Rev. 3 - September 5th, 2025

742 Quick Start Guide

First Connection and Acquisition with 742 series

CAEN
Tools for Discovery

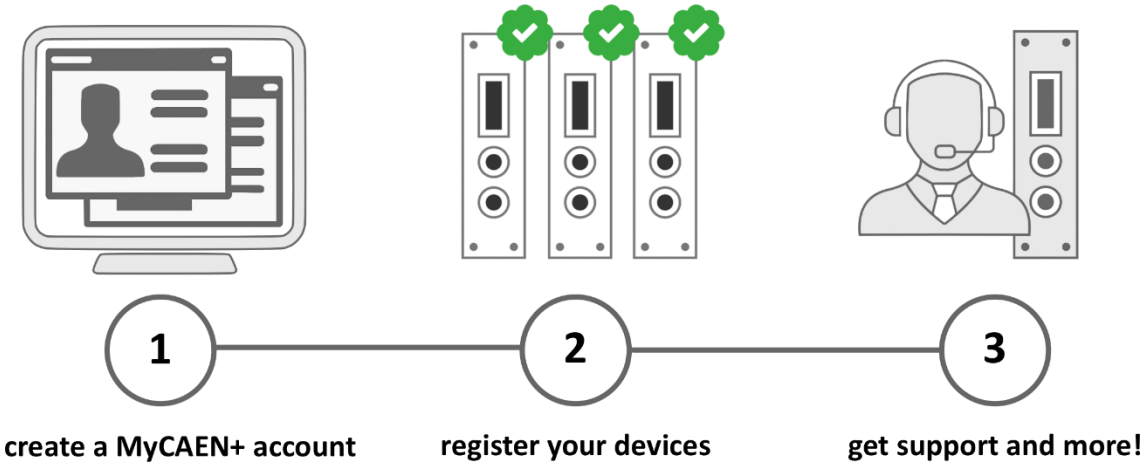
User Guide

GD5695

Register your device

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

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



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

Purpose of the Document

This guide wants to provide instructions for the first installation, connection, and acquisition with a digitizer of the 742 series. The software used in this guide is CAEN WaveDump, which can be used as a starting point for a custom acquisition software.

Change Document Record

Date	Revision	Changes
Jan. 5th, 2017	00	Initial release
Dec. 17th, 2018	01	Updated **MADE IN ITALY** in the **Disclaimer**. Updated Sec. **Drivers and Software** with new configuration file management. Introduced the **WaveDumpConfig_X742.txt** file in Sec. **Practical Use**.
Dec. 10th, 2021	02	Updated copyright. Updated Chap. **1**, Sec**. How to make an Acquisition with Self-Trigger**, Sec. **WaveDump Specific Functions for 742**
Sept. 5th, 2025	03	Reviewed Cover and End pages. Added Sec. **2.6.4**.

Symbols, Abbreviated Terms, and Notations

ADC	Analog-to-Digital Converter
AMC	ADC & Memory Controller
DAQ	Data Acquisition
DPP	Digital Pulse Processing
MCA	Multi-Channel Analyzer
OS	Operating System
PC	Personal Computer
PMT	Photo Multiplier Tube
QDC	Charge-to-Digital Converter
ROC	ReadOut Controller
TDC	Time-to-Digital Converter
USB	Universal Serial Bus

Reference Documents

[RD1] UM4279 – V1742 User Manual

[RD2] UM2091 - CAEN WaveDump User Manual

[RD3] UM1935 - CAENDigitizer User & Reference Manual

[RD4] UM5698 – 742 Raw Waveform Registers Description

[RD5] V1718 & VX1718 User Manual

[RD6] V2718 & VX2718 User Manual

[RD7] UM7685 - V3718 & VX3718 User Manual

[RD8] UM8305 - V4718 & VX4718 User Manual

[RD9] UM1934 - CAENComm User & Reference Manual

[RD10] GD2783 - First Installation Guide to Desktop Digitizers & MCA

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

Manufacturer Contacts

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

Limitation of Responsibility

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

Disclaimer

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

Made in Italy

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

Index

List of Figures

List of Tables

1 Introduction

This Guide provides simple instructions for those who are dealing with a x742 digitizer **[RD1]** for the first time.

The board functioning is based on the Switched Capacitor DRS4 chip[1]. The chip samples the input into 1024 analog memory cells (capacitors) written in circular buffers at high frequency (up to 5 GHz). The analog to digital conversion is not simultaneous with the sampling phase, and it starts as soon as the trigger condition is met. The trigger stops the DRS4 chip sampling (holding phase) and makes the cell content available for the ADC (see **Fig. 1.1**).

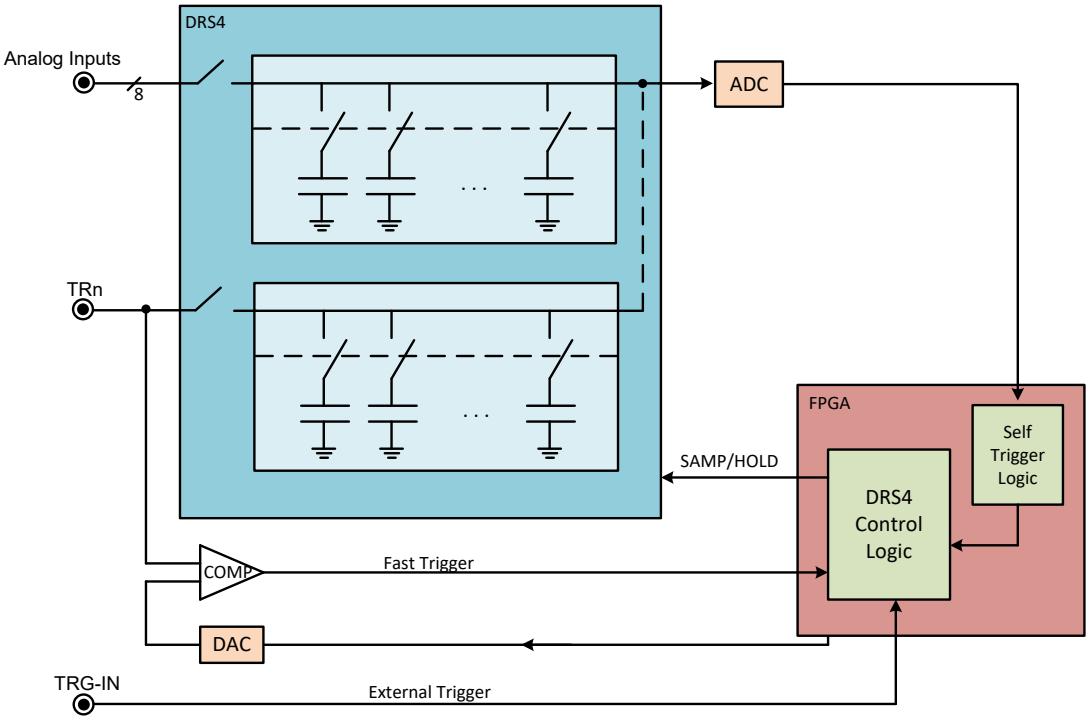


Fig. 1.1: Trigger management in the 742 series

This functioning has two major consequences:

1. there is an unavoidable dead-time when the DRS4 chip stops its acquisition and the ADC converts the capacitances, which corresponds to 110 µs in case only the analog inputs are digitized, and 181 µs in case also TRn are digitized.
2. the acquisition window is fixed to 1024 samples, that in case of 5 GHz corresponds to a maximum of about 200 ns. Options 512, 256, and 136 can be selected by software to reduce the amount of data to be transferred, but all the 1024 cells are converted anyway (no dead-time reduction).

Moreover, the trigger processing introduces a latency between the trigger arrival and the DRS4 holding phase that varies according to the trigger source. The user must consider it when choosing the trigger source for its setup and the type of signal. Four possible trigger sources are available:

1. **Software Trigger** (common to all enabled groups). The trigger is issued through a software write on the relevant FPGA register. This mode is mainly used for debugging purposes.
2. **External Trigger** (trigger on TRG-IN connector, common to all enabled groups). The TRG-IN connector accepts NIM and TTL input logic, which can be programmed via software. The TRG-IN signal is first processed by the mother board with a clock of about 58 MHz, and sent to the DRS4 to stop its acquisition with a latency of about

[1] Designed at Paul Scherrer Institute, PSI. Detailed documentation of the DRS4 chip is available at *http://drs.web.psi.ch/*

115 ns and a jitter of about 17 ns[2]. The latency of the external trigger makes this mode difficult to use at 5 GHz, where the maximum acquisition window is about 200 ns.

3. **Fast (Low Latency) Local Trigger** (trigger on TR0 and TR1 connectors, common to couples of groups[3]). The TRn connector accepts signals with maximum amplitude of 2 Vpp in case of Mezzanine PCB revision ≥ 1 (3 Vpp in case of Mezzanine PCB revision = 0). The signal is then attenuated to 1 Vpp to make it consistent with the DRS4 chip dynamic range. Refer to Sec. **How to make an Acquisition with Fast Trigger TRn** and **[RD1]** for additional details.

   This mode is called "Fast" or "Low Latency" since the latency from the trigger arrival and the DRS4 stop acquisition is reduced to about 42 ns with a jitter of about 8.5 ns. The signal on TRn is sent to a comparator with programmable threshold (no mother-board processing) whose output is sampled at about 117 MHz (twice the external trigger processing). When the TRn signal crosses the threshold, the acquisition of the DRS4 chip is stopped and the digitalization process starts.

   This trigger mode is convenient for high precision timing measurements, since the TRn can be digitized as the other analog inputs and reported in the output data as channel number 8 of each group. The trigger can therefore be used as a time reference for the input. The DRS4 sampling period becomes the time jitter of the trigger with respect to an input of the same group, which can reach 200 ps in case of sampling at 5 GHz. The resolution in a time of flight measurement reaches up to 50 ps in case of signals and TRn in the same TRn group, and 100 ps for signals and TRn in different groups.

Note: TR0 (TR1) is split into the two DRS4 of the mezzanine and follows two different path (two different ADCs and two memory buffers). This might imply that the digitized samples of TRn might have small differences from one group to the other.

4. **Self-trigger** (common to couples of groups, or to all groups in case of Global Trigger mode), the acquisition is controlled by combinations in logic OR of the channel self-triggers. For each group is possible to select a mask of channels that provides a trigger whenever the input crosses the threshold. In case of self-trigger mode, the channels belonging to group 0 and group 1 manages the acquisition of group 0 and group 1 simultaneously, while the self-trigger of the channels belonging to group 2 and group 3 (VME form factor only) controls the acquisition of group 2 and group 3 simultaneously. The two couples of groups (gr0/gr1, and gr2/gr3) can be considered independent (VME form factor only). In case of Global Trigger mode, the over-threshold signals from enabled channels are sent to the motherboard, which then generates a global acquisition trigger for the entire board.

IMPORTANT: *Since the self-trigger is made on the samples digitized by the ADC while the acquisition of the DRS4 is running (refer to Sec.* ***How to make an Acquisition with Self-Trigger*** *and* ***[RD1]****) and the ADC works at a frequency of about 30 MHz, the input signal width must be at least greater than 30 ns.*

IMPORTANT: *The self-trigger mode introduces a latency of about 320 ns (with a jitter of 17 ns) from the threshold crossing and the stop of the DRS4, and it is not possible to use this mode with DRS4 frequency equal to 5 GHz. One of the other options must be used to work with the self-trigger mode.*
In Global trigger mode, an additional latency of about 100 ns is introduced, making this mode not compliant with the DRS4 frequency = 5 GHz and 2.5 GHz, but it can be used when the board works at 1 GS/s, or 750 MS/s.

Note: Self-trigger option is available from AMC Firmware release greater or equal to 0.4.

Note: Global Trigger option is available starting from release **4.30_1.08**.

Sec. **Practical Use** will explain in detail how to configure the board to work with the three trigger modes.

[2] The TRG-IN latency has been reduced to 115 ns from ROC firmware revision 4.07, while for firmware revisions less than 4.07 the latency was 255 ns with a jitter of about 34 ns.
[3] TR0 manages the acquisition of group 0 and group 1, while TR1 manages the acquisition of group 2 and group 3 (VME form factor only).

Another important feature of x742 are the **corrections** to compensate the differences in the DRS4 chips **[RD1]**. These corrections are managed at software level, since the firmware on-board retrieves the raw data. There are three available corrections:

1. **Cell Index Offset** correction, which compensates the signal offset for the differences in cell amplitudes;
2. **Sample Index Offset** correction, which corrects the signal offset for a noise over the last 30 samples;
3. **Time** correction, which compensates the differences of the delay line of the chips.

The default corrections tables are provided by CAEN in the memory flash of the board. Wavedump software **[RD2]** (and the underlying CAENDigitizer library **[RD3]**) then can retrieve the tables and make the appropriate corrections. The user can leave the software automatically apply all the corrections, or decide which correction applies to which group. Sect. **WaveDump Specific Functions for 742** and **Corrections for 742** give more details about the 742 corrections.

2 Getting Started

2.1 Scope of the Chapter

This chapter provides an example on how to configure the 742 settings to make an acquisition with the three trigger modes described in Sec. **Introduction**.

2.2 System Overview

The system used for the acquisition is made of the following CAEN products:

- DT5742, 16+1 Channel 12 bit 5 GS/s Switched Capacitor Digitizer.
- Default Firmware for Waveform Recording for 742 series (release 4.25_1.05), running on the Digitizer.
- Wavedump demo software, release 3.10.3, running on the host station.

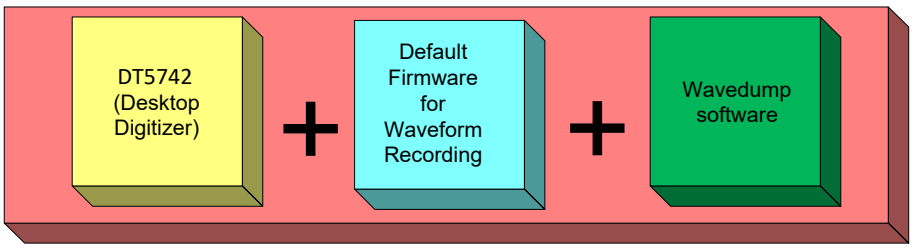


Fig. 2.1: System components

2.3 Hardware Setup

For the example described in this chapter we make use of a pulse generator to generate two simultaneous NIM pulses, which are used as trigger (TRG-IN or TRn) and as signal (CH0). Two cables of the same length are also used.

TRG-IN connector accepts also TTL logic signals prior board programming, while TRn can accept a wide selection of signals (negative, positive, bipolar), providing that the absolute amplitude is less than 2 Vpp. **Tab. 2.1** and **Tab. 2.2** show some examples of accepted signals on TRn and the corresponding values of DC Offset and Threshold. In the example of this Quick Start Guide, we make use of a NIM signal which is suitable both as a trigger and as analog input.

The pulse frequency is set to 200 Hz, and the pulse width to 50 ns.

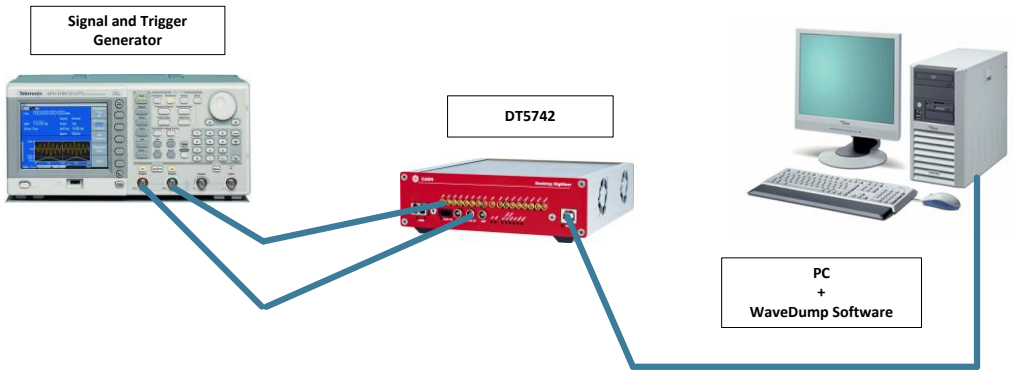


Fig. 2.2: The hardware setup

2.4 Drivers and Software

To manage the 742-acquisition 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 this case, we are going to describe the procedure for USB connection.

- ✓ DRIVERS
  - **USB 2.0** CAEN driver.

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

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

Note: Detailed installation steps of CAEN USB drivers for communicating with desktop digitizers are described for several Microsoft Windows OSs in **[RD10]**.

How to install the driver (Windows)

Download the latest release of the **USB driver** for Windows on CAEN website in the Download area of the DT5742 page, under the "Software" tab.

Unpack the **driver package**.

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

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

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

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

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

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

How to install the driver (Linux)

Download the latest release of the **USB driver** for Linux on CAEN website in the Download area of the DT5742 page, under the "Software" tab.

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

For Windows users:

CAEN provides the full installation package for WaveDump 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 WaveDump installation package** compliant with your OS from CAEN website on the WaveDump page in the 'Downloads' area (**login required**).
- **Extract** the **files** to your host.
- **Complete** the **installation wizard**.

WaveDump is then installed under the folder:

C:\Program Files\CAEN\Digitizers\WaveDump

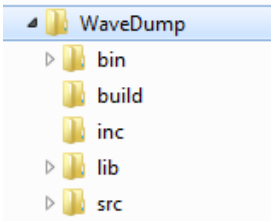


Fig. 2.3: Subfolders structure of WaveDump main directory

The "*bin*" subfolder contains the executable file (WaveDump.exe), a **general-purpose default configuration file (*WaveDumpConfig.txt*)**, the **x742 models default configuration file (*WaveDumpConfig_X742.txt*)** and the **x740 models default configuration file (*WaveDumpConfig_X740.txt*)**.

The *"build"* folder contains the Visual Studio project, while the header and the source code of the WaveDump are in the *"inc"* and *"src"* folders, respectively.

Note: Administrator rights are required to modify the *configuration files* of WaveDump 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 WaveDump, respectively.

- **Set** the proper connection settings in the ***WaveDumpConfig.txt*** file and **save**.
- **Set** all the other digitizer settings in the ***WaveDumpConfig_X742.txt*** file. *WaveDumpConfig.txt* must be used only for the connection settings
- **Double click** on the **WaveDump executable** file to run the software.
- Data is saved into the home path of the user: C:\Users\$USER_NAME$\WaveDump

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**). **Installation instructions** can be found in the **README file** inside each library folder.

- **Download** the WaveDump installation package for Linux on CAEN website in the 'Downloads' area at the WaveDump software page (**login is required**).
- **Unpack** the **installation package** (tar –zxf <WaveDump-x.y.z.tar.gz).
- **Follow** the instruction on the **INSTALL** file

  Type: ./configure

  make

  sudo make install

  Launch the software typing **wavedump**

The default configuration files are in:

/etc/wavedump/

Note: Administrator rights are required to modify the configuration file of the WaveDump software under the "/etc/wavedump" folder.

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

~/wavedump/Setup/<Configuration File>

where < Configuration File > can be:

WaveDumpConfig.txt (general purpose default configuration file);

WaveDumpConfig_X742.txt (default configuration file for x742 digitizers);

WaveDumpConfig_X740.txt (default configuration file for x740 digitizers).

and launch the software typing **wavedump** and the **path** of the configuration file.

2.5 Firmware

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

✓ **How to install the firmware**

Download the **Default Firmware** for Waveform Recording for 742 series on CAEN website in the 'Download' area at the DT5742 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**.

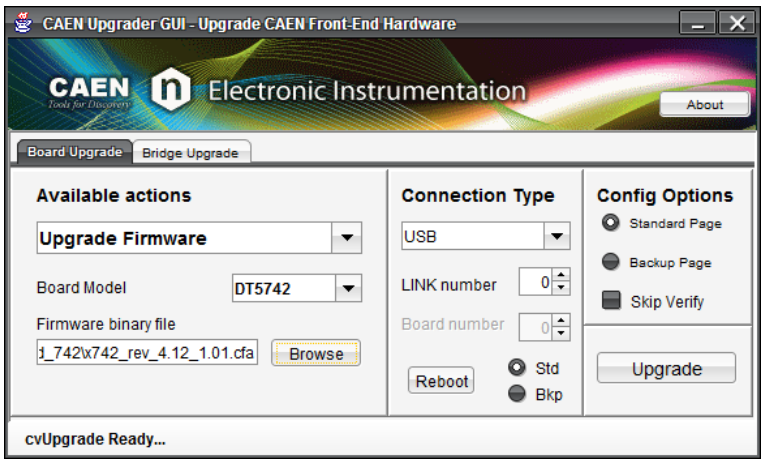


Fig. 2.4: CAENUpgrader settings for USB connection to DT5742.

2.6 Practical Use

With x742 digitizers, WaveDump software parses the ***WaveDumpConfig.txt*** file at start-up; once connected, it recognizes the target board as a x742, and configures it by executing all the other settings parsing the ***WaveDumpConfig_X742.txt*** file.

The user must so edit the ***WaveDumpConfig.txt*** only for the connection parameters, while any other parameter must be modified by editing ***WaveDumpConfig_X742.txt***.

2.6.1 How to make an Acquisition with Fast Trigger TRn

As described in Sec. **Hardware Setup**, we used two simultaneous NIM signals on CH0 and TR0. It is also possible to split the same NIM signal, since the TRn accepts a wide range of signals providing that the maximum amplitude is not higher than 2 V_{pp} for mezzanine PCB revision ≥ 1, and 3 V_{pp} for PCB revision 0.

Note: To check the PCB revision number, read bit[9] of register 0x1n88 **[RD4]**.

IMPORTANT: *The TRn input is attenuated by a factor of 2 (PCB revision ≥ 1), or 3 (PCB revision 0) to make it compliant with the 1 V_{pp} dynamics of the DRS4 chip. For signals higher than 2 V_{pp} (3 V_{pp}) it is recommended to use an external attenuator.*

Since the TRn acts as an input signal, it is possible to adjust its baseline position (i.e. the 0 Volt) to cover the full scale. This permits the use of several types of signals, bi-polar, negative, and positive. A list of accepted signals is reported in **Tab. 2.1** and **Tab. 2.2.** The TRn signal is then sent to a comparator that compares the TRn to the Trigger Threshold. When TRn crosses the threshold, the trigger is issued.

TRn DC Offset and TRn Trigger Threshold can be modified at the end of the ***WaveDumpConfig_X742.txt*** file (refer to Sec. **Drivers and Software**) according to the values reported in **Tab. 2.1** and **Tab. 2.2**. The modification of the DC Offset implies the modification of the Threshold as well. For additional details about the principles of TRn DC Offset and Threshold setting, please refer to the **Fast ("Low Latency") Trigger** Section of the Digitizer User Manual **[RD1]**.

Tables **Tab. 2.1** and **Tab. 2.2** report few examples of DC Offset and Threshold values (hexadecimal / decimal) for typical signals that can be fed into the TRn connector. The reported Threshold values allow the user to trigger at half of the signal height.

Mezzanine PCB Rev. ≥ 1	
ECL signal on TRn	TRn DC Offset = 0x55A0 / 21920 TRn Threshold = 0x6666 / 26214
NIM signal on TRn	TRn DC Offset = 0x8000 / 32768 TRn Threshold = 0x51C6 / 20934
Negative signal on TRn: V = 0 ÷ -400mV	TRn DC Offset = 0x8000 / 32768 TRn Threshold = 0x5C16 / 23574
Negative signal on TRn: V = 0 ÷ -200mV	TRn DC Offset = 0x8000 / 32768 TRn Threshold = 0x613E / 24894
Bipolar signal on TRn	TRn DC Offset = 0x8000 / 32768 TRn Threshold = 0x6666 / 26214
TTL on TRn or Positive signal on TRn: V= 0 ÷ ≥2V	TRn DC Offset = 0xA800 / 43008 TRn Threshold = 0x6666 / 26214
Positive on TRn: V= 0 ÷ 2V	TRn DC Offset = 0x91A7 / 37287 TRn Threshold = 0x6666 / 26214

Tab. 2.1: Examples of DC Offset and Trigger Threshold (in hexadecimal / decimal) for typical signals on TRn connector. Values are valid for mezzanine PCB revision ≥ 1.

Mezzanine PCB Rev.0	
NIM signal on TRn	TRn DC Offset = 0x1000 / 4096 TRn Threshold = 0x717D / 29053
Negative signal on TRn: V= 0 ÷ -400mV	TRn DC Offset = 0x1000 / 4096 TRn Threshold = 0x6E72 / 28274
Bipolar signal on TRn	TRn DC Offset = 0x1000 / 4096 TRn Threshold = 0x6C80 / 27776
TTL on TRn or Positive signal on TRn: V= 0 ÷ ≥2V	TRn DC Offset = 0x4000 / 16384

	TRn Threshold = 0x7158 / 29016

Tab. 2.2: Examples of DC Offset and Trigger Threshold (in hexadecimal / decimal) for typical signals on TRn connector. Values are valid for mezzanine PCB revision 0.

In case of PCB rev. ≥ 1 and NIM signal, write in the ***WaveDumpConfig_X742.txt*** file (end of the file):

```
[TR0]
DC_OFFSET               32768
TRIGGER_THRESHOLD       20934
```

In case the NIM signal is split into CH0 and TR0, use settings "*Negative signal on TRn: V= 0 ÷ -400mV".*

```
[TR0]
DC_OFFSET               32768
TRIGGER_THRESHOLD       23574
```

Finally adjust the DC offset of channel 0. It is possible to modify the field `DC_OFFSET` to adjust the DC Offset of all channels of group 0, or to set the field `GRP_CH_DC_OFFSET` to modify the DC Offset of each channel independently. Here the settings in the ***WaveDumpConfig_X742.txt*** file.

Option 1 -> modify the DC Offset of CH0 only

```
[0]
ENABLE_INPUT            YES
#DC_OFFSET               0
GRP_CH_DC_OFFSET        -12,0,0,0,0,0,0,0
```

Option 2 -> modify the DC Offset of all channels of group 0

```
[0]
ENABLE_INPUT            YES
DC_OFFSET               -12
#GRP_CH_DC_OFFSET        0,0,0,0,0,0,0,0
```

Verify that TRn is used to acquire and possibly to be digitized.

```
# FAST_TRIGGER: ...
FAST_TRIGGER   ACQUISITION_ONLY
```

```
# FAST_TRIGGER_DIGITIZING: ...
ENABLED_FAST_TRIGGER_DIGITIZING       YES
```

Leave the other settings of WaveDump as they are by default,

Note: In this example, we left the Post Trigger setting at its default value = 20%. In real cases, customers might take advantage to set it to 0 since the post trigger adds an additional latency to the fast trigger.

Start the acquisition and the board should start triggering based on the TR0. Here how the waveform plot should appear.

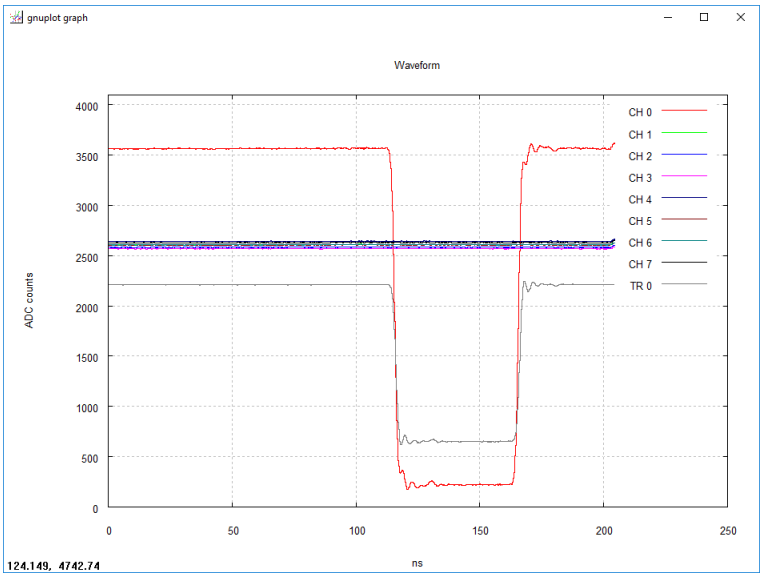


Fig. 2.5: Trigger on TR0. The signal on TR0 is also digitized and it can be used for an accurate measurement of time differences.

The two signals are superimposed as expected. The board is triggering on the falling edge of the TR0 signal, the post trigger is set to 20 % of the record length (+ about 42 ns of delay and about 8.5 ns of jitter). The DRS4 sampling is 5 GHz and the record length is equal to 1024 samples, i.e. 204 ns of record length.

From the WaveDump command line we can check that the trigger rate correctly reports 200 Hz, as we set 200 Hz of pulse frequency.

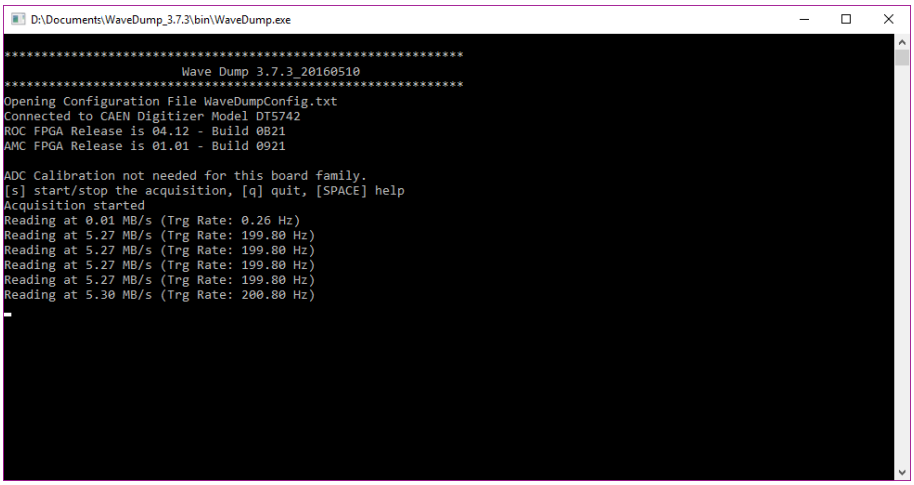


Fig. 2.6: WaveDump command interface showing the Readout Rate and the Trigger Rate.

2.6.2 How to make an Acquisition with TRG-IN

Sending a trigger in the TRG-IN connector will make all channels of the board trigger at the same time. As explained in Sec. **Introduction**, the major difference in using the TRG-IN vs TRn is the latency between the time the signal is processed and the stop of the DRS4 chip. In case of TRG-IN, this delay is about 115 ns (+17 ns of jitter).

To exploit the TRG-IN functioning, just move the signal from TRn to the TRG-IN connector and leave the WaveDump settings as in the previous example, just make sure that:

```
# EXTERNAL_TRIGGER: ...
EXTERNAL_TRIGGER    ACQUISITION_ONLY
```

Or:

```
# EXTERNAL_TRIGGER: ...
EXTERNAL_TRIGGER    ACQUISITION_AND_TRGOUT
```

The waveform plot will then appear as follows. The CH0 signal appears in advance with respect to the previous configuration, while the trigger has a relative delay of about 72 ns.

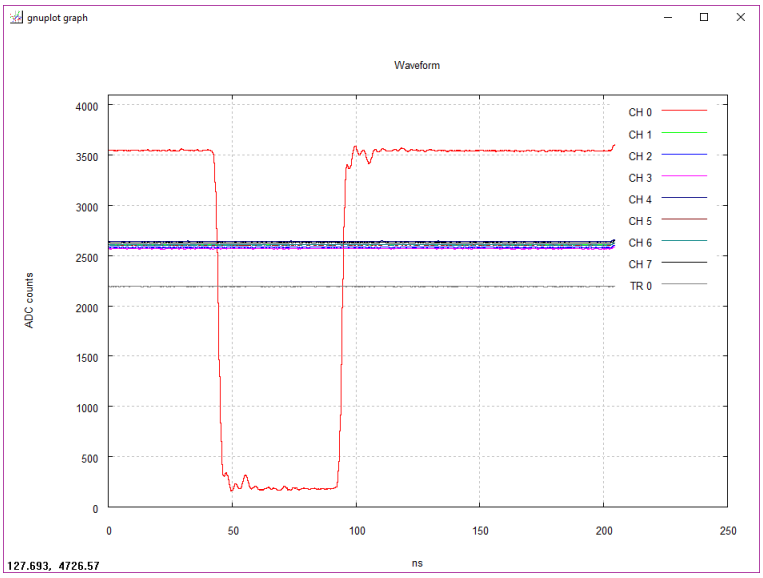


Fig. 2.7: Trigger on TRG-IN. The input on CH0 is shown in the waveform plot.

2.6.3 How to make an Acquisition with Self-Trigger

An additional trigger mode is available from mezzanine firmware revision 0.4, where each channel can self-trigger on its own input – leading edge discrimination – and combinations in logic OR of the self-triggers enable the groups to acquire at the same time. In particular, the self-trigger of the channels belonging to group 0 and group 1 manages the acquisition of group 0 and group 1 simultaneously, while the self-trigger of the channels belonging to group 2 and group 3 (VME form factor only) controls the acquisition of group 2 and group 3 simultaneously. Refer to register 0x1nA8 **[RD4]** for more details.

The DRS4 chip has two operating modes: "Transparent" and "Output". In Transparent mode (see **Fig. 2.8**), the input pulse is both sampled by the DRS4 capacitors (analog sampling) at high frequency, and made available at the output for the ADC digital sampling at a smaller rate, about 30 MHz. In transparent mode, the output stage is not a pure differential, since it has an offset and it is attenuated with respect to the signal properly stored on capacity. Transparent mode is the standard operating mode of the DRS4 chip, which continuously samples the input.

In Output mode (see **Fig. 2.9**), the input is no longer sampled, and the capacitors hold the acquired samples and send them one at a time to the ADC at a frequency controlled by the FPGA (readout frequency). The Output mode starts when a trigger condition is met (see **Fig. 1.1**). Samples in Output mode are those available in the readout for the user and they are correctly shaped.

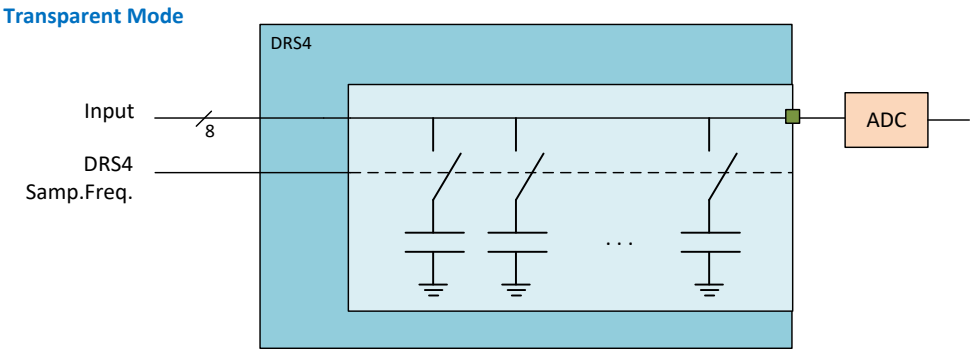


Fig. 2.8: Diagram showing the "Transparent Mode" functioning. The analog input is both sampled by the DRS4 capacitors (analog sampling) and sampled by the ADC (digital sampling) at a smaller rate. The output stage is distorted with respect to the Output mode.

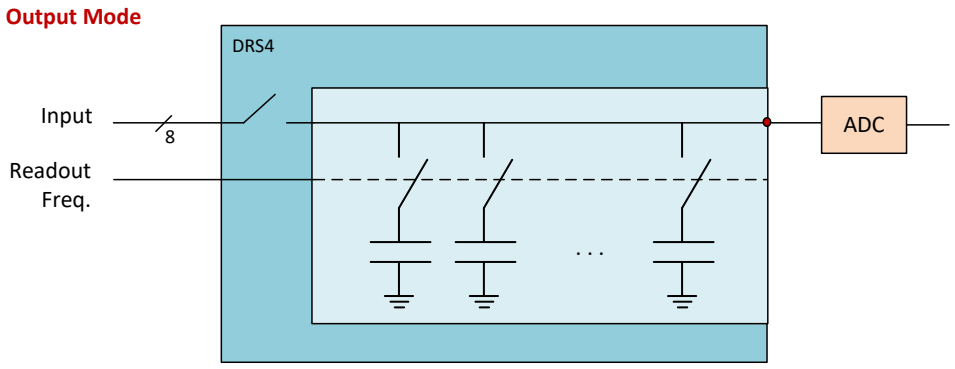


Fig. 2.9: Diagram showing the "Output Mode" functioning. The DRS4 capacitors release their capacitance at a frequency controlled by the FPGA (readout frequency), and the analog value is converted by the ADC. The output stage is correctly differential.

Since the Self-Trigger Logic inside the FPGA reads data from the ADC while the DRS4 chip works in Transparent mode, the Trigger Threshold has to be referred to the values read in Transparent mode itself, rather to the values reported in Output mode.

To correctly set the threshold value, it is first required to make an acquisition in Transparent mode to visualize the waveform as sampled by the ADC.

Considering that the ADC frequency is about 30 MHz, it is important that the input can be sampled by the ADC itself. In particular, the pulse width should be greater than 30 ns, and the input frequency should be high enough to visualize some pulses. We therefore increased the input frequency up to 200 KHz.

The procedure of reading from the ADC and processing data by the FPGA introduces a latency of about 320 ns (with a jitter of 17 ns) to stop the chip. This mode is therefore not compliant with the DRS4 frequency = 5 GHz. Select one of the other options to use the self-trigger mode.

The steps to enable the self-trigger mode are listed below:

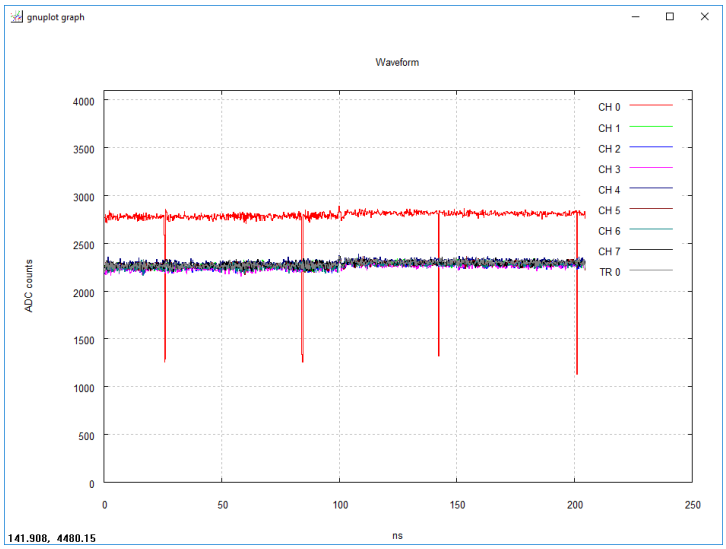
1. Enable the "Transparent Mode" writing bit[13] = 1 of register 0x8000, i.e. write 0x2000 with bit mask = 0x2000. In the ***WaveDumpConfig_X742.txt*** file, under the section `WRITE_REGISTER`:

```
# WRITE_REGISTER: generic write register access.
WRITE_REGISTER 8000 2000 2000
```

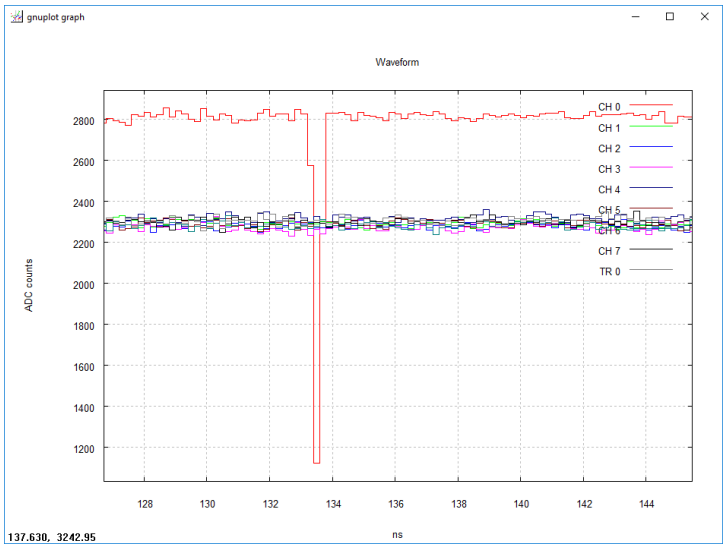
Note: Any time the config file is modified, save the file, close WaveDump and open it again to reload the configuration.

2. Feed the input pulse with 200 KHz of frequency into the CH0 connector;
3. Enable the continuous software trigger by pressing "T" in the WaveDump command line;
4. Start the acquisition. The following plot should appear:

Note: Corrections are not applied in Transparent mode. Consider also that the time scale in the plot is not correct, since it is not referred to the DRS4 chip frequency.



5. Stop the acquisition and make a zoom of one pulse to check a valid threshold value. In this case, we choose 2500 LSB counts.



6. Modify the ***WaveDumpConfig_X742.txt*** file to set the Threshold value of channel 0 through register 0x01n80, i.e. write 0x9C4 at 0x1080, with bit mask 0xFFFF.

```
WRITE_REGISTER 1080 9C4 FFFF
```

7. In the ***WaveDumpConfig_X742.txt*** file, disable the Transparent mode by commenting (or removing) the corresponding line.
8. In the ***WaveDumpConfig_X742.txt*** file, enable channel 0 to generate a self-trigger through register 0x1nA8, i.e. write 0x1 at 0x10A8, with bit mask 0xF

```
WRITE_REGISTER 10A8 1 F
```

To summarize, settings are now:

```
# WRITE_REGISTER: generic write register access.
#WRITE_REGISTER 8000 2000 2000
WRITE_REGISTER 1080 9C4 FFFF
WRITE_REGISTER 10A8 1 F
```

Before starting the acquisition remember to:

1. Set back the pulse frequency to 200 Hz to avoid dead time in the acquisition;
2. Disable the SW Trigger if still enabled;
3. Option DRS4 frequency = 5 GHz is not suitable for the self-trigger mode. Enable one of the other possible frequencies, as for example the 2.5 GHz:

```
DRS4_FREQUENCY 1
```

2.6.4 How to make an Acquisition with Global Trigger

Starting from firmware revision **4.30_1.08**, self-trigger signals from enabled channels can be used to generate a global trigger for the entire board. Specifically, over-threshold signals from the selected groups are sent to the motherboard, where a logical OR is performed among them to generate a global trigger distributed to all groups.

The following is a step-by-step example demonstrating how to configure the registers to generate a global trigger for the entire board, using the over-threshold signal from CH0 as the trigger source.

To enable the Global Trigger mode, it is first necessary to activate the self-trigger acquisition on the desired channels. The procedure to enable the self-trigger on CH0 has already been described in Sec. **2.6.3**. At the end of that configuration, the `WRITE_REGISTER` section in the ***WaveDumpConfig_X742.txt*** file should appear as shown below:

```
# WRITE_REGISTER: generic write register access.
#WRITE_REGISTER 8000 2000 2000
WRITE_REGISTER 1080 9C4 FFFF
WRITE_REGISTER 10A8 1 F
```

The further steps to enable the global trigger mode are listed below:

1. In the ***WaveDumpConfig_X742.txt*** file, enable the Global trigger mode by configuring the register 0x8000 as follows:
    a. Set bit[21]= 1 to disable the automatic acquisition from channel CH0;
    b. Set bits[31:28] = 0100 to send the over-threshold signal from CH0 to the motherboard.

```
WRITE_REGISTER 8000 40200000 F0200000
```

2. In the ***WaveDumpConfig_X742.txt*** file, enable GROUP0 to participate to the global trigger generation, setting the register 0x810C:

```
WRITE_REGISTER 810C 1 F
```

Before starting the acquisition remember to:

4. Set back the pulse frequency to 200 Hz to avoid dead time in the acquisition;
5. Disable the SW Trigger if still enabled;
6. Option DRS4 frequencies = 5 GHz, 2.5 GHz are not suitable for the global trigger mode. Enable one of the other possible frequencies, as for example the 1 GHz:

```
DRS4_FREQUENCY 2
```

To summarize, settings are now:

```
# WRITE_REGISTER: generic write register access.
#WRITE_REGISTER 8000 2000 2000
WRITE_REGISTER 1080 9C4 FFFF
WRITE_REGISTER 10A8 1 F
WRITE_REGISTER 8000 40200000 F0200000
WRITE_REGISTER 810C 1 F
```

2.6.5 WaveDump Specific Functions for 742

Here follows the list of WaveDump settings for 742 included in the ***WaveDumpConfig_X742.txt*** file:

2.6.6 **CORRECTION_LEVEL** <CORR_MASK> <CUST_TABLE_MASK> <FILENAME1> <FILENAME2> ...

This command allows to apply the data correction (742 digitizer family only). There are three types of corrections: cell offset, index sampling, and time correction (see **[RD1]**). The three correction files are available on each digitizer flash and they can be automatically applied during the event decode. The user can also use his/her custom correction files. Custom files should have the following name structure:

a. BaseInputFileName + "_cell.txt" for the cell offset corrections

b. BaseInputFileName + "_nsample.txt" for the index sampling correction

c. BaseInputFileName + "_time.txt" for the time correction

CORR_MASK (correction mask) allows to select the combination of corrections to be applied. Options are:

- CORR_MASK = AUTO the three corrections are automatically read and applied to the event (this is the default configuration). The following fields must be blank.
- CORR_MASK corresponds to a 3-bit number, where bit[0] corresponds to the cell offset correction, bit[1] to the index sampling correction, and bit[2] to the time correction. For example: if you want to apply only the first and the third correction, CORR_MASK = 5, etc.

CUST_TABLE_MASK identifies the groups to which the corrections are applied.

This field must be filled only when CORR_MASK value is different from AUTO. Options are:

- CUST_TABLE_MASK = AUTO: the corrections specified in CORR_MASK are applied to all groups.
- CUST_TABLE_MASK corresponds to a 4-bit number, where *n* bit corresponds to the *n* group to be enabled for corrections. For example, if you want to set the corrections for groups 0, 2 and 3, CUST_TABLE_MASK = 13, etc.

When CUST_TABLE_MASK is different from AUTO the user must specify the file name to be used for each group of interest.

FILENAME1, FILENAME2, … corresponds to the BaseInputFileName of the correction files to be used for the group enabled by the CUST_TABLE_MASK value.

EXAMPLES:

1. Use of the default configuration. The software automatically reads the three correction files from the digitizer flash and applies them to the events.

   ```
   CORRECTION_LEVEL AUTO
   ```

2. Only some of the corrections are enabled and applied to all groups. For example, you can apply the cell offset and the time corrections.

   ```
   CORRECTION_LEVEL 5 AUTO
   ```

   Analogously it is possible to disable all corrections.

   ```
   CORRECTION_LEVEL 0 AUTO
   ```

3. Different corrections are applied to different groups. The specific file name for each group must be specified. For example, if you want to apply the cell offset and time corrections to group 0, 1 and 2 (VME form factor) you should write:

   ```
   CORRECTION_LEVEL 5 7 FILE_GR0 FILE_GR1 FILE_GR2
   ```

   Where "FILE_GRn" is the "BaseInputFileName" for group n. All files must be available in the working folder of WaveDump, otherwise the full path must be specified.

2.6.7 DRS4_FREQUENCY option

This setting programs the DRS4 chip frequency.

option can be:

0: 5 GHz (default value);

1: 2.5 GHz;

2: 1 GHz.

3: 750 MHz

Note: Option 3 (750 MHz) requires a 742 AMC firmware release 1.00 or higher. Furthermore, the board should have the data corrections for this frequency. In case your board does not have the 750 MHz corrections, contact CAEN (see Chapter Errore. L'origine riferimento non è stata trovata.) for the upgrade.

2.6.8 RECORD_LENGHT Ns

Indicates the number Ns of samples to be acquired for each trigger (acquisition window).

Note: for 742 family, the options available are only 1024, 520, 256 and 136.

2.6.9 FAST_TRIGGER option

This command allows to use the fast trigger inputs TR0 and TR1 to trigger the data acquisition of groups 0-1, and 2-3 respectively.

option can be:

ACQUISITION_ONLY to enable it;

DISABLED to disable it.

2.6.10 ENABLED_FAST_TRIGGER_DIGITIZING option

Signal from fast trigger can be digitized and made available for readout on the eighth channel of each group.

option can be:

YES to enable it;

NO to disable it.

2.6.11 GRP_CH_DC_OFFSET dc_0, dc_1, dc_2, dc_3, dc_4, dc_5, dc_6, dc_7

The GRP_CH_DC_OFFSET command allows to adjust the DC_OFFSET level for each channel of a group.

dc_0 .. dc_n are float numbers that indicate the DC offset level for channel 0, …, n of the groups. Values range from −50 to 50, where −50 corresponds to a dynamic from -3FSR/2 to -FSR/2 (maximum negative signal), 50 corresponds to a range from +FSR/2 to +3FSR/2 (maximum positive signal). Default value is 0, which corresponds to a signal dynamics of-FSR / 2 to +FSR / 2 (bipolar signal).

2.6.12 POST_TRIGGER value

This command indicates the post-trigger size in percentage of the total record length. In case of x742 digitizers there is an additional delay of 42 ns on TRn, 115 ns on TRG-IN, and 320 ns on Self-Trigger, which is added to the post-trigger.

value is an integer value ranging from 0 to 100.

2.6.13 [TR0], [TR1]

Adjust the DC_OFFSET and TRIGGER_THRESHOLD for the TR0 and TR1 signals.

2.6.14 DC_OFFSET `value`

This setting programs the `DC_OFFSET` level of the Fast Trigger channel.

`value` ranges from `0` to `65535`, where `0` corresponds to a signal dynamics from -FSR to 0 (completely negative signal), and `65535` corresponds to a signal dynamics from 0 to FSR (completely positive signal).

2.6.15 TRIGGER_THRESHOLD `value`

Set the `TRIGGER_THRESHOLD` for the comparison level of the Fast Trigger channel.

`value` ranges from `0` to `65535`.

Refer to **Tab. 2.1** and **Tab. 2.2** for examples of possible values of DC_OFFSET and TRIGGER_THRESHOLD.

2.7 Corrections for 742

To compensate the unavoidable differences in the DRS4 chips construction it is necessary to correct the raw data. Corrections are not directly applied in the FPGA, and they must be implemented either run-time or off-line.

For *run-time corrections*, the user can take advantage of the CORRECTION_LEVEL function of WaveDump (see **CORRECTION_LEVEL** <CORR_MASK> <CUST_TABLE_MASK> <FILENAME1> <FILENAME2> ... paragraph), which allows the user to select either automatic corrections, or select which correction is applied to which group.

If the user wants to apply its own corrections, he/she can use the CAENDigitizer function `GetCorrectionTable` **[RD3]** to retrieve the default correction files from the board and modify them with his/her own values.

The list of CAENDigitizer functions to be used on-line are:

- **LoadDRS4CorrectionData**, loads the correction parameters stored on board. The correction parameters to load depend on the operating sampling frequency.
- **DecodeEvent**, decode the event and apply the correction to data if **LoadDRS4CorrectionData** has been previously called
- **Enable/Disable DRS4Correction**, enables/disables the data correction in the x742 series. When enabled, the data correction through the **DecodeEvent** function only applies if **LoadDRS4CorrectionData** has been previously called, otherwise the **DecodeEvent** runs the same, but data will be provided out not compensated
- **GetCorrectionTables**, reads the correction tables from the x742 digitizer flash memory, related to the selected sampling frequency, and fills in a structure with the read values. In this way, the stored correction table become available for the user.

Finally, it is also possible to save the raw data and *apply the corrections off-line*. An example code is available in the CAENDigitizer library. To access the code, download and install the CAENDigitizer library (the prior installation of CAENVMElib **[RD5]**, **[RD6], [RD7], [RD8]** and CAENComm library **[RD9]** is required) and include the examples in the installation. Then access to the subfolder called “x742_DataCorrection”.

C:\Program Files\CAEN\Digitizers\Library\Samples\x742_DataCorrection

Here the list of CAENDigitizer functions **[RD3]** to be used off-line.

- **LoadCorrectionTables**, loads the correction tables stored onto the board into a user defined structure.
- **ApplyDataCorrection**, applies the desired correction data (configured through a mask) to the raw data acquired by the user.
- **GetNumEvents**, gets the current number of events stored in the acquisition buffer.,
- **GetEventPtr**, retrieves the event pointer of a specified event in the acquisition buffer.
- **X742_DecodeEvent**, decodes a specified event stored in the acquisition buffer writing data in Evt memory.

3 Technical Support

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

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

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

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

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

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

CAEN GmbH
Brunnenweg 9
64331 Weiterstadt
Germany
Phone +49 212 254 40 77
Fax +49 212 254 40 79
info@caen-de.com
www.caen-de.com

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

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

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

UM2462 - SY4527 - SY4527LC Power Supply Systems rev. 19 - September 5th, 2025 00000-00-R5560SE-MUTX