/dec).



LPF with pole positioned at 1 KHz LPF with pole positioned at 100 KHz

Fig. 10.21: Comparison between a continuous time low pass filter (LPF) with pole at 1 KHz (and 100 KHz) and the corresponding digital implementation obtained through an IIR digital filter. The frequency response in the digital domain has been obtained by applying the FFT function to the output of the filter with a white noise with spectral density equal to 1 as input (blue line). The red curve is extracted from the blue one by averaging on 100 samples.

Random walk

Very low speed baseline drift approximated as the integral of a Gaussian white noise.

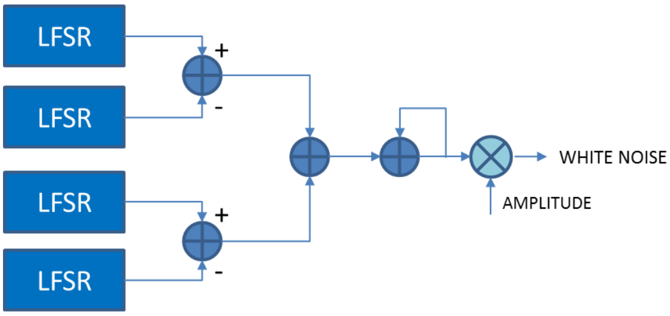


Fig. 10.22: Random walk emulation process

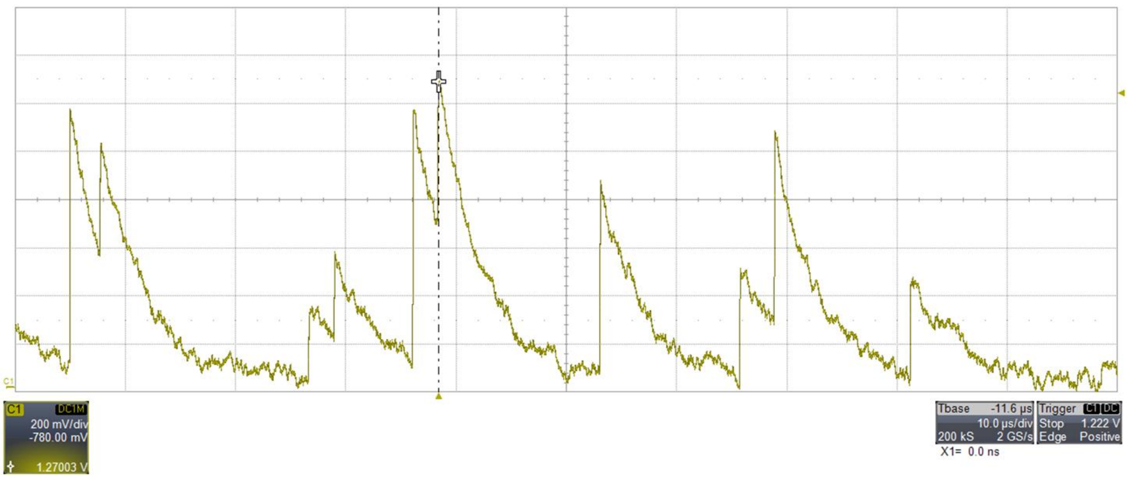


Fig. 10.23: Analog output signal with white and 1/f noise.

Baseline drift emulation

The reference ("baseline") of the electrical signal representing the detected event can be not stable because it depends on both the impinging radiation rate (D.C. coupled systems or systems with poor pole-zero compensation), and on the thermal drift of detector and read-out electronics. The baseline drift is a source of inaccuracy and broadening of the spectral measurements, and usually its amplitude is estimated and subtracted from the signal by means of a custom analog circuit (baseline restorer) or by proper processing techniques.
The Fast Digital Detector Emulator is able to emulate the drift of the baseline in a time following a programmable non-linear profile. **Fig. 10.24** shows the block diagram of the baseline emulator stage. A small memory that contains the key-points of the baseline function is read and the output is interpolated by a linear interpolator.
The reset architecture emulates the quenching circuit that restores the baseline value. The reset takes place any time the memory is completely read or by an external trigger signal. The external reset input allows the debug of the quenching circuit of the user analog front-end in a detection system. For each key-point the user can program the interpolation factor, choosing among "slow" and "fast". A long line can be therefore defined by setting only the two limit points. A slow interpolation is performed to emulate slowly variable signals up to drifts of several seconds. A fast interpolation is performed to emulate fast variable signals corresponding to the presence of circuits of reset or quenching. The precision of both interpolation procedures can be regulated by means of the slow and fast interpolation sliders.

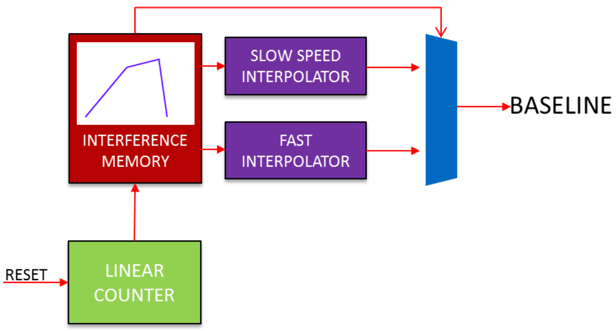


Fig. 10.24: Baseline drift emulation**.**

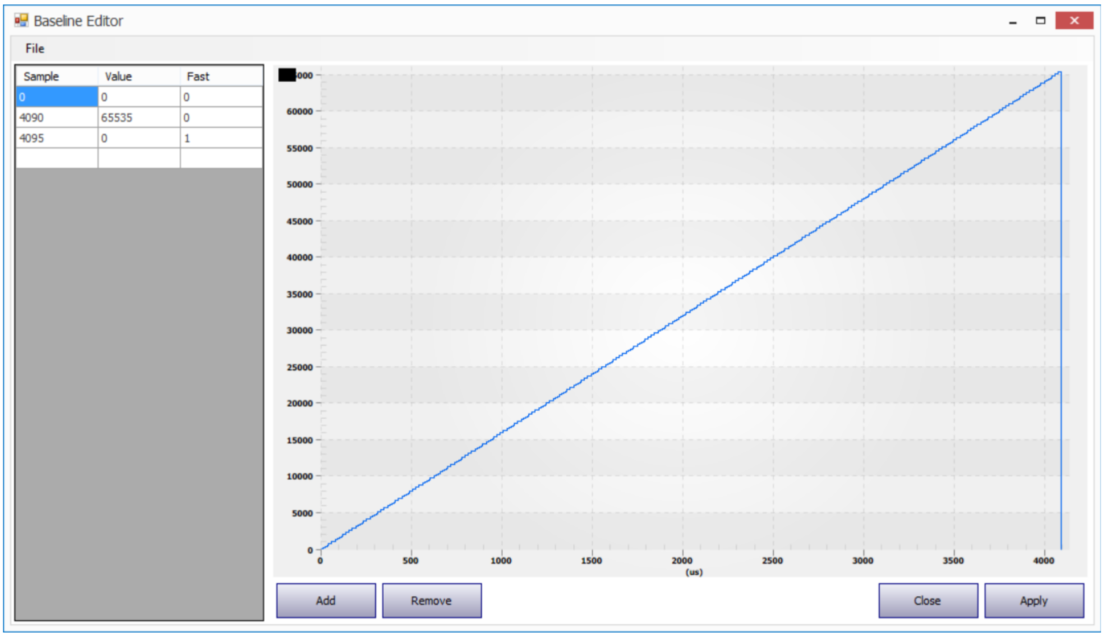


Fig. 10.25: Baseline drift emulation. In the table an example of values that the user can set and the corresponding preview.

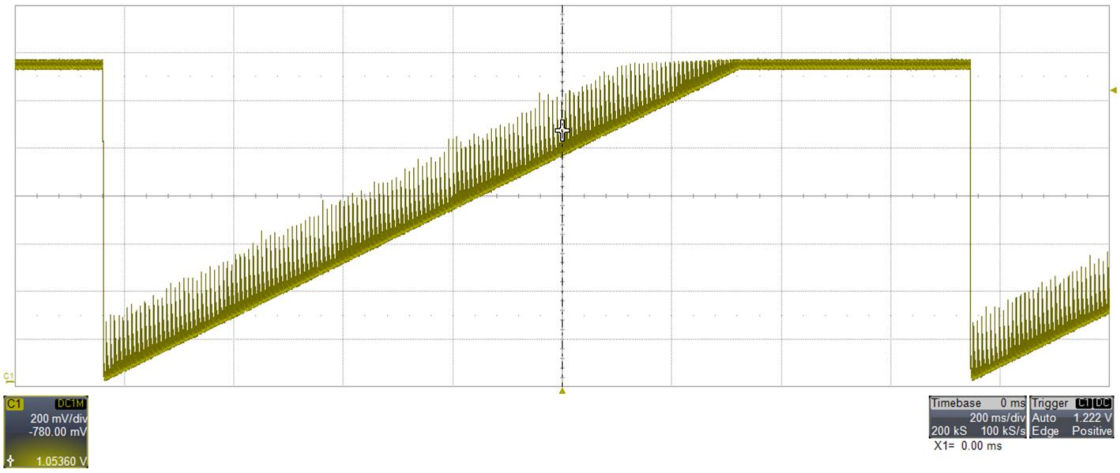


Fig. 10.26: Typical output of the baseline drift emulation with superimposed pulses.

Multi-shape emulation

To debug pulse shape discrimination systems, the emulator is able to generate up to four different programmable shapes on the same output channel. The user can program statistical distributions of energy and time as well as the number of pile-up events. The “custom” shape (memory based) option should be enabled. The user can program the relative percentage of four different shapes to be generated.

Memories are organized into blocks in order to simulate the pile-up for the different forms. With two active forms, 8 memories are available for each form and with 3 or 4 active forms, 4 memories per form are available.
For any start signal each block decides which shape has to be generated, according to the programmed percentage.

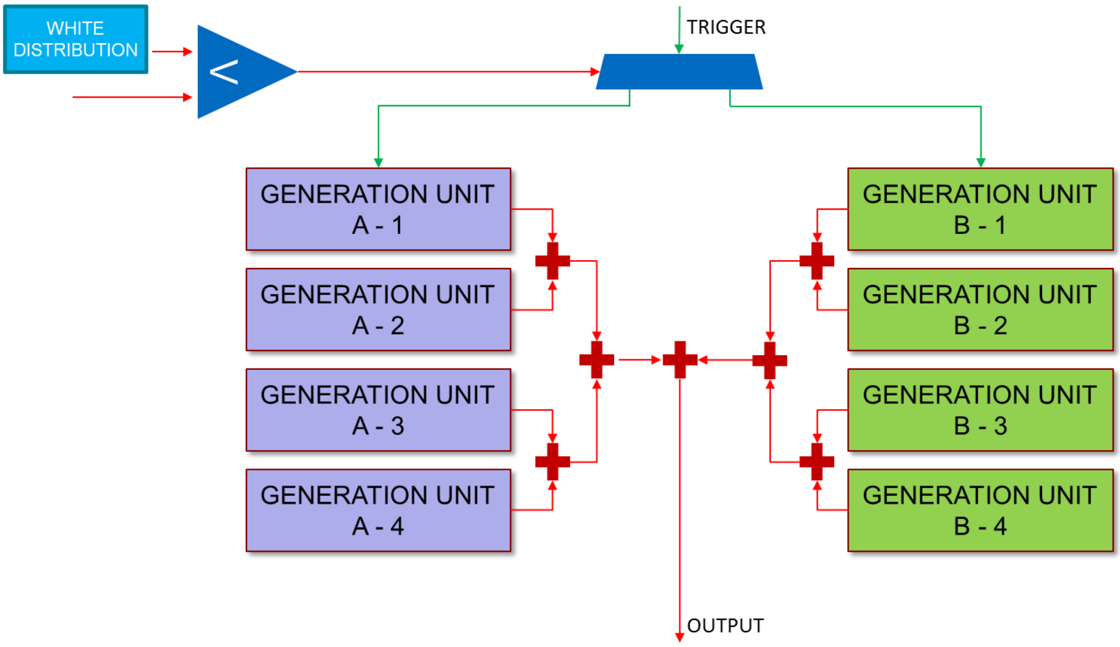


Fig. 10.27: Multi shape emulation algorithm**.**

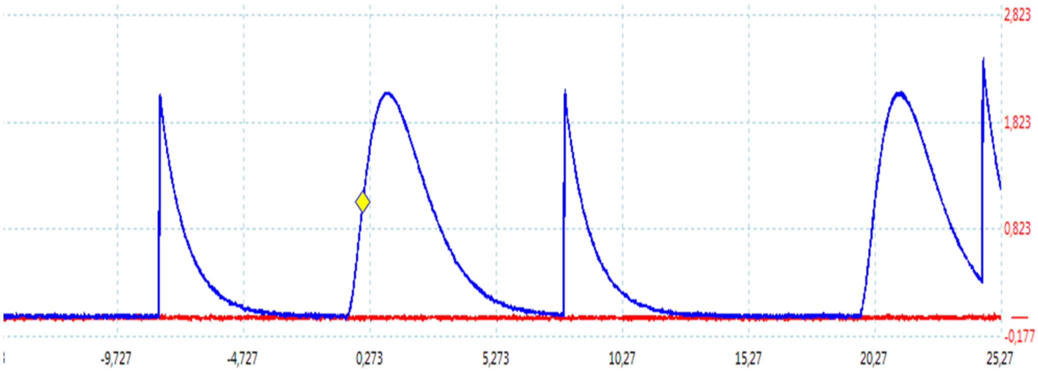


Fig. 10.28: Typical output of one emulated channel with two different shapes (50% mixture).

11 Getting Started

Scope of the chapter

This chapter is intended to provide a quick guide of the Detector Emulator Control Center Software in order to manage the first practical use of the Digital Detector Emulator.

System Overview

The Digital Detector Emulator system proposed in this chapter consists of the following products:

- DT5810B, 2-channel with standard firmware revision 10.0.0.0
- Digital Detector Emulator Control Center GUI release 2.5.0.0, running on the host station

Note: The DT5810D model is not supported by software release > 2.3.8.0 and firmware release higher than 6.2_3.0

Hardware Setup

The getting-started demo proposed in this chapter makes use of the DT5810B connected via USB to a computer equipped with Microsoft Windows 10 Pro 64-bit OS. The Digital Detector Emulator driver and software are properly installed on the work station (see the next section). The analog output of DT5810B is sent to the CAEN Desktop Digitizer DT5725 with DPP-PHA firmware. The DT5725 readout is monitored through the MC2 Analyzer Software, running on the host station. Refer to **[RD2]** for more information about the DPP-PHA Control Software installation and practical usage. Alternatively, it is possible to use one of the CAEN Digitizers with DPP firmware installed -as the 751 or 730 families- or an oscilloscope.

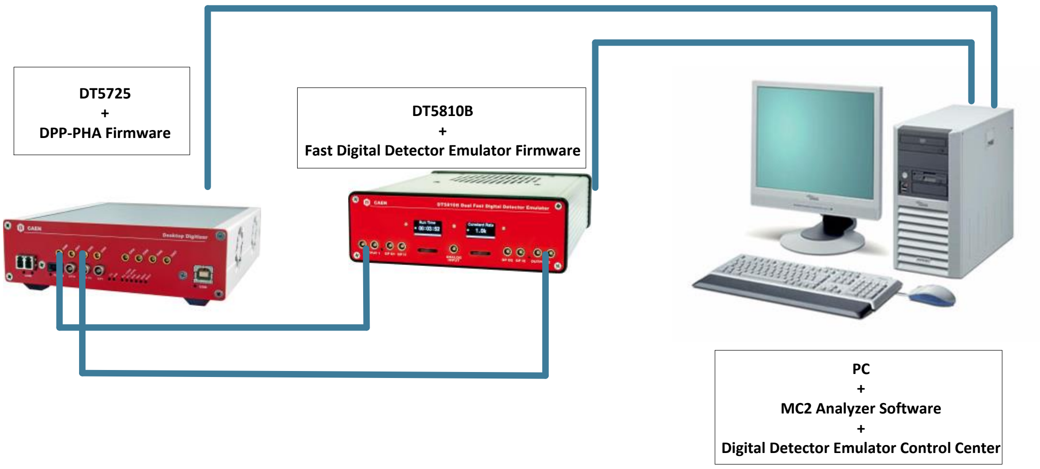


Fig. 11.1: The hardware setup including the Digital Detector Emulator DT5810B and the DT5725 used for the practical application.

Note: The DT5810B model is not supported by software release < 2.5.0.0

Software and Drivers

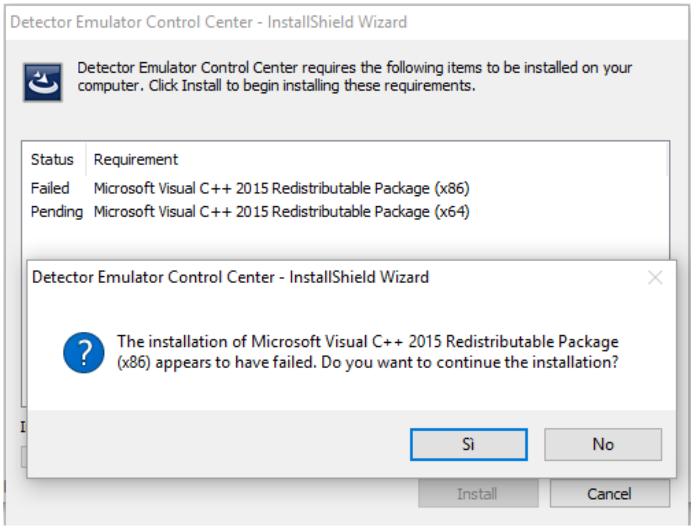
The Digital Detector Emulator Control Center is compliant with Windows 7 SP1, 8 and 10 OS, both 32 and 64 bit.

- **Digital Detector Emulator Control Center** for Windows OS.

  Download the standalone **Caen Digital Detector Emulator Control Center 2.5.0.0** full installation package on CAEN website in the 'Download' area of the Digital Detector Emulator page (**login is required before the download**).

  Unpack the **installation package**, login as administrator, **launch** the **setup file (as Administrator).**

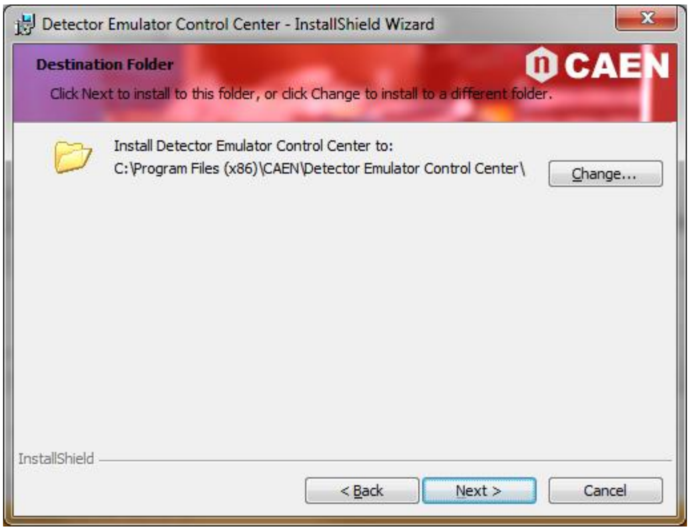
  If Microsoft .NET is not already installed, the setup will ask to install it. The operation may take some minutes.



  If Microsoft .NET is already installed, the setup returns a failure message which means that it detected a previous installation. **Press yes** to continue with the installation.

  Complete the **Installation wizard.**

Select the **destination folder**

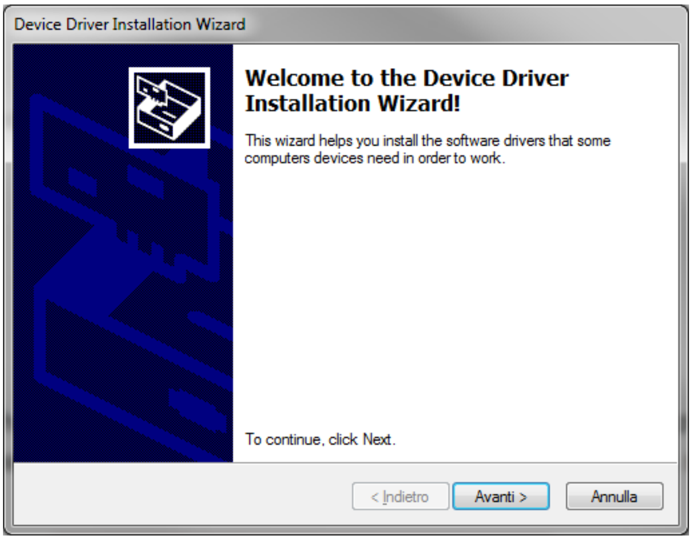


The setup will create a Desktop icon.

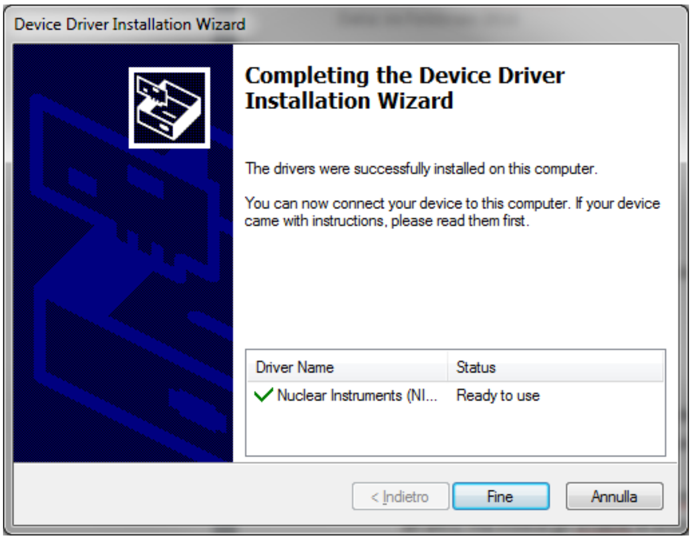
Click **Finish** when the installation is complete.

The setup will then start the installation of the **drivers**

Click on **Extract** to unpack the driver installation package and start the installation wizard



Click Finish when the driver installation is complete

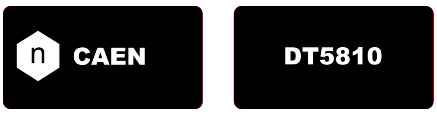


- **Drivers has been installed during setup.**
- **The Installation Setup install in the Detector Emulator Control Center installation folded a a C/C# SDK with a demo code ready to be used but the user.**

Note: If the driver installation fails, the user can install them manually from the ‘Drivers’ folder in the CAEN Digital Detector Emulator installation folder.

✓ Power On

Power ON the Digital Detector Emulator. The device starting may last about 30 seconds. On both the instrument displays the boot logo will appear. When the instrument is ready the two blue led switch off.



Connect the emulator to a USB port.

Firmware

The Digital Detector Emulator Control Center works with the **Digital Detector Emulator Firmware**. **The firmware is already installed into the device**.

In case the installed firmware is not up-to-date, a message will appear as the instrument is connected. The user will need to update the firmware.

How to upgrade the firmware

Despite the DT5810 has Ethernet connection, the firmware can be updated via USB only.

1) Turn off the instrument
2) Press the Factory Reset button on the back of the instrument and switch on the instrument until the two output enable leds start to blink
3) Release the Factory Reset button
4) Connect the instrument to the PC via USB.
5) Open the upgrade utility from the start menu -> CAEN -> DT5810 Firmware upgrade

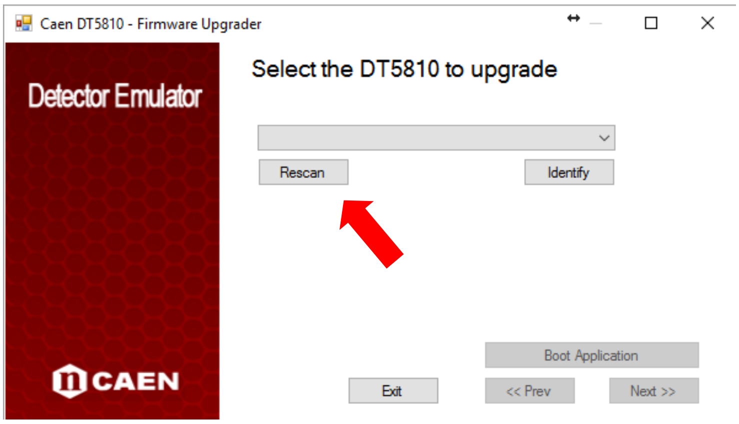


Fig. 11.2: Firmware Upgrade window – Startup

6) Choose the **ID** of the emulator to update in the top menu or press **Rescan** if you want to check the connected instruments

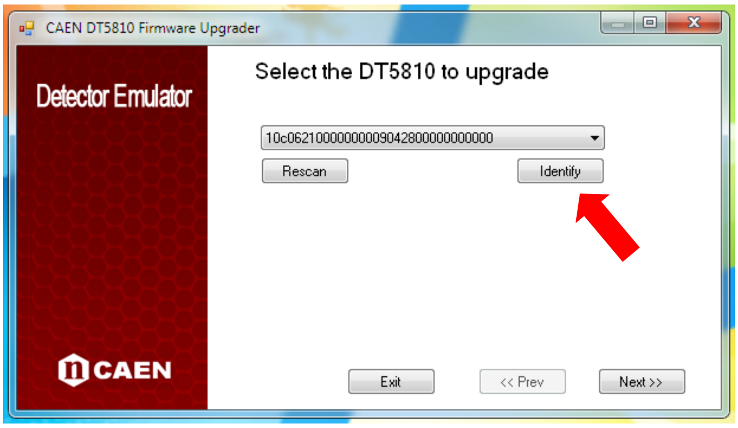


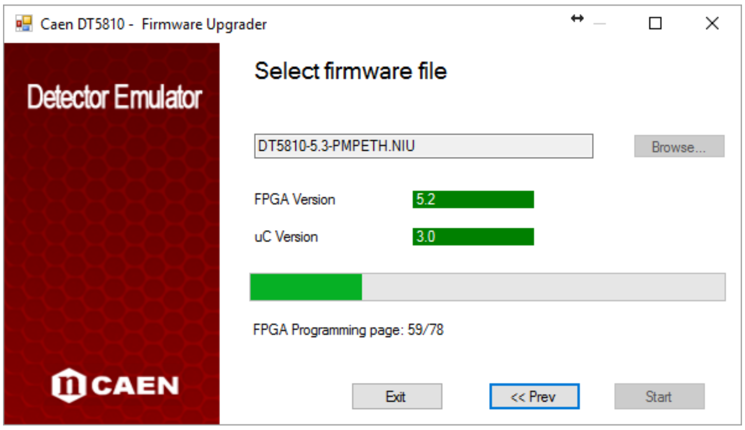
Fig. 11.3: Firmware Upgrade window – Hardware identification

If there are more then one DT5810 connected to the PC press **Identify**. The output enable leds start to blink faster for ten seconds.

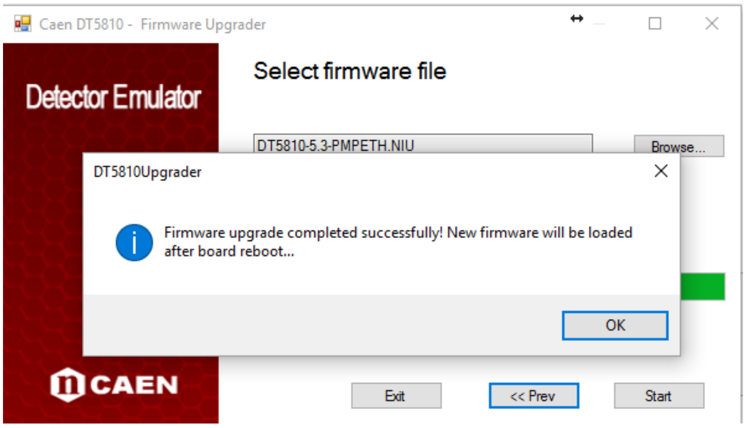
7) Press the Next button.
8) Choose the file containing the firmware to load and press update. The software will start the upgrade process either for the FPGA device and for the CPU. USB3 controller firmware is automatically updated when you update the device driver of the instrument

Fig. 11.4: Firmware Upgrade window

Note: The FPGA programming can take some minutes. The green bar shows the upgrade status. Please, **wait until the end of the firmware upgrade**.



9) When the firmware is successfully upgraded, **restart** the emulator software and the device.



Note: The DT5810D and DT5810B features two firmwares not compatible one with each other. The Firmware Upgrader is aware of such an incompatibility and does not allow the user to upload the DT5810D firmware (5.3.0.0) into the DT5810B (10.0.0.0) and viceversa.

Practical Use

The following step-by-step procedure shows how to use the Digital Detector Emulator Control Center and how to get a simple output signal.

1. **Check that the whole hardware in your setup is properly connected and powered on**
2. Download the **examples.zip** file from the 'Download' section of the Digital Detector Emulator from CAEN web site
3. **Run the software**

   Run the **DDE Control Center software** according to one of the following options:

   - The **Desktop icon**
   - The **Quick Launch icon**
   - The **.exe file** in the main folder from the installation path on your host
4. **Connect to the emulator**

At the moment Only the USB connection is supported

The software starts in demo mode. This means that the tool connected by default is a generator software that allows to explore all the features of the detector emulator and prepare configuration files even in the absence of the instrument.

On the left of the startup screen will appear a list of all the emulators connected to the system. This list is updated automatically. If an emulator is connected and the list does not show it, the user can press the Rescan button in order to force a rescan of either the USB bus and the Ethernet. If the emulator does not appear, you need to check the drivers accuracy and your PC network settings.

Note: If the DDE drivers are not properly installed, the device will not appear in the list of emulators available.

Note: The DT5810B model is not supported by software release < 2.5.0.0

Note: The DT5810D model is not supported by software release > 2.3.8.0 and firmware release >6.2_3.0

Note: At the moment only the USB connection is supported.

To connect to an emulator simply double-click the name of the device. The LED on the side indicates its status:

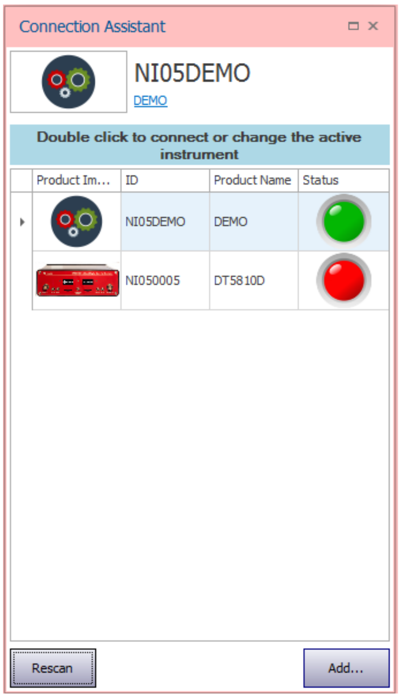
Red: emulator ready but not connected

Green: emulator connected and ready

Yellow: The emulator has been connected but the USB driver is having communication problems

When an emulator is connected, it automatically becomes the active device and it is controlled by the software. The graphs show the information related to that specific emulator. It is possible to connect several emulators - even heterogeneous ones- to the same software and check parameters simultaneously. Double-click on an already connected DDE in order to interact with that particular device. The connected emulator is shown on top of the list. The **Disconnect** button allows to disconnect the emulator from the software so that it is released to the operating system and can be controlled with a different instance of the software or by the library.

Note: When a DDE is disconnected, the emulation process does not end until the power is off, and the on / off control of the channels via jog dial can continue as well as the navigation through menus.



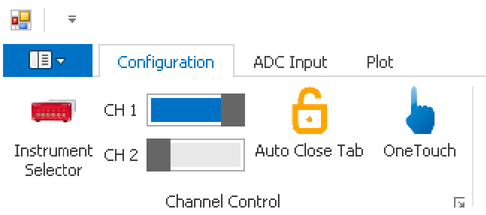
5. OneTouch interface

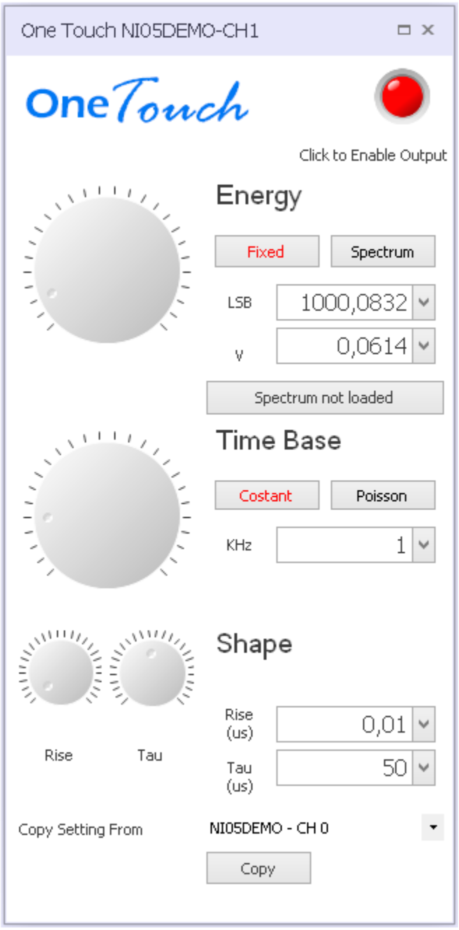
The Detector Emulator comes with two main software GUIs, the **general GUI** that provides all the advanced functionalities of the Digital Detector Emulator, and the **One-Touch interface**, a simpler GUI which allows the user to setup a signal in a very straightforward way.

The OneTouch Interface generates exponential shape analog output with as many piled-up events as half of the DAC frequency.

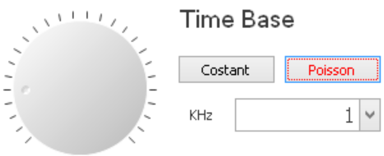
The user can set the analog output signal rate, the amplitude and the rise and fall time of the exponential shape.

The user can open the one-touch interface by pressing the icon with the finger on the upper ribbon bar. The One-Touch interface that opens, is always referred to the current emulator.





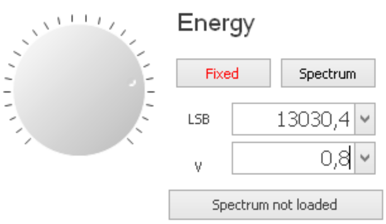
<u>Action 1</u>: set the **RATE** to **"Poisson"**, and change the **rate value** to 1 kHz.



Note: The user can **change the rate value** by either moving the knob pointer or by writing the value in the underlying box.

Note: After typing the value of a parameter in a box menu, it will be automatically applied.

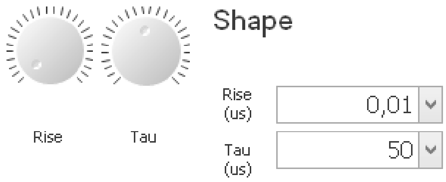
<u>Action 2</u>: set the **AMPLITUDE** to **"Fixed Value"**, and change the **amplitude value** to 0.8 V.



Note: The user can **change the amplitude value** by either moving the knob pointer or by writing the values in LSB or Volts in the underlying boxes.

Note: After typing the value of a parameter in a box menu, it will be automatically applied.

Action 3: set the **SHAPE** parameters. **“Rise Time”** = 0.01 us, and **“Fall Time”** = 50.00 us.



If the emulator is set to generate custom forms or multishapes or fast exponential acting on these controls, the form will be automatically forced to Digital RC.

Note: The user can **change the shape values** by either moving the knob pointer or by writing the value in the underlying box.
Note: If the user set the Rise Time = 0.00 he/she is asking for the best the device can provide, i.e. about 1 ns (due to the DAC sampling).

Action 4: copy settings from channel 1 to channel 2.

Use the appropriate list to copy the channel 1 settings to channel 2

Action 5: enable channel 1 and channel 2 **OUTPUT**.

Press the red LED to enable the channel. The LED will turn green and the emulation process will be initiated.

The front panel LED will be ON.

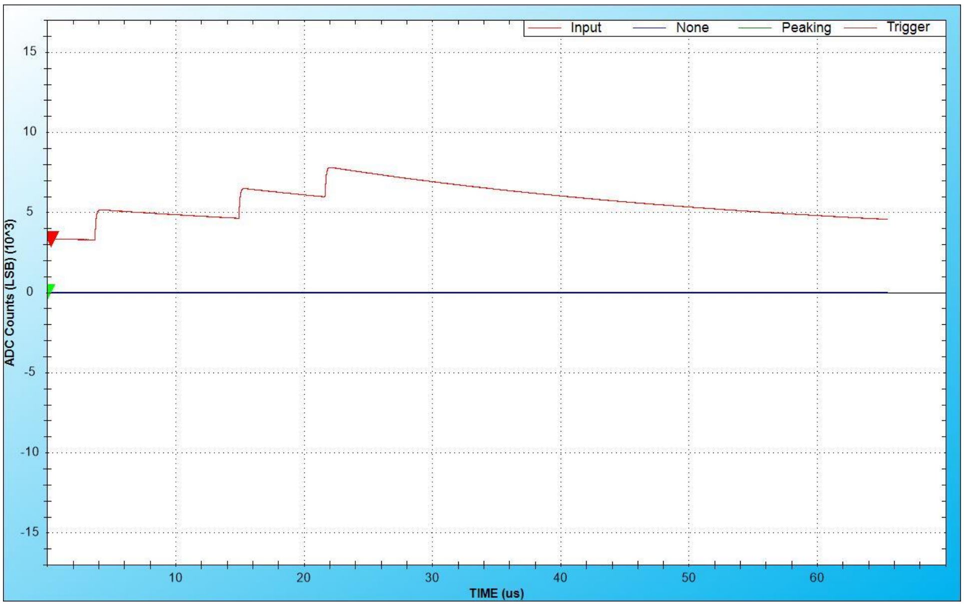
6. Check the analog output on the DT5725 with DPP-PHA firmware installed

For the particular example of this chapter we are using a DT5725 with proper DPP-PHA firmware installed. The Detector Emulator settings chosen in the previous step are consistent with a signal coming from a pre-amplifier device, compliant with the DPP-PHA firmware itself. The MC2 Analyzer Software must be properly installed in the host PC. Please refer to the DPP-PHA and MC2 Analyzer User Manual for any details about the DPP-PHA settings.

Note: The same test can be performed with any other CAEN digitizer with proper DPP or Standard firmware. Be aware that the standard firmware allows to check the waveform output only, as the DPP allows to check both the waveform and the spectrum (see the next step). In that case please set the signal parameters accordingly.

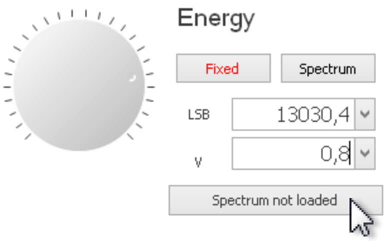
Note: The user can use an oscilloscope as well to check the waveform output. Use 50 Ω termination.

The signal inspector plot of the DT5725 should appear as in the following picture.

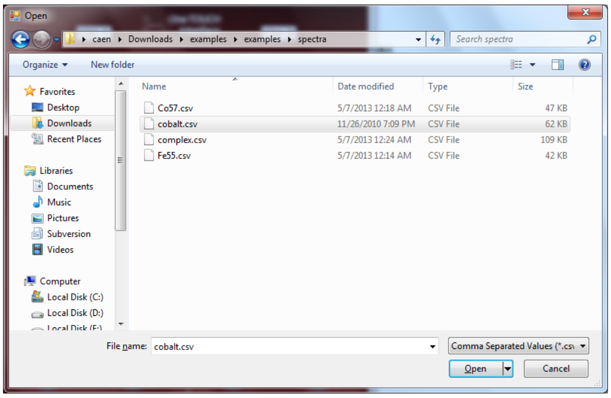

Input
None
Peaking
Trigger
ADC Counts (LSB) (10^3)
15
10
5
0
-5
-10
-15
10
20
30
40
50
60
TIME (us)

7. Enable the spectrum amplitude generation

Action 1: Switch the generation mode to Spectrum. Click *"Spectrum not loaded"* button

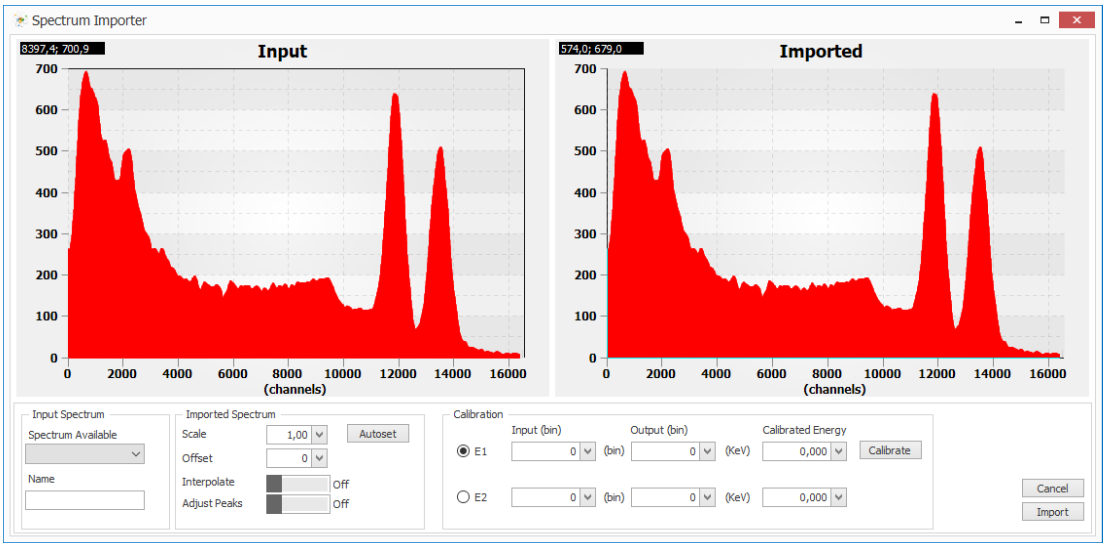


Action 2: If no spectrum is already loaded, the Import window will appear. On the other hand, if a spectrum is already loaded, emulation will start immediately.
Browse and select a **spectrum file** on your computer.
A folder with some examples can be found in the Detector Emulator Control Center installation folder. Look for the .csv files inside the "spectra" folder. Select the "cobalt.csv" file.

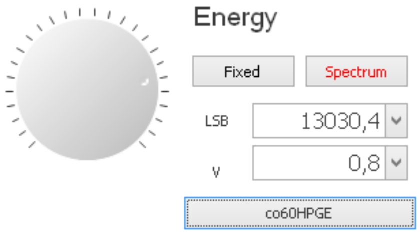


Note: accepted file formats are: .csv (comma separated values), .dat (CAEN spectrum files), .spectrum (digital detector emulator internal spectrum format), .xml (ANSI4242)

Action 3: Import the spectrum into the Detector Emulator. Click **"Import"** to import the spectrum.

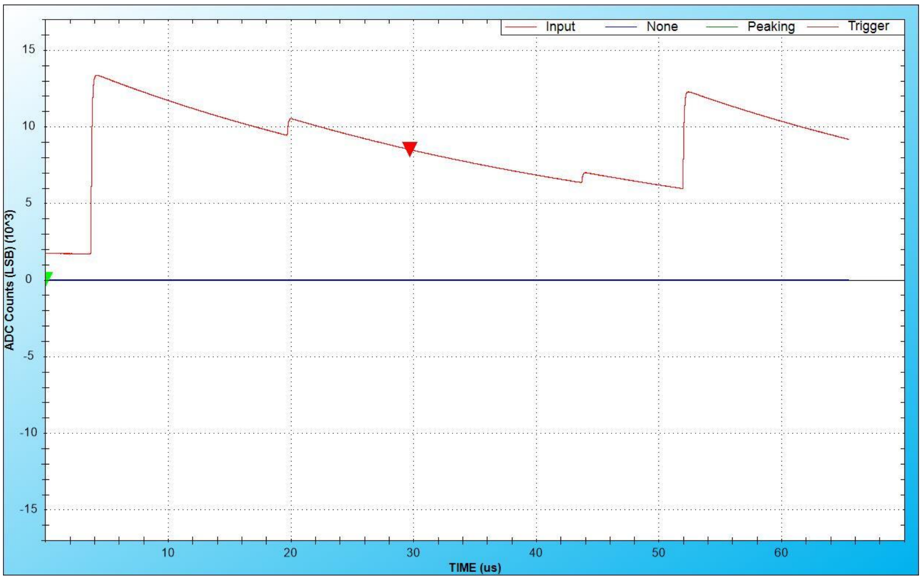


Switch to spectrum mode

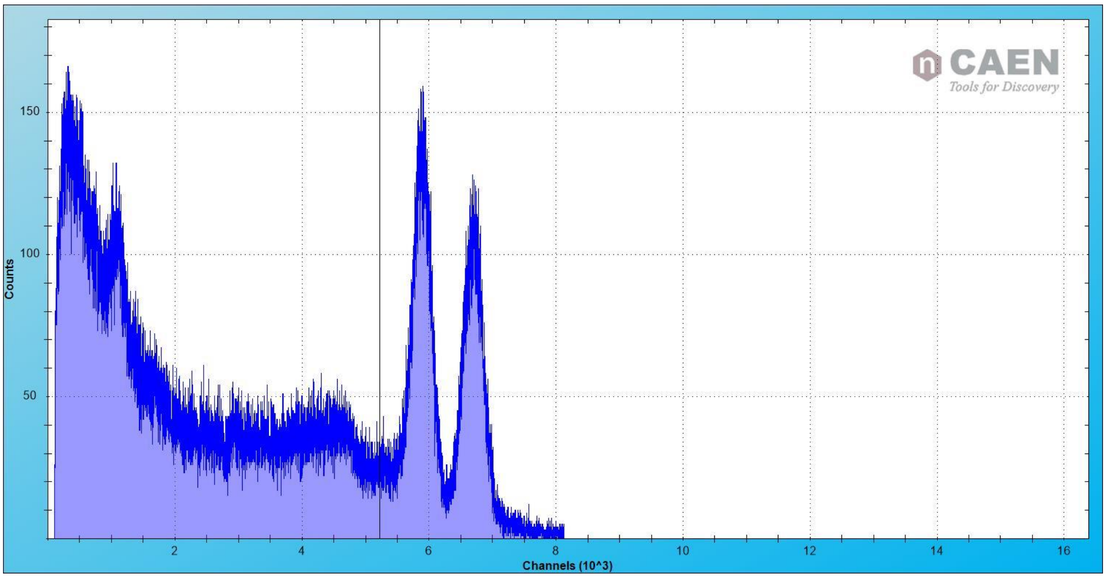


8. Check the spectrum generation with the DT5725

In the Oscilloscope mode of the MC2 Analyzer software the user should see the signal amplitude modulated with the input spectrum.



If the user switches to the Histogram acquisition mode, he/she should see something like in the following figure.



12 Software Interface

Introduction

The Digital Detector Emulator Control Center allows to manage the communication and to set the parameters for the signal generation. This chapter is intended to give to the user a complete description of all the functionalities of the emulator that can be set through the software interface.

Installation

The Digital Detector Emulator Control Center is compliant with Windows 7 SP1, 8 and 10 OS, both 32 and 64 bit.

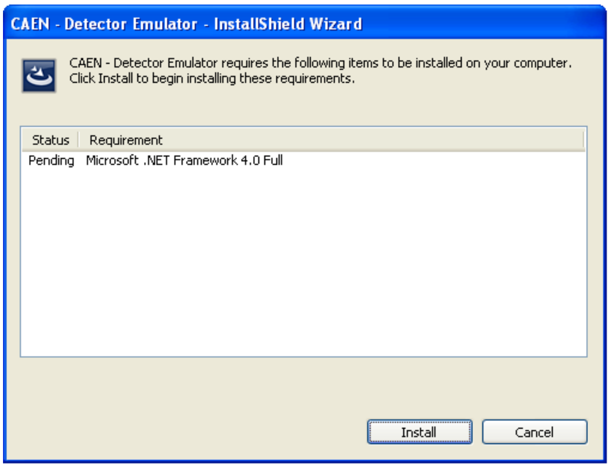
Download the standalone Digital Detector Emulator Control Center 2.5.0.0 full installation package on CAEN website in the “Software/Firmware” area of the Digital Detector Emulator page (login is required before the download).

Unpack the installation package, login as administrator, launch the setup file, and complete the Installation wizard. See Section **Software and Drivers** for further details.

The setup automatically creates a link on the PC Desktop.

Requirements

The software requires Microsoft .NET 4.0 or higher. If the framework is not available on the machine, the Detector Emulator setup will install the framework. Click on **“Install”** if this message appears:



If Microsoft .NET is already installed, the setup returns a failure message which means that it detected a previous installation. **Press yes** to continue with the installation.

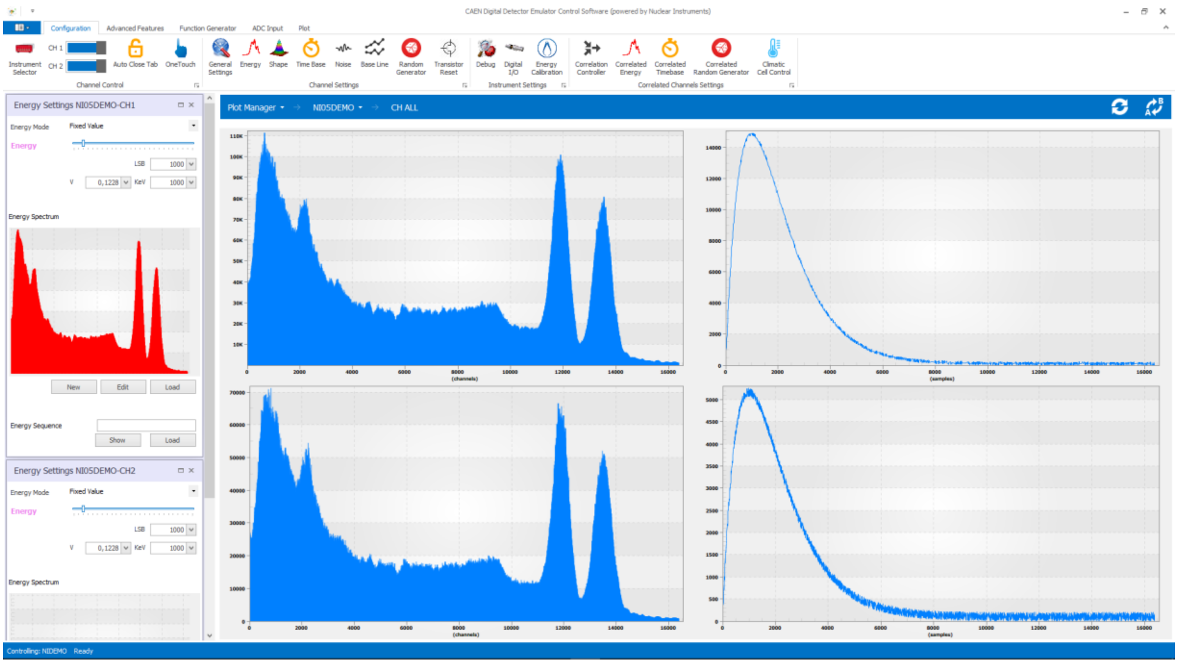
Program Execution

The user can run the program in three ways:

- The **Desktop icon** of the program
- The **Quick Launch** icon if the program
- The **.exe file** in the installation path on user's PC.

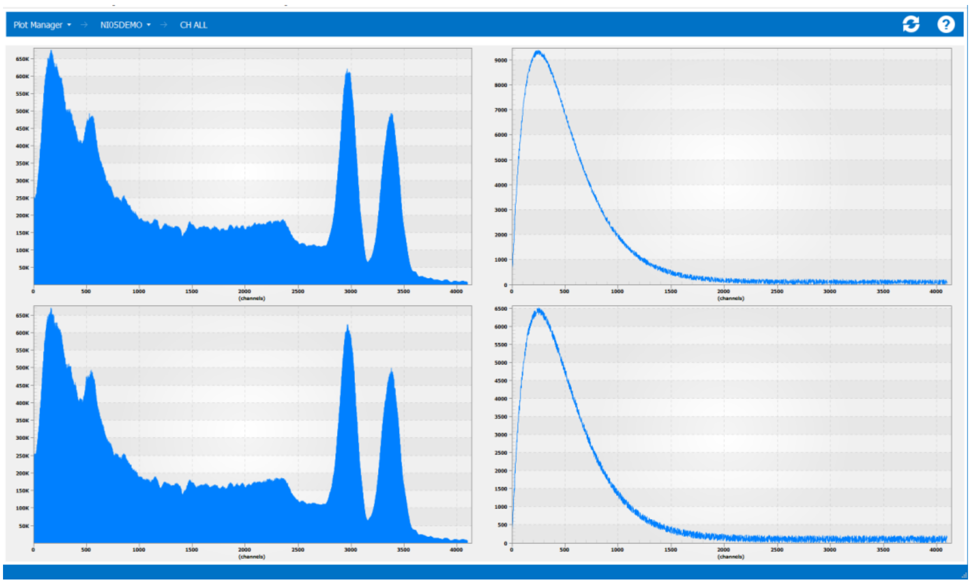
User Interface

The picture below shows the main panel of the User Interface.



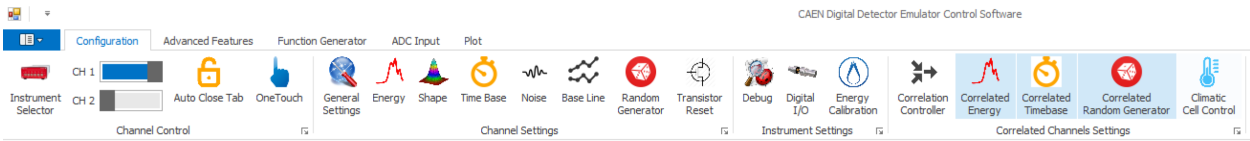
The interface is divided into 4 distinct areas:

PLOT
It is the region where are shown in real time the emulation results, meaning the spectrum and the output signal of each channel.

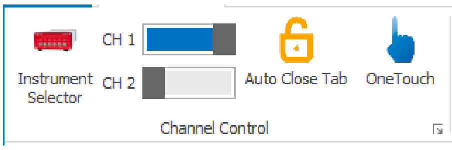


CONTROL MENU

The **Ribbon** bar contains keys and links to control the widgets of the various parameters of the instrument. It is organized into several tabs and each tab contains groups of buttons.



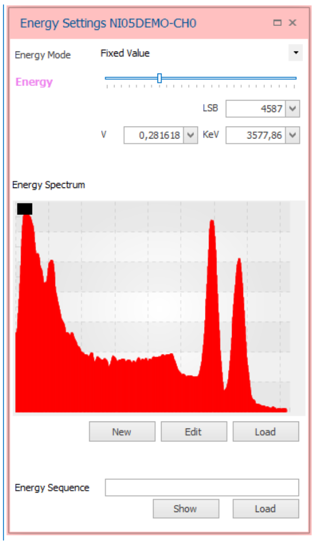
When pushing a button, the corresponding widget is loaded in the sidebar. Each widget refers to a specific function -e.g. Energy settings – and to the selected emulator and its channels (if enabled).
The following icons are fundamental to understand the correct use of the control software.



The key **Instrument Selector** allows the user to access the tools menu. With this button it is possible to choose which instrument to check among the ones connected to the user's PC.
Once the user chooses the tool he/she can select which channel/s to control. Clicking on the switch **CH1 / CH2** the user can enable the channel/s to control. For example, if the user activates only switch CH1 he/she will only be able to control the channel 1 and each time the user clicks on an icon of the Ribbon bar, a single control widget related to the CH1 will be opened. If the user selects both CH1 and CH2 by pressing a control button, two widgets will be opened -one for each channel.
The **Auto Close Tab** button allows the user to switch between the two modes of widget management. The first (default), passing from one control category to the other, closes the widgets previously loaded. The second one allows the user to keep the widgets open so as to customize the user's operating desktop.
The **OneTouch** button opens the One Touch interface that will be described later.

WIDGET AREA

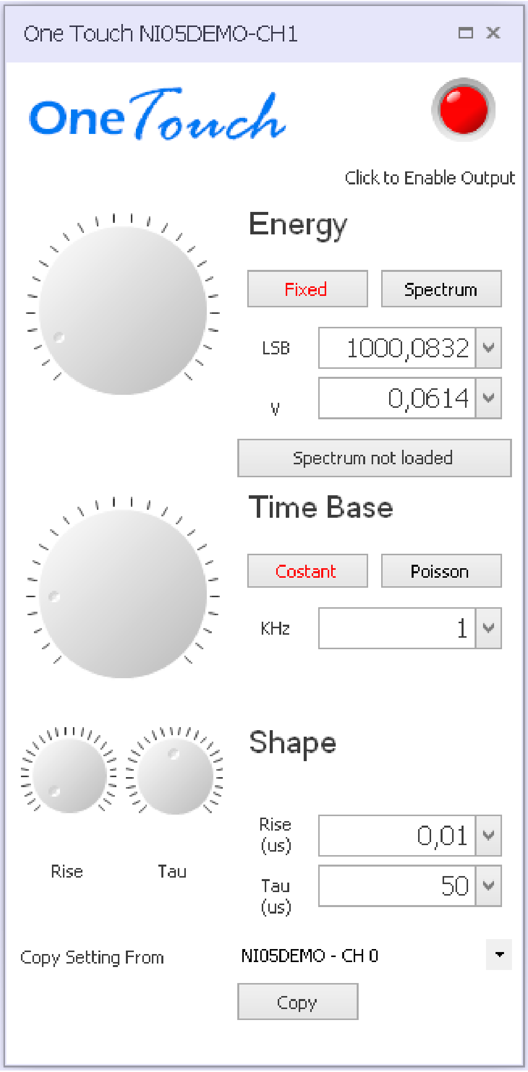
This column contains all widgets opened through the control menu. Each widget allows the user to control a specific set of functions of a specific emulator.



Each widget has a title bar that allows the user to recognize the emulator and the channel which it relates to. Widgets can be enlarged or closed using the buttons on the title bar. The user can also freely rearrange the layout of the widget via drag and drop.

OneTouch Interface

The OneTouch interface is particularly indicated for quickly setting up the instrument. Customers who are used to deal with a classical pulser may find this interface quite familiar. The OneTouch Interface enables the generation of analog output according to the method described in **Digital RC Section**. The user can set the analog output signal rate, the amplitude, the rise and fall time of the exponential shape.



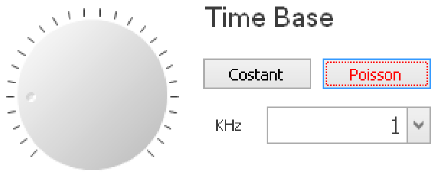
The OneTouch interface is opened by clicking the OneTouch button in the MAIN form of the Ribbon menu. Unlike all the other control buttons, the OneTouch interface opens a number of widgets equal to the number of channels, independently from the position of the channel selector switches. In this way the user can have control over all channels at a glance.

The main parameters for the signal generation can be set through the virtual knob and the underlying boxes.

Frequency

The user can choose between two options: “constant rate” or “Poisson distribution” (random).

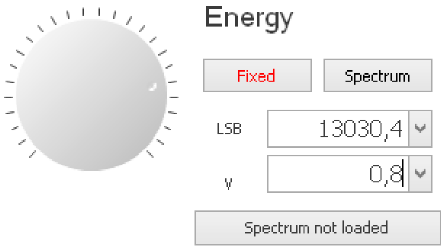
It is possible to set the rate value (in KHz) either moving the knob pointer or by writing the value in the underlying box.



Amplitude

It is possible to choose between a “Fixed value” of amplitude or to load a file with the spectrum samples.

The amplitude value can be set by moving the knob pointer, by writing the value in the underlying box or by writing the desired value in Volts. For details about the available file formats please refer to the **Accepted File Formats Section**.

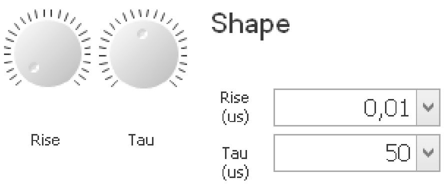


Note: After typing the value of a parameter in a box menu, it will be automatically applied.

When the Spectrum option is enabled, the software opens a folder that allows the user to select a file. The spectrum name is then shown in the grey box. Users can load a new spectrum by clicking on the same box.

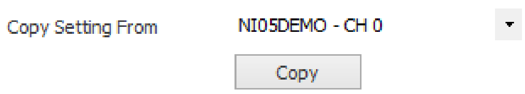
Shape

The OneTouch interface allows to generate exponential shapes only. The user can set the “rise time” and the “decay time” in µs. Values can be changed by either moving the knob pointer or by writing the value in the underlying box.



Note: If the user sets the Rise Time = 0.00 he/she is asking for the best the device can provide, i.e. about 1 ns (for fast front-end).

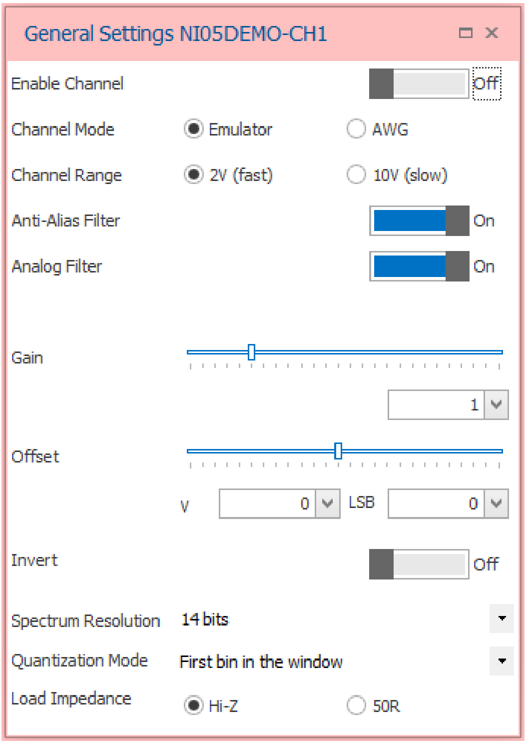
The user can copy the options from another emulator or even from another channel by choosing the source of the copy from the scrolling list.



Setting Area

Use the Ribbon Bar to load the corresponding widget to the set of parameters on which to intervene.

Channel Global Settings



Enable Channel: if selected, the output is activated. Otherwise the output is fixed to zero with low impedance. Output Enable is equivalent to the Main button.

Channel Mode: the channel operates as "Emulator/Pulser" or "Waveform Generator" (AWG).

Channel Range: it selects between high speed (2V) and high voltage (8V) output front-end.

Anti-alias filter: does not apply to the DT5810 Fast Digital Detector Emulator (disabled).

Analog filter: it inserts an LC 7th order filter (Analog Bessel Filter) on the output. It can reduce the output noise by a factor of 5 and it can improve the signal shape. Rise time increases from 1ns to 25/30ns.

Digital Gain: digital amplification of the output signal between 0 and 2. It is applied at the end of the processing chain, therefore the whole output is multiplied by the digital gain, including the offset.

Digital Offset: it refers to the offset expressed as a fraction of 2^{16} levels of quantization -i.e. between -2.2 V and +2.2 V. The offset is applied before the gain stage.

Invert: if selected, the polarity of the output is inverted.

Spectrum resolution and spectrum quantization: the re-quantization of a spectrum over a different number of bins can introduce distortions in its shape due ultimately to an aliasing effect. The spectrum emulated should have at least the same bin resolution as the device that reconstructs it from the emulated pulses.

As a rule of thumb, the reference spectrum in the emulator has resolution above the request one. In order to get the matching of bin resolution, a decimation process is implemented in the instrument with three possible processing procedures:

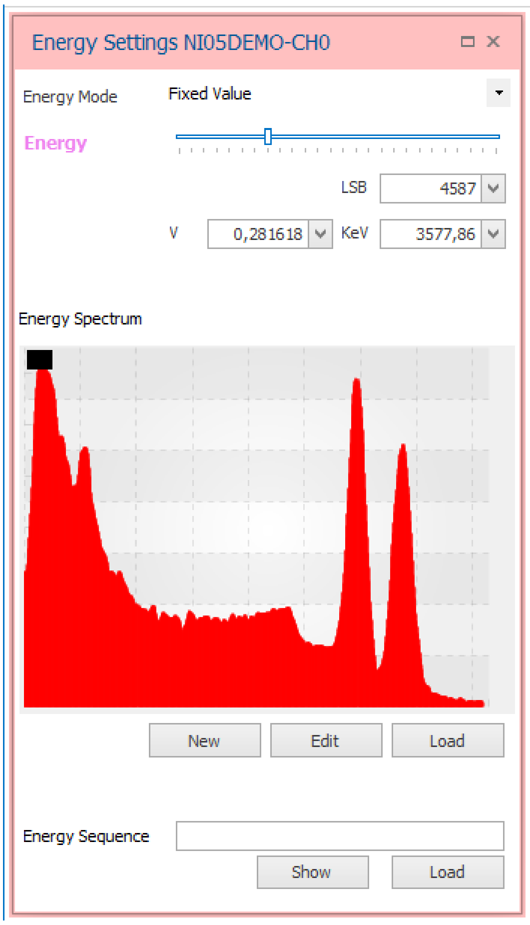
"First bin in the window": the spectrum is automatically divided into bins and only the first value of each bin is considered;

"Max in the window": the spectrum is divided in bins and only the highest value of each bin is considered;

"Slice (first part of the spectrum)": only the lower values of the spectrum are considered, up to a number of bins equal to the target resolution.

Load Impedance: allows the user to select the output impedance of the channel, 50 Ω or High-Z

Channel Energy Settings



There are three possible emulation modes:

- **Pulser**, for the generation of pulses with fixed amplitude and time distributions;
- **RTG – Random Tail Generator**, for the generation of pulses with fixed amplitude and pseudo-random statistical time distribution;
- **Emulator of Radiation Sources**, for the generation of pulses statistically distributed whose spectrum corresponds to a given starting spectrum.

Pulser – Select the flag "Fixed value" and adjust the slider position to regulate the amplitude of the pulses at the output. Energy value can be set in three ways: LSB, Volt and KeV (if the instrument has been previously calibrated in KeV).
In the panel box "Time Distribution" set a constant value of rate, from 10^{-2} cps to 11 Mcps

RTG – Random Tail Generator – Select the flag "Fixed value" and adjust the slider position to regulate the amplitude of the pulses at the output. Energy value can be set in three ways: LSB, Volt and KeV (if the instrument has been previously calibrated in KeV).
In the panel box "Time Distribution" select the flag "Poisson Distribution".

Emulator of Radiation Sources – Select the flag "Spectrum Emulation" in the Energy Mode box. Then select "Load File" to load a custom reference energy spectrum.

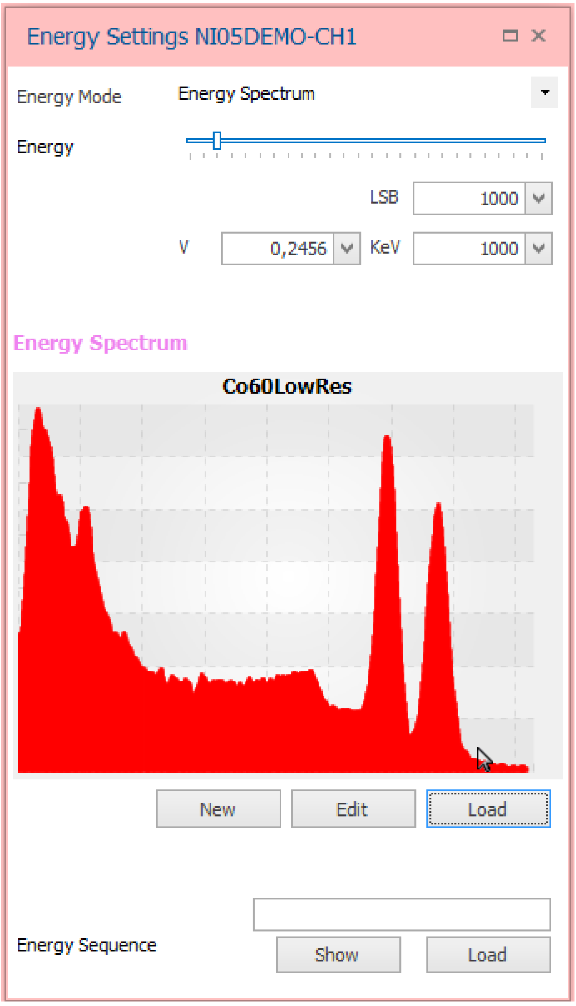
Energy Mode – It allows you to choose the emulation energy mode between: "Fixed Amplitude" (Pulser/RTG Mode), "Spectrum emulation" and "Sequences".

Fixed Mode

When the instrument is in Fixed Mode, use the Energy bar or the input box in order to set the amplitude of the signal to be emulated.

Spectrum emulation

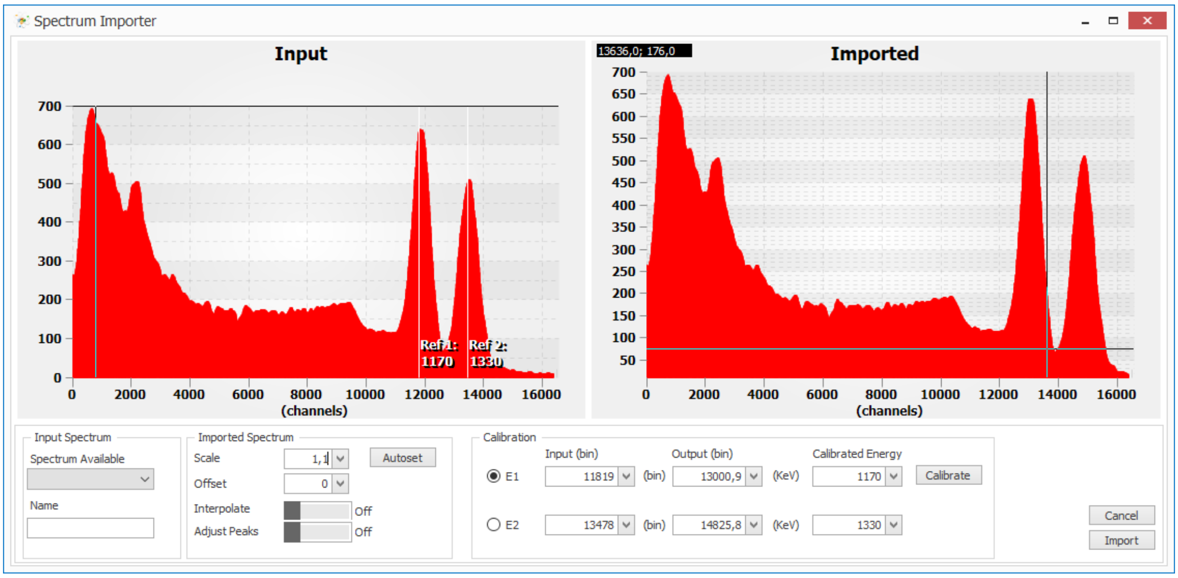
There are four file formats accepted: .csv file, ANSI N4242, .dat (Caen Digitizer Output Format), .spectrum (internal file format). See **Section Accepted File Formats** for more details.



When the file is loaded a spectrum preview is shown in the widget.
This does not mean that the spectrum is active on the instrument. Ensure Energy Mode is set on "spectrum" in order to actually generate the desired spectrum.

Import an Energy Spectrum from File

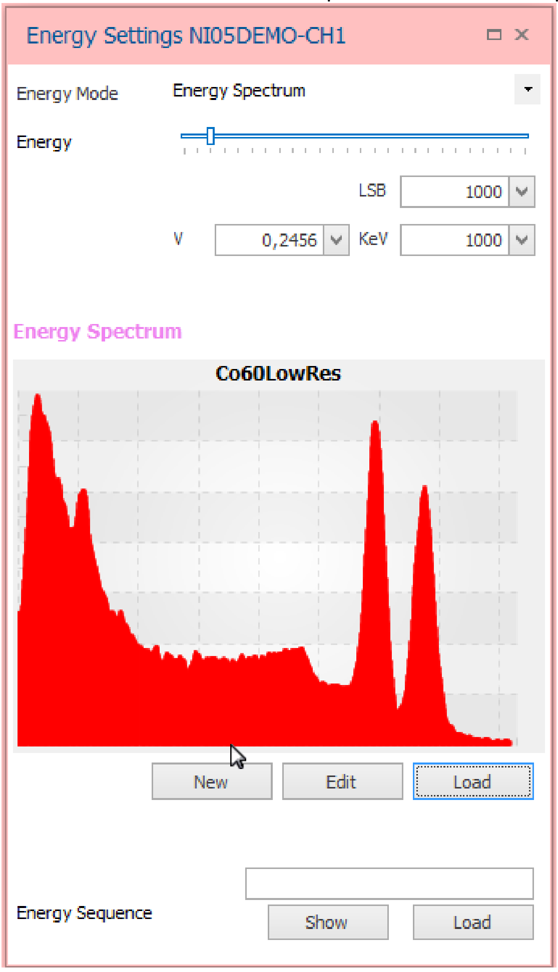
After selecting the spectrum file to be imported, the GUI opens the "Spectrum Import Tool" which allows the user to upload the spectrum, to make preliminary operations on the spectrum itself and to make the calibration.



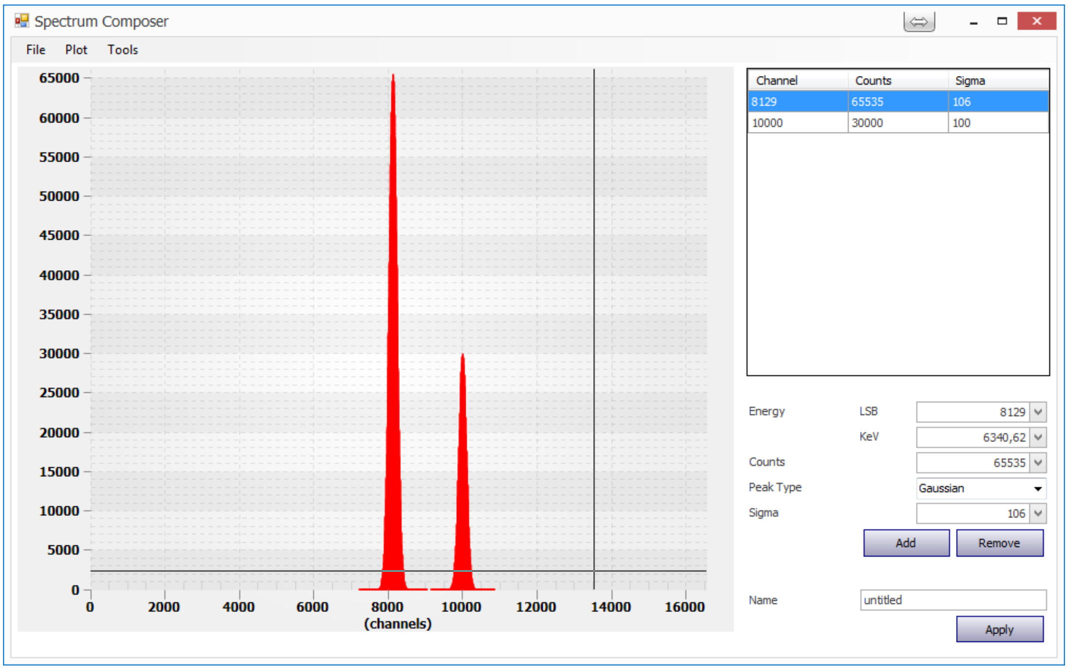
The **"Input Spectrum"** is the spectrum as read by the selected file.
The **"Imported Spectrum"** is the spectrum as imported by the device. Some operations can be made before importing and this plot allows to see in real-time the effect of those operations.
"Spectrum Available" works for multiple N4242 spectra -where it is possible to select the desired spectrum.
Under the **"Imported Spectrum"** box it is possible to set a **"Scale factor"** and an **"Offset"** values. The first allows to rebin the x-axis of the spectrum. The second allows to set an offset of the x-axis (in number of channels).
The tool allows the user to adjust the dynamic of the input spectrum to the resolution of 16384 bins of the instrument.
By checking the **"Interpolate"** button, the scaling is performed by means of a linear interpolation, otherwise each new bin is created on the basis of the nearest value of the bin on the left. The function of **"Adj Peak"** sets the highest peak of the input spectrum at the value 65535.
It is also possible to calibrate the instrument through the box **"Calibration"**. Press "Calibrate" when ready.

How to generate an Energy Spectrum with the internal tool

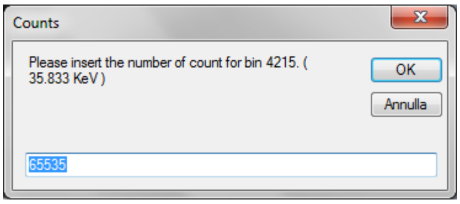
The instrument allows to generate a custom spectrum by adding specific lines and setting their width. It is also possible to select the radioactive lines of specific elements of the periodic table.



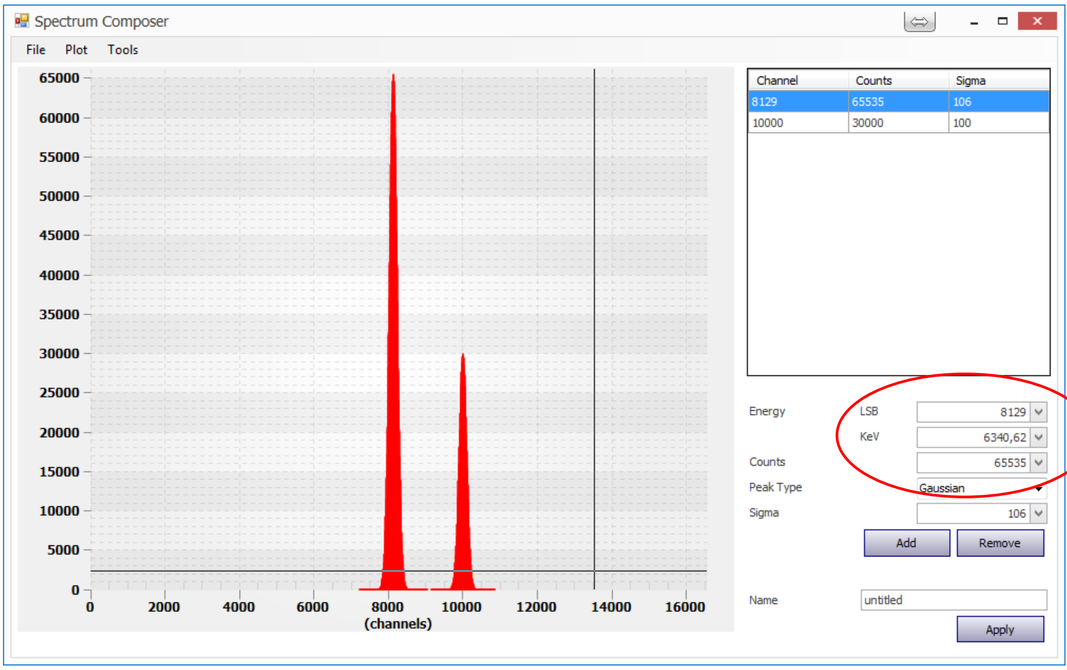
Select "New" to open the **Spectrum Composer** window.



Point the cursor on the diagram and double click to add a line. The x-axis is in counts unit.
In the pop-up window write the corresponding value of height (65535 is the maximum value allowed).



Alternatively, write the value of KeV and LSB in the corresponding boxes on the right part of the window.



Select the **"Peak type"** between **"Gaussian"** and **"Rectangular"**. The **"Sigma"** value corresponds to the classical sigma of the gaussian function in the first case and to the width of the rectangular in the second. It is espressed in counts unit.

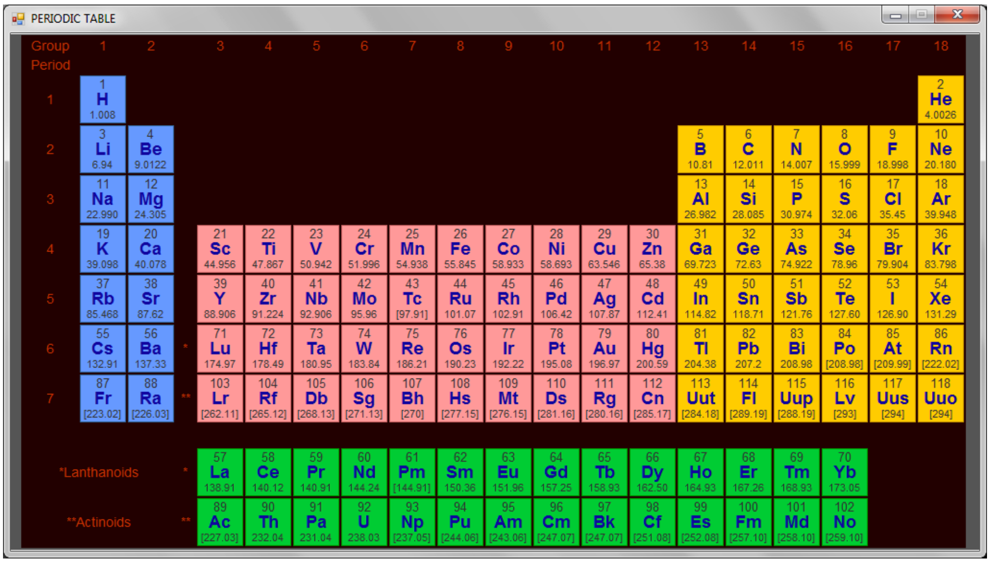
The button **Add** allows to add more line as the button **Remove** to delete them.
The tool allows to **"Save"** the spectrum in its internal file format .spectrum. Specify the file **"Name"** before saving. Use the menu **File** to **Save and Load** a custom spectrum. A browser is then open to select the destination folder.

The spectrum is represented on a scale with maximum resolution of 14 bits while the resolution of the Emulator is 16 bits. A scale factor is present to adjust the spectrum resolution to the instrument range, e.g. Scale x2 for spectrum with resolution of 14 bits.

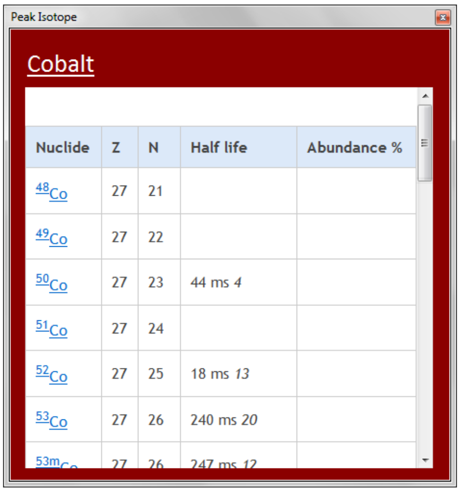
Generate an Energy Spectrum with Isotopes Database

From the **Spectrum Composer** select menu Tools -> Isotopes DATABASE

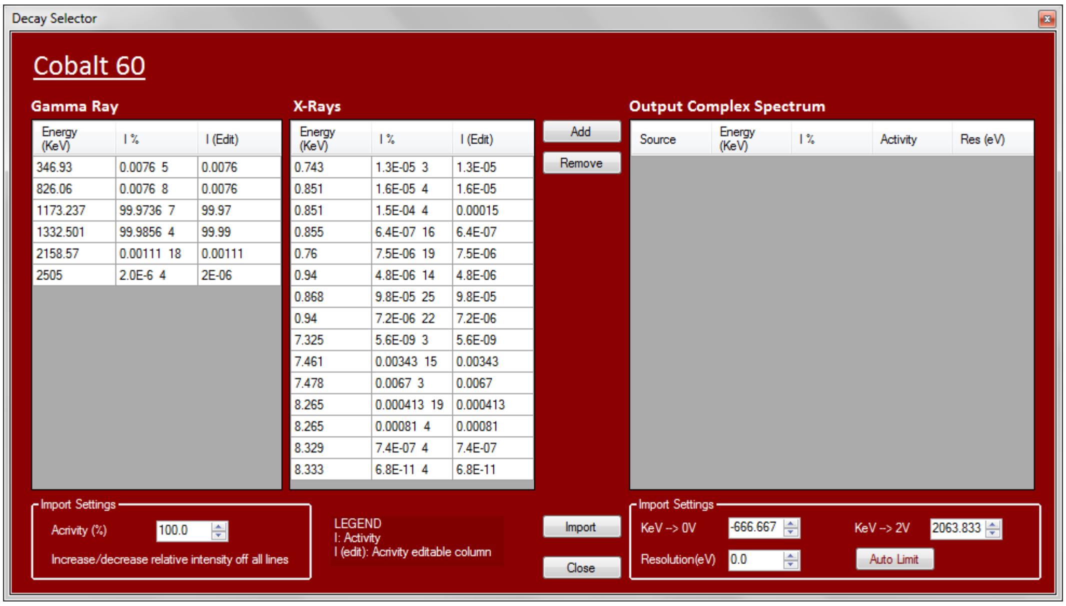
The list of isotopes will then appear.



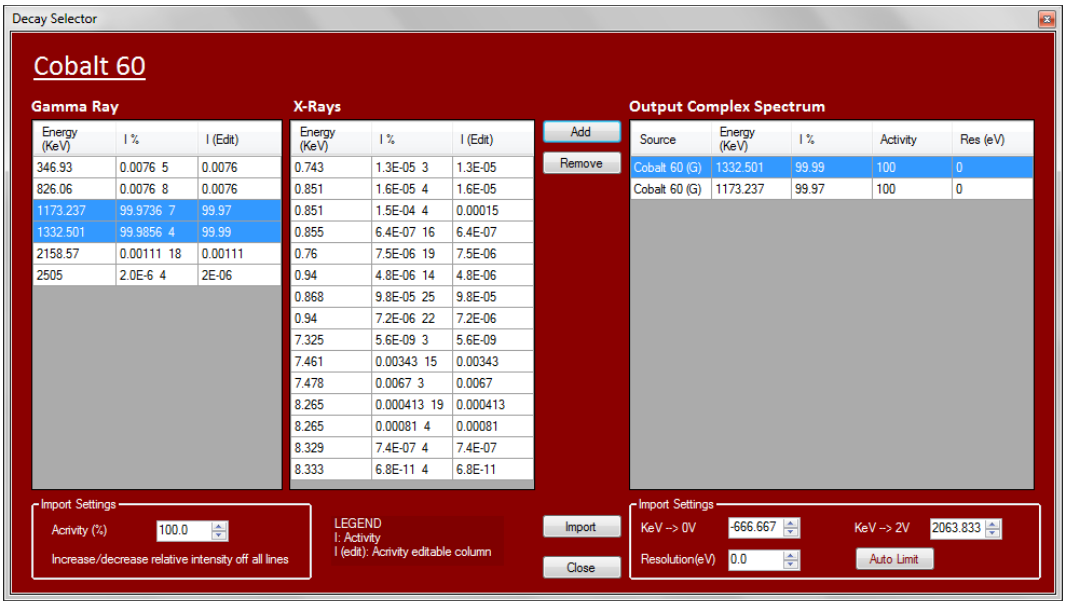
Select from the periodic table one of the element you want to add into the spectrum, as for example the Cobalt (Co).

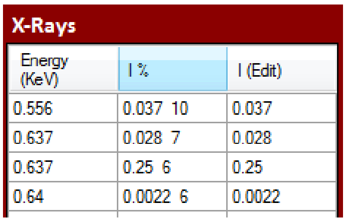


Scroll the bar up to the radioactive 60Co and click on the name to open the “Decay Selector” Window.

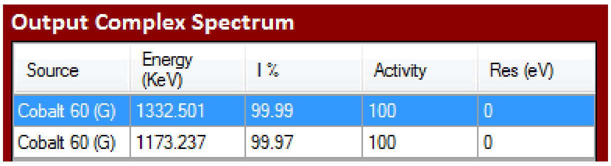


The Decay Selector shows on the first column the list of 60Co emitted "**Gamma Ray**" and on the second one the list of "**X-Ray**" lines. The user can select the desired lines and add them to the output spectrum. For example select the 1173 and 1332 KeV Gamma Ray lines and click "**Add**". Click "**Remove**" to remove undesired lines from the output spectrum.





The column **I** represents the relative activity of the line of the isotope. The column "**I (Edit)"** allows the user to edit the activity (in count per second) of the selected line.



Output Complex Spectrum

Source	Energy (KeV)	I %	Activity	Res (eV)
Cobalt 60 (G)	1332.501	99.99	100	0
Cobalt 60 (G)	1173.237	99.97	100	0

Columns **Activity** and **Res (eV)** can be edited also from the "**Output Complex Spectrum**" tab.

The user must take care that the selected lines are in the calibrated range of energy of the Detector Emulator. The energy calibration is performed by setting the energy in KeV unit corresponding to 0 V, the energy corresponding to the full scale of 2 V to fit the dynamic of the output spectrum. Changing those values will automatically re-calibrate the device.

Press "**Auto limit**" to find the best calibration for the designed spectrum.

Set the desired Gaussian resolution in eV unit for all lines of the spectrum. A resolution equal to 0 eV corresponds to mono-energetic lines.

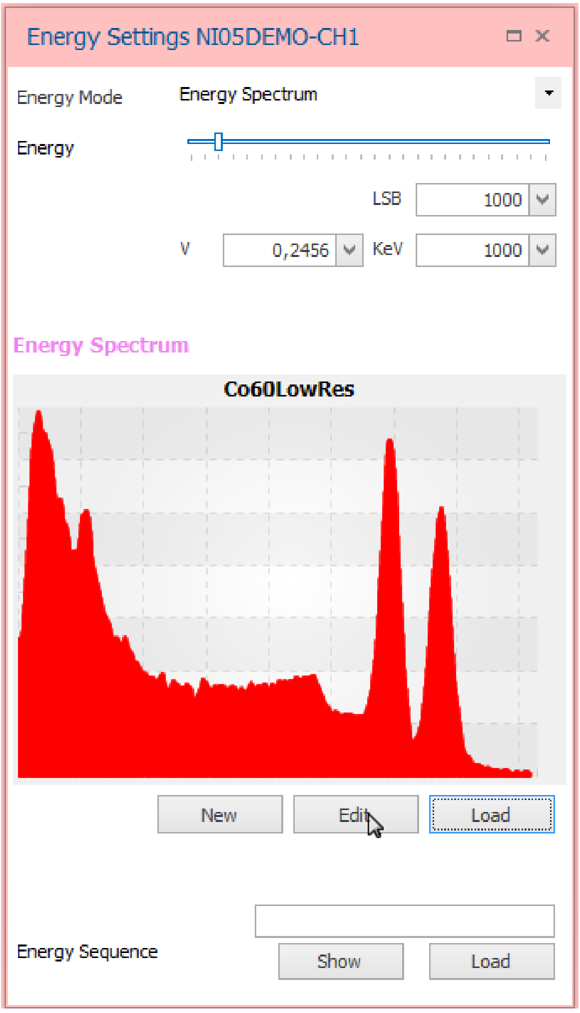
Note: The resolution can be set both from the "**Decay Selector**" and the "**Spectrum Editor**" windows.

The "**Import Settings**" box allows the user to select the relative activity of the current isotopes when a complex mixture of isotopes is selected. The activity value acts to all the lines of the current isotope.

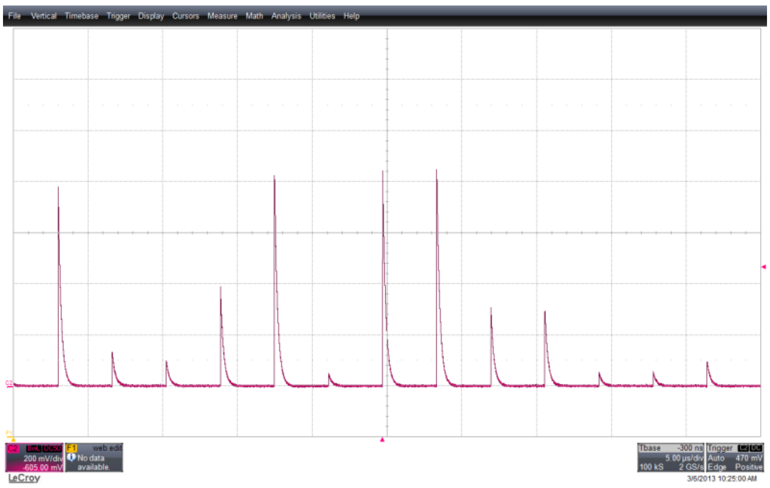
Click "**Import**" to import the spectrum with the desired settings, or "**Cancel**" to cancel it.

How to modify a spectrum

Select “**Edit**” to open the “**Spectrum Importer**”. If the spectrum has been previously loaded from file, the “**Spectrum Import Tool**” will appear; otherwise the “**Spectrum Editor**” allows to modify the custom spectrum.

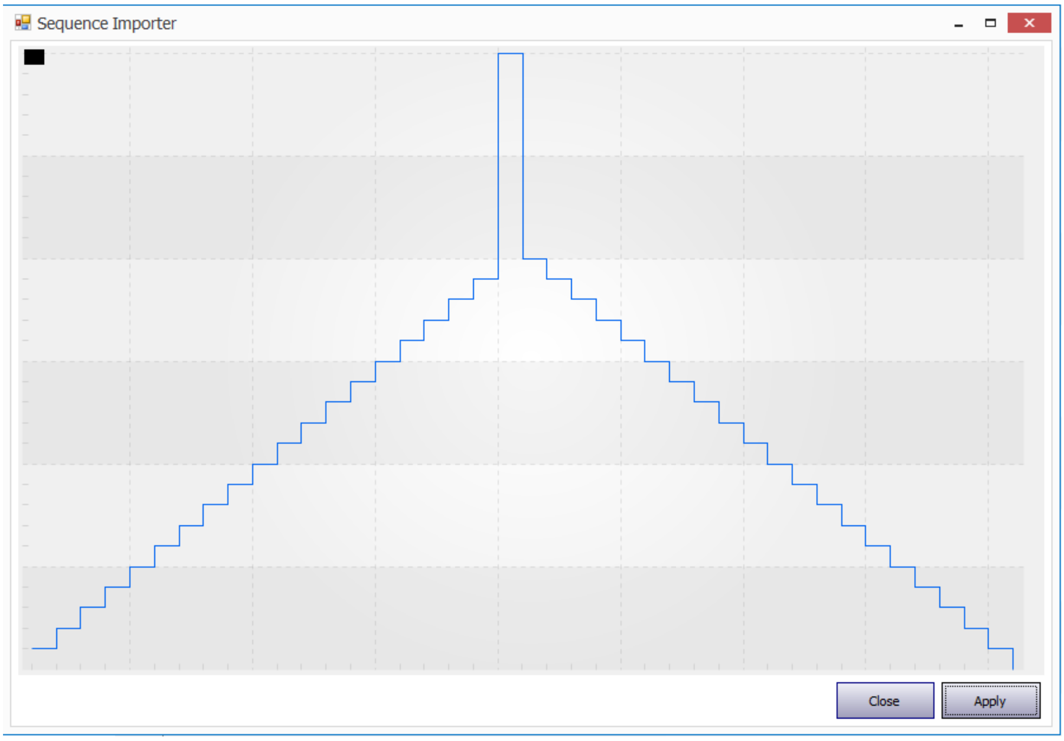


The picture below shows a screenshot of the output of the emulator initialized as in the example above with the spectrum of the isotope ^{60}Co.



Sequence

The "**Sequence**" option allows the user to load the amplitudes of an a-priori defined sequence of pulses. The same option is in the section "**Time Distribution**" and allows the user to load the temporal distances between the pulses.



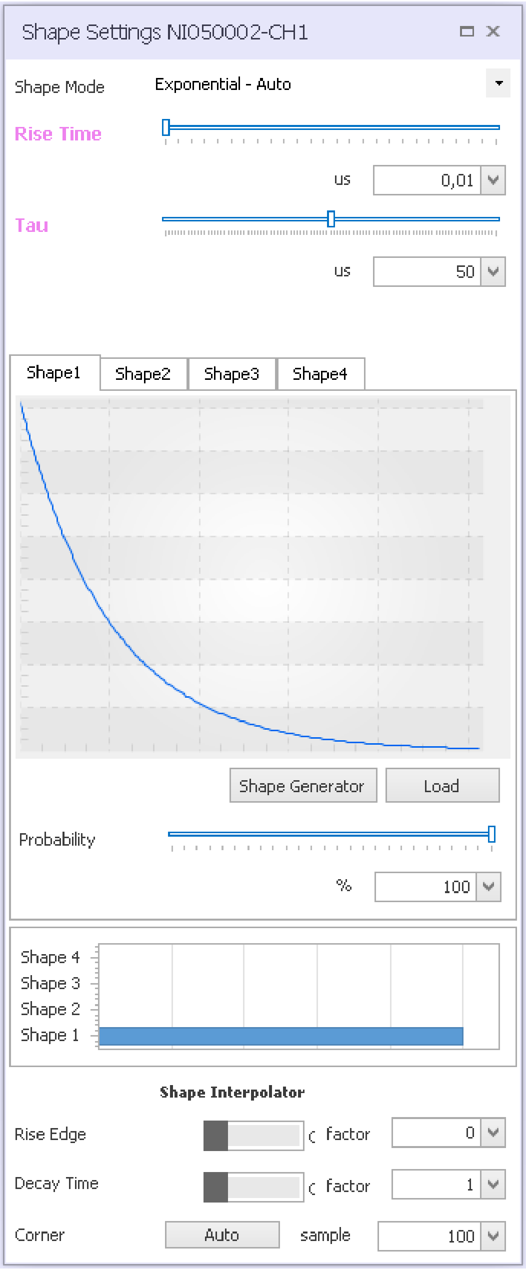
The "**Sequence**" option allows the user to emulate repetitively the stored sequence made up to 10 kpulses. Also a partially defined sequence can be stored. The following combinations are available:

- Load a sequence of amplitudes with constant or Poisson statistical rate;
- Load a sequence of temporal distances and constant (or statistically generated) amplitudes, as described above.

All features of the emulation process (noise, signal shape, …) are active in this option, except the generation of amplitudes from a spectrum. The picture above shows a screenshot of stored amplitude sequence

Signal Shape

The emulator is able to generate both exponential and arbitrary shape signals. The exponential settings are located in the top part of the Widget; the custom shape settings are on the bottom part.



Shape Mode:

- **Auto**: selects automatically between Fast and DRC
- **Fast**: Exponential memory based
- **Digital RC**: Exponential mode based on a digital emulation of a real continuous reset charge preamplifier
- **Custom Shape**: memory based user defined shape
- **Multishape**: up to 4 different shapes on the same channel

There are three possible options for the *exponential shape*:

- **Auto**: the system is initialized in the "Auto" operation mode and automatically selects between the two options **"Fast (Limited Pileup)"** and **"Digital (RC)"** mode depending on the values of time constant and rise time programmed by the user (if fall time is smaller than 100ns the FAST exponential is automatically selected).
- **Fast**: the "Fast" mode uses 16 memories to generate the exponential signal. After setting the time constants of rise and fall of the signal, the system automatically implements the shape and activates the interpolators (if

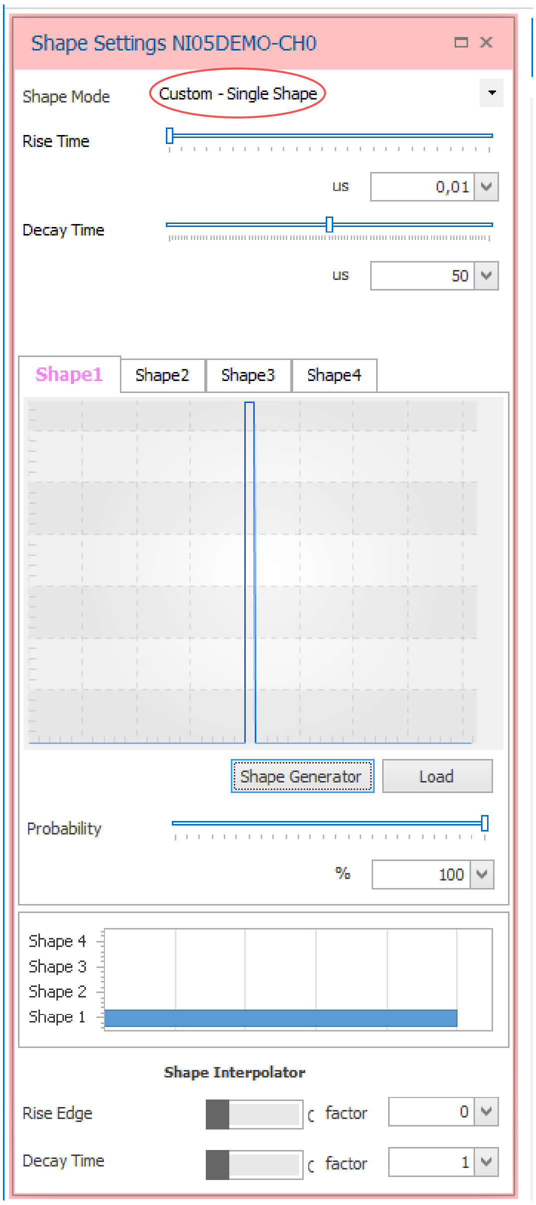
necessary) choosing the most suitable interpolation factor. Refer to **Section Custom Shape – Memory Based data-path** for more details about the method implementation.

- **Digital (RC)**: the exponential signal is generated through a cascade of two IIR filters. The first one sets the "Fall Time" of the exponential shape as the second one adjusts the rise time. For the analogy with an analog pulser this operation mode is referred as "Digital (RC)". The maximum number of piled-up events is limited only by the saturation of the analog output stage and the Poisson statistical distribution of the events is guaranteed at any rate (refer to **Section Digital RC** for more details).

Rise-time and **fall-time** can be adjusted by the user.

Note: The block "Digital (RC)" operates at a frequency equal to half the system clock and it is therefore not able to generate correct signals with fall time and/or rise time shorter than 16 ns.

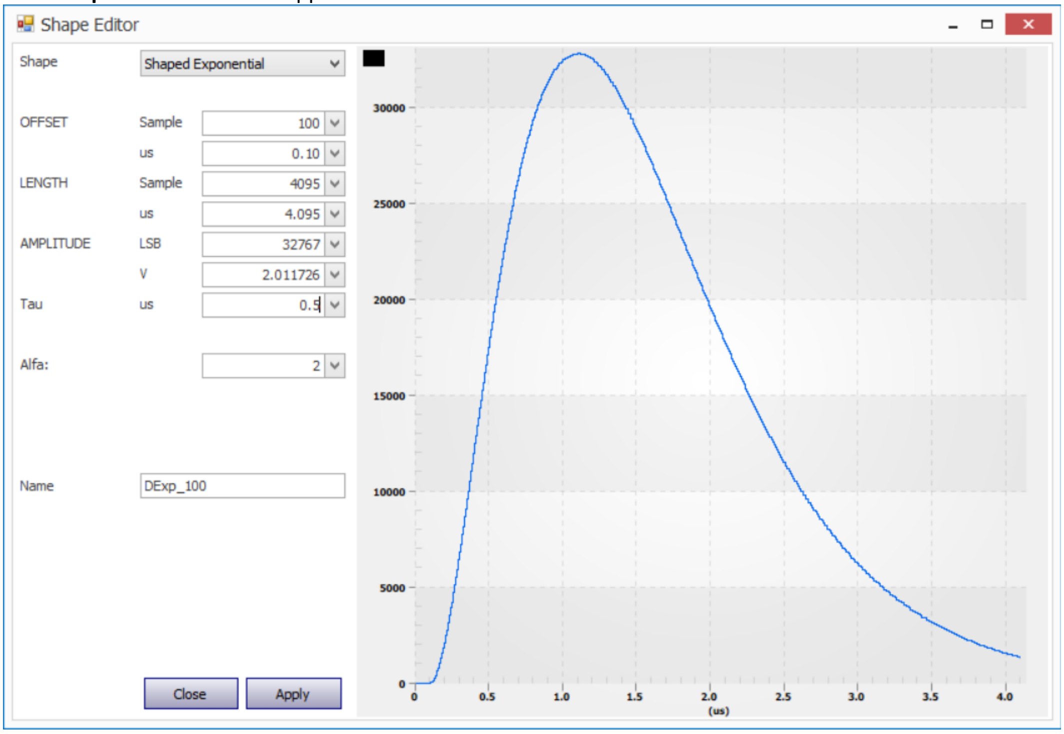
The generation of signals of ***arbitrary shape*** is realized by means of 16 memories of 4096 points each. The method is the same of the Fast exponential shape. Signals longer than 4096 points can be obtained with a linear interpolator available at the output of each memory. If piled-up events exceed the maximum number of 16, the statistical distribution of the output events is not Poissonian.
The output signal of the emulator can be loaded from a file o generated by an integrated shape design tool. Click on "**Custom**" in the Shape Mode list to enable the memory-based shape generation.



The shape can be arbitrarily defined over a maximum of 4096 samples. A set of available shapes can be created and the actual shape of the output signal chosen from time to time.

Use **"Load"** to load a custom .csv file that can be the result of a measurement or a Matlab© synthesis.
The tool is able also to generate some pre-defined shapes. Click on **"Shape Generator"** to internally generate a shape. Click **again on the shape generator** to edit the generated shape.

The **"Shape Preset"** window appears as follows.



On the left the user can select one of the pre-defined functions, choosing among "Delta", "Pulse", "Exponential", "Shaped Exponential", "Double Exponential" and "Gaussian". The characteristic parameters can be adjusted either in samples (LSB), or Volt or us when needed.
The user can decide a "**Name**" for the custom shape and click on **Apply** to import the shape. Left click and drag on the plot to zoom in. Right click and select "**Undo Zoom**" to zoom out.
In the following the list of available shapes from the device.

Delta

The formula used for the implementation is:

$$A * \begin{cases} 1 & t = OFFSET \\ 0 & otherwise \end{cases}$$

Where A is the AMPLITUDE value and OFFSET is the position of the Delta function.
The user can modify the **OFFSET** value, the **LENGTH** of the window, and the **AMPLITUDE** – both in samples and in us – by modifying the corresponding box value.

Note: The user can set the maximum value of amplitude and then rescale it on the main GUI.

Pulse

The formula used for the implementation is:

$$A * \begin{cases} 1 & OFFSET \leq t \leq OFFSET + WIDTH \\ 0 & otherwise \end{cases}$$

Where A is the AMPLITUDE value, OFFSET is the starting position of the Pulse function and WIDTH is the width of the Pulse function.
The user can modify the **OFFSET** value, the **WIDTH** of the Pulse, the **LENGTH** of the window, the **AMPLITUDE**, the **RISE SLOPE** and **FALL SLOPE** – both in samples and in us or Volt – by modifying the corresponding box value.

Note: The user can set the maximum value of amplitude and then rescale it on the main GUI.

Note: The Rise and Fall slope are limited by the DAC frequency (i.e. minimum value 1 ns).

Exponential

The formula used for the implementation is:

$$A * \begin{cases} e^{\frac{-(t-OFFSET)}{\tau}} & t \geq OFFSET \\ 0 & otherwise \end{cases}$$

Where A is the AMPLITUDE value, OFFSET is the starting position of the Exponential function and τ (Tau) is the decay time of the Exponential function.
The user can modify the **OFFSET** value, the **LENGTH** of the window, the **AMPLITUDE** and the **Tau** – both in samples and in us – by modifying the corresponding box value.

Note: The user can set the maximum value of amplitude and then rescale it on the main GUI.

Note: The Tau is limited by the DAC frequency (i.e. minimum value 1 ns).

Shaped Exponential

The formula used for the implementation is:

$$A * \begin{cases} \left(1 - e^{\frac{-(t-OFFSET)}{\tau 1}}\right) * e^{\frac{-(t-OFFSET)}{\tau 2}} & t \geq OFFSET \\ 0 & otherwise \end{cases}$$

where A is the AMPLITUDE value, OFFSET is the starting position of the Shaped Exponential function, τ_1 (Tau 1) is the characteristic time used for the rise region and τ_2 (Tau 2) is the characteristic time used for the falling region of the Shaped Exponential function.
The user can modify the OFFSET value, the LENGTH of the window, the **AMPLITUDE** and the **Tau 1** and **Tau 2** values – both in samples and in us – by modifying the corresponding box value.

Note: The user can set the maximum value of amplitude, and then rescale it on the main GUI.

Note: The Tau is limited by the DAC frequency (i.e. minimum value 1 ns).

Double Exponential

The formula used for the implementation is:

$$A * \begin{cases} e^{\frac{-(t-OFFSET)}{\tau_1}} * e^{\frac{-(t-OFFSET)}{\tau_2}} & t \geq OFFSET \\ 0 & otherwise \end{cases}$$

where A is the AMPLITUDE value, OFFSET is the starting position of the Double Exponential function, τ_1 (Tau 1) is the characteristic time used for the rise region and τ_2 (Tau 2) is the characteristic time used for the falling region of the Double Exponential function.
The user can modify the OFFSET value, the **LENGTH** of the window, the **AMPLITUDE**, and the **Tau 1** and **Tau 2** values – both in samples and in us – by modifying the corresponding box value.

Note: The user can set the maximum value of amplitude and then rescale it on the main GUI.

Note: The Tau is limited by the DAC frequency (i.e. minimum value 1 ns).

Gaussian

The formula used for the implementation is:

$$A * \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(t-MEAN)^2}{2\sigma}}$$

where A is the AMPLITUDE value and MEAN and σ (Sigma) are the characteristic mean and sigma of the Gaussian function.
The user can modify the **MEAN** value, the **LENGTH** of the window, the **AMPLITUDE** and the **Sigma** values – both in samples and in us – by modifying the corresponding box value.

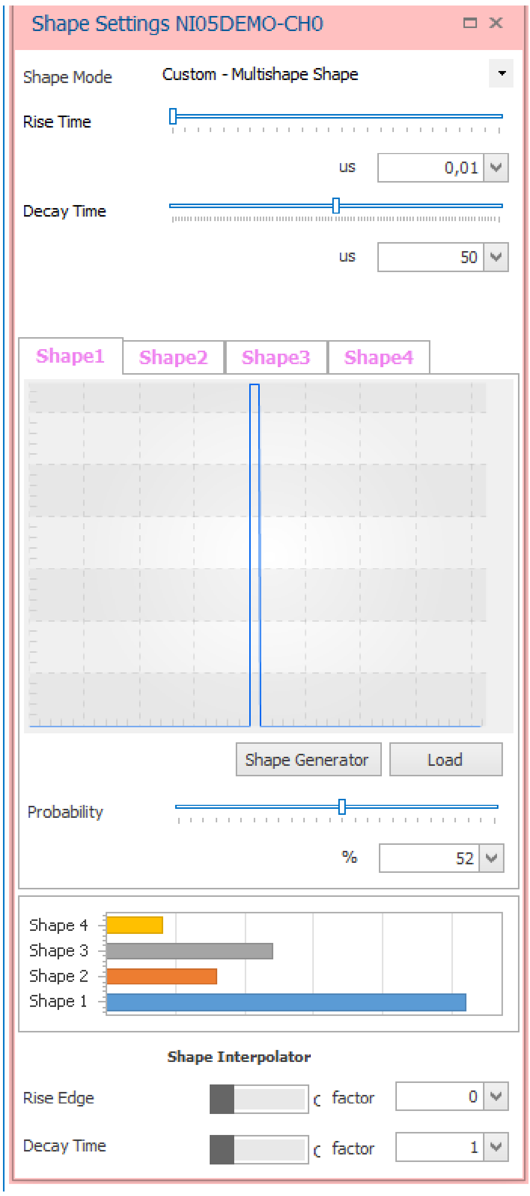
Note: The user can set the maximum value of amplitude and then rescale it on the main GUI.

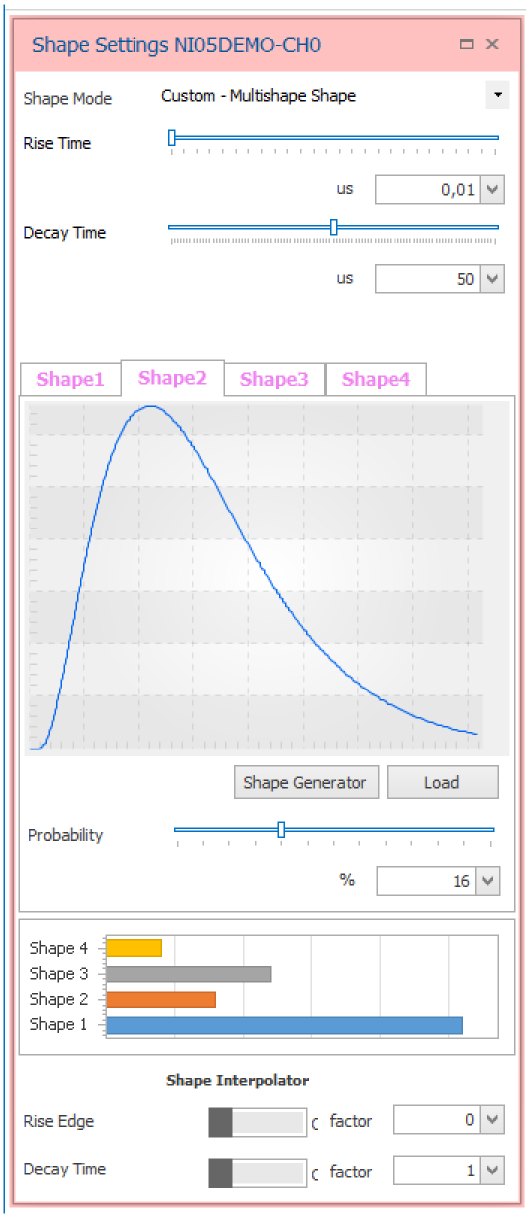
Note: The Tau is limited by the DAC frequency (i.e. minimum value 1 ns).

Multi-Shape

The instrument can be initialized to simultaneously generate signals with different shapes to test pulse shape discrimination. Two different waveforms can be programmed on the same output channel with programmed probability of occurrence.
The option is enabled by selecting “**Multi-Shape**” from shape mode list.

Doing so, all the tabs Shape 1 ... 4 will be activated and it will become possible to assign to each memory class a different shape. For each shape the user wants to use, please proceed as for loading a single shape.





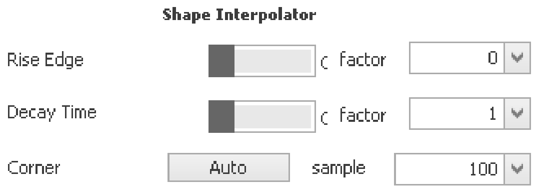
Modify the relative probability of occurrence of the shapes by changing the values in the box or moving the sliding bar.

Set the probability to zero for the shapes which are not going to be used.

In case of Multi-Shape operation mode, the resources for the pile-up emulation are shared between the enabled shapes (see the **Pile-up section**).

Shape Interpolator

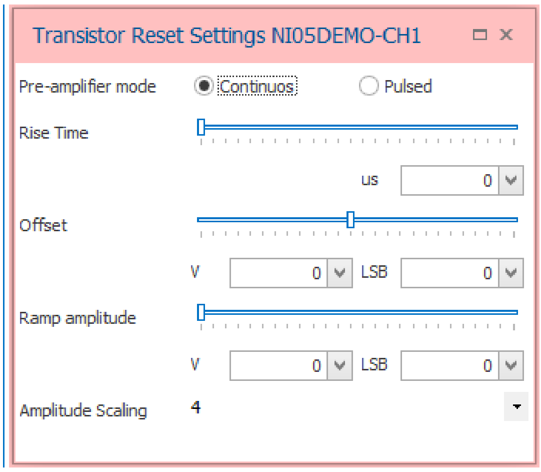
Since the system clock is equal to 1 GHz, i.e. 1 ns of sampling period, and the compliant signal shapes are at maximum 4096 samples long, the maximum time duration of a signal is 4.09 us.
In order to get signals up to 26 ms of length, the real signal can be ideally divided in two parts: the "rising" and "falling" regions divided by a "**Corner**" point. Each region can be interpolated with different factors.



Click on "Rise Edge/Decay Time" to enable the interpolator on the Rise/Fall Time region.
The factor value corresponds to the number of points to be added between two consecutive samples (see Section **Custom Shape – Memory Based data-path** for more details).

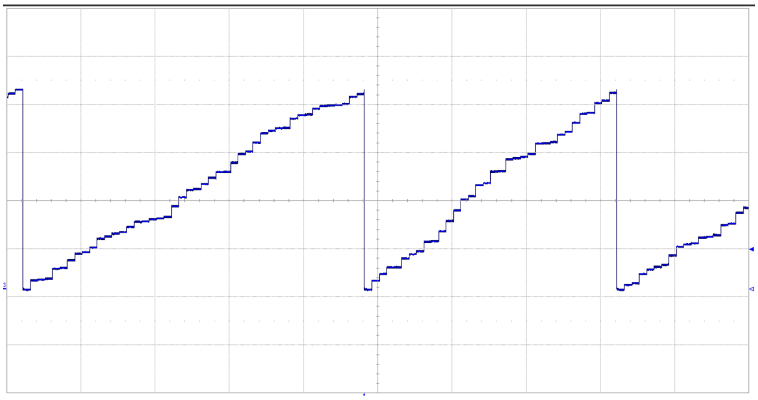
Write the value of the "**Corner**" or click on "**Auto**" to let the algorithm find the actual corner point.

Transistor reset



- **Pre-amplifier mode**: select between Continuous reset and pulsed reset amplifier.
- **Amplitude scaling**: corresponds to the maximum number of events that can be generated in the range Vmax – Vmin. Allowed values are 2, 4, 8, 16, 32, 64, 128;
- **Offset**: add/subtract an offset to the ramp. By default, the ramp is positive only; using the offset is possible to move up/down the output ramp
- **Ramp amplitude**: corresponds to the maximum - minimum value of the staircase.

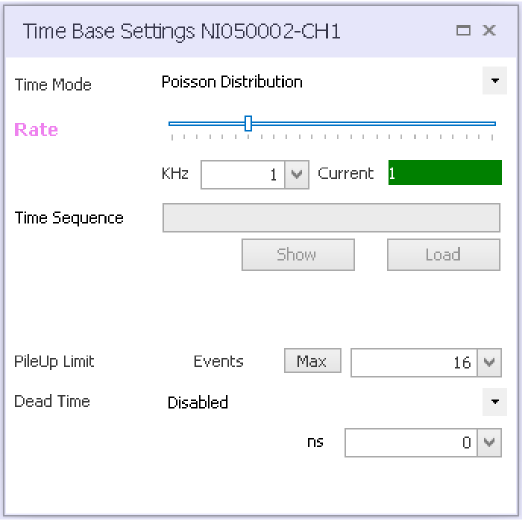
A typical output is shown in the following figure:



The output signal is the integrated amplitude of each input pulses. When the "**Maximum Value**" is reached, the output is instantaneously reset to the "**Minimum Value**".
The "**Amplitude Scaling**" factor works as follows. The device can generate pulses with amplitude ranging from 0 to about 2.2 V each and the maximum allowed dynamics goes from -2.2 to +2.2 V. If two consecutive pulses have the maximum amplitude, they would immediately saturate the output dynamics. Therefore, it is more convenient to apply an amplitude scaling before integrate the pulses. The higher is the scale factor the smaller will be the signal staircase steps. The "Amplitude Scaling" factor corresponds to the number of events at maximum amplitude that can be fitted in the programmed output dynamics. The maximum energy generated will correspond to 4.4 V / amplitude_scaling.
The best amplitude scaling value is a trade-off between the intrinsic resolution and the average number of events in a reset period. For a small scaling factor, the steps will be very large, giving a high signal to noise ratio (SNR). This situation does not always correspond to reality. For higher scaling values the steps will be smaller, up to be compatible with the quantization noise and the intrinsic noise of the device. Indeed, each staircase step is described with 16 bit/$\log_2$(amplitude_scaling), rather than 15 bit for not pulsed reset emulation.

Channel Timebase

There are three options to emulate the time distribution of a setup (selectable by the list **Time Mode**).



- **Constant Rate**: the range of constant rate is from 10^{-2} cps to 5 Mcps. It can be changed writing the value in the white box or moving the sliding bar. The actual generated value is shown under the CURRENT green box.
- **Poisson distribution**: click on the option to enable the Poisson time distribution. The average value can be set from the white box or moving the sliding bar. The actual generated value is shown under the CURRENT green box.
  The plots below show some screenshots in case of constant (top) and random rate (bottom).

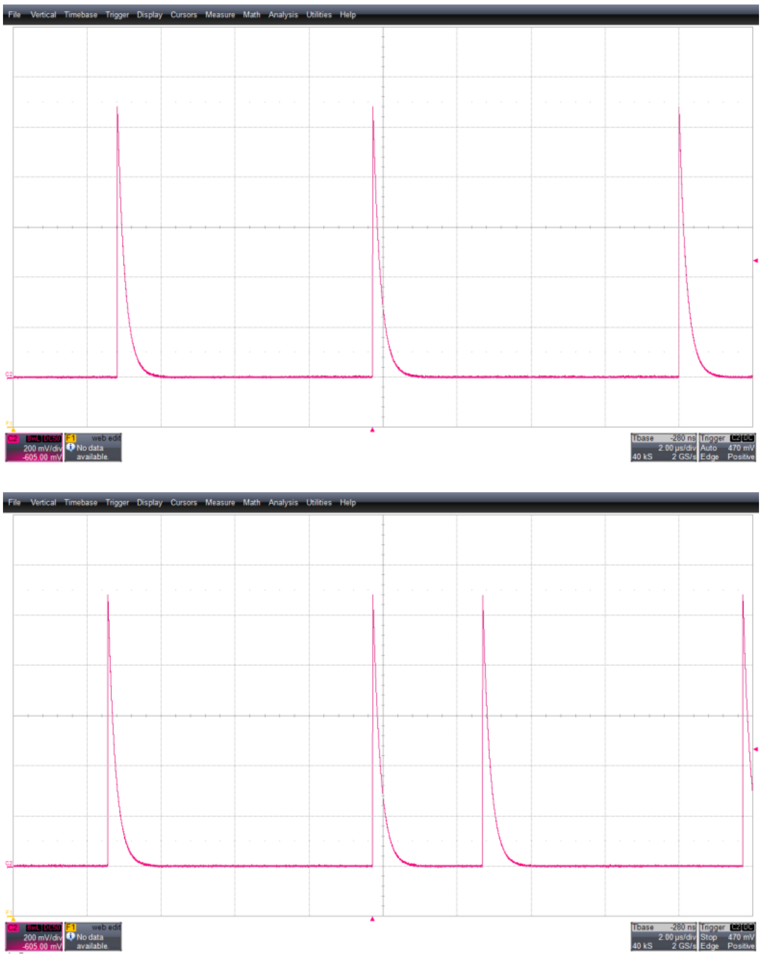
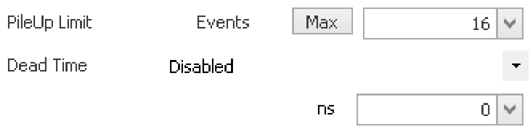


Fig. 12.1: Example of events with constant rate (top) and Poisson (bottom) time distribution.

- **Sequence**: it is possible to load a file with 1 Mpulses of time differences among consecutive pulses. This method allows the generation of an a-priori defined sequence of pulses.
  The picture below shows a screenshot of a temporal distances sequence stored.

Pile-up



The user can set the maximum number of event in pile-up for the “Fast” exponential shape and the “custom” shape choosing a value from 1 (no pile-up) to 15. The “Digital (RC)” option has no limitation of pile-up.

If a critical condition of saturation occurs due to excessive pile-up of pulses, the time-base generator continues to calculate new occurrence times but with no actual generation of the corresponding pulses.

⚠ If this is the case, attention should be paid that the reset and restart of the LFSRs even with the same seed does not correspond to the reproduction of the same pulse sequence.

In case of Multi-Shape operation mode, the resources for the pile-up emulation are shared between the shapes, meaning that only 8 events (with 2 shapes) or 4 (with 4 shapes) with the same shape may be piled-up. The maximum number of piled-up events for each shape can be set.

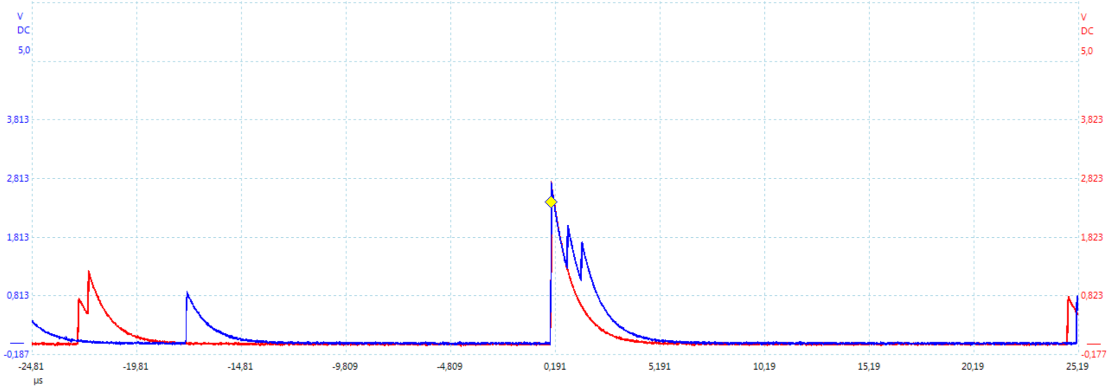
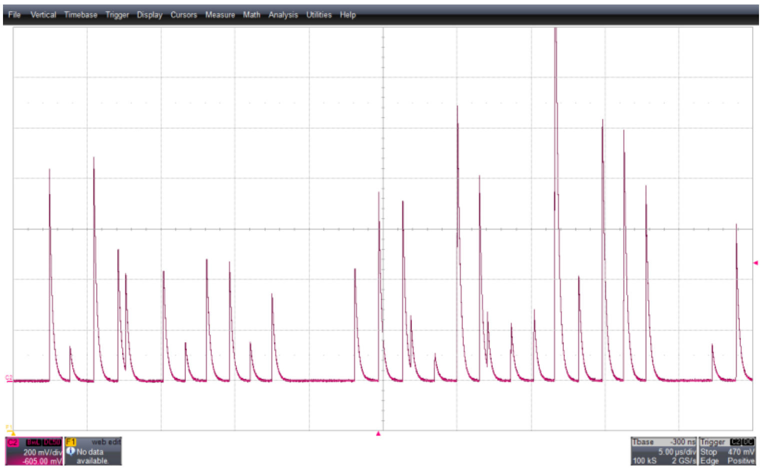


Fig. 12.2: Events in pile-up on both output channels (red and blue).

The Dead time parameter fixes the minimum distance between two adjacent pulses. Click on “**Paralyzable**” or “**Non Paralyzable**” to enable the desired operation mode. In the paralyzable system, if a new event fires during the dead-time interval, the counter of the generation time is reset and re-initialized with a new value. In the non-paralyzable system an event generated in the dead-time is simply discarded.
The pictures show screenshots of the output signal in case of different dead-time specifications – paralyzable on the top and non-paralyzable on the bottom.



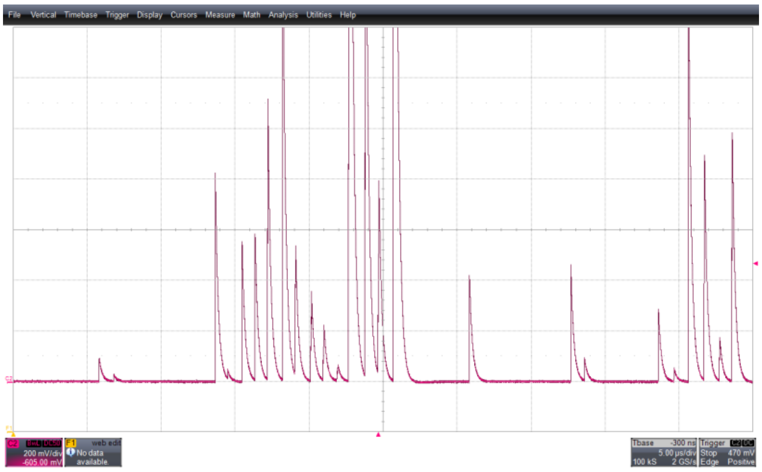
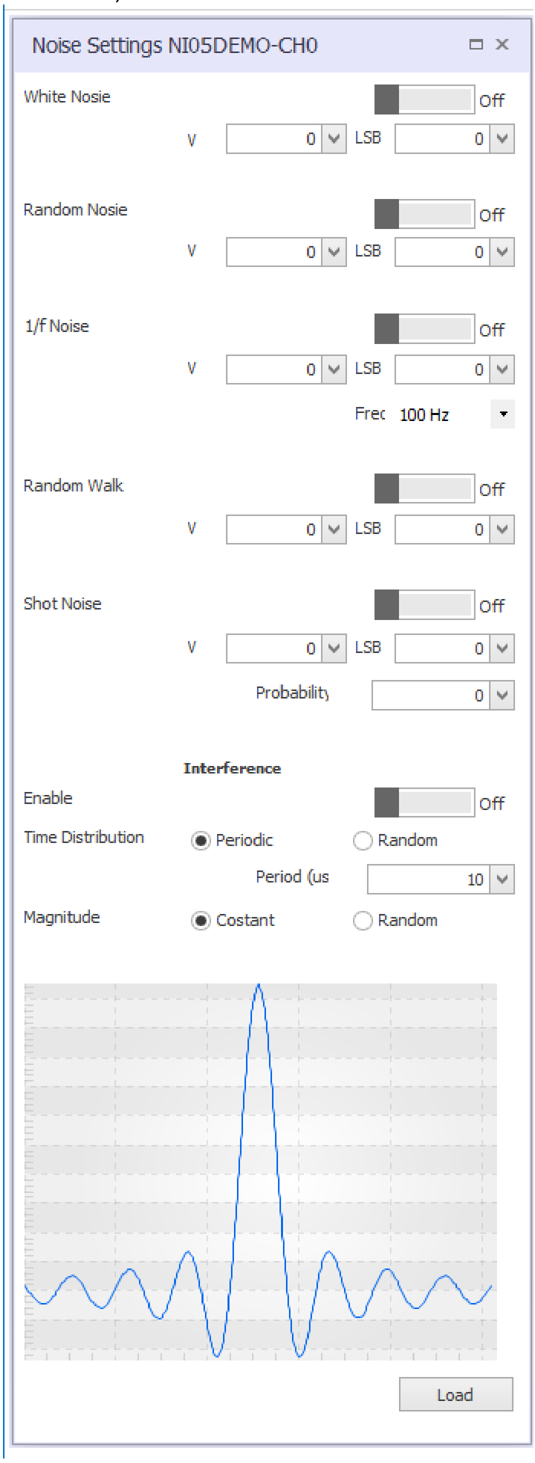


Fig. 12.3: Emulation of events with paralyzable dead-time (top), and non-paralyzable dead-time (bottom).

Noise Emulation

There are several kinds of noise that can be emulated through the device (refer the Section **Noise Emulation** for more details). Click on the desired option to enable it. The possible choices are: random numbers and flicker, i.e. 1/*f*, noise, white noise, random walk.



For each noise it is possible to set the magnitude – both in LSB units, or in Volts – by editing the corresponding box. The user can also set the the desired frequency for the 1/f Noise and the Probability for the Shot Noise.

Note: The white noise is a pseudo-white noise since it is digitally generated and limited in band by the 1 GHz sampling frequency of the DAC.

The white noise is generated with a 64 bits LFSR. The flicker noise too is generated starting from a custom LFSR and shaped by an array of filters, which corresponds to the following set of available noise corner frequencies.

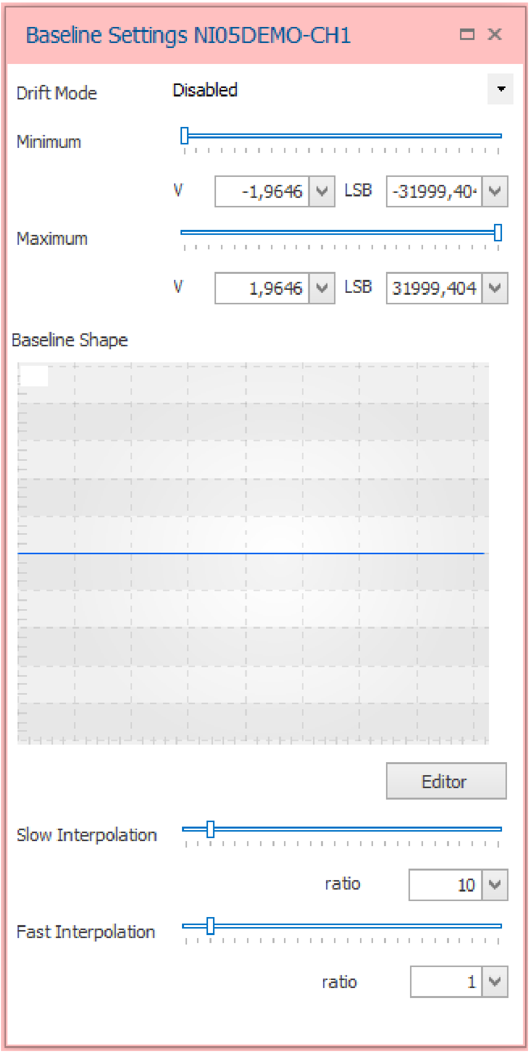
Noise Corner Frequencies (kHz)
0,1
0,5
1
5
10
50
100
500
1000

Baseline Drift

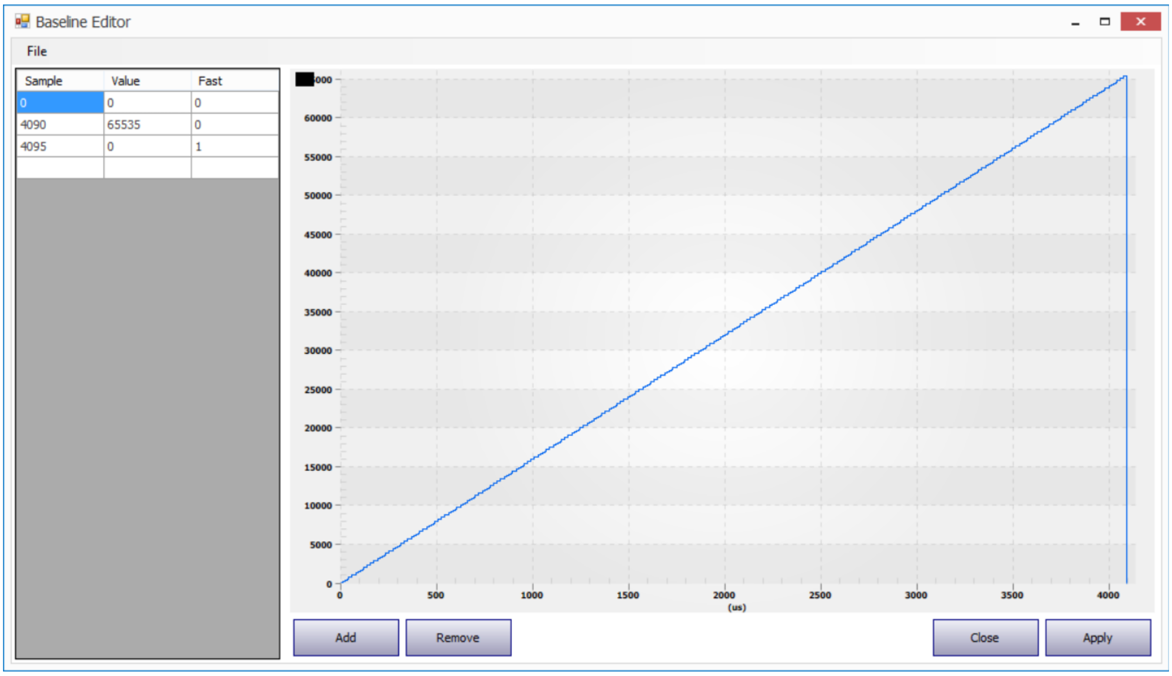
The emulator is able to sum the desired baseline drift to the signal output.

The Digital Detector Emulator can emulate the baseline drift through a memory of 4096 samples in which the user can load the "keypoints" which, once interpolated, represent the trend of the baseline.
It is possible to select 2 types of drift (Drift Mode list): **Linear Drift** and **Custom Drift** based on the keypoints given by the user.
The Linear Drift selects the minimum and maximum excursion of the baseline and then set the value of interpolation slow.

- **Slow Interpolation:** slide to set the interpolation factor to emulate slowly variable signals up to drifts of several seconds
- **Fast Interpolation:** slide to set the interpolation factor to emulate fast variable signals corresponding to the presence of reset or quenching circuits



The Custom Drift uses the keypoints to define a baseline trend. Click on the "**Editor**" to open the "**Baseline Editor**" window. The system reserves 4,096 points for the baseline signal, which are keypoints for a process of linear interpolation.



Cells of the left table can be modified as well. The “Sample” value is the x-axis position of the key point, the “Value” is the y-axis position of the key point, “Fast” = 0/1 to select a “fast” or “slow” interpolator factor. Refer to **Section Baseline drift emulation** for more details.
Use **Add** and **Remove** to add or remove key points. Use (x) to remove all the key points. Use the file menu to load a .cvs file with the desired baseline drift shape, then save the current configuration.

Press “**Apply**” to set the drift on the device.

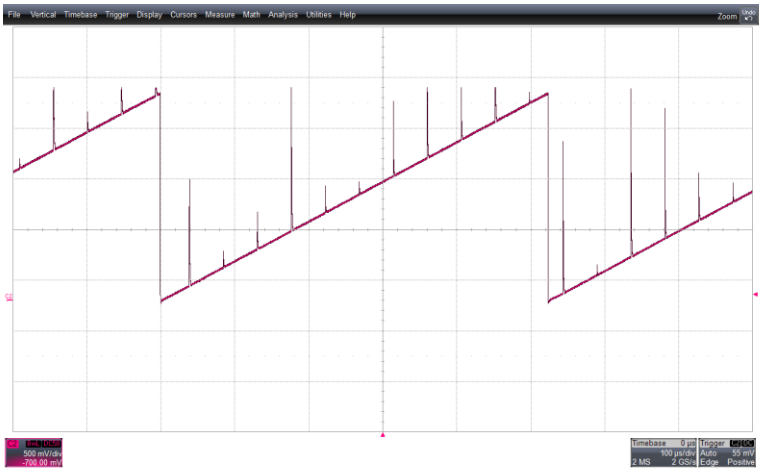
The precision of both interpolation procedures can be regulated by means of the slow and fast interpolation sliders on the “Baseline Settings” box.

The reset of the baseline signal can be accomplished:

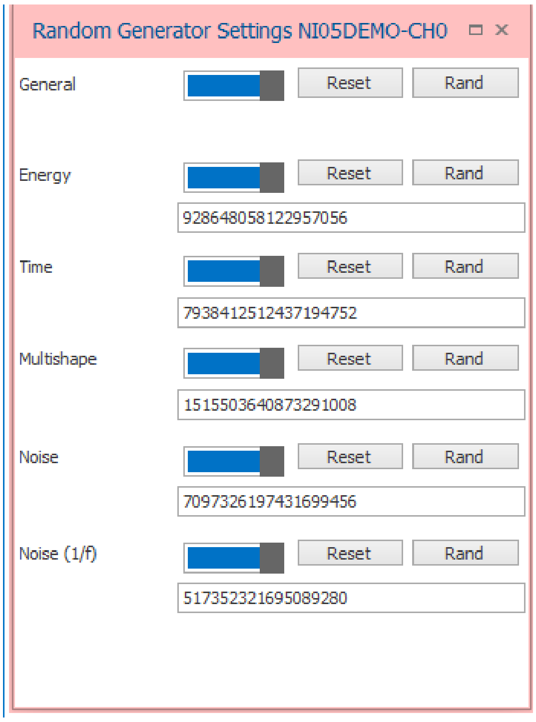
- automatically at the end of the sequence representing the signal;

- externally by means of a digital signal (for instance from a quenching circuit) connected to one of the digital inputs of the Emulator.

The picture below shows the output of the emulator in presence of a linearly varying superposed signal.



Random Generator (LFSR) Settings

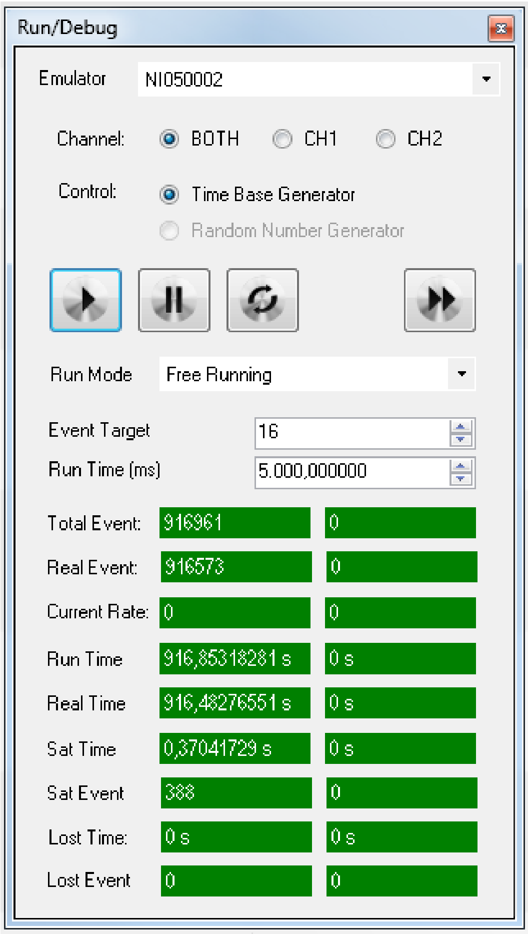


All the LFSRs of each channel can be initialized in terms of the values of the starting seeds. The seeds can be forced for each LFSR by the operator or can automatically and randomly be set by an internal procedure, either global or local. Each LFSR can be stopped, reset and activated independently.

Debug

The device can operate either in “Free-running” mode (default mode at start-up) or in “Step-by-step” debug mode.

Select in the list which emulator channel you are going to use.



The free-running emulation is based on the generation of pseudo-random numbers by means of Linear Feedback Shift Registers (LFSR). The randomness of the seed value starting the generation process ensures the emulation of different sequences, even if in agreement with the set statistical characteristics. On the other side, fixing the seed value allows to generate the same identical sequence every time. This is at the basis of the step-by-step debug. After setting a value of the seed – possible for the “amplitude”, “time” and “noise” from tabs CH1 and CH2 – the generation process starts and evolves either free-running or one step at a time.
In the Step-by-step mode – activated by enabling the “Random Number Generator” control flag (as in the picture above) – the LFSRs can be stopped . While the LFSRs are freezing, the output is still present and it corresponds to the last generated pulse that is repeated iteratively. While the LFSRs are paused, the reset command can be used to reset the status of the LFSRs to the initial seed value previously set. After reset, the same former sequence is re-generated either in free-running or step-by-step mode .

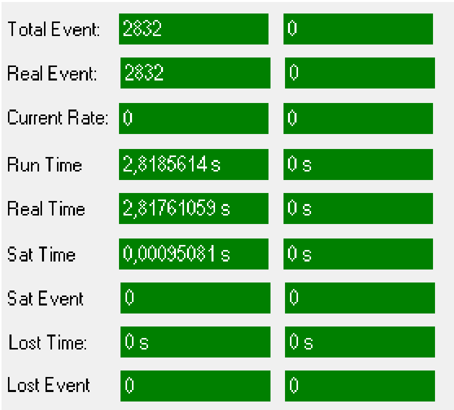
Alternatively, by enabling the “Time Base Generator” control flag, the debug procedure stops the time base generators, which operatively means that the generation of pulses is inhibited and can be activated to produce a pulse at a time at the output of the emulator. If reset is asserted when the generation is stopped, **the LFSRs are reset and the sequence can be started back from the beginning equal to itself**, either in Free-running or Pulse-by-pulse mode.

Run mode menu allows the user to select the target for the emulation process between:

- Free running
- Run Time Limit (chronometer)
- Real Time Limit (amount of time in which the instrument was not in saturation or busy)
- Total Count Limit (number of event generated by the timebase generator)
- Real Count Limit (number of event not in saturation / busy)
- Stop (prevent to run)

The emulation process is controlled by the following buttons

-  Start emulation run
-  Stop, but not reset, the emulation run
-  Reset the current run
-  Restart the current run (stop – reset – run)

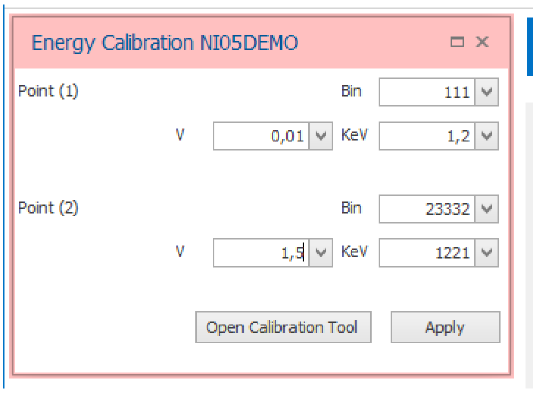


The event display shows (for both channels) the following RUN information in realtime

- **Total event**: number of events generated by the timebase generator
- **Real Event**: number of event that has been really generated on the output (with saturation and busy events)
- **Current rate**: measured rate
- **Run Time**: length of the current run with resolution of 10 ns
- **Real Time**: number of seconds in which the instrument was not in saturation or busy
- **Sat Time**: number of seconds in which the instrument was saturated
- **Sat Event**: number of event triggered while the instrument was saturated
- **Lost Time**: number of seconds in which the instrument was busy
- **Lost Event**: number of events triggered while the instrument was busy

Energy Calibration

The device can be calibrated in energy (KeV). All plots can be visualized in KeV and the user can set the output energy directly in energy value, according to the physical source that it wants to emulate.



The user can set two reference values (either in LSB counts, or in Volts), and assign them the desired value in energy (KeV). Press "**Apply**" to set the calibration.

Digital I/O

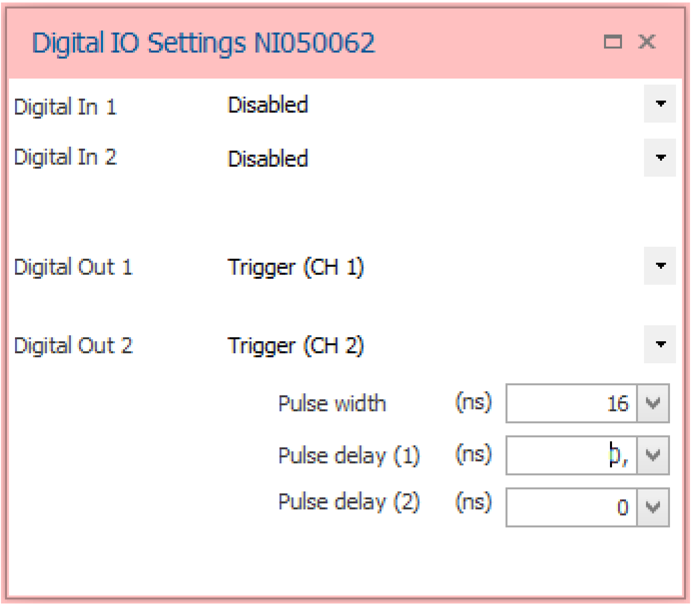
The emulator front-end consists of 6 configurable I/O analog and digital ports. The picture shows the configuration sheet for I/O digital settings.

CAUTION. All I/O gates are LVCMOS compliant. The dynamic range of the analog outputs is 4.4 V.

All signals are considered normally in the low LVCMOS state and active in correspondence of transitions to high LVCMOS state.

Here the list of the main digital input/output functionalities **Digital Input 1 (Digital Input 2)**:

- **Disabled**: the digital input is disabled;
- **Trigger Gate 1 / Trigger Gate 2**: the internal trigger of pulses is put in AND with an external signal of trigger;
- **Veto 1 / Veto 2**: the external signal instantaneously inhibits the output signal;
- **Trigger 1 / Trigger 2**: the pulse generation is synchronized with the external signal. The uncertainty in the signal generation is 4 ns corresponding to ¼ of the sampling frequency (16 ns);



Outputs are pulses with duration defined in the field Pulse width (ns) of the Connector Router mask (see picture above).

- **Disabled**: the digital output is disabled;
- **RUN**: state of activation of the channel;
- **Trigger**: the direct output of the time-base generator, irrespective of its real generation at the output;
- **Accepted Trigger**: the direct output of the time-base generator that corresponds to the actual generation of the correspondent pulse. This means one Accepted Trigger pulse for any analog pulse at the output;
- **Busy**: no more availability of generation engines, which means excess of pile-up. The signal is asserted high until the condition persists;
- **Saturation**: saturation of the dynamic range of the analog output (below screenshot of Analog Saturation digital signal – green – correspondent to the saturation of the analog output – red)
- **DInLoop**: redirect on the Digital Out the input of the Digital In.

It is also possible to set adjust the relative delay between the analog pulse and the corresponding digital output in the field Pulse delay (1/2) (ns).

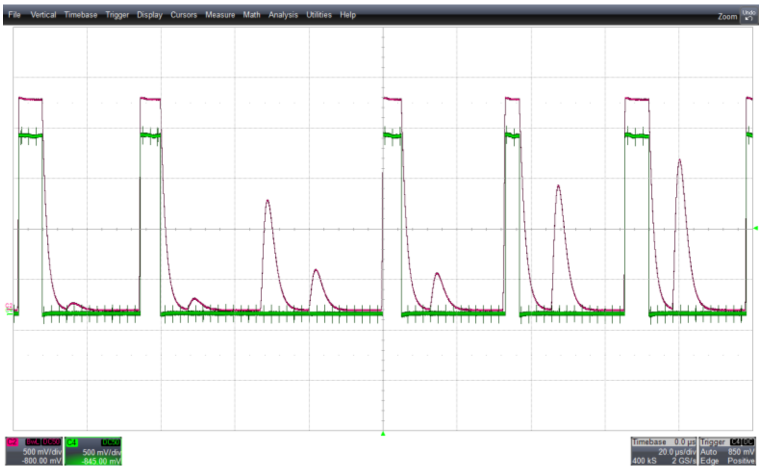
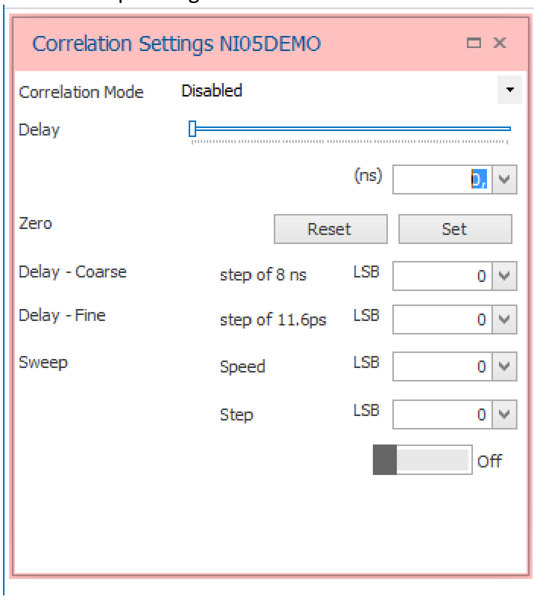


Fig. 12.4: Example of signal saturation. The green trace is on when the analog signal saturates.

Delay-Correlation

Timing Settings

Select the operating mode of the correlation block through **Correlation mode**.

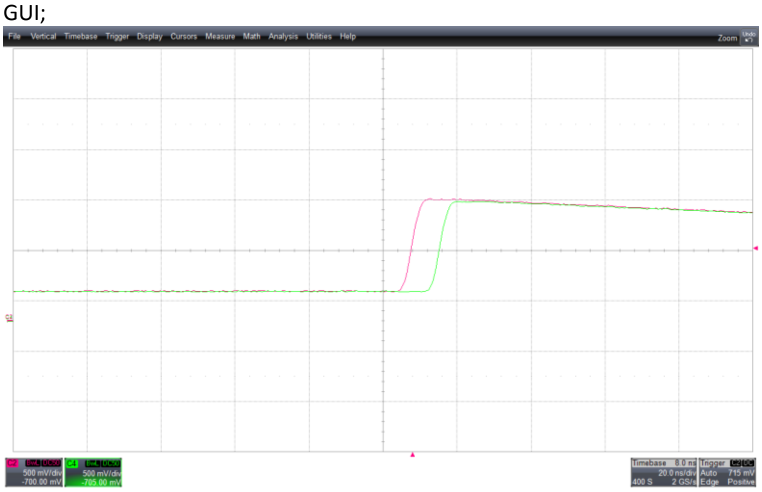


Set the **Delay** using the toolbar or by setting the delay in ns.

The **"Sweep"** button allows to sweep the delay with a programmable **"Speed"** and **"Step"**.

There are three **"Statistical Settings"** options to generate correlated channels.

- **"Channel 2 follows exactly channel 1"**: CH2 output is the exact replica of CH1, also in terms of noise. The user can set the time delay among the two outputs (as from the picture below). The "CH2" tab is disabled from GUI;

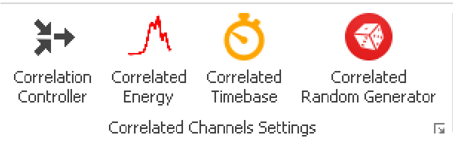


- **"Shared Time Based Generator"**: the two output channels are only time correlated. All the other settings are uncorrelated. The time correlation is set as a fixed delay between channels. The "CH2" tab in the One Touch Interface is enabled but the "Time Distribution" box; the CH2 Time Base widget is not available
- **"Ch3 injects correlated pulses in CH1 and CH2"**: this option allows to completely uncorrelate the two output channels and have a subset of correlated event in the two outputs.

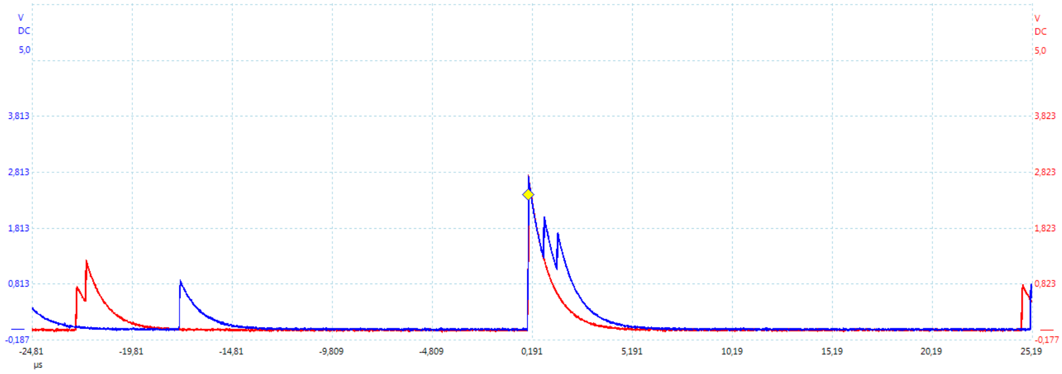
  The device then enables a third internal channel that can be programmed in energy and time. The third channels inject the same event in both CH1 and CH2. Therefore CH1 and CH2 can be programmed to emulate two uncorrelated background events, as the third channel can be programmed to emulate the same signal events.

The energy and time distributions of the third internal channel can be programmed from the following boxes. The shape will be the same as channel 1 and channel 2 respectively. For details on how to set the “Energy mode” and “Time Distribution” refer to the **Channel Energy Settings** and **Channel Timebase Sections**.

When the instrument is set in " **Ch3 injects correlated pulses in CH1 and CH2**", the user can use the controls to set energy, timebase and random generators of the related channel.

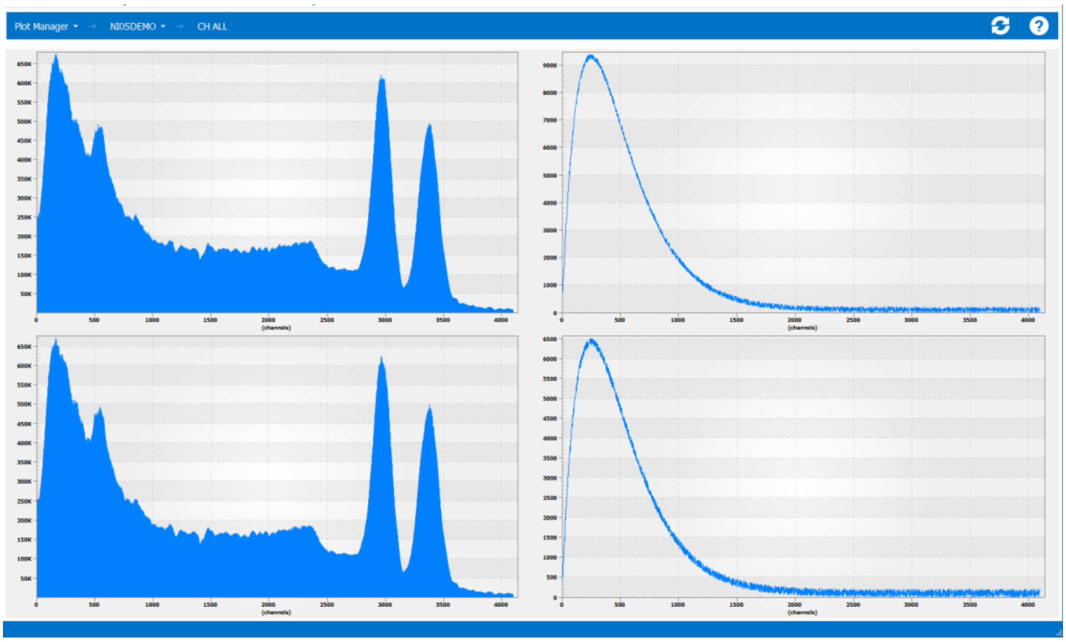


It is possible to emulate pile-up among event from the Channel 1 and Channel 2 and the correlated signals but also emulate pile-up among correlated signals.

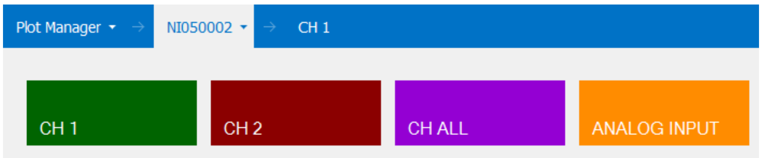


Plot Area

The plot area shows in real time the graphs related to the energy spectrum and to the signal in the time domain generated by the emulator.



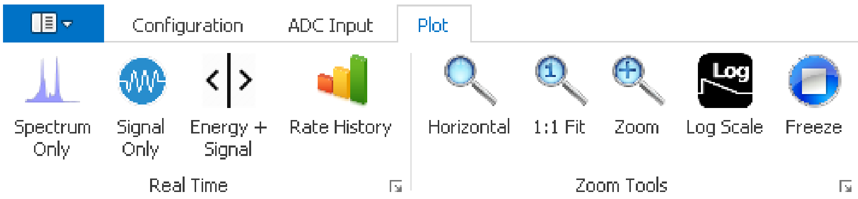
Through the top navigation bar, the user can choose which emulator and which channel to display the charts. For DT5810B options are CH1, CH2, CH1 + 2 and Analog Input.



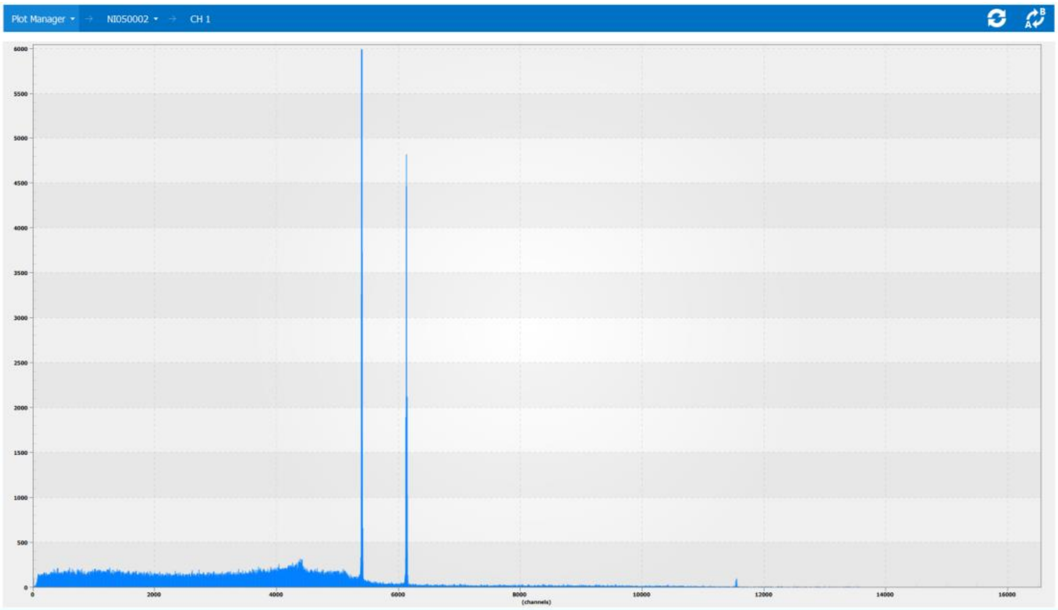
Use the icon to reset the plots and erase the generated spectrum. The A-B icon to change the plot scales.

The **toolbar plot** in the top ribbon bar allows to perform the following actions:

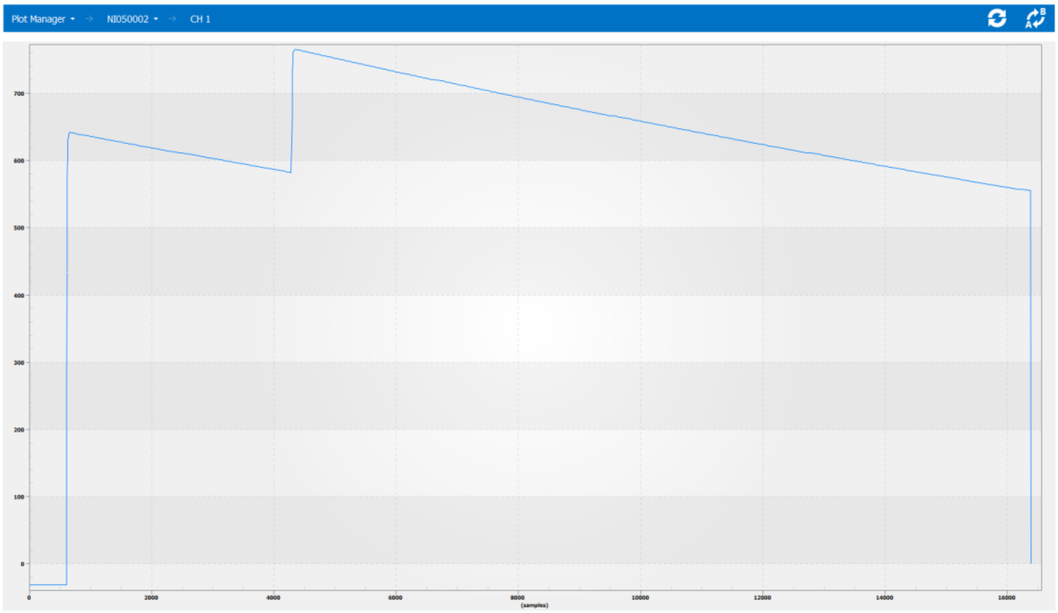
• Choose the plot mode: Energy, Sign, Energy + Signal, Rate History
• Zoom in and out and then reset the zoom
• Switch to Log / Linear scale
• Pause the real-time reading



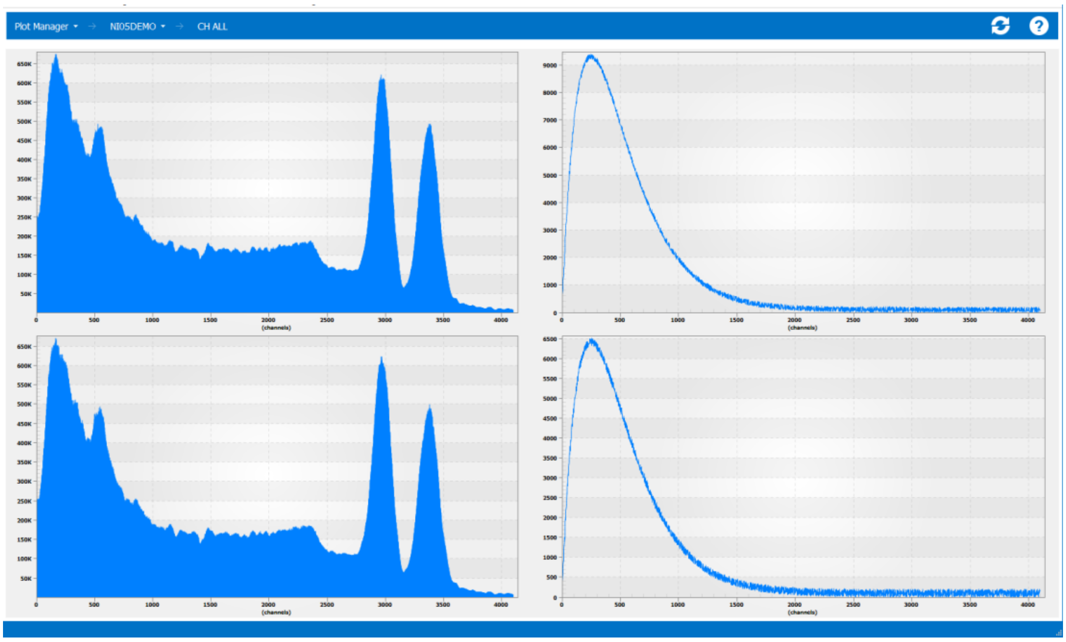
The charts displayed in the plot area depends on what selected in the top navigation area. If one emulation channel is selected the plots show energy and time of what is generated by the output channel(s). If Analog Input is selected the plots are related to the input channel and the MCA feature.
The plot form layout could be split into 1,2 or 4 for dual channels emulator and 1,2 regions for Analog Input.



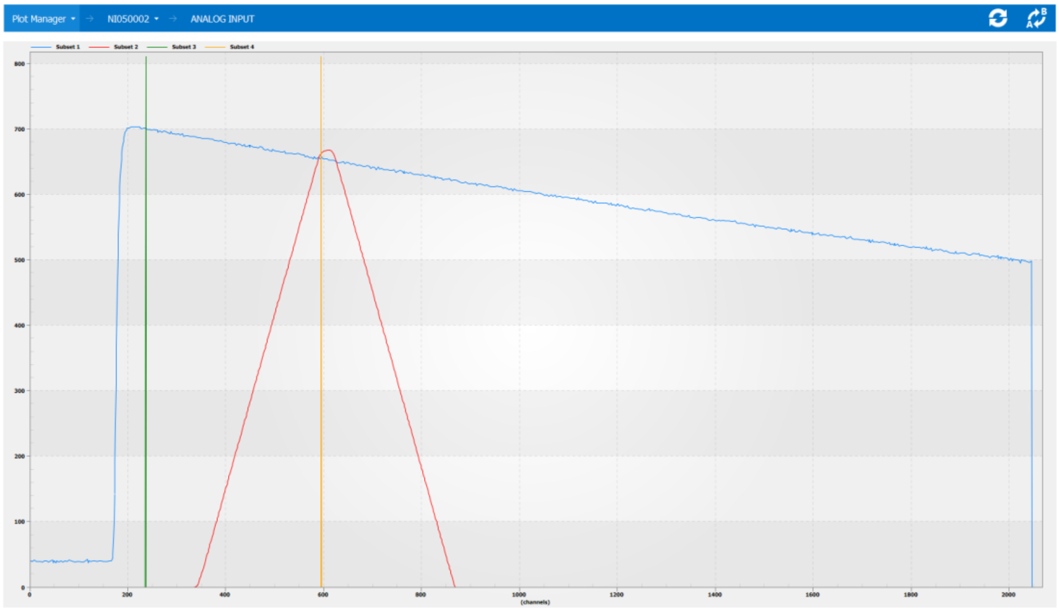
Emulator plot: single view, 1 channel energy



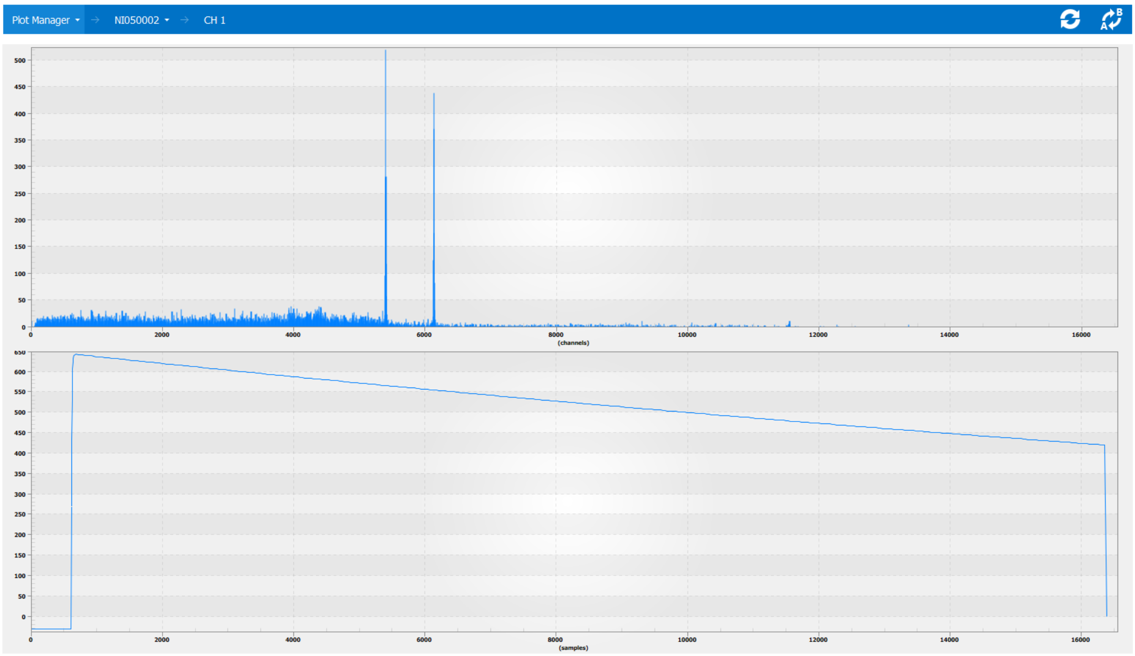
Emulator plot: single view, 1 channel waveform



Emulator plot: quad view, 2 channel waveform + spectrum

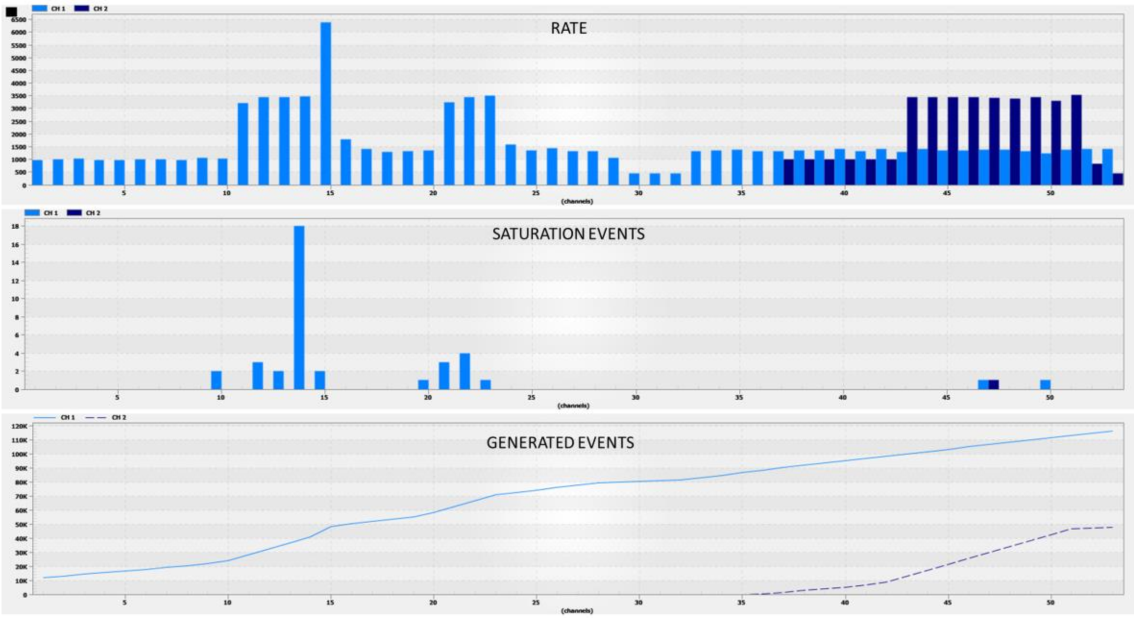


Analog Input: single view, input channel + digital filter



Analog Input: dual view, spectrum + input channel

The rate history plot shows the channels rate trend in the last 100s. The first plot shows the rate, the second shows the number of events generated in saturation while the last one shows the trend of the number of the generated events.



Function Generator

This instrument includes a dual channel function generator able to generate analog signal in frequency range 0÷50MHz. Both option AWG (Arbitrary Waveform Generator) and WG (Waveform Generator) are available on the instrument.

To enable the function generator, select **General Settings** and set **Channel Mode** to **AWG (WG)**.

Note: The emulator is not equipped with a real reconstruction filter (unlike standard waveform generator). This is due to the fact that our main goal is to achieve the fastest possible rise time on exponential signals. The drawback is that sinusoidal waveforms, without filtering, appear like steps.
There are two ways to get around this problem:

- Use the Analog Filter; the output bandwidth will be limited to 19 MHz
- Use the *12 V (slow)* output stage; the output bandwidth will be limited to about 50MHz and the output dynamic will be extended to ±12V

Both options are available in the Channel Settings menu

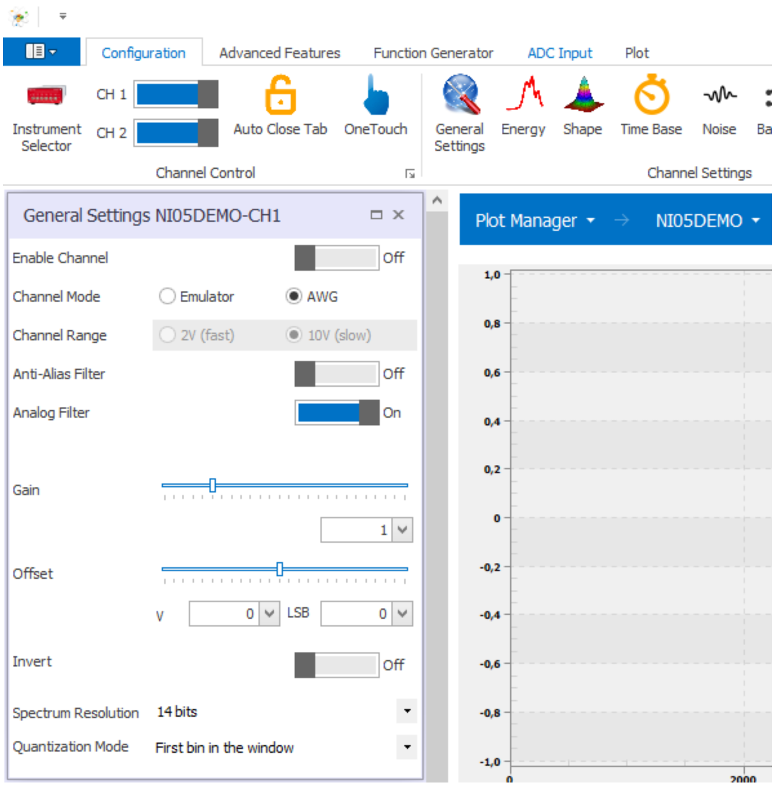


Fig. 12.5:Instructions to enable the waveform generator

In the main menu click on Function Generator. Buttons CH1 and CH2 act like the Output Enable. User could use those button to enable/disable the two outputs and avoid to change to General Settings Tab. AWG CH1 and CH2 buttons open the AWG Widgets.

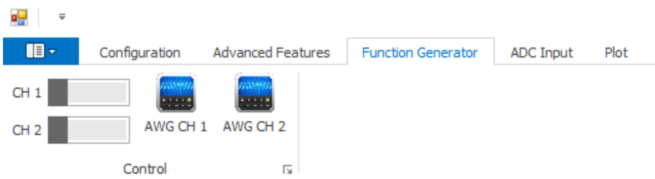


Fig. 12.6: Arbitrary waveform generator main menu

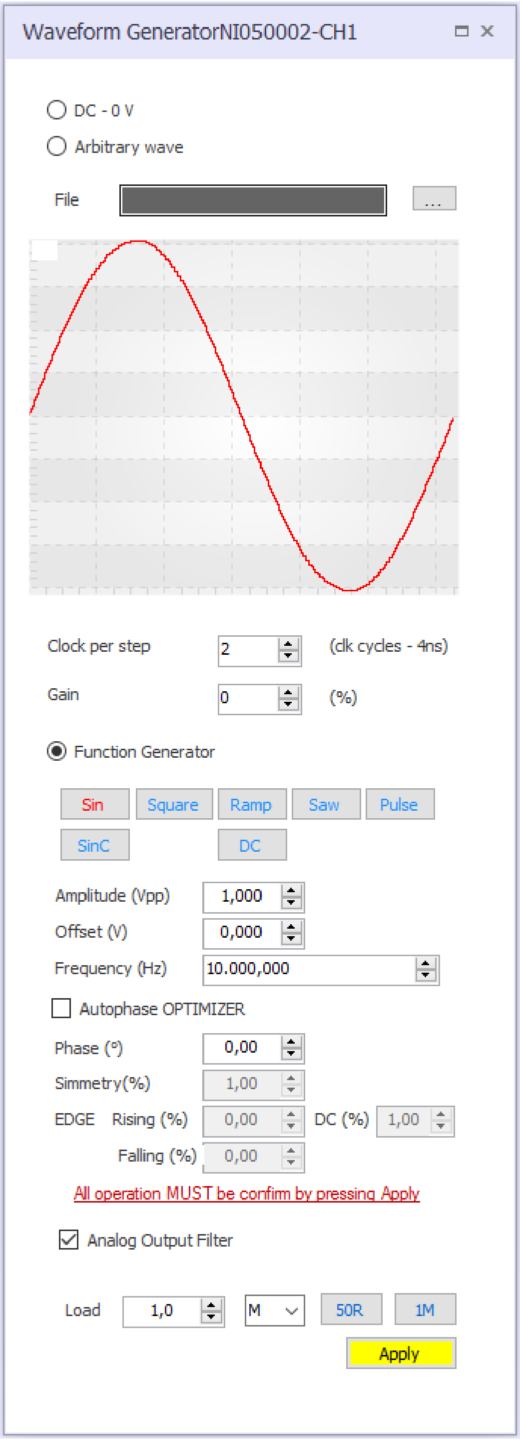


Fig. 12.7: Waveform Generator Control Widget

Arbitrary Waveform Generator

The Emulator can be used as a traditional waveform generator, reproducing the output signals loaded as .csv files in a memory of 64 Mpoints (corresponding to the length of 256 ms) for each channel. In order to reproduce longer signals, the physical samples of the loaded waveform can be used as key-points of an interpolation process. The interpolation factor can be fixed in the field “Clock per step” up to 2^{23}. The amplitude of the generated signal can be controlled by the parameter “Gain”.

Note: All operation MUST be confirmed by pressing “Apply”. The button “Apply” turns yellow when there is a modification to be applied. Then it turns green when there is no modification to be applied.

Note: Full 64 Mpoints memory is available only when input file has a length multiple of 16 points otherwise only the first 8 Mpoints could be loaded in the instrument memory. This limitation is due to the alignment to 256 bits of the DDR3 memory.

Function Generator

The Emulator can be used also as programmable generator of the following waveform: Sine, Square wave, Ramp, Saw tooth, Pulse, SinC and DC level. Output values are automatically adjusted according to the output impedance. The output impedance must be selected through the panel:

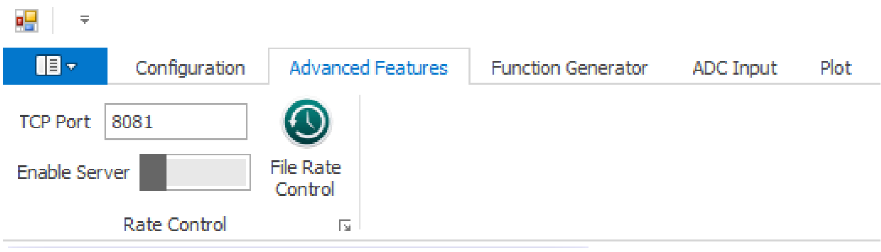
Remote Control and Rate Control

The Advanced Feature Tab includes two main feature:

- a web server to remote control by http protocol some parameters of the emulator software
- a software to control the rate in realtime of multiple channels and multiple emulators reproducing a file

Web Server

The web server uses standard http GET to set parameters on the software from a remote machine or a local software. Default HTTP port is 8081. It could be changed from Advanced Feature Menu.



Check the switch Enable Server to startup the web service.

Note: Depending on the Windows Firewall configuration it is possible that Windows Firewall asks to authorized the software to open the port. If a firewall software is installed on the machine, please open this port for remote access. Ask your IT administrator to add a firewall rule / NAT rule on your company routers to allow the remote control of this the software on the selected port.

The features control supported by the remote interface are:

- Time distribution (costant rate / poisson)
- Rate
- Energy distribution (costant / spectrum)
- Energy value

Remote load of the spectrum is not supported. Spectrum must be loaded in the software before try to select it remotely

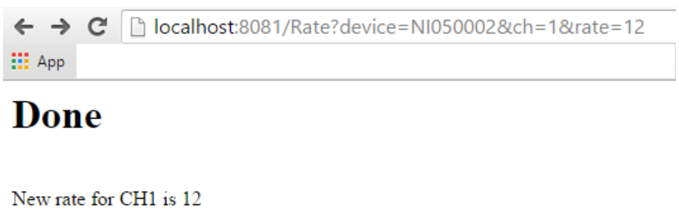
The syntax to set the parameters is the following:

Example: http://localhost:8081/Rate?device=NI050002&ch=1&rate=12

SET RATE: http://<ip>:<port>/Rate?device=<SERIAL NULBER>&ch=<1/2>&rate=<VALUE IN KHz>
SET COSTANT RATE: http://<ip>:<port>/Emission?device=<SERIAL NULBER>&ch=<1/2>&mode=costant
SET POISSON DISTRIBUTION: http://<ip>:<port>/Emission?device=<SERIAL NULBER>&ch=<1/2>&mode=poisson
SET ENERGY: http://<ip>:<port>/Energy?device=<SERIAL NULBER>&ch=<1/2>&lsb=<VALUE IN LSB (0..16383)>
SET COSTANT ENERGY: http://<ip>:<port>/EnergyMode?device=<SERIAL NULBER>&ch=<1/2>&mode=costant
SET SPECTRUM ENERGY: http://<ip>:<port>/EnergyMode?device=<SERIAL NULBER>&ch=<1/2>&mode=spectrum

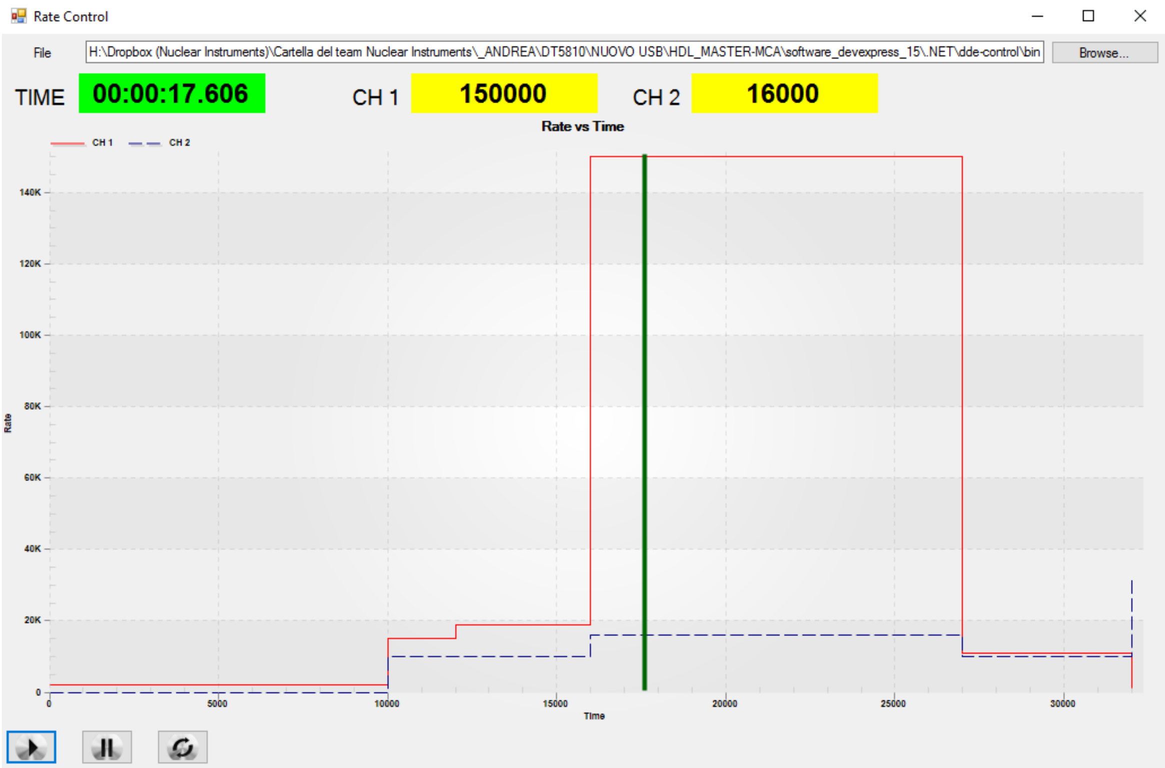
Note: URLs are case sensitive

Try to control one of this parameter by type the url in your browser.



File Rate Control

The file rate control feature allows the user to reproduce in loop a file containing the rate profile for both cannels of the selected emulator.
This feature is usefull when the user wants to emulate a radiation source moving in front of a detector.



Select a CSV file by click browse. The plot will show the profile of the rate of the two channels in function of the time.
Press play and the software will start to reproduce the file. A green bar will show the current position in the file.
The top displays indicate the current position in the file and the current rate for both channels.

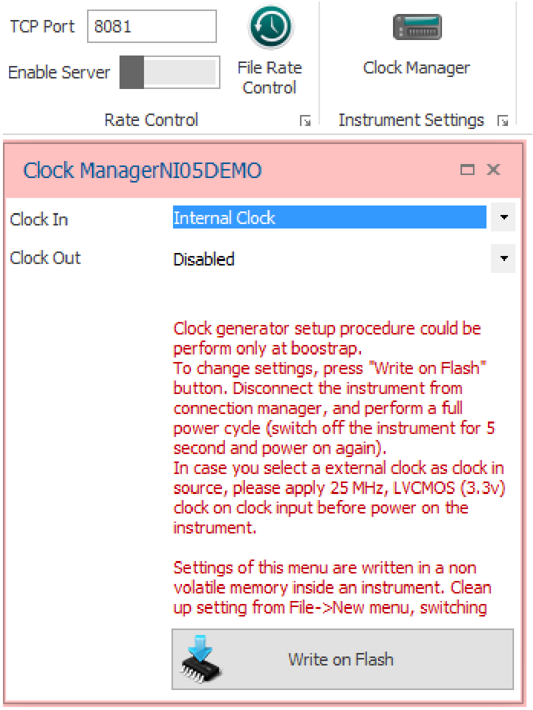
The file format is the following

TIME (ms), RATE CH1 (ev/s), RATE CH 2 (ev/s)

Example
0,2000,3000
10000,15000,8000
12000,19000,3000
16000,1500005000
27000,11000,1000
32000,1000,100

Clock Manager

The DT5810 features the possibility of sharing the clock among sevelar modules in order to have all of them emulating synchronously. This can be performed thought the CLK-OUT/IN lemo connectors on the DT5810 rear panel that allow to connect the emulators in daisy chain.



Through the Clock In menu the user can choose if the instrument works using the internal clock or a 25 MHz extrenal one. In the latter case the procedure written in the widget has to be followed. Through Clock Out menu the user can choose if the instrument will propagate (or not) a 25 MHz +/- 250 mV signal.

Analog Input

This instrument includes a 156 MSPS 14 bit ADC that could be used for two purposes

- Mix the analog input with the emulation output In this configuration the result of the emulation process is mixed to the signal acquired by the ADC from the analog input. Analog input could be scaled and traslated by a fixed offset.
  This feature could be used to inject in the signal coming from a real detector an emulated signal with a different spectrum and time distribution. For example, it is useful to test processing systems designed to separate a particular source from a background.
- Analyze the signal from a real detector to extract the spectrum and the signal shape to initalize the emulation process. The instrument behaves like a MCA (Multi Channel Analyzer): it takes the output signal of a real detector and calculate in realtime the spectrum and the waveform averaging the input signal.
  The output of the MCA could be imported by one or both emulator channels to initialize the emulation process.

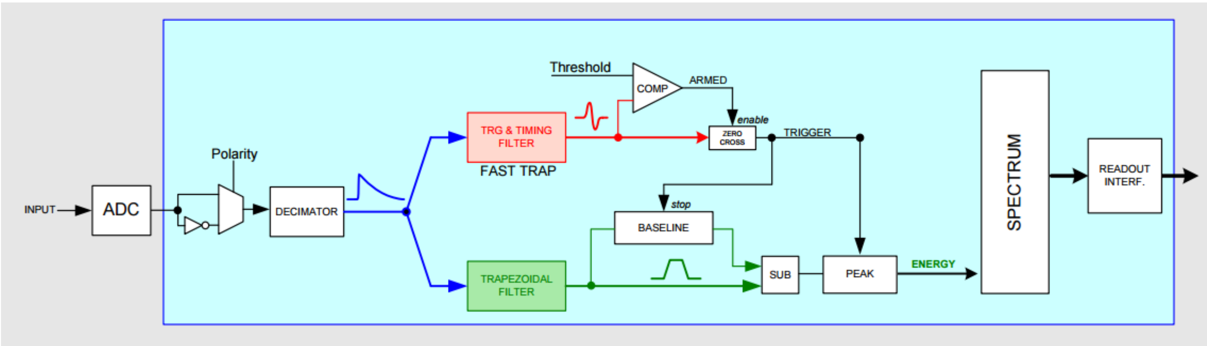
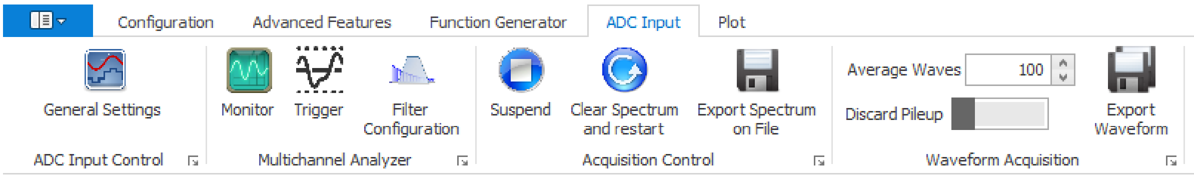


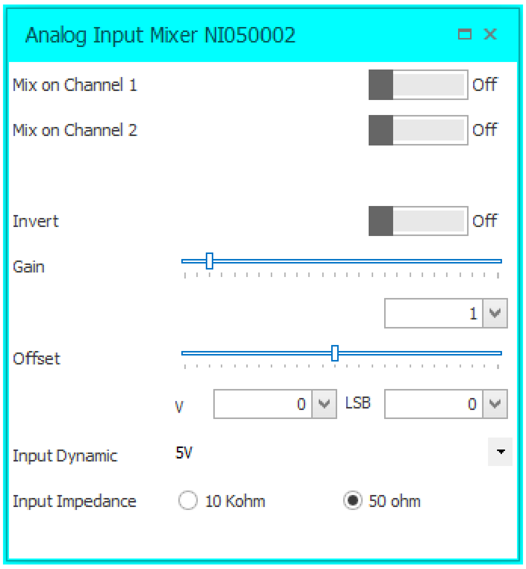
Fig. 12.8: Block diagram of the Analog Input / MCA of the DT5810.

The analog input features are controled from the ADC Input menu



General Settings

General Settings widget groups all settings related to the input analog front-end.



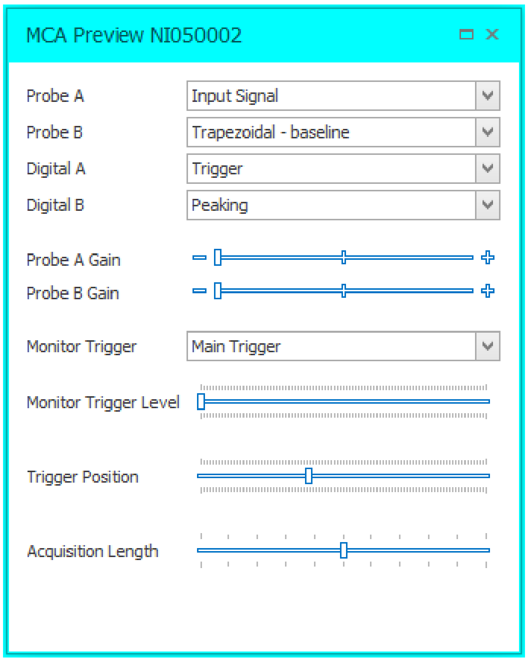
Mix on channel 1 / 2: Select to sum the emulation output 1 / 2 to the data acquired by the ADC
Invert: Change polarity of input signal
Gain : Digital gain on input signal
Offset: Add / Subtract a digital offset from input signal
Input Dynamic: Max Amplitude (on 50R) of the input signal
Input Impedance: Select 10 Kohm or 50 ohm impedance.

Multichannel Analyzer Menu

MCA menu includes all features to control the analyzer core that calculate the spectrum from a detecor. There are three submenu:

- **Monitor**: include all settings related to the control monitor
- **Trigger**: include settings related to the trigger and fast trapezoid filter
- **Filter Configuration**: allows the user to configure the energy filter adapting it to the input signal

Monitor



Probe A/B : Select the input channel for the two analog probes between:

- Input Signal
- Trigger (fast trapezoid)
- Trigger CR-RC (second derivative of the fast trapezoid)
- Trapezoidal
- Trapezoidal – baseline (trapezoidal filter corrected by the baseline)
- Baseline
- Energy (value of energy sampled on the center of the flat top)

Digital A/B: Select the input channel for the two digital probes between:

- Trigger (main trigger point)
- Peaking (point where the energy is sampled)
- Saturation Inibit (high when the instrument is discarding events after an input saturation)
- Baseline Inibit (baseline extimation inhibit)
- Pileup Inibit (high when the instrument is discarding events after a pulse to avoid pileup)

Probe Gain A/B: scale the two digital tracks to optimized the amplitude in respect to the signal amplitude

Monitor Trigger: Select the trigger source for the monitor (only):

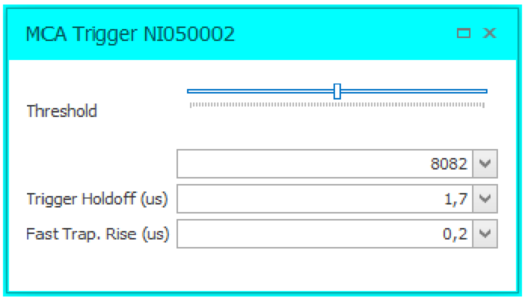
- Self Trigger: Trigger when the input signal crosses (on rising edge) the Monitor Trigger Level threshold. It beahaves exaclty like an oscilloscope trigger
- Main Trigger: uses the main system trigger (the same trigger of the energy sampling) to trigger on the input waveform
- Free Running: the monitor grabs the signal in free running without any syncronization with the input signal

Monitor Trigger Level: threshold used by the monitor trigger when self trigger is selected

Trigger Position: move the trigger sampling point on the time axis (similar to the horizontal position of an oscilloscope)

Acquisition Length: scale the monitor time axis (similar to the timebase of the oscilloscope)

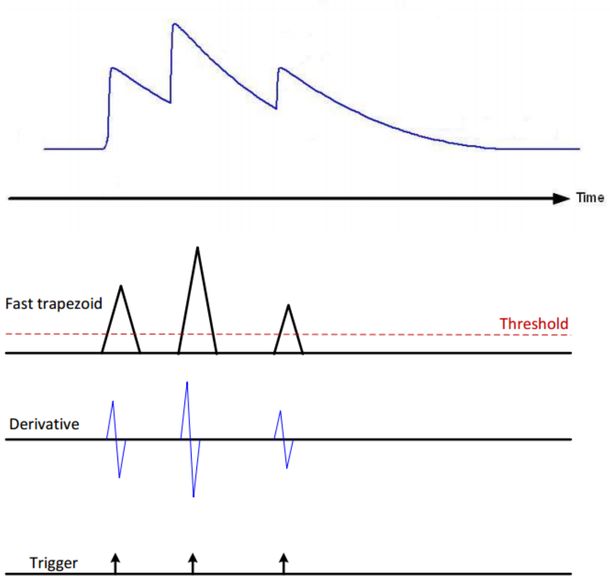
Trigger



The MCA can trigger on a fast trapezoid signal, whose rise time can be defined by the user. The fast trapezoid rise time can range from 10 ns up to 300 ns (**Fast Trap. Rise**).

The threshold is then referred to the fast trapezoid itself, and the threshold crossing arms the trigger. To get higher precision in the trigger position, a subsequent derivative of the fast trapezoid is created. The trigger fires at the zero crossing of the derivative signal itself. The user can only see the fast trapezoid trace on the signal inspector.

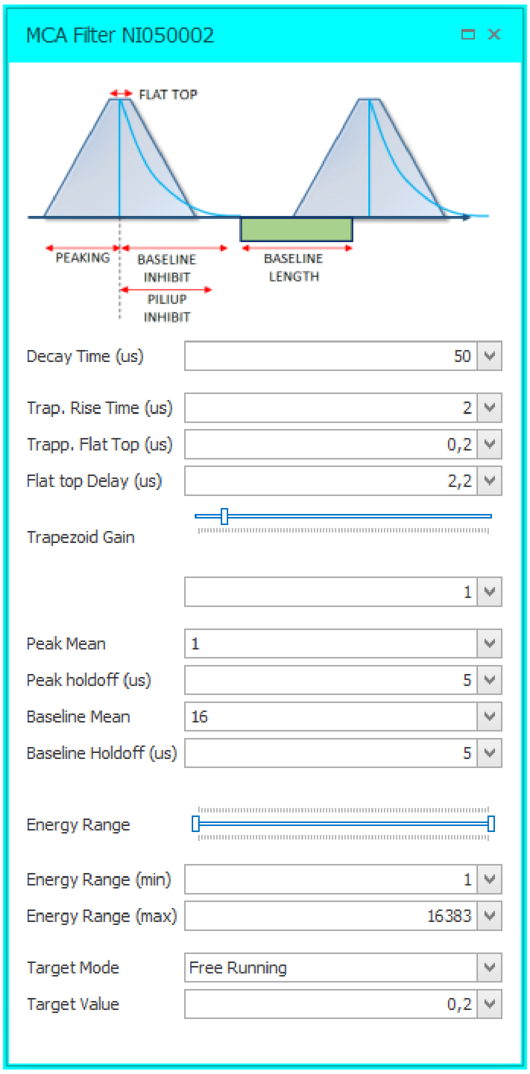
Setting the threshold value corresponds to set the LLD (lower level discrimination) of the energy spectrum (**Threshold**). The user can check from the histogram which value corresponds to the set threshold level.



Filter Configuration

The trapezoidal filter is a filter able to transform the typical exponential decay signal generated by a charge sensitive preamplifier into a trapezoid whose flat top height is proportional to the amplitude of the input pulse (that is to the energy released by the particle in the detector).

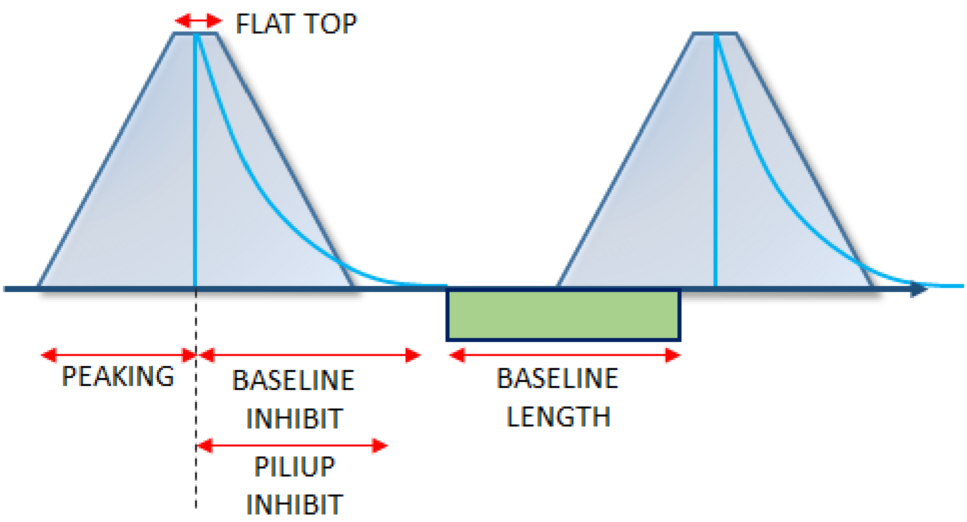
This trapezoid plays almost the same role of the shaping amplifier in a traditional analog acquisition system. There is an analogy between the two systems: both have a “shaping time” constant and must be calibrated for the pole-zero cancellation. For both, a long shaping time gives a better resolution but has higher probability of pile-up. Both are AC coupled with respect to the output of the preamplifier whose baseline is hence removed, but both have their own output DC offset and this constitutes another baseline for the peak detection.



Like the Gaussian pulse of the Shaping Amplifier, also the trapezoid requires an accurate pole-zero cancellation in order to guarantee the correct return to the baseline at the end of the falling edge. To correctly set the pole-zero cancellation the user must take care of setting the proper Trapezoid **Decay Time** (which corresponds also to the Input Decay Time value) to avoid either undershoot or overshoot effects. Pole Zero Cancellation can reduce the signal artifacts due to pulses pile up occurring when the counting rate is high compared to the pulse decay.

The Decay Time also affects the trigger due to the fact that the fast trapezoidal filter is calculated on using this value for pole-zero compensation.

The energy filter parameter are shown in the following picture

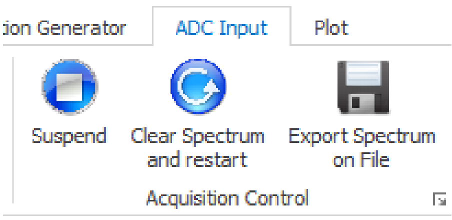


An IIR-based trapezoidal filter is implemented into the FPGA to emulates an analog shaper and generates in output a continue shape that could be monitored selecting Trapezoidal on one of the Analog Probes.
If the user sets correcly the exponential decay time and the input signal is exponential like, the filter generate in output trapezoidal shape for each exponential input. This is the low speed path. On the high speed path the trigger circuit identifies the incoming pulses separating each other.

The **Trap Rise Time** (peaking time) parameters sets the half length of the trapezoid filter.
The **Trap Flat Top** sets the flat top length.
The **Flat top Delay** sets the sampling point for the energy (peaking time) respect to the trigger point. For example, if Trap Rise Time is set to 2us and flat top is 0.2us a good value for the Flat top Delay is 2.1 us.
Trapezoidal Gain allows the user to rescale the trapezoid to fit at best the 14 bits of the spectrum.
Peak Mean sets number of samples averaged on flat top to extimate the energy.
Peak holdoff sets the length of the pileup filter inibition starting from the sampling point. If an event occurred in this interval the event is discarded.
Baseline holdoff sets the length of the baseline extimation inibition starting from the sampling point. Baseline samples are not averaged until the inibition time expires.
Baseline Mean is number of samples used to extimate the baseline. The system uses the last n-samples before a trigger to extimate the baseline.
Energy Range set the two sliders that allow the user to limit the energy values that are accepted by the spectrum histrogram.
Target Mode: Select between free running, limited by time (ms), limited by number of total counts.

Acquisition Control

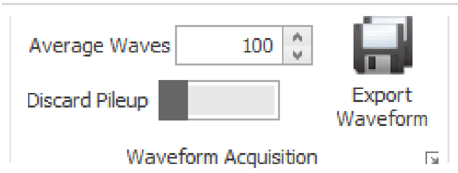
Acquisition Control menu allows the user to start the measurement, export the spectrum or reset it. Spectrum could be exported in proprietary format and its only purpose is to initialize the emulation process.



Start/Suspend: start and stop spectrum acquisition
Clear Spectrum and restart: Clean up the spectrum
Export Spectrum on File: Save the file on a file on disk. The file could be imported as energy spectrum in one of the emulation channel.

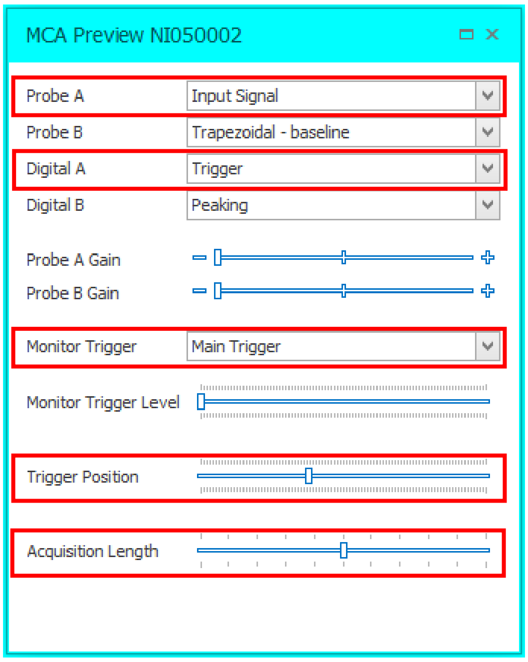
Waveform acquisition

This tool allows the user to acquire the input signal in order to calculate an averaged ideal shape of the input signal. The input signal (as seen on probe A of the monitor) is acquired and averaged until the Average Waves target is not reached. In the average process is excluded waveform in pileup if descard pileup switch is selected. As the signal is acquired, it is normalized to 1 and then averaged. Press **Export Waveform** to start the acquisition process.
The measured signal shape could be imported as custom shape in one of the emulator channel.



While acquisition is in progress, the status bar reports the progress

Note: In order to proper acquire the signal user must select on **Analog Probe A**, the input channel and on **Digital Probe A** the trigger. Moreover the user must regulate the **Trigger Position** and **Acquisition Length** in order to fit at best the input signal with the available samples shown by the monitor windows. All these settings are available in the **Monitor** Menu.



Accepted File Formats

The accepted file formats are the following:

- **Comma Separated Values (.csv)**. This file format can be used for any kind of settings of the device.

For the energy spectrum the user can choose one of the following file formats:

- **CAEN digitizer files (.dat)**. The histogram data is written in the following format:
  Number_of_sample_#1 Energy_value_#1
  Number_of_sample_#2 Energy_value_#2
  … …
- **Spectrum files (.spectrum)**. This is the internal spectrum file format. Any spectrum created by the GUI will have this file format.
- **ANSI4242 (.xml)**. This is the standard ANSI4242 file format.

In the next section we are going to see in details the .csv file format for the specific setting.

Shape (.csv)

Values are organized in column format. Each value begins a new paragraph:

Value #1

Value #2

Value #3

Value #4

Value #5

Value #6

……..

The values represent the amplitude of the shape and are quantized to 16 bits (from -32768 to 32767 integers). The maximum number of samples of the waveform is 4096, but a lower size can be used.

Spectrum (.csv)

The values are organized in column format. Each value begins a new paragraph

Value #1

Value #2

Value #3

Value #4

Value #5

Value #6

……..

The values represent the probability and are quantized to 16 bits (from 0 to 65535). The number of bins is equal to 16384 (14 bits). If a lower number of bins is inserted, the remaining samples are automatically set to zero.

Interference (.csv)

Values are organized in column format. Each value begins a new paragraph:

Value #1

Value #2

Value #3

Value #4

Value #5

Value #6

……..

The values represent the amplitude of the interference samples and are quantized into 16 bits (from -32768 to 32767 integers). The maximum number of samples of the waveform is 4096, but a lower size can be used.

Signal Generator (.csv)

Values are organized in column format. Each value begins a new paragraph:

Value #1

Value #2

Value #3

Value #4

Value #5

Value #6

……..

The values represent the amplitude of the interference samples and are quantized to 16 bits (from -32768 to 32767 integers). The maximum number of samples of the waveform is 10^6, but a lower size can be used.

Sequence in Energy and Time Mode (.csv)

The user must generate two files, one about the amplitude of events and one about the temporal distance between two consecutive events.

Values are organized in column format. Each value begins a new paragraph:

Value #1

Value #2

Value #3

Value #4

Value #5

Value #6

……..

The maximum length of the list is 500 kword.
The values of the file of amplitudes modulate the amplitudes of the signal shape.

13 Instructions for Cleaning

The equipment may be cleaned with isopropyl alcohol or deionized water and air dried. Clean the exterior of the product only.
Do not apply cleaner directly to the items or allow liquids to enter or spill on the product.

13.1 Cleaning the Touchscreen

To clean the touchscreen (if present), wipe the screen with a towelette designed for cleaning monitors or with a clean cloth moistened with water.
Do not use sprays or aerosols directly on the screen; the liquid may seep into the housing and damage a component. Never use solvents or flammable liquids on the screen.

13.2 Cleaning the air vents

It is recommended to occasionally clean the air vents (if present) on all vented sides of the board. Lint, dust, and other foreign matter can block the vents and limit the airflow. Be sure to unplug the board before cleaning the air vents and follow the general cleaning safety precautions.

13.3 General cleaning safety precautions

CAEN recommends cleaning the device using the following precautions:

1) Never use solvents or flammable solutions to clean the board.
2) Never immerse any parts in water or cleaning solutions; apply any liquids to a clean cloth and then use the cloth on the component.
3) Always unplug the board when cleaning with liquids or damp cloths.
4) Always unplug the board before cleaning the air vents.
5) Wear safety glasses equipped with side shields when cleaning the board

14 Device decommissioning

After its intended service, it is recommended to perform the following actions:
- Detach all the signal/input/output cable
- Wrap the device in its protective packaging
- Insert the device in its packaging (if present)

THE DEVICE SHALL BE STORED ONLY AT THE ENVIRONMENT CONDITIONS SPECIFIED IN THE MANUAL, OTHERWISE PERFORMANCES AND SAFETY WILL NOT BE GUARANTEED

15 Disposal

The disposal of the equipment must be managed in accordance with Directive 2012/19 / EU on waste electrical and electronic equipment (WEEE).

The crossed bin symbol indicates that the device shall not be disposed with regular residual waste.

16 Technical Support

To contact CAEN specialists for requests on the software, hardware, and board return and repair, it is necessary a MyCAEN+ account on www.caen.it:

https://www.caen.it/support-services/getting-started-with-mycaen-portal/

All the instructions for use the Support platform are in the document:

CAEN
Tools for Discovery

SAFETY, STORAGE AND SETUP INFORMATION
PRODUCT SUPPORT SERVICE AND REPAIR

INFORMAZIONI SU SICUREZZA STOCCAGGIO E INSTALLAZIONE
ASSISTENZA E RIPARAZIONE DEL PRODOTTO

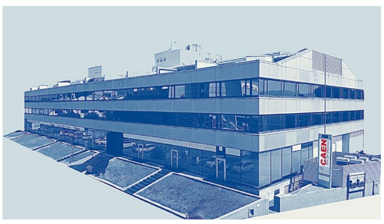
A paper copy of the document is delivered with CAEN boards.

The document is downloadable for free in PDF digital format at:

https://www.caen.it/safety-information-product-support

This page has been left blank on purpose.

CAEN S.p.A.
Via Vetraia 11
55049 - Viareggio
Italy
Phone +39 0584 388 398
Fax +39 0584 388 959
info@caen.it
www.caen.it



CAEN GmbH
Brunnenweg 9
64331 Weiterstadt
Germany
Tel. +49 (0)212 254 4077
Mobile +49 (0)151 16 548 484
info@caen-de.com
www.caen-de.com

CAEN Technologies, Inc.
1 Edgewater Street - Suite 101
Staten Island, NY 10305
USA
Phone: +1 (718) 981-0401
Fax: +1 (718) 556-9185
info@caentechnologies.com
www.caentechnologies.com

CAENspa INDIA Private Limited
B205, BLDG42, B Wing,
Azad Nagar Sangam CHS,
Mhada Layout, Azad Nagar, Andheri (W)
Mumbai, Mumbai City,
Maharashtra, India, 400053
info@caen-india.in
www.caen-india.in