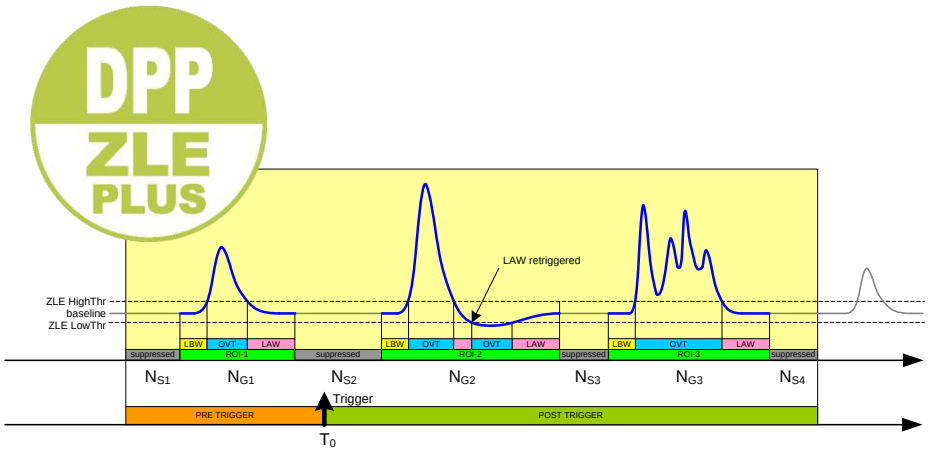
CAEN
Tools for Discovery

Electronic Instrumentation



User Manual UM2764

DPP-ZLEplus User Manual

Digital Pulse Processing for Zero Length Encoding

Rev. 0 - 17 June 2014

Purpose of this Manual

This User Manual contains the full description of the DPP-ZLE*plus* for the 751 digitizer series. The description is compliant with DPP-ZLE*plus* firmware release **4.2_133.3**, and DPP-ZLE*plus* Control Software release **1.0**. For future release compatibility check the firmware and software revision history files.

Change Document Record

Date	Revision	Changes
17 June 2014	00	Initial release

Symbols, abbreviated terms and notation

ADC	Analog to Digital Converter
DPP	Digital Pulse Processing
FSR	Full Scale Range
ZLE	Zero Length Encoding

Reference Documents

[RD1] GD2512 - CAENUpgrader QuickStart Guide

[RD2] GD2783 – First Installation Guide to Desktop Digitizers & MCA

All documents can be downloaded at: http://www.caen.it/csite/LibrarySearch.jsp

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

Index

List of Figures

List of Tables

1 Introduction

CAEN S.p.A. provides a wide family of digitizers for many fields of application. Proper firmware running on board allows to work with both waveforms and histograms. Two main firmware categories are available for CAEN digitizers: *standard firmware* and *digital pulse processing (DPP) firmware*.

The present manual is meant to describe a special DPP firmware for Zero Length Encoding (DPP-ZLEp*lus* firmware) and the demo software to start using the firmware. Source codes are also available for those who want to customize their software.

The ZLE algorithm is based on the zero suppression algorithm, i.e. on the idea to remove useless data (zeros) from an acquisition. The great advantage is to compress the data and to reduce the throughput. The DPP-ZLE*plus* algorithm is mainly intended for those experiments where the amount of “interesting” information is quite low with respect to the total information available. For example when there are large acquisition windows with respect to the length of a single event, or when the input signal rate is low.

If the whole acquisition window was entirely transmitted, many of the information would be related to samples with no events. Those samples correspond to what we call “signal baseline”. If the signal baseline is frozen for the whole acquisition window, the algorithm can only transfer the triggered events and the number of samples corresponding to the baseline only. Indeed the DPP-ZLE*plus* firmware dynamically evaluates the signal baseline, and freezes its value at the beginning of the acquisition window. This greatly improves the memory efficiency and the readout throughput.

2 Principle of operation

The Digital Pulse Processing for Zero Length Encoding Plus (DPP-ZLE*plus*) is a special firmware developed for 751 family of CAEN digitizers. It allows to transfer the event in compressed mode, discarding data included in a band around the baseline. Indeed this algorithm is mainly intended to discard any sample that doesn't contain useful information, such as the baseline samples. Indeed if the baseline is dynamically evaluated and frozen at the beginning of the acquisition window, then it is possible to transmit only the number of samples corresponding to the baseline.

An important feature of the DPP-ZLE*plus* is that it requires an external trigger to start the acquisition window. The trigger has to be sent to the TRG IN connector. Then the user can define "Pre-Trigger" and the "Record Length" values, which are therefore referred to the external trigger. The algorithm calculates the "baseline" of the input signal and searches for the significant input pulses that cross the programmable "high" and "low threshold".

Note: The external trigger is a global trigger, common to all the channels of the board.

The ZLE*plus* algorithm works for all the digitizer channels, and each channel can work independently from the others. The operation of the ZLE algorithm is shown in **Fig. 2.1:.** The main parameters involved are:

- Pre-Trigger
- Record Length
- High Threshold
- Low Threshold
- Number of Look Back Window (NLBW)
- Number of Look Ahead Window (NLAW)

The user can program the corresponding number of samples of those settings, where one sample corresponds to 1 ns. Where explicitly written the number of sample is multiplied by 8. For example, the acquisition window starts *Pre-Trigger**8 samples (corresponding to *Pre-Trigger**8 ns) before the TRG IN itself and it lasts for the whole *Record Length* duration (corresponding to the *Record Length* value in ns).

The baseline mean value can be evaluated in a moving window of N programmable samples. The baseline then remains frozen for the whole acquisition window.

Within the acquisition window, the algorithm searches for the signal exceeding the *High Threshold* (ZLE High Thr) or falling below the *Low Threshold* (ZLE Low Thr). The signal recording starts *NLBW* samples before the threshold crossing (LBW in the picture) and stops *NLAW* samples after the crossing in the opposite direction (LAW in the picture). The recorded regions defines the ROI (region of interest) of the algorithm. Outside the ROI the algorithm registers the fixed value of the baseline.

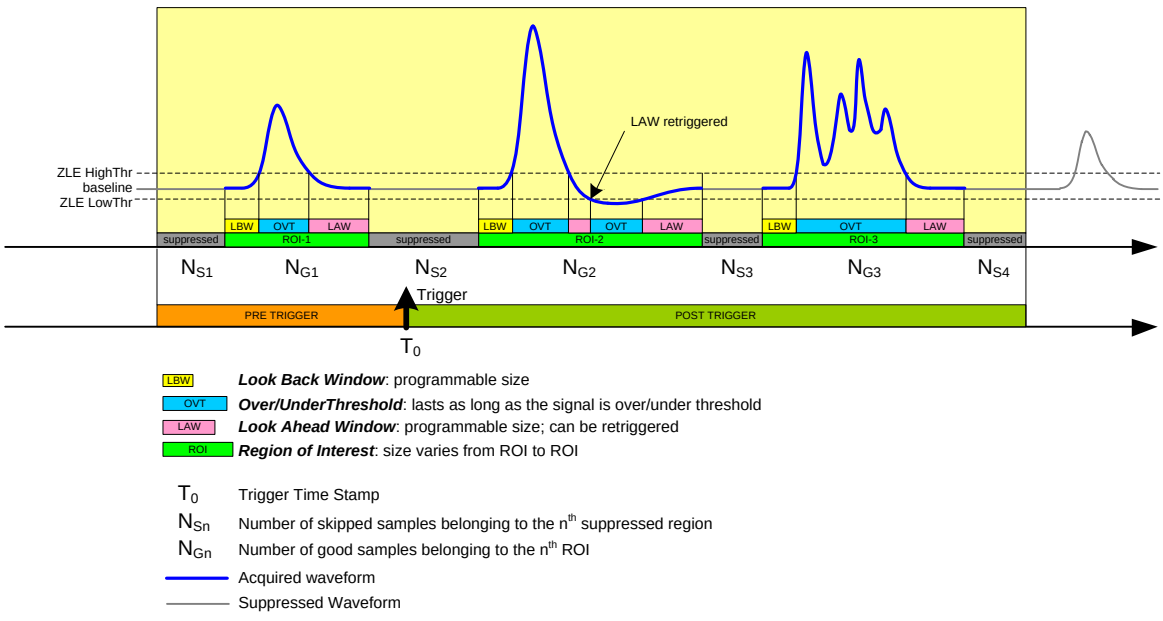


Fig. 2.1: DPP-ZLE*plus* algorithm description

Each trigger (TRG IN) determines the event building. The board provides the data format described in **Tab. 2.1**. Those who want to write their own acquisition code can use the "DecodeZLEWaveforms" function of the CAENDigitizer libraries, that automatically manages the event structure.

EVENT 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
CHANNEL EVENT SIZE																																HEADER
EVENT TRIGGER TIME TAG																																
BASELINE VALUE																																

					EVENT DATA
0	0	NUMBER OF SKIPPED SAMPLES (N_{S0}) * 8			
1	1	SAMPLE 2 (ROI 1)	SAMPLE 1 (ROI 1)	SAMPLE 0 (ROI 1)	
1	1	SAMPLE ... (ROI 1)	SAMPLE ... (ROI 1)	SAMPLE ... (ROI 1)	
1	1	SAMPLE J (ROI 1)	SAMPLE J-1 (ROI 1)	SAMPLE J-2 (ROI 1)	
0	0	NUMBER OF SKIPPED SAMPLES (N_{S1}) * 8			
1	1	SAMPLE 2 (ROI 2)	SAMPLE 1 (ROI 2)	SAMPLE 0 (ROI 2)	
1	1	SAMPLE ... (ROI 2)	SAMPLE ... (ROI 2)	SAMPLE ... (ROI 2)	
1	1	SAMPLE K (ROI 2)	SAMPLE K-1 (ROI 2)	SAMPLE K-2 (ROI 2)	
...					
0	0	NUMBER OF SKIPPED SAMPLES (N_{Slast}) * 8			

Tab. 2.1: DPP-ZLE*plus* event format

The event format consists of two parts, the header and the event data.

The three-line **header** has the following information:

- **Channel event size**, corresponds to the number of 32 bit-word to be read in the event;
- **Event trigger time tag**, corresponds to the time where the external trigger occurred;
- **Baseline value**, is the value of the baseline. The baseline remains frozen for the whole event.

The **event data** has the following structure:

- Words corresponding to skipped samples:
  - Bits [31:30] = 00;
  - Bits [29:0] = number of skipped sample in groups of 8 samples, i.e. the value has to multiplied by 8 to get the real number of skipped samples.
- Words corresponding to good samples:
  - Bits [31:30] = 11;
  - Samples are represented as 10 bit numbers, where each sample is written in bits [9:0], bits [19:10], and bits [29:20] of each word of the event structure.

The **software** then decodes the information in the following **output file format**:

- Four lines header:
  - Record Length;
  - Channel Number;
  - Event Baseline;

  - Trigger Time Tag.
- Two columns data:
  - First column = data sample;
  - Second column = 1 if the sample is reconstructed (i.e. baseline), 0 if the sample is read from input.

The number of skipped samples is provided in order to preserve the timing information within the event, for an offline event reconstruction.

The ZLE*plus* features a high flexibility allowing the user for programming all the specific parameters of the algorithm for the application:

- The size of the acquisition window (*Record Length*) is programmable from 0 to about 8 seconds. See the **RECORD_LENGTH** register.
- The *Pre-Trigger* is programmable in the range of 0 to about 8 us. The user can set a value from 0 to 1023, where this value is then multiplied by 8 by the algorithm. See the **PRE_TRIG** register.
- *High Threshold* and *Low Threshold* for the ZLE*plus* are programmable in the full scale range (from 0 to 1023 ADC counts). Their values are relative to the baseline, when the baseline calculation is enabled; they are absolute values when the baseline calculation is disabled. See **LOW_THRESHOLD** and **HIGH_THRESHOLD** registers.
- The size of LBW and LAW (i.e. *NLBK* and *NLAW*) are programmable in the range of 0 up to 8184 samples. The user can set a value from 0 to 1023, where this value is then multiplied by 8 by the algorithm. See **NLBW_SAMPLES** and **NLAW_SAMPLES** registers.
- The baseline in dynamically calculated over a programmable number of samples (choosing among 0, 8, 16, 32, 64, 128, 256, or 512 samples). See the **BASELINE_AVERAGE_SEL** register. The baseline is frozen at the beginning of the acquisition window, and it remains frozen for the whole window duration. When zero is selected then the baseline is not evaluated, and the values of High and Low Threshold becomes absolute values.
- The user can select an acceptance bandwidth for the baseline calculation. Samples with values outside the bandwidth are not included in the baseline calculation. See **BASELINE_THRESHOLD (blThr)** register. It is also possible to set the maximum amount of time ("Baseline Timeout") the signal can stay outside the acceptance bandwidth before being considered again for the baseline calculation (**Fig. 2.2**). See the **BASELINE_TIMEOUT (blTo)** register.

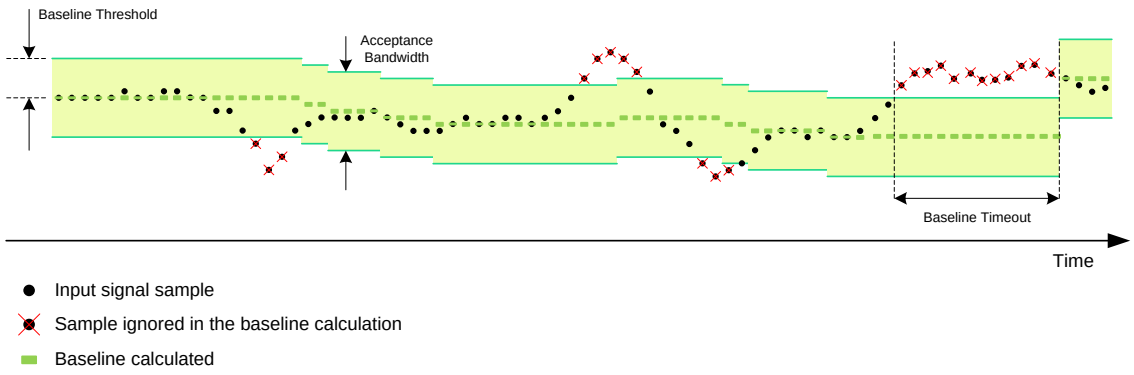


Fig. 2.2: Baseline calculation of the DPP-ZLE*plus* algorithm.

Besides:

- External Trigger is latched by a 62.5 MHz clock (16 ns timing granularity). Indeed the external trigger is sampled by the mother board every 8 ns, and it is transferred to the mezzanine every 16 ns.
- All ZLE*plus* timings (LBW, LAW, ROI, etc.) are affected by an uncertainty of 24 samples (24 ns), where 16 are due to the mother board trigger transferring, and 8 are due to the mezzanine data clustering.

Compared to the old ZLE algorithm for V1721, the ZLE*plus* developed for the 751 digitizer series offers the following advantages:

- Double Threshold (positive and negative)
- On-line Baseline calculation
- Thresholds relative to the baseline
- Unlimited number of ROIs in the acquisition window
- Readout bandwidth reduction

Notes on Firmware and Licensing

The DPP-ZLE*plus* firmware runs on 751 digitizer series only. The supported digitizer models and the DPP firmware license are listed in the table below.

Desktop Digitizers(*)	Description	Product Code
DT5751	2/4 Ch. 10 bit 2/1 GS/s Digitizer: 1.8/3.6MS/ch, EP3C16, SE	WDT5751XAAAA
NIM Digitizers(*)	**Description**	**Product Code**
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	WN6751CXAAAA
VME Digitizers(*)	**Description**	**Product Code**
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
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
DPP Firmware(*)	**Description**	**Product Code**
DPP-ZLE*plus*	Digital Pulse Processing for Zero Length Encoding Plus	WFWDPPZLAA51

Tab. 2.2: Compliance table of the ZLE*plus* with CAEN digitizers and DPP firmware.
(*) For accessories and customizations related to digitizers and for multiple DPP-ZLEplus license packs, refer to the board User Manual or browse on CAEN web site: www.caen.it

The DPP-ZLE*plus* firmware is available for trial version on www.caen.it:

Home / Products / Firmware/Software / DPP Firmware/Software Tools (Digitizer) / DPP Firmware / DPP-ZLEplus

The trial version is 30 minutes limited per power cycle; this means that all the firmware functionalities are free available for 30 minutes after power on, then the board needs a power cycle to exploit other 30-minutes of usage.

The DPP-ZLE*plus* firmware requires to be licensed in order to unlock the time limitation. The user can purchase a license from CAEN (refer to the **Tab. 2.2** above) following the licensing procedure detailed in **[RD1]** and can use the CAENUpgrader software to finalize the unlocking. When a new firmware release is available online, the CAENUpgrader tool can be used to upgrade the firmware on a x751 digitizer. The upgrade does not erase the firmware license.

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

Note: When ordering one of digitizers in the list above together with a DPP-ZLE*plus* license, the user will receive the board with the DPP firmware already loaded on it and unlocked.

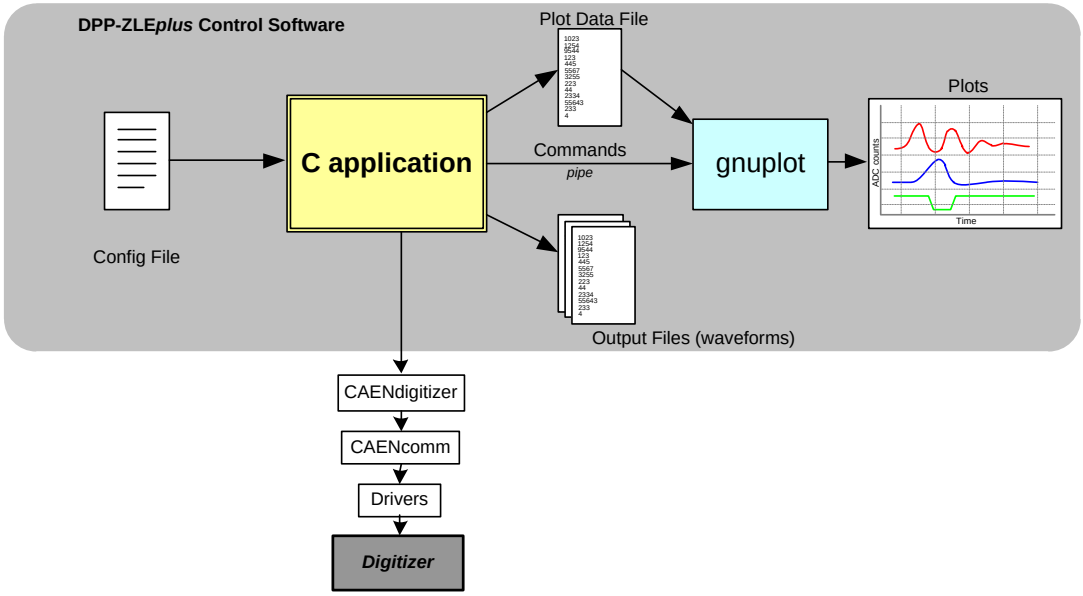
3 Software Interface

Introduction

CAEN provides a demo software in C for the ZLE*plus* readout, called ZLE*plus* Control Software. The user can make an entire acquisition through this software, as well use the source code to develop his/her customized readout program. Indeed the package includes the C source files and the Visual Studio project (compliant with Visual Studio Professional 2010).

Block Diagram

The Control Software is a C-based application that programs the Digitizer according to a set of parameters in the configuration text file, starts/stops the acquisition and manages the data readout. The waveforms elaborated by the ZLE*plus* algorithm are plotted using *gnuplot*, an external plotting tool, or saved to output text files.



Tab. 3.1: DPP-ZLE*plus* Control Software block diagram

Libraries and Drivers

CAEN provides the drivers for all the three different types of physical communication featured by the x751 digitizers running the DPP-ZLE*plus* firmware. Drivers are compliant with Windows and Linux OS:

- **USB 2.0**. The driver installation package is available on CAEN website in the 'Software/Firmware' area at the Digitizer or V1718 page;
- **CONET Optical Link**, managed by the A2818 PCI card or A3818 PCIe card. The driver installation package is available on CAEN website in the 'Software/Firmware' area at the A2818 or A3818 page;
- **VME bus**, accessed by the V1718 and V2718 bridges;

in addition there is a set of C and LabView libraries. All the required libraries are already included in the setup for Windows OS, while they have to be prior installed in Linux OS.

- **CAENVMELib** is a set of ANSI C functions which allows to program the use and the configuration of CAEN Bridges V1718/VX1718 (VME-USB2.0 Bridge), V2718/VX2718 (VME-PCI Optical Link Bridge), A2818/A3818 (PCI CONET Controller).

  The CAENVMElib installation package is available on CAEN website in the 'Download' area at the CAENVMELib Library page.

- **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 levels, masking the details of the physical channel and its protocol, thus making the libraries and applications that rely on the CAENComm independently from the physical layer. Customers not using CAEN VME bridges can write their own CAENComm functionalities according their custom protocol and use higher level CAEN libraries.

  CAENComm is based on CAENVMElib and it requires the CAENVMELib library (access to the VME bus) even in those cases where the VME is not used. This is the reason why it is necessary that **the CAENVMELib is already installed on your PC before installing the CAENComm**.

  The CAENComm installation package is available on CAEN website in the 'Download' area at the CAENComm Library page.

- **CAENDigitizer** is a library of functions designed specifically for the Digitizer family and it supports also the boards running the DPP firmware, as it happens in the DPSD. The CAENDigitizer library is based on the CAENComm which is based on CAENVMELib, as said above. For this reason, **the CAENVMELib and 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.

As far as the x751 digitizer series is concerned, the CAENComm (and so the CAENDigitizer) supports the following communication channels (see also **Tab. 3.2**):

PC → USB → Digitizer DT5751 or N6751 - Desktop and NIM models

PC → USB → V1718/VX1718 → VME → Digitizer V1751/VX1751 - VME models

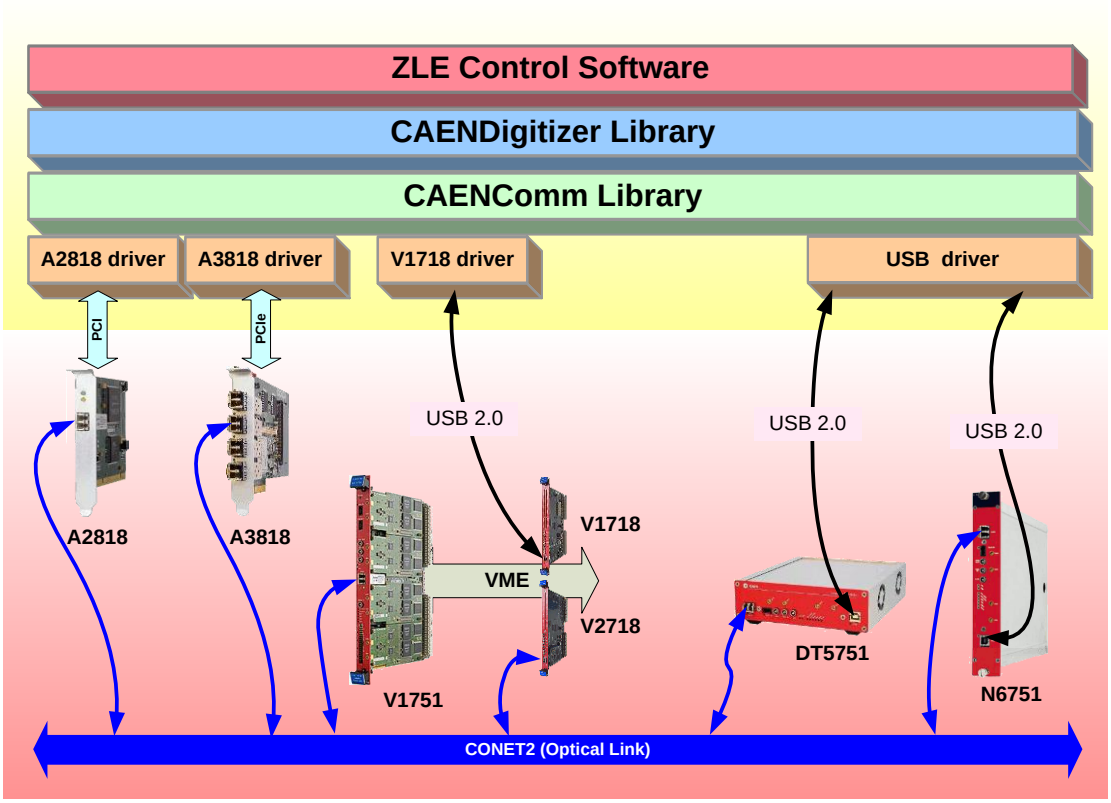
PC → PCI (A2818) → CONET → Digitizer x751 - All models of the x751 series

PC → PCI (A2818) → CONET → V2718/VX2718 → VME → Digitizer V1751/VX1751 - VME models

PC → PCIe (A3818) → CONET → Digitizer x751 - All models of the x751 series

PC → PCIe (A3818) → CONET → V2718/VX2718 → VME → Digitizer V1751/VX1751 - VME models

where **CONET** (Chainable Optical NETwork) indicates the CAEN proprietary protocol for communication on Optical Link.



Tab. 3.2: **Libraries and drivers required by the 751 digitizer series equipped with the DPP-ZLE*plus* firmware**

Installation

The DPP-ZLE*plus* Control Software is compliant with both Windows and Linux OS, 32 and 64 bits.

Before installing the DPP-ZLE*plus* Control Software perform the following steps:

- **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** that the **DPP-ZLE*plus* firmware** is **running on the board** (refer to **[RD1]** for firmware upgrade).
- **Make sure** you **have installed the driver** for your OS and the physical communication layer to be used. Driver installation packages are downloadable on CAEN website (**login required before to download**) as reported in the **Libraries and Drivers** paragraph (refer to the related User Manual for driver installation instructions).

For Windows users:

CAEN provides the full installation package for the **DPP-ZLE*plus* 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).

- **Download the DPP-ZLE*plus* Control Software installation package** compliant with your OS from CAEN website under the 'Download' area at the DPP-ZLE*plus* Control Software page (**login required before to download**).
- **Extract files** to your host.
- **Complete** the **installation wizard**.

The *ZLEControlSoftware* is installed under the folder (see **Fig.** 3.1 below):

C:\Program Files\CAEN\Digitizers\ZLEControlSoftware\

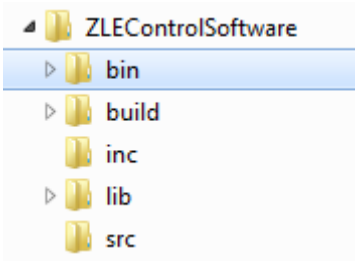


Fig. 3.1: Folder path of the DPP-ZLE*plus* Control Software executable file

The "*bin*" subfolder contains the executable file (CAENZLEControlSoftwareSetup-1.0.exe) and the configuration file (ZLEControlSoftwareConfig.txt)

Note: Administrator rights are required to modify the *configuration file* of the DPP-ZLE*plus* Control Software under the "*Program Files*" folder. To modify the file and use the software without the administrator rights, copy the entire "bin" folder under another location, as for example the "Documents" folder.

Under the *"build"* folder there is the Visual Studio project, while in the *"inc"* and *"src"* folders there are the header and the source code of the DPP-ZLE*plus* Control Software, respectively.

For Linux users:

Linux users must also take care of proper installation of **gnuplot** graphical tool, as well as of the **CAEN Libraries**: CAENVMElib, CAENComm, CAENDigitizer. The latter can be downloaded from CAEN website (login required before to download). **Installation instructions** can be found in the **README file** inside each library folder.

- **DPP-ZLE*plus* Control Software** for Linux.**Download** the ZLEControlSoftware-1.0.tar.gz package on CAEN website in the 'Download' area at the DPP-ZLE*plus* Control Software page (**login is required before the download**).

- **Unpack** the **installation package** (tar –zxf ZLEControlSoftware-1.0.tar.gz).
- **Follow** the instruction on the **INSTALL** file

  Type: ./configure

  make

  sudo make install

  Launch the Control Software typing **ZLEControlSoftware**

The default configuration file location is:

/etc/ZLEControlSoftware/ZLEControlSoftwareConfig.txt

Note: Administrator rights are required to modify the *configuration file* of the DPP-ZLE*plus* Control Software under the "/etc/ZLEControlSoftware" folder.

Alternatively the user can modify the ZLEControlSoftwareConfig.txt file that is under the path:

~/ZLEControlSoftware-1.0.0/setup/Linux/ZLEControlSoftwareConfig.txt

and launch the control software typing **ZLEControlSoftware** and the **path** of the config file.

On-line Commands

Once started (following the instructions in Chapter **4**), the DPP-ZLE*plus* Control Software executes the settings written in the configuration file; if a formal error occurs, it is displayed on the shell.

The software accepts some on-line commands associated with particular keys that are described in **Tab. 3.3** below.

Key	Function
Space	Displays the online command help
q	Quits the DPP-ZLE*plus* Control Software
s	Starts / Stops the acquisition
t	Sends a software trigger (single shot): this command forces the acquisition of one event. In analogy with an oscilloscope, this command corresponds to the “Force trigger” button
T	Enables / disables the continuous generation of software trigger: inside the acquisition loop, the program sends a trigger, reads the corresponding event and executes the algorithm analysis. This command corresponds to the "Auto trigger” in an oscilloscope.
p	Single-event plot of the waveforms of each channel enabled for plotting (which does not necessarily coincide with the n channels enabled for acquisition). The plot is single-shot.
P	Enables / disables the continuous plot of the enabled channels**.** When enabled, the input signal trace of channel 0 is plotted by default. Check the “c” command to enable the other channels.
w	Saves to disk a single event, writing a file for each enabled channel, named wave*N*.txt, where *N* is the channel number. **Warning:** the files are overwritten each time. Each file is an ASCII file with has a 4-line header (Record Length, Channel Number, Event Baseline, and Trigger Time Stamp), and 2-columns data, where the first column is the waveform data, and the second is a flag for reconstructed or read samples.
W	Enables / disables the continuous events saving to file. As for the “w” command, it creates one file per channel; then it writes the events consecutively. **Attention:** the file size can grow very quickly**.**
1 .. 3	Enables/Disables the selected trace on the plot; a maximum of three traces can be simultaneously displayed: 1 Input signal 2 Event baseline (The baseline value is frozen for the whole acquisition window, and used for the ZLE_UND_THR and ZLE_UPP_THR crossing) 3 Stored/Not Stored samples. The trace is high when the sample is reconstructed (baseline), and it is low when the sample is read from input.
c	Selects the channel to be shown; after “c” type the desired number of channel. The ZLE Control Software manages the plot of only one channel at a time.
0 .. 7	After the *c* command, selects the channel *n* to be plotted (*n* =[0:3] for DT5751 and N6751, or [0:7] for V1751 and VX1751 digitizers). If the channel is not enabled for acquisition from the configuration file, a message will appear on shell.

Tab. 3.3: DPP-ZLE*plus* Control Software on-line commands

Configuration File Syntax

The configuration file is located into the *"bin"* subfolder of the main *ZLEControlSoftware* folder (see **Fig. 3.1**), and it is divided into two parts: **common settings**, indicated in the `[COMMON]` section, and **individual settings** for individual channels indicated in the `[n]` section, where n is the number of channels. The common settings are set equal to all channels, as the individual settings can be set different for different channels. Please, note that some of the settings of the common section can be overwritten by the corresponding settings in the individual section.

The individual setting can be made also in the common settings part: in this case, they are applied to all channels.

Note: The special commands `@ON` and `@OFF` allow to skip entire blocks of lines: indeed the DPP-ZLE*plus* Control Software can ignore all the configurations from the `@OFF` command to the `@ON`.

Common Settings

OPEN `LinkType LinkNumber NodeNumber BaseAddress`

Specifies the path of the physical channel to open communication with the digitizer:

`LinkType`	Identifies the type of communication channel, choosing between **`USB`** and **`PCI`**. USB corresponds to both the direct connection from PC to digitizer (Desktop models or NIM), and the connection through V1718 and VME bus (VME models). PCI corresponds to both the direct connection from PC A2818 (PCI controller) or A3818 (PCIe controller) to the digitizer through optical fibre (all models), and connection through V2718 and VME bus (VME models).
`LinkNumber`	The number of the connection. Typically is **0** (only one digitizer connection to the PC). In case of more digitizers connected it is necessary to specify which has to be accessed. Remember that the DPP-ZLE*plus* Control Software demo can handle only one digitizer at a time. LinkNumber identifies which USB or A2818/A3818 is in use. Be aware that it is not known in advance which LinkNumber corresponds to which USB port or PCI slot.
`NodeNumber`	This parameter must be specified only when connected via optical link (PCI) and indicates the node in the daisy chain. Typically is **0** (only one digitizer in the optical chain), it may be different if more than one digitizer (or V2718) is connected in a daisy chain.
`BaseAddress`	Indicates the Base Address (32-bit hexadecimal number) to access the digitizer via the VME bus. This number should be 0 for the direct connections from PC to digitizer.

GNUPLOT_PATH `"path"`

Path for the *gnuplot* executable file; for Windows installation, it is normally `".\"` since *gnuplot* is copied into the working directory.

EXTERNAL_TRIGGER `option`

This command manages how the External Trigger is used; `option` can be:

`ACQUISITION_ONLY`: the arrival of a trigger on the front panel causes the acquisition of one event in all the channels of the board.

`ACQUISITION_AND_TRGOUT`: the same as `ACQUISITION_ONLY`. In addition, the external trigger is also propagated to the TRG-OUT (or GPO for the Desktop and NIM versions) front panel connector.

`DISABLED`: the external trigger is ignored.

Note: the only option managed by the DPP-ZLE*plus* firmware is the `ACQUISITION_ONLY` option.

MAX_NUM_EVENTS_BLT `Ne`

It indicates the maximum number of events `Ne` that can be transferred in a block transfer. Higher values of `Ne` may lead to a more efficient usage of the readout bandwidth, requiring more memory allocation for the block transfer.

`Ne` is an integer value ranging from `0` to `600`.

Note: `Ne` = 200 is a recommended value for the DPP-ZLE*plus* firmware.

FPIO_LEVEL `option`

Indicates the electrical level for the front panel LEMO I/Os (TRG_IN, TRG_OUT and S_IN for VME; TRG_IN, GPI and GPO for Desktop and NIM). `option` can be:

`TTL` if the desired I/O level is TTL,

`NIM` if the desired I/O level is NIM.

WRITE_REGISTER `address data`

This command allows to write register values on the board.

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

`data` is the data to be written into the register (16 or 32 bit value);

Note: all the direct write accesses are executed **before** the other settings, thus be aware that the consecutive settings may overwrite the register content.

RECORD_LENGTH `Ns`

Indicates the number `Ns` of samples (where 1 sample corresponds to 1 ns) to be acquired for each trigger (acquisition window). **Note**: due to constraints on the granularity of this setting, it is possible that the real number of acquired samples is approximated to a value close to what set.

`Ns` is an integer value ranging from `1` to `1048575 * 8` (see also the **RECORD_LENGTH** register)

PRE_TRIGGER `value`

Defines the number of samples to be acquired in the acquisition window before the trigger (NS_{PRE}).

`value` is an integer value ranging from `0` to `1023`. The desired value of pre trigger is multiplied by 8: NS_{PRE} = `value * 8` (see also the **PRE_TRIG** register)

TEST_PATTERN `option`

Data from the ADC can be replaced by an internal test pattern, that is a triangular wave ranging from 0 to full scale. `option` can be:

`YES` to enable the TEST_PATTERN;

`NO` to disable it.

The following settings are located in the **common settings** of the configuration file by default, but they can also be applied channel by channel by writing them in the individual settings.

ZLE_NSAMP_BACK `value`

Specifies the number of samples in the Look Back Window (N_{LBK}) corresponding to the number of samples looked back from the threshold crossing.

`value` is an integer number ranging from `2` to `1023`. The desired value of ZLE_NSAMP_BACK is multiplied by 8: N_{LBK} = `value * 8` (see also the **NLBW_SAMPLES** register).

ZLE_NSAMP_AHEAD `value`

Specifies the number of samples in the Look Ahead Window (N_{LAW}), that corresponds to the number of samples looked forward from the threshold crossing.

`value` is an integer number ranging from `0` to `1023`. The desired value of ZLE_NSAMP_AHEAD is multiplied by `8`: N_{LAH} = `value * 8`. (see also the **NLAW_SAMPLES** register)

ZLE_UPP_THRESHOLD `value`

Sets the ZLE*plus* high threshold.

`value` is the high threshold value referred to the input baseline when SEL_NSBL ≠ 0;

`value` is the absolute high threshold value when SEL_NSBL = 0;

`value` (LSB units) is an integer number ranging from `0` to `1023` (see also the **HIGH_THRESHOLD** register)

ZLE_UND_THRESHOLD `value`

Sets the ZLE*plus* low threshold.

`value` is the low threshold value referred to the input baseline when SEL_NSBL ≠ 0;

`value` is the absolute low threshold value when SEL_NSBL = 0;

`value` (LSB units) is an integer number ranging from `0` to `1023` (see also the **LOW_THRESHOLD** register)

SEL_NSBL `option`

Indicates the number of samples that can be used for the baseline mean calculation. `option` can be:

`0`: 0 samples (the baseline is not calculated, and ZLE_UND_THRESHOLD and ZLE_UPP_THRESHOLD become absolute values)

`1`: 8 samples

`2`: 16 samples

`3`: 32 samples

`4`: 64 samples

`5`: 128 samples

`6`: 256 samples

`7`: 512 samples

BSL_THRESHOLD `value`

Sets the threshold level (in LSB) for the baseline acceptance bandwidth.

`value` is an integer number ranging from `1` to `127` (see also the **BASELINE_THRESHOLD (blThr)** register)

BSL_TIMEOUT `value`

Specifies the value for the baseline timeout, corresponding to the number of samples outside the acceptance bandwidth, after that the algorithm restarts the baseline calculation.

`value` is an integer number ranging from `1` to `255` (see also the **BASELINE_TIMEOUT (blTo)** register).

Individual Settings

The following settings are usually individually applied on each channel; however the user can put them also in the [COMMON] section to apply them to all channels.

Every parameter not present in the Individual Settings section is intended to assume the value defined in the Common Settings section.

ENABLE_INPUT `option`

This command enables or disables the corresponding channel for the acquisition. `option` can be

`YES` to enable it;

`NO` to disable it.

DC_OFFSET `value`

The DC_OFFSET allows to shift the input dynamics (-FSR / 2 to +FSR / 2, where FSR is the full scale range (1 Vpp for x751 series)) towards negative or positive values.

`value` ranges from `-50` to `50`, where `-50` corresponds to a signal dynamics from -FSR to 0 (completely negative signal), and `50` corresponds to a signal dynamics from 0 to FSR (completely positive signal). The default value is `0`, corresponding to the signal dynamics of -FSR / 2 to +FSR / 2 (bipolar signal).

Each list of individual parameters set for the same channel has to be reported after the [i] keyword, where "i" is the number of the selected channel:

```
[0]
ENABLE_INPUT            YES         # setting 1 of channel "0" section
DC_OFFSET               10          # setting 2 of channel "0" section
[1]
ENABLE_INPUT            NO          # setting 1 of channel "1" section
DC_OFFSET               0           # setting 1 of channel "1" section
...
```

4 Getting Started

Scope of the chapter

This chapter is intended to provide a quick guide of the DPP-ZLE*plus* Control Software, in order to manage the first practical use of a 751 series running the DPP-ZLE*plus* firmware. The example below can be a starting point to understand the ZLE*plus* functionalities and to customize the readout code.

System Overview

For our setup we make use of the following devices and tools:

- DT5751, 4 Channels 10 bit 1 GS/s Digitizer.
- DPP-ZLE*plus* firmware release 4.2_133.03.
- DPP-ZLE*plus* Control Software rel. 1.0.

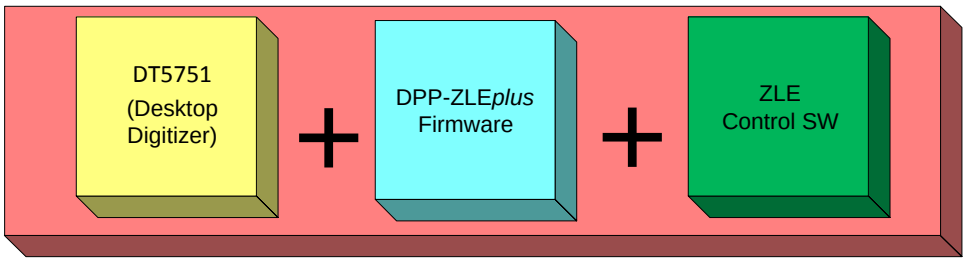


Fig. 4.1: DPP-ZLE*plus* System components

Hardware Setup

An external trigger is sent to the TRG-IN front panel connector of a DT5751 with DPP-ZLE*plus* firmware installed. A signal source is sent to channel 0 of the digitizer. A computer equipped with a Microsoft Windows 7 Professional 64-bit OS acts as host station. The digitizer is connected via USB to the PC.

Fig. 4.2: The hardware setup for the DPP-ZLE*plus* test.

The external trigger provided to the TRG IN front panel input of the digitizer is a TTL signal coming from a signal generator. We set its frequency equal to 5 kHz.

We used two input signal shapes: the first is a typical exponential shape from a detector, the second is a more complex shape with positive and negative values.

Drivers and Software

In order to manage the DPP-ZLE*plus* system, the host station needs either Windows or Linux OS. 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 getting started example 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 strongly recommended to install the driver before connecting the hardware.

Note: Detailed installation steps of CAEN USB drivers for communicating with desktop digitizers are described for several Microsoft Windows OSs in **[RD2]**.

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 DT5751 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**" may be displayed), 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 of the DT5751 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-ZLE*plus* Control Software** for Windows OS.

  Download the standalone **DPP-ZLE*plus* Control Software 1.0** full installation package on CAEN website in the 'Download' area at the DPP-ZLE*plus* Control Software page (**login is required before the download**).

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

  The default *ZLEControlSoftware* installation path is: C:\Program Files\CAEN\Digitizers\ZLEControlSoftware\ , where the "*bin*" subfolder contains the executable file (CAENZLEControlSoftwareSetup-1.0.exe) and the configuration file (ZLEControlSoftwareConfig.txt)

Note: Administrator rights are required to modify the configuration file of the DPP-ZLE*plus* Control Software under the "Program Files" folder. To modify the file and use the software without the administrator rights, copy the entire "bin" folder under another location, as for example the "Documents" folder.

Note: all required libraries are automatically installed by the setup itself (Windows only).

- **DPP-ZLE*plus* Control Software** for Linux.

  Download the ZLEControlSoftware-1.0.tar.gz package on CAEN website in the 'Download' area at the DPP-ZLE*plus* Control Software page (**login is required before the download**).

  Unpack the **installation package** (tar –zxf ZLEControlSoftware-1.0.tar.gz).

  Follow the instruction on the **INSTALL** file

  Type: ./configure

  make

  sudo make install

  Launch the Control Software typing **ZLEControlSoftware**

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.

The default configuration file location is: /etc/ZLEControlSoftware/ZLEControlSoftwareConfig.txt

Note: Administrator rights are required to modify the *configuration file* of the DPP-ZLE*plus* Control Software under the "/etc/ZLEControlSoftware" folder.

Alternatively the user can modify the ZLEControlSoftwareConfig.txt file that is under the path:

~/ZLEControlSoftware-1.0.0/setup/Linux/ZLEControlSoftwareConfig.txt

and launch the control software typing **ZLEControlSoftware** and the **path** of the config file.

Firmware upload

The DPP-ZLE*plus* Control Software works with the **DPP-ZLE*plus* Firmware**.

✓ **How to install the firmware**

Download the **DPP-ZLE*plus* Firmware** (.cfa) for 751 series on CAEN website in the 'Download' area at the DPP-ZLE*plus* page.

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**.

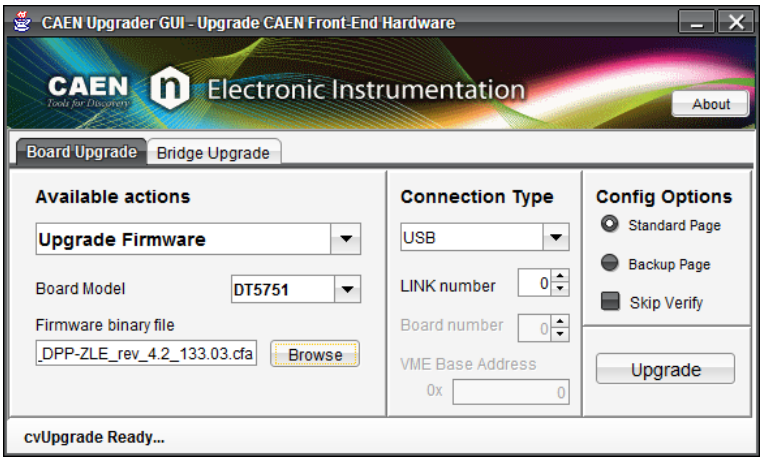


Fig. 4.3: CAENUpgrader settings for DPP-ZLE*plus* firmware upgrade.

Practical Use

This paragraph describes how to use the DPP-ZLE*plus* Control Software in few steps for setting the DPP-ZLE*plus* parameters, for displaying the waveforms and for saving the data.

1. **Set the Configuration file**

The first step is to **set the parameters for the connection, the acquisition and the DPP-ZLE*plus* algorithm.**

Check the **ZLEControlSoftwareConfig.txt** file in the *bin* subfolder.

Note: Administrator rights are required to modify the configuration file of the DPP-ZLE*plus* Control Software under the "Program Files" folder. To modify the file and use the software without the administrator rights, copy the entire "bin" folder under another location, as for example the "Documents" folder.

Note: For those parameters not explicitly reported in this section, the default values are applied.

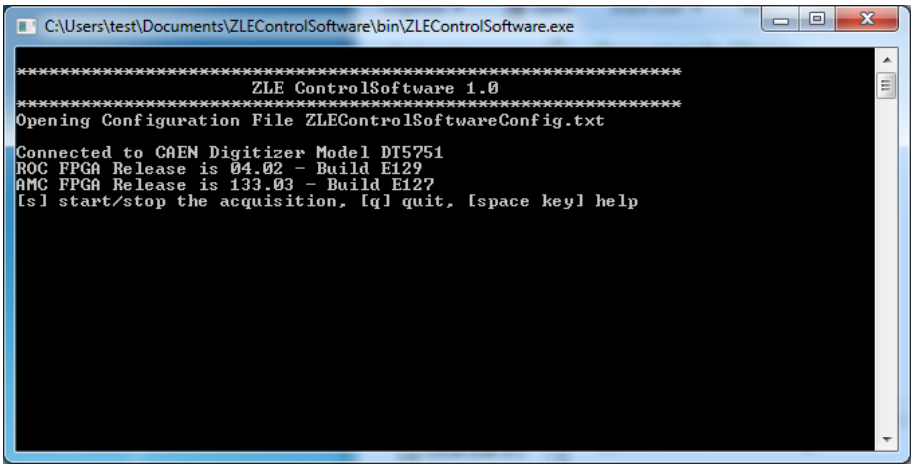
```
# ------------------------------------------------------------------
# Settings common to all channels
# ------------------------------------------------------------------
[COMMON]
#OPEN: open the digitizer
OPEN USB 0 0 0

#FPIO_LEVEL: type of the front panel I/O LEMO connectors
FPIO_LEVEL    TTL

#RECORD_LENGTH
RECORD_LENGTH       20000

#PRE_TRIGGER
PRE_TRIGGER         20

#ZLE_NSAMP_BACK
ZLE_NSAMP_BACK      10

#ZLE_NSAMP_AHEAD
ZLE_NSAMP_AHEAD     50

#ZLE_UPP_THRESHOLD
ZLE_UPP_THRESHOLD    30

#ZLE_UND_THRESHOLD
ZLE_UND_THRESHOLD    100

#SEL_NSBL
SEL_NSBL             3
```

```
#BSL_THRESHOLD
BSL_THRESHOLD        4

#BSL_TIMEOUT
BSL_TIMEOUT          100

# ------------------------------------------------------------
# Individual Settings
# ------------------------------------------------------------
#Only channel 2 is enabled; for the other channels: ENABLE_INPUT  NO.
[2]
ENABLE_INPUT           YES
DC_OFFSET              0
```

Save the configuration file and get ready to start the acquisition.

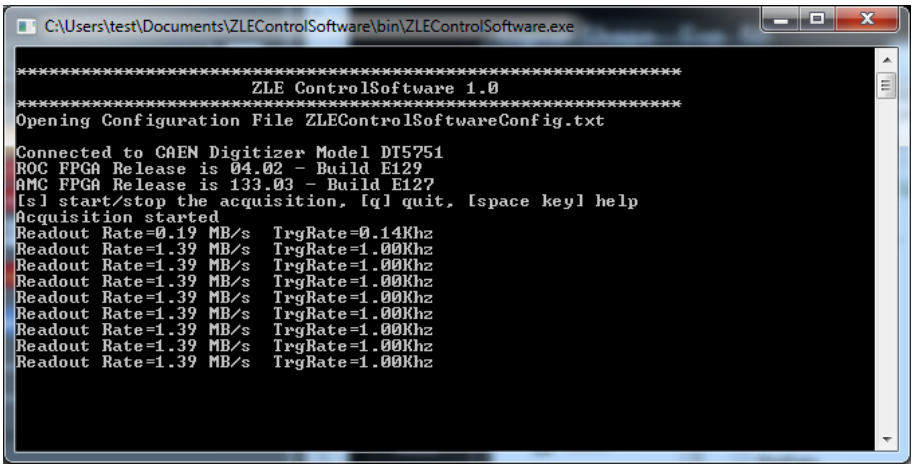
2. Start the acquisition

Launch the program by the file **ZLEControlSoftware.exe** in the *bin* subfolder.



The program connects to the digitizer and displays the FW release information.

Press 's' key to start the acquisition.



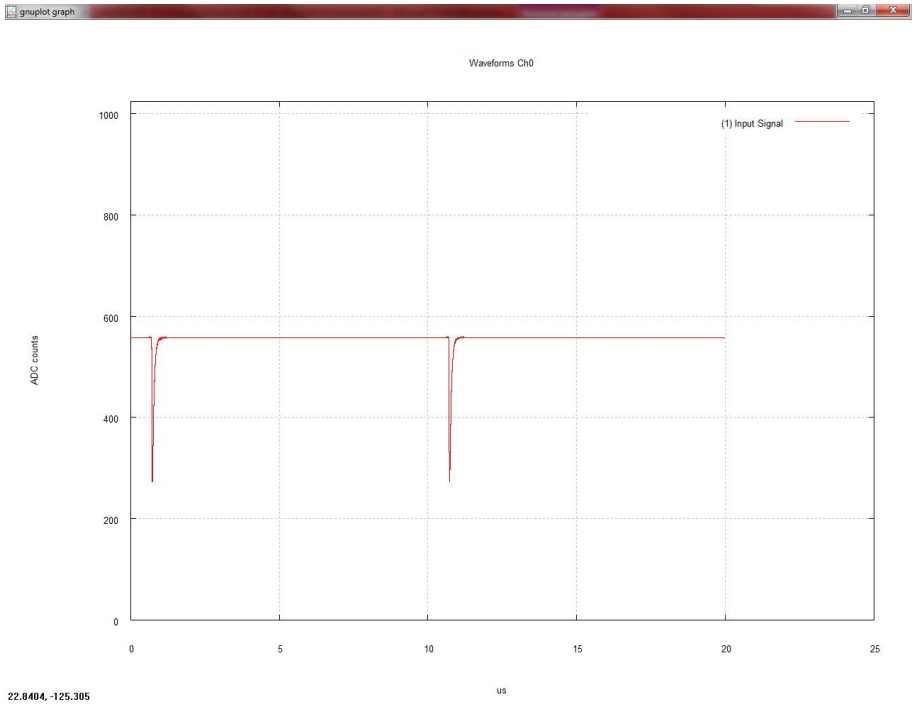
The program will display the Readout Rate (MB/s) and Trigger Rate (KHz) information in real time.

Note: If the settings are not correct the board is not able to trigger. In that case you can enable the "Software Trigger" feature, typing **'t'** for a single software trigger shot, or **'T'** for the continuous trigger.

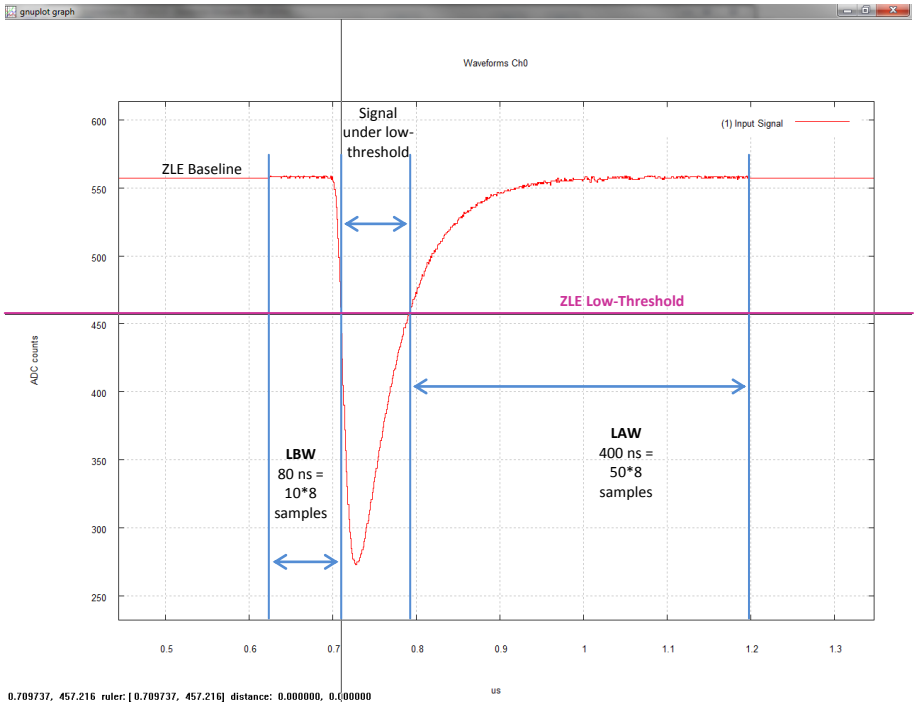
3. Select the channel and plot the waveforms

The **default channel displayed is channel 0**. To enable other channels display press **'c'** and type the corresponding **channel number**.

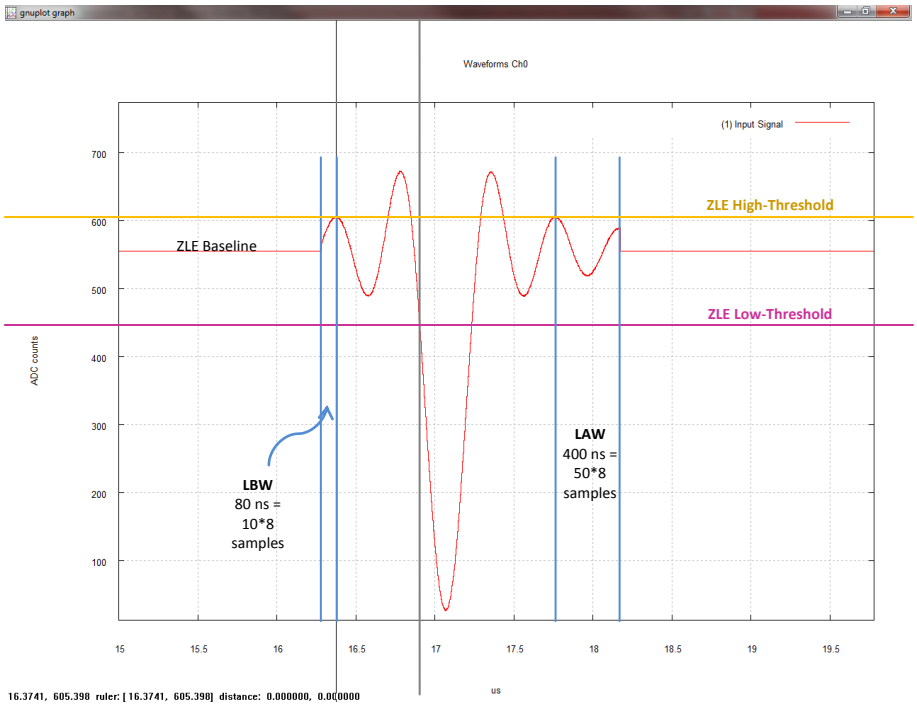
Type **'P'** to **plot** the input waveform **continuously**. 'p' for a single shot plot.



Check that the waveform corresponds to what set in the algorithm.



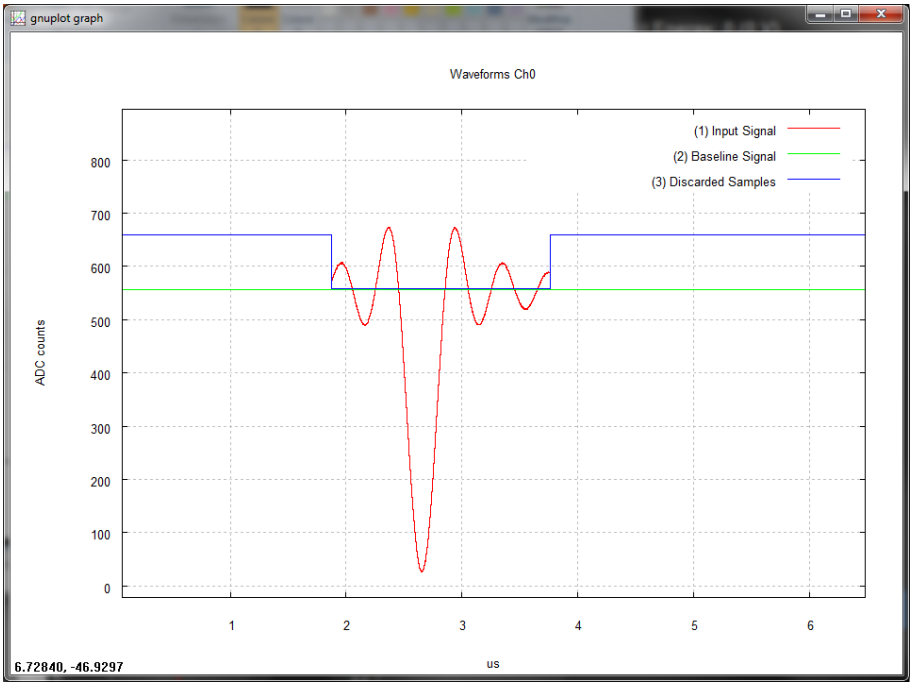
You can use a more complicate input signal to exploit both the high and low thresholds.


gnuplot graph
Waveforms Ch0
(1) Input Signal
ZLE High-Threshold
ZLE Baseline
ZLE Low-Threshold
ADC counts
LBW
80 ns =
10*8
samples
LAW
400 ns =
50*8
samples
700
600
500
400
300
200
100
15
15.5
16
16.5
17
17.5
18
18.5
19
19.5
us
16.3741, 605.398 ruler: [16.3741, 605.398] distance: 0.000000, 0.000000

4. Add the digital traces to the waveform plot

The demo software allows to add two digital traces, the reconstructed “Baseline”, and the “Discarded Samples”. The “Baseline” corresponds to the ZLE reconstructed baseline, while the “Discarded Samples” is high when the samples are calculated (i.e. in correspondence with the baseline), and it is low when the samples are really read from input.

Press **‘2’** to enable/disable the “Baseline” visualization, and press **‘3’** to enable/disable the “Discarded Samples” trace.



5. Save data

Press **‘w’** to write a single event, or **‘W’** to enable the continuous writing. Those commands create one file for each enabled channel. The output file is called wave*N*.txt, where *N* is the channel number.

The output file format is the following:

- Four lines header:
  - Record Length;
  - Channel Number;
  - Event Baseline;
  - Trigger Time Tag.
- Two columns data:
  - First column = data sample;
  - Second column = 1 if the sample is reconstructed (i.e. baseline), 0 if the sample is read from input.

Note: The output files are overwritten every time the commands ‘w’ or ‘W’ are pressed.

Note: Enabling the continuous writing the file size can grow very quickly.

5 ZLE Plus Registers

This chapter describes the accessible registers related to the DPP-ZLE*plus* firmware and those registers that gain or lose some functionality with respect to the standard review of the firmware.

Please, refer to the User Manual of the specific digitizer in the 751 family for the description of all the non-reported registers.

Register map

Register name	Address	Mode
CONFIG	0x8000, 0x8004 (BitSet), 0x8008 (BitClear)	R/W
RECORD_LENGTH	0x1n20 (channel n), 0x8020 (all channels)	R/W
BASELINE_AVERAGE_SEL	0x1n34 (channel n), 0x8034 (all channels)	R/W
PRE_TRIG	0x1n38 (channel n), 0x8038 (all channels)	R/W
NLBW_SAMPLES	0x1n54 (channel n), 0x8054 (all channels)	R/W
NLAW_SAMPLES	0x1n58 (channel n), 0x8058 (all channels)	R/W
LOW_THRESHOLD	0x1n5C (channel n), 0x805C (all channels)	R/W
HIGH_THRESHOLD	0x1n60 (channel n), 0x8060 (all channels)	R/W
BASELINE_THRESHOLD (blThr)	0x1n64 (channel n), 0x8064 (all channels)	R/W
BASELINE_TIMEOUT (blTo)	0x1n68 (channel n), 0x8068 (all channels)	R/W

Tab. 5.1: Registers address map

CONFIG

Address: 0x8000, 0x8004 (BitSet), 0x8008 (BitClear)
Bits: [9:0]
Access Mode: Read and Write

Bit(s) Info	Bit(s) Description
[0]	**Must be "0"**
[1]	**Must be "0"**
[2]	**Reserved**
[3] 0x00000010 Default: 0	**Test Mode:** when the test pattern is enabled, the input signal samples are replaced by a sawtooth test signal: **0:** Test mode disabled **1:** Test mode enabled
[4]	**Must be "1"**
[7:5]	**Must be "0"**
[8] Default: 0	**Individual Trigger: must be "0"**
[9]	**Reserved**

Tab. 5.2: CONFIG

RECORD_LENGTH

Address: 0x8020
Bits: [19:0]
Access Mode: Read and Write

Bit(s) Info	Bit(s) Description
[19:0] Default: 0	**Cust_Size:** this register allows the user to set the acquisition window width (i.e. the Record Length). The Record Length is Ns = Cust_Size * 8 (for example, if you want a Record Length of 32 samples, the value 4 has to be written in the register) ⚠ **WARNING:** unlike the standard FW, in the ZLE*plus* FW the number of samples is only decided by this register and it cannot be zero, otherwise no sample will be saved
[31:20]	**Reserved**

Tab. 5.3: RECORD _LENGTH

BASELINE_AVERAGE_SEL

Address: 0x1n34 (channel n), 0x8034 (all channels)
Bits: [2:0]
Access Mode: Read and Write

Bit(s) Info	Bit(s) Description
[2:0] Default: 0	**Baseline Average Select:** this parameter defines the number of samples for the baseline calculation: **000:** 0 sample (Baseline is always 0, i.e. not calculated) **001:** 8 samples **010:** 16 samples **011:** 32 samples **100:** 64 samples **101:** 128 samples **110:** 256 samples **111:** 512 samples
[31:3]	**Reserved**

Tab. 5.4: BASELINE_AVERAGE_SEL

PRE_TRIGGER

Address: 0x1n38 (channel n), 0x8038 (all channels)
Bits: [9:0]
Access Mode: Read and Write

Bit(s) Info	Bit(s) Description
[9:0] Default: 0	**Pre-Trigger:** defines the number of samples of the input signal that has to be stored before the trigger: NS_{PRE} = Pre-Trigger * 8. (Max Pre-Trigger = 1023 samples, that is to say 8.184 µs)
[31:10]	**Reserved**

Tab. 5.5: PRE_TRG

NLBW_SAMPLES

Address: 0x1n54 (channel n), 0x8054 (all channels)
Bits: [9:0]
Access Mode: Read and Write

Bit(s) Info	Bit(s) Description
[9:0] Default: 0	**Number Samples Look Back:** this register allows to set the number of samples in the Look Back Window with respect to the threshold crossing: N_{LBK} = NLBW_SAMPLES * 8 (for example, if you want to set 24 samples in the Look Back Window, the value 3 has to be written in the register) ⚠ **WARNING:** This parameter must be greater than 2 because the ZLE algorithm works correctly.
[31:10]	**Reserved**

Tab. 5.6: NLBW_SAMPLES

NLAW_SAMPLES

Address: 0x1n58 (channel n), 0x8058 (all channels)
Bits: [9:0]
Access Mode: Read and Write

Bit(s) Info	Bit(s) Description
[9:0] Default: 0	**Number Samples Look Ahead:** this register allows to set the number of samples in the Look Ahead Window with respect to the threshold crossing: N_{LAH} = NLAW_SAMPLES * 8 (for example, if you want to set 24 samples in the Look Ahead Window, the value 3 has to be written in the register)
[31:10]	**Reserved**

Tab. 5.7: NLAW_SAMPLES

LOW_THRESHOLD

Address: 0x1n5C (channel n), 0x805C (all channels)
Bits: [9:0]
Access Mode: Read and Write

Bit(s) Info	Bit(s) Description
[9:0] Default: 0	**LOW_THR:** sets the threshold under the input signal baseline (expressed in LSB). When the input signal crosses the LOW_THR level falling, the samples storage starts (NLBW samples before); when the input signal crosses the LOW_THR level rising, the samples storage stops (after NLAW samples).
[31:10]	**Reserved**

Tab. 5.8: LOW_THRESHOLD

Note: When the **BASELINE_AVERAGE_SEL**= 0, LOW_THR is an absolute threshold: the storage of samples starts when (input < LOW_THR).
In all other cases, LOW_THR is a relative threshold below the baseline: the storage of samples starts when ((baseline – input) > LOW_THR).

HIGH_THRESHOLD

Address: 0x1n60 (channel n), 0x8060 (all channels)
Bits: [9:0]
Access Mode: Read and Write

Bit(s) Info	Bit(s) Description
[9:0] Default: 0	**HIGH_THR:** sets the threshold over the input signal baseline (expressed in LSB). When the input signal crosses the HIGH_THR level rising, the samples storage starts (NLBW samples before); when the input signal crosses the LOW_THR level falling, the samples storage stops (after NLAW samples).
[31:10]	**Reserved**

Tab. 5.9: HIGH_THRESHOLD

Note: When the **BASELINE_AVERAGE_SEL**= 0, HIGH_THR is an absolute threshold: the storage of samples starts when (input > HIGH_THR).
In all other cases, HIGH_THR is a relative threshold below the baseline: the storage of samples starts when ((input – baseline) > HIGH_THR).

BASELINE_THRESHOLD (blThr)

Address: 0x1n64 (channel n), 0x8064 (all channel)
Bits: [6:0]
Access Mode: Read and Write

Bit(s) Info	Bit(s) Description
[6:0] Default: 0	**blThr:** defines the acceptance band for the baseline calculation. When the input signal goes outside this interval (band), the related samples are not included in the baseline mean calculation. When { abs(baseline – signal) < blTh } the baseline mean is calculated
[31:7]	**Reserved**

Tab. 5.10: BASELINE_THRESHOLD (blThr)

BASELINE_TIMEOUT (blTo)

Address: 0x1n68 (channel n), 0x8068 (all channels)
Bits: [7:0]
Access Mode: Read and Write

Bit(s) Info	Bit(s) Description
[8:0] Default: 0	**blTo**: this parameter defines the maximum time (number of 125 MHz clock cycles) in which the input signal stays outside the acceptance band for the baseline mean calculation. During this time the baseline mean is kept frozen. After that time, the baseline value is initialized (is forced) to the input signal value and the baseline starts to be calculated again but it might take some time (depending on the number of samples for the baseline mean) to recover
[31:8]	**Reserved**

Tab. 5.11: BASELINE_TIMEOUT (blTo)

6 Technical support

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

support.nuclear@caen.it
(for questions about the hardware)

support.computing@caen.it
(for questions about software and libraries)

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 Vetraia, 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

CAEN
Tools for Discovery
Electronic Instrumentation

Guide GD2764 - DPP-ZLEplus User Manual rev. 0 - 17 June 2014
00110-10-DGT29-MUTX