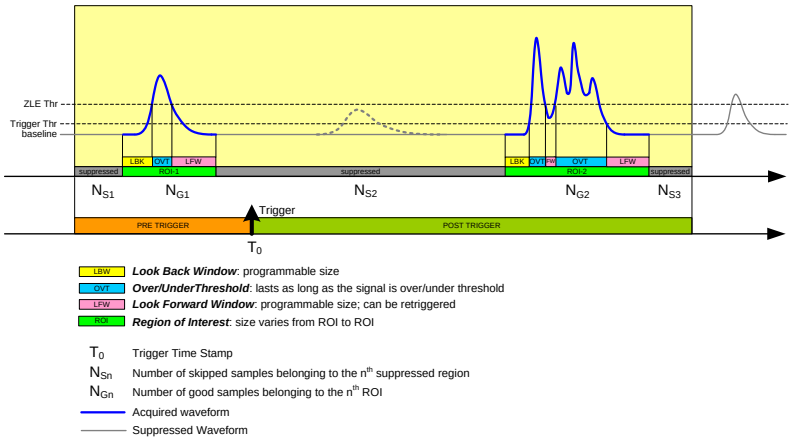
CAEN
Tools for Discovery

Electronic Instrumentation



User Manual UM6064

725-730 DPP-ZLE*plus*

Digital Pulse Processing for Zero Length Encoding

Rev. 2 - July 8th, 2020

Purpose of this Manual

The User Manual contains the full description of the DPP-ZLE*plus* firmware for 725 and 730 series. The description is compliant with the DPP-ZLE*plus* firmware revision **4.17_140.4** for 725 and 730 series. For future release compatibility check in the firmware history files.

Change Document Record

Date	Revision	Changes
July 8th 2019	00	Initial Release
May 15th 2020	01	Added support to x725S and x730S models
July 8th, 2020	02	Modified BLINE_DEFVALUE parameter

Symbols, abbreviated terms and notation

ADC	Analog-to-Digital Converter
AMC	ADC & Memory Controller
DAQ	Data Acquisition
DAC	Digital-to-Analog Converter
DC	Direct Current
DPP	Digital Pulse Processing
DPP-ZLE	DPP for Zero Length Encoding
LVDS	Low-Voltage Differential Signal
ROC	ReadOut Controller
USB	Universal Serial Bus

Reference Documents

[RD1] UM1935 – CAENDigitizer User & Reference Manual.

[RD2] UM6065 – 725-730 DPP-ZLEplus Registers Description.

All CAEN documents can be downloaded at:
`www.caen.it/support-services/documentation-area`

CAEN S.pA.
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

This manual describes the special DPP firmware for Zero Length Encoding Plus (DPP-ZLE*plus* firmware) designed for 725 and 730 digitizer families. The board configuration and the ZLE acquisition can be managed through a demo software, which relies on CAENDigitizer library functions **[RD1]**. Customers can take advantage of the library functions and the C sources of the demo to develop and customize their own acquisition software.

The DPP-ZLE*plus* firmware is able to transfer the significant samples of the waveform, by performing an enhanced zero suppression algorithm on the input signal. The idea is to avoid collecting irrelevant data (zeros), thus reducing the data throughput. The DPP-ZLE*plus* firmware is mainly intended for those experiments where the amount of "relevant" information is quite low with respect to the full data bandwidth, as for example in case of long acquisition windows, where the events are made of small signals with high relative time distance.

The DPP-ZLE*plus* firmware identifies and saves the samples whose distance from the baseline is larger than a programmable threshold, while discarding samples not fulfilling this requirement. The acquisition window, triggered by a global trigger, is completely scanned, but only the relevant data is saved. The event data size can be therefore quite small compared with the data produced by the ADCs in the same window.

2 Principle of Operation

The Digital Pulse Processing for Zero Length Encoding Plus (DPP-ZLE*plus*) is a special firmware developed for 725 and 730 CAEN digitizer families.

A global trigger (see register 0x810C **[RD2]**) is required to start an acquisition window lasting for the duration of the Record Length value (see register 0x8020). A programmable number of samples can be also acquired before the trigger through the Pre Trigger value (register 0x1n38). The acquisition window is common for all enabled channels, while the ZLE algorithm is applied independently to each channel.

The algorithm is able to calculate the signal 0-Volt level (baseline) and to verify which sample is under/over a programmable trigger threshold level (register 0x1n60), according to the trigger polarity (bit[11] of register 0x1n64). The threshold level is relative to the baseline; depending on the polarity, a trigger is issued when the samples are either below (negative polarity) or above (positive polarity) the baseline value by the threshold value.
The user can define a fixed baseline value, which remains constant for the whole acquisition, or let the board calculate it dynamically (see register 0x1n34). In the second case it is required to define the level of the baseline noise in ADC counts. The baseline is then recursively evaluated as the mean value over 16 samples. When the difference between consecutive baseline estimations is within the noise for at least four iterations, a new value for the baseline is acquired.

The relevant samples are identified when the following condition is met:

$$(S_i - BSL) > TRG\_Thr, \text{ for positive polarity}$$

$$(BSL - S_i) < TRG\_Thr, \text{ for negative polarity}$$

where S_i is the i^{th} sample, BSL is the baseline value, and TRG_Thr is the threshold value.

The algorithm further checks whether the ZLE threshold (register 0x1n5C) is crossed and saves the over-threshold samples (under threshold in case of negative polarity - register 0x1n64), which are referred as ”good” samples. In addition, also a programmable number of samples can be acquired before and after the over/under threshold (N_LBK and N_LFW respectively, which can be defined through registers 0x1n54 and 0x1n58). ”Skipped” samples are discarded; only a word reporting the number of skipped samples is present in the final data.

Fig. **2.1** shows the parameters of the ZLE algorithm. The signal recording starts N_LBK samples before the ZLE threshold crossing (LBW in the picture) and stops N_LFW samples after the crossing in the opposite direction (LFW in the picture). The recorded regions define the ROI (region of interest) of the algorithm. Outside the ROI the algorithm reports the number of skipped samples. The demo software reports the baseline level in the first five samples of the plot.

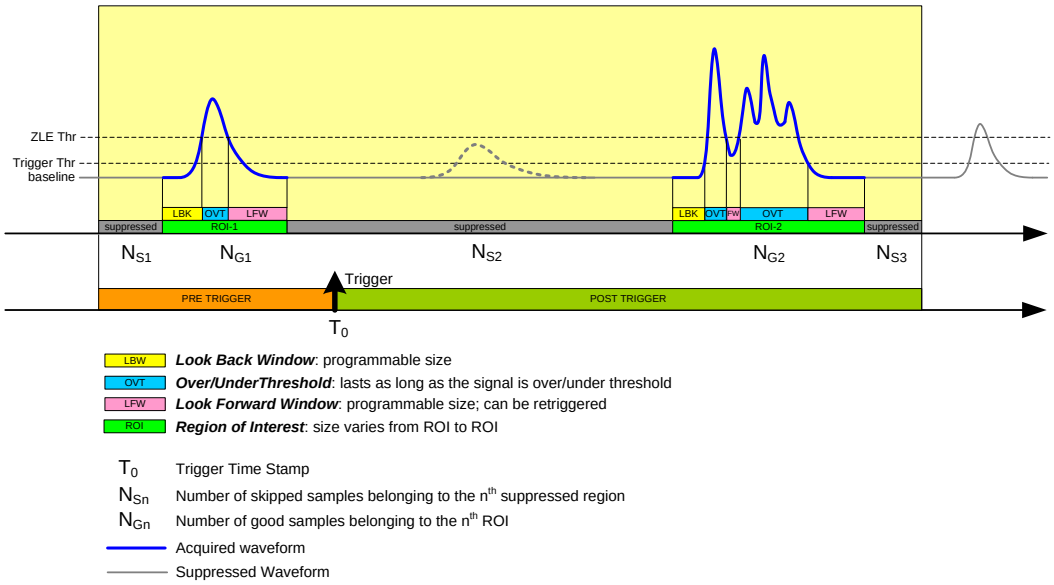


Fig. 2.1: ZLE algorithm description

Event Structure

An event is structured as:

- **Header** (four 32-bit words)
- **Data** (variable size and format)

Header

The event Header consists of four words including the following information:

- **EVENT SIZE** (bits[27:0] of 1^{st} header word) is the total size of the event, i.e. the number of 32-bit long words to be read.
- **BOARD ID** (bits[31:27] of 2^{nd} header word) is the GEO address, meaningful for VME64X modules.
- **BOARD FAIL FLAG** (bit[26] of 2^{nd} header word) is set to "1" in consequence of a hardware problem (e.g. PLL unlocking). The user can collect more information about the cause by reading register 0x8104 and contacting CAEN Support Service if necessary (see Chap. **Technical Support**).
- **PATTERN/TRG OPTIONS** (bits[23:8] of 2^{nd} header word) where PATTERN is the 16-bit PATTERN latched on the LVDS I/Os as the trigger arrives (VME only) and TRG OPTIONS are the trigger information according to bits[22:21] of register 0x811C (see Tab **2.1**).
- **CHANNEL MASK[7:0]** (and **CHANNEL MASK[15:8]** in case of VME) (bits[7:0] of 2^{nd} header word (and bits[31:24] of the 3^{rd} header word)) is the mask of the channels participating in the event (e.g. CH5 and CH7 participating → Channel Mask = 0xA). This information must be used by the software to acknowledge from which channel the samples are coming (the first event contains the samples from the channel with the lowest index).
- **EVENT COUNTER** (bits[23:0] of 3^{rd} header word) is the trigger counter; it can count either accepted triggers only or all triggers (bit[3] of register address 0x8100).
- **TRIGGER TIME TAG** (bits[31:0] of 4^{th} header word) is the 31-bit Trigger Time Tag (TTT) information (31 bit counter and 32^{nd} bit as roll-over flag), which is the trigger time reference. If the ETTT option

REGISTER 0x811C Bits[22:21]	FUNCTIONAL DESCRIPTION	PATTERN/TRG OPTIONS INFORMATION (16 bits in the 2nd header word)
00 (default)	PATTERN (VME) Reserved (DT/NIM)	Pattern of the 16 LVDS signals(VME) Must be 0 (DT/NIM).
01	Event Trigger Source	Indicates the trigger source causing the event acquisition: Bits[23:19] = 00000 Bit[18] = Software Trigger Bit[17] = External Trigger Bit[16] = Trigger from LVDS connector (VME only) Bits[15:8] = Channel self-trigger
10	Extended Trigger Time Tag (ETTT)	A 48-bit Trigger Time Tag (ETTT) information is configured, where Bits[23:8] contributes as the 16 most significant bits together to the 32-bit TTT field (4th header word). **Note**: in the ETTT option, the overflow bit is not provided.
11	Not used	If configured, it acts like “00” setting.

Tab. 2.1: Pattern/Trg Options configuration table.

is enabled, then this field becomes the 32 less significant bits of the 48-bit Extended Trigger Time Tag information in addition to the 16 bits (MSB) of the TRG OPTIONS field (2nd event word). Note that, in the ETTT case, the roll-over flag is no more provided. The trigger time tag is reset either at the start of acquisition, or via front panel signal on S-IN or LVDS I/O connectors, and increments with 125 MHz frequency (i.e. every 4 ADC clock cycles in case of 730 and 2 ADC clock cycles in case of 725). The TTT value is read at half of this frequency (i.e. 62.5 MHz) so that the specifications are 16 ns resolution and 17 s range (8 ns $\times$ ($2^{31}-1$)), which can be extended to 625 h (8 ns $\times$ ($2^{48}-1$)) if ETTT is enabled.

Data

The data section for each enabled channels is made by a channel header for each active channel and by the data of the corresponding channel. The channel header is a single 32-bit word reporting the following information:

- **CHx SIZE** (bits[15:0]) represents the number of words to be read for channel x.
- **BASELINE** (bits[29:16]) is the baseline value of the event in channel x.
- **MEM FULL** (bit[31]) flags whether the memory is full and the event is truncated.

The channel header is followed by a sequence of words representing skipped and good samples of the channel itself. A ”skipped word” is a 32 bit word which reports:

- **SKIPPED SAMPLES/2** (bits[27:0]) is the number of couples of skipped samples.
- **bit[31:28]** = 1000 identifies a skipped word.

A ”good word” is a 32 bit word made of two acquired samples of 14 bits each. Bits[31:30] and [15:14] are reserved.

An example of data format is reported in Fig. **2.2**

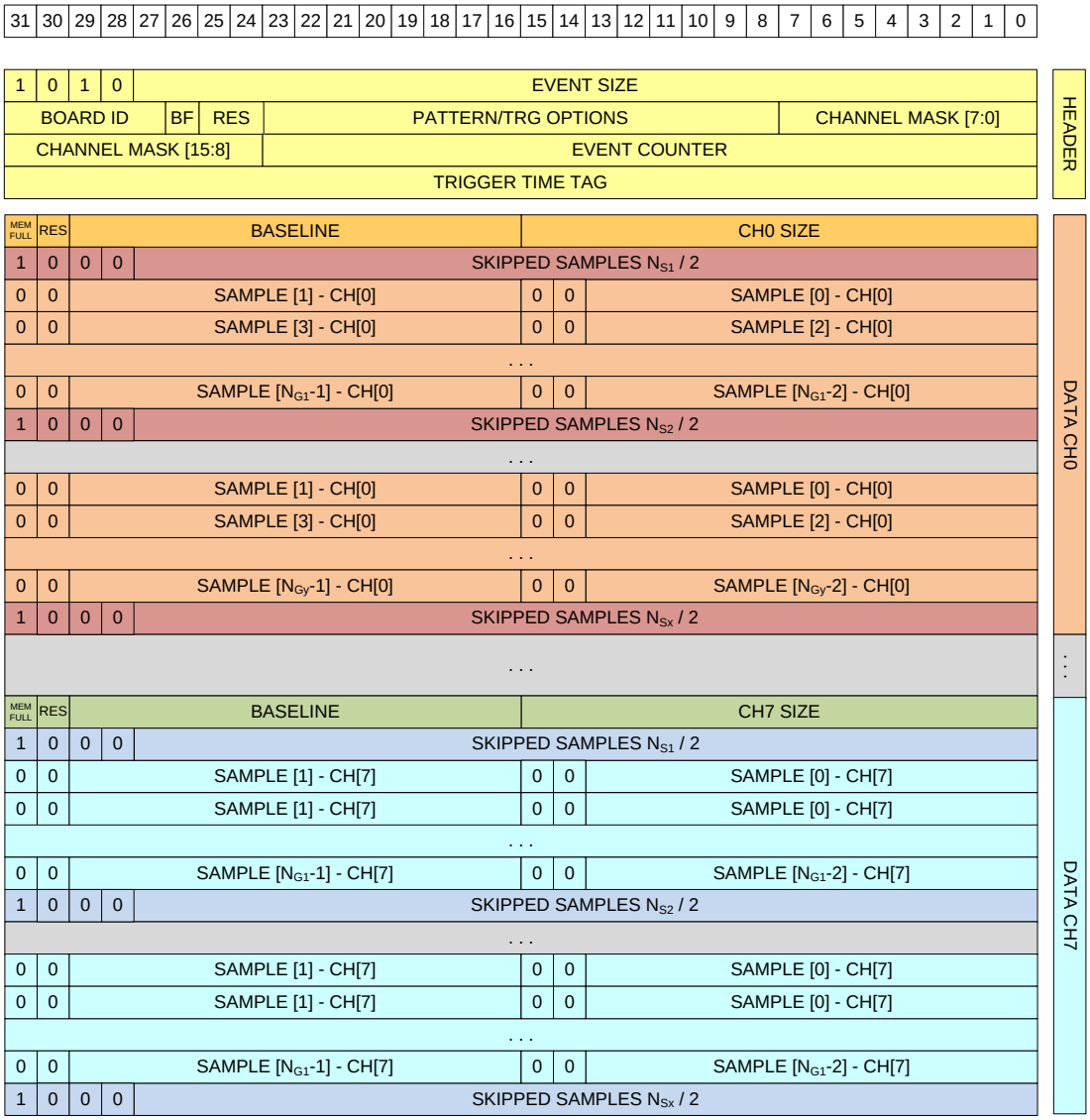


Fig. 2.2: Event Format example

Notes on Firmware and Licensing

The DPP-ZLE*plus* firmware runs on 725-730 digitizer series. The supported digitizer models and the DPP firmware license are listed in the table below.

Desktop Digitizer	Description
DT5725	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
DT5725B	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
DT5730	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
DT5730B	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
DT5725S	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
DT5725SB	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
DT5730S	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
DT5730SB	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
NIM Digitizer	**Description**
N6725	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
N6725B	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
N6730	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
N6730B	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
N6725S	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
N6725SB	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
N6730S	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
N6730SB	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
VME Digitizer	**Description**
V1725	16 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
V1725B	16 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
V1725C	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
V1725D	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
VX1725	16 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
VX1725B	16 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
VX1725C	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
VX1725D	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
V1730	16 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
V1730B	16 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
V1730C	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
V1730D	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
VX1730	16 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
VX1730B	16 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
VX1730C	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
VX1730D	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
V1725S	16 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
V1725SB	16 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
V1725SC	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
V1725SD	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
VX1725S	16 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
VX1725SB	16 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
VX1725SC	8 Ch. 14 bit 250 MS/s Digitizer: 640kS/ch, CE30, SE
VX1725SD	8 Ch. 14 bit 250 MS/s Digitizer: 5.12MS/ch, CE30, SE
V1730S	16 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
V1730SB	16 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
V1730SC	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
V1730SD	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE

VX1730S	16 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
VX1730SB	16 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
VX1730SC	8 Ch. 14 bit 500 MS/s Digitizer: 640kS/ch, CE30, SE
VX1730SD	8 Ch. 14 bit 500 MS/s Digitizer: 5.12MS/ch, CE30, SE
DPP Firmware	**Description**
DPP-ZLE (16ch x725)	DPP-ZLE - Digital Pulse Processing for Zero Length Encoding (16ch x725)
DPP-ZLE (8ch x725)	DPP-ZLE - Digital Pulse Processing for Zero Length Encoding (8ch x725)
DPP-SUP 16ch x725	DPP-SUP - Super Licence for 16ch x725 Digital Pulse Processing
DPP-SUP 8ch x725	DPP-SUP - Super Licence for 8ch x725 Digital Pulse Processing
DPP-ZLE (16ch x730)	DPP-ZLE - Digital Pulse Processing for Zero Length Encoding (16ch x730)
DPP-ZLE (8ch x730)	DPP-ZLE - Digital Pulse Processing for Zero Length Encoding (8ch x730)
DPP-SUP (16ch x730)	DPP-SUP - Super Licence for 16ch x730 Digital Pulse Processing
DPP-SUP (8ch x730)	DPP-SUP - Super Licence for 8ch x730 Digital Pulse Processing

Tab. 2.2: Compliance table of the DPP-ZLE*plus* with CAEN digitizers and DPP firmware.

3 Software Interface

CAEN provides a C demo software for the DPP-ZLE*plus* readout. The user can make the acquisition through this software, or take the source codes as an example to access the CAENDigitizer library functions and develop his/her customized readout program. The package includes the C source files, the Visual Studio project and (for the Linux version) the Linux Makefile.

Installation

In order to be able to install the demo software, the host station needs Windows or Linux OS. CAEN provides the full installation package for the DPP-ZLE*plus*demo software in a standalone version for Windows and Linux OS. This version installs all the binary files and required libraries.

1. Download the DPP-ZLE*plus* demo software for your OS from CAEN Website under the path: `www.caen.it/download/?filter=DPP-ZLEplus`
2. Extract the files

Linux Installation

To install the software type

```
./configure
make
sudo make install
```

Windows Installation

Run the executable “ZLE_swSetup-1.0” and the Setup Wizard will guide you throughout the installation procedure.

Note: The Windows OS must be up-to-date to correctly install the Visual C++ Redistributable (see Figs. **3.6**, **3.7**, and **3.8**).

Note: The following screenshots are taken with Window 10 OS, and they can be generalized for Windows 7 and Windows 8.

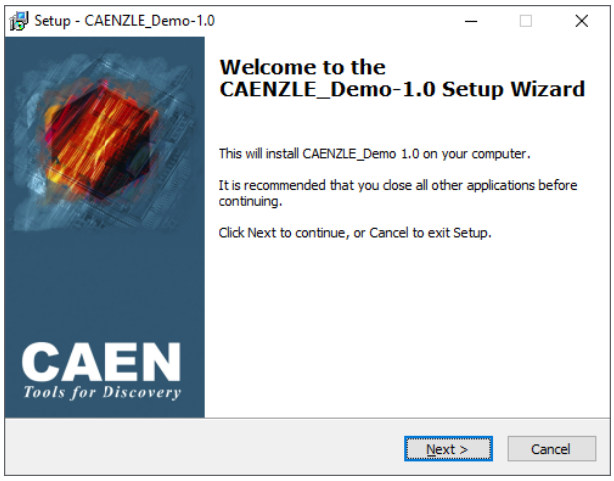


Fig. 3.1: DPP-ZLE*plus* demo Wizard Dialog Box - Start Installation.

Left click on "**Next**" (or left click on "**Cancel**" any time during the installation process to abort the installation).

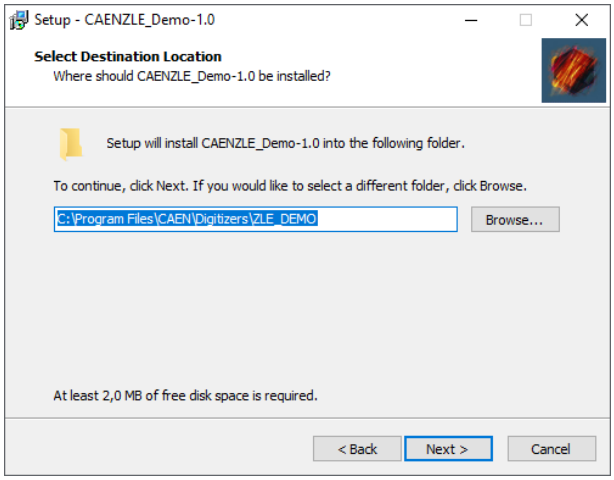


Fig. 3.2: DPP-ZLE*plus* demo Wizard Dialog Box - Installation Path.

Left click on "**Next**" (or left click on "**Back**" at any time during the installation process to modify the previous settings).

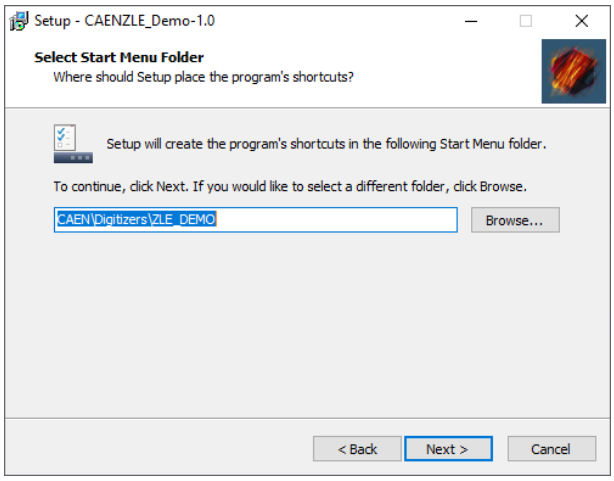


Fig. 3.3: DPP-ZLE*plus* demo Wizard Dialog Box - Start Menu Folder.

Left click on "**Next**" to continue.

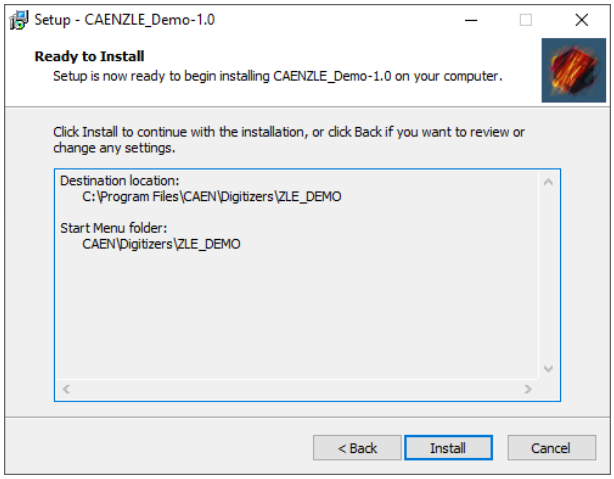


Fig. 3.4: DPP-ZLE*plus* demo Wizard Dialog Box - Installation.

Left click on "**Install**" to install the DPP-ZLE*plus* Software.

The DPP-ZLE*plus* demo Setup Wizard will extract and install the relevant files.

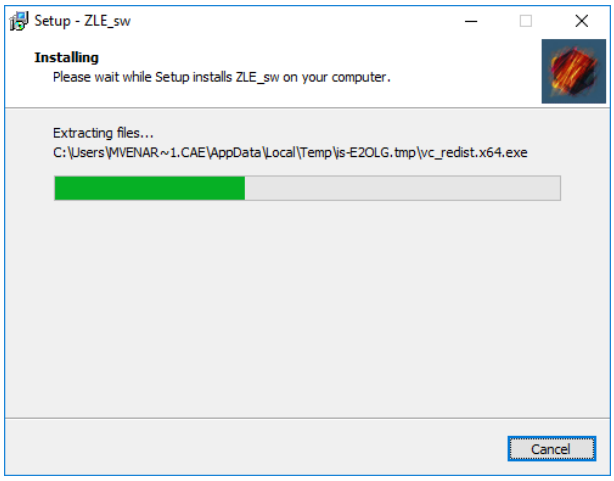


Fig. 3.5: DPP-ZLE*plus* demo Wizard Dialog Box - Installation progress.

The DPP-ZLE*plus* demo Setup Wizard will then install Microsoft Visual C++ 2015 Redistributable as needed requirement.

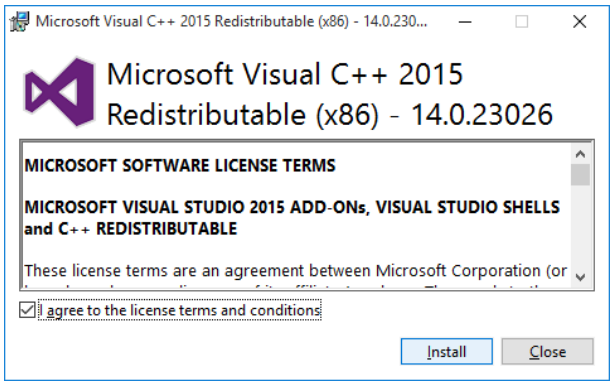


Fig. 3.6: DPP-ZLE*plus* demo Wizard Dialog Box - Microsoft Visual C++ 2015 Redistributable installation start.

If the Wizard detects an already present installation it will abort the procedure. In case no installation is found the Wizard will proceed.

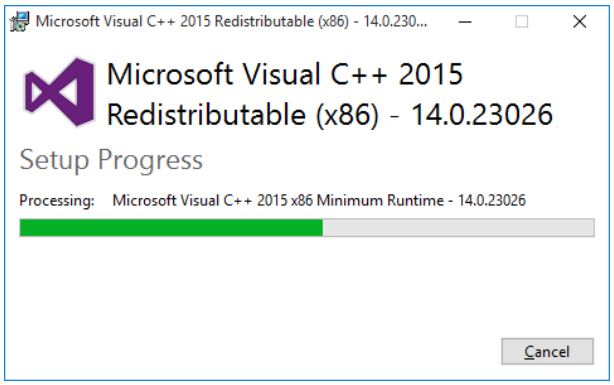


Fig. 3.7: DPP-ZLE*plus* demo Wizard Dialog Box - Microsoft Visual C++ 2015 Redistributable installation ongoing.

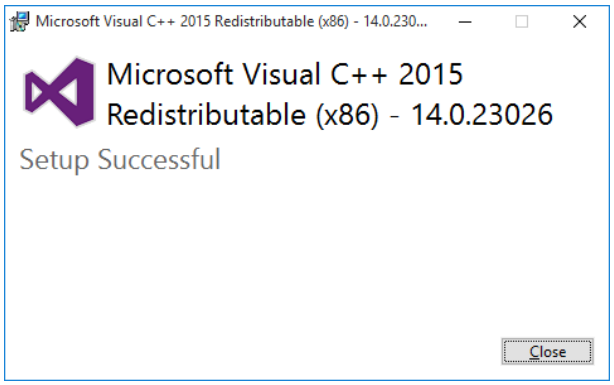


Fig. 3.8: DPP-ZLE*plus* demo Wizard Dialog Box - Microsoft Visual C++ 2015 Redistributable installation completed.

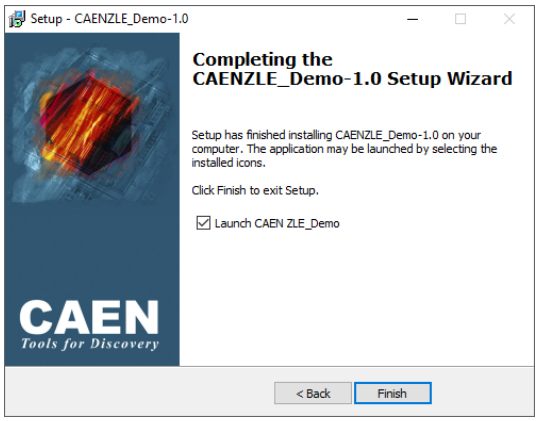


Fig. 3.9: DPP-ZLE*plus* demo Wizard Dialog Box - Finish Installation.

To complete the DPP-ZLE*plus* demo Installation left click on "**Finish**". The DPP-ZLE*plus*demo Program can be launched by left clicking on the installed icons in the Start Menu Folder or from the desktop icon.

725-730 On-line commands

At start-up, the demo software retrieves general information from the digitizer, program it and waits for the start of the acquisition ('s' key). The table below lists the accepted on-line commands associated to specific keys.

Key	Function
s	Start/Stop acquisition.
t	Single Software Trigger; this command sends a software trigger (single shot), useful especially when the card has no data (no trigger) because it forces the acquisition of an event. In analogy with the oscilloscope, this command corresponds to the "Force trigger" button.
p	This command plots a single event.
g	This command select which octet (ch0-ch7 or ch8-ch15) is plotted.
1-8	This command enables / disables channel n (n = 0 .. 7, or n = 8, ..., 15) of the selected octet from the plot.
+/-	In case of multi-board systems, this command plots the next/previous board respectively.
q	This command quits the software.
Space	This command displays the online help.

Tab. 3.1: Software on-line commands.

Inside the Gnuplot window, there are active bindkeys and functions associated to the mouse:

Key	Function
a	x and y-axis auto-scale.
r	Enable / Disable ruler.
g	Enable / Disable grid.
y	Set the scale y at full scale, while x-scale remains unchanged
p	Return to previous zoom.

Tab. 3.2: Gnuplot on-line commands.

Zoom Area: right click on one corner of the area, release the button, left click on the opposite corner.

Click with the right button on the window bar to open a menu that allows to make the print, copy the screen-shot to the clipboard, change colours, etc.

725 and 730 Config File Sintax

The configuration file is located in the main software folder and it is called ”ZLE_Config.txt”.

The setting modes can be [COMMON] (all boards-related), ”[BOARD] #” (single board-related) or ”[CHANNEL] #” (single channel-related).

If a setting is CHANNEL-related, it can also be BOARD-related or COMMON (i.e. the setting refers to all the channels in all the boards of the system). However not all options that are COMMON can also be BOARD- or CHANNEL-related. The ”[BOARD] #” or ”[CHANNEL] #” directives have no effect on such settings. The ”[BOARD] #” and ”[CHANNEL] #” modes are alternative to [COMMON] but do not exclude each other. This means that one can issue a command to a specific channel of a specific board, to all channels of a specific board or to a specific channel of all boards.

When the ”[BOARD] #” mode is set, the following settings apply to all channels (even if a specific channel was previously selected). The board index follows the board initialization order using the OPEN command (see below). The lines between the commands @OFF and @ON will be skipped. This can be used to exclude parts of the config file.

Board 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. It is typically 0 (only one digitizer connection to the PC); in case of many digitizers connected it is necessary to specify which has to be accessed. Remember that the DPP-ZLE*plus* demo software 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. It is typically 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 a digitizer.

Common Settings

START_ACQ `option`

This command specifies the source of the start acquisition and it is relevant only when `SYNC_ENABLE` is set to `NO` (see next parameter). The possible `options` are:

`SW`: a software command starts the acquisition;

`S_IN`: a level on the S_IN/GPI front panel connector drives the acquisition;

`FIRST_TRG`: first trigger on TRG-IN front panel connector starts the acquisition;

`LVDS`: signal on LVDS connectors starts the acquisition;

SYNC_ENABLE `option`

Sets all the registers required for the synchronization procedure described in Chapter **Board Synchronization**. Board delays might need a further tuning according to the experimental setup used.

Note: The start acquisition is SW-controlled for the master board and propagated through LVDS connectors to the slaves. Command `START_ACQ` is not applied.

The possible `options` are:

`YES` to enable it;

`NO` to disable it.

STAT_REFRESH `value`

Statistics Refresh period (msec), including event plotting if enabled.

PERIODIC_PLOT `option`

If enabled, the spectrum will be plotted periodically, the period being set by the `STAT_REFRESH` variable. On slower PCs, disabling this feature might improve the demo software performances. Single events can be plotted by pressing the ”p” key during acquisition. The possible `options` are:

`YES` to enable it;

`NO` to disable it.

PLOT_TYPE `option`

This command allows the user to plot all the channels of a single board (`BOARDPLOT`) or the 0-channel of each board (`SYNCPLOT`, useful to verify the synchronization). The possible `options` are:

`BOARDPLOT` to plot all the channels of a single board;

`SYNCPLOT` to plot the 0-channel of each board.

GNUPLOT_PATH `"path"`

Path of the gnuplot executable file, which is the working directory by default. In case of custom gnuplot installation it is recommended to verify the correct path.

OUTFILE_ENABLE `option`

In case of `OUTFILE_RAW`, it is possible to dump a file of all the acquired waveform in binary format, where the format is defined in Sec. **Event Structure**. In case of `OUTFILE_WAVE`, the program saves a .txt file with the last acquired waveform for each enabled channel. The possible `options` are:

`OUTFILE_RAW YES/NO` to enable/disable the board dump (.bin);

`OUTFILE_WAVE YES/NO` to enable/disable the dump of the last waveform of each active channel (.txt).

OUTFILE_PATH `path`

This command sets the path for the output file write.
The default output directory in Windows is <UserDir>\ZLE_Demo\ZLE_output (created by the program).
The default output directory in Linux is "UserDir"\ZLE_output (created by the program)
The default file name is run0, then *b*, *c*, and *seg* identify the board, channel and output segment respectively (segmented according to OUTFILE_MAXSIZE):

`"file_name"_raw_b#_seg#.bin` for the raw

`"file_name"_wave_b#_c#.txt` for the wave

OUTFILE_MAXSIZE `value`

This command defines the maximum size of a file. After this size has been reached, a new file is created with a _seg(#+1) suffix.

`value` is expressed in MBytes.

SW_TRIGGER `option`

This command sets the software trigger input settings. When enabled, the software trigger can be either propagated (ACQUISITION_AND_TRGOUT, TRGOUT_ONLY) or not (ACQUISITION_ONLY) through the TRGOUT (or GPO for the Desktop and NIM versions) connector

The possible `options` are:

`DISABLED` to disable it;

`ACQUISITION_ONLY` to enable it for the acquisition only;

`ACQUISITION_AND_TRGOUT` to enable it for the acquisition and propagate it through the TRGOUT (GPO) connector;

`TRGOUT_ONLY` to propagate it through the TRGOUT (GPO) connector only.

CONT_SWTRIGGER `option`

This command enables or disables the continuous software trigger. The possible `options` are:

`YES` to enable it;

`NO` to disable it.

EXTERNAL_TRIGGER `option`

This command manages how an external trigger on the TRG IN front panel connector is used. The possible `options` are:

`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.

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). The possible `options` are:

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

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

RECORD_LENGTH `Ns`

Indicates the number of samples (where 1 sample corresponds to 2 ns for 730 and 4 ns for 725 series) to be acquired for each trigger in one acquisition window.
Note: only a maximum of 2032 ”good” samples can be saved despite the record length value. If an event exceeds 2032 samples, it is truncated and flagged.

`Ns` is an integer value ranging from 1 to 1048576. The number of samples of record length corresponds to `Ns` · 4.

TRIGCOUPLE_MASK `value`

This command defines which couples of channels participate to the global trigger generation.

`value` is the bit mask corresponding the enabled couple. For example, to enable the first couple (i.e. channel 0 and channel 1) to participate to the global trigger, `value = 0x1`. To enable all couples (VME form factor), `value = 0xFF`.

TRGOUTCOUPLE_MASK `value`

This command defines which couples of channels participate to the generation of a signal to the TRG OUT (GPO for Desktop/NIM form factors) front panel connector.

`value` is the bit mask corresponding the enabled couple. For example, to enable the first couple (i.e. channel 0 and channel 1) to participate to the TRG OUT (GPO) signal, `value = 0x1`. To enable all couples (VME form factor), `value = 0xFF`.

TRIGLOGIC_COUPLE `couple logic`

This command defines how the channels in the couple participate to the couple trigger logic.

`couple` is the couple index, from 0 to 7;

`logic` is the channels logic. The possible options are: `OR, AND, FIRSTONLY, SECONDONLY`.

GAIN_FACTOR `value`

Sets the input dynamic range of the board. The possible `values` are:

`0x0` = 2 V_{pp};

`0x1` = 0.5 V_{pp}.

PRE_TRIGGER `value`

Defines the number of samples to be acquired in the acquisition window before the trigger.
NOTE: A fixed latency should be added to the pre trigger value, equal to 16 ns in case of trigger generated by over-threshold channels; 186 and 132 ns in case of external trigger from TRG IN connector for 725 and 730 series respectively.

`value` is an integer value ranging from 0 to 240. The corresponding value of pre trigger is multiplied by 4.

BLINE_DEFMODE `option`

This command enables or disables the use of fixed baseline value. The possible `options` are:

`YES` the algorithm uses the default value written in BLINE_DEFVALUE `value`;

`NO` the algorithm automatically evaluates the baseline.

BLINE_DEFVALUE `value`

Sets the fixed value of baseline in case of `BLINE_DEFMODE=YES`.

`value` is an integer value ranging from 0 to 16383.

Note: if, in any of the channels taking part to the trigger generation, the difference between the signal and the set baseline is always larger than the trigger threshold (as set in the TRG_THRESHOLD parameter), that channel, or any other channel, will not be able to generate a trigger.

BLINE_NOISE `value`

This value corresponds to input baseline noise level (sigma) in ADC LSB counts, which is used by the algorithm to dynamically evaluate the baseline.

`value` is an integer value ranging from 0 to 255.

PULSE_POLARITY `value`

Sets the trigger polarity. The possible `values` are:

`POSITIVE`: trigger condition met for over-threshold samples;

`NEGATIVE`: trigger condition met for under-threshold samples.

NO_THRESHOLD `option`

When enabled, the algorithm saves all the samples in the record length without considering the threshold and polarity. The possible `options` are:

YES to enable it;

NO to disable it.

TEST_PULSE option

When enabled, the input channels are replaced by an exponentially decaying pulse internally generated by the FPGA. The possible options are:

YES to enable it;

NO to disable it.

TP_RATE option

Sets the rate of the internal test pulse, when enabled. The possible options are:

0 = 1 Hz for 730, 0.5 Hz for 725;

1 = 10 Hz for 730, 5 Hz for 725;

2 = 100 Hz for 730, 50 Hz for 725;

3 = 1 kHz for 730, 500 Hz for 725;

4 = 10 kHz for 730, 5 kHz for 725;

5 = 100 kHz for 730, 50 kHz for 725;

6 = 1 MHz for 730, 500 kHz for 725;

7 = 10 MHz for 730, 5 MHz for 725.

TP_SCALE option

Sets the amplitude of the internal test pulse when enabled. The possible options are:

0 = peak height of 2000 ADC counts, baseline at mid-scale;

1 = peak height of 4000 ADC counts, baseline at mid-scale;

2 = peak height of 6000 ADC counts, baseline at mid-scale;

3 = peak height of 8000 ADC counts, baseline at mid-scale.

TP_POLARITY value

Sets the polarity of the internal test pulse. The possible values are:

POSITIVE;

NEGATIVE.

Individual Settings

Individual settings are written under the [n] command, where n is the channel index. If they are written in the common settings area, then they are applied to all channels at the same time.

ENABLE_INPUT `option`

This command includes or excludes the corresponding channel in the data throughput. One channel can be present in the data without contributing to the global trigger generation and vice-versa. Refer to the **TRIGCOUPLE_MASK** and **TRIGLOGIC_COUPLE** parameters for the trigger generation. The possible `options` are:

`YES` to include it;

`NO` to exclude it.

ENABLE_GRAPH `option`

This command enables or disables the plot of the waveform for corresponding channel. The plotted channel can be enabled/disabled by using the '1'-'8' keys, which correspond to ch. 0-7 and ch. 8-15. The groups ch.0-7 and ch.8-15 can be toggled by pressing the 'g' key. In case of multiple boards, it is possible to select the plotted board by pressing the '+'/'-' keys. This parameter is meaningless if the channel is not enabled (`ENABLE_INPUT = YES`). The possible `options` are:

`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, 2 V_{pp} or 0.5 V_{pp} for 730 and 725 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).

Note: the dynamic range of the DAC does not correspond exactly to the ADC range, so there is not an exact correspondence between the 0-level of DC offset and the baseline at 8192 ADC channels.

N_LBK `value`

Specifies the number of samples in the Look Back Window (N_LBK) corresponding to the number of samples acquired before the over/under threshold.

`value` is an integer value ranging from 0 to 255. The corresponding value of N_LBK is multiplied by 2.

N_LFW `value`

Specifies the number of samples in the Look Forward Window (N_LFW) corresponding to the number of samples acquired after the over/under threshold.

`value` is an integer value ranging from 0 to 255. The corresponding value of N_LFW is multiplied by 2.

ZLE_THRESHOLD `value`

Sets the ZLE threshold relative to the baseline value.

`value` is an integer value ranging from 0 to 16383.

TRG_THRESHOLD `value`

Sets the trigger threshold relative to the baseline value.

`value` is an integer value ranging from 0 to 16383.

4 Board Synchronization

This chapter details the required steps in order to synchronize several boards are reported. Although the guide refers to the DPP-ZLE*plus* package for V1725/30 boards, the described procedure also fits other board models: in what follows you should refer to the DPP-ZLE*plus* package for what concerns the code and to the V1725/30 manuals for the registers and their description. The pictures refer to two synchronized boards, but the procedure applies to any number of boards. This procedure can be verified in the plot through the option **PLOT_TYPE** `SYNCPLOT`.

1. **CLOCK**: all the boards should work with the same clock, and all the clock of all the boards should ideally be in phase. In order to do that, one board (the clock master, CM) distributes its clock to all the others (clock slaves, CSs) through the CLK IN / CLK OUT connectors in the upper part of the front panel (see Fig. **4.1**). The connections should be made through the A317 connectors provided by CAEN, as shown in the picture below:

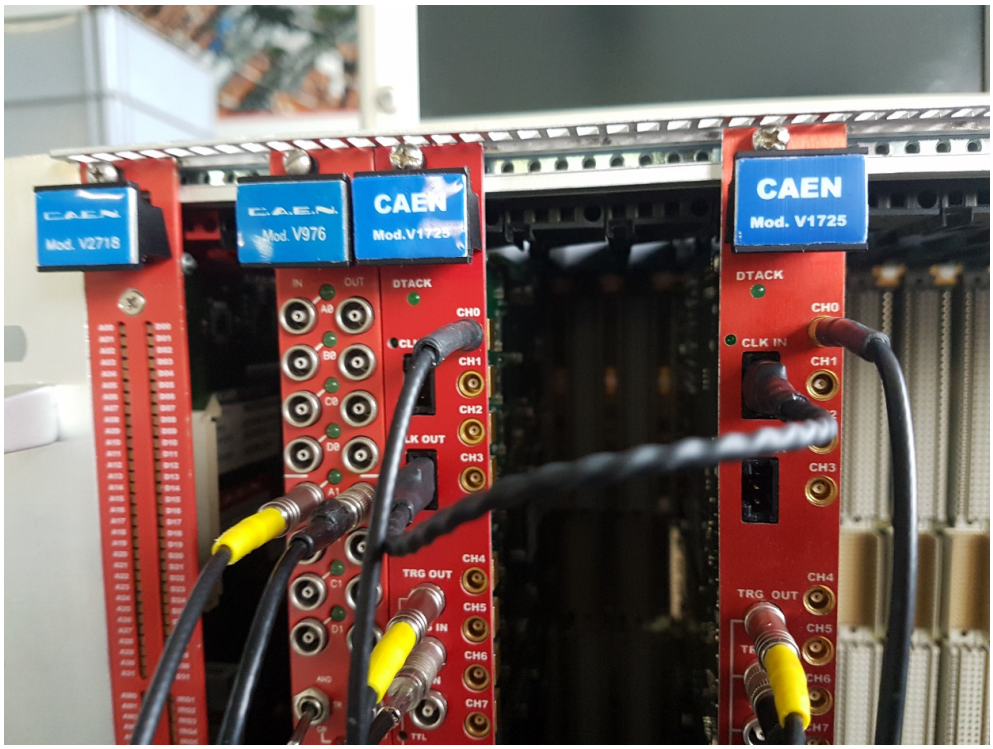


Fig. 4.1: Digitizer clock daisy-chain.

Please remember to set the clock switch on the board PCB (located in the space between the two daughter-boards) to **INT** for the CM only, while it should be set to **EXT** for the CSs.

An update of the internal PLL must also be made. In the ZLE packages the required PLL configuration files are provided: the user should use the 50 MHz internal oscillator reference configuration (PLL-ClockMaster.rbf) for the CM and the 62.5 MHz reference configuration (PLLClockSlave.rbf, the 62.5 MHz clock is received from the previous board in the daisy chain) for the CSs.

In order to upgrade the board PLL the user have to do the following for each board

(a) Open the CAENUpgrader software
(b) Select Upgrade PLL into the ”Board Upgrade” tab
(c) Choose the relevant board from the menu

(d) Select the PLL file to upload
(e) Select the connection type
(f) Press ”Upgrade”

Once this operation has been done for all boards, the user must power-cycle the crate.

The clocks can be checked to be synchronous and in phase by setting the register 0x811c to 0x90000 and observing the clocks on an oscilloscope from the TRG OUT connector. As the phase setting might be slightly dependent from the board and the running conditions, in case of relevant phase differences between two or more clocks please refer to the CAEN Technical Support.

2. **SYNCHRONIZATION PROCEDURE**: The synchronization procedure described in this Chapter assumes a fanned-out trigger in input to the TRG IN connector of each board via equal-length LEMO cables.

   Furthermore, the TRG OUT connector must not be used for triggering purposes, as it is used to stop triggering the boards in case one of them is in a BUSY state (i.e. it will soon reach a situation in which received triggers cannot be processed).

   The relevant registers to be configured for this synchronization procedure can be seen directly in the ProgramDigitizers() function included in the demo: if the `SYNC_ENABLE` demo option is set to `YES`, the demo will configure the board according to the described synchronization procedure. Here we only highlight the most relevant points of the procedure:

   (a) The first board of the daisy chain is configured to start the run via software (in the demo, this corresponds to pushing the ‘s’ key), while the other boards are configured to start via LVDS signals;
   (b) The RUN IN signal is propagated from one board to the other in daisy-chain through the LVDS connectors in the lower part of the board front panel, as shown in Fig. **4.2**. In particular, the LVDS pair 7 of each board is configured as a RUN IN output and should be connected to the LVDS pair 3 of the following board in the daisy chain (configured as a RUN IN input), see Fig. **4.2**;

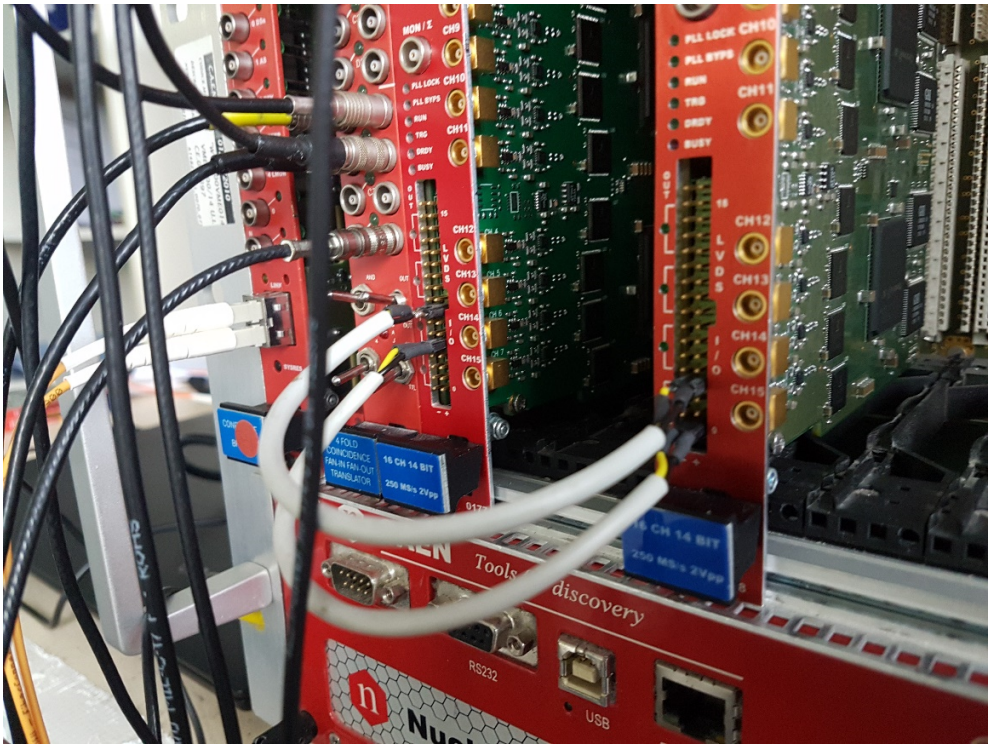


Fig. 4.2: RUN IN and BUSY daisy-chains through the LVDS connectors

   (c) The BUSY signal is propagated from one board to the other in daisy-chain through the LVDS connectors in the lower part of the board front panel, as shown in Fig. **4.2**. In particular, the LVDS

pair 4 of each board is configured as a BUSY output and should be connected to the LVDS pair 0 of the following board in the daisy chain (configured as a BUSY input), see Fig. **4.2**. Note that each board adds his own BUSY signal to the daisy-chained BUSY, so that at the end of the daisy chain one has the OR of all the BUSY signals from the boards. This latter signal, that is fanned-out from the TRG OUT connector of the last board in the daisy chain (see Fig. **4.3**) must be used to veto triggers to the boards;

Fig. 4.3: the signal from the TRG OUT connector must be used to veto the global trigger

(d) The timestamp of each board is conveniently delayed (through the register 0x8170), in order to eliminate timestamp shifts given by the daisy-chain delay. NB: This does not eliminate a possible shift of +/- 1 LSB given by the fact that triggers are non-synchronous.

5 Technical Support

CAEN makes available the technical support of its specialists for requests concerning the software and hardware. Use the support form available at the following link:

www.caen.it/support-services/support-form

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
Eckehardweg 10
42653 Solingen
Germany
Tel. +49.212.2544077
Mobile +49(0)15116548484
Fax +49.212.2544079
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

UM6064 - 725-730 DPP-ZLEplus User Manual rev. 2 - July 8th, 2020

00117-10-DGT28-MUTX