

User Manual UM2580

**DPP-PSD**

Digital Pulse Processing for Pulse Shape Discrimination

Rev. 8 - September 29th, 2016

## Purpose of this Manual

This User Manual contains the full description of the Digital Pulse Shape Discrimination for 720, 725, 730, 751, and DT5790 Digitizer series. The description is compliant with DPP-PSD firmware release **4.9\_131.11** for 720 series and DT5790, DPP-PSD firmware release **4.17\_136.14** for 725 series and 730 series, DPP-PSD firmware release **4.11\_132.8** and **4.12\_132.33** for 751 series<sup>1</sup>, and DPP-PSD Control Software release **1.3.5**. For future release compatibility check in the firmware and software revision history files.

## Change Document Record

Date	Revision	Changes
May 10 <sup>th</sup> , 2012	00	Initial release
June 7 <sup>th</sup> , 2012	01	Updated § 6
October 10 <sup>th</sup> , 2012	02	Document revised to support 751 series
June 5 <sup>th</sup> , 2013	03	Support to new firmware and software releases. Removed Register Chapter(*)
July 21 <sup>st</sup> , 2014	04	Added support to 730 series and DT5790
April 21 <sup>st</sup> , 2015	05	Modified board event header
September 1 <sup>st</sup> , 2015	06	Revised Chapter 1 and Chapter 2. Added 730/751 calibration
October 2 <sup>nd</sup> , 2015	07	Added support to 725 series
September 29 <sup>th</sup> , 2016	08	Added section about 725/730/751 ADC Calibration. Added support to DPP-PSD firmware for 751 >= 132.32 (with CFD). Modified Event Data format for 720 series, added Trigger Hysteresis, Input Smoothing, Veto and Zero Suppression based on Charge Sections for 725/730 series. Revised Sect. CFD Implementation. Added FreeWrites Syntax Section and removed Config File Syntax Section
February 27 <sup>th</sup> , 2018	09	Small corrections to the Memory Organization chapter. Updated Fig. 2.10

(\*) A new document for DPP-PSD register description of 725-730 series **[RD13]**, of 720 **[RD17]**, DT5790 **[RD18]**, and of 751 **[RD19]** is available. For any request contact CAEN Support (see contact details in Chapter **Technical support**)

## Symbols, abbreviated terms and notation

ADC	Analog-to-Digital Converter
DAQ	Data Acquisition
DPP	Digital Pulse Processing
DPP-PSD	DPP for Pulse Shape Discrimination
MCA	Multi-Channel Analyzer
OS	Operating System
PC	Personal Computer
PMT	Photo Multiplier Tube
QDC	Charge-to-Digital Converter
TDC	Time-to-Digital Converter
USB	Universal Serial Bus

## Reference Documents

[RD1]	WP2081 - Digital Pulse Processing in Nuclear Physics
[RD2]	AN2506 - Digital Gamma Neutron discrimination with Liquid Scintillators
[RD3]	“Pulse shape discrimination with fast digitizers”, L. Stevanato et al, NIMA 748 (2014) 33–38
[RD4]	GD2827 - How to make coincidences with CAEN digitizers
[RD5]	UM1935 - CAEN Digitizer User & Reference Manual
[RD6]	GD2783 – First Installation Guide to Desktop Digitizers & MCA

---

<sup>1</sup> Refer to the document “x751 Family DPP-PSD Firmware Compatibility” for additional details.

- [RD7] GD2512 - CAENUpgrader QuickStart Guide
- [RD8] AN2086 - Synchronization of a multi-board acquisition system with CAEN digitizers
- [RD9] UM2784 – CAENDigitizer LabView User & Reference Manual
- [RD10] UM3074 – Digital Detector Emulator User Manual
- [RD11] UM1935 - CAENComm User & Reference Manual
- [RD12] AN2472 - CONET1 to CONET2 migration
- [RD13] UM4380 – 725-730 DPP-PSD Registers
- [RD14] Technical Information Manual of V1718 and VX1718 VME – USB2.0 Bridge
- [RD15] Technical Information Manual of A3818 PCI Express Optical Link Controller
- [RD16] Technical Information Manual of A2818 PCI Optical Link Controller
- [RD17] UM4855 – 720 DPP-PSD Registers
- [RD18] UM5416 – DT5790 DPP-PSD Registers
- [RD19] UM5110 – 751 DPP-PSD Registers

All documents can be downloaded from: <http://www.caen.it/csite/LibrarySearch.jsp>

---

CAEN S.p.A.  
Via Vetraila, 11 55049 Viareggio (LU) - ITALY  
Tel. +39.0584.388.398 Fax +39.0584.388.959  
[info@caen.it](mailto:info@caen.it)  
[www.caen.it](http://www.caen.it)

© CAEN SpA – 2018

#### **Disclaimer**

No part of this manual may be reproduced in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of CAEN SpA.

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](http://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

Purpose of this Manual.....	2
Change Document Record.....	2
Symbols, abbreviated terms and notation .....	2
Reference Documents .....	2
<b>Index.....</b>	<b>4</b>
<b>List of Figures .....</b>	<b>5</b>
<b>List of Tables .....</b>	<b>6</b>
<b>1. Introduction .....</b>	<b>7</b>
Notes on ADC Calibration (725, 730, and 751 series).....	10
<b>2. Principle of Operation .....</b>	<b>13</b>
Baseline.....	15
CFD implementation for 725, 730, and 751 series .....	16
DPP-PSD trigger management .....	22
Trigger Hysteresis (725 and 730 series only) .....	23
Input Smoothing (725 and 730 series only) .....	24
Veto (725 and 730 series only).....	24
Online PSD selection .....	26
Zero Suppression based on Charge .....	26
<b>3. Acquisition Modes.....</b>	<b>27</b>
<b>4. Memory Organization .....</b>	<b>28</b>
720 (DT5790), 725 and 730 series .....	29
751 series.....	31
Event Data Format.....	32
Channel Aggregate Data Format for 720 (DT5790) series .....	32
Channel Aggregate Data Format for 725 and 730 series.....	34
Channel Aggregate Data Format for 751 series .....	37
Board Aggregate Data Format .....	41
Data Block.....	43
<b>5. Getting Started.....</b>	<b>44</b>
Scope of the chapter.....	44
System Overview .....	44
Hardware Setup .....	45
Drivers and Software .....	46
Firmware and Licensing.....	47
Practical Use .....	49
<b>6. Coincidences and Synchronization .....</b>	<b>81</b>
<b>7. Software Interface.....</b>	<b>82</b>
Introduction.....	82
Block Diagram.....	82
Drivers & Libraries .....	83
Drivers.....	83
Libraries .....	83
Installation .....	85
GUI Description.....	86
The Tab “General” .....	87
The Tab “Channels” .....	90
The Tab “Oscilloscope” .....	93
The Tab “Histogram” .....	96
The Tab “Stats” .....	98
The Tab “Output” .....	99
The Tab “Logger” .....	103
The Tab “HV Config” .....	104
FreeWrites File Syntax .....	106
Example 1: How to generate a global trigger from external trigger on TRG-IN connector .....	107
Example 2: How to cut on PSD threshold online .....	107
Example 3: How to modify the input range on 725/730 series .....	108

Notes on Firmware and Licensing.....	108
<b>8. Technical support .....</b>	<b>109</b>
<b>Appendix A .....</b>	<b>110</b>
Pile-up management in DPP-PSD firmware for 751 digitizer family .....	110
Definition of pile-up in DPP-PSD firmware (751 family).....	110
Pile-up management in DPP-PSD firmware (751 family) .....	112
Pile-up with the DPP-PSD Control Software (751 family).....	113
<b>Appendix B .....</b>	<b>119</b>
Pile-up management for 720, 725 and 730 digitizer family .....	119
Definition of pile-up in DPP-PSD firmware (720, 725 and 730 family).....	119
How to set the pile-up rejection for DPP-PSD firmware (720, 725 and 730 family) .....	119

## List of Figures

Fig. 1.1: Plot of typical Gamma-Neutron waveforms .....	8
Fig. 2.1: Functional Block Diagram of the DPP-PSD .....	13
Fig. 2.2: Long and short gate graphic position with respect to a couple of input pulses. The blue pulse has a longer tail than the red one.....	13
Fig. 2.3: Diagram summarizing the DPP-PSD parameters. The trigger fires as soon as the signal crosses the threshold value. Long Gate, Short Gate, Gate Offset, Pre-Trigger, Trigger Hold-Off, and Record Length are also shown for one acquisition window.....	14
Fig. 2.4: Baseline calculation as managed by the DPP-PSD algorithm .....	15
Fig. 2.5: Classical implementation of the Constant Fraction Discriminator. The input signal is first attenuated by a factor $f$ , then inverted and delayed. The resulting signal has its zero crossing corresponding to the set fraction $f$ .....	16
Fig. 2.6: Implementation of the digital CFD in the DPP-PSD firmware of 725, 730, and 751 series. Mid-scale value corresponds to 8192 in case of 725 and 730 series, 512 for 751 series. ....	17
Fig. 2.7: A typical CFD signal. Red points are the digital samples. The Sample Before the Zero Crossing (SBZC) and the Sample After the Zero Crossing (SAZC) are the samples before and after the zero crossing. The SBZC corresponds to the Coarse Time Stamp. The algorithm can also evaluate the Fine Time Stamp, and the corresponding time will be the sum of the SBZC and the Fine Time Stamp .....	18
Fig. 2.8: Diagram showing the structure of the trigger management of the DPP-PSD firmware.....	22
Fig. 2.9: Memory management of 725 and 730 series .....	23
Fig. 2.10: Local Trigger Management inside couple 0 of 725-730 digitizer series. Couple 0 is made of channel 0 and channel 1. The same applies for the other couples of the 725 and 730. ....	23
Fig. 2.11: Trigger Hysteresis in DPP-PSD firmware. Any other triggers are inhibited after the over-threshold until the input reaches the value of half the threshold. ....	23
Fig. 2.12: Example of smoothing over four samples. The input samples are averaged over four samples and replaced in the smoothed samples by the mean value. ....	24
Fig. 2.13: Example of input of opposite polarity. Top left and top right pictures shows the case of a negative pulse polarity, where the trigger is correctly evaluated both for LED and CFD discrimination. Bottom left picture shows an opposite pulse polarity (positive) and the corresponding CFD (right). To avoid distortions on the baseline (green line) the baseline is kept frozen (yellow) and the event is not triggered. Also the CFD is not inhibited by default. ....	25
Fig. 2.14: 2D scatter plot of PSD parameter vs Energy in a neutron-gamma application. On the left the 2D plot before the cut, on the right the plot after the cut on PSD.....	26
Fig. 4.1: Data organization into the Internal Memory of x720 digitizer and DT5790 .....	29
Fig. 4.2: Data organization into the Internal Memory of x725 and x730 digitizer .....	29
Fig. 4.3: Data organization into the Internal Memory of x751 digitizer.....	31
Fig. 4.4: Channel Aggregate Data Format scheme for 720 series .....	32
Fig. 4.5: Channel Aggregate Data Format scheme for 725 and 730 series .....	34
Fig. 4.6: Channel Aggregate Data Format scheme for 751 series .....	37
Fig. 4.7: Board Aggregate Data Format scheme for 720 (DT5790) series .....	41
Fig. 4.8: Board Aggregate Data Format scheme for 725 and 730 series.....	41
Fig. 4.9: Board Aggregate Data Format scheme for 751 series.....	42
Fig. 4.10: Data Block scheme .....	43
Fig. 5.1: CAEN DPP-PSD System components .....	44
Fig. 5.2: The hardware setup including the digitizer running the DPP-PSD firmware used for the practical application .....	45
Fig. 5.3: CAENUpgrader settings for DPP-PSD firmware upgrade.....	48
Fig. 5.4: Input signal DC offset adjustment description .....	59
Fig. 5.5: 2D scatter plot of PSD parameter vs Energy in a neutron-gamma application [RD2] .....	79
Fig. 7.1: The DPP-PSD Control Software block diagram.....	82
Fig. 7.2: Libraries and drivers required for the DPP-PSD system .....	84
Fig. 7.3: Common Bar .....	86
Fig. 7.4: Tab "General" in case of 720 series. ....	87
Fig. 7.5: Tab "General" in case of 725, 730 and 751 series.....	87

Fig. 7.6: Connection Window.....	88
Fig. 7.7: Tab “Channels” .....	90
Fig. 7.8: Input signal DC offset adjustment description .....	91
Fig. 7.9: Tab “Oscilloscope” .....	93
Fig. 7.10: The Trace Settings Window.....	96
Fig. 7.11: Tab “Histogram” .....	96
Fig. 7.12: Energy Calibration window .....	97
Fig. 7.13: Tab “Stats” in case of 720 series .....	98
Fig. 7.14: Tab “Stats” in case of 725, 730 and 751 series .....	98
Fig. 7.15: Tab “Output” .....	99
Fig. 7.16: Header file structure .....	100
Fig. 7.17: Header file structure for DPP firmware.....	101
Fig. 7.18: Warning message about data saving .....	102
Fig. 7.19: Tab “Logger” .....	103
Fig. 7.20: Tab “HV Config” .....	104
Fig. 7.21: Firmware unlicensed warning message .....	108
Fig. A.1: From top to bottom: (1) two distinct events do not overlap into the same integration gate. This is the case when no pile-up occurred. (2) Two events trigger into the same gate. The pile-up flag is high and three possible scenarios are available: (a) no action is taken and the two pulses are integrated into the integration gate; (b) the event is discarded and no event is saved; (c) a second gate is opened for the second pulse (see next section for further details). (3) Two pulses overlap into the same gate, but the second pulse does not overcome the threshold. The event is not recognized as pile-up .....	111
Fig. B.1: Pile-up definition for 720, 725, and 730 series .....	119

## List of Tables

Tab. 1.1: Supported CAEN digitizers for DPP-PSD firmware .....	9
Tab. 2.1: Summary of the options for the EXTRAS word write in the 725 and 730 series. Extended Time Stamp are 16 bits that become the most significant bits of the time stamp value; “Flags” identifies trigger lost and saturation conditions, Fine Time Stamp, Sample After and Sample Negative Zero Crossing values come from the CFD calculation. ....	19
Tab. 2.2: Summary of the options for the EXTRAS word write for the 751 series. Extended Time Stamp are 16 bits that become the most significant bits for the time stamp value; Fine Time Stamp, Sample Before and After Zero Crossing values come from the CFD calculation; TR identifies whether the CDF or LED option are selected; PP corresponds to the rising/falling edge trigger; Mid-Scale Value/Threshold correspond to 512 in case of CFD, or the Trigger Threshold value in case of LED option. ....	20
Tab. 2.3: Summary of the CFD options for 725 and 730 series, where “Time Stamp” is the timing information of a single event, $T_{coarse}$ corresponds to the SBZC (Fig. 2.7), TTT is the 31-bit TRIGGER TIME TAG word of the event structure (refer to section Channel Aggregate Data Format for 725 and 730 series), EXTRAS is the word of event structure, and $T_{fine}$ corresponds to the difference between the ZC and the SBZC (Fig. 2.7).....	21
Tab. 2.4: Summary of the CFD options for 751 series, where “Time Stamp” is the timing information of a single event, $T_{coarse}$ corresponds to the SBZC (Fig. 2.7), TTT is the 32-bit TRIGGER TIME TAG word of the event structure (refer to section Channel Aggregate Data Format for 751 series), EXTRAS is the word of event structure, and $T_{fine}$ corresponds to the difference between the ZC and the SBZC (Fig. 2.7) .....	21
Tab. 5.1: Examples of connection settings .....	50
Tab. 7.1: Table of the Connection icon values .....	86
Tab. 7.2: Examples of connection settings .....	89

# 1. Introduction

CAEN S.p.A. offers a wide range of digitizers to meet different needs of sampling frequency, resolution, form factor, etc. Besides the use of digitizers as waveform recorder (oscilloscope mode), the user can upload special versions of the FPGA firmware for the **Digital Pulse Processing (DPP)** algorithms. A digitizer running in DPP mode becomes a multipurpose instrument which replaces most of the traditional modules such as MCAs, QDCs, TDCs, Discriminators, etc. (for more details refer to the DPP overview [RD1]).

The purpose of the DPP is to perform online signal processing on detector signals directly digitized, able to transform the raw sequence of samples into a compressed data packet that preserves the information required, minimizing the event data size. DPP algorithms are implemented in the FPGA of the board and can be reprogrammed at any time. In one single module it is possible to have the complete information of the event and the capability to extract all the quantities of interest.

This user manual is intended to describe the **Digital Pulse Processing for Pulse Shape Discrimination** firmware (**DPP-PSD**) running on the 720 series digitizers with the EP1C20 FPGA and DT5790 (12 bit, 250MS/s), on the 725 series (14 bit, 250 MS/s), on the 730 series (14 bit, 500 MS/s), and on the 751 series digitizers (10 bit, 1 GS/s; DES mode not supported). The complete list of digitizers running the DPP-PSD firmware is summarized on **Tab. 1.1**.

A digitizer running the DPP-PSD firmware becomes a multichannel data acquisition system for nuclear physics or other applications requiring radiation detectors. The digitizer accepts signals directly from the detector and implements a digital replacement of *Dual Gate QDC, Discriminator and Gate Generator*. All these functionalities are performed inside the board FPGA without any use of external cables, nor additional boards or delay lines. The acquisition is therefore performed by a single compact system which replaces the traditional analog boards. It is also possible to operate with multi-board systems: the front panel clock, the trigger and the general purpose LVDS I/Os connectors (VME only) make possible the synchronization of several boards.

Finally both the board configuration and the acquisition can be completely managed by a software interface. The CAEN **DPP-PSD Control Software** is a user-friendly software interface which allows the user to set the parameters for the acquisition, to configure the hardware, and to perform the data readout. It allows also to collect the energy and PSD spectra, and to plot and to save data. Drivers, libraries and demo source codes are also available for those who need either to modify the program for their specific needs or to integrate it into their DAQ software.

The main functionalities of a digitizer running DPP-PSD firmware are listed below, where any parameter can be programmed by the provided software:

- Auto selection of the events with a digital leading edge discrimination or a digital constant fraction discrimination (725, 730, and 751 series only);
- Input signal baseline (pedestal) calculation and pedestal subtraction for energy calculation;
- Single gate integration for the energy spectra calculation;
- Double integration of the prompt and delayed charge for Pulse Shape Discrimination

Beside the generation of the energy spectrum of the input pulses coming from a radioactive source, the DPP-PSD firmware allows the charge integration in two different gates for the discrimination of the slow and fast components of the input signal itself. **Fig. 1.1** shows a typical example of signals from neutrons and gamma, where it is possible to see a difference in the waveform shapes. The slow and fast components are used by the algorithm to compute the variable used for the PSD, using the following function:

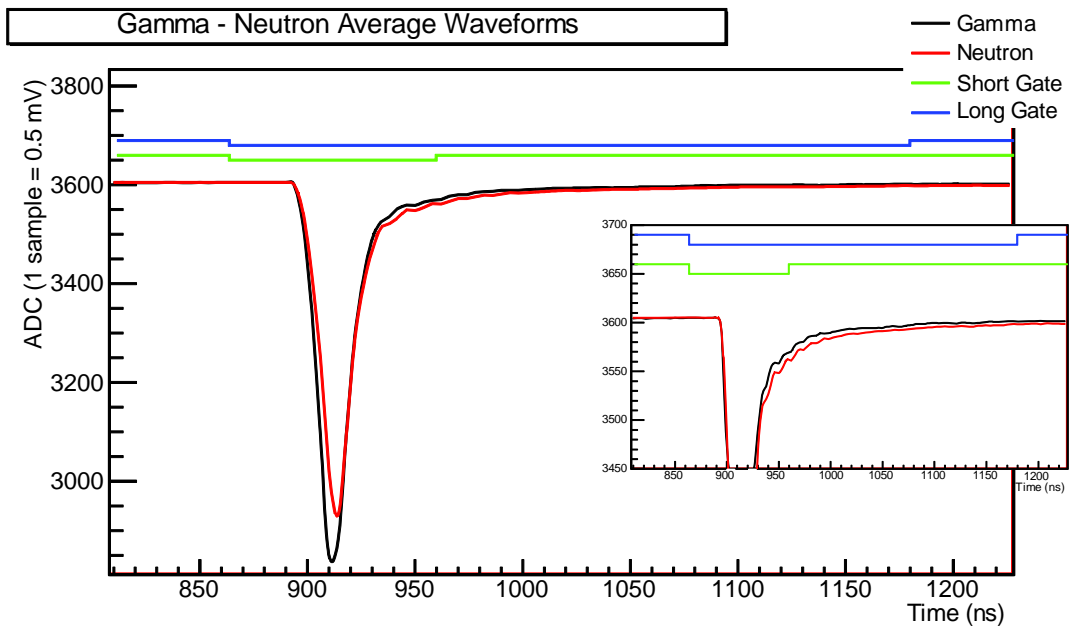
$$PSD = \frac{Q_{Long} - Q_{Short}}{Q_{Long}}$$

See [RD2] for an example of DPP-PSD application for gamma-neutron discrimination in liquid scintillators, and [RD3] for an application of the performances of the event selection based on the described algorithm.



**Note:** The description of the DPP-PSD system of this Manual is compliant with DPP-PSD firmware release **4.9\_131.11** for 720 series and DT5790, DPP-PSD firmware release **4.17\_136.14** for 725 series and 730 series, DPP-PSD firmware release **4.11\_132.8** and **4.12\_132.33** for 751 series<sup>2</sup>, and DPP-PSD Control Software release **1.3.5** For future release compatibility check in the firmware and software revision history files.

<sup>2</sup> Refer to the document “x751 Family DPP-PSD Firmware Compatibility” for additional details.



**Fig. 1.1:** Plot of typical Gamma-Neutron waveforms

Desktop Digitizers(*)	Description	Product Code
DT5720B	4 Ch. 12 bit 250 MS/s Digitizer: 1.25MS/ch, C20, SE	WDT5720BXAAA
DT5720C	2 Ch. 12 bit 250 MS/s Digitizer: 1.25MS/ch, C20, SE	WDT5720CXAAA
DT5720D	4 Ch. 12 bit 250 MS/s Digitizer: 10MS/ch, C20, SE	WDT5720DXAAA
DT5720E	2 Ch. 12 bit 250 MS/s Digitizer: 10MS/ch, C20, SE	WDT5720EXAAA
DT5725	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE	WDT5725XAAAA
DT5725B	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE	WDT5725BXAAA
DT5730	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE	WDT5730XAAAA
DT5730B	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE	WDT5730BXAAA
DT5751	2/4 Ch. 10 bit 2/1 GS/s Digitizer: 1.8/3.6MS/ch, EP3C16, SE	WDT5751XAAAA
DT5790N	2 Ch. 12 bit 250Ms/digitizer with 2HV ch -4kV/3mA for PSD	WDT5790XNAAA
DT5790M	2 Ch. 12 bit 250Ms/digitizer with 1HV ch +4kV/3mA 1HV ch -4kV/3mA for PSD	WDT5790XMAAA
DT5790P	2 Ch. 12 bit 250Ms/digitizer with 2HV ch +4kV/3mA for PSD	WDT5790XPAAA
NIM Digitizers(*)	Description	Product Code
N6720B	4 Ch. 12 bit 250 MS/s Digitizer: 1.25MS/ch, C20, SE	WN6720BXAAAA
N6720C	2 Ch. 12 bit 250 MS/s Digitizer: 1.25MS/ch, C20, SE	WN6720CXAAAA
N6720D	4 Ch. 12 bit 250 MS/s Digitizer: 10MS/ch, C20, SE	WN6720DXAAAA
N6720E	2 Ch. 12 bit 250 MS/s Digitizer: 10MS/ch, C20, SE	WN6720EXAAAA
N6725	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE	WN6725XAAAAA
N6725B	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE	WN6725BXAAAAA
N6730	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE	WN6730XAAAAA
N6730B	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE	WN6730BXAAAAA
N6751	2/4 Ch. 10 bit 2/1 GS/s Digitizer: 1.8/3.6MS/ch, EP3C16, SE	WN6751XAAAAA
N6751C	2/4 Ch. 10 bit 2/1 GS/s Digitizer: 14.4/28.8 MS/ch, EP3C16, SE	WN6751CXAAAAA
VME Digitizers(*)	Description	Product Code
V1720E	8 Ch. 12 bit 250 MS/s Digitizer: 1.25MS/ch, C20, SE	WV1720EXAAAAA
V1720F	8 Ch. 12 bit 250 MS/s Digitizer: 1.25MS/ch, C20, DIFF	WV1720FXAAAAA
V1720G	8 Ch. 12 bit 250 MS/s Digitizer: 10MS/ch, C20, SE	WV1720GXAAAAA
V1725	16 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE	WV1725XAAAAA
V1725B	16 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE	WV1725BXAAAAA
V1725C	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE	WV1725CXAAAAA
V1725D	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE	WV1725DXAAAAA
V1730	16 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE	WV1730XAAAAA
V1730B	16 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE	WV1730BXAAAAA
V1730C	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE	WV1730CXAAAAA



V1730D	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE	WV1730DXAAAA
V1751	4/8 Ch. 10 bit 2/1 GS/s Digitizer: 1.8/3.6MS/ch, EP3C16, SE	WV1751XAAAAA
V1751B	4/8 Ch. 10 bit 2/1 GS/s Digitizer: 1.8/3.6MS/ch, EP3C16, DIFF	WV1751BXAAAA
V1751C	4/8 Ch. 10 bit 2/1 GS/s Digitizer: 14.4/28.8MS/ch, EP3C16, SE	WV1751CXAAAA
VX1720E	8 Ch. 12 bit 250 MS/s Digitizer: 1.25MS/ch, C20, SE	WVX1720EXAAA
VX1720F	8 Ch. 12 bit 250 MS/s Digitizer: 1.25MS/ch, C20, DIFF	WVX1720FXAAA
VX1725	16 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE	WVX1725XAAAA
VX1725B	16 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE	WVX1725BXAAA
VX1725C	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE	WVX1725CXAAA
VX1725D	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE	WVX1725DXAAA
VX1730	16 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE	WVX1730XAAAA
VX1730B	16 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE	WVX1730BXAAA
VX1730C	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE	WVX1730CXAAA
VX1730D	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE	WVX1730DXAAA
VX1751	4/8 Ch. 10 bit 2/1 GS/s Digitizer: 1.8/3.6MS/ch, EP3C16, SE	WVX1751XAAAA
VX1751B	4/8 Ch. 10 bit 2/1 GS/s Digitizer: 1.8/3.6MS/ch, EP3C16, DIFF	WVX1751BXAAA
VX1751C	4/8 Ch. 10 bit 2/1 GS/s Digitizer: 14.4/28.8MS/ch, EP3C16, SE	WVX1751CXAAA
<b>DPP Firmware(*)</b>	<b>Description</b>	<b>Product Code</b>
DPP-PSD	Digital Pulse Processing for Pulse Shape Discrimination (x720 and DT5790)	WFWDPPNGAA20
DPP-PSD	Digital Pulse Processing for Pulse Shape Discrimination (x725 and x730)	WFWDPPNGAA30
DPP-PSD	Digital Pulse Processing for Pulse Shape Discrimination (x751)	WFWDPPNGAA51

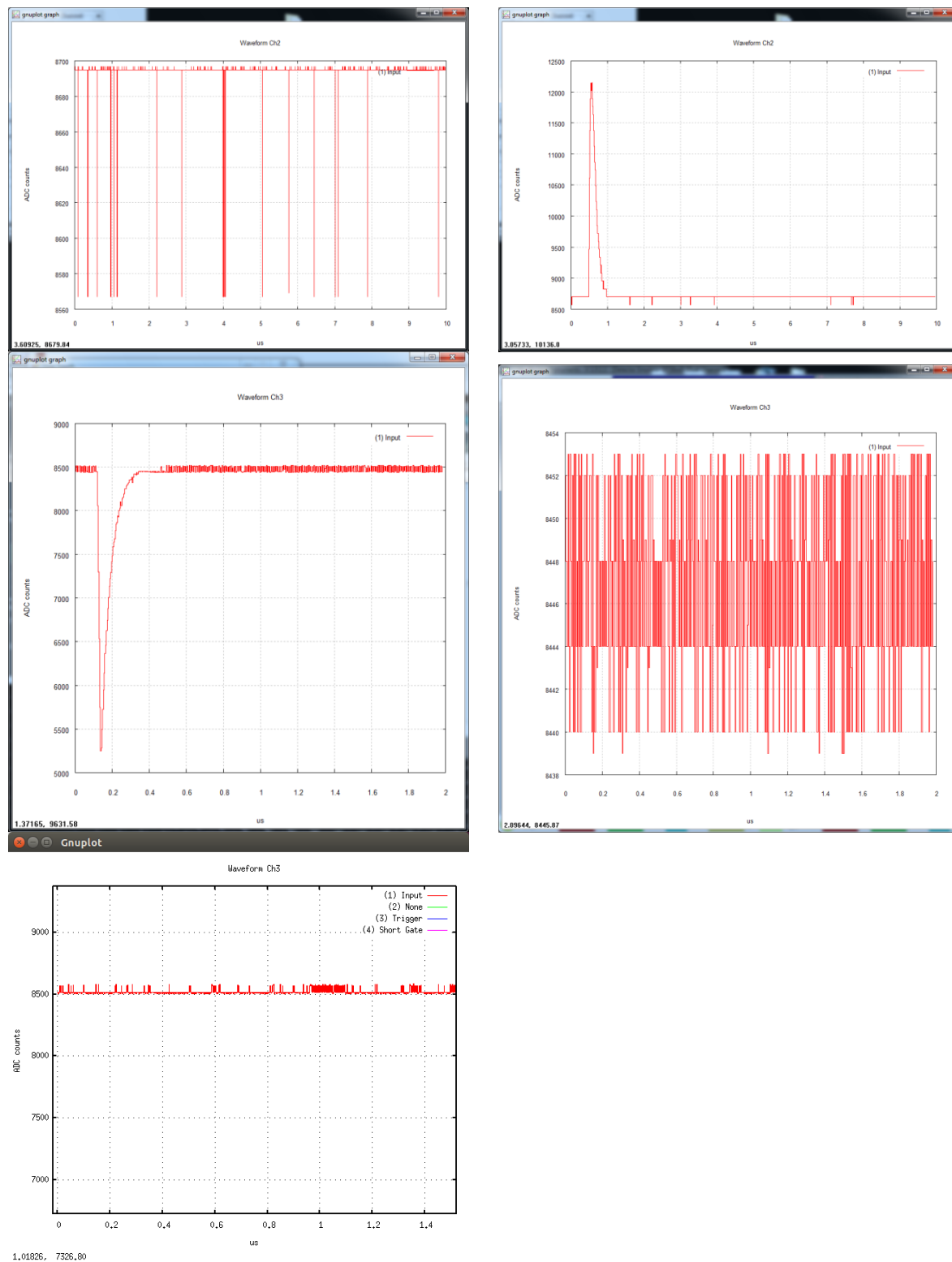
**Tab. 1.1:** Supported CAEN digitizers for DPP-PSD firmware

(\*) For accessories and customizations related to digitizers and for multiple DPP-PSD license packs, refer to the board User Manual or have a look at the board page on CAEN web site: [www.caen.it](http://www.caen.it)

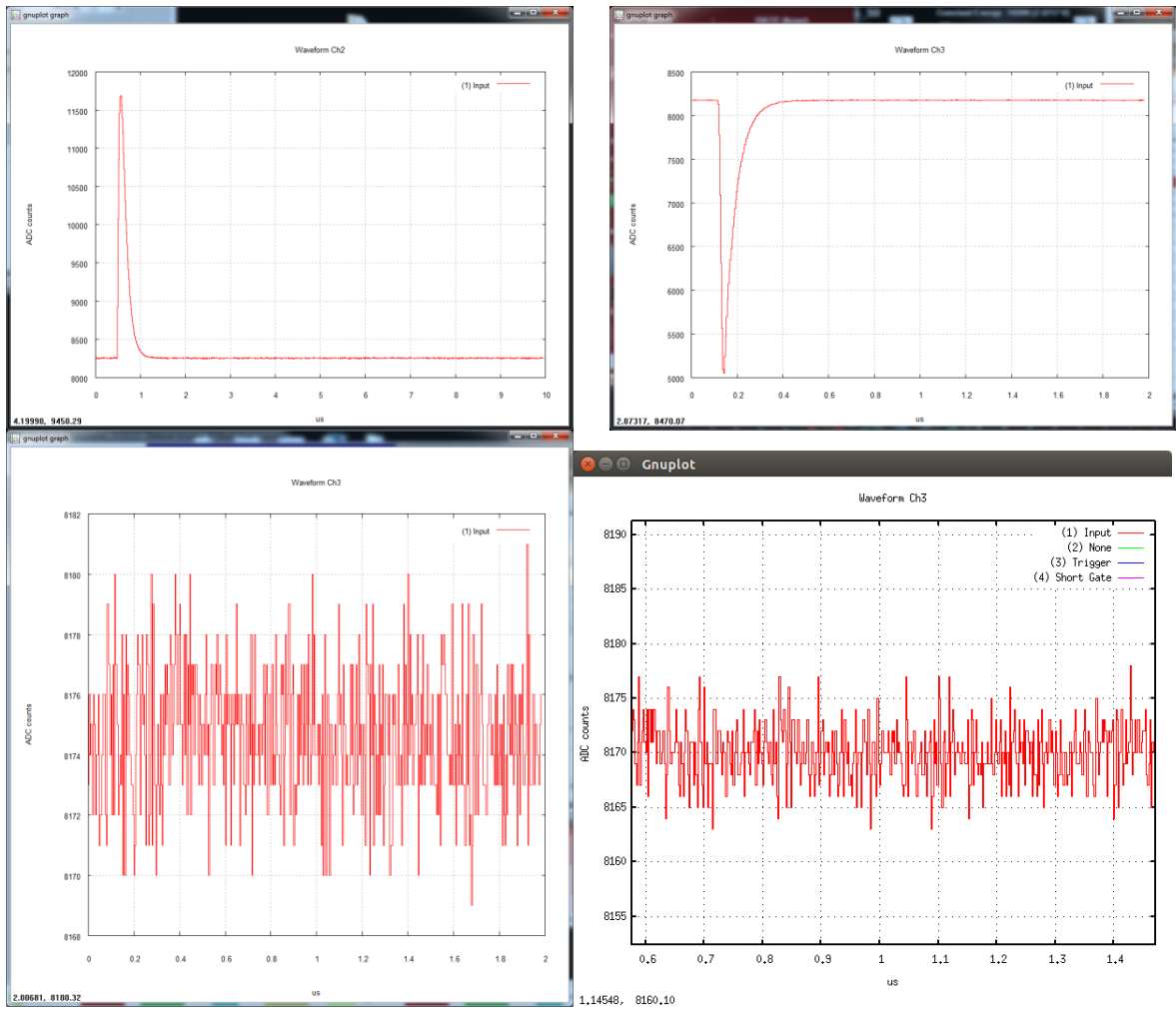
## Notes on ADC Calibration (725, 730, and 751 series)

In case of 725, 730, and 751 series it is very important to perform the ADCs calibration before starting the acquisition, thus ensuring to have the same performances during the overall acquisition. Indeed the ADCs performances are strongly dependent on the chip temperature itself. When the temperature changes it is required to perform the ADCs calibration, as for example after the power on, when the ADCs warm up.

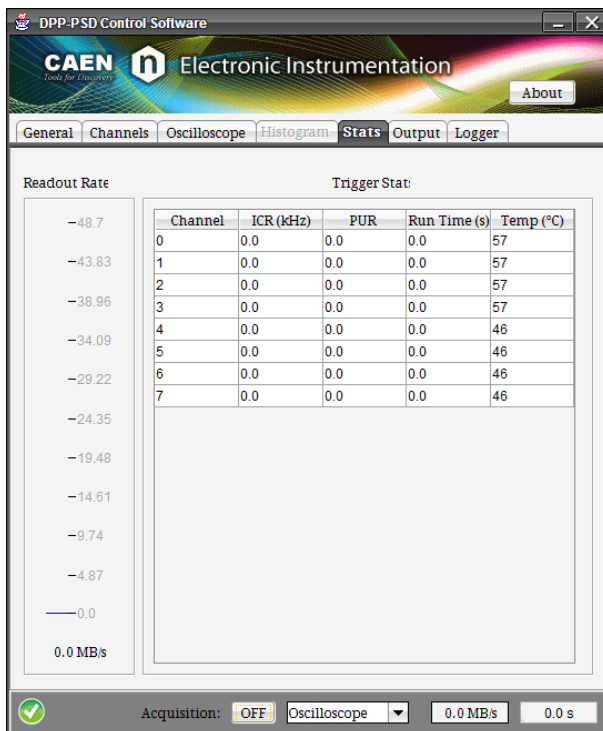
In case the calibration is not performed the user might see the following waveforms:



Conversely when the ADC calibration has been made, the waveform should appear like in the following pictures:



Before performing the ADC calibration, the user must check the ADC temperature. This can be done via software GUI, under the “STATS” tab. The column “Temperature” is active only when the acquisition is OFF. Alternatively access to register 0x1nA8 (ADC Temperature) **[RD13]**.



Once the temperature is stable perform the ADC calibration, via software GUI under the tab “GENERAL” press “Calibrate”. Alternatively, via register access to address 0x809C (725 and 730 series) and bit[1] of register 0x1n9C (751 series).



## 2. Principle of Operation

The figure below shows the functional block diagram of the DPP-PSD firmware:

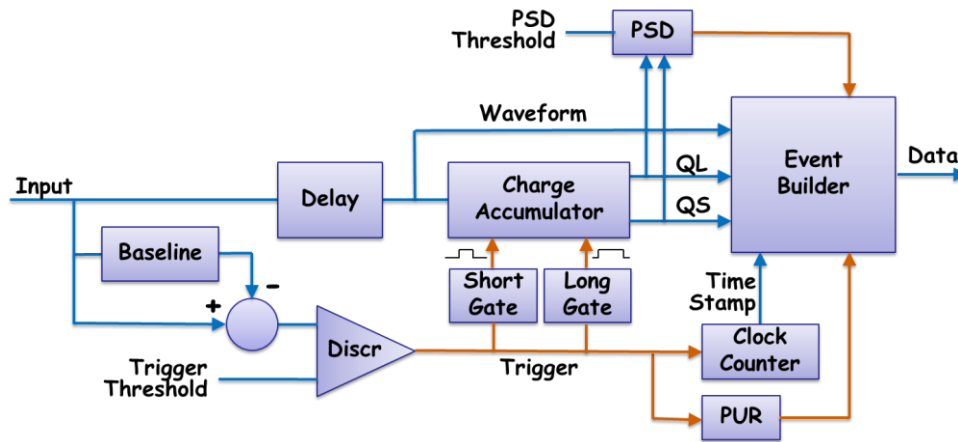


Fig. 2.1: Functional Block Diagram of the DPP-PSD

The aim of the DPP-PSD firmware is to calculate the two charges  $Q_{short}$  and  $Q_{long}$ , performing a double gate integration of the input pulse. The ratio between the charge of the tail (slow component) and the total charge gives the  $PSD$  parameter used for the gamma-neutron discrimination:

$$PSD = \frac{Q_{LONG} - Q_{SHORT}}{Q_{LONG}}$$

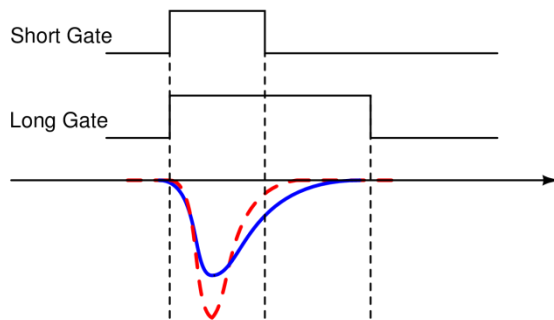
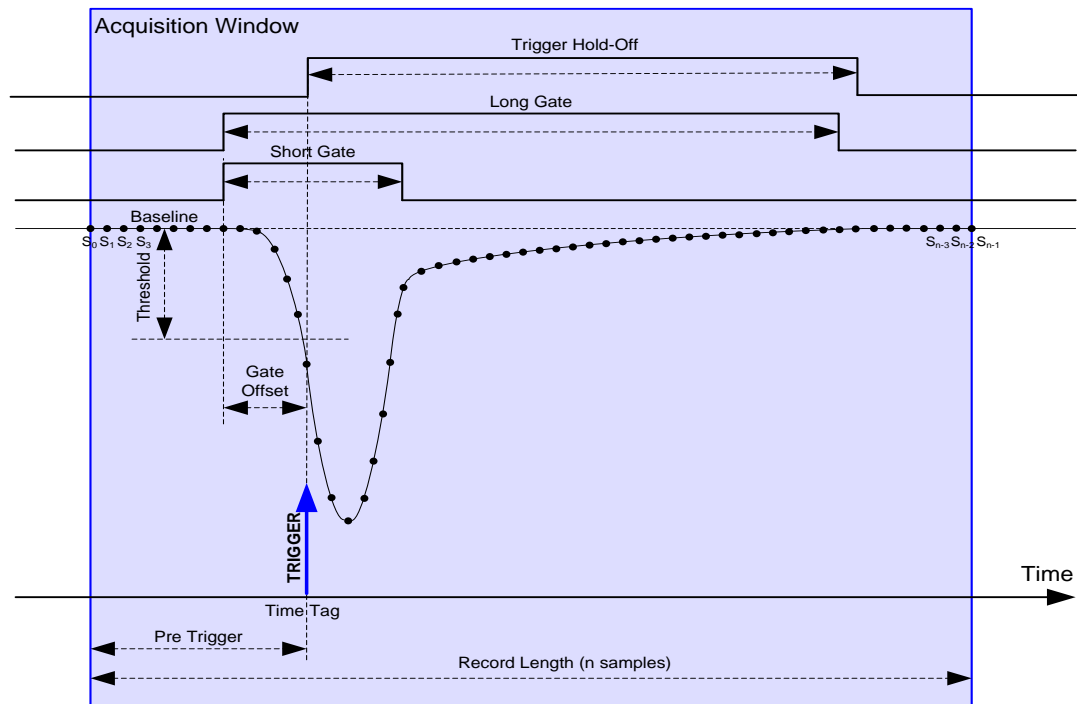


Fig. 2.2: Long and short gate graphic position with respect to a couple of input pulses. The blue pulse has a longer tail than the red one

The main operations of the DPP-PSD firmware can be summarized as follows:

- receive an input signal directly from the detector and digitize it continuously. It is possible to adjust the dynamic range with a programmable DC offset to exploit the full dynamics of the digitizer;
- the algorithm continuously calculates the *baseline* of the input signal by averaging the samples belonging to a moving window of programmable size (see Sect. **Baseline**). The baseline is subtracted from the input signal, giving  $input\_sub = input - baseline$  (Digital Baseline Restorer);
- the  $input\_sub$  value is compared with the value of the trigger threshold and the event is selected as soon as the  $input\_sub$  signal crosses the threshold (see Fig. 2.3). In case of 725, 730, and 751 series, the digital CFD (Constant Fraction Discriminator) zero crossing can be used for a better timing information (refer to Sect. **CFD implementation for 725, 730, and 751 series**). Once the event is selected a local trigger is generated (refer to Sect. **DPP-PSD trigger management** for further details);

- at the trigger fire, the signal is delayed by a programmable number of samples (corresponding to the “pre-trigger” value in ns).to be able to integrate the pulse before the trigger (“Gate Offset”). The gates for charge integration are then generated and are received by the charge accumulator before the signal. While the gates are active the baseline remains frozen to the last averaged value and its value is used as charge integration reference. **Fig. 2.3** summarizes all the DPP-PSD parameters;
- for the whole duration of a programmable “trigger hold-off” value, other trigger signals are inhibited. It is recommended to set a trigger hold-off value compatible with the signal width. The baseline remains frozen for the whole trigger hold-off duration. For 751 series the baseline remains frozen for a longer time;



**Fig. 2.3:** Diagram summarizing the DPP-PSD parameters. The trigger fires as soon as the signal crosses the threshold value. Long Gate, Short Gate, Gate Offset, Pre-Trigger, Trigger Hold-Off, and Record Length are also shown for one acquisition window

- the trigger enables the event building, that includes the waveforms (i.e. the raw samples) of the input, the trigger time stamp, the baseline, and the charge integrated within the gates. After that the system gets ready for a new event;
- the event data is saved into a memory buffer. The Control Software automatically optimize both the number of events inside the buffer, and the number of total buffers that the memory is divided in. The user can also choose to set these parameters by hand. If the buffer contains only one event, that buffer becomes immediately available for the readout and the acquisition continues into another buffer. If more events are written in one buffer, only when the buffer is complete those events become available for the readout;
- the software can then plot the signal waveforms for debugging and parameters adjustment, as well as plot energy spectrum of  $Q_{long}$  and timing distribution, the 2-D scatter plot of the  $PSD$  parameter (see the definition in the **Principle of Operation** section) vs  $Q_{long}$  energy is also available;
- finally output files (list and waveform) can be generated in different formats suitable for external spectroscopy analysis software tools. Energy, time, and 2D spectra are not managed onboard but they can be generated and saved by the DPP-PSD Control Software.

Further on the above list there are additional features that can be performed by the DPP-PSD firmware, listed below:

- automatically discard events according to a programmable  $PSD$  threshold (see section **Online PSD selection**);
- detect pile-up conditions (See **Appendix A** for 751 series **Appendix B** for 720, 725, and 730 series;
- implement coincidences between channels of the same board (**[RD4]**) as well as among channels of different boards through the LVDS I/O connectors (VME only), external trigger, and external modules.

## Baseline

The baseline calculation is an important feature of the DPP-PSD firmware, since its value is used as a reference value for the charge integration of the input pulses. Moreover, most of the DPP parameters are related to the baseline value. This paragraph describes in detail how the baseline calculation works.

The user can choose to set a fixed value for the baseline, or to let the DPP firmware calculate it dynamically. In the first case the user must set the *baseline value in LSB units*. This value remains fixed for the entire acquisition run. In the latter case the firmware dynamically evaluates the baseline as the mean value of  $N$  points inside a moving time window. The user can choose the  $N$  value among 8, 32, and 128 for 720 series (DT5790); 8, 16, 32, 64, 128, 256, and 512 for 751 series, and 16, 64, 256, 1024 for 725 and 730 series.

The baseline is then frozen from few clocks before the gates start, up to the end of the maximum value between the long gate and the trigger hold-off. For 751 series the freeze lasts some trigger clocks more than this maximum value. After that the baseline restarts again its calculation considering in the mean value also the points before the freeze. This allows to have almost no dead-time due to the baseline calculation.



**Note:** In case of 725 and 730 series the user can set the time before the gate for the baseline freeze start, through register 0x1nD8 (default value = 16 ns). This option can be useful when the gate does not cover the beginning of the signal, and the baseline becomes distorted. The baseline freeze lasts for the maximum value among the long gate, the trigger hold-off, and the over-threshold signals.



**Note:** In case of 725 and 730 series the baseline remains frozen also on events clipping in the gate (saturation) and on opposite polarity. Refer to Sect. **Veto (725 and 730 series only)** for additional details.

Fig. 2.4 shows how the baseline calculation and freeze work. The trigger threshold dynamically follows the baseline variations.

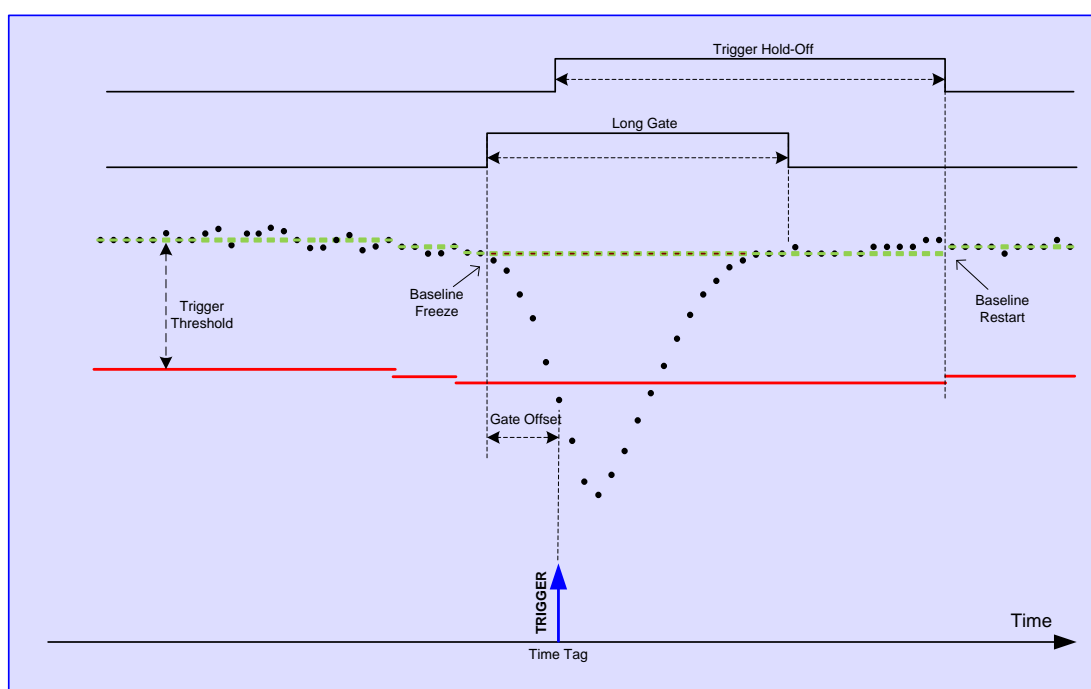


Fig. 2.4: Baseline calculation as managed by the DPP-PSD algorithm

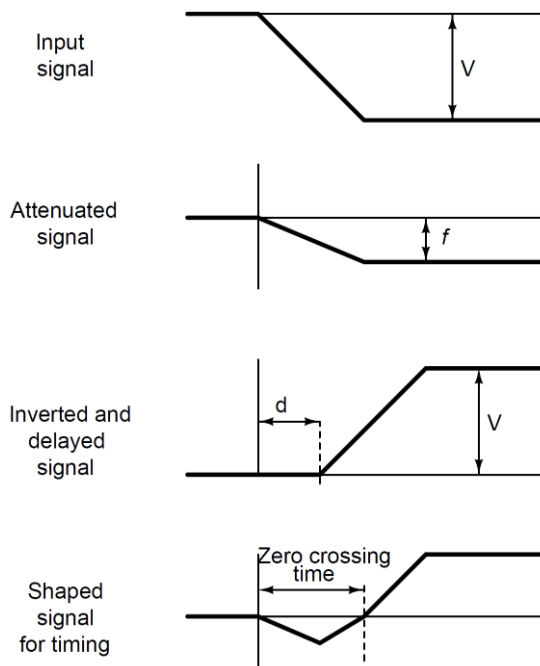
## CFD implementation for 725, 730, and 751 series



**Note:** The CFD for 751 series is supported from DPP-PSD firmware release **greater than 132.32**.

Using analog signals the Time Stamp determination is traditionally done with CFD (Constant Fraction Discriminator) modules. This technique sets the time stamp of a pulse to the time when the amplitude reaches a fixed fraction of the full amplitude. The DPP-PSD firmware for 725, 730, and 751 families intends to exploit the advantages of a CFD technique using a digital sampling device. The standard implementation for 720 and DT5790 series DPP-PSD firmware is based on the leading edge trigger, that may suffer from amplitude walk issues. Conversely triggering on a constant fraction of the input may reduce this issue since it is independent from the amplitude pulse. On the other side a simple linear interpolation between two points can solve the problem of the sampling clock granularity, thus improving the timing resolution.

The digital CFD signal has been implemented in the classical way. The input waveform is attenuated by a factor  $f$  equal to the desired timing fraction of full amplitude, then the signal is inverted and delayed by a time  $d$  equal to the time it takes the pulse to rise from the constant fraction level to the pulse peak; the latest two signals are summed to produce a bipolar pulse, the CFD, and its zero crossing – corresponding to the fraction  $f$  of the input pulse – is taken as the time trigger (see Fig. 2.5).



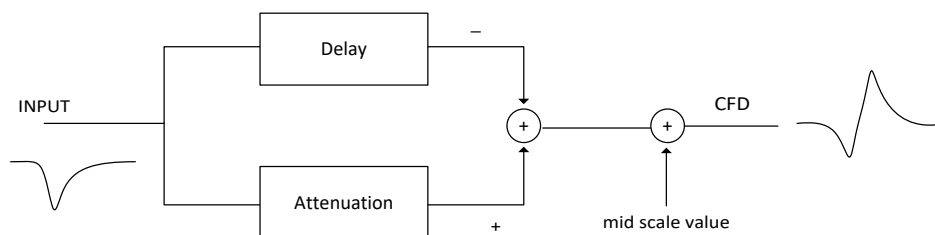
**Fig. 2.5:** Classical implementation of the Constant Fraction Discriminator. The input signal is first attenuated by a factor  $f$ , then inverted and delayed. The resulting signal has its zero crossing corresponding to the set fraction  $f$

The digital implementation of the CFD is shown in Fig. 2.6. The input sample is split into two paths: the first performs the delay in steps of the sampling clock (4 ns in case of 725, 2 ns in case of 730 series, 1 ns in case of 751 series), the second performs the attenuation. Possible choices of attenuation are: 25%, 50%, 75%, and 100% (i.e. no attenuation) with respect to the input amplitude. The CFD signal is referred to the mid-scale of the dynamics, i.e. channel 8192 in case of 725 and 730 series, channel 512 for 751 series.



**Note:** Since the gate position is referred to the trigger, in case of CFD it is necessary to increase the gate offset with respect to the same acquisition with LED (Leading Edge) trigger.





**Fig. 2.6:** Implementation of the digital CFD in the DPP-PSD firmware of 725, 730, and 751 series. Mid-scale value corresponds to 8192 in case of 725 and 730 series, 512 for 751 series.

To enable the CFD discrimination the user must:

- set to 1 bit[6] of register 0x1n80.

Then it is possible to set the values of the delay and of the fraction, writing register 0x1n3C “CFD Settings”:

- bits[9:8] correspond to the four options of the fraction:
  - 00: fraction = 25%;
  - 01: fraction = 50%;
  - 10: fraction = 75%;
  - 11: fraction = 100%;
- bits[7:0] correspond to the CFD delay value.
- bits[11:10] in case of 725 and 730, decide which of the n-th sample before and after the zero crossing are used for the interpolation. Options are:
  - 00: the sample before and after the zero crossing;
  - 01: second sample before and after the zero crossing;
  - 10: third sample before and after the zero crossing;
  - 11: fourth sample before and after the zero crossing.
- bits[12:10] in case of 751 series, decide which of the n-th sample before and after the zero crossing are used for the interpolation. Options are:
  - 000: the sample before and after the zero crossing;
  - 001: second sample before and after the zero crossing;
  - 010: third sample before and after the zero crossing;
  - 011: fourth sample before and after the zero crossing;
  - 100: fifth sample before and after the zero crossing;
  - 101: sixth sample before and after the zero crossing;
  - 110: seventh sample before and after the zero crossing;
  - 111: eighth sample before and after the zero crossing.



**Note:** The interpolation points defined in bits[11:10] (bits[12:10]) correspond to the n-th sample before and after the zero crossing that are used for the linear interpolation.



**Note:** The interpolation points can be used in case of Leading Edge Discrimination too.

A typical signal from CFD is shown in **Fig. 2.7**, where the red points are the digital samples. The Sample Before the Zero Crossing (SBZC) and the Sample After the Zero Crossing (SAZC) are the samples before and after the zero crossing. In case bits[11:10] (bits[12:10]) of register 0x1n3C are enabled, the SBZC and SAZC are defined as the n-th sample before and after the zero crossing, according to the options described above and in the Register Description Manual **[RD13] [RD19]**.

The SBZC corresponds to the Coarse Time Stamp ( $T_{coarse}$ ), that is the trigger time stamp as evaluated by the standard PSD algorithm.

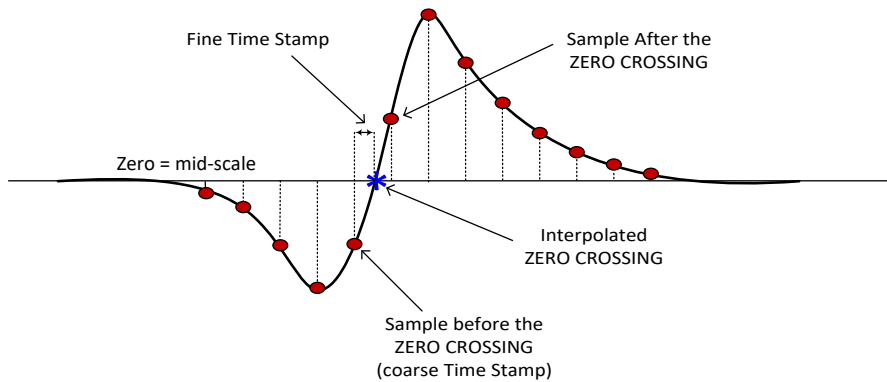
The value of the Fine Time Stamp  $T_{fine}$  (see **Fig. 2.7**) is calculated as the linear interpolation of the SBZC and the SAZC according to the formula:

$$T_{fine} = \frac{midScale - SBZC}{SAZC - SBZC} \cdot Tsampl$$

where *midScale* corresponds to 8192 in case of 725 and 730 series, and 512 for 751 series, and *Tsampl* is the sampling period of the specific series (4 ns in case of 725, 2 ns for 730, 1 ns in case of 751).

The “Interpolated Zero Crossing” (ZC) then corresponds to the sum of the Coarse Time Stamp and the Fine Time Stamp.

$$ZC = T_{coarse} + T_{fine}$$



**Fig. 2.7:** A typical CFD signal. Red points are the digital samples. The Sample Before the Zero Crossing (SBZC) and the Sample After the Zero Crossing (SAZC) are the samples before and after the zero crossing. The SBZC corresponds to the Coarse Time Stamp. The algorithm can also evaluate the Fine Time Stamp, and the corresponding time will be the sum of the SBZC and the Fine Time Stamp

The user can choose which information from the CFD algorithm is saved in the event structure for the readout.

In addition it is also possible to reserve 16 bit for the time stamp extension. The extra 16 bit must be read as the most significant bits of the time stamp. The extended time stamp is therefore expressed as a 31+16 = 47 bit number for 725 and 730 series, and 32+16=48 bit number for 751 series.

Both the CFD and the time stamp extension are written in an additional word to the event format, which is called “EXTRAS” word (see section **Channel Aggregate Data Format for 725 and 730 series**, and **Channel Aggregate Data Format for 751 series**). The EXTRAS format varies according to the user settings. The specific registers that must be enabled are:

- bit[6] of register 0x1n80 set to 1 to enable the event selection via CFD;
- bit[17] of register 0x8000 set to 1 to enable the EXTRAS word recording;
- bits[10:8] of register 0x1n84 (where n is the channel number) to select the specific information saved into the EXTRAS word. Options are summarized in **Tab. 2.1** for 725 and 730 series, and **Tab. 2.2** for 751 series.

Condition	EXTRAS word format (725 and 730 series)																															
no extras enabled	Not available																															
EX = 000	<div><div><div>313029282726252423222120191817161514131211109876543210</div><div>Extended Time StampBaseline * 4</div></div><div>BITEXTRAS</div></div>																															
EX = 001	<div><div><div>313029282726252423222120191817161514131211109876543210</div><div>Extended Time StampFlags</div></div><div>BITEXTRAS</div></div>																															
EX = 010	<div><div><div>313029282726252423222120191817161514131211109876543210</div><div>Extended Time StampFlagsFine Time Stamp</div></div><div>BITEXTRAS</div></div>																															
EX = 100	<div><div><div>313029282726252423222120191817161514131211109876543210</div><div>Lost Trigger CounterTotal Trigger Counter</div></div><div>BITEXTRAS</div></div>																															
EX = 101	<div><div><div>313029282726252423222120191817161514131211109876543210</div><div>Sample After Zero CrossingSample Before Zero Crossing</div></div><div>BITEXTRAS</div></div>																															
EX = 111	Fixed value = 0x12345678 (debug purposes only)																															

**Tab. 2.1:** Summary of the options for the EXTRAS word write in the 725 and 730 series. Extended Time Stamp are 16 bits that become the most significant bits of the time stamp value; “Flags” identifies trigger lost and saturation conditions, Fine Time Stamp, Sample After and Sample Negative Zero Crossing values come from the CFD calculation.

Where “Flags” are:

bit[15]: Trigger Lost (the first event after a trigger lost has this flag set);

bit[14]: Over-range (identifies an event saturating inside the gate - clipping);

bit[13]: 1024 trigger counted (every 1024 counted events this flag is high);

bit[12]: N lost trigger counted (every N counted lost events this flag is high, where N is set from bits[17:16] of register 0x1n84).



**Note:** In case bits[11:10] (bits[12:10]) of register 0x1n3C are enabled, the SBZC and SAZC are defined as the n-th sample before and after the zero crossing, according to the options described above or in the Register Description **[RD13]** **[RD19]**.

Condition	EXTRAS word format (751 series)																																
no extras enabled	Not available																																
EX = 000	<div> <div>31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</div> <div>Extended Time StampBaseline * 8</div> </div>																																
	<div> <div>31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</div> <div>Extended Time StampFlags</div> </div>																																
EX = 001	<div> <div>31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</div> <div>Extended Time StampFlags</div> </div>																																
	<div> <div>31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</div> <div>Extended Time StampFlagsFine Time Stamp</div> </div>																																
EX = 010	<div> <div>31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</div> <div>Extended Time StampFlagsFine Time Stamp</div> </div>																																
	<div> <div>31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</div> <div>FlagsFine Time StampBaseline * 8</div> </div>																																
EX = 011	<div> <div>31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</div> <div>FlagsFine Time StampBaseline * 8</div> </div>																																
	<div> <div>31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</div> <div>DMPPSample Before Zero CrossingSample After Zero CrossingMid Scale Value/Threshold</div> </div>																																
EX = 101	<div> <div>31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</div> <div>DMPPSample Before Zero CrossingSample After Zero CrossingMid Scale Value/Threshold</div> </div>																																
EX = 111	Fixed value = 0x12345678 (debug purposes only)																																

**Tab. 2.2:** Summary of the options for the EXTRAS word write for the 751 series. Extended Time Stamp are 16 bits that become the most significant bits for the time stamp value; Fine Time Stamp, Sample Before and After Zero Crossing values come from the CFD calculation; TR identifies whether the CDF or LED option are selected; PP corresponds to the rising/falling edge trigger; Mid-Scale Value/Threshold correspond to 512 in case of CFD, or the Trigger Threshold value in case of LED option.

Where:

“Flags” are currently N.A.,

DM (Discrimination Mode) identifies whether the CDF or LED option are selected (LED = 0; CDF = 1);

PP (Pulse Polarity) identifies the pulse polarity and the trigger on the rising/falling edge (rising = 0; falling = 1);

Mid-Scale Value/Threshold correspond to 512 in case of CFD, or the Trigger Threshold value in case of LED.

In case of 725 and 730 series, the trigger time tag (“TTT”) is a 31 bit number (refer to the word “TRIGGER TIME TAG” of section **Channel Aggregate Data Format for 725 and 730 series**), Tsampl = 4 ns for 725, Tsampl = 2 ns for 730 series, the final time stamp can be equal to one of the possibilities reported in **Tab. 2.3**, according to the selected choice (EXTRAS is a 32 bit word).

Condition	Time Stamp (725 and 730 series)	Number of bits	Step	Roll Over
no extras enabled	Time Stamp = $T_{coarse} \cdot Tsampl$ , where $T_{coarse} = TTT$	31	$Tsampl$	$2^{31} \cdot Tsampl$
EX = 000/001	Time Stamp = $T_{coarse} \cdot Tsampl$ , where $T_{coarse} = EXTRAS[31:16] + TTT$	47	$Tsampl$	$2^{47} \cdot Tsampl$
EX = 010	Time Stamp = $T_{coarse} \cdot Tsampl + T_{fine}$ , where $T_{coarse} = EXTRAS[31:16] + TTT$ , and $T_{fine} = \frac{EXTRAS[9:0]}{1024} \cdot Tsampl$	47 + 10	$\sim Tsampl / 1000$	$2^{47} \cdot Tsampl$
EX = 101	Time Stamp = $T_{coarse} \cdot Tsampl + T_{fine}$ , where $T_{coarse} = TTT$ , and $T_{fine} = \frac{MidScale-SBZC}{SAZC-SBZC} \cdot Tsampl$	31 + 10	$\sim Tsampl / 1000$	$2^{31} \cdot Tsampl$

**Tab. 2.3:** Summary of the CFD options for 725 and 730 series, where “Time Stamp” is the timing information of a single event,  $T_{coarse}$  corresponds to the SBZC (Fig. 2.7), TTT is the 31-bit TRIGGER TIME TAG word of the event structure (refer to section **Channel Aggregate Data Format for 725 and 730 series**), EXTRAS is the word of event structure, and  $T_{fine}$  corresponds to the difference between the ZC and the SBZC (Fig. 2.7)

In case of 751 series, the trigger time tag (“TTT”) is a 32 bit number (refer to the word “TRIGGER TIME TAG” of section **Channel Aggregate Data Format for 751 series**),  $Tsampl = 1$  ns, the final time stamp can be equal to one of the possibilities reported in **Tab. 2.4**, according to the selected choice (EXTRAS is a 32 bit word).

Condition	Time Stamp (751 series)	Number of bits	Step	Roll Over
no extras enabled	Time Stamp = $T_{coarse} \cdot Tsampl$ , where $T_{coarse} = TTT$	32	$Tsampl$	$2^{32} \cdot Tsampl$
EX = 000/001	Time Stamp = $T_{coarse} \cdot Tsampl$ , where $T_{coarse} = EXTRAS[31:16] + TTT$	48	$Tsampl$	$2^{48} \cdot Tsampl$
EX = 010	Time Stamp = $T_{coarse} \cdot Tsampl + T_{fine}$ , where $T_{coarse} = EXTRAS[31:16] + TTT$ , and $T_{fine} = \frac{EXTRAS[9:0]}{1024} \cdot Tsampl$	48 + 10	$\sim Tsampl / 1000$	$2^{48} \cdot Tsampl$
EX = 011	Time Stamp = $T_{coarse} \cdot Tsampl + T_{fine}$ , where $T_{coarse} = TTT$ , and $T_{fine} = \frac{EXTRAS[9:0]}{1024} \cdot Tsampl$	32 + 10	$\sim Tsampl / 1000$	$2^{32} \cdot Tsampl$
EX = 101	Time Stamp = $T_{coarse} \cdot Tsampl + T_{fine}$ , where $T_{coarse} = TTT$ , and $T_{fine} = \frac{midScale-SBZC}{SAZC-SBZC} \cdot Tsampl$	32 + 10	$\sim Tsampl / 1000$	$2^{32} \cdot Tsampl$

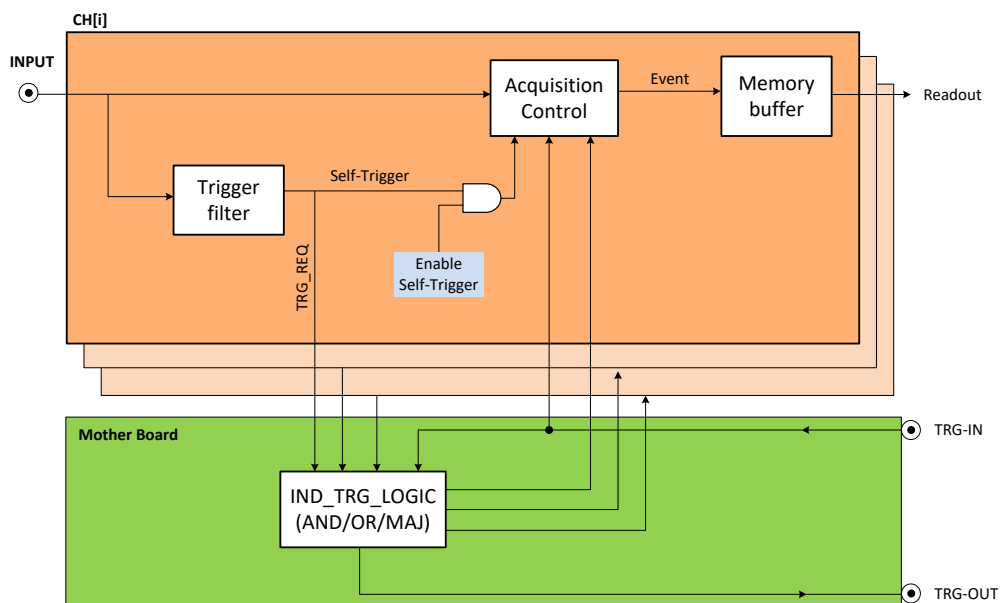
**Tab. 2.4:** Summary of the CFD options for 751 series, where “Time Stamp” is the timing information of a single event,  $T_{coarse}$  corresponds to the SBZC (Fig. 2.7), TTT is the 32-bit TRIGGER TIME TAG word of the event structure (refer to section **Channel Aggregate Data Format for 751 series**), EXTRAS is the word of event structure, and  $T_{fine}$  corresponds to the difference between the ZC and the SBZC (Fig. 2.7)

Refer to sections **Channel Aggregate Data Format for 725 and 730 series** and **Channel Aggregate Data Format for 751 series** for further details.

## DPP-PSD trigger management

The DPP-PSD firmware allows for several way of trigger generation:

- 1) each channel can “self-trigger” on its own input signal when the input crosses a programmable threshold, or through the CFD (725, 730, and 751 only). The self-trigger works on each channel independently from the other channels;
- 2) each channel triggers independently, then only those events satisfying programmable conditions are saved. Referring to **Fig. 2.8**: the IND\_TRG\_LOGIC (Individual Trigger Logic) can combine (AND/OR/MAJ) the trigger request (TRG\_REQ) from each self-trigger. When the logic condition is met, a trigger validation (TRG\_VAL) is sent back to each channel individually to enable the acquisition of that event. Events not receiving a TRG\_VAL signal are discarded. This technique allows to make coincidence and anti-coincidence requests among different channels (refer to **[RD4]** for further details);
- 3) the board can accept an external trigger on the TRG IN connector. The external trigger can be used in OR logic operation with the channel self-trigger, or it can be used as a VETO to inhibit the individual self-trigger. If the self-trigger is disabled, the acquisition is managed by the external trigger only;
- 4) individual trigger and logic combination of self-trigger of different channels (AND/OR/MAJ) can be propagated through the TRG-OUT connector.



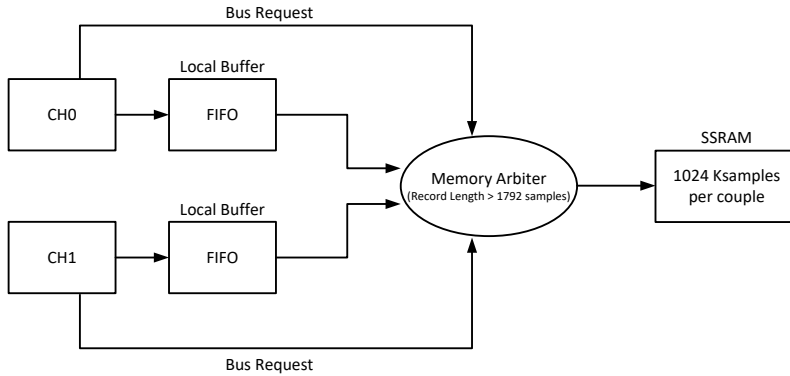
**Fig. 2.8:** Diagram showing the structure of the trigger management of the DPP-PSD firmware

In case of 725 and 730 series, each channel can manage the self-trigger independently but it shares the same memory buffer with the other channel of the couple (0-1, 2-3, etc.) (see **Fig. 2.9**). The user must take care in case he/she wants to acquire large waveform length. Indeed the DPP-PSD algorithm is meant to acquire small size events, typically time stamps and charges, and possibly small portions of the waveform for post-processing. In case of record length less than 1792 samples, each channel has enough memory in its local buffer to acquire events independently from the other channel. For record length greater than 1792 samples, the couple must use an external SRAM memory, and a memory arbiter decides in “fair mode” which event of the two channels is saved.



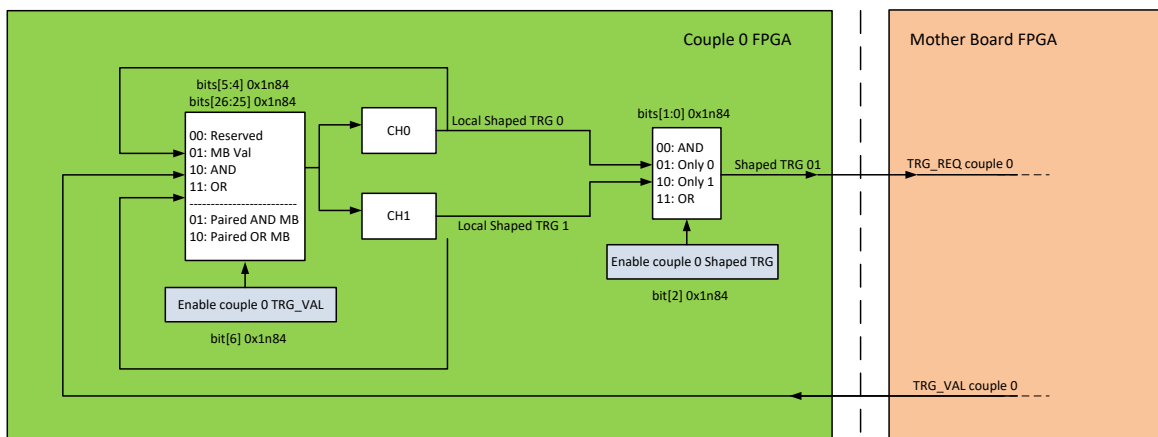
**Note:** In case of List mode, the Record Length is ignored.

Moreover the channels of the couple share the same TRG\_REQ for the coincidence logic. This means that it is not possible to set different coincidence logic among channels of two different couples (refer to **[RD4]** for further details). Anyway a great advantage comes in case of coincidences between channels of the same couple, which can be managed inside the channel FPGA, with no propagation to the mother board.



**Fig. 2.9:** Memory management of 725 and 730 series

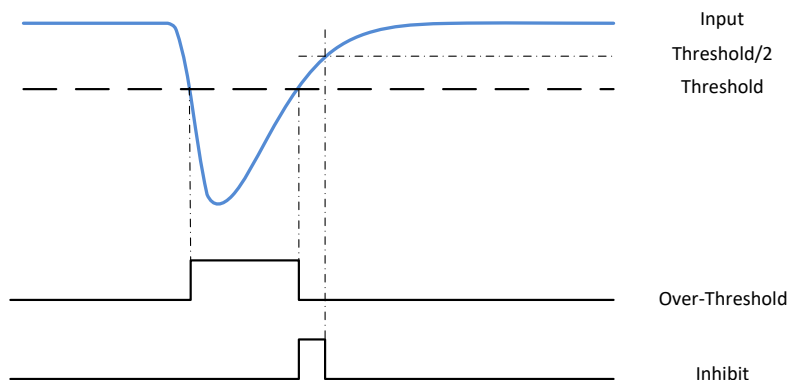
The TRG\_REQ from couple and the TRG\_VAL to the couple are managed by bits[6:0] of register 0x1n84, according to the bit scheme reported in **Fig. 2.10**.



**Fig. 2.10:** Local Trigger Management inside couple 0 of 725-730 digitizer series. Couple 0 is made of channel 0 and channel 1. The same applies for the other couples of the 725 and 730.

## Trigger Hysteresis (725 and 730 series only)

When the input signal is no more over-threshold, the trigger could fire again in the tail of the pulse, especially in case the tail contains spikes or noise. The “Trigger Hysteresis” feature inhibits the trigger until the input pulse reaches half of the threshold value itself. See **Fig. 2.11** for a diagram of this feature. This option is enabled by default. To disable set bit[30] = 1 of register 0x8040 (DPP Algorithm Control) [RD13].



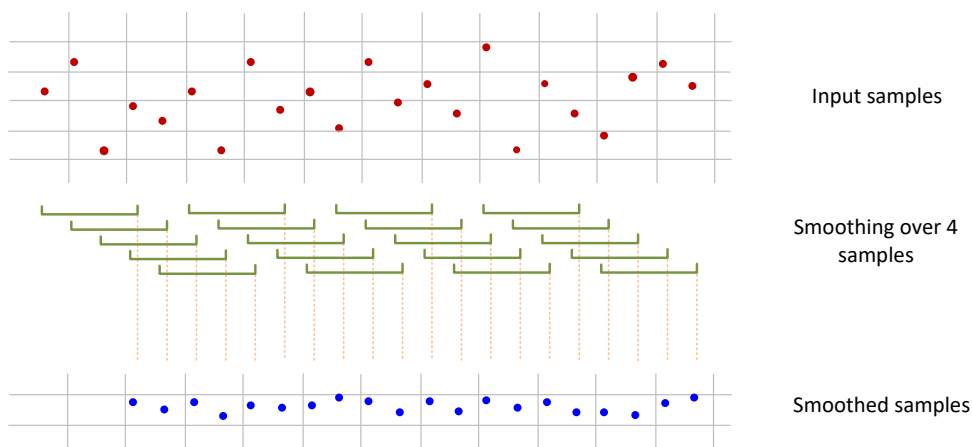
**Fig. 2.11:** Trigger Hysteresis in DPP-PSD firmware. Any other triggers are inhibited after the over-threshold until the input reaches the value of half the threshold.

## Input Smoothing (725 and 730 series only)

The smoothing is a moving average filter, where the input samples are replaced by the mean value of the previous  $n$  samples.  $n$  is defined by bits[15:12] of register 0x1n80 (DPP Algorithm Control) **[RD13]**, whose options are:

- 0000: disabled;
- 0001: 2 samples;
- 0010: 4 samples;
- 0011: 8 samples;
- 0100: 16 samples.

When enabled, the trigger is applied on the smoothed samples, thus reducing triggering on noise. Both CFD and LED triggering modes can be used on the smoothed input. The charge integration is either performed on the input samples or on the smoothed samples, according to bit [11] of register 0x1n80 (DPP Algorithm Control) **[RD13]**.



**Fig. 2.12:** Example of smoothing over four samples. The input samples are averaged over four samples and replaced in the smoothed samples by the mean value.

## Veto (725 and 730 series only)



**Note:** In case of 720, DT5790, and 751 series the veto is managed by the Global Trigger Mask register (0x810C) and Trigger Validation Mask (0x8180 + 4n)

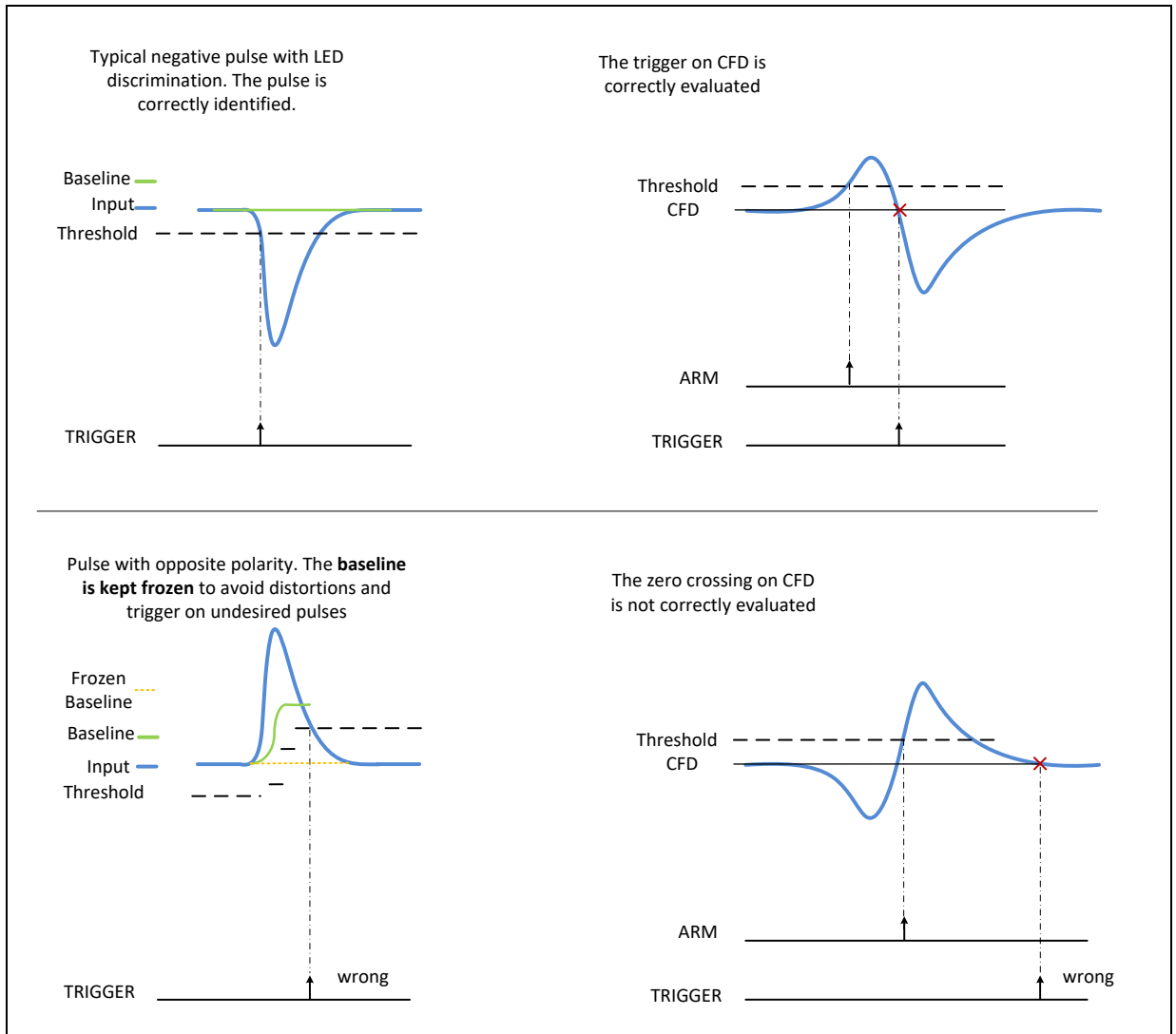
In case of 725 and 730 series the user can set a different coincidence logic between channels inside the couple and the couples themselves. For example, it is possible to set the AND between the channels inside the couple, and set a veto from external trigger for all couples (see **[RD4]** for additional examples). Other digitizer series can handle just one type of coincidence logic.

In particular, while the coincidence inside the couple is managed by bits[6:0] of register 0x1n84 (see **Fig. 2.10**), bits[19:18] of the same register **[RD13]** manage the veto source, whose options are:

- 00 = disabled;
- 01 = the veto signal is common among all channels. It can be set through register 0x810C, and it can be generated by an external trigger or by a combination of the trigger requests from couples;
- 10 = the veto signal is individually set for the couple of channels (each couple can have a different veto). It can be set through register 0x8180 (+4n), where  $n$  is the couple index, and it can be generated by an external trigger or by a combination of the trigger requests from couples;
- 11 = the veto signal comes from events saturating inside the gate (clipping) or from events with opposite polarity, as for example, in case of undershoot/overshoot, the signal can trigger on noise while it returns to zero. The firmware automatically detects pulses with opposite polarity and in case of LED discrimination freezes the baseline. This avoids distortions in the baseline and triggering of wrong pulses (see bottom left of **Fig. 2.13**). In case of CFD discrimination, since the CFD is a bipolar signal, there is a zero crossing even for opposite



polarity (see bottom right of **Fig. 2.13**). Triggers on opposite polarity are inhibited by default; set bit[31] = 1 of register 0x1n80 to disable this option.



**Fig. 2.13:** Example of input of opposite polarity. Top left and top right pictures shows the case of a negative pulse polarity, where the trigger is correctly evaluated both for LED and CFD discrimination. Bottom left picture shows an opposite pulse polarity (positive) and the corresponding CFD (right). To avoid distortions on the baseline (green line) the baseline is kept frozen (yellow) and the event is not triggered. Also the CFD is not inhibited by default.

The user can set the width of the veto duration through register 0x1nD4. In particular bits[15:0] define the width, and bits[17:16] define the step, that can be chosen among 8 ns, 2 us, 524 us, and 134 ms.



**Note:** A veto width equal to 0 means that the veto lasts for the duration of the signal that generated it. A veto width different from 0 sets the veto duration to the value written in the register.

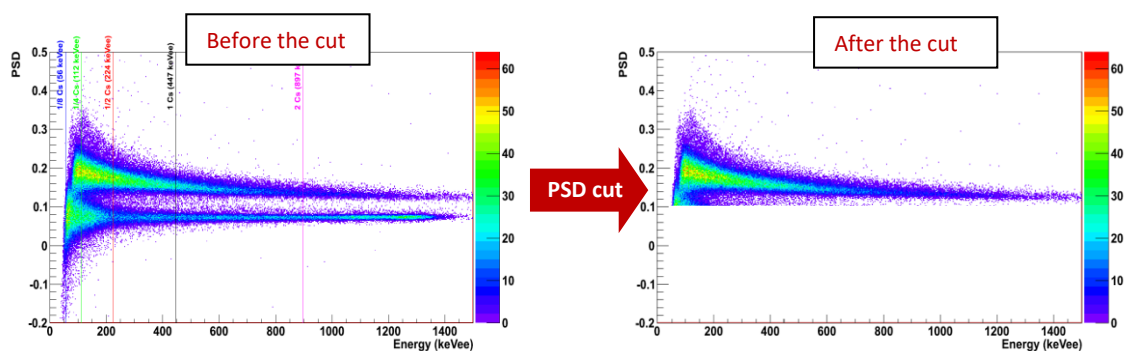
## Online PSD selection

The PSD value as defined in the **Introduction** section can be used to online select the events. Indeed, it is possible to select events under or below a programmable PSD threshold.

Referring to the example of neutron/gamma discrimination shown in **Fig. 2.14**, the cut on PSD allows to reject most of the gamma events, allowing to record only neutrons and the small amount of gamma overlapping with the neutrons. The data throughput after the cut has been significantly reduced.

The user must set the following FPGA registers (refer to **[RD13]**, **[RD17]**, **[RD18]**, **[RD19]**):

- 0x1n78 (where n is the channel number) "Threshold for the PSD cut", 10 bits length. Write the desired PSD cut value multiplied by 1024. Convert this value into hexadecimal if the register write is set in hexadecimal. For example, to cut at 0.2, write  $0.2 * 1024 = 205$  (dec) = CD (hex);
- 0x1n80 (where n is the channel number) "DPP Algorithm Control"; enable either bit[27] or bit[28] to cut on gamma or neutron respectively.



**Fig. 2.14:** 2D scatter plot of PSD parameter vs Energy in a neutron-gamma application. On the left the 2D plot before the cut, on the right the plot after the cut on PSD

## Zero Suppression based on Charge

Input signals of small amplitude can be selected by setting a small threshold level. Unfortunately, this might result in an increase of the noise level. The noise can be distinguished by real signals in the energy spectrum, since it usually appears as a peak close to the 0 energy. Register Charge Zero Suppression Threshold 0x1n44 allows the user to set a threshold in the spectrum ( $Q_{thr}$ ) to cut events with charge  $Q_{long} < Q_{thr}$ . This option can be enabled by setting bit[25] of register 0x1n80.  $Q_{thr}$  is expressed as a 16 bit number, where 1 LSB corresponds to a specific value of charge which depends on the *Charge Sensitivity* value.

### 3. Acquisition Modes

As described on Sect. **DPP-PSD trigger management** each individual channel of the digitizer can trigger independently from the others. When the input signal fires the trigger the DPP-PSD firmware integrates the input samples within the programmed time window.

The main acquisition mode is called “List mode”, where the digitizer provides the time of arrival of the input (also called “Trigger Time Stamp”) and its charge. As soon as the list reaches a certain size, it is made available for readout and the acquisition continues in another buffer. Being the size of the event very small (typically few bytes), the throughput is extremely reduced. The firmware is not designed to make histogram onboard, anyway it can transfer the list information to the software for the histogram management.

The DPP-PSD firmware allows also to acquire waveform samples (i.e. a sequence of samples within a programmable acquisition window) in the “Waveform” acquisition mode (not available for 725-730 series). This acquisition mode is mainly intended to debug and to set the DPP parameters. For each trigger (internal or external), the digitizer saves a portion of the waveform into a local memory buffer. Running in Oscilloscope Mode, the user can view the input signal, the baseline, and other control signals (such as the trigger, the gate, the trigger hold-off, etc.) in the same plot, and easily adjust the parameters for the acquisition. Running in oscilloscope mode implies a very high data throughput, due to the huge number of samples saved into the board memory and then read out by the DAQ software.

The DPP-PSD firmware can manage both acquisition modes together in the “Mixed” acquisition mode, where it is possible to read the charge, the baseline and the time stamp information together with a portion of the waveform, so that the user can retrieve further information and use it off-line, keeping a reasonable level of throughput bandwidth

The DPP-PSD Control Software can manage both the list and the waveform acquisition mode, except for the mixed mode. In the list mode, the software can retrieve the list information from the digitizer to make the relevant histogram and save the output files. Working in waveform mode the software can plot the digital pulse for online monitoring and save the output file. Users who wants to acquire in mixed mode must write their own software. Example codes are available in the Samples folder of the CAENDigitizer library package [RD5].

## 4. Memory Organization

Each channel has a fixed amount of RAM memory to save the events. The memory is divided into a programmable number of buffers (also called “aggregates”), where each buffer contains a programmable number of events. For the 725 and 730 families each buffer is shared between two channels, i.e. channel 0 and channel 1, channel 2 and channel 3, etc. The event format is programmable as well. The board registers involved are the following (refer to [RD13], [RD17], [RD18], [RD19]):

- “Aggregate Organization” ( $N_b$ ), address 0x800C: defines how many aggregates can be contained in the memory ( $n\_aggr = 2^{N_b}$ ).
- “Number of Events per Aggregate” ( $N_e$ ), address 0x1n34: defines the number of events contained in one aggregate. The maximum allowed value is 1023.
- “Record Length” ( $N_s$ ), address 0x1n20: defines the number of samples for the waveform acquisition, when enabled ( $rec\_len = N_s * 8$  for 720, DT5790, 725, and 730 series, and  $rec\_len = N_s * 12$  for 751 series).
- “Board Configuration”, address 0x8000: defines the acquisition mode and the event data format.



**Note:** Those who need to write their own DAQ software, must take care to choose the  $N_e$  value according to the event and buffer size, as explained in the examples in the next section.

For a detailed description, refer to the specific User Manual. Information about the use of these parameters in the CAENDigitizer library can be found in [RD5].

According to the programmed event format, an event can contain a certain number of samples of the waveform, one trigger time stamp, the two charges  $Q_{short}$  and  $Q_{long}$ , and the Baseline/Extras information.

## 720 (DT5790), 725 and 730 series

The following section describes the structure of the memory organization of 720 (DT5790), 725 and 730 series that are quite similar each other. Differences will be explicitly pointed out.

The physical memory inside the board is made of memory locations, each of 128-bit (16B).

In terms of location occupancy:

Trigger Time Stamp = 1 location;

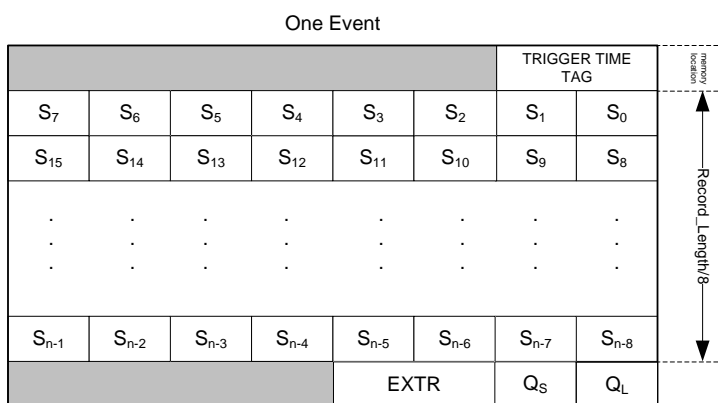
Waveform (if enabled) = 1 location every 8 samples;

Charges ( $Q_L$  and  $Q_S$ ) and Extras (EXTR) = 1 location.

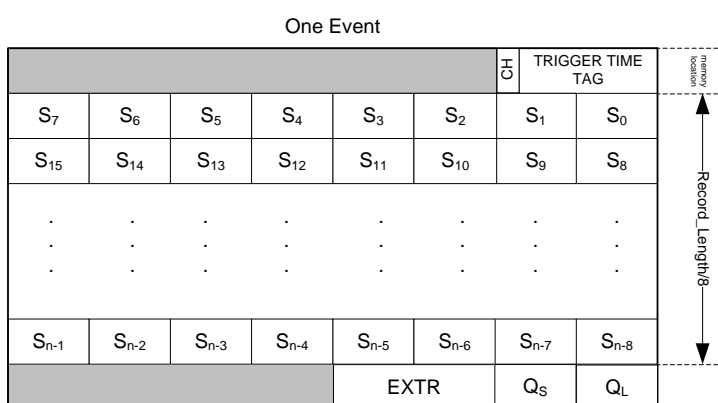
**Fig. 4.1** and **Fig. 4.2** show the data format as saved into the physical memory for 720 (DT5790), 725 and 730 series, respectively. The structure is the same apart for the Trigger Time Tag, that has one bit less in the 725 and 730 format. Since two channels share the same buffer one bit is reserved to store the channel number, where 0 corresponds to the odd channel of the couple, and 1 to the even channel.



**Note:** **Fig. 4.1** and **Fig. 4.2** refers to the event storage into the physical memory of the board. Data are then organized in a different format for the event readout. The event readout format is shown in the **Event Data Format** section.



**Fig. 4.1:** Data organization into the Internal Memory of x720 digitizer and DT5790



**Fig. 4.2:** Data organization into the Internal Memory of x725 and x730 digitizer

As previously said, the “Record Length” and the “Board Configuration” settings determine the event size; the user must calculate the number of event per buffer ( $N_e$ ) and the number of buffers ( $2^{Nb}$ ) accordingly. When the board runs in List Mode, the event memory contains only two locations, one for the Trigger Time Tag and one for the Charge and Baseline. Therefore it is very small and it is suggested to use a big value for  $N_e$  to make the buffer size as big as at least a few KB. Small buffer size results in low readout bandwidth. The only drawback of setting high values for  $N_e$  is that the events are

not available for the readout until the buffer is complete; hence there is some latency between the arrival of a trigger and the readout of the relevant event data. Conversely, when the board runs in Oscilloscope Mode, especially when the record length is large, it is more convenient to keep  $N_e$  low (typically 1).

*Example1:* suppose that mixed mode is enabled and  $N_s$  is set to 400 samples:

event size (in locations) =  $1(\text{Time\_Stamp}) + N_s/8(\text{Waveform}) + 1(\text{Charge\_Baseline}) = 52 \text{ loc.}$

Suppose to set  $N_e = 60$  (number of events per buffer), hence:

buffer\_size (in locations) =  $52 * 60 = 3120 \text{ loc.}$

Supposing that the memory board is made of 128k loc./ch, the number of buffers will be:

$128k/3120 = 42$  (buffers).

This value corresponds to the maximum number of buffers that the memory can contain. However, since the programmable value must be a power of two, the user has to choose the closest number smaller than 42 which can be represented as a power of two, that is  $2^5 = 32$  (i.e.  $N_b = 5$  has to be written in the "Aggregate Organization" register).

*Example2:* suppose that mixed mode is enabled and  $N_s$  is set to 24 samples:

Event size (in locations) =  $1(\text{Time\_Stamp}) + N_s/8(\text{Waveform}) + 1(\text{Charge\_Baseline}) = 5 \text{ loc.}$

Having a small event size, is it convenient to divide the memory into few buffers of bigger size to store a large number of events.

Suppose to have set  $N_b = 3$ , so that the number of buffers is 8.

Supposing that the board memory option is made of 64k locations, each buffer consists in  $64k/8 = 8k$  locations and so the resulting number of event per aggregate should be:

$N_e = 8k/5 = 1639.$

**IMPORTANT:** in this case, the real number of events stored per aggregate is 1023, due to the register length constraint already mentioned.

## 751 series

The physical memory of a board is made of memory locations, each of 128-bit (16B)

In terms of location occupancy:

Trigger Time Stamp = 1 location;

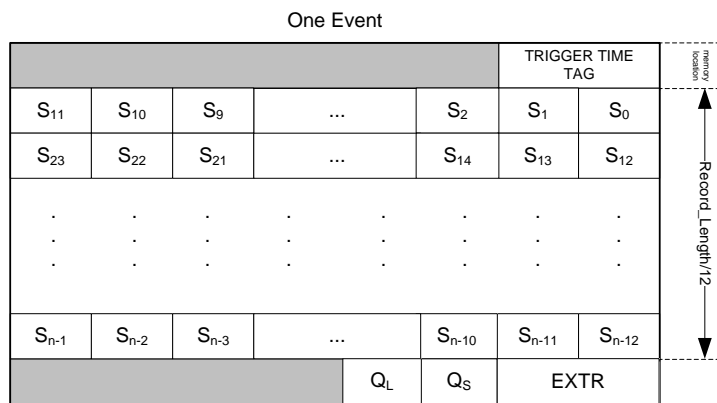
Waveform (if enabled) = 1 location every 12 samples;

Charges ( $Q_L$  and  $Q_S$ ) plus Baseline = 1 location.

Therefore, the events size can be easily calculated. **Fig. 4.3** shows how the data are saved into the physical memory.



**Note:** **Fig. 4.3** refers to the event storage in the physical memory, while the event readout format is shown in the **Event Data Format** section.



**Fig. 4.3:** Data organization into the Internal Memory of x751 digitizer

The same relation between the readout bandwidth and  $N_e$  in the 720 series is valid for the 751 series.

*Example:* suppose that the mixed mode is enabled and  $N_s = 480$  samples:

event size (in locations) =  $1(\text{Time\_Stamp}) + N_s/12(\text{Waveform}) + 1(\text{Charges\_Baseline}) = 42$ .

Suppose to have  $N_e = 30$  (number of events per buffer), hence:

buffer\_size (in locations) =  $42 * 30 = 1260$  loc.

Supposing that the board memory is made of 128k loc./ch, the number of buffers will be:

$128k/1260 = 102$  (buffers).

This value corresponds to the maximum number of buffers that the memory can contain. However, since the programmable value must be a power of two, the user has to choose the closest number smaller than 102 which can be represented as a power of two, that is  $2^6 = 64$  (i.e.  $N_b = 6$  has to be written in the `BUFF_ORG` register).

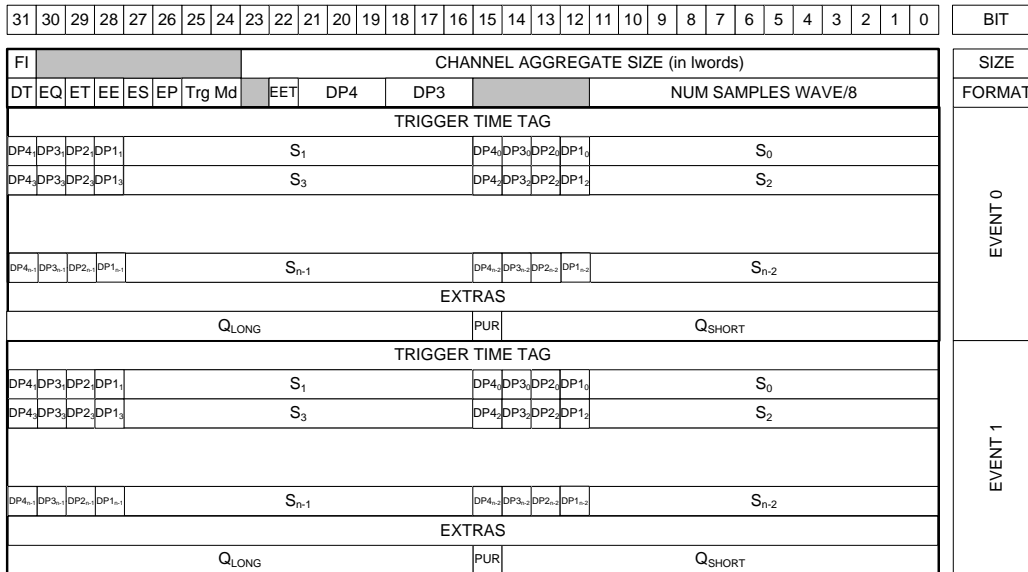
## Event Data Format

When the data readout is performed by the Control Software, the data format has the following encoding. Those who need to write their own acquisition software must take care of the following sections.

### Channel Aggregate Data Format for 720 (DT5790) series

The Channel Aggregate is composed by the set of  $N_e$  events, where  $N_e$  is the programmable number of events contained in one aggregate (see the previous section). The structure of the Channel Aggregate of two events (EVENT 0 and EVENT 1) for 720 (DT5790) series is shown in **Fig. 4.4**, where:

**“CHANNEL AGGREGATE” DATA FORMAT**



**Fig. 4.4:** Channel Aggregate Data Format scheme for 720 series

**FI:** if 1, the second word is the Format Info

**DT:** Dual trace enabled flag (1 = enabled, 0 = disabled)

**EQ:** Charge enabled flag

**ET:** Time Tag enabled flag

**EE:** Extras enabled flag.

**ES:** Waveform (samples) enabled flag

**EP:** Pedestal enabled flag

**Trg Md:** Trigger Mode enabled flag

**EET:** Enable Extended Time Stamp flag. When enabled the extended time stamp is written in the EXTRAS word, according to the options of the EE flag.



**Note:** to enable the Extended Time Stamp word set bit[7] of register 0x1n80.

**DP3:** Digital Virtual Probe 3 selection among:

000 = “External TRG”, the external trigger signal when enabled;

001 = “Over Threshold”, digital signal that is 1 when the input signal is over the requested threshold;

010 = “Shaped TRG”, logic signal generated by a channel in correspondence with its local self- trigger. It is used to propagate the trigger to the other channels of the board and to other external boards, as well as to feed the coincidence trigger logic (refer to **[RD4]**);

011 = “TRG Val. Acceptance Win.”, logic signal corresponding to the time window where the coincidence validation is accepted. The validation enables the event dump into the memory (see **[RD4]**);



100 = "Pile Up", logic pulse set to 1 when a pile up event occurred (to be implemented);

101 = "Coincidence", logic pulse set to 1 when a coincidence occurred (refer to [RD4]).

**DP4:** Digital Virtual Probe 4 selection among:

000 = "Short Gate";

001 = "Over Threshold", digital signal that is 1 when the input signal is over the requested threshold;

010 = "TRG Validation", digital signal that is 1 when a coincidence validation signal comes from the mother board FPGA(refer to [RD4]);

011 = "TRG HoldOff", logic signal generated by a channel in correspondence with its local self- trigger. Other triggers are inhibited for the overall Trigger Hold-Off duration;

100 = "Pile Up", logic pulse set to 1 when a pile up event occurred (to be implemented);

101 = "Coincidence", logic pulse set to 1 when a coincidence occurred (refer to [RD4]).

**NUM SAMPLES WAVE/8** corresponds to the number of words to be read in the event related to the waveform / 4 (2 samples per word)

**DP<sub>i</sub><sub>m</sub>** ( $i=1, \dots, 4; m=0, 1, \dots, n-1$ ): Digital Virtual Probe value  $i$  for sample  $m$

DP1<sub>m</sub> is always the "Trigger" probe value

DP2<sub>m</sub> is always the "Long Gate" probe value

DP3<sub>m</sub> is the value of the probe written in DP3 flag

DP4<sub>m</sub> is the value of the probe written in DP4 flag

**S<sub>m'</sub>** ( $m'=0, 2, 4, \dots, n-2$ ): Even Samples of input signal at time  $t=m'$ . If DT=1, S<sub>m'</sub> corresponds to the Baseline at time  $t=m'+1$

**S<sub>m''</sub>** ( $m''=1, 3, 5, \dots, n-1$ ): Odd Samples of input signal at time  $t=m''$



**Note:** when the "Dual Trace" option is enabled half of the samples are used to store the baseline. Therefore only the remaining half samples are used for the input waveform. In the plot visualization each input sample is duplicated to keep the same granularity. Those who need to acquire waveforms with full resolution should disable the dual trace option. In the DPP-PSD Control Software select the "NONE" option for "Analog Probe 2".

**EXTRAS:** According to the value of EET, the EXTRAS word is read as follows:

If EET = 0, EXTRAS[31] = memory full flag. This bit is set in the first event after the full memory event;

EXTRAS[11:0] = Baseline.

If EET = 1, EXTRAS[31] = memory full flag. This bit is set in the first event after the full memory event;

EXTRAS[14:0] = Extended Time Stamp, corresponding to the most significant bits of the Time stamp. The time stamp therefore becomes a  $32 + 15 = 47$  bit number.



**Note:** to enable the "EXTRAS" word set bit[17] of register 0x8000.

**Q<sub>short/long</sub>**: integrated charge value in the short/long gate

**PUR**: identifies a piled-up event (refer to **Appendix B** for more details)

## Channel Aggregate Data Format for 725 and 730 series

The Channel Aggregate is composed by the set of  $N_e$  events, where  $N_e$  is the programmable number of events contained in one aggregate (see the previous section). The structure of the Channel Aggregate of two events (EVENT 0 and EVENT 1) for 725 and 730 series is shown in **Fig. 4.4**, where:

### “CHANNEL AGGREGATE” DATA FORMAT

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		BIT	
1											CHANNEL AGGREGATE SIZE (in lwords)																		SIZE					
DT	EQ	ET	EE	ES	EX		AP	DP2		DP1		NUM SAMPLES WAVE/8																	FORMAT					
CH	TRIGGER TIME TAG																																EVENT 0	
DP2 <sub>1</sub>	DP1 <sub>1</sub>	S <sub>1</sub>																DP2 <sub>0</sub>	DP1 <sub>0</sub>	S <sub>0</sub>														
DP2 <sub>2</sub>	DP1 <sub>2</sub>	S <sub>3</sub>																DP2 <sub>3</sub>	DP1 <sub>3</sub>	S <sub>2</sub>														
DP2 <sub>n-1</sub>	DP1 <sub>n-1</sub>	S <sub>n-1</sub>																DP2 <sub>n-2</sub>	DP1 <sub>n-2</sub>	S <sub>n-2</sub>														
EXTRAS																																		EVENT 1
Q <sub>LONG</sub>																PUR	Q <sub>SHORT</sub>																	
CH	TRIGGER TIME TAG																																	
DP2 <sub>1</sub>	DP1 <sub>1</sub>	S <sub>1</sub>																DP2 <sub>0</sub>	DP1 <sub>0</sub>	S <sub>0</sub>														
DP2 <sub>2</sub>	DP1 <sub>2</sub>	S <sub>3</sub>																DP2 <sub>3</sub>	DP1 <sub>3</sub>	S <sub>2</sub>														
DP2 <sub>n-1</sub>	DP1 <sub>n-1</sub>	S <sub>n-1</sub>																DP2 <sub>n-2</sub>	DP1 <sub>n-2</sub>	S <sub>n-2</sub>														
EXTRAS																																		
Q <sub>LONG</sub>																PUR	Q <sub>SHORT</sub>																	

**Fig. 4.5:** Channel Aggregate Data Format scheme for 725 and 730 series

**DT:** Dual trace enabled flag (1 = enabled, 0 = disabled)

**EQ:** Charge enabled flag, must be 1

**ET:** Time Tag enabled flag, must be 1

**EE:** Extras enabled flag

**ES:** Waveform (samples) enabled flag

**EX:** Extras option enabled flag:



**Note:** to enable the “EXTRAS” word set bit[17] of register 0x8000 (refer to Sect. **CFD implementation for 725, 730, and 751 series** for more details).

000 = the word “EXTRAS” will be read as:

[31:16] = extended time stamp: those 16 bits must be read as the most significant bits of the time stamp, which becomes a 31+16=47 bit number;

[15:0] = the baseline value multiplied by 4.

001 = the word “EXTRAS” will be read as:

[31:16] = extended time stamp: those 16 bits must be read as the most significant bits of the time stamp, which becomes a 31+16=47 bit number;

[15:0] = flags, where bit[15]: Trigger Lost (the first event after a trigger lost has this flag set); bit[14]: Over-range (identifies an event saturating inside the gate - clipping); bit[13]: 1024 trigger counted (every 1024 counted events this flag is high); bit[12]: N lost trigger counted (every N counted lost events this flag is high, where N is set from bits[17:16] of register 0x1n84);

010 = the word “EXTRAS” will be read as:

[31:16] = extended time stamp – the trigger time stamp becomes a 31+16=47 bit number;

[15:10] = flags, where bit[15]: Trigger Lost (the first event after a trigger lost has this flag set); bit[14]: Over-range (identifies an event saturating inside the gate - clipping); bit[13]: 1024 trigger counted (every 1024 counted events this flag is high); bit[12]: N lost trigger counted (every N counted lost events this flag is high, where N is set from bits[17:16] of register 0x1n84);

[9:0] = fine time stamp ( $T_{fine}$  – see the definition in Sect. **CFD implementation for 725, 730, and 751 series** ).

100 = the word “EXTRAS” will be read as:

[31:16] = Lost Trigger Counter;

[15:0] = Total Trigger Counter.

101 = the word “EXTRAS” will be read as:

[31:16] = CFD Sample After the Zero Crossing (SAZC);

[15:0] = CFD Sample Before the Zero Crossing (SBZC).

111 = the word “EXTRAS” will be read as the fixed value of 0x12345678 (debug use only).

**AP:** Analog Probe selection. For 725 and 730 series possible selections are:

If DT = 0:

00 = “Input”;

01 = “CFD”;

If DT = 1:

00 = “Input” and “Baseline”;

01 = “CFD” and “Baseline”;

10 = “Input” and “CFD”.



**Note:** The CFD trace is available only if the discrimination mode is CFD (bit[6] = 1 of register 0x1n80). If discrimination mode is LED (bit[6] = 0 of register 0x1n80), the CFD trace becomes the Smoothed Input (refer to Sect. **Input Smoothing (725 and 730 series only)**).

**DP1:** Digital Virtual Probe 1 selection among:

000 = “Long Gate”;

001 = “Over Threshold”, digital signal that is 1 when the input signal is over the requested threshold;

010 = “Shaped TRG”, logic signal generated by a channel in correspondence with its local self- trigger. It is used to propagate the trigger to the other channels of the board and to other external boards, as well as to feed the coincidence trigger logic (refer to **[RD4]**);

011 = “TRG Val. Acceptance Win.”, logic signal corresponding to the time window where the coincidence validation is accepted. The validation enables the event dump into the memory (see **[RD4]**);

100 = “Pile Up”, logic pulse set to 1 when a pile up event occurred (not implemented);

101 = “Coincidence”, logic pulse set to 1 when a coincidence occurred (refer to **[RD4]**);

110 = reserved;

111 = “Trigger”.

**DP2:** Digital Virtual Probe 2 selection among:

- 000 = "Short Gate";
- 001 = "Over Threshold", digital signal that is 1 when the input signal is over the requested threshold;
- 010 = "TRG Validation", digital signal that is 1 when a coincidence validation signal comes from the mother board FPGA (refer to **[RD4]**);
- 011 = "TRG HoldOff", logic signal generated by a channel in correspondence with its local self- trigger. Other triggers are inhibited for the overall Trigger Hold-Off duration;
- 100 = "Pile Up", logic pulse set to 1 when a pile up event occurred (to be implemented);
- 101 = "Coincidence", logic pulse set to 1 when a coincidence occurred (refer to **[RD4]**);
- 110 = reserved;
- 111 = "Trigger".



**Note:** Digital Virtual Probes can be disabled by setting bit[31] of register 0x8000.

**NUM SAMPLES WAVE/8** corresponds to the number of words to be read in the event related to the waveform / 4 (2 samples per word)

**CH:** since two consecutive channels share the same buffer, the CH flag identifies if the even channel or the odd channel participated to the event (0 for even, 1 for odd).

**DP<sub>i</sub><sub>m</sub>** ( $i=1, 2$ ;  $m=0, 1, \dots, n-1$ ): Digital Virtual Probe value  $i$  for sample  $m$

DP1<sub>m</sub> is the value of the probe written in DP1 flag

DP2<sub>m</sub> is the value of the probe written in DP2 flag

**S<sub>m'</sub>** ( $m'=0, 2, 4, \dots, n-2$ ): Even Samples of the analog probe (whose flag is stored in AP) at time  $t=m'$ .

**S<sub>m''</sub>** ( $m''=1, 3, 5, \dots, n-1$ ): Odd Samples of the analog probe (whose flag is stored in AP) at time  $t=m''$ .

If DT=1, S<sub>m''</sub> corresponds to the second analog probe at time  $t=m''-1$ .

For example if DT= 1, and the analog probe selection is "Input" – "Baseline", S<sub>m'</sub> will corresponds to the "Input" sample at time m', while S<sub>m''</sub> will corresponds to the "Baseline" at the same time m' ( $= m''-1$ ).

**Q<sub>short/long</sub>**: integrated charge value in the short/long gate

**PUR:** detect an event whose energy is not correctly evaluated. This might be the case of a pile-up event or of an event saturating inside the gate (see bit[24] of register 0x1n84)



**Note:** when the "Dual Trace" option is enabled half of the samples are used to store the baseline. Therefore only the remaining half samples are used for the input waveform. In the plot visualization, each input sample is duplicated to keep the same granularity. Those who need to acquire waveforms with full resolution should disable the dual trace option. In the DPP-PSD Control Software select the "NONE" option for "Analog Probe 2".

## Channel Aggregate Data Format for 751 series

The Channel Aggregate is composed by the set of  $N_e$  events, where  $N_e$  is the programmable number of events contained in one aggregate (see the previous section). The structure of the Channel Aggregate of two events (EVENT 0 and EVENT 1) for series 751 is shown in Fig. 4.6, where:

### “CHANNEL AGGREGATE” DATA FORMAT

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	BIT
FORMAT						CHANNEL AGGREGATE SIZE (in lwords)																					SIZE					
DT	EQ	ET	EE	ES	AP		DP2		DP1		ED			NUM SAMPLES WAVE/12														FORMAT				
																							INTERP		DM	PP	EX		EVENT 0			
TRIGGER TIME TAG																																
TrgCod		S <sub>2</sub>								S <sub>1</sub>								S <sub>0</sub>														
TrgCod		S <sub>5</sub>								S <sub>4</sub>								S <sub>3</sub>														
TrgCod		S <sub>n-1</sub>								S <sub>n-2</sub>								S <sub>n-3</sub>														
EXTRAS																																
Q <sub>LONG</sub>													PUR	Q <sub>SHORT</sub>																		
TRIGGER TIME TAG																																
TrgCod		S <sub>2</sub>								S <sub>1</sub>								S <sub>0</sub>								EVENT 1						
TrgCod		S <sub>5</sub>								S <sub>4</sub>								S <sub>3</sub>														
TrgCod		S <sub>n-1</sub>								S <sub>n-2</sub>								S <sub>n-3</sub>														
EXTRAS																																
Q <sub>LONG</sub>													PUR	Q <sub>SHORT</sub>																		

Fig. 4.6: Channel Aggregate Data Format scheme for 751 series

**FORMAT:** defines the structure of the two consecutive words of FORMAT. Options are:

1000: format consistent with DPP-PSD firmware revision **less than 132.32**, where the second word of the FORMAT is not present, AP = 00, and EXTRAS[9:0] = BASELINE. This option is valid if bit[31] of register 0x8000 is equal to 0.

0100: format consistent with DPP-PSD firmware revision **equal/greater than 132.32**. The event data format is defined below. Enable this option by setting bit[31] = 1 of register 0x8000.

**DT:** Dual trace enabled flag (1 = enabled, 0 = disabled)

**EQ:** Charge enabled flag

**ET:** Time Tag enabled flag

**EE:** Extras enabled flag

**ES:** Waveform (samples) enabled flag

**AP:** Analog Probe selection. Possible selections are:

If DT = 0:

00 = “Input”;

01 = “CFD”;

If DT = 1:

00 = “Input” and “Baseline”;

01 = “CFD” and “Baseline”;

10 = “Input” and “CFD”.

**DP1:** Digital Virtual Probe 1 selection among:

- 000 = "Long Gate";
- 001 = "Over Threshold", digital signal that is 1 when the input signal is over the requested threshold;
- 010 = "Shaped TRG", logic signal generated by a channel in correspondence with its local self- trigger. It is used to propagate the trigger to the other channels of the board and to other external boards, as well as to feed the coincidence trigger logic (refer to **[RD4]**);
- 011 = "TRG Val. Acceptance Win.", logic signal corresponding to the time window where the coincidence validation is accepted. The validation enables the event dump into the memory (see **[RD4]**);
- 100 = "Pile Up", logic pulse set to 1 when a pile up event occurred. This trace remains high for the whole long gate duration;
- 101 = "Coincidence", logic pulse set to 1 when a coincidence occurred (refer to **[RD4]**);
- 110 = Reserved;
- 111 = "Trigger"

**DP2:** Digital Virtual Probe 2 selection among:

- 000 = "Short Gate";
- 001 = "Over Threshold", digital signal that is 1 when the input signal is over the requested threshold;
- 010 = "TRG Validation", digital signal that is 1 when a coincidence validation signal comes from the mother board FPGA (refer to **[RD4]**);
- 011 = "TRG HoldOff", logic signal generated by a channel in correspondence with its local self- trigger. Other triggers are inhibited for the overall Trigger Hold-Off duration;
- 100 = "Pile Up Trigger", digital signal that is 1 whenever a pile-up occurred;
- 101 = "PSD cut high", logic pulse that is 1 when the event has a PSD value greater than the set PSD threshold (see register 0x1n78);
- 110 = "Baseline Freeze", logic pulse that is 1 when the baseline calculation is frozen for the gate integration or during the trigger hold-off;
- 111 = "Trigger".

**ED:** Digital Probe enabled flag.



**Note:** When  $ED \neq 0$ , the analog trace sample is represented into 8 bits, rather than 10 bits. Indeed the two most significant bits of each samples are reserved for the two digital probes. Conversely, when  $ED = 0$ , each analog trace sample is represented into 10 bits.

**INTERP:** Interpolation Points used for the CFD or LED linear interpolation. The n-th sample before and after the zero crossing are defined according to the options:

- 000 = the sample before and after the zero crossing;
- 001 = second sample before and after the zero crossing;
- 010 = third sample before and after the zero crossing;
- 011 = fourth sample before and after the zero crossing;
- 100 = fifth sample before and after the zero crossing;
- 101 = sixth sample before and after the zero crossing;
- 110 = seventh sample before and after the zero crossing;
- 111 = eighth sample before and after the zero crossing.

**DM:** Discrimination Mode. Options are:

- 0 = Leading Edge Discrimination;
- 1 = digital Constant Fraction Discrimination.

**PP:** Pulse Polarity identifies the pulse polarity and the trigger on the rising/falling edge.

0 = trigger on rising edge;

1 = trigger on falling edge.

**EX:** Extras option enabled flag:



**Note:** to enable the “EXTRAS” word set bit[17] of register 0x8000 (refer to Sect. **CFD implementation for 725, 730, and 751 series** for more details).

000 = the word “EXTRAS” will be read as:

[31:16] = extended time stamp: those 16 bits become the most significant bits of the Time Stamp representation, which becomes a 32+16=48 bit number;

[15:0] = the baseline value multiplied by 8.

001 = the word “EXTRAS” will be read as:

[31:16] = extended time stamp: those 16 bits become the most significant bits of the Time Stamp representation, which becomes a 32+16=48 bit number;

[15:0] = flags, currently N.A.;

010 = the word “EXTRAS” will be read as:

[31:16] = extended time stamp – the trigger time stamp becomes a 32+16=48 bit number;

[15:10] = flags, currently N.A.;

[9:0] = fine time stamp ( $T_{fine}$  – see the definition in Sect. **CFD implementation for 725, 730, and 751 series** ).

010 = the word “EXTRAS” will be read as:

[31:26] = flags, currently N.A.;

[26:16] = fine time stamp ( $T_{fine}$  – see the definition in Sect. **CFD implementation for 725, 730, and 751 series** ).

[15:0] = the baseline value multiplied by 8.

101 = the word “EXTRAS” will be read as:

[31] = Discrimination Mode. Options are the same of DM field;

[30] = Pulse Polarity. Options are the same of PP field;

[29:20] = n-th Sample Before the Zero Crossing (SBZC), according to bits[12:10] of register 0x8000 (see definition from Sect. **CFD implementation for 725, 730, and 751 series** );

[19:10] = n-th Sample After the Zero Crossing (SAZC), according to bits[12:10] of register 0x8000 (see definition from Sect. **CFD implementation for 725, 730, and 751 series** );

[9:0] = Mid-Scale value (= 512) in case of CFD, or Threshold Value in case of LED.

**NUM SAMPLES WAVE/12** corresponds to the number of words to be read in the event related to the waveform / 4 (3 samples per word)

$S_{m'}$  ( $m'=0, 2, 4, \dots, n-2$ ): Even Samples of input signal at time  $t=m'$ . (\*)

$S_{m''}$  ( $m''=1, 3, 5, \dots, n-1$ ): Odd Samples of input signal at time  $t=m''$ . If  $DT=1$ ,  $S_{m''}$  corresponds to the Baseline at time  $t=m''-1$ . (\*)

**Trg Cod**: encodes on which sample (of the same word) the trigger fired

00 = no trigger

01 = trigger on the first sample  $S_0$

10 = trigger on the second sample  $S_1$

11 = trigger on the third sample  $S_2$

$Q_{short/long}$ : integrated charge value in the short/long gate

**PUR**: identifies a piled-up event (refer to **Appendix B** for more details)



**Note**: when the “**Dual Trace**” option is enabled half of the samples are used to store the baseline. Therefore only the remaining half samples are used for the input waveform. In the plot visualization each input sample is duplicated to keep the same granularity. Those who need to acquire waveforms with full resolution should disable the dual trace option. In the DPP-PSD Control Software select the “**NONE**” option for “**Analog Probe 2**”.



**Note**: when the “**Digital Probes**” are enabled, the input is represented into 8 bits rather than 10 bits. This means that the input granularity in the plot visualization is four. Those who need to acquire waveforms with full resolution should select “**NONE**” for both Digital Trace 2 and 3 in the DPP-PSD Control Software (see Sect. **GUI Description** for further details).

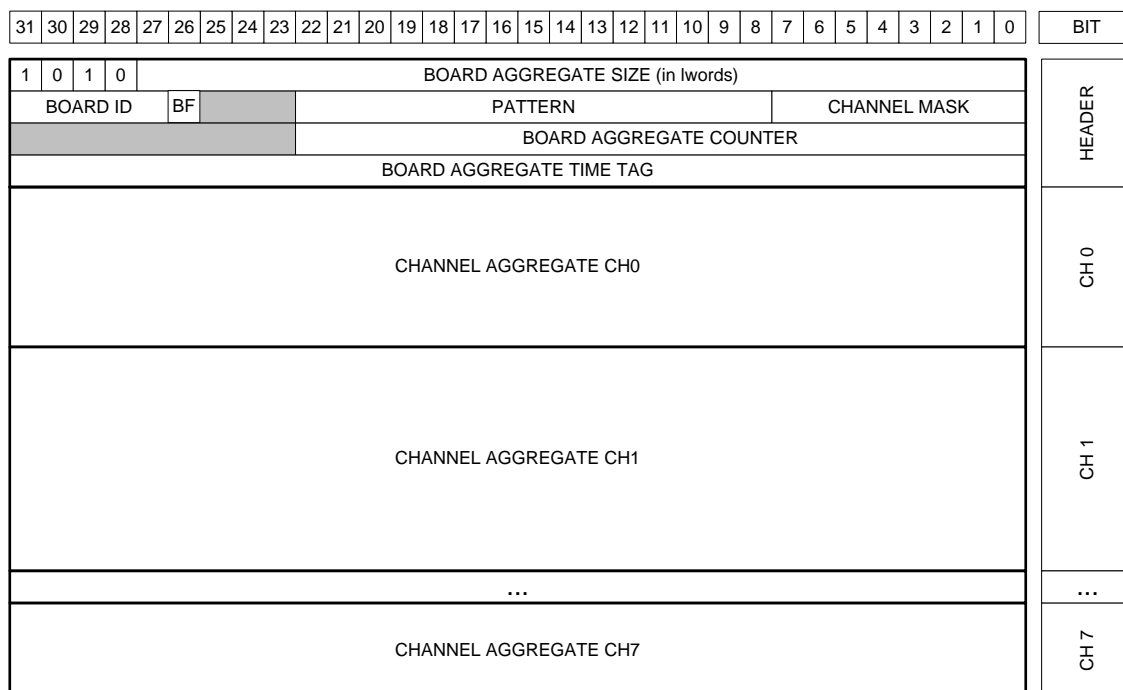


## Board Aggregate Data Format

For each readout request (occurring when at least one channel has available data to be read) the “interface FPGA (ROC)” reads one aggregate from each enabled channel memory. No more than one aggregate per channel is read each time. The sample of Channel Aggregates is the Board Aggregate. If one channel has no data, that channel does not come into the Board Aggregate.

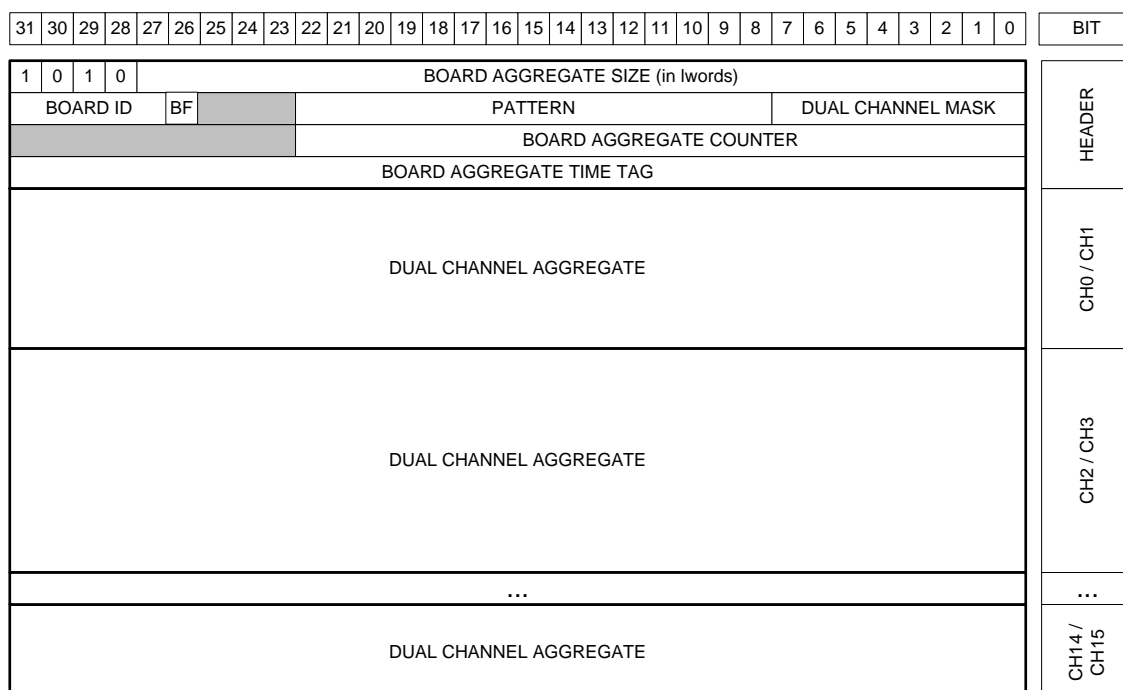
The data format when all 8 channels of a VME have available data is as shown in **Fig. 4.7** for 720 (DT5790) series, in **Fig. 4.8** for 725 and 730 series, and in **Fig. 4.9** for 751 series:

### “BOARD AGGREGATE” DATA FORMAT for 720 and DT5790 series

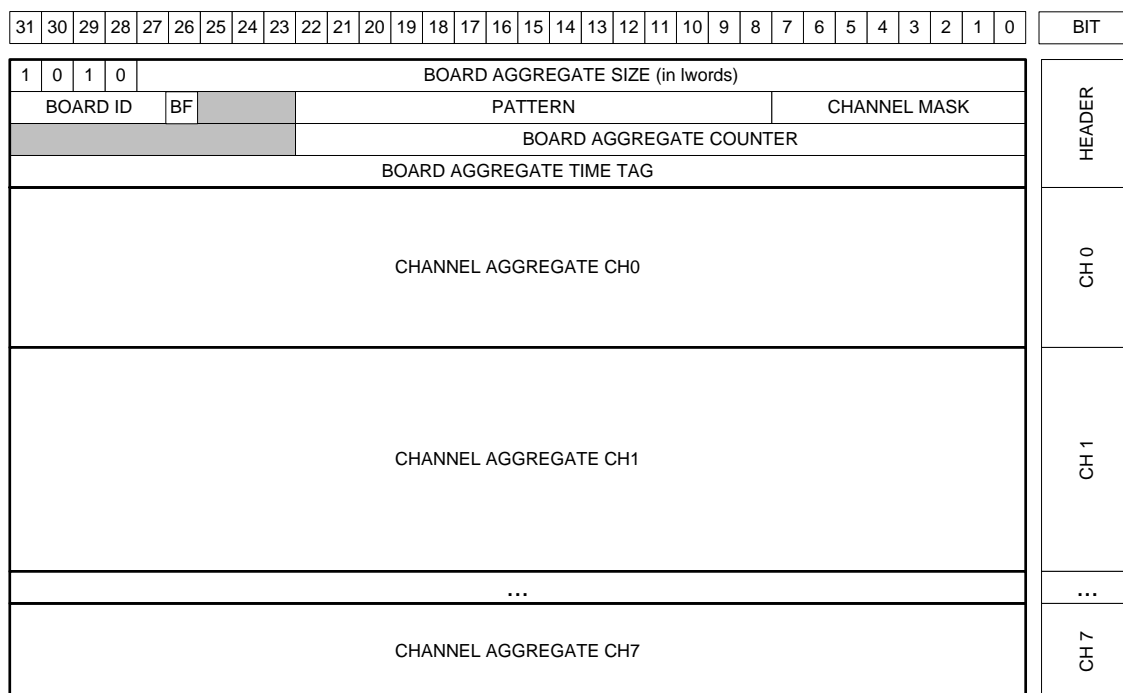


**Fig. 4.7:** Board Aggregate Data Format scheme for 720 (DT5790) series

### “BOARD AGGREGATE” DATA FORMAT for 725 and 730 series



**Fig. 4.8:** Board Aggregate Data Format scheme for 725 and 730 series

**“BOARD AGGREGATE” DATA FORMAT for 751 series**

**Fig. 4.9:** Board Aggregate Data Format scheme for 751 series

**BOARD AGGREGATE SIZE:** total size of the aggregate

**BOARD ID:** corresponds to the GEO address of the board. In case of VX boards this number is automatically set for each board. In case of VME boards this value is by default = 0 for all boards. It is possible to set the Board ID through register 0xEF08. The GEO address is quite useful in case of concatenate BLT (CBLT) read.

**BF:** Board Fail flag. This bit is set to “1” because of a hardware problem, as for example the PLL unlock. The user can investigate the problem checking the *Board Failure Status* register 0x8178, or contacting CAEN support (refer to Chapter **Technical support**).



**Notes:** BF bit is meaningful only for ROC FPGA firmware revision greater than 4.5. It is *reserved* for previous releases.

**PATTERN:** is the value read from the LVDS I/O (VME only);

**CHANNEL MASK:** corresponds to those channels participating to the Board Aggregate;

**DUAL CHANNEL MASK:** corresponds to the couple of channels participating to the Board Aggregate (725 and 730 only);

**BOARD AGGREGATE COUNTER:** counts the board aggregate. It increases with the increase of board aggregates;

**BOARD AGGREGATE TIME TAG:** is the time of creation of the aggregate (this does not correspond to any physical quantity).

## Data Block

The readout of the digitizer is done using the Block Transfer (BLT, refer to [RD5]); for each transfer, the board gives a certain number of Board Aggregates, consisting in the Data Block. The maximum number of aggregates that can be transferred in a BLT is defined by the READOUT\_BTL\_AGGREGATE\_NUMBER. In the final readout each Board Aggregate comes successively. In case of  $n$  Board Aggregates, the Data Block is as in Fig. 4.10.

### DATA BLOCK

BOARD AGGREGATE 0
BOARD AGGREGATE 1
...
BOARD AGGREGATE $n-1$

Fig. 4.10: Data Block scheme

## 5. Getting Started

### Scope of the chapter

This chapter is intended to provide the user with a quick guide of the DPP-PSD Control Software, to deal with a DPP-PSD System in a practical use. For a demo purpose we used a gamma source of Cobalt-60 only. The user can refer to **[RD2]** and **[RD3]** for a real neutron – gamma discrimination application.

All steps in this chapter are made with the 720 series. The same behaviour can be generalized for the 725, 730 and 751 series, as well as the DT5790. Exceptions and specific functioning of the 725, 730 and 751 series will be explicitly mentioned in the text.

### System Overview

CAEN's DPP-PSD System proposed in the chapter consists of the following CAEN products:

- DT5720B, 4-channel 12-bit 250 MS/s Desktop Digitizer.
- DPP-PSD firmware for 720 series, release 4.9\_131.11, running on the Digitizer.
- DPP-PSD Control Software, release 1.3.3, running on the host station.

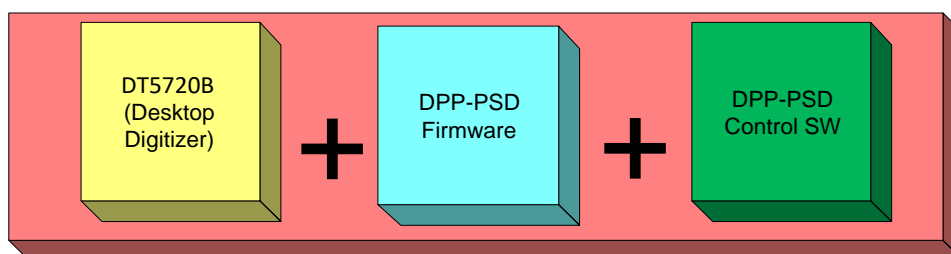
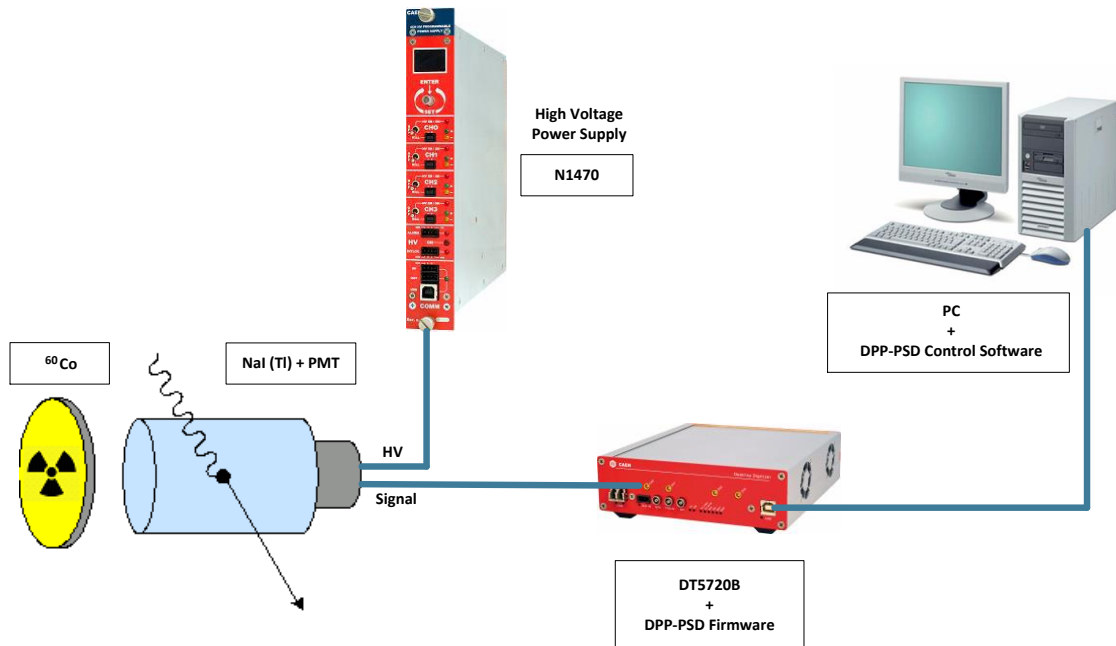


Fig. 5.1: CAEN DPP-PSD System components

## Hardware Setup

The DPP-PSD receives on channel 0 of the DT5720B the signal from a NaI(Tl) coupled with a PMT. The CAEN N1470 (a 4-channel, HV Programmable Power Supply board) provides the supply to the detector ( $V_{bias} = 800$  V). A Cobalt-60 ( $^{60}\text{Co}$ ) gamma ray source (counting rate  $\sim 1$  KHz) is used. A computer equipped with a Microsoft Windows 7 Professional 64-bit OS acts as host station. The communication protocol between the computer and the Digitizer is USB (2.0 version). The use of DT5790 replaces on the same board both the external power supply and the DT5720B.



**Fig. 5.2:** The hardware setup including the digitizer running the DPP-PSD firmware used for the practical application

## Drivers and Software

To manage the DPP-PSD System, the host station needs either Windows or Linux OS, and the third-party software **Java Runtime Environment 7** or later (trademark of Oracle Inc., downloadable from <http://www.java.com>). Linux users must also take care of proper installation of **gnuplot** graphical tool, as well as of **CAEN Libraries**. The latter can be downloaded from CAEN website (login required before to download).

According to the preferred way of connection to the digitizer, users must also take care of proper installation of USB or optical drivers. In our case we are going to describe the procedure for USB connection.

### ✓ DRIVERS

- **USB 2.0** CAEN driver.



**Note:** If you're using a different communication interface (i.e. Optical Link or VME), the related driver is required.



**Note:** It is recommended to install the driver before to connect the hardware.



**Note:** Detailed installation steps of CAEN USB drivers for communicating with desktop digitizers are described for several Microsoft Windows OSs in [RD6].

#### How to install the driver (Windows)

**Download** the latest release of the **USB driver** for Windows on CAEN website in the 'Software/Firmware' area at the DT5720 page.

**Unpack** the driver package.

**Power on** the **Digitizer** and **plug** the **USB cable** in a USB port on your computer.

Windows will try to find drivers and, in case of failure the message "**Device driver software was not successfully installed**" is displayed and the driver needs to be installed manually:

**Go to** the system's **Device Manager** through the Control Panel and **check** for the **CAEN DT5xxx USB1.0** unknown device.

**Right click** and **select Driver software update** in the scrolling menu.

**Select** the option to **browse my computer for driver software**.

**Point to** the **driver folder** and finalize the installation.

#### How to install the driver (Linux)

**Download** the latest release of the **USB driver** for Linux on CAEN website in the 'Software/Firmware' area at the DT5720 page.

**Unpack** the driver package (tar -zxf CAENUSBDrvB-xxx.tgz).

**Go to** the driver **folder** (cd CAENUSBDrvB-xxx).

**Follow** the **instructions** on the **Readme.txt** file.

**Type:** make

sudo make install

**Reboot** your machine

### ✓ SOFTWARE

- **DPP-PSD Control Software** for Windows OS.

**Download** the standalone **DPP-PSD Control Software 1.3.3** full installation package on CAEN website in the 'Download' area at the DPP-PSD Control Software page (**login is required before the download**).

**Unpack** the **installation package**, **launch** the **setup file** and **complete** the **Installation wizard**.

- **DPP-PSD Control Software** for Linux.

**Download** the DPP-PSD\_ControlSoftware-1.3.3.tar.gz package on CAEN website in the 'Download' area at the DPP-PSD Control Software page (**login is required before the download**).

**Unpack** the **installation package** (tar -zxf DPP-PSD\_ControlSoftware-1.3.3.tar.gz).

**Follow** the instruction on **Setup/Linux/Readme.txt**

**Type:** ./configure

make

sudo make install

**Launch** the Control Software typing **DPP-PSD\_ControlSoftware**



**Note:** in the Linux environment it is required to first install CAENVME, CAENComm and CAENDigitizer. You can find those libraries in the CAEN web page. In the Windows environment all libraries come within the control software package.

## Firmware and Licensing

The firmware upgrade is an advanced feature that can be performed only in case the user wants to upgrade the current firmware to a new version, or to upload a different firmware on the board. The .cfa file format checks for the board model to ensure that firmware upgrade is made on the correct board.

### ✓ How to install the firmware

**Download** the **DPP-PSD Firmware** (.cfa) for 720 series on CAEN website in the 'Download' area at the DPP-PSD page. Download the correct file according to the digitizer family in use.

**Download** the **CAENUpgrader** software to upload the firmware on your board. The program full installation package for Windows OS is available on CAEN website in the 'Download' area at the CAENUpgrader page.

**Unpack** the **installation package**, **launch** the **setup file** and **complete** the **Installation wizard**.

**Run** the **CAENUpgrader GUI** by one of the following options:

- The **desktop icon** for the program
- The **Quick Launch icon** for the program
- The **.jar file** in the *bin* folder from the installation path on your host

**Select** '**Upgrade Firmware**' in the '**Available actions**' scroll box menu of the '**Board Upgrade**' tab.

**Select** the **model** of your board in the '**Board Model**' scroll box menu.

**Enter** the **.cfa file** in the '**Firmware binary file**' text box by the '**Browse**' button.

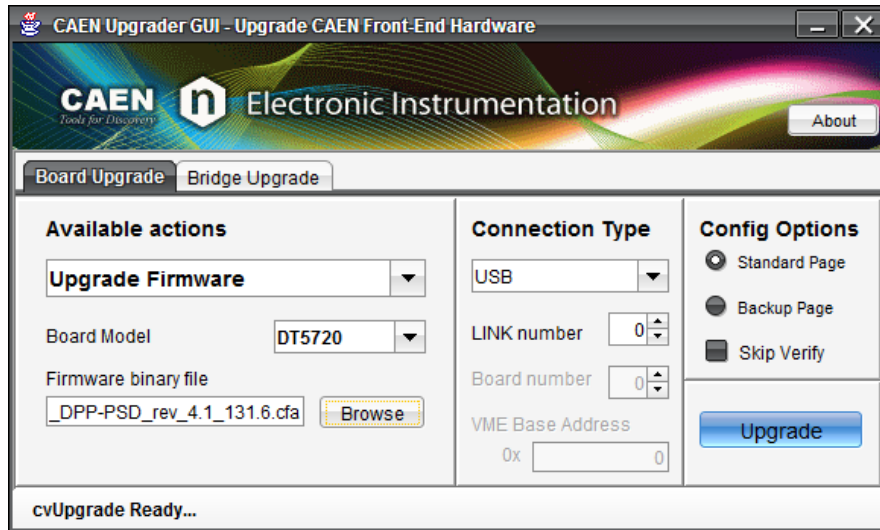
**Set** '**USB**' in the '**Connection Type**' scroll box menu.

**Set** '**0**' as '**Link number**' setting.

**Check** '**Standard Page**' in the '**Config Options**'.

**Press** the '**Upgrade**' button to perform the upload; after few seconds, a pop up message will inform you about the successful upgrade.

**Power cycle** the **board**.



**Fig. 5.3:** CAENUpgrader settings for DPP-PSD firmware upgrade

Note that when running the DPP-PSD Control Software, the program checks for the firmware loaded in the target Digitizer. If no license is found, a pop-up warning message shows up and reports the time left before the acquisition is stopped (trial version). To unlock the DPP firmware and to use it without any time limitation, you need to purchase a license from CAEN. Refer to **[RD7]** for detailed instructions on how to use CAENUpgrader and the licensing procedure.



## Practical Use

The following step-by-step procedure shows how to use the DPP-PSD Control Software (see Chapter **Software Interface**) in an application of gamma ray detection. In particular it is shown how to set the relevant DPP parameters and plot the signals (Oscilloscope mode). Finally how to display the energy histogram and the 2D-histogram for the Neutron -Gamma discrimination (Histogram mode), and how to save the acquired data.

Check that the whole hardware in your setup is properly connected and powered on.



**Note:** After typing the value of a parameter in a box menu, press the “Enter” key on your keyboard to activate the setting.

### 1. Run the software.

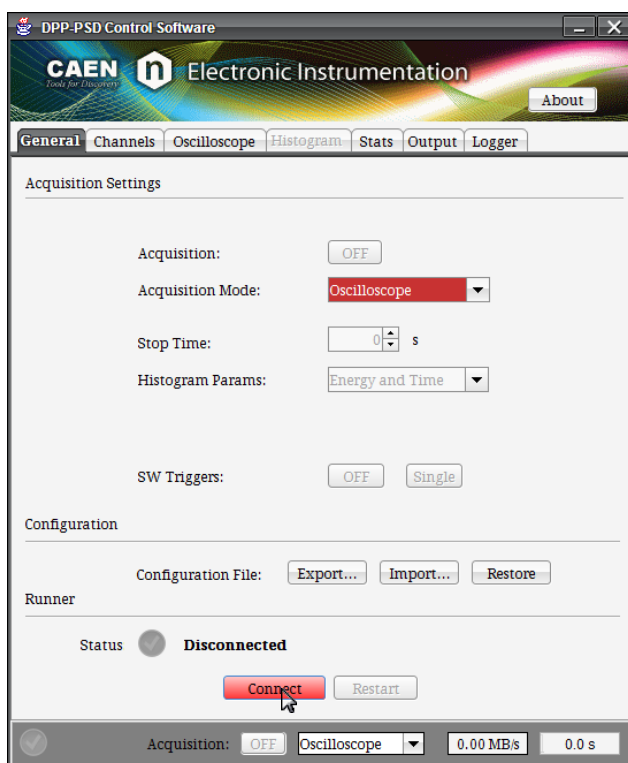
Run the **DPP-PSD Control Software GUI**, according to the options selected in the installation wizard, choosing one of the following options:

- The **Desktop icon** for the program
- The **Quick Launch icon** for the program
- The **.bat file** in the main folder from the installation path on your host

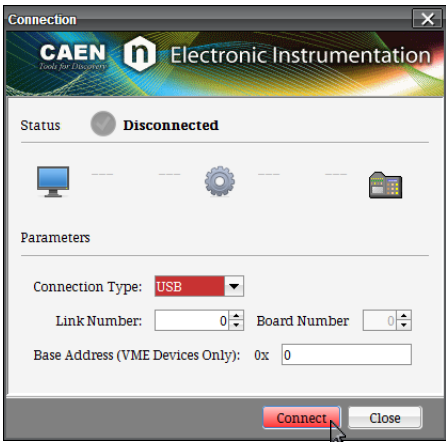
### 2. Connect to the Digitizer.

**Path1:** Tab **GENERAL** → Section **RUNNER**

**Action1:** click the button **CONNECT**. The “Connection” window will appear.



**Action2:** set the connection parameters values. Using a USB communication link with a Desktop digitizer, the correct settings are: **TYPE** = “**USB**”, **LINK** = “**0**” and **ADDRESS** = “**0**”. **Tab. 5.1** shows the setting values for common communication channels and Digitizers. Further examples are in **Tab. 7.2**.

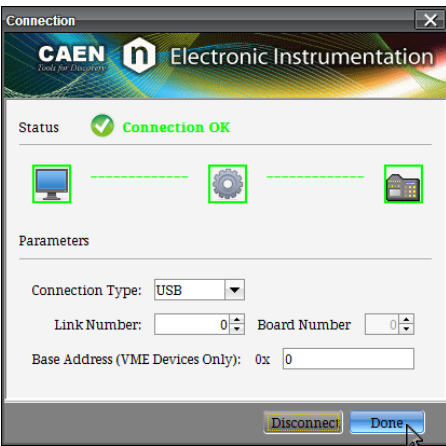


Connection chain	Type	Link	Slave	Address
PC → <b>USB</b> → DT5720 / N6720	USB	0	0	0
PC → <b>USB</b> → V1718 → <b>VME</b> → V1720	USB	0	0	32100000*
PC → <b>PCI/PCIe</b> → A2818/A3818 → <b>CONET</b> → DT5720 / N6720	PCI	0	0	0
PC → <b>PCI/PCIe</b> → A2818/A3818 → <b>CONET</b> → V1720	PCI	0	0	32100000*

**Tab. 5.1:** Examples of connection settings

(\*)For the correct VME base address to be used, please refer to the Digitizer’s User Manual.

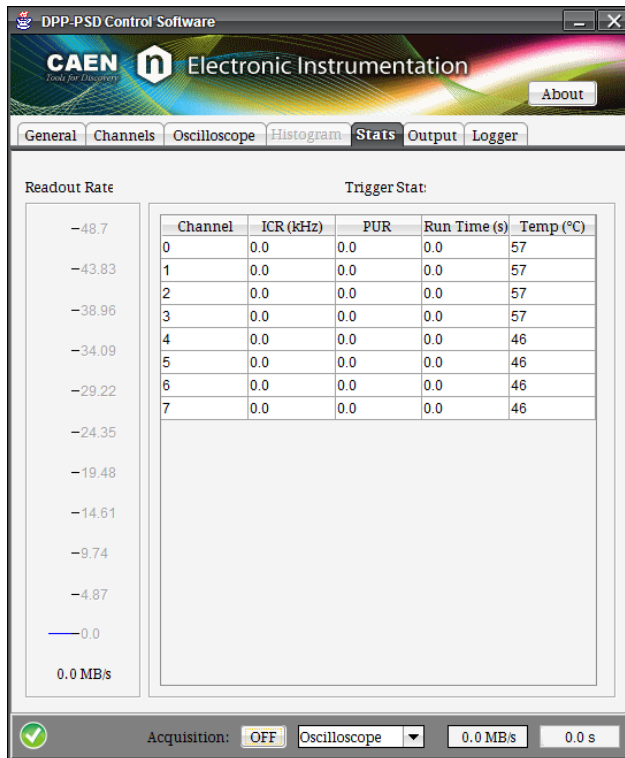
**Action3:** click the button **CONNECT** and verify that the connection Status turns on green (i.e. Connection OK), then click the button **DONE**.



#### Action4: calibration of 725, 730 and 751 series.

In case of 725, 730 and 751 digitizer families it is required to calibrate the ADCs for a correct digitalization of the samples.

Wait few minutes after the power on to thermalize the ADCs and check the temperature sensors from the “Stats” tab. There is one sensor for each ADC, i.e. one sensor each four channels for the 725 and 730 series, and one sensor each two channels for 751 series.



**Note:** For 725 and 730 series only, in case the temperature exceeds 70 °C there is an automatic shutdown of the ADCs. Refer to User Manual of each the specific form factor for further details.

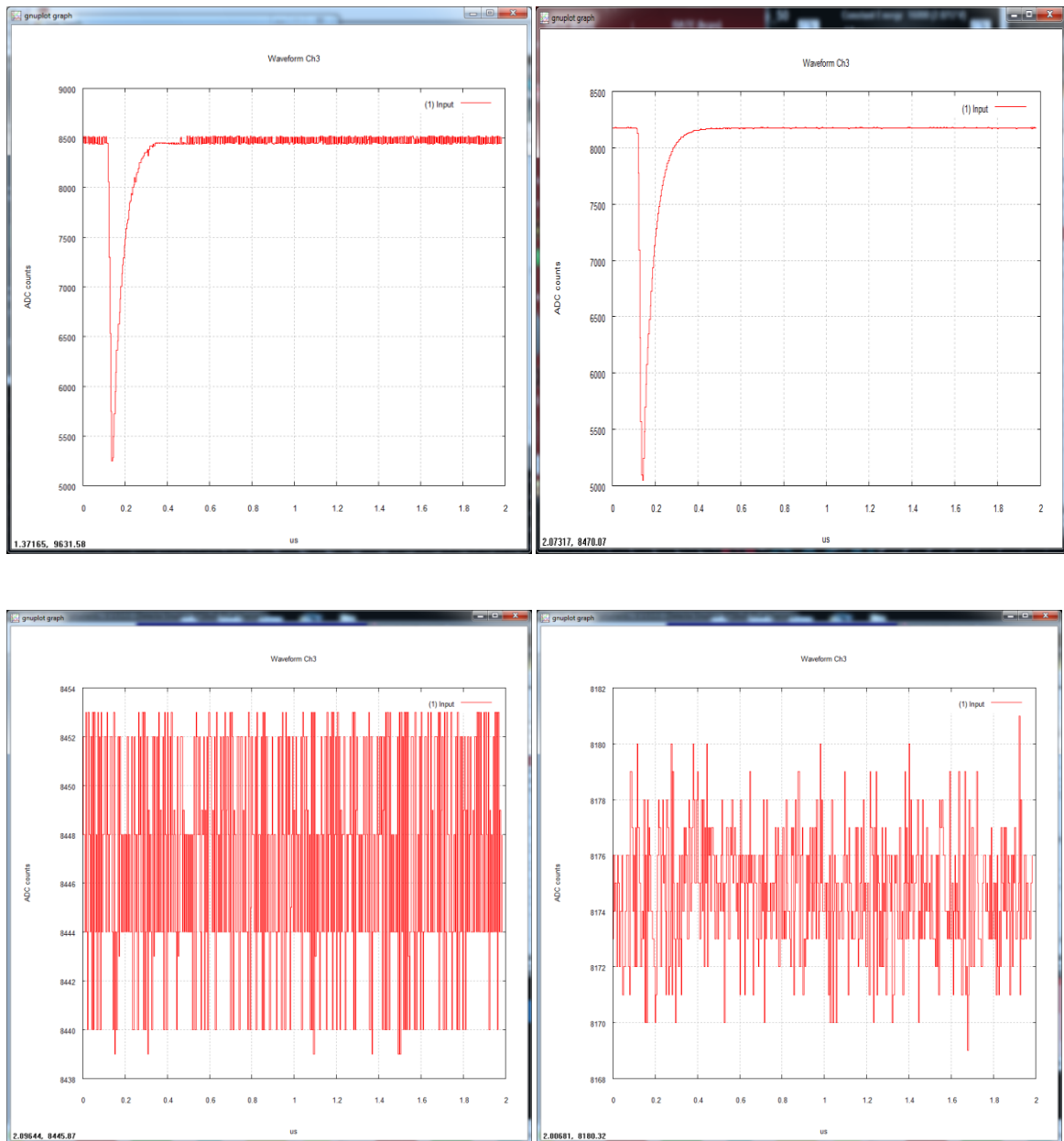
Once the temperature is stable go to the “General” tab and press “Calibrate”



The Digitizer is now ready to start the acquisition.



**Note:** In case the calibration is not performed, the ADC samples may appear noisy and meaningless, as in the following pictures. Here a couple of examples of calibration not performed. On the left the samples before the calibration, on the right the samples after the calibration. Note the different vertical scale.



3. Set Oscilloscope mode, enable Channel 0, check signal polarity and start acquisition.

**Path1:** Tab **GENERAL** → Section **ACQUISITION SETTINGS**

**Action1:** set **ACQUISITION MODE** on “Oscilloscope” using the scroll menu box.



**Path2:** Tab **CHANNELS** → Field **CHANNEL SETTINGS FOR** → Section **GENERAL SETTINGS**

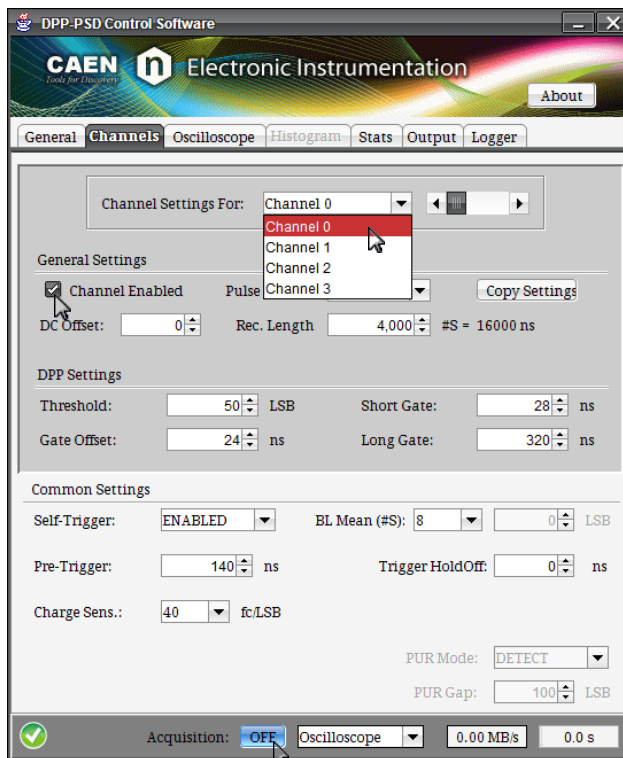
**Action1:** Select “Channel 0” using the scroll menu box or the side bar.

**Action2:** check “Channel Enabled” so that the settings in the tab are active for the selected channel.

**Action3:** press the **ACQUISITION** button in the deep grey common bar at the bottom of the GUI:

- **OFF** = acquisition is off.
- **ON** = acquisition is on.

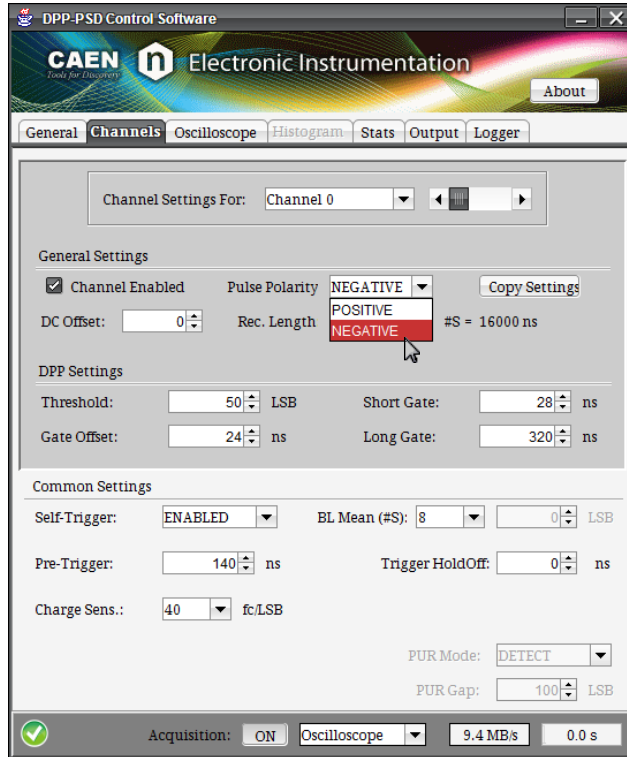
Check that the “Run” green LED on the Digitizer’s front panel lights on.



**Action4:** set the **PULSE POLARITY** on “**Negative**”, according to the polarity of the input signal from the detector.



**Note:** The algorithm is internally designed to work with negative pulses. When the analog input pulse is positive, it is necessary to invert it before the DPP algorithms are applied. The option PULSE POLARITY = “Positive” enables the internal inversion of the pulse polarity. Please notice that the inversion is applied to the DPP algorithms only, while it doesn’t affect the waveform recording (both plots and output files keep the original pulse polarity).



## 4. Set the parameters for self-triggering (DC Offset, Baseline, Threshold).

With the acquisition on, since it is not guaranteed that the channel is properly triggering on the input pulses, the Software Trigger is used to force the acquisition. It is so possible to adjust parameters like the DC Offset, the Baseline and the Threshold to enable the Digitizer to self-trigger.

**Path1:** Tab **GENERAL** → Section **ACQUISITION SETTINGS**

**Action1:** Press the **SOFTWARE TRIGGERS “ON/OFF”** button to enable a software trigger to be continuously issued.

- **OFF** = Software Trigger is disabled.
- **ON** = Software Trigger is enabled.

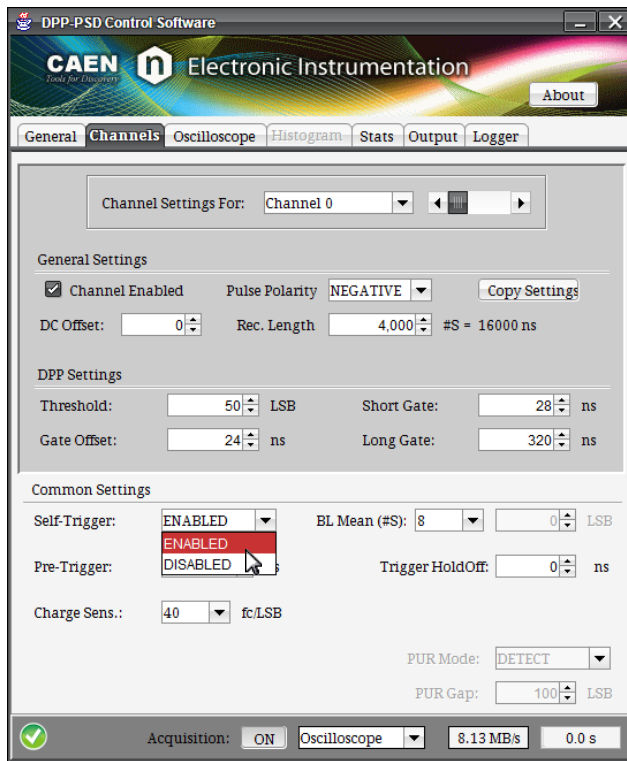


Make sure that the self-trigger option is enabled. Otherwise, perform the **path2** below.



**Path2:** Tab **CHANNELS** → Section **COMMON SETTINGS**

**Action1:** set **SELF-TRIGGER** on “**Enabled**” in the using the scroll menu box.



Now it is worth plotting the input signal to estimate the DC Offset adjustment.

**Path3:** Tab **OSCILLOSCOPE** → Section **OSCILLOSCOPE PLOT**

**Action1:** set “Channel 0” in the **CHANNEL** list.

**Action2:** check the “Wave” box as **PLOT MODE**.

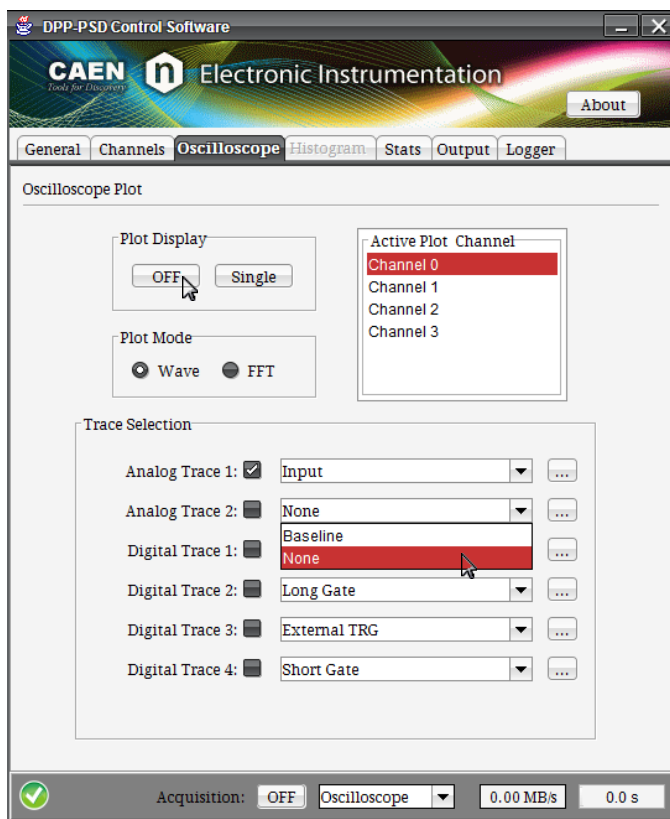
**Action3:** disable the **DUAL TRACE** mode. From “Analog Trace 2” select the “NONE” option. This will ensure that all the samples are used for the Input signal digitalization. When the “Baseline” option is active, half of the samples are used for the Baseline trace, and the waveform will appear at half of the sampling frequency. Refer to Sect. **Event Data Format**.

**Action4:** check **ANALOG TRACE 1** and select “Input” in the scroll box. The input pulse from the Detector-PMT system will be plotted.

**Action5:** disable **ANALOG TRACE 2**, **DIGITAL TRACE 1**, **DIGITAL TRACE 2**, **DIGITAL TRACE 3** and **DIGITAL TRACE 4** (DIGITAL TRACE 4 not present for the 751 series).

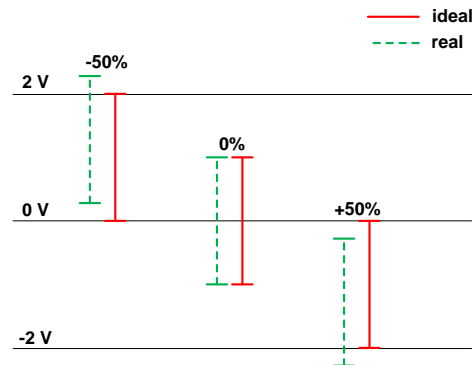
**Action6:** enable the continuous plotting by pressing the **PLOT DISPLAY “ON/OFF”** button:

- **OFF** = Plot displaying is off.
- **ON** = Plot displaying is on.



The DC Offset is a DC value added to the input signal at the input stage of the Digitizer to fit the signal dynamic range to the ADC input dynamics.

The DC Offset parameter is expressed in percentage of the Digitizer's ADC input dynamics and ranges between -50 and +50 (%). Theoretically, the value of 0 (DC Offset = "0") means an input pulse DC level set to half of the ADC dynamics (i.e. 2048 counts for the 12-bit and 2 V input range DT5720). The value of "50" (DC Offset = "50") sets the DC level at the lower dynamics limit (i.e. 0 counts), while the value of "-50" (DC Offset = "-50") sets the DC level at the upper dynamics limit (i.e. 4095 counts). The real DC offset adjustment implemented in the DPP-PSD firmware is shown in **Fig. 5.4**: in order to preserve from saturation the input signals near the dynamics boundaries, setting the DC offset to +50 or -50 puts the signal baseline respectively a step up the upper boundary and a step under the lower boundary.



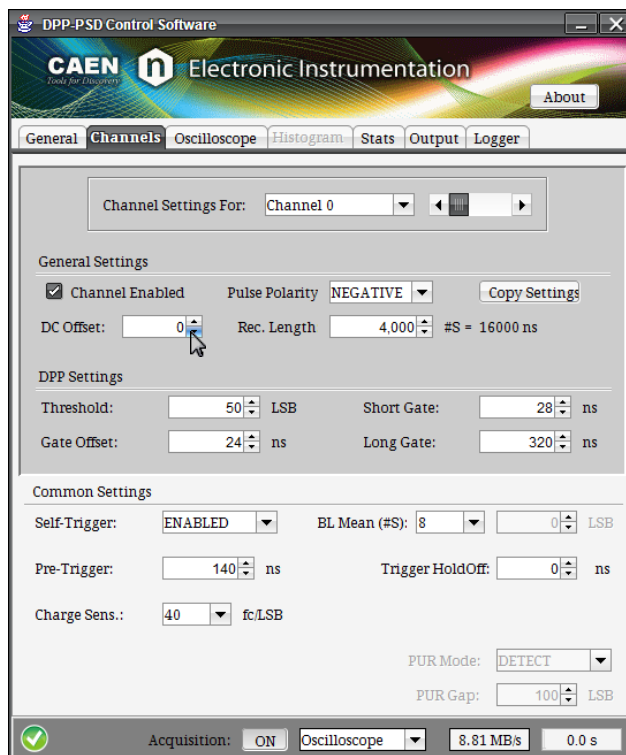
**Fig. 5.4:** Input signal DC offset adjustment description

**Action7:** look at the DC level of the input pulse (DC offset) in your plot to check if an adjustment is needed according to the Digitizer's ADC dynamic range. In the specific case, we chose to set the DC offset around half the dynamics (theoretically 2048 counts).

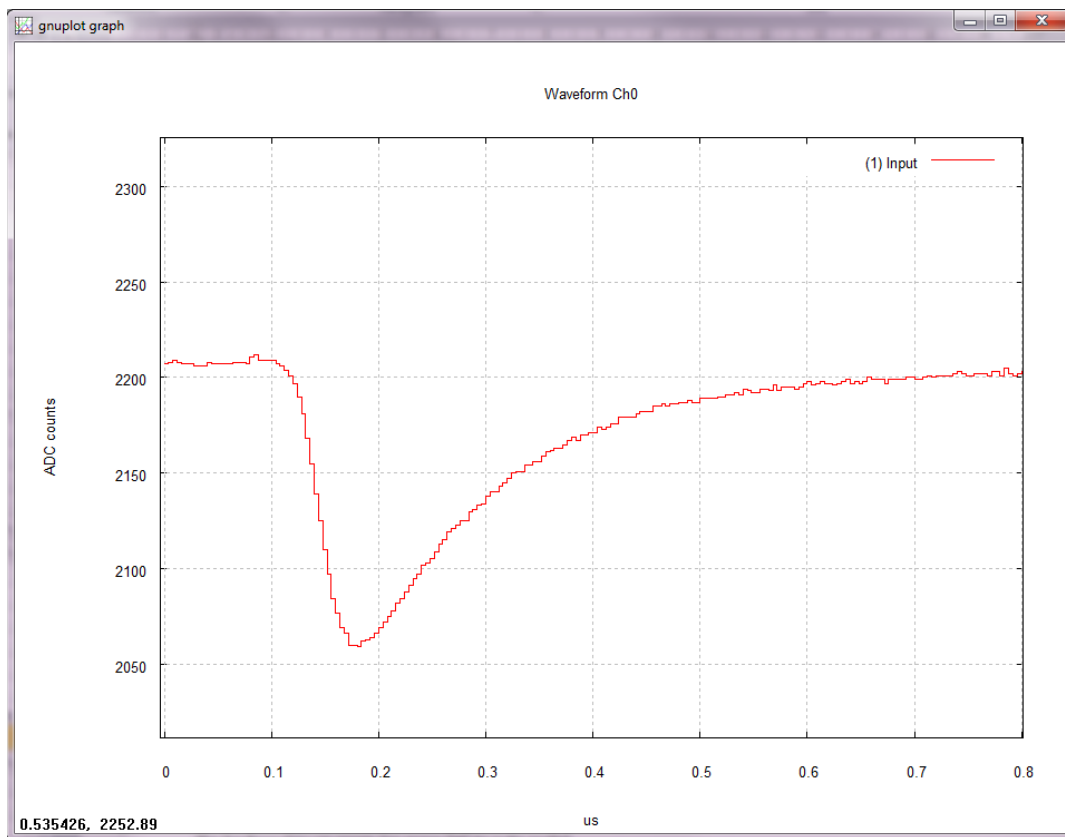
**Path4:** Tab **CHANNELS** → Section **GENERAL SETTINGS**

**Action1:** verify that the "Channel Enabled" box is checked.

**Action2:** type or set "0" in the **DC OFFSET** box menu.



**Action3:** check the effect of the previous settings in the plot window.



**Note:** The plot above has been zoomed. To do that, right click on the plot in a point near the portion of the signal you want to zoom, then release the mouse button, move to a point on the opposite corner and left click. Press “u” key on the keyboard to un-zoom (or press “a” to auto scale).

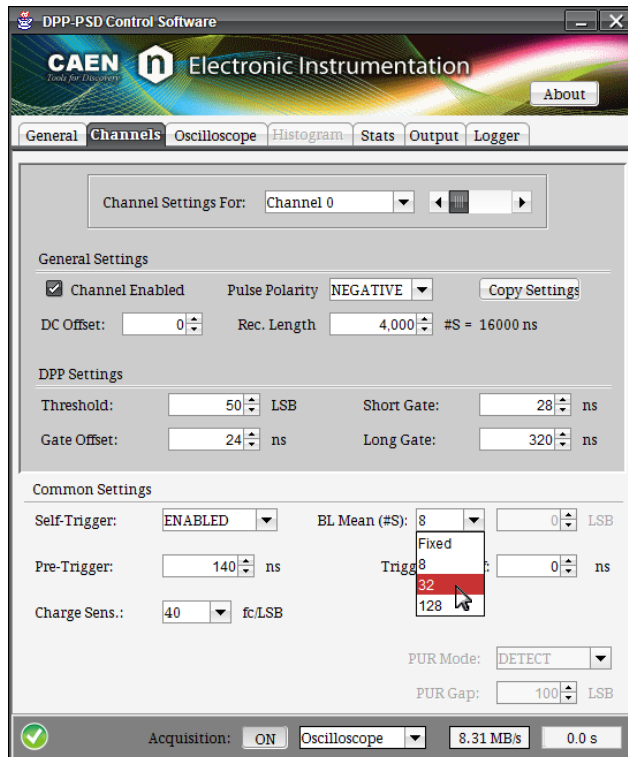
Next step is to set the input signal baseline calculation. Both the Threshold parameter, i.e. the trigger threshold, and the charge integration are referred to the baseline value.

The Control Software provides two options for setting the baseline:

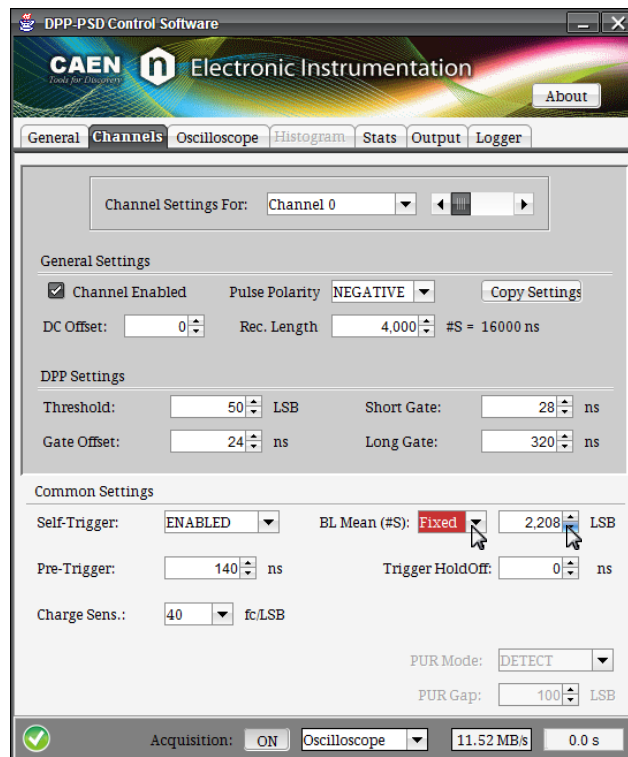
- Baseline mean calculation, where the DPP-PSD algorithm calculates *online* the input signal baseline through a mean filter over a number of samples set by the BASELINE MEAN parameter.
- Absolute baseline, where a fixed baseline value is set by the ABSOLUTE BASELINE parameter.

**Path5:** Tab **CHANNELS** → Section **COMMON SETTINGS**

**Action1:** set the **BASELINE MEAN** to “32” in the scrolling box menu. Those values correspond to the number of samples used in the baseline calculation.



**Action2:** set the **BASELINE MEAN** to “Fixed” in the scrolling box menu and set the absolute value for the baseline according to the waveform plot.

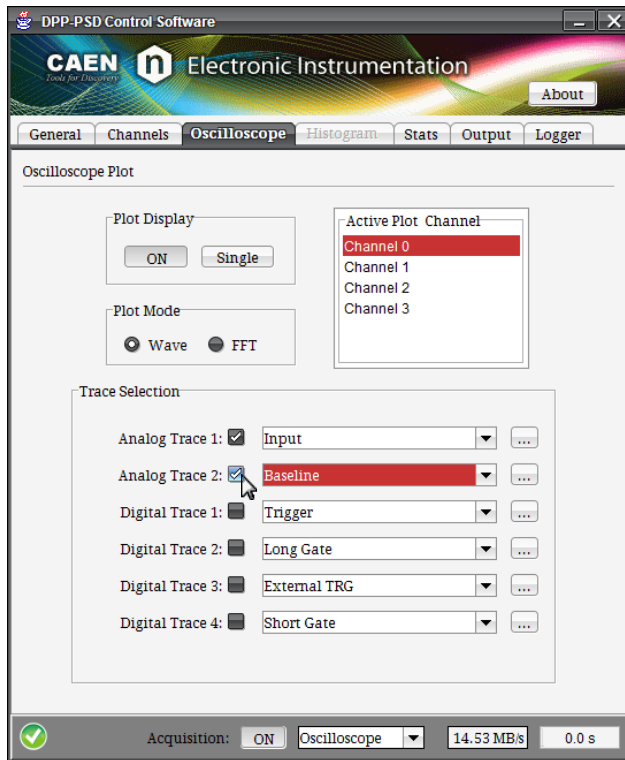


The user must choose either one of the two methods.

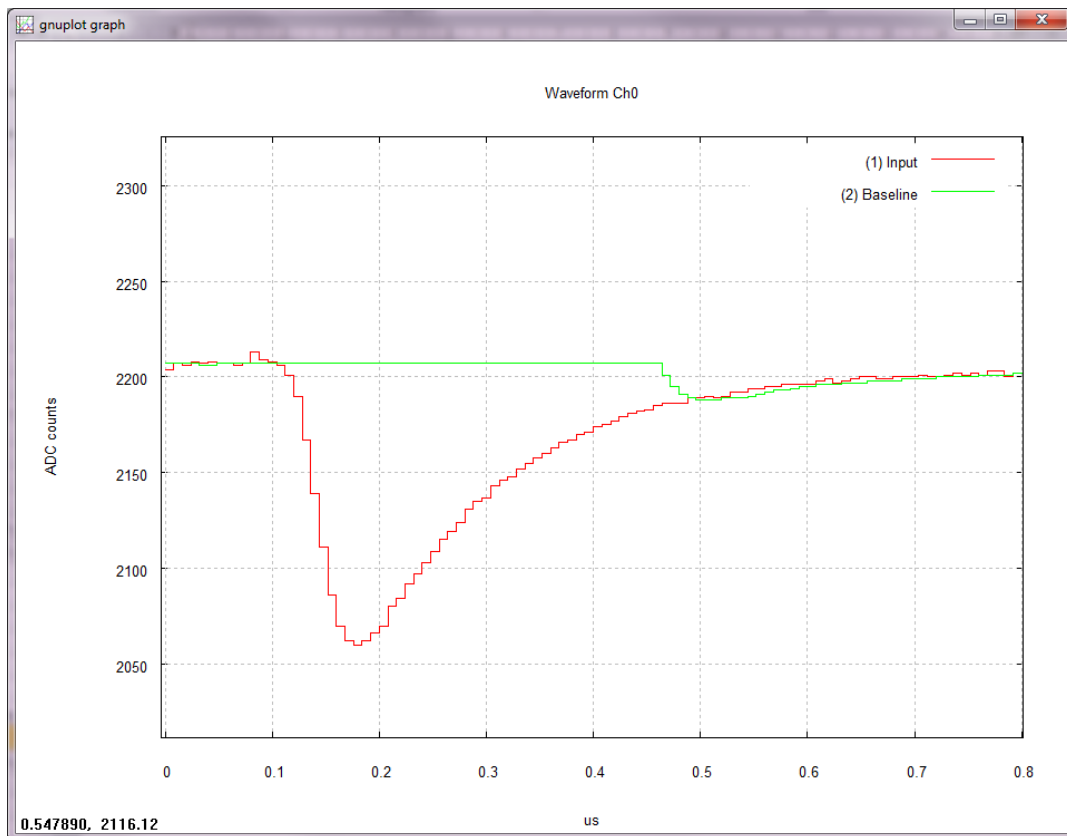
The plotting of the input signal and the baseline can help to check the effect of the setting.

**Path6:** Tab **OSCILLOSCOPE** → Section **OSCILLOSCOPE PLOT**

**Action1:** enable **ANALOG TRACE 2** check box and select “**Baseline**” in the scroll menu. In this way half of the samples are used for the second analog trace, thus the Input pulse will appear at half of the sampling rate of the digitizer.



When selecting the automatic baseline calculation the oscilloscope plot will appear as follows:

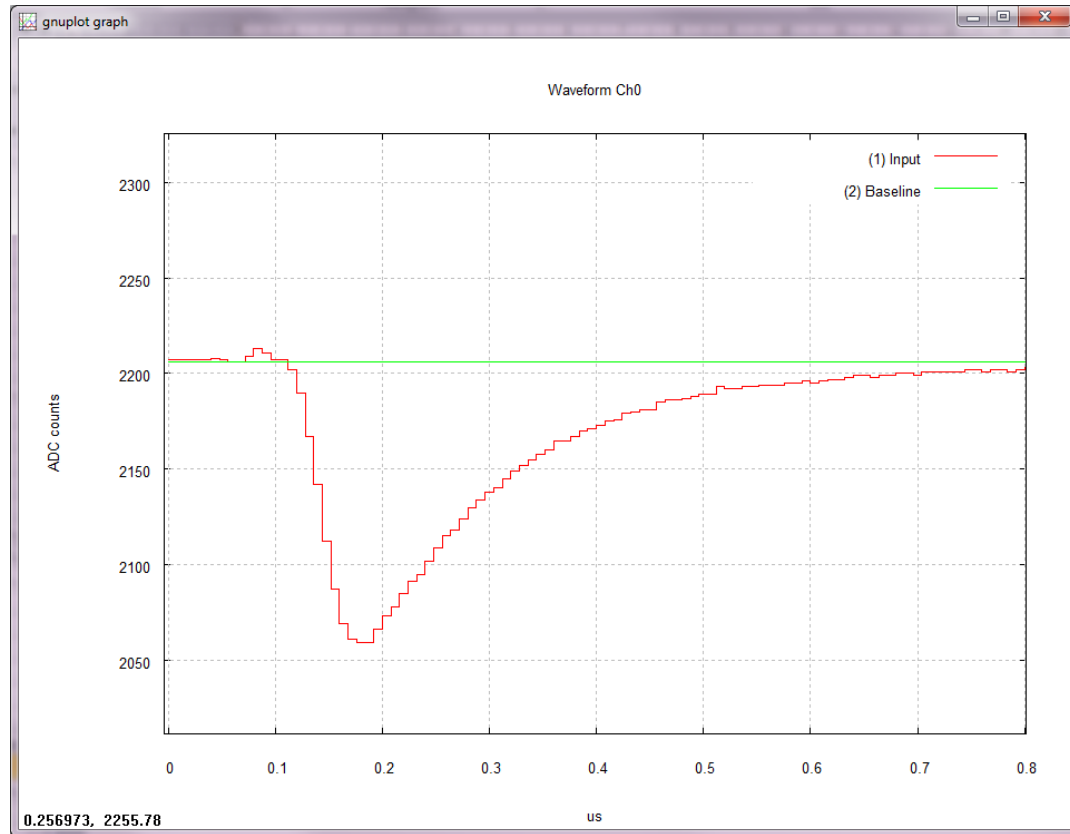


The baseline remain frozen for the whole duration of the maximum value between the gate and the trigger hold-off parameters. To freeze it for the whole signal width you must adjust the gate and trigger hold-off parameters. Instructions on how to change those parameters are explained in the following steps.



**Note:** For 751 series, the baseline freeze lasts for a longer time than the greater value between Long Gate and Trigger Hold-off.

When choosing the fixed value for the baseline, the baseline remains frozen for the whole acquisition window, as you can see from the following plot.



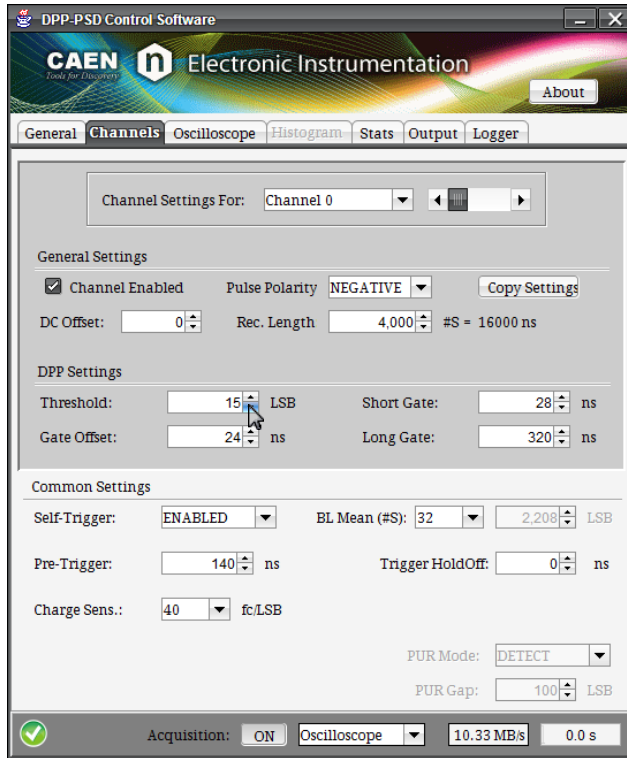
**Note:** The oscilloscope plots above have been displayed using the auto scale function of *gnuplot* (press “a” on your keyboard to activate the auto scale).

We choose the first method, i.e. the automatic baseline calculation. It is now possible to set the trigger threshold. For this purpose, the parameter THRESHOLD is defined as the relative absolute value of the trigger threshold with respect to the baseline.

Once the baseline calculation has been fixed, it is possible to set the trigger threshold. For this purpose, the parameter **THRESHOLD** is defined as the relative absolute value of the trigger threshold with respect to the baseline value.

**Path7:** Tab **CHANNELS** → Section1 **DPP SETTINGS** → Section2 **COMMON SETTINGS**

**Action1:** set a value of **THRESHOLD** in the box menu according to the noise level of the input signal baseline. In our example we fix “15” LSB as threshold level.





**Path8:** Tab **GENERAL** → Section **ACQUISITION SETTINGS**

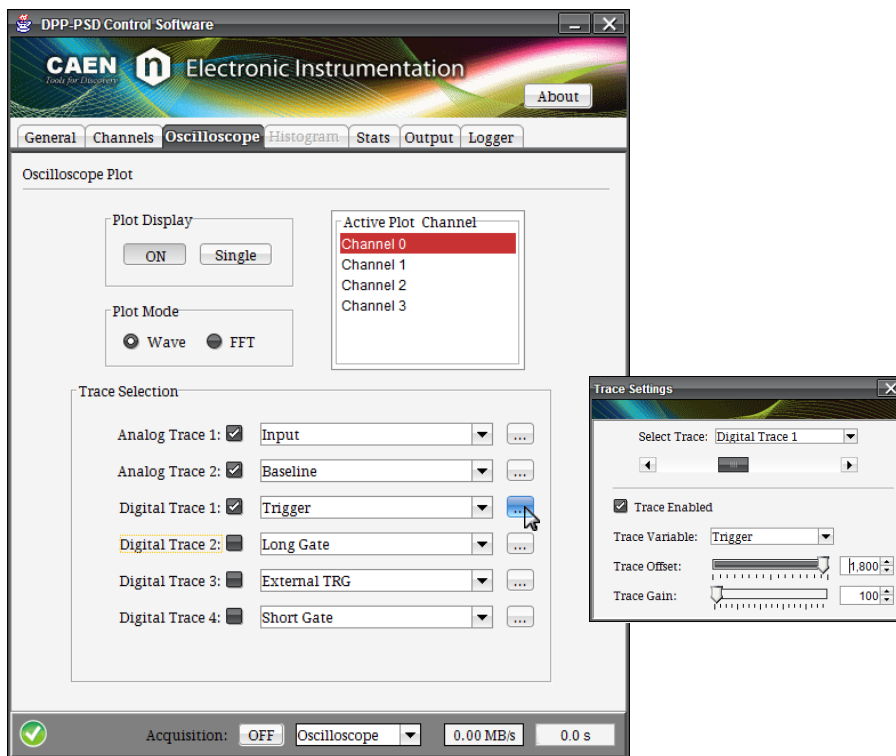
**Action1:** Disable Software Trigger by the “ON/OFF” button. You should observe the board going on self-triggering.

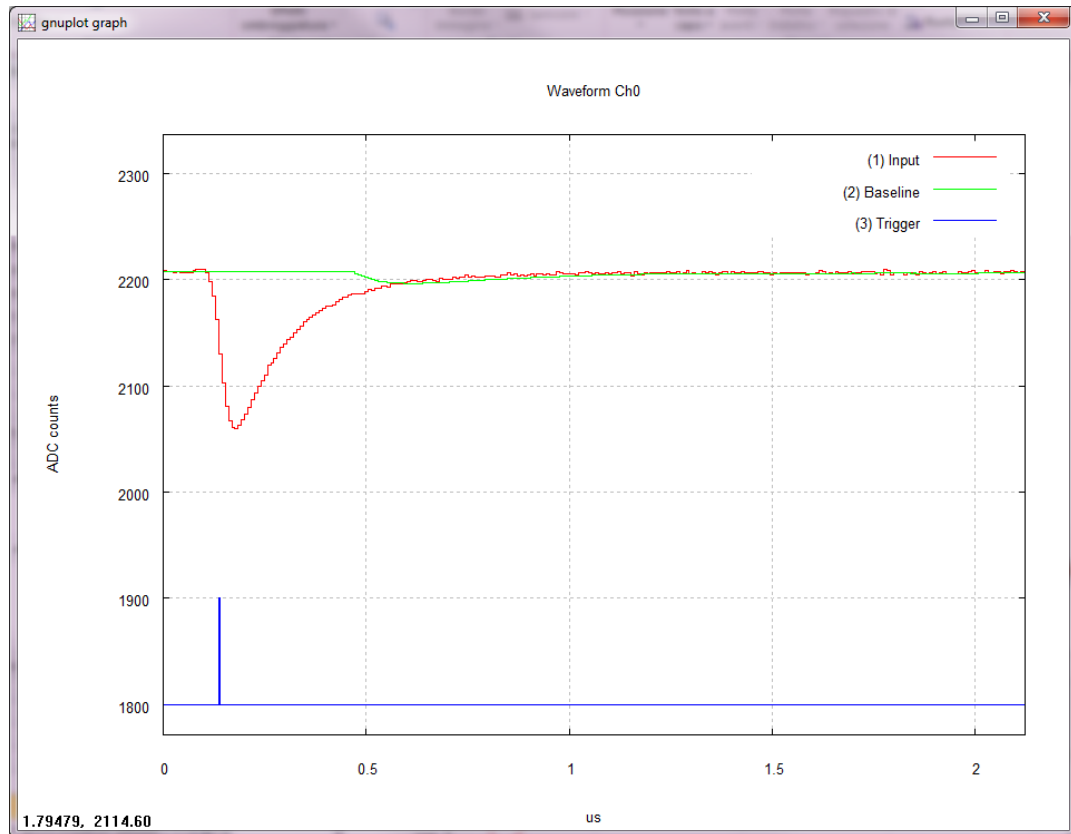
**Path9:** Tab **OSCILLOSCOPE** → Section **OSCILLOSCOPE PLOT**

**Action2:** Enable **DIGITAL TRACE 1** and select “Trigger”. Press the “...” button and use the options in the “Trace Settings” window to set the proper digital offset and gain to apply to the trace.



**Note:** in case of 725 and 730 the Trigger trace can be selected both from “Digital Trace 1” and “Digital Trace 2”





You can look at the plot if the trigger is properly issued.

### 5. Set the parameters for the charge integration (gate offset, short gate, long gate)

Setting the SHORT GATE and the LONG GATE widths enables the firmware to integrate the input pulse and calculate the charges  $Q_{\text{short}}$  and  $Q_{\text{long}}$ . We will increase the GATE width to integrate the whole width of the signal <sup>3</sup>. First enable the gate visualization in the oscilloscope plot.

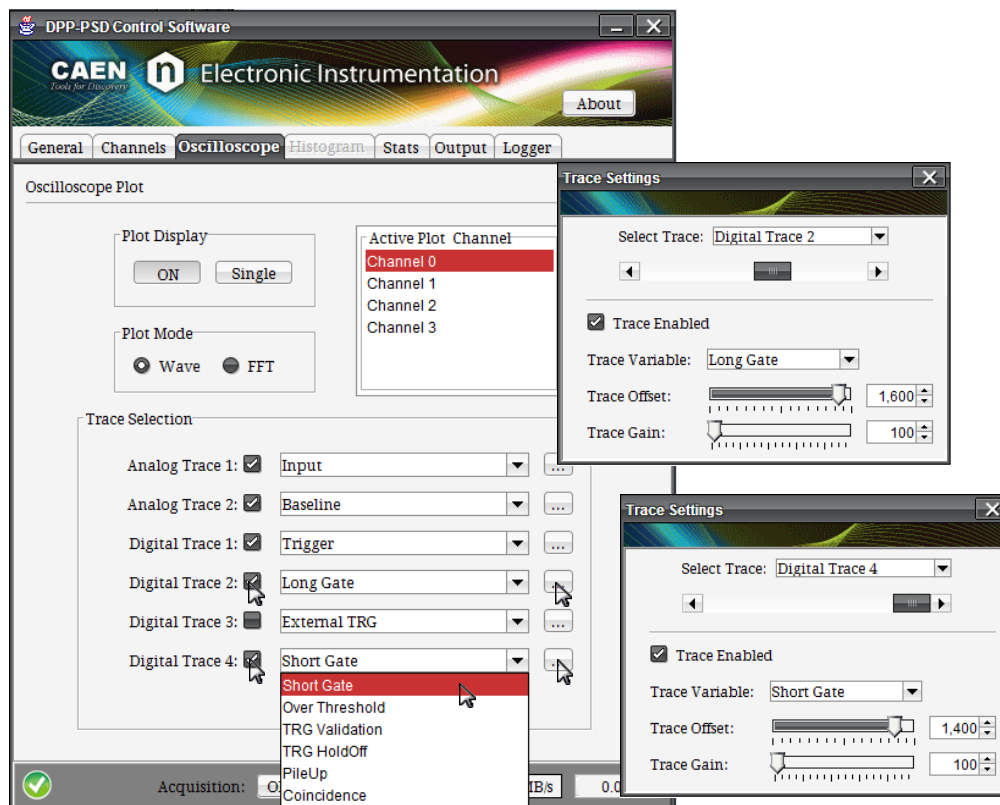
**Path1:** Tab **OSCILLOSCOPE** → Section **OSCILLOSCOPE PLOT**

**Action1:** Enable **DIGITAL TRACE 2, “Long Gate”**, and enable **DIGITAL TRACE 4**, selecting “Short Gate” (i.e. DIGITAL TRACE 3 in case of 751 series).

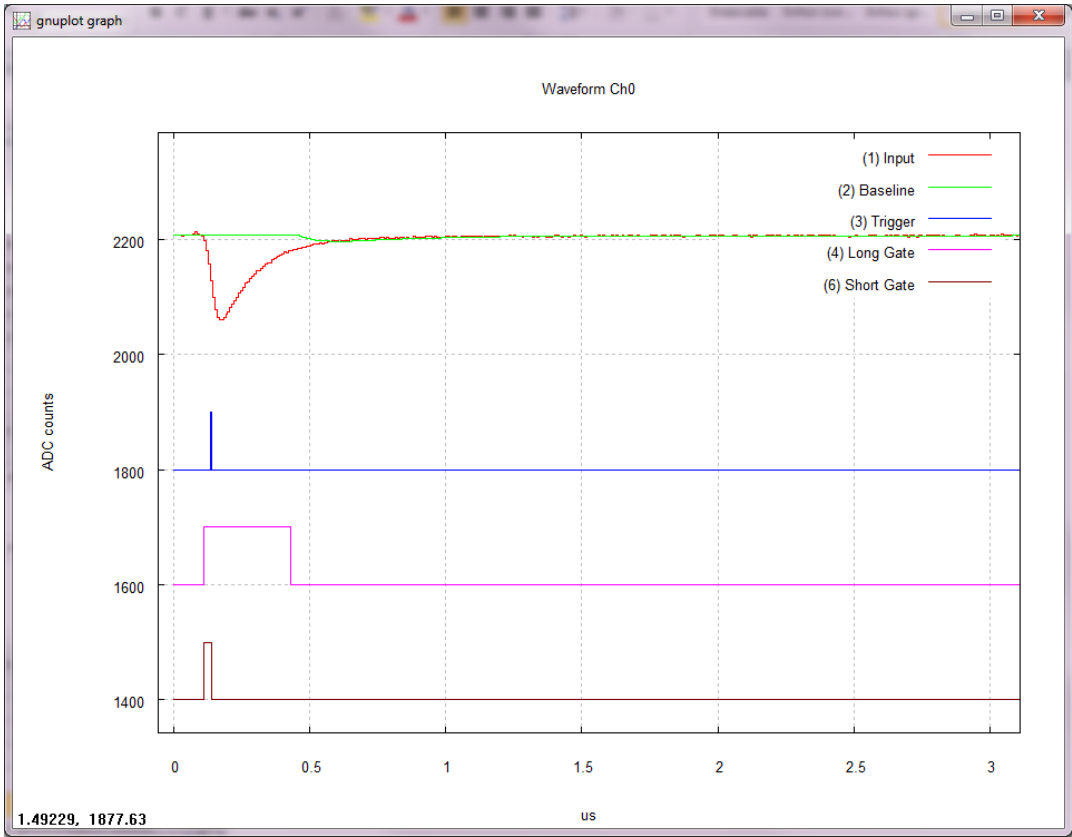


**Note:** In case of 725 and 730 select the “Long Gate” from “Digital Trace 1”, and “Short Gate” from “Digital Trace 2”.

With the default values of short and long gate the histogram plot will appear as in the following figure.



<sup>3</sup> In [RD2], a configuration of such parameters is reported for a typical Neutron-Gamma discrimination measurement with the BC501-A detector.



**Path2:** Tab **CHANNELS** → Section **DPP SETTINGS**

**Action1:** First adjust the **GATE OFFSET** value in the box menu. This value corresponds to the number of ns the gate will start before the trigger. Indeed the input signal is delayed by the “**Pre-Trigger**” value, so that the gate can start before the trigger. Gate Offset and Pre-Trigger must follow the relation:

$$Gate - Offset \leq Pre - Trigger - 32ns$$

We set the gate offset value to 20 ns.

For 751 series

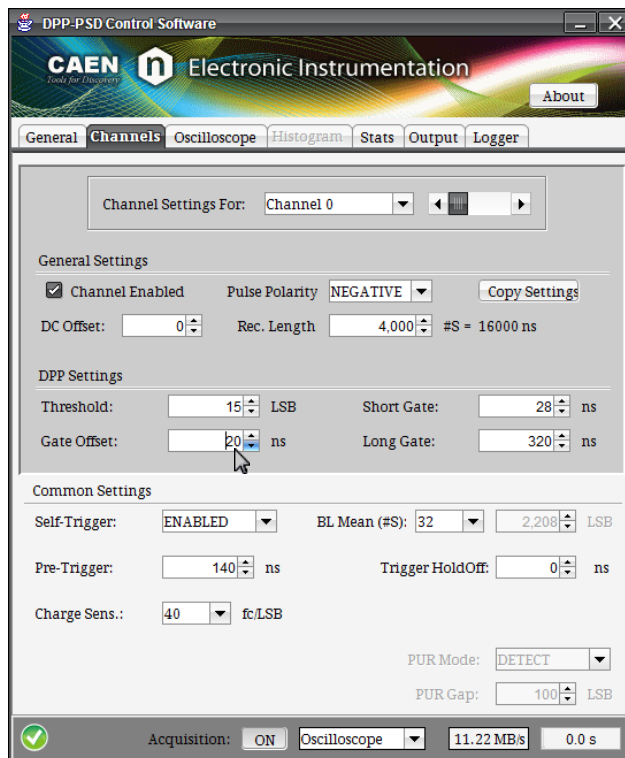
$$Gate\_Offset \leq Pre\_Trigger - 8ns$$



**Note:** In case of 725 and 730 series, if too short values of the Pre Trigger are set, the firmware automatically adjusts them to the minimum correct value.

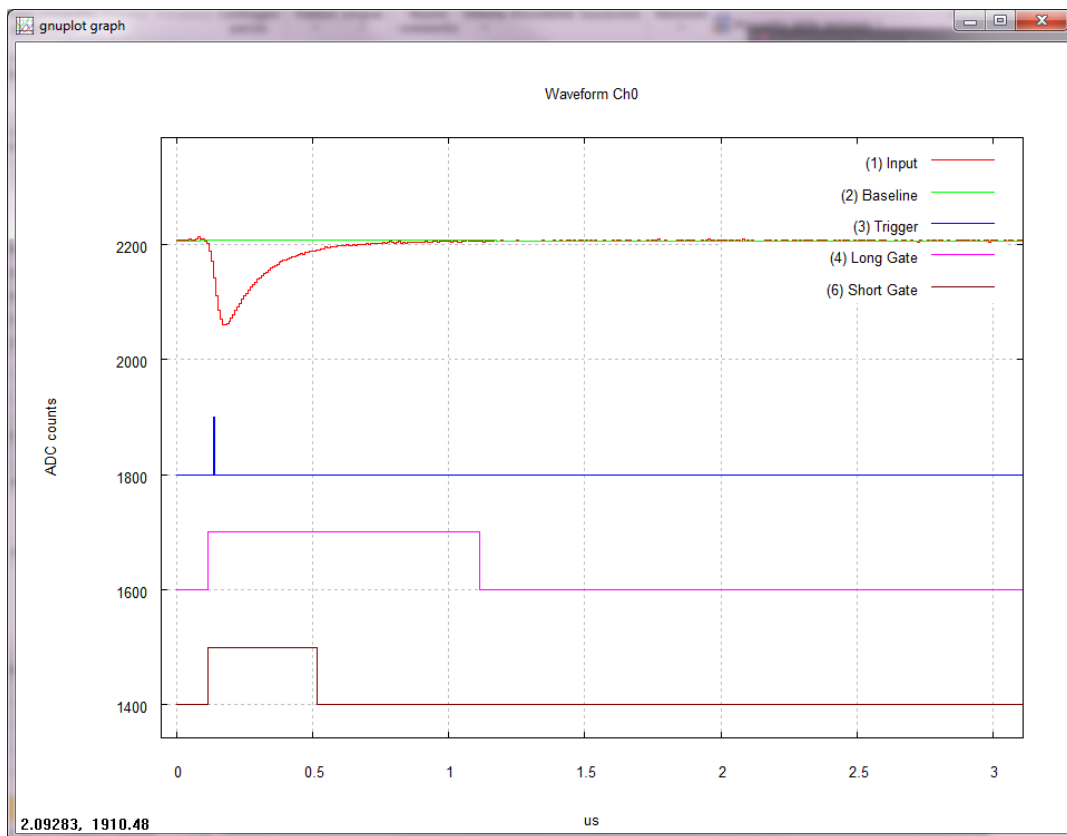
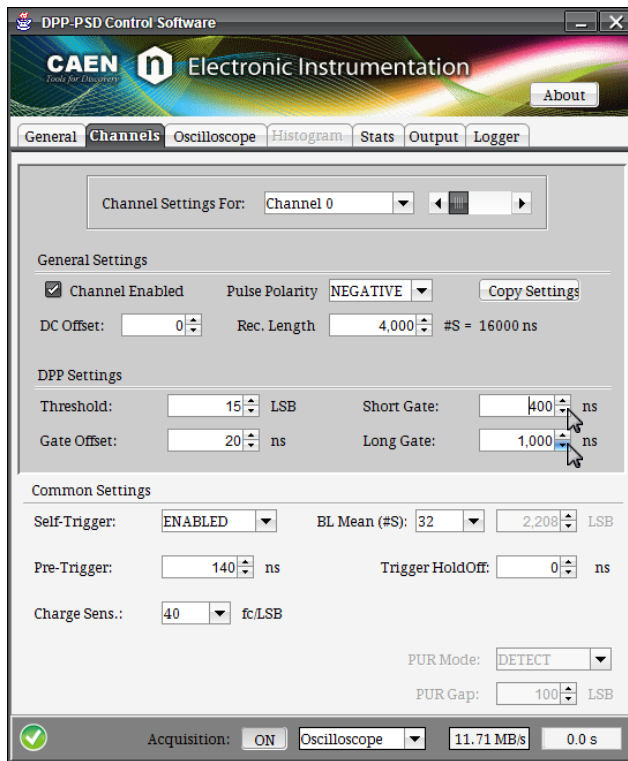


**Note:** When switching to the histogram mode the Pre-Trigger value is automatically set to the minimum allowed, i.e. Gate-Offset + 32 ns for 720 series (DT5790), Gate-Offset + 38 ns for 725 and 730 series, and Gate-Offset + 8 ns for 751 series.



**Action2:** Set the Short Gate and Long Gate widths to the proper value according to the input signal. In this example we choose 400 ns for the Short Gate and 1000 ns for the Long Gate. The following relation must be satisfied:

$$Gate - Offset \leq Short\_Gate \leq Long\_Gate$$



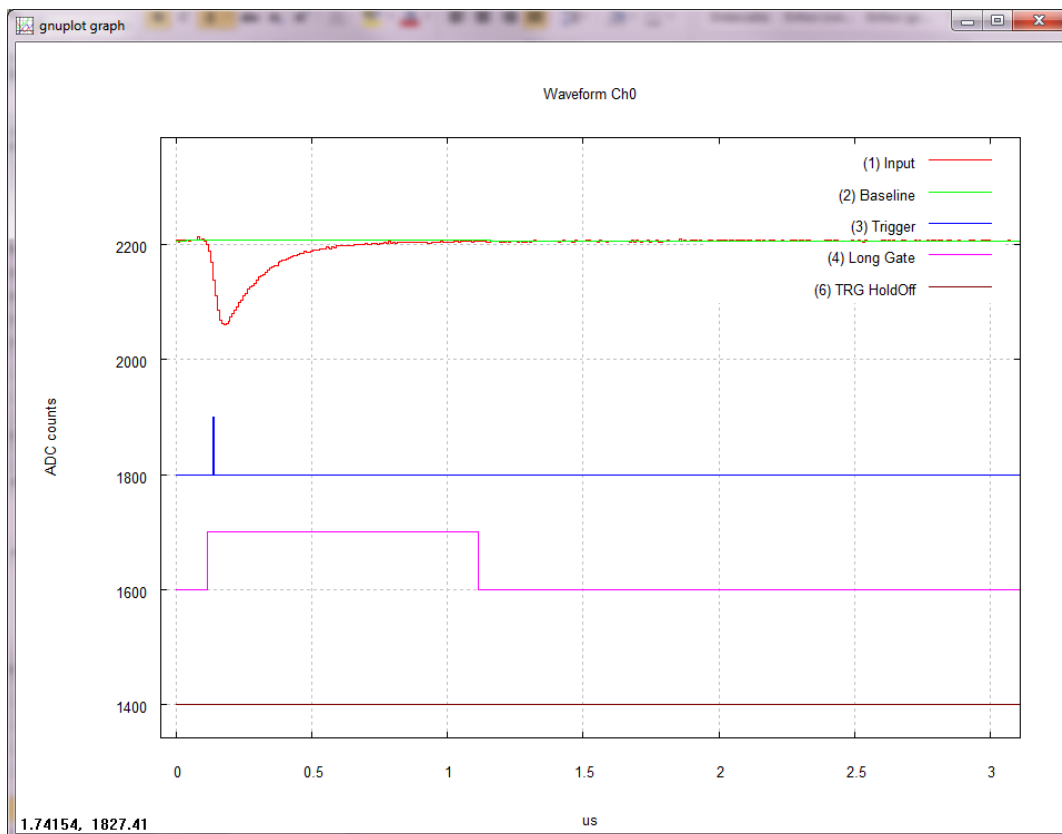
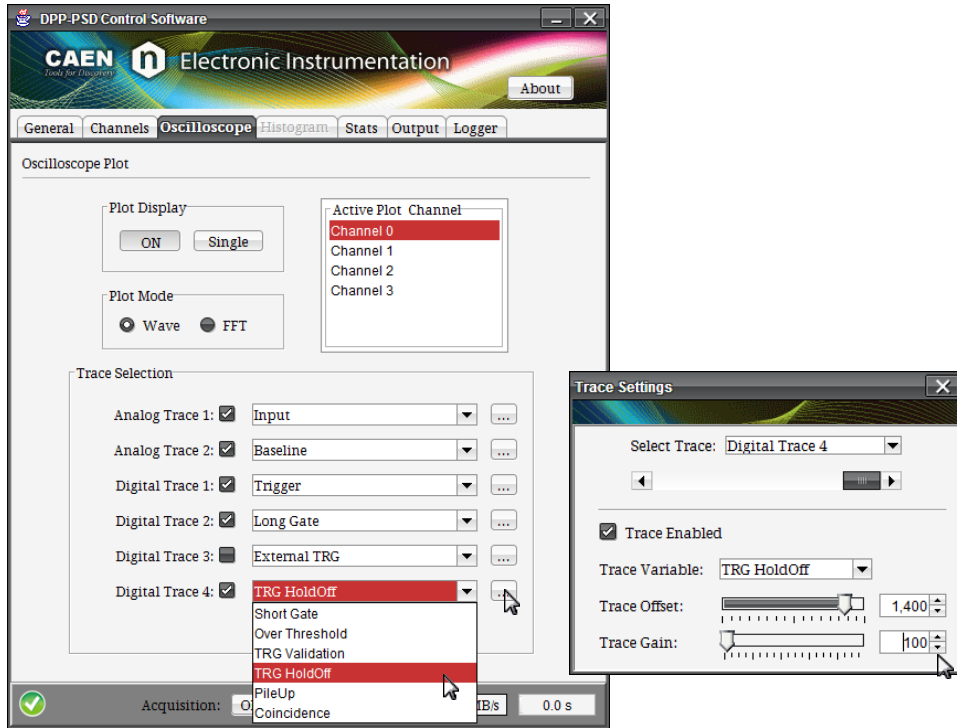
You can see that the baseline is now frozen for the whole signal width.

**6. Set the Trigger Hold-Off and Pre-Trigger parameters.**

The Trigger Hold-Off enables a time window after the trigger, where any other triggers are inhibited. Make sure to set the proper value of the Trigger Hold-Off according to your signal width, especially in case of high frequency signals.

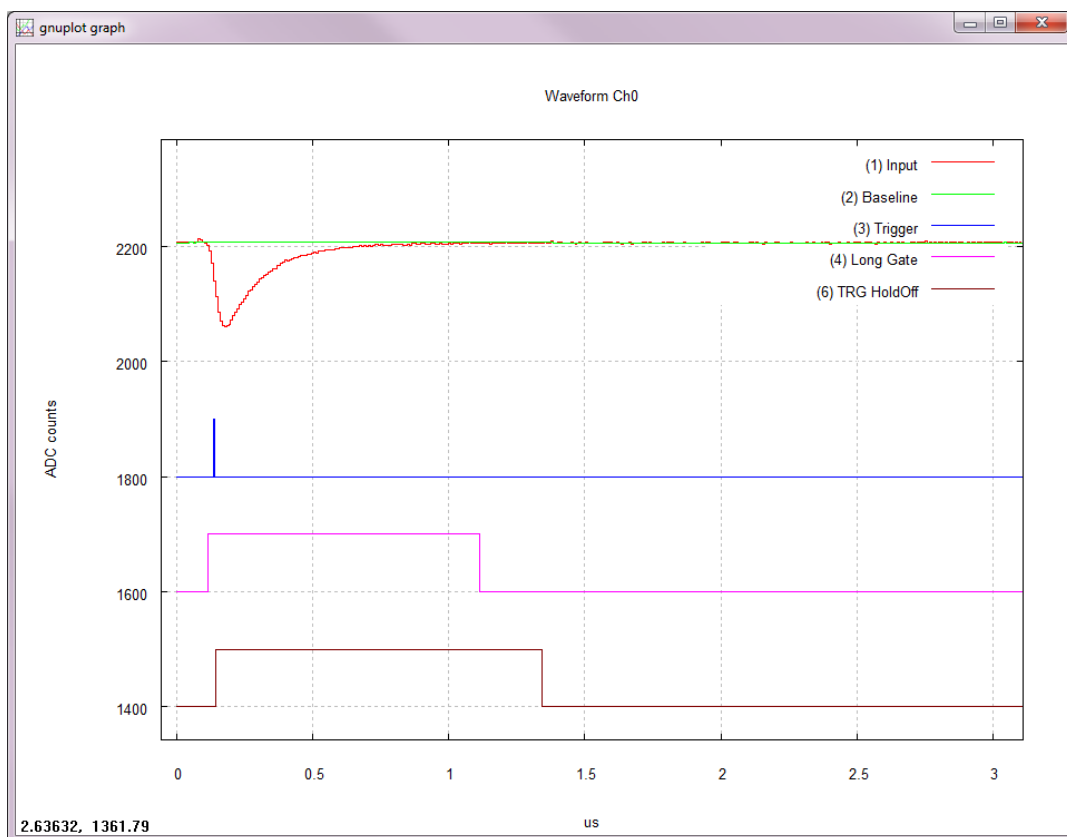
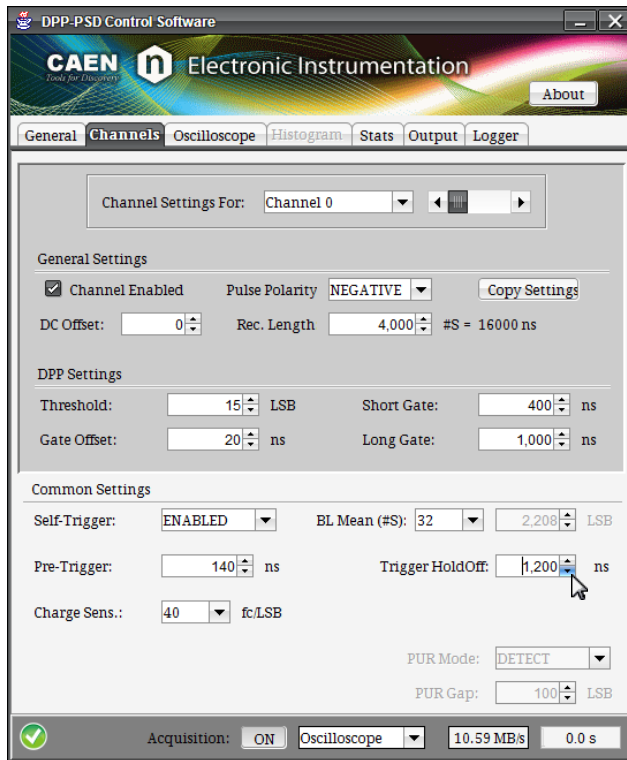
**Path1:** Tab **OSCILLOSCOPE** → Section **OSCILLOSCOPE PLOT**

**Action1:** Enable **DIGITAL TRACE 4**, and select “**TRG HoldOff**” in the menu.



**Path2:** Tab **CHANNELS** → Section **COMMON SETTINGS**

**Action1:** Set the **TRIGGER HOLD-OFF** value in the box menu. The Trigger Hold-Off goes in steps of 8 ns. We set the Trigger Hold-Off value to 1200 ns.



**Action3:** Stop the oscilloscope mode plotting through the **PLOT DISPLAY "ON/OFF"** button.



## 7. Switch to Histogram mode, plot the Energy Spectrum and the 2D-plot of Energy vs PSD.

Once the DPP-PSD parameters has been properly set, it is possible to plot the Energy Histogram of the gamma-ray source. For a complete Neutron-Gamma discrimination, the software allows for a 2D-plot of Energy vs PSD . The PSD parameter is calculated as reported in Chapter 2.



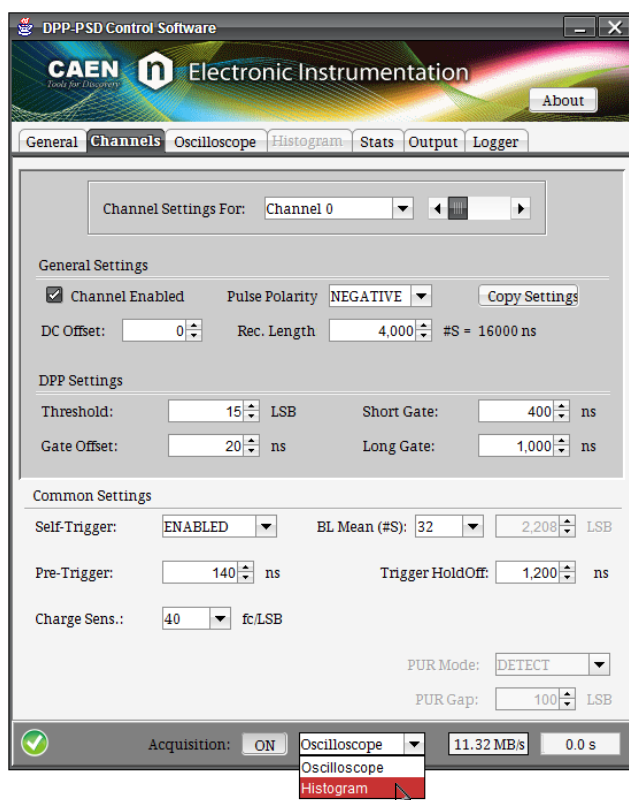
**Note:** The DPP-PSD Control Software calculates every histogram by post processing the data coming out from the Digitizer (i.e. the list of events being Time Stamp,  $Q_{short}$ ,  $Q_{long}$  and PSD).



**Note:** In the Oscilloscope acquisition mode it may happens that a lot of data are processed and transmitted, so that the Digitizer memory can go full and some data are lost. You can check it through the red “**BUSY**” LED on the Digitizer front panel. Usually this happens when you have high frequency input signals, and/or the “**RECORD LENGHT**” window is big. Conversely in the Histogram acquisition mode, the overall data throughput of the Digitizer is significantly reduced since only few data are transmitted (Trigger Time Stamp and Charge), and the busy state can disappear.

**Path1:** Any Tab.

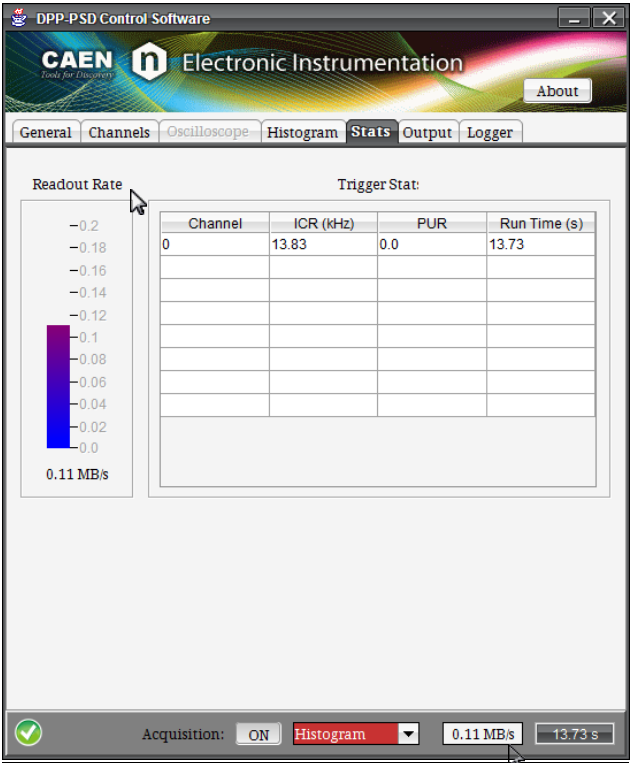
**Action1:** set **ACQUISITION MODE** on “**Histogram**” using the scroll box menu in the deep grey common bar at the bottom of the GUI.



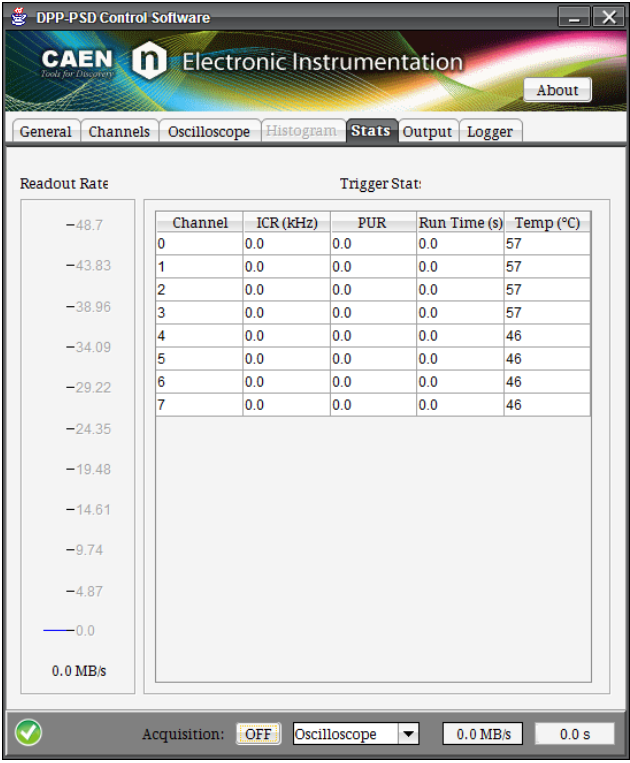
You can check that the readout rate is significantly reduced

**Path2:** Tab **STATS**

**Action1:** check the readout in the left banner of the tab, under **READOUT RATE**. The same parameter is written in the bottom box, common to all tabs.

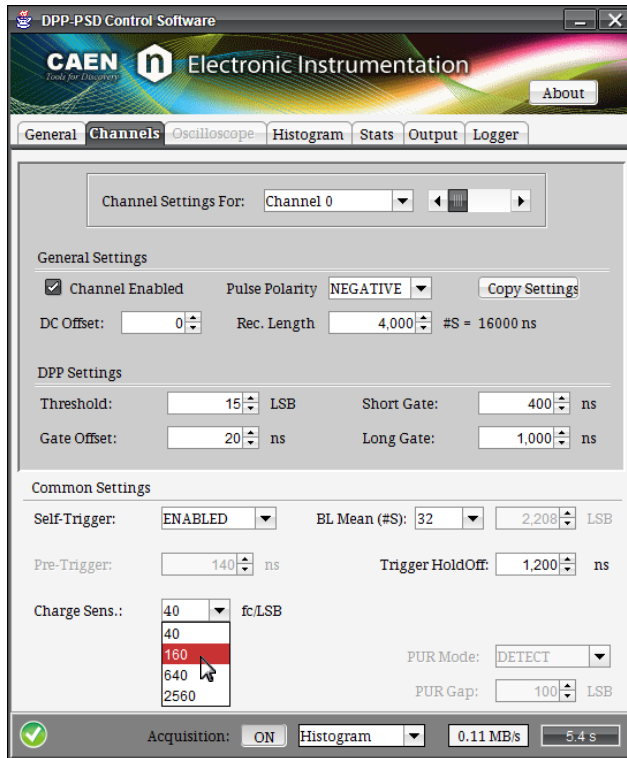


In case of 725, 730 and 751 series an additional column is available to monitor the ADC temperature. The monitor works only when the acquisition is stopped.



**Path3:** Tab **CHANNELS** → Section **COMMONS SETTINGS**

**Action1:** set **CHARGE SENS** on “160” in the box menu. The charge sensitivity allows to rescale the signal charge. This is useful especially when the charge exceeds the full-scale range (0xFFFF in 16 bits). In case of saturation only a spike corresponding to the overflow events is visible in the histogram plot.



**Path4:** Tab **HISTOGRAM** → Section **HISTOGRAM PLOT**

**Action1:** set “channel 0” in the **CHANNELS** list box.

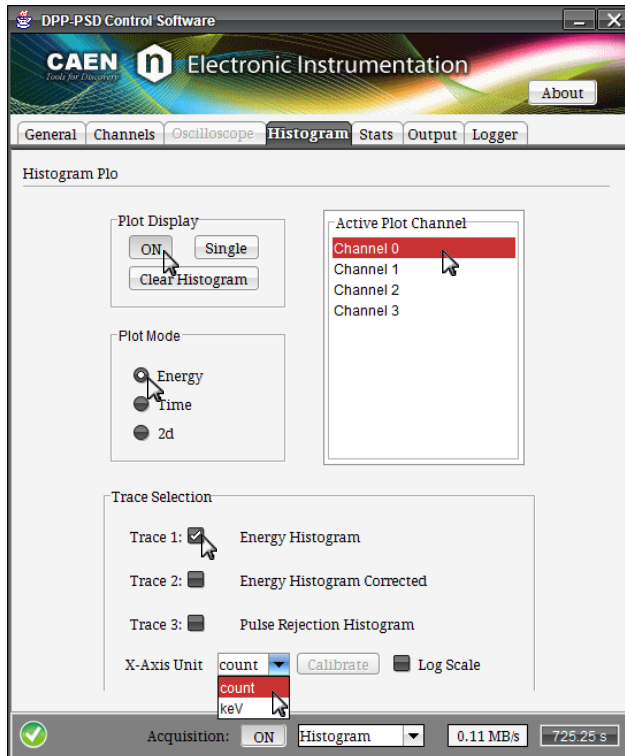
**Action2:** check “Energy” as **PLOT MODE**.

**Action3:** check “Energy Histogram” as **TRACE 1**:

**Action4:** select “count” as **ENERGY X-AXIS**. This means that the X-axis are the bin numbers in ADC counts.

**Action5:** Press the **ENERGY PLOT “ON/OFF”** button to issue the energy histogram continuous plotting.

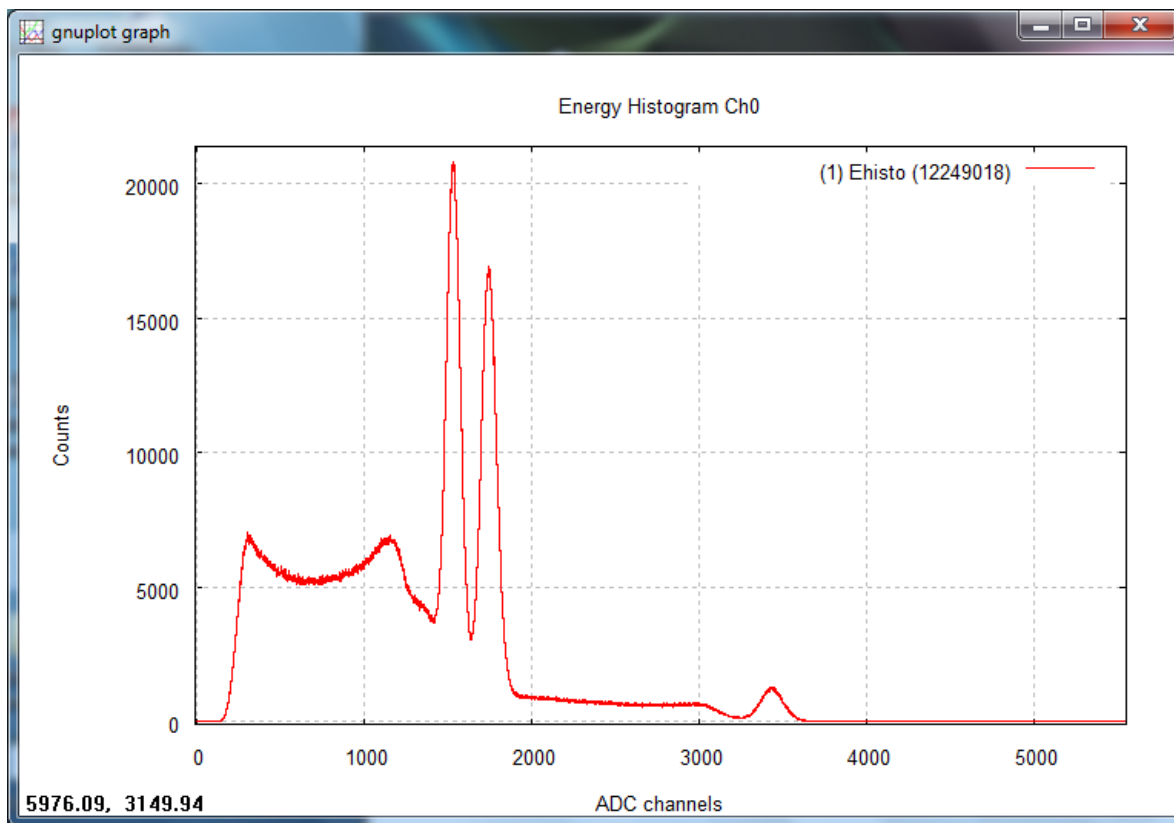
- **OFF** = Plot displaying is off.
- **ON** = Plot displaying is on.



The  $^{60}\text{Co}$  energy spectrum shows the typical two relevant peaks which should correspond to 1.33 MeV and 1.17 MeV.

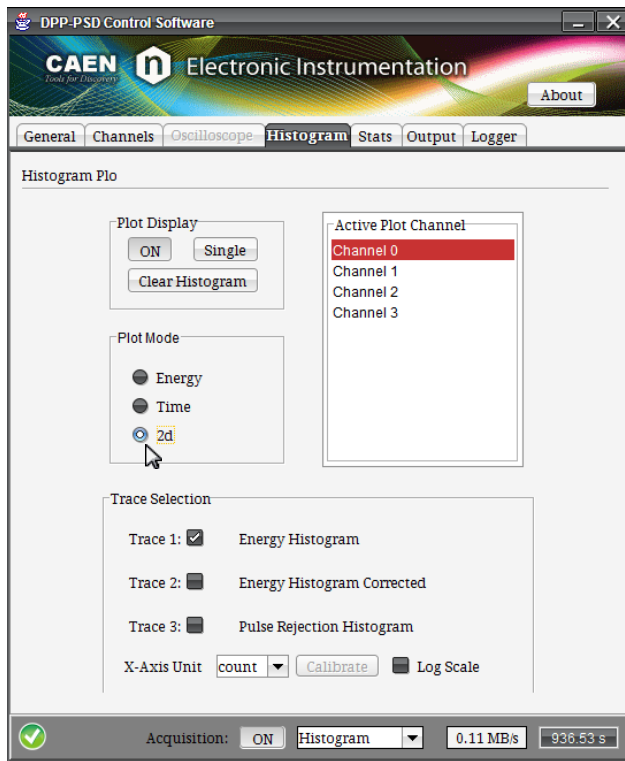


**Note:** The DPP-PSD Control Software provides only the energy histogram related to Qlong.

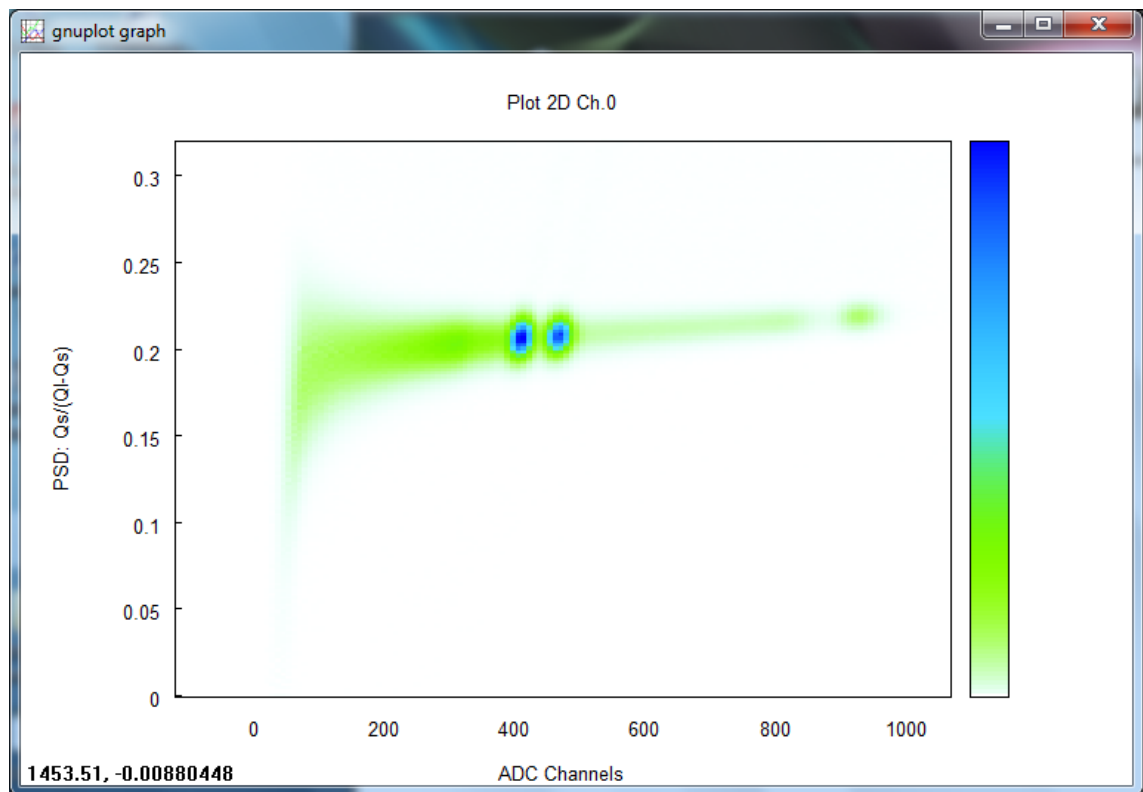


**Path3:** Tab **HISTOGRAM** → Section **HISTOGRAM PLOT**

**Action1:** check “2D” as **PLOT MODE**.

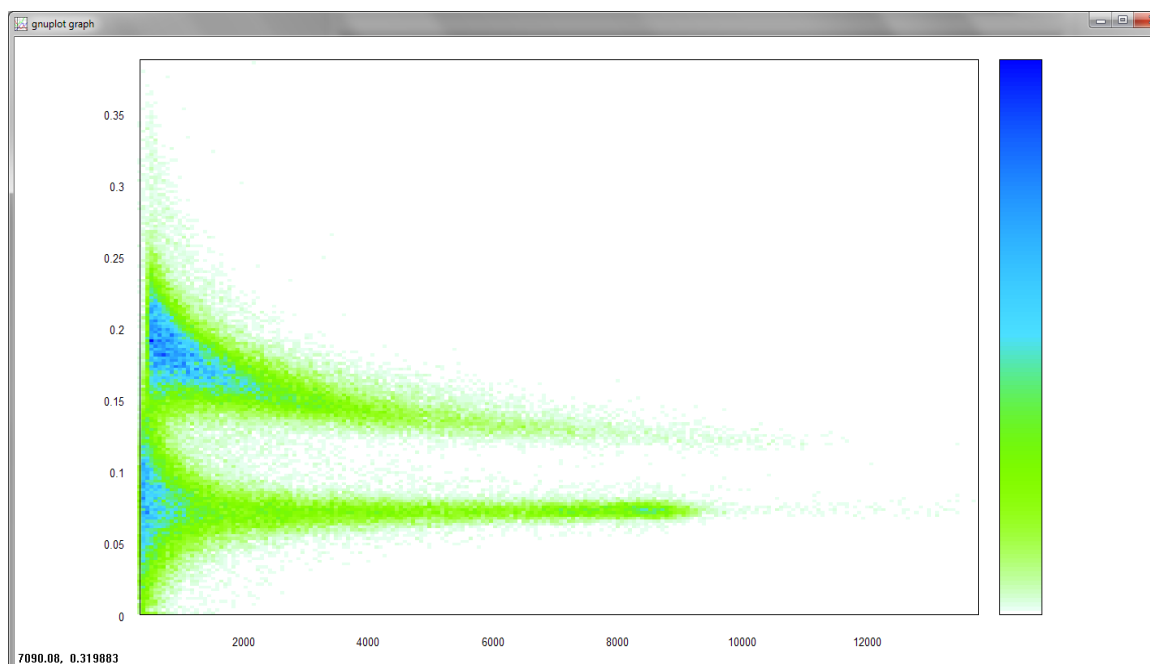


The 2D-plot is the scatter plot of the PSD parameter (Y-axis) vs the pulse Energy  $Q_{\text{long}}$  (X-axis). In the following picture we can clearly see the two lobes corresponding to the  $^{60}\text{Co}$  peaks



**Note:** The positive Z-axis (vertical and pointing up) is represented by the chromatic scale on the right. Going to light green to deep blue, the more the colour is deep, the more the correspondent histogram value is high.

In a Neutron-Gamma experiment the 2D scatter plot (**Fig. 5.5**) will show two distinctive areas, one for Neutron (top) and one for gamma (bottom). The discrimination among the two sources is easier if they are well separated and do not overlap. **[RD2]** gives a reference method to quantify the separation of the two areas.



**Fig. 5.5:** 2D scatter plot of PSD parameter vs Energy in a neutron-gamma application **[RD2]**

#### 8. Generate the output file and save the Energy Histogram-plot data.

The steps below explain how to save the List file (Time Stamp and Charge) by generating an output file on the host station disk.

**Path1:** Tab **OUTPUT** → Section1 **OUTPUT**

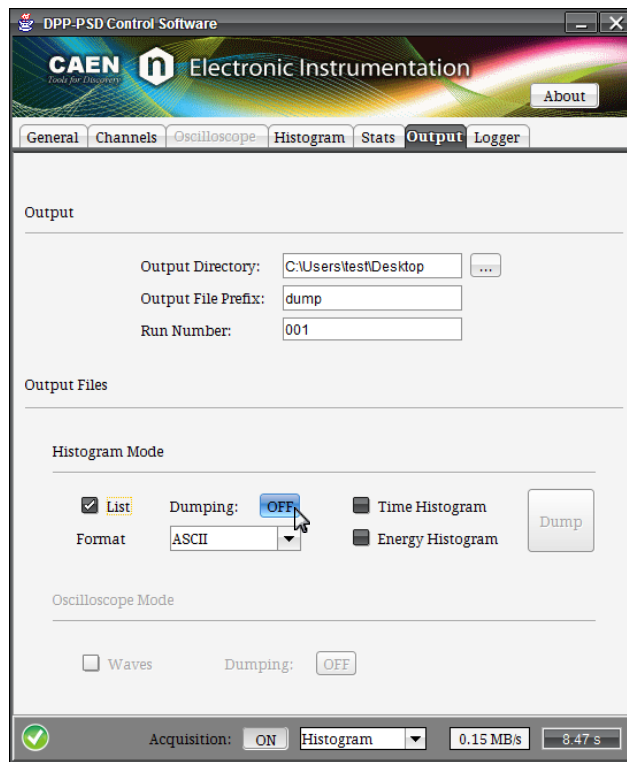
**Action1:** use the **OUTPUT DIRECTORY "..."** button to browse a specific destination folder where to save the output file on the host station; in the example we use the **"Desktop"** folder. If no destination is selected, data will be saved into the user *home* folder.

**Action2:** in the **OUTPUT FILE PREFIX** text box write the name of the prefix for the output file ("**dump**", in this case) and press **"Enter"** on your keyboard. If no prefix is written by the user, the program generates the output file with a default prefix.

**Action3:** in the **RUN NUMBER** text box write the run number. This number automatically increase by one unit when you start a new acquisition through the ON/OFF Acquisition button.

**Path2:** Tab **OUTPUT** → Section2 **OUTPUT FILES**

**Action4:** click on the **LIST** checkbox, select the data writing format between ASCII and BINARY, and press the **ON/OFF** button to start/stop the output file writing.



**Action5:** press the **ACQUISITION “ON/OFF”** button to start/stop the acquisition session.

**Action6:** check the file saved in the selected destination folder. The file format for the current acquisition data is: **dump\_001\_ls\_0.dat**, where “**dump**” is the chosen prefix, “**ls**” stands for list, “**001**” is the run number, and “**0**” is the channel number. The file is a 4-column file where the first column is the time stamps of each triggered event in the sampling clock unit, the second is the  $Q_{long}$ , the third is the word EXTRAS of the output format (see Sect. **Event Data Format**), and the fourth is the  $Q_{short}$ . Binary writing is particularly efficient for high readout throughput. Both ASCII and Binary files have a header describing the data type and format. The header format is described in Sect. **The Tab “Output”**.



## 6. Coincidences and Synchronization

The DPP-PSD firmware allows for coincidences among different channels and synchronization of different boards.

### Coincidences

Acquiring coincident events from different channels is a common task in physics. Through the DPP firmware each channel of the digitizer can trigger independently from the others, and generate a “trigger request”. All the trigger requests can be sent to the common “ROC FPGA” (see **Fig. 2.1**) for the coincidence evaluation. The ROC can be programmed to look for triggers within a programmable window, through the Individual Trigger Logic (ITL) that can perform the logic operation of AND, OR, or Majority. When the coincidence condition is met, the ROC sends back a “trigger validation” signal, one per channel. The coincidence logic is individual, so that it is possible to program different coincident conditions for each channel.

The trigger validation enables the data saving into the memory buffer. In this way the channel uses its local trigger for the event building (time stamp, gate, etc..) but only those events having the validation are saved into the memory.

More information and detailed instructions on how to make coincidences among channels of the same board can be found in **[RD4]**.

### Synchronization among different boards

In cases when multi-board systems are involved in the experiment, it is necessary to synchronize different boards. In this way the user can acquire from N boards with Y channel each, like if they were just one board with (N x Y) channels.

The main issue in the synchronization of a multi-board system is to propagate the sampling clock among the boards. This can be made by FAN-IN of an external clock into the CLK-IN front panel connector of each board, or, only in case of VME models, propagating in Daisy chain the clock along the digitizers through the CLK-IN / CLK-OUT front panel connectors. One board must be chosen to be the “master” board that propagates its own clock (internal or external) to the others. A programmable phase shift can adjust possible delays in the clock propagation. This allows to have both the same ADC sampling clock, and the same time reference for all boards. Having the same time reference means that the acquisition starts/stops at the same time, and that the time stamps of different boards is aligned to the same absolute time.

There are several ways to implement the trigger logic. The synchronization tool allows to propagate the trigger to all boards and acquire the events accordingly. Moreover in case of busy state of one or more boards, the acquisition is inhibited for all boards.

Refer to **[RD8]** for more details on how to synchronize CAEN digitizers.

# 7. Software Interface

## Introduction

The DPP-PSD Control Software is an application that manages the communication and the data acquisition from digitizers where the DPP-PSD firmware is installed. The software allows the user to select proper communication and DPP settings. Waveforms and histograms can also be plotted in real time for one channel at a time (as described in Sect. **GUI Description**).



**Note:** DPP-PSD Control Software is not provided with data analysis features and it is developed to work with 720, 725, 730, 751 Digitizer series and the DT5790.



**Note:** Limited to the DT5790 Digital Pulse Analyzer, the DPP-PSD Control Software manages also the programming of the variable input dynamics of the analog channels and features a special tab for the configuration of the HV channel settings.

## Block Diagram

The DPP-PSD Control Software (**Fig. 7.1**) is made by different parts: there is a user-friendly java *GUI* that allows to easily configure all the relevant parameters for the DPP-PSD acquisition. The GUI directly handles the Acquisition Engine (*DPPRunner*) through run time commands and generates a textual configuration file that contains all the selected parameters values. This file is read by the DPPRunner, a C console application that programs the Digitizer according to those parameters. DPPRunner can also start/stop the acquisition and manage the data readout. Data (both as waveforms, and list of time stamps and energies/time) can be plotted using the external plotting tool *gnuplot*, or saved to output files and analyzed offline.

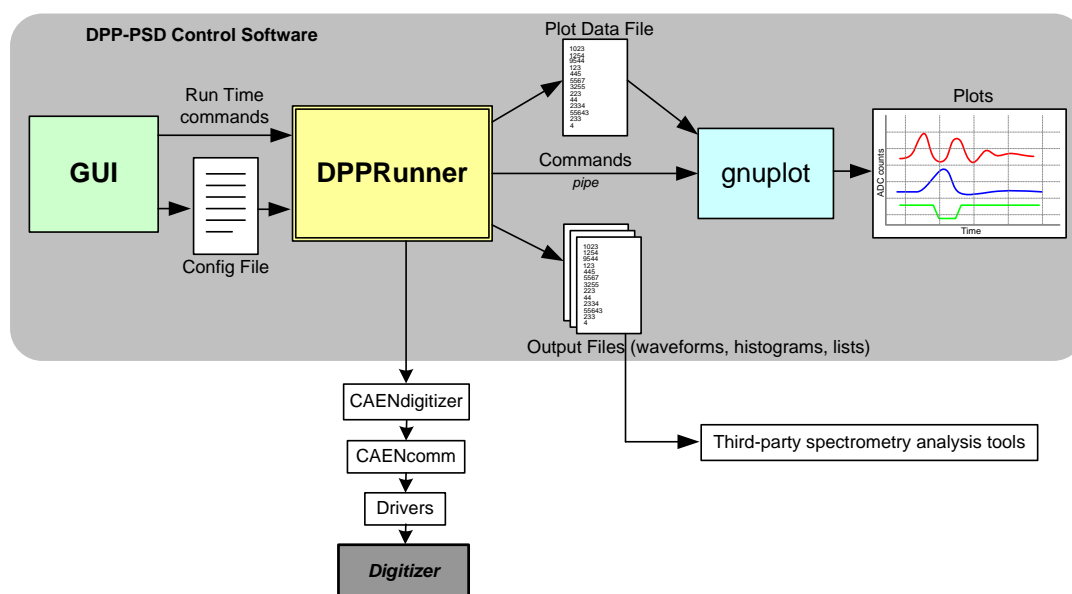


Fig. 7.1: The DPP-PSD Control Software block diagram

## Drivers & Libraries

### Drivers

To deal with the hardware, CAEN provides the drivers for all the different types of physical communication interfaces featured by the specific digitizer and compliant with Windows and Linux OS:

- **USB 2.0 Drivers for NIM/Desktop** boards are downloadable on CAEN website ([www.caen.it](http://www.caen.it)) in the “Software/Firmware” tab of the digitizer web page (**login required**).



**Note:** Windows OS USB driver installation for Desktop/NIM digitizers is detailed in [RD6].

- **USB 2.0 Drivers for V1718** CAEN Bridge, required for VME boards interface, is downloadable on CAEN website ([www.caen.it](http://www.caen.it)) in the “Software/Firmware” tab of the V1718 web page (**login required**).



**Note:** For the installation of the V1718 USB driver, refer to the User Manual of the Bridge ([RD14]).

- **Optical Link Drivers** are managed by the A2818 PCI card or the A3818 PCIe card. The driver installation package is available on CAEN website in the “Software/Firmware” area at the A2818 or A3818 page (**login required**).



**Note:** For the installation of the Optical Link driver, refer to the User Manual of the specific Controller ([RD15], [RD16]).

### Libraries

CAEN libraries are a set of middleware software required by CAEN software tools (including WaveDump) for a correct functioning. These libraries, including also demo and example programs, represent a powerful base for users who want to develop customized applications for the digitizer control (communication, configuration, readout, etc.):

- **CAENDigitizer** is a library of functions designed specifically for the Digitizer family and it supports also the boards running the DPP firmware. The CAENDigitizer library is based on the CAENComm library. For this reason, **the CAENComm libraries must be already installed on the host PC before installing the CAENDigitizer**.

The CAENDigitizer installation package is available on CAEN website in the ‘Download’ area at the CAENDigitizer Library page. Reference document: [RD5].

- **CAENComm** library manages the communication at low level (read and write access). The purpose of the CAENComm is to implement a common interface to the higher software layers, masking the details of the physical channel and its protocol, thus making the libraries and applications that rely on the CAENComm independent from the physical layer. Moreover, the CAENComm requires the CAENVMElib library (access to the VME bus) even in the cases where the VME is not used. This is the reason why **CAENVMElib has to be already installed on your PC before installing the CAENComm**.

The CAENComm installation package, and the link to the required CAENVMElib, is available on CAEN website in the ‘Download’ area at the CAENComm Library page. Reference document: [RD11].



**Note:** For Windows only, all libraries are automatically installed through the standalone DPP-PSD Control Software Setup tool. Linux users have to install them in the order described above.

Currently, the CAENComm (and so the CAENDigitizer) supports the following communication interfaces:

PC → USB → Digitizer DT5720 (DT5725/DT5730/DT5751/DT5790) or N6720 (N6725/N6730/N6751) - Desktop and NIM models

PC → USB → V1718/VX1718 → VME → Digitizer V1720/(V1725/V1730/V1751)/VX1720/(VX1725/VX1730/VX1751) - VME models

PC → PCI (A2818) → CONET → Digitizer x720 (x730/x751/DT5790) - All models of the 720 (725/730/751) series and DT5790

PC → PCI (A2818) → CONET → V2718/VX2718 → VME → Digitizer V1720/(V1725/V1730/V1751)/VX1720/(VX1725/VX1730/VX1751) - VME models

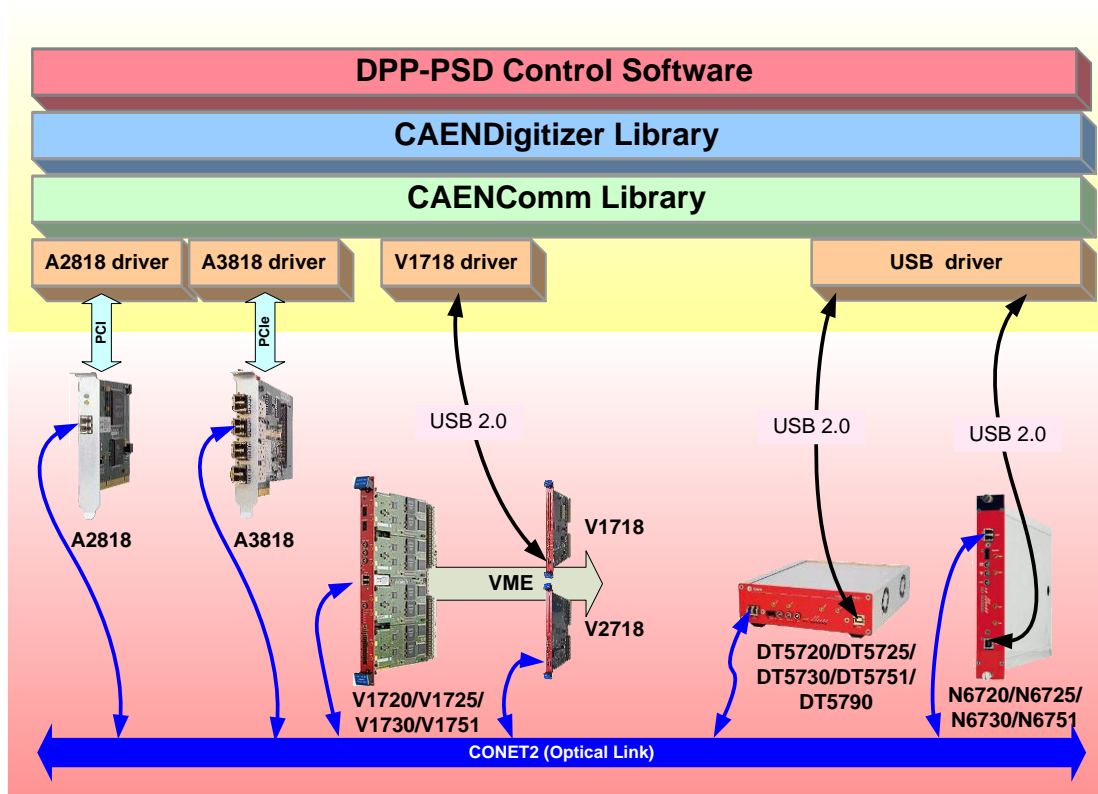
PC → PCIe (A3818) → CONET → Digitizer x720 (x730/x751/DT5790) - All models of the 720 (725/730/751) series and DT5790

PC → PCIe (A3818) → CONET → V2718/VX2718 → VME → Digitizer  
V1720/(V1725/V1730/V1751)/VX1720/(VX1725/VX1730/VX1751) - VME models

**CONET** (Chainable Optical NETWORK) indicates the CAEN proprietary protocol for communication on Optical Link. Refer to **[RD12]** for useful information.



**Note:** CAENDigitizer library for LabVIEW (only for Windows OS) is also available. CAENDigitizer LabVIEW needs the *labview* subfolder of CAENComm to be installed. Please, refer to **[RD9]** for detailed information.



**Fig. 7.2:** Libraries and drivers required for the DPP-PSD system

## Installation

To manage the DPP-PSD system, the host station needs either Windows or Linux OS, and the third-party software **Java Runtime Environment 7** or later (trademark of Oracle, Inc, downloadable from <http://www.java.com>). Linux users must also take care of proper installation of **gnuplot** graphical tool, as well as of **CAEN Libraries**. The latter can be downloaded from CAEN website (login required before to download).

- **Make sure** that your **hardware** (Digitizer and/or Bridge, or Controller) is **properly installed** (refer to the related User Manual for hardware installation instructions).
- **Make sure** you **have installed** the **driver** for your OS and for the communication to be used. Driver installation packages are downloadable on CAEN website (**login required before to download**) as reported in the **Drivers & Libraries** paragraph.

CAEN provides the full installation package for the **DPP-PSD Control Software** in a **standalone version** for **Windows OS**. This version installs all the binary files required to directly use the software (i.e. no need to install the required CAEN libraries in advance). The installation package for **Linux OS** needs other libraries to be installed apart.

- **Download the specific DPP-PSD Control Software installation package** for your OS from CAEN website in the 'Download' area at the DPP-PSD Control Software page (**login required to download**).
- **Extract files** in your host.

*For Linux users:*

- **Click on the red link** above the DPP-PSD Control Software package to download the required CAEN Libraries.
- **Install the libraries** in the following order:
  1. CAENVMLib
  2. CAENComm
  3. CAENDigitizer

The **installation instructions** can be found in the **README file** inside each library folder.

- **Install** the **DPP-PSD Control Software** according to the **installation instructions** of the **Setup/Linux/Readme.txt** file inside the program folder. **Launch** the Control Software typing **DPP-PSD\_ControlSoftware**

*For Windows users*

- **Launch** the **DPP-PSD Control Software installer** and complete the Installation Wizard
- **Run** the **DPP-PSD Control Software GUI** by one of the following options:
  1. The **Desktop icon** for the program
  2. The **Quick Launch** icon for the program
  3. The **.bat file** in the main folder from the installation path on your host.

## GUI Description




The Graphical User Interface (GUI) is composed by seven (7) Tabs, each one divided in one or more sections. In the different tabs there are all the commands needed to manage the connections, to set the board and the channel parameters, and to control the acquisition. An additional tab for HV management is featured by the GUI when interfacing with the Digital Pulse Analyzer DT5790.

A **Common Bar** lays at the bottom of the GUI (**Fig. 7.3**), being visible from any active tab. It contains:



**Fig. 7.3:** Common Bar

- **“Acquisition” button:** starts and stops the acquisition session.
- **“Acquisition Mode” menu:** sets the *“Oscilloscope”* or the *“Histogram”* acquisition mode.
- **“Readout Rate” box:** displays the data throughput rate during the acquisition.
- **“Time acquisition” box:** displays the time duration (sec) of the acquisition. This works only in the Histogram Mode. The acquisition will automatically stop if a *“non-zero”* value is set for the Stop Time parameter.
- **“Connection” icon:** updates itself according to the connection status:

Icon	Status
	Disconnected
	Connection OK
	Connection Error

**Tab. 7.1:** Table of the Connection icon values

## The Tab “General”



Fig. 7.4: Tab “General” in case of 720 series.



Fig. 7.5: Tab “General” in case of 725, 730 and 751 series.

The **General Tab** is divided into three sections: **Acquisition Settings**, **Configuration**, and **Runner**.

The **Acquisition Settings** section includes:

- **“Acquisition” button:** starts and stops the acquisition:
  - **OFF** = acquisition is off
  - **ON** = acquisition is on

It is duplicated in the Common Bar.

- **“Calibrate” button:** to enable the ADCs calibration in case of 725, 730 and 751 series. Check from the Stats tab the ADC temperature. Once it is stable press “Calibrate” and start the acquisition. See **Sect. Practical Use** for more details. It is possible to make the calibration only when the acquisition is OFF.
- **“Acquisition Mode” menu:** selects between the *“Oscilloscope”* mode, where the raw waveforms can be visualized and saved, and the *“Histogram”* mode, where the spectra can be visualized and saved. This command is duplicated in the Common Bar.
- **“Stop Time” box:** sets the value (in seconds) for the Real Time acquisition in Histogram mode. At the end of the fixed time (visualized in the “Acquisition Time” window of the Common Bar) the acquisition is automatically stopped. Set Stop time to *“0”* for an infinite acquisition time.
- **“Histogram Params” menu:** selects the data that the Digitizer provides for the *“Histogram”* acquisition mode (Energy only, Energy and Time).



**Note:** Not available for 725 and 730. Only “Energy and Time” can be selected.




- **“SW Triggers” buttons:** start/stop a software trigger input from the computer to the board in a continuous (“ON/OFF”) or single-shot (“Single”) way.

The **Configuration** section includes:

- **“Configuration File” buttons:** store (“Export”) and recall (“Import”) the configuration of the software and the parameters for the acquisition. The user can so easily manage the different parameters during different acquisitions. “Restore” resets online the parameters to their default values.

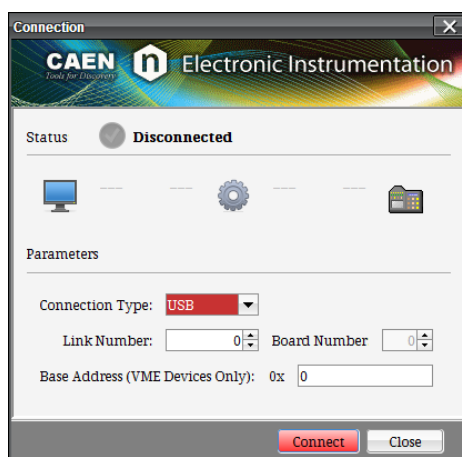
The **Runner** section shows the status of the connection between the board and the Control Software. It is made of:

- **“Status” label:** shows the status of the connection:

Label Options	
Status: 	Disconnected
Status: 	Connection OK
Status: 	Can't open the digitizer

It is duplicated in the Connection window.

- **“Connect” button:** opens the Connection window (**Fig. 7.6:**).



**Fig. 7.6:** Connection Window



In this window the connection parameters can be set:

- **“Connection Type” menu:** selects between “USB” or “OPTLINK” according to the way the board is connected to the PC.
- **“Link Number” box:** sets the number of the port used in the connection and is valid for multiple boards connection.
- **“Board Number” box:** indicates the number of the desired board in a Daisy chain connection between different boards.
- **“Base Address” box:** for VME boards only, sets the VME base address. Set “0” for direct connection.

More details about the connection parameters can be found in **[RD1]** and **[RD9]**.

As a reference, in **Tab. 7.2** there are shown some connection examples involving the DPP-PSD supporting boards.

Connection chain	Type	Link	Slave	Address
PC → <b>USB</b> → DT5720 (DT5725/DT5730/DT5751/DT5790)	USB	0	0	0
PC → <b>USB</b> → V1718 → <b>VME</b> → V1720 (V1725/V1730/V1751)	USB	0	0	32100000*
PC → <b>PCI</b> → A2818 → <b>CONET</b> → N6720 (N6725/N6730/N6751)	PCI	0	0	0
PC → <b>PCI</b> → A2818 → <b>CONET</b> → V1720 (V1725/V1730/V1751)	PCI	0	0	32100000*
PC → <b>PCI</b> → A2818 → <b>CONET</b> → V1720 (V1725/V1730/V1751)**	PCI	0	1	0
PC → <b>USB</b> → DT5720 (DT5725/DT5730/DT5751/DT5790)***	USB	1	0	0

**Tab. 7.2:** Examples of connection settings

(\*) For the correct VME base address to be used, please refer to the Digitizer’s User Manual.

(\*\*) The VME Digitizer is intended to be part of a Daisy chain (see the examples at the end of **[RD1]**)

(\*\*\*) It is supposed that at least two USB ports are used by the PC to communicate with digitizers (see the examples at the end of **[RD9]**).

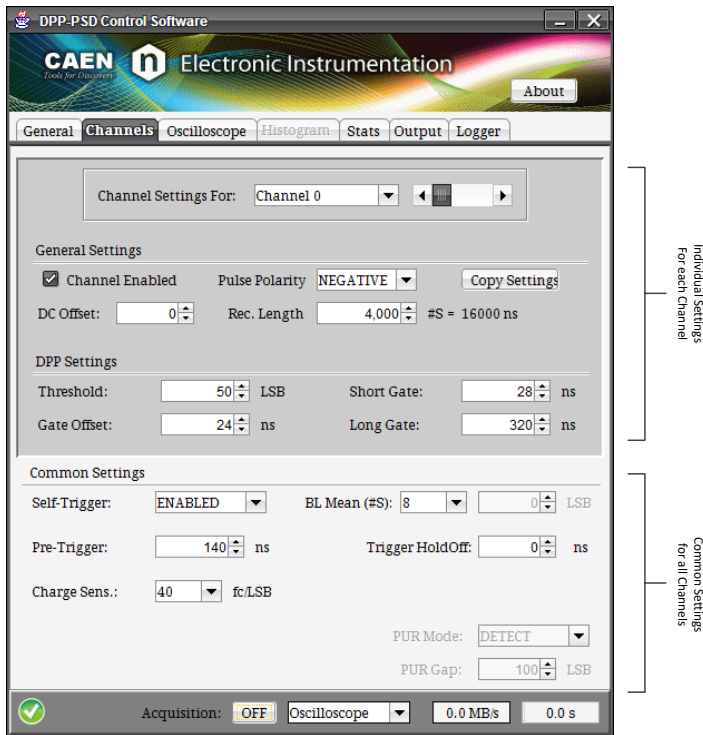
The connection is handled by:

- **“Connect” button:** establishes the connection. A green sign will confirm the correct connection.
- **“Close” button:** closes the connection window.

When a connection is established, in the Runner section the **“Status”** field shows the **“Connected”** value.

- **“Restart” button:** restarts the software causing the board to reset and to reprogram with the actual parameters; the communication is unaffected. In case of DT5790, the Restart function has no effect on the HV channels.

## The Tab “Channels”



**Fig. 7.7:** Tab “Channels”

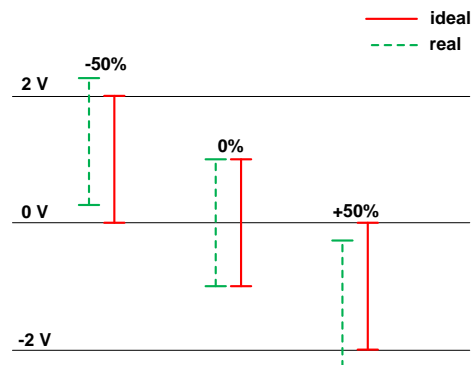
The **Channels Tab** is the core of the DPP-PSD Control Software where it is possible to set all the parameters required by the algorithm.

The tab consists in the **Channel Settings For** field with the sections: **General Settings**, **DPP Settings**, **Common Settings**. Through the “**Channel Settings For**” menu it is possible to select the channel which the parameters are referred to. The channels are selectable either through the drop-down menu or through the slider. Two macro areas can be noticed: the deep grey area for the individual settings of each channel, and the light grey area for the common settings valid for all channels. Once all parameters have been set for the current channel, it is possible to select another channel and a new configuration tab will be available.

The **General Settings** section includes the following commands:

- “**Channel Enabled**” checkbox: enables the selected channel to acquire data.

“**DC Offset**” box: sets the value of the DC Offset applied to the channel, expressed as the percentage of the Full Scale Range. Allowed values range between -50% (Full negative) and +50% (Full Positive). The DC Offset is a DC value added to the input signal by the input stage of the Digitizer in order to fit the signal dynamic range to the ADC input dynamics. Theoretically, the value of 0 (DC Offset = “0”) means that the input pulse DC level is set at half of the ADC dynamics (e.g. 2048 counts for the 720 series and DT5790, 8192 for the 725 and 730 series, and 512 counts for the 751 series). The value of “50” (DC Offset = “50”) sets the DC level at the lower dynamics limit (i.e. 0 counts), while the value of “-50” (DC Offset = “-50”) sets the DC level at the upper dynamics boundary (i.e. 4095 counts for the 720 series and DT5790, 16384 for 725 and 730 series, and 1024 counts for the 751 series). The real DC offset adjustment implemented in the DPP-PSD firmware is shown in **Fig. 7.8**: in order to preserve from saturation the input signals near the dynamics boundaries, setting the DC offset to -50 or +50 puts the signal baseline respectively a step up the upper boundary and a step under the lower boundary.



**Fig. 7.8:** Input signal DC offset adjustment description

- **“Pulse Polarity” menu:** selects the polarity (NEGATIVE/POSITIVE) of the input signal to be processed by the DPP-PSD algorithm. The algorithm is internally designed to work with negative pulses. When the analog input pulses are positive, it is necessary to invert them before the DPP algorithm is applied; in this case, the option PULSE POLARITY = “Positive” enables the internal inversion of the pulse polarity. Note that the inversion is applied to the DPP algorithms only, while it doesn’t affect the waveform recording (both plots and output files keep the original pulse polarity). With negative analog input pulses, use PULSE POLARITY = “Negative”.
- **“Record Length” box:** selects the length of the acquisition window expressed in number of samples (1 sample= 4 ns for the 720, 725 series and DT5790, 2 ns for the 730 series, and 1 ns for the 751 series). This is the number of samples that are saved of the waveform when the board is running in Oscilloscope Mode.
- **“Copy Settings” button:** copies the general settings of the current channel to other channels.

The **DPP Settings** section includes:

- **“Threshold” box:** sets the absolute value of the trigger threshold referred to the input pulse baseline. The value is expressed in LSB. The value in mV for 1 LSB can be calculated according to the board input range. For a digitizer of the 720 series, with 12-bit resolution and input range of 2 Vpp, the LSB is:

$$\text{LSB} = 2 / 2^{12} = 0.488 \text{ mV}$$

For a 2Vpp input range and 10-bit resolution 751 series, 1LSB = 0.97 mV. For the 725 and 730 series there are two possible input ranges: 2 Vpp and 14-bit resolution 1 LSB = 0.12 mV; 0.5 Vpp and 14-bit resolution 1 LSB = 0.03 mV.

- **“Gate Offset” box:** sets the “pre-gate” parameter, i.e. the starting position of the long and short gates before the trigger signal. Values are expressed in ns and can vary in steps of clock units (i.e. 4 ns if 720 series, DT5790 and 725 series, 2 ns for 730 series, and 1 ns if 751 series).
- **“Short Gate” box:** sets the short gate width for the  $Q_{\text{short}}$  calculation. Values are expressed in ns and can vary in steps of clock units.
- **“Long Gate” box:** sets the long gate width for the  $Q_{\text{long}}$  calculation. Values are expressed in ns and can vary in steps of clock units.



**WARNING:**  $\text{Gate\_Offset} \leq \text{Short\_Gate} \leq \text{Long\_Gate}$ ; this relation must be always true.

The **Common Settings** section includes:

- **“Self-Trigger” menu:** ENABLE/DISABLE the DPP *self-trigger* of each channel. When disabled, the DPP algorithm still generates the self-trigger (pulse auto trigger) with the programmed threshold, but the signal is propagated only to the TRG-OUT panel output and it is not used for the acquisition. In these conditions, the acquisition is activated when an external trigger signal is sent to the TRG-IN panel input or by the combination of the other channel self-triggers.
- **“Pre-Trigger” box:** sets the portion of the waveform acquisition window to be saved before the trigger. Its value is expressed in ns. The following relations must be true.

$$Gate - Offset \leq Pre - Trigger - 32ns \text{ (for 720 series and DT5790);}$$

$$Gate - Offset \leq Pre - Trigger - 8ns \text{ (for 751 series);}$$



**Note:** In case of 725 and 730 series, if too short values of the Pre Trigger are set, the firmware automatically adjusts them to the minimum correct value.



**Note:** When switching to histogram mode the Pre-Trigger value is automatically set to the minimum allowed, i.e. Gate-Offset + 32 ns for 720 series and DT5790, Gate-Offset + 38 ns for 725 and 730 series, and Gate-Offset + 8 ns for 751 series.



**Note:** For 751 series, the firmware allows to set negative values of pre-trigger as well, even if the DPP-PSD Control Software does not manage it. This can be enabled through the PRE\_TRG register.

- **“Charge Sens.” Menu:** sets the *charge sensitivity*, the weight of the LSB for the charge data (16 bit). For instance, if Q = 100 counts and Charge Sens. = 40 fC/LSB, the integrated charge is 4 pC. When the charge pulse exceeds the full scale range (0xFFFF), it is recommended to reduce the sensitivity in order to avoid saturation. The allowed values (fC/LSB) are:

for the 720 series and DT5790

40, 160, 640, 2560;

for the 725 and 730 series

5, 20, 80, 320, 1280;

for the 751 series

20, 40, 80, 160, 320, 640.



**Note:** The charge is integrated inside the FPGA over a 22-bit accumulator; the sensitivity defines the dividing factor (i.e. the right shift) of this accumulator to rescale the energy value before it is saved into the memory buffer.

- **“Baseline Mean” menu:** sets the number of samples used by the mean filter to calculate the input pulse baseline. Allowed values are:

for the 720 series and DT5790

“Fixed”, 8, 32, 128.

for the 725 and 730 series

“Fixed”, 16, 64, 256, 1024.

For the 751 series

“Fixed”, 8, 16, 32, 64, 128, 256, 512.

The “Fixed” option enables the absolute baseline calculation.

- **“Baseline” menu:** sets the fixed value (in LSB) of the baseline when the “Fixed” value is selected in the “Baseline Mean” menu.
- **“Trigger Hold-Off” menu:** sets the time width of the trigger hold-off. The trigger hold-off starts with the trigger and corresponds to the time window where any other triggers are inhibited. It is expressed in steps of 8 ns. Accepted values are:

0, 8, 16, 24, ..., 8184.

## The Tab “Oscilloscope”

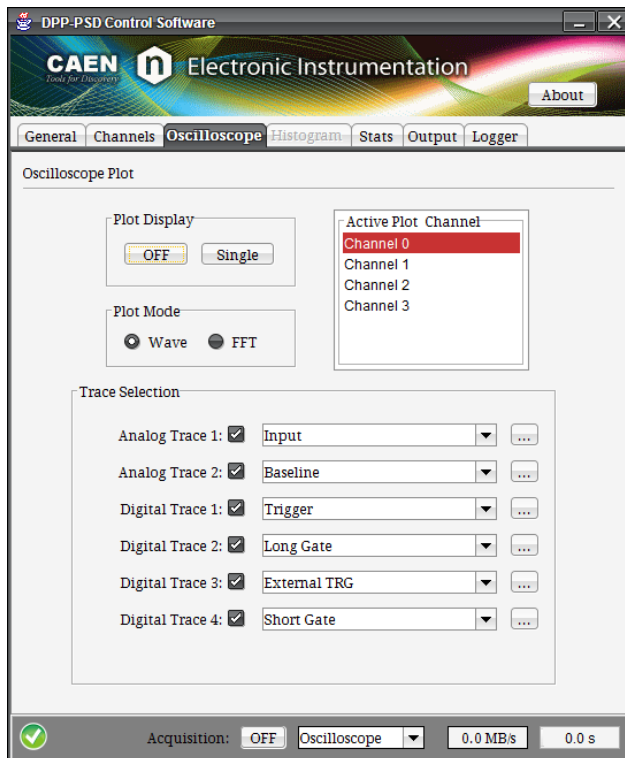


Fig. 7.9: Tab “Oscilloscope”

The **Oscilloscope Tab** consists only of the **Oscilloscope Plot** section, in which all the parameters for the signals visualization are set. A maximum of six (6) traces can be simultaneously displayed for 720 series and DT5790, four (4) for 725 and 730 series, and five (5) for 751 series.

The **Oscilloscope Plot** section includes:

- **“Plot Display” buttons:** enable/disable the plot visualization and let the user visualize the waveforms continuously (“ON/OFF”) or in single shots (“Single”).
- **“Plot Mode” check-cells:** selects if the Waveform (“Wave”) or the Fast Fourier Transform (“FFT”) has to be visualized. The FFT option uses only the signal in “Analog Trace 1”.
- **“Active Plot Channel” menu:** selects the channel whose signals are visualized. Only one channel at a time can be plotted. The selected channel has to be checked in the Channels Tab.
- **“Trace Selection” box** that includes:
  - **“Analog Trace 1” menu:** selects the first analog trace to be visualized in the plot:  
For 720, 751 series and DT5790:  
“Input”.  
For 725 and 730 series:  
“Input”;  
“CFD”.
  - **“Analog Trace 2” menu:** selects the second analog trace to be visualized in the plot. Dual Trace option must be enabled to plot the Analog Trace 2.  
For 720, 751 series and DT5790:  
“Baseline”;  
“None”.

For 725 and 730 series:

“Baseline”;  
 “CFD”;  
 “None”.



**Note:** when the “None” option is enabled all the available samples are used to store the Analog Trace 1. When one of the other options is selected, half of the samples are used to store the Analog Trace 2. Therefore only the remaining half samples are used for the input waveform, and in the plot visualization the Analog Trace 1 appears at half of the sampling frequency.

- **“Digital Trace 1” menu:** selects the first digital trace to be visualized in the plot:

For 720, 751 series and DT5790:

“Trigger”.

For 725 and 730 series:

“Long Gate”;

“Over Threshold”, digital signal that is 1 when the input signal is over the requested threshold;

“Shaped TRG”, logic signal of programmable width generated by a channel in correspondence with its local self- trigger. It is used to propagate the trigger to the other channels of the board and to other external boards, as well as to feed the coincidence trigger logic (refer to **[RD4]**);

“TRG Val. Acceptance Win.”, logic signal corresponding to the time window where the coincidence validation is accepted. The validation enables the event dump into the memory (see **[RD4]**);

“Pile Up”, logic pulse set to 1 when a pile up event occurred (to be implemented);

“Coincidence”, logic pulse set to 1 when a coincidence occurred (refer to **[RD4]**);

“Trigger”.

- **“Digital Trace 2” menu:** selects the second digital trace to be visualized in the plot:

For 720 series and DT5790:

“Long Gate”.

For 725 and 730 series:

“Short Gate”;

“Over Threshold”, digital signal that is 1 when the input signal is over the requested threshold;

“TRG Validation”, digital signal that is 1 when a coincidence validation signal comes from the mother board FPGA (refer to **[RD4]**);

“TRG HoldOff”, logic signal generated by a channel in correspondence with its local self- trigger. Other triggers are inhibited for the overall Trigger Hold-Off duration;

“Pile Up”, logic pulse set to 1 when a pile up event occurred (to be implemented);

“Coincidence”, logic pulse set to 1 when a coincidence occurred (refer to **[RD4]**);

“Trigger”.

For 751 series:

“Long Gate”;

“Over Threshold”, digital signal that is 1 when the input signal is over the requested threshold;

“Shaped TRG”, logic signal of programmable width generated by a channel in correspondence with its local self- trigger. It is used to propagate the trigger to the other channels of the board and to other external boards, as well as to feed the coincidence trigger logic (refer to **[RD4]**);

“TRG Val. Acceptance Win.”, logic signal corresponding to the time window where the coincidence validation is accepted. The validation enables the event dump into the memory (see **[RD4]**);

*"Pile Up"*, logic pulse set to 1 when a pile up event occurred (to be implemented);

*"Coincidence"*, logic pulse set to 1 when a coincidence occurred (refer to [RD4]);

*"None"*, when both Digital Trace 2 and Digital Trace 3 are set to *None*, the event sample is represented into 10 bits of memory location. Otherwise 2 bits are reserved for the Digital Traces 2 and 3, and each sample is represented into 8 bits. Select None for both trace to enable to visualization of the Input trace at 10 bit.

- **"Digital Trace 3" menu:** selects the third digital trace to be plotted:

For 720 series and DT5790:

*"External TRG"*, the external trigger signal when enabled;

*"Over Threshold"*, digital signal that is 1 when the input signal is over the requested threshold;

*"Shaped TRG"*, logic signal of programmable width generated by a channel in correspondence with its local self- trigger. It is used to propagate the trigger to the other channels of the board and to other external boards, as well as to feed the coincidence trigger logic (refer to [RD4]);

*"TRG Val. Acceptance Win."*, logic signal corresponding to the time window where the coincidence validation is accepted. The validation enables the event dump into the memory (see [RD4]);

*"Pile Up"*, logic pulse set to 1 when a pile up event occurred (to be implemented);

*"Coincidence"*, logic pulse set to 1 when a coincidence occurred (refer to [RD4]).

For 751 series:

*"Short Gate"*;

*"Over Threshold"*, digital signal that is 1 when the input signal is over the requested threshold;

*"TRG Validation"*, digital signal that is 1 when a coincidence validation signal comes from the mother board FPGA(refer to [RD4]);

*"TRG HoldOff"*, logic signal generated by a channel in correspondence with its local self-trigger. Other triggers are inhibited for the overall Trigger Hold-Off duration;

*"Pile Up"*, logic pulse set to 1 when a pile up event occurred (to be implemented);

*"Coincidence"*, logic pulse set to 1 when a coincidence occurred (refer to [RD4]);

*"None"*, when both Digital Trace 2 and Digital Trace 3 are set to *None*, the event sample is represented into 10 bits memory location. Otherwise 2 bits are reserved for the Digital Traces 2 and 3, and each sample is represented into 8 bits. Select None for both trace to enable to visualization of the Input trace at 10 bit.



**Note:** when the **"Digital Probes"** are enabled, the input is represented into 8 bits rather than 10 bits. This means that the input granularity in the plot visualization is four. Those who need to acquire waveforms with full resolution should select **"NONE"** for both Digital Trace 2 and 3.

- **"Digital Trace 4" menu:** selects the fourth digital trace to be plotted (720 series and DT5790 only) among:

*"Short Gate"*;

*"Over Threshold"*, digital signal that is 1 when the input signal is over the requested threshold;

*"TRG Validation"*, digital signal that is 1 when a coincidence validation signal comes from the mother board FPGA(refer to [RD4]);

*"TRG HoldOff"*, logic signal generated by a channel in correspondence with its local self-trigger. Other triggers are inhibited for the overall Trigger Hold-Off duration;

*"Pile Up"*, logic pulse set to 1 when a pile up event occurred (to be implemented);

*"Coincidence"*, logic pulse set to 1 when a coincidence occurred (refer to [RD4]).

- **"..." button:** opens the Trace Setting window (Fig. 7.10) to set the DC offset and the gain of the traces in the Oscilloscope plot. These settings have no effect on the real input signal, but only on the

visualization of the Oscilloscope plot. For this reason it is recommended to adjust only the digital traces offset and gain for a correct visualization.

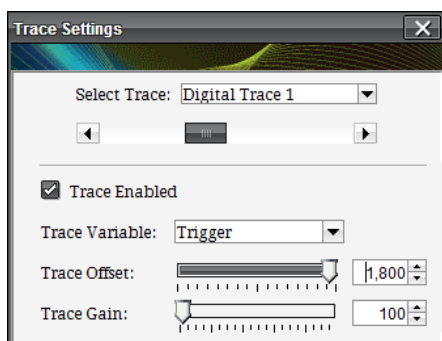


Fig. 7.10: The Trace Settings Window

## The Tab “Histogram”

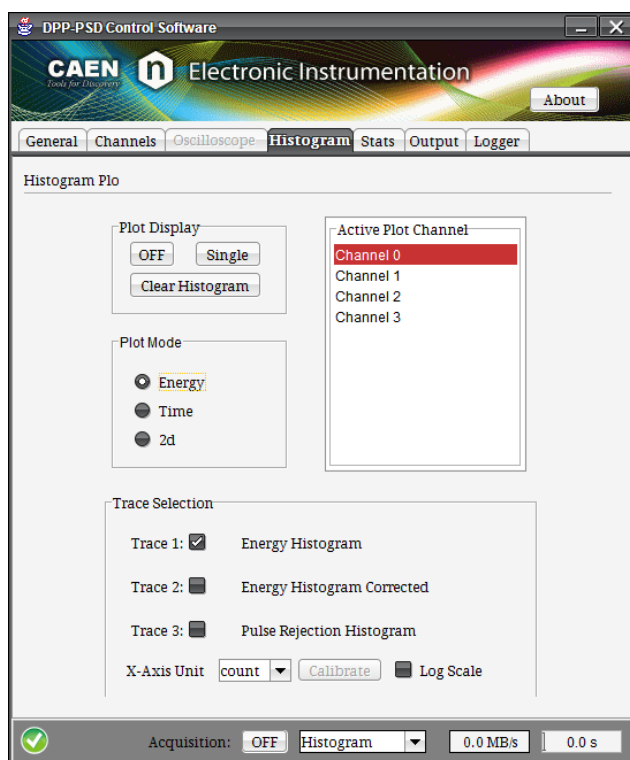


Fig. 7.11: Tab “Histogram”

The **Histogram Tab** contains in the **Histogram Plot** section all functions for plotting the Energy or Time Histograms.

The **Histogram Plot** section includes the following settings:

- **“Plot Display” buttons:** The “ON/OFF” button enables/disables the continuous plotting of the histogram. In the OFF position it is possible to update manually the plot by pushing the “Single” button. The “Clear Histogram” button clears the histogram (i.e. resets the plotting to zero).
- **“Active Plot Channels” menu:** selects the channel whose signals are visualized. Only one channel at a time can be plotted. The selected channel has to be checked in the Tab Channels.
- **“Plot Mode” check cells:** selects if a “Energy”, “Time” or “2D” histogram has to be visualized.



- “Trace Selection” menu that includes:
  - “Trace 1”: Energy/Time Histogram when Energy/Time plot mode is enabled. In the Energy Histogram pile-up events are not included. Time Histogram is intended to be the histogram of the time intervals between subsequent triggers;
  - “Trace 2” menu: for Energy plot mode only, enables/disables the visualization of the Energy Histogram Corrected by adding the redistribution of the pile-up counts (Trace 3) to the Energy Histogram (Trace 1);
  - “Trace 3” menu: for Energy plot mode only, enables/disables the visualization of the Rejected Pulses Histogram. This is the redistribution of the pile-up counts, within fixed acquisition time windows, with respect to the energy distribution in the same windows;
  - “X-Axis Unit” menu: for Energy plot mode only, selects the unit of measurement of the Energy x-axis in the histogram between “ADC Counts” and “KeV”. In order to define a KeV scale, the “Calibrate” button opens the Energy Calibration window (Fig. 7.12) where it is possible to calibrate the spectrum by a dedicated menu: a customized Calibration line can be built here. Different calibration lines are available for the different channels;



**Note:** The “KeV” option affects only the histogram plot, while the histogram x-axis data will be always saved as ADC counts in the Output tab.

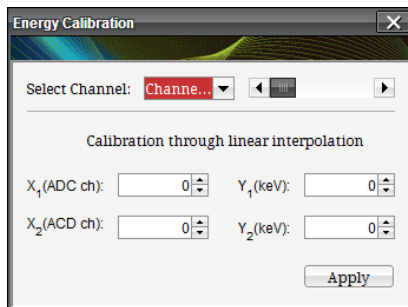


Fig. 7.12: Energy Calibration window

- “Log Scale” menu: enables/disables the plot log scale.

The Tab “Stats”

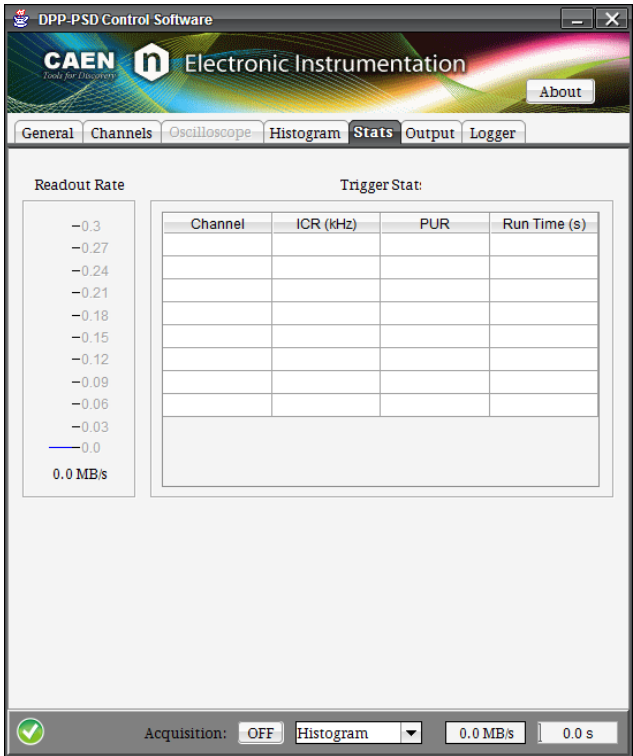


Fig. 7.13: Tab “Stats” in case of 720 series

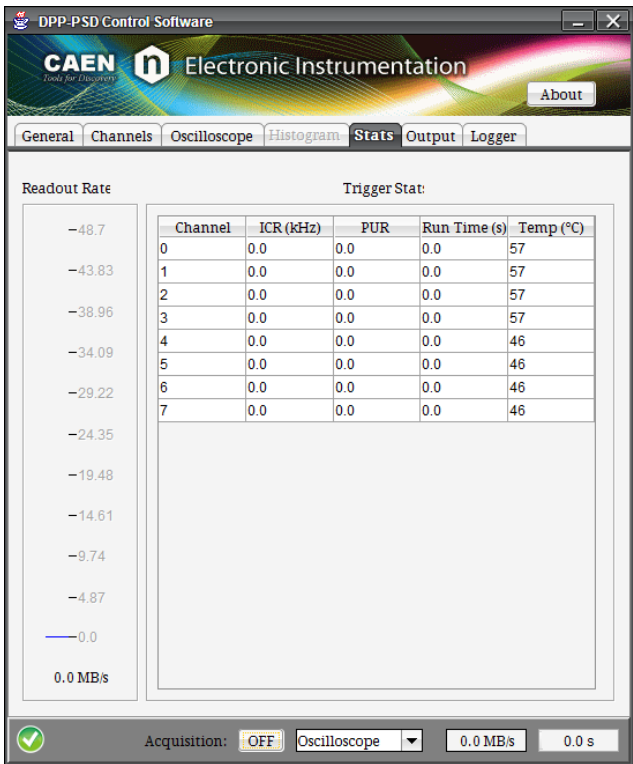


Fig. 7.14: Tab “Stats” in case of 725, 730 and 751 series

In the **Stats Tab** are summarized most of the important statistic information about every enabled channels. The tab is composed by two sections: **Readout Rate** and **Trigger Stat**.

The **Readout Rate** section hosts:

- **“Readout Rate” display:** shows the data throughput from the board to the computer in MB/s (the same value is visible in the Common Bar).

The **Trigger Stats** section is made by:

**“Trigger Stats” table:** reports for each enabled channel (“*Channel*”) the real time Incoming Counting Rate (“*ICR*” expressed in kHz), the Pile Up Rejection (“*PUR*” in percentage), and the Run Time (in seconds). An additional column is available for 725, 730 and 751 series, reporting the ADCs temperature (“*Temp* (°C)”). The temperature monitoring works only when the acquisition is stopped.

## The Tab “Output”

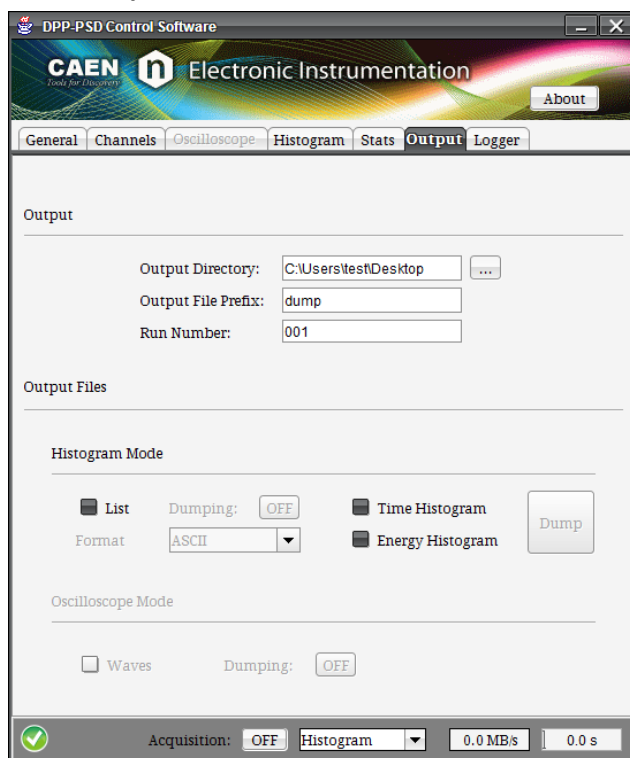


Fig. 7.15: Tab “Output”

In the **Output Tab** there are all the commands to save output files containing spectra, waveforms and lists on a PC. The sections in this tab are **Output** and **Output Files**.

The **Output** section includes the following settings:

- **“Output Directory” box:** contains the destination path for the output files. Directly write the path or inserted it using the “...” button. If no destination is selected, data is saved into the user home folder.
- **“Output File Prefix” box:** contains the prefix for the output file name. If no prefix is chosen, the default one is used.
- **“Run number” box:** contains the run number for the output file. This number automatically increase by one unit when at the start of a new acquisition through the ON/OFF Acquisition button.

The complete file name will be *Prefix\_Run\_XY\_N.dat*, where:

XY is “eh” in case of Energy Histogram, “th” in case of Time Histogram, “ls” in case of List and “wf” in case of Waveform, and *N* is the channel number.

Data saving is activated by selecting the options in the **Output Files** section. For the **“Histogram Mode”** there is the possibility to choose among:

- **“List” checkbox:** saves every event in ASCII or BINARY format, according to the list format menu.

**ASCII list format:** is a 4-column file where the first column is the Trigger Time Tag, the second is the integrated charge  $Q_{long}$  (in counts unit), the third is the EXTRAS word of the Event Format (refer to Sect. **Event Data Format**), and the fourth is the  $Q_{short}$  (in counts unit). Note that the Trigger Time Tag is expressed in clock units, where the clock unit is 4 ns for 720, 725 series and DT5790, 2 ns for 730 series, and 1 ns for 751 series. The Time Stamp is not corrected for the roll-over, therefore once it reaches its maximum it starts again from zero.

**BINARY list format:** the file contains the sequence of the recorded events in the binary format:

Each event is represented in the following format:

unsigned 32 bit	signed 16 bit	signed 16 bit	BIT FORMAT
Trigger Time Tag	Q <sub>long</sub>	Q <sub>short</sub>	EVENT

Events are then written consecutively in the format:

64 bit	64 bit	64 bit	64 bit	BIT FORMAT
EVENT 1	EVENT 2	EVENT 3	EVENT 4	DATA

Data saving starts/stops by the **DUMPING “ON/OFF”** button. Binary writing is particularly efficient for high readout throughput.

**Header file format:** both the ASCII and Binary files have a header describing the data type and format. The header structure is as follows.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	BIT
																NWORDS HEADER												PROTOCOL VERSION				HEADER 0
TYPE FORMAT																												DATA TYPE				HEADER 1
TYPE FORMAT																												DATA TYPE				HEADER 2
...																												...				...
TYPE FORMAT																												DATA TYPE				HEADER n

**Fig. 7.16:** Header file structure

The first word of the header describes the header itself, where:

- Bits[7:0] describes the protocol version number;
- Bits[15:8]: corresponds to the number of words of the header itself, including the first word;
- Bits[31:16]: reserved.



**Note:** the number of bits will be unchanged also for future protocol versions.

From “Header 1” to “Header n” the corresponding word identifies the type of data to be read in the list:

- Bits[7:0]: describes the **“Data Type”** of the event that has been saved into the list. Each header corresponds to a different type, i.e. to a different information saved for each event. The complete list is described below:
  - 0 = Trigger Time Tag type
  - 1 = Energy type
  - 2 = Extras type
  - 3 = Short Energy type (for DPP-PSD only)
  - 4 = DPP Code
  - 255 = Fake (in case of error)

- Bits[31:8]: corresponds to the “**Type Format**” of the corresponding type of the word. Possible choices are:
  - 0 = INT8;
  - 1 = UINT8;
  - 2 = INT16;
  - 3 = UINT16
  - 4 = INT32;
  - 5 = UINT32;
  - 6 = INT64;
  - 7 = UINT64;
  - 8 = STRING;
  - 9 = LONG;
  - 10 = DOUBLE;
  - 11 = CHAR;
  - 255 = none.

For any DPP firmware the protocol version 1 is organized as follows, where words from “Header 1” to “Header 4” are optional, according to the selected type of information for the list dump:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	BIT
																NWORDS HEADER						PROTOCOL VERSION						HEADER 0				
Trigger Time Tag Format																						Trigger Time Tag						HEADER 1				
Energy Format																						Energy						HEADER 2				
Extras Format																						Extras						HEADER 3				
Short Energy Format																						Short Energy						HEADER 4				
DPP Code Format																						DPP Code						HEADER 5				

**Fig. 7.17:** Header file structure for DPP firmware

The order of the Header words defines the order to read the data list.

For example, in case the header has the following structure:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	BIT
																0x6						0x1						HEADER 0				
0x7																						0x0						HEADER 1				
0x2																						0x1						HEADER 2				
0x5																						0x2						HEADER 3				
0x2																						0x3						HEADER 4				
0x88																						0x4						HEADER 5				

the user must read 6 words of header, and the data format has to be read as:

- Trigger Time Tag (uint64);
- Energy (charge long) (int16);
- Extras (uint32)
- Energy (charge short) (int16)

i.e. 8 Bytes of Time Tag, 2 Bytes of Energy long, 4 Bytes of Extras, and 2 Bytes of Energy short.

The DPP code 0x88 corresponds to the DPP-PSD for x725 and x730.



**Note:** In the ASCII format file the header words has to be read as in the binary format. For each event in the list there is a corresponding column with the event type as described in the header.

- **“Energy Histogram” checkbox:** saves the Energy Histogram data in a 2-column file, where the first column is made of the x-data values (i.e. histogram bins, always ADC counts), and the second is the frequency for each event. Data saving is performed by the **DUMP** button.



**Note:** The Energy Histogram data refer to Qlong only.

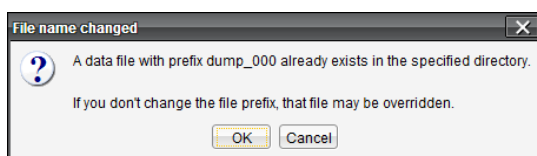
- **“Time Histogram” checkbox:** saves the Time Histogram data in a 2-column file (histogram bins as ADC counts on the first column and frequency values on the second). Data saving is performed by the **DUMP** button.

For the **“Oscilloscope Mode”**, only one option is enabled:

- **“Waves” checkbox:** saves the samples of the digitised waveforms from all the enabled channels. All digital probes that are enabled in the “Oscilloscope Tab” will be saved as well. Data saving starts/stops by the **DUMPING “ON/OFF”** button.



**Note:** Both for *List* and *Waves* dumping, a warning message will appear if you are changing the prefix name while the data saving option is enabled (see **Fig. 7.18**).



**Fig. 7.18:** Warning message about data saving

### The Tab “Logger”

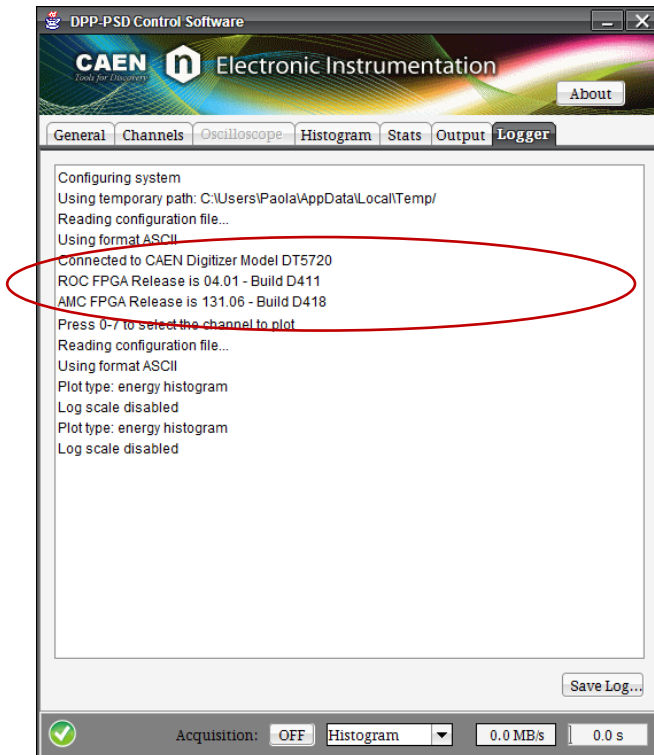


Fig. 7.19: Tab “Logger”

In the **Logger Tab** the user can read board and firmware information (**Fig. 7.19**) and have a direct view of the parameters values and the mode options being set during the current program session. The session log can be saved on disk by using the “Save Log” button.

### The Tab “HV Config”

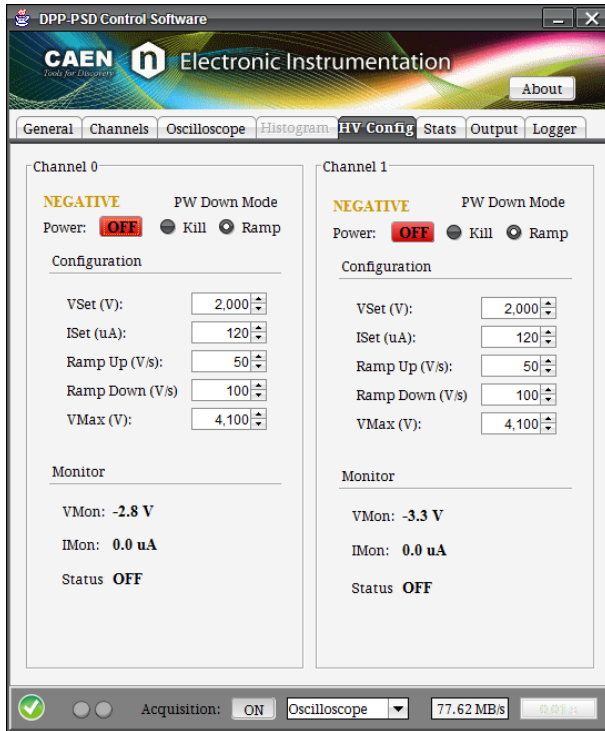


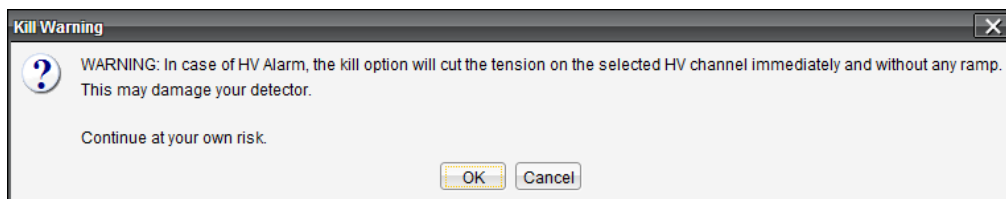
Fig. 7.20: Tab “HV Config”

Available only for the DT5790 Digital Pulse Analyzer, the **HV Config** tab controls the settings of the two HV channels in the sections **Channel 0** and **Channel 1** by means of:

- **“Power” button:** enables/disables the HV channel power supply according to the Configuration settings.
- **“PW Down Mode” checkcells:** selects the power down mode of the HV channel in case of an overcurrent or an inhibit event occurs. The selectable options are:  
*“Kill”*, that performs an immediate shutdown;  
*“Ramp”*, that performs a stepwise shutdown.



**WARNING:** Use with caution the ***Kill*** setting, because the abrupt stop of the HV channels may cause damage to the attached detector.



The **Configuration** section contains:

- **“VSet (V)” box:** sets the value of the channel bias voltage in Volt [range 0:5100 in steps of 0.1 V].
- **“ISet (uA)” box:** sets the value of the channel bias current in micro Ampere [range 0:315 in steps of 0.01 V].
- **“Ramp UP (V/s)” box:** sets the step width of the bias voltage ramp up in Volt per second [range 0:500 in steps of 1 V/s].
- **“Ramp Down (V/s)” box:** sets the step width of the bias voltage ramp down in Volt per second [range 0:500 in steps of 1 V/s].
- **VMax (V)” box:** sets the maximum value of the channel bias voltage [range 0: 5100 in steps of 10 V].



In the **Monitor** section, the following parameters are monitored in real time:

- **“VMon”**: monitors the actual value of the channel bias voltage.
- **“IMon”**: monitors the actual value of the channel bias current.
- **“Status”**: monitors the channel status. Possible options are:
  - **“OFF”**: the HV channel is powered off.
  - **“ON”**: the HV channel is powered on.
  - **“Disabled”**: the HV channel is in inhibit.
  - **“Ramp Up”**: the bias voltage of the HV channel is ramping up.
  - **“Ramp Down”**: the bias voltage of the HV channel is ramping down.
  - **“Over Voltage”**: the monitored bias voltage of the HV channel is over the VSet value.
  - **“Under Voltage”**: the monitored bias voltage of the HV channel is under the VSet value.
  - **“Over Current”**: the monitored bias current of the HV channel is over the ISet value.
  - **“MaxV Protection”**: the monitored bias voltage of the HV channel is over the VMax value.
  - **“MaxI Protection”**: the monitored bias current of the HV channel is over the maximum value allowed.
  - **“Over Temperature”**: the temperature of the HV channel is over the safety limit.
  - **“Temperature Warning”**: the temperature of the HV channel is close to the over temperature condition.

## FreeWrites File Sintax

The FreeWrites file allows the user to direct access to internal FPGA registers and implement functionalities not directly managed by the DPP-PSD Control Software GUI. Some of the functionalities that can be implemented using the FreeWrites are the coincidences **[RD4]**, the PSD cut, pile-up rejection, etc.

An empty template called “**FreeWritesTemplate.txt**” can be found under the installation folder of the DPP-PSD Control Software. In particular:

**Windows:** C:\Program Files\CAEN\Digitizers\DPP-PSD Control Software\data

**Linux:** /<user\_download\_folder>/DPP-PSD\_ControlSoftware-x.x.x/

The FreeWritesTemplate file contains the instructions on how to use the FreeWrites file itself.

1. Copy the **FreeWritesTemplate.txt** file under the following path and rename it as **FreeWrites.txt**

**Windows:** %HomePath%\AppData\Local\DPP-PSD\_ControlSoftware\DppRunnerConfig.txt

Where:

- Windows XP: %HomePath% is C:\Documents and Settings\<USER>
- Windows 7/8: %HomePath% is C:\Users\<USER>

**Linux:** /home/<USER>/DPP-PSD\_ControlSoftware/DppRunnerConfig.txt



**Note:** The path might refer to hidden folders.

2. The syntax is as follows:

```
GENERIC_WRITE 0x<address> 0x<data> 0x<mask>
```

Where:

address is the hexadecimal address offset of the register (16 bit value);

data is the data to be written into the register;

mask is the bit masking for the data writing.

For example:

1. Set only bit [12] of register 0x1080 to 1, leaving the other bits to their previous value:

```
GENERIC_WRITE 0x1080 0x1000 0x1000
```

2. Set bit [12] = 1 and bit [13] = 0 of register 0x1080, leaving the other bits to their previous value:

```
GENERIC_WRITE 0x1080 0x1000 0x3000
```

3. Set register 0x1080 to the value of 0x45:

```
GENERIC_WRITE 0x1080 0x45 0xFFFFFFFF
```

Once the file is ready open the DPP-PSD Control Software. The settings will be automatically loaded by the Control Software. In case any modification of the file is made, the user can either close and restart the DPP-PSD Control Software, or modify one of the DPP settings in the GUI, as the Trigger Threshold. Every time one of the setting is modified the software reloads all the settings, including those written in the FreeWrites file.

Since the FreeWrites commands are executed at the end of the digitizer programming, the user must take care of not overwriting the other settings.

### Example 1: How to generate a global trigger from external trigger on TRG-IN connector

This example explains how to configure the FreeWrites file to generate a global trigger on the external trigger. The external trigger should be provided on the TRG-IN connector. Moreover the self-trigger of each channel will be disabled.

Registers to be set are:

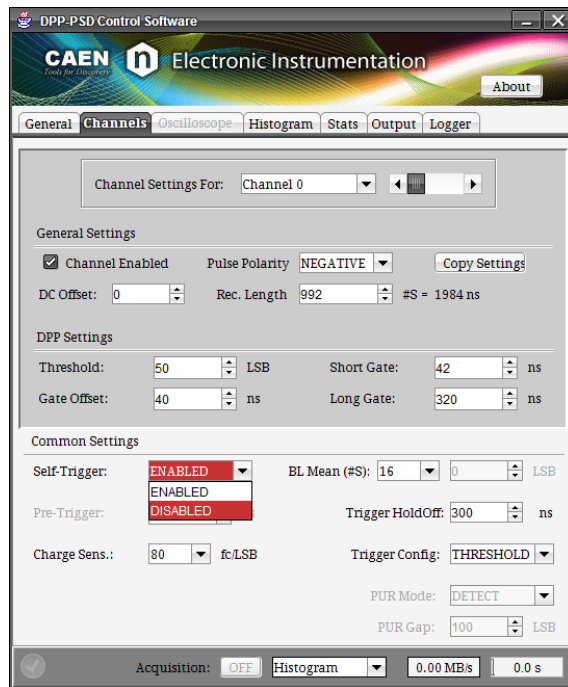
1. register 0x811C (Front Panel I/O Control) to select between NIM/TTL input logic. In particular bit[0]=0 for NIM , bit[0]=1 for TTL.
2. register 0x810C (Global Trigger Mask) to generate a global trigger on the external trigger. Set bit[30] = 1.

Write the following lines in the FreeWrites file (uncomment the relevant line):

```
GENERIC_WRITE 0x811C 0x0 0x1 # for NIM
# GENERIC_WRITE 0x811C 0x1 0x1 # for TTL
GENERIC_WRITE 0x810C 0x40000000 0xFFFFFFFF
```

Finally from the GUI disable the channel self-trigger of each enabled channel.

Under the "Channels" tab -> "Common Settings" -> "Self-Trigger" select "Disabled", as in the following picture.



### Example 2: How to cut on PSD threshold online

This example explains how to configure the FreeWrites file to set a PSD threshold and cut on gamma/neutrons (see also Sect. **Online PSD selection**). Involved registers are:

1. 0x1n78 (Threshold for the PSD cut) set the PSD threshold value for channel n. Multiply the desired PSD value by 1024 and converted it in a hexadecimal value.
2. 0x1n80 (DPP Algorithm Control) to enable the cut on gamma or neutrons for channel n.

For example, to cut on gammas at the PSD value of 0.2:

- Multiply  $0.2 * 1024 = 205$  (dec)
- Convert the number in hex:  $205$  (dec) = CD (hex)

Write on register 0x1078 (for channel 0) the value CD. The mask is 3FF (10 bits).

Then set either bit 27 or 28 of register 1080 equal to 1 to cut on gamma or neutron respectively.

The following code has to be written in the FreeWrites file (uncomment the relevant line) for channel 0.

```
GENERIC_WRITE 0x1078 0xCD 0x3FF
GENERIC_WRITE 0x1080 0x8000000 0x18000000 # to cut on gamma
# GENERIC_WRITE 0x1080 0x10000000 0x18000000 # to cut on neutrons
```

### Example 3: How to modify the input range on 725/730 series

This example explains how to configure the FreeWrites file to set the 0.5 Vpp input range on 725 and 730 series (default is 2 Vpp).

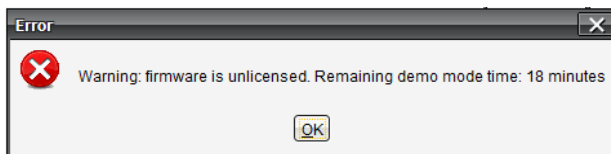
The register involved is 0x1n28 (Input Dynamic Range); set bit[0] = 1 to enable the 0.5 Vpp input on channel n.

The following code has to be written in the FreeWrites file for channel 0:

```
GENERIC_WRITE 0x1028 0x1 0x1
```

## Notes on Firmware and Licensing

The DPP-PSD supports the DPP-PSD Firmware for the 720, 725, 730, 751 series and the DT5790 CAEN digitizers. When running the DPP-PSD Control Software, the program checks the loaded firmware in the target Digitizer and pops up a warning message if no licensed version is found (**Fig. 7.21**).



**Fig. 7.21:** Firmware unlicensed warning message

The DPP-PSD Firmware is unlocked by purchasing a license from CAEN. The licensing procedure is detailed in **[RD1]** and makes use of CAENUpgrader software to finalize the unlocking. This program can also upgrade the DPP-PSD by loading new firmware versions on the digitizer.



**Note:** Download CAENUpgrader full installation package on CAEN web site at the Digitizer Tools area. Refer to **[RD1]** for detailed information and instructions on how to use it.



**Note:** The DT5790 runs a licensed version (i.e. not time limited) of the DPP-PSD Firmware. This means that no license needs to be bought by the user when purchasing a DT5790.

## 8. Technical support

CAEN makes available the technical support of its specialists at the e-mail addresses below:

[support.nuclear@caen.it](mailto:support.nuclear@caen.it)  
(for questions about the hardware)

[support.computing@caen.it](mailto:support.computing@caen.it)  
(for questions about software and libraries)

# Appendix A

## Pile-up management in DPP-PSD firmware for 751 digitizer family

### Definition of pile-up in DPP-PSD firmware (751 family)

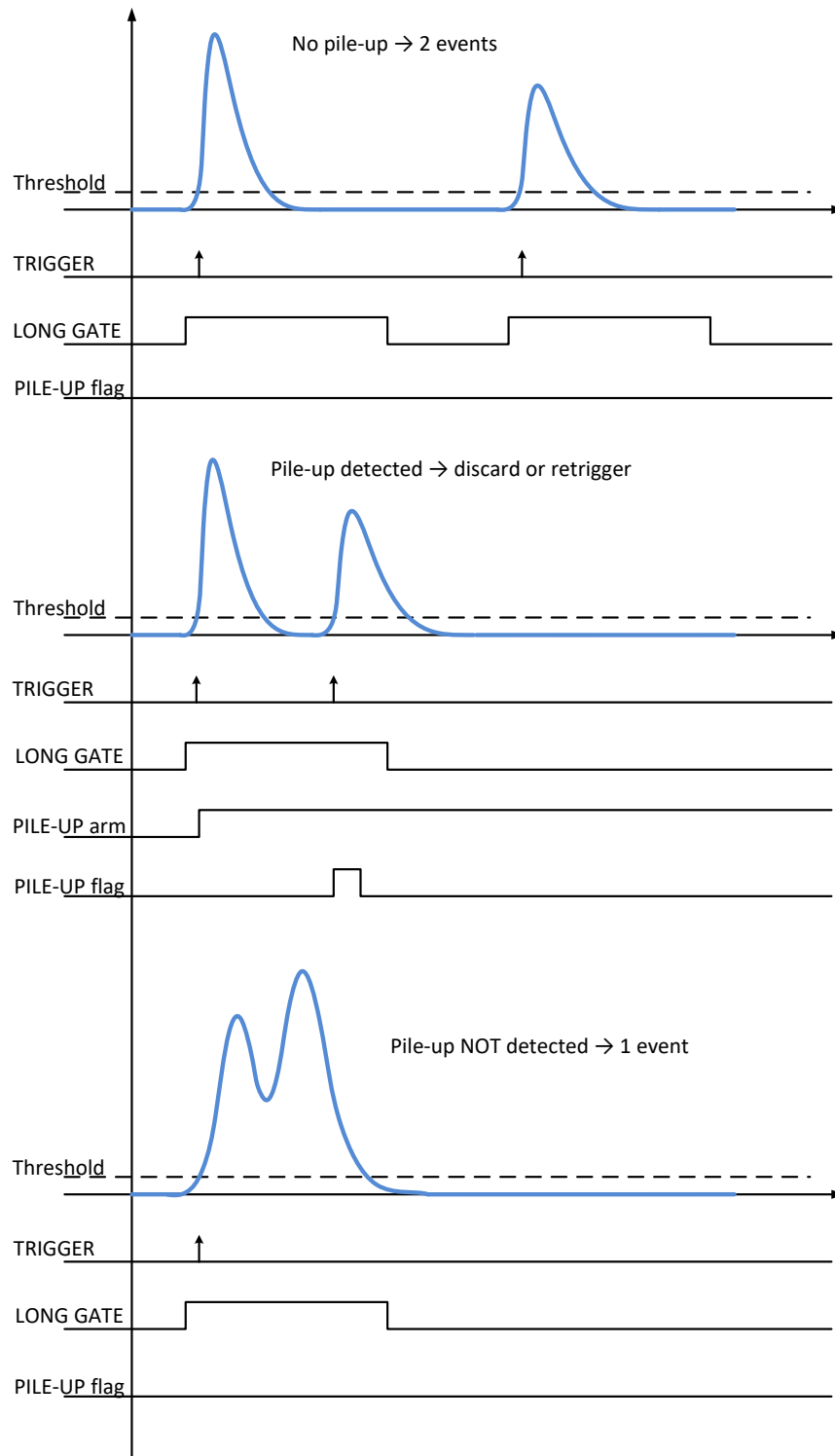
The DPP-PSD firmware is mainly designed to work with fast signals like those coming from scintillation detectors coupled with Photomultiplier Tubes. The relevant output signals do not show long decay tails as in the case of charge sensitive preamplifiers, and the probability of pile-up between two pulses is quite low. In particular, the case of a second pulse sitting on the exponential tail of the previous one is rather rare. However, with the PSD algorithm, it is important to separate fast and slow components of the light emitted by the scintillation detector. Typically, the fast component is a quick pulse (few tens of ns) while the slow component is a quite long tail (typically a few  $\mu$ s) having amplitude much smaller than the fast component (see [RD8]). To get the best results in the pulse shape discrimination, it is necessary to set the “Long Gate” as long as the full duration of the slow component. Under this conditions, most likely the events in pile-up occur during the long gate and cause an error in the calculation of the charge of the slow component. For this reason, it is important to detect these cases, especially high rate PSD acquisitions.

In the DPP-PSD firmware (751 family only) two events are considered in pile-up only when they both cross the threshold within the same integration gate. When the first signal triggers, the short and long gates are opened for integration. If another event crosses the threshold within the long gate, they are flagged as pile-up. If the second event does not cross the threshold, then the pile-up condition is not detected. Referring to **Fig. A.1**., from top to bottom there are three possible cases:

- 1) two distinct events do not overlap into the same integration gate. This is the case when no pile-up occurred;
- 2) two events trigger into the same gate. The pile-up flag is high and three possible scenarios are available:
  - a. no action is taken and the two pulses are integrated into the same integration gate;
  - b. the event is discarded and no charge is evaluated and saved;
  - c. a second gate is opened for the second pulse.

See next section for further details;

- 3) two pulses overlap into the same gate, and the second pulse does not overcome the threshold. The event is not recognized as pile-up.

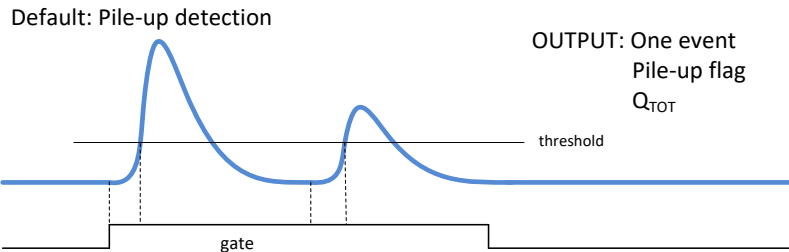


**Fig. A.1:** From top to bottom: (1) two distinct events do not overlap into the same integration gate. This is the case when no pile-up occurred. (2) Two events trigger into the same gate. The pile-up flag is high and three possible scenarios are available: (a) no action is taken and the two pulses are integrated into the integration gate; (b) the event is discarded and no event is saved; (c) a second gate is opened for the second pulse (see next section for further details). (3) Two pulses overlap into the same gate, but the second pulse does not overcome the threshold. The event is not recognized as pile-up

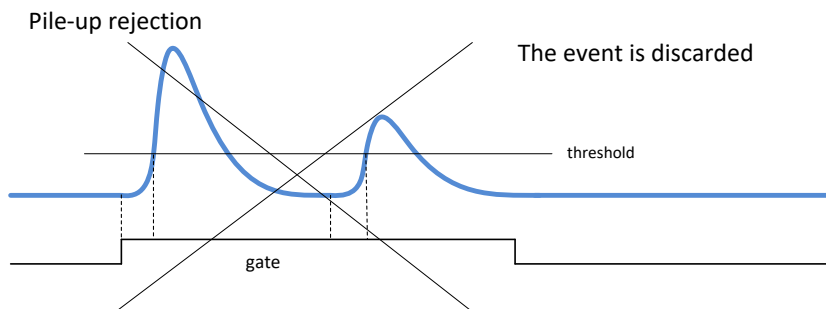
## Pile-up management in DPP-PSD firmware (751 family)

In the DPP-PSD firmware there are three options for the pile-up management:

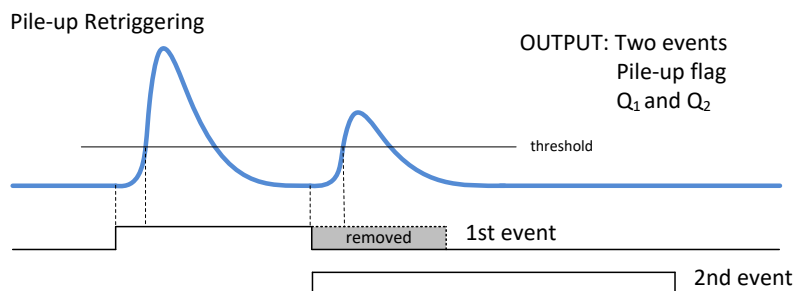
- 1) **Default:** in the default configuration the algorithm can detect pile-up events, but no action is taken. Indeed the firmware evaluates the total charge of the two pulses – which is therefore considered as a single “event”, and flags that event as “pile-up” (bit 16 of  $Q_{SHORT}$ ).



- 2) **Pile-up rejection:** this configuration is enabled by setting bit [26] = 1 of register 0x1n80 (where n is the channel number), or register 0x8080 for broadcast write. Events recognized as pile-up are completely rejected by the firmware. In this mode, the readout throughput rate is reduced and the rejected events are not anymore available for software readout.



- 3) **Pile-up Retriggering:** this configuration is enabled by setting bit [25] = 1 of register 0x1n80 (where n is the channel number), or register 0x8080 for broadcast write. In this case, when another pulse arrives within the long gate (and after the end of the short gate), the charge integration is stopped prematurely. A new event is created, by opening short and long gate again. The charge in the first event is the result of the integration of the signal up to the start of the new gate; because of the truncation of the gate (see figure below), it is not guaranteed that the full charge (i.e. energy) of the first pulse is completely integrated in the first gate. The second gate will integrate the charge of the second pulse, including the pre-gate region; it is worth noticing that the second gate can integrate also part of the charge belonging to the first gate. The user can apply some corrections to the charge of the two events based on the two time stamps, that give information about the separation between pulses. Both events are tagged as “pile-up”.



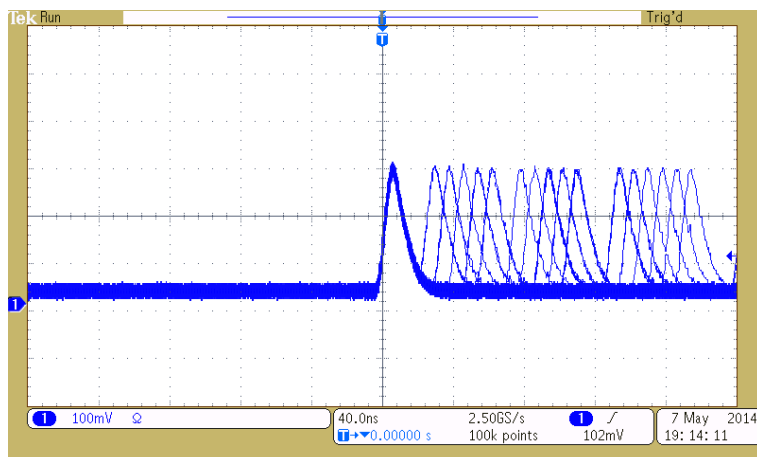


## Pile-up with the DPP-PSD Control Software (751 family)



**Note:** DPP-PSD Control Software does not manage yet the pile-up. Anyway it can be used for a quick check of the pile-up behaviour (see the following steps). Users must write their own code to decode the events with the pile-up flag. Check Sect. Event Data Format for further details about the event structure.

**Input signal:** We used the digital detector emulator DT5800D (refer to [RD10]) to generate exponential signals with *constant amplitude* and *Poisson time distribution* with mean frequency of **100 KHz** (see the oscilloscope picture below).

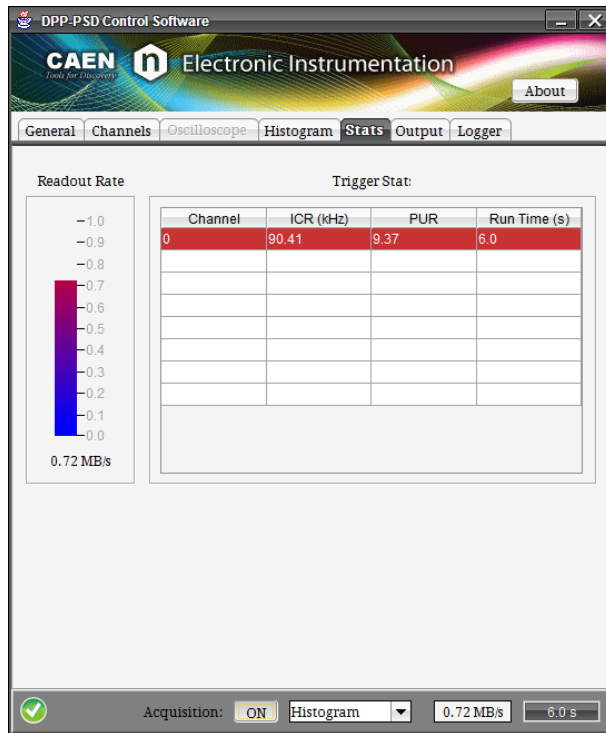


**DPP-PSD Control Software** parameters:

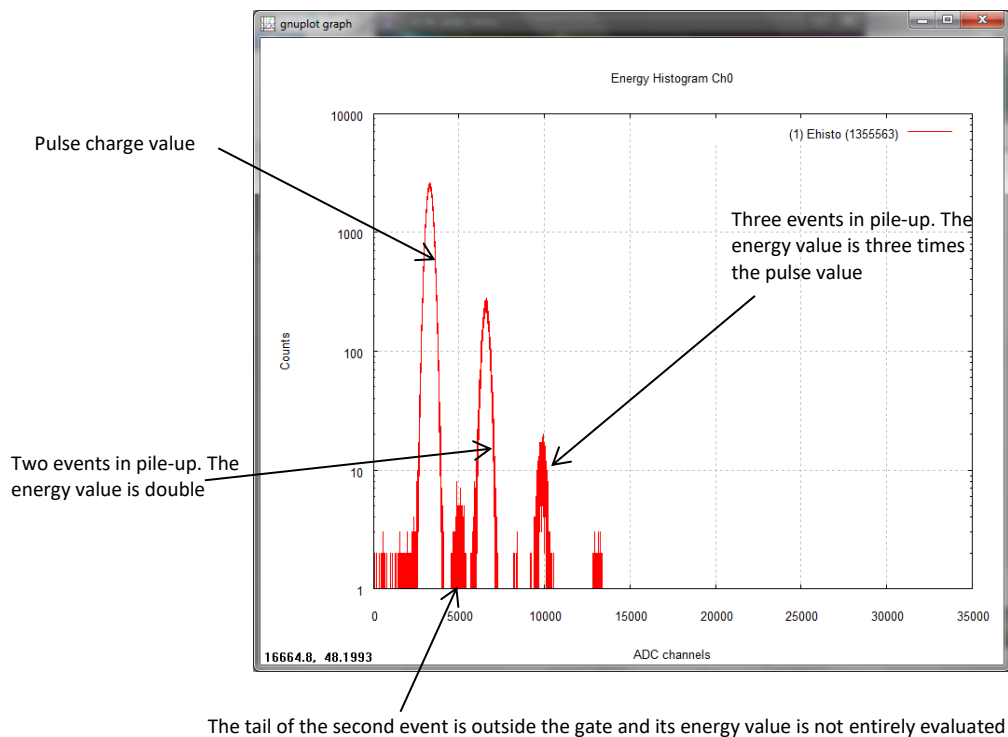
- **positive** polarity;
- **trigger** threshold set to 50 LSB;
- Long **Gate** equal to 1 us (i.e. much longer than the signal itself).

We are going to test the three pile-up settings.

- 1) **Default:** Once the DPP-PSD parameters are correctly set, we can enable the histogram mode. The tab STATS should appear as in the following figure. The ICR (Incoming Counting Rate) value is less than 100 KHz since there are a certain number of events falling in the same integration gate, thus they are counted once. The PUR value is the percentage of events tagged as "pile-up". In this case, ICR=90.41 KHz and PUR=9.37, so  $90.41 + 9.37\% = 98.88$  KHz, which is almost the actual rate (100KHz). The missing 1.12% is due to multiple pile-up conditions.

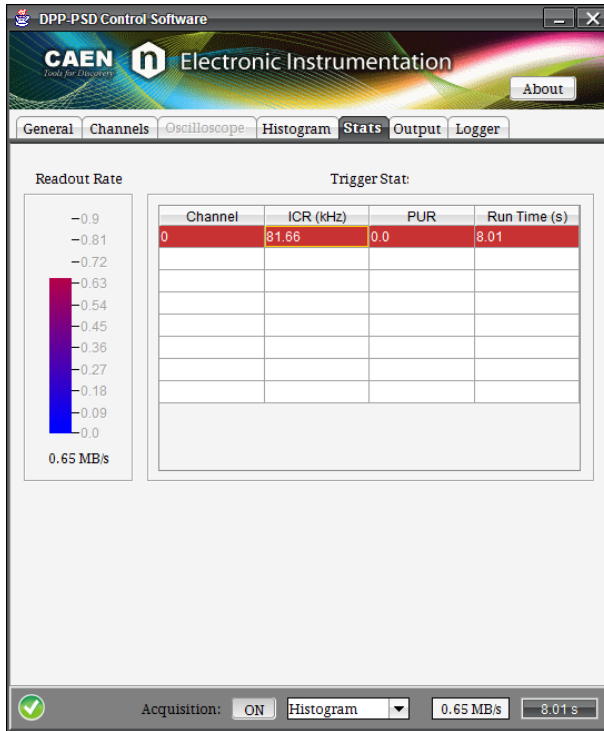


The Histogram plot is as follows, where the peak at about 3000 (x-axis) corresponds to the energy value of the input pulse, the peak at about 6000 corresponds to those events where two input pulses were overlapping and could not be recognized as pile-up, etc... Values in the middle correspond to pile-up events, where the tail of the second event is outside the integration gate.



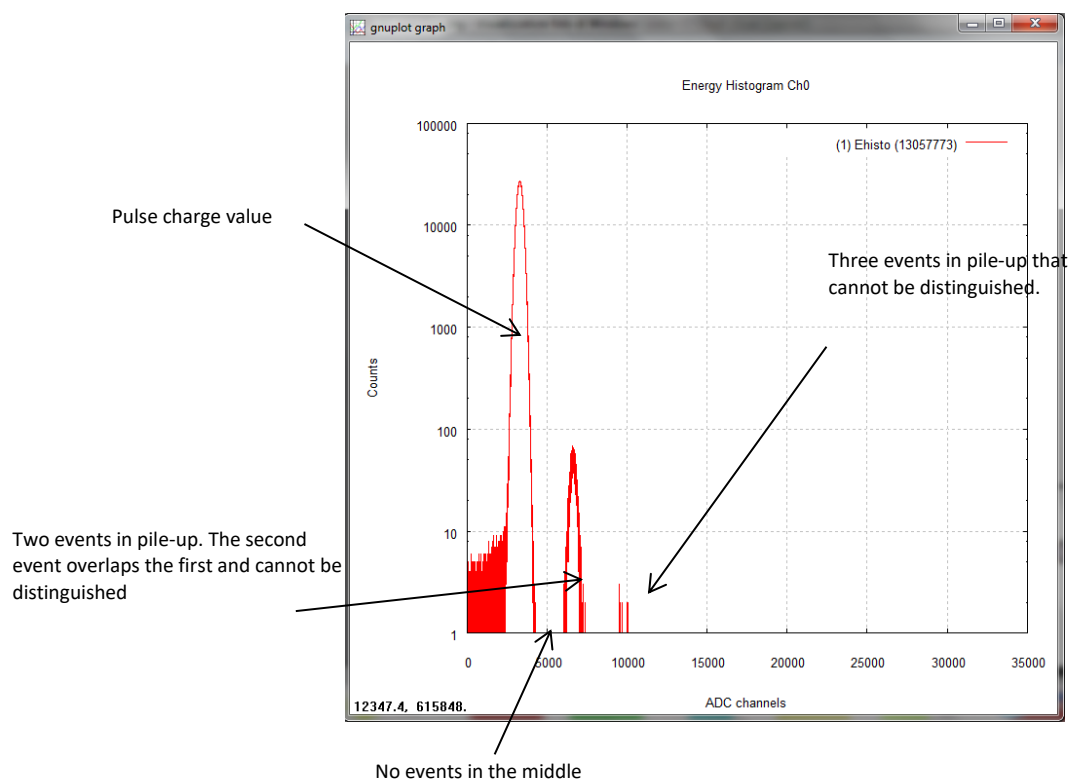
- 2) **Pile-up rejection:** Once the DPP-PSD parameters are correctly set, we can enable the pile-up rejection bit (bit[26]=1 of register 0x1n80).

Pile-up events are completely rejected at the firmware level, therefore they are not even processed by the software. The STATS tab will appear as follows:

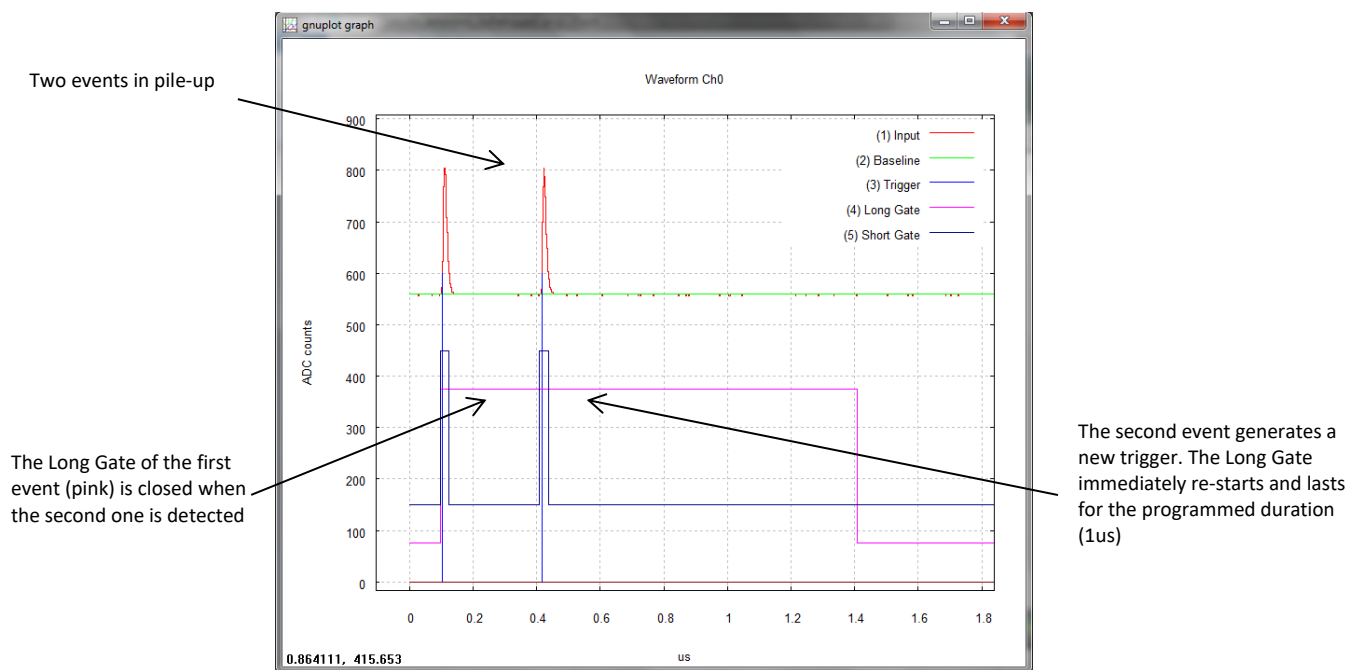


The PUR (pile-up) value is zero (events have been already rejected) and the ICR (incoming counting rate) corresponds to the ICR value of the “default” configuration minus the PUR value of the default configuration itself (ICR=90.41 KHz and PUR=9.37, so  $90.41 - 9.37\% = 81.04$  KHz).

Since none of the events has the pile-up flag, it is possible to use the DPP-PSD Control Software as is.



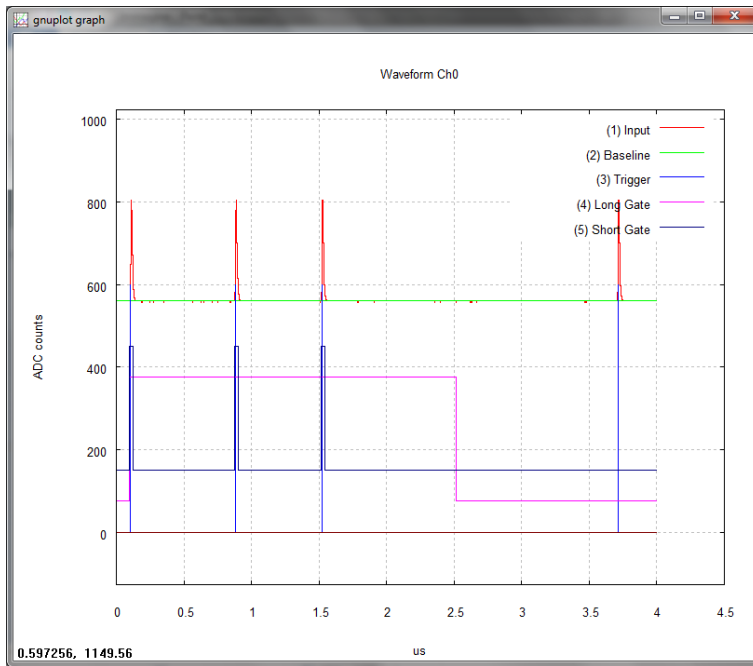
- 3) **Pile-up Retriggering:** Once the DPP-PSD parameters are correctly set, we can enable the pile-up extension bit (bit[25]=1 of register 0x1n80).



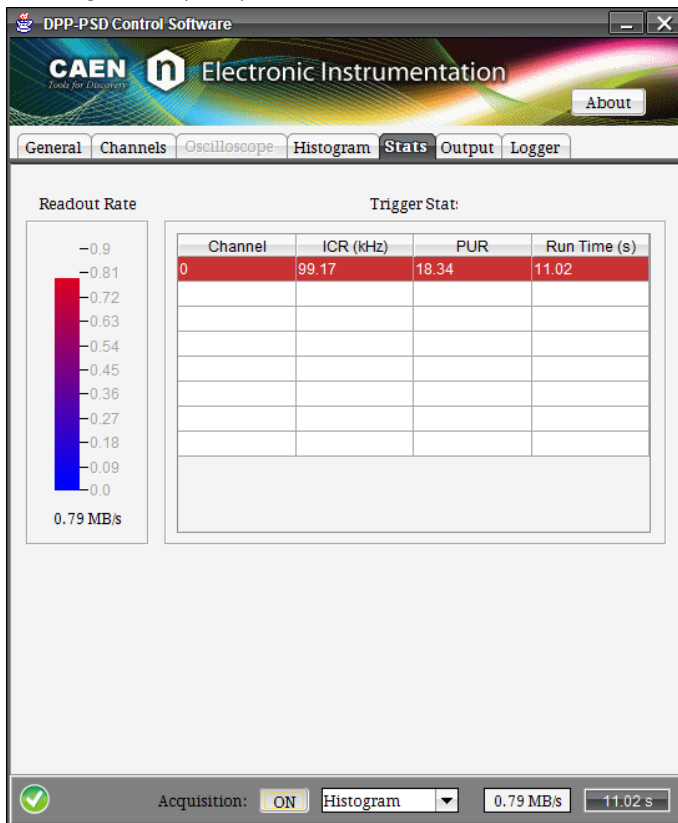
Two events are saved with their specific integrated charge, and pile-up flag.

In our example the two charges are the same. In real experiments the tail of the first event may be cut off by the first gate, and included in the second event.

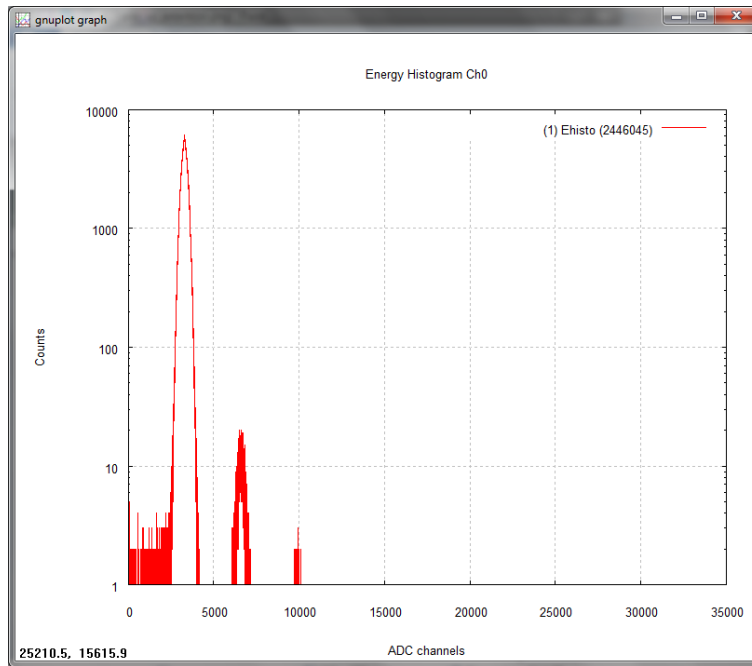
Here an example of three events in pile-up. The long gate stops and restarts for the two more events.



Enable the histogram acquisition mode. The STATS tab appears as follows. The PUR percentage is double than the PUR percentage of the “default” configuration, while the ICR is close the 100 KHz, with a slight difference due to undistinguishable pile-up events.



Histogram plot appears as in the following picture:



The histogram is quite similar to the histogram of the “pile-up rejection” configuration, though the rate is quite higher.

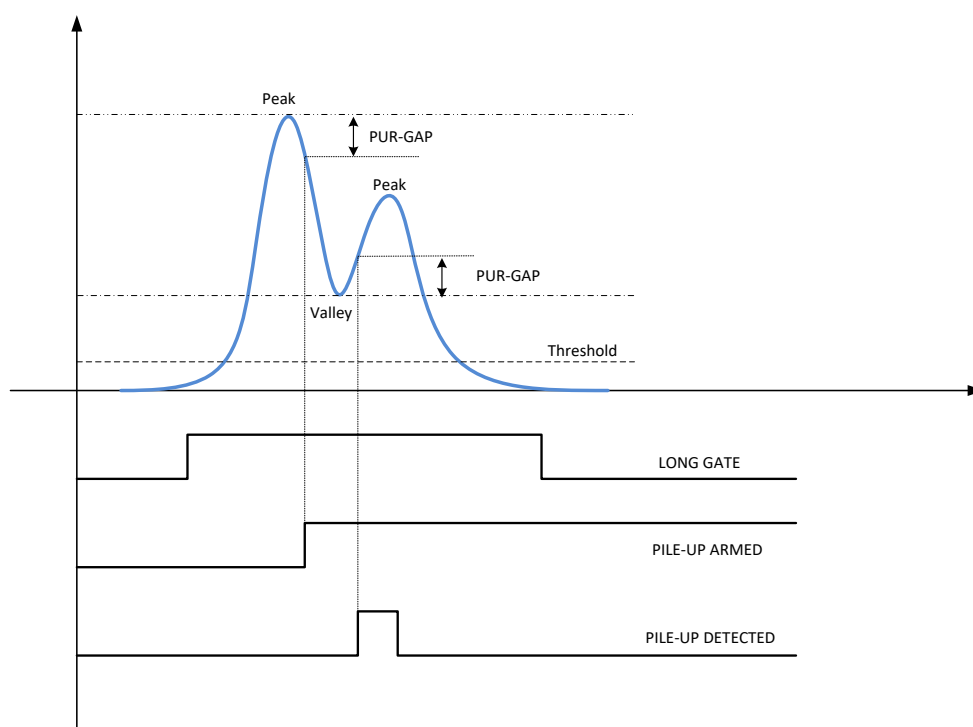
# Appendix B

## Pile-up management for 720, 725 and 730 digitizer family

### Definition of pile-up in DPP-PSD firmware (720, 725 and 730 family)

In the DPP-PSD firmware (720, 725 and 730 family only) two events are considered in pile-up when there is a situation of peak-valley-peak inside the same gate, where the gap between the valley and the peak is a programmable value.

Referring to **Fig. B.1**, when the peak value is reached the algorithm evaluates the point corresponding to the PUR-GAP value and gets ready to detect a pile-up event (PILE-UP ARMED). If there is a condition of “valley”, and the input signal overcomes the PUR-GAP threshold, then the event is tagged as pile-up. In the default configuration the firmware does not take any action and the total charge of the event is evaluated within the gate and saved into memory.



**Fig. B.1:** Pile-up definition for 720, 725, and 730 series

### How to set the pile-up rejection for DPP-PSD firmware (720, 725 and 730 family)

Those who want to enable the pile-up rejection have to modify the following two registers:

- PUR-GAP value, set register 0x1n7C (where n is the channel number). Accepted values range from 0 to 4095;
- Pile-up rejection, set bit [26] = 1 of register 0x1n80 (DPP-CTRL).

When the pile-up rejection is enabled, events flagged as pile-up are rejected by the firmware itself, therefore they are no more available for readout.



CAEN SpA is acknowledged as the only company in the world providing a complete range of High/Low Voltage Power Supply systems and Front-End/Data Acquisition modules which meet IEEE Standards for Nuclear and Particle Physics. Extensive Research and Development capabilities have allowed CAEN SpA to play an important, long term role in this field. Our activities have always been at the forefront of technology, thanks to years of intensive collaborations with the most important Research Centres of the world. Our products appeal to a wide range of customers including engineers, scientists and technical professionals who all trust them to help achieve their goals faster and more effectively.

**CAEN S.p.A.**

Via Vetraria, 11  
55049 Viareggio  
Italy  
Tel. +39.0584.388.398  
Fax +39.0584.388.959  
info@caen.it  
www.caen.it

**CAEN GmbH**

Klingenstraße 108  
D-42651 Solingen  
Germany  
Phone +49 (0)212 254 4077  
Fax +49 (0)212 25 44079  
Mobile +49 (0)151 16 548 484  
info@caen-de.com  
www.caen-de.com

**CAEN Technologies, Inc.**

1140 Bay Street - Suite 2 C  
Staten Island, NY 10305  
USA  
Tel. +1.718.981.0401  
Fax +1.718.556.9185  
info@caentechnologies.com  
www.caentechnologies.com