Rev. 3 – July 12th, 2021

CAENScope

CAEN Digitizer Oscilloscope Tool

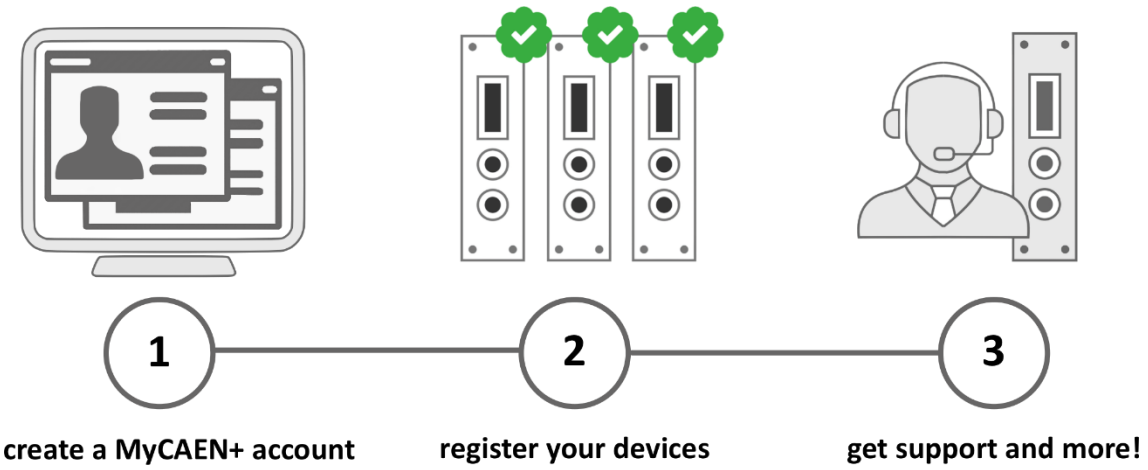
CAEN
Tools for Discovery

User Manual
UM2092

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 this Manual

This User Manual contains the full description of the CAENSCOPE software rev. 1.3, the CAEN Digitizer Oscilloscope Tool.

Change Document Record

Date	Revision	Changes
Apr 23rd, 2015	00	Initial release
Jun 23rd, 2015	01	Updated CAENSCOPE logo in the cover picture. Updated: Chap. **2**; Sec. **Cursors and Region of Interest (ROI)** in Chap. **10**
Dec 22nd, 2019	02	Added support to the 725 digitizer series in **Tab. 1.1**. Updated Linux installation instructions in Sec. **Software**. Added *clocktime* description in Sec. **Text Event Log Files**
Jul 12th, 2021	03	Added support to A4818 and V3718 in **Tab. 1.2**, **Tab. 2.1**. Tab. 4.1: **Available Connections involving A4818 link adapter**Added "Advanced" menu item in Sec. **Menu Bar Items: File / View / Advanced / Help**. Added **Tab. 4.1** for A4818 connections. Added raw binary format description in Sec. **Menu Bar** and Sec. **Binary File Recording**. Inserted Sec. **Binary (Raw) Event File Format** and Sec. **Playing Back Recorded Data**. Inserted Chap. **13** describing the C++ class for raw binary file processing.

Symbols, abbreviated terms and notation

AMC	ADC Mezzanine Card
CPU	Central Processing Unit
DPP	Digital Pulse Processing
DTD	Data Type Definition (XML file validation)
GPIO	General Purpose Input Output
LED	Light Emitting Diode
OS	Operating System
PCI	Peripheral Component Interconnect
PCIe	Peripheral Component Interconnect express
ROC	ReadOut Controller
ROI	Region Of Interest
USB	Universal Serial Bus
DTD	Data Type Definition (XML file validation)
SCA	Switched Capacitor Array

Reference Documents

[RD1] GD2783 – First Installation Guide to Desktop Digitizers & MCA

[RD2] GD2512 – CAENUpgrader QuickStart Guide

[RD3] UM1935 – CAENDigitizer User & Reference Manual

[RD4] AN2472 - CONET1 to CONET2 migration

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

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

The CAENSCOPE software provides a User-friendly interface that puts the acquisition power of CAEN digitizers at the fingertips of Users concerned with the simultaneous and synchronized acquisition of transient waveforms from multi-channel systems. The User interface is similar to that of a digital oscilloscope, with some differences, namely in the front-end section, that will be indicated in this manual.

CAENSCOPE supports a wide range of CAEN Digitizers in all form factors, provided they run the waveform recording firmware. Currently, the CAEN DT57XX Series (Desktop Digitizers), CAEN V17XX Series (VME Modular Digitizers) and N67XX Series (NIM Modular Digitizers) indicated in **Tab. 1.1** are supported.

Digitizer Series
720
724
725
730
751

Tab. 1.1: Supported digitizer series

The CAENSCOPE software DOES NOT support the CAEN Digitizers when equipped with any version of the DPP firmware.

How to check and identify the firmware currently running on the target digitizer is reported in Sec. **Firmware Compatibility**.

Firmware Compatibility

The CAENSCOPE software can be used with the Waveform Recording version of the Digitizers firmware only and DOES NOT support the CAEN Digitizers that are running any version of the DPP firmware.

To check if the firmware version installed on your CAEN digitizer is a Waveform Recording or a DPP one, CAEN provides the CAENUpgrader software tool **[RD2]**.

To read the firmware version:

- Connect your CAEN digitizer to the computer through one of the supported communication links, then run the CAENUpgrader software.
- Select the *Get Firmware Release* option in the "Board Upgrade" tab. Then, select your board model and set the connection parameters. Some sample hardware connection options are described in **Tab. 1.2** with the relevant connection parameters.

Connection chain	Conn. Type	Conn. Parameters		
		Link Nr	Board Nr	VME Base Address
PC –> **USB** –> DT57xx/N67xx	USB	0	not required	not required
PC –> **USB** –> V1718 –> **VME** –> V17xx	USB	0	not required	32100000[1]
PC –> **USB** –> DT57xx/N67xx[2]	USB	1	not required	not required
PC –> **PCI** –> A2818 –> **CONET** –> DT57xx/N67xx[3]	OPTLINK	0	0	not required
PC –> **PCI** –> A2818 –> **CONET** –> V17xx[3]	OPTLINK	0	0	0
PC –> **PCI** –> A2818 –> **CONET** –> V2718 –> **VME** –> V17xx[3]	OPTLINK	0	0	32100000[1]
PC –> **PCI** –> A2818 –> **CONET** –> V17xx[3][4]	OPTLINK	0	1	0
PC –> **USB** –> V3718 –> **VME** –> V17xx	USB	0	not required	32100000[1]
PC –> **PCI** –> A2818 –> **CONET** –> V3718 –> **VME** –> V17xx[3]	OPTLINK	0	0	32100000[1]
PC –> **USB** –> A4818 –> **CONET** –> V17xx	USB_A4818	15105[5]	0	not required
PC –> **USB** –> A4818 –> **CONET** –> DT57xx/N67xx	USB_A4818	15105[5]	0	not required
PC –> **USB** –> A4818 –> **CONET** –> V2718 –> **VME** –> V17xx	USB_A4818	15105[5]	0	32100000[1]
PC –> **USB** –> A4818 –> **CONET** –> V3718 –> **VME** –> V17xx	USB_A4818	15105[5]	0	32100000[1]
PC –> **USB** –> A4818 –> **CONET** –> V17xx[4]	USB_A4818	15105[5]	1	0

[1] *This VME base address is only as an example. In order to know the correct VME base address to be used, please refer to the Digitizer's User Manual*
[2] *It is supposed that at least two USB ports are used by the PC to communicate with as many digitizers (see the examples at the end of **[RD3]**)*
[3] *The same applies if using the A3818 PCIe card*
[4] *The Digitizer is intended to be part of an optical Daisy chain (see the examples at the end of **[RD2]**)*
[5] *In case of a connection via the A4818 link adapter, the Link Nr field is substituted by the A4818 PID field. The PID number inserted is only an example.*

Tab. 1.2: Examples of hardware connections and relevant communication parameters

- Press the *Get Fw Rel* button. The firmware version will be displayed in a pop-up window like that in **Fig. 1.1**. The typical Waveform Recording firmware release has the following format:

  XX.YY_WW.ZZ

  where XX.YY is major/minor revision number of the mainboard (ROC) FPGA firmware and WW.ZZ is the major/minor revision number of the channel (AMC) FPGA firmware.

  The digitizer firmware type is determined by the AMC version number. Waveform recording firmware will have an AMC major version number equal to 0. On the contrary, a DPP firmware release format features a major revision number which is equal to or greater than 128 for the AMC FPGA, identifying the kind of DPP algorithm, typically in the following format:

  XX.YY_www.ZZ

In case a DPP firmware is running, use CAENUpgrader to upgrade the digitizer with the latest available version of the Waveform Recording firmware. Firmware updates are available for free download (**login is required**) on CAEN website at the digitizer web page.

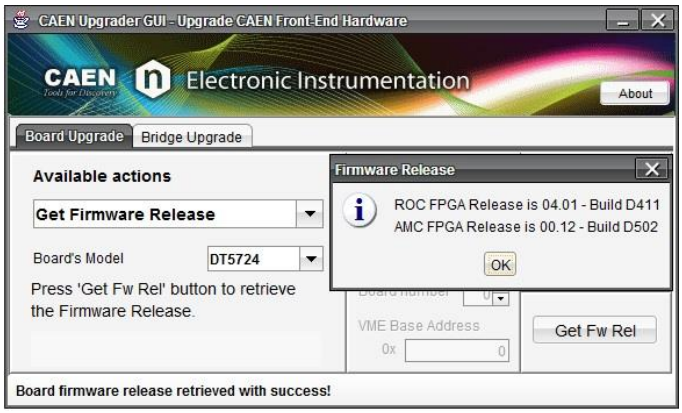


Fig. 1.1: The *Get Firmware Release* option of CAENUpgrader

2 Installation

CAENSCOPE software is provided both in Windows® and Linux® installation packages.

Before installing the CAENSCOPE software:

- **Make sure** that your **hardware** (Digitizer and/or Bridge, or Controller) is **properly installed** (according to the instructions in the User Manual of the related hardware) and the relevant **communication cable** (optical fiber or USB cable) is **properly connected**.
- **Make sure** you **have installed** the required **driver** for your OS and for the communication link to be used (see Sec. **Drivers**).
- **Make sure** you have installed the required additional packages, if working with Linux OS:
  - ***sharutils***
  - ***libXft***
  - ***libXss*** (specifically for Debian derived distribution, e.g. Debian, Ubuntu, etc.)
  - ***libXScrnSaver*** (specifically for RedHat derived distributions like RHEL, Fedora, Centos, etc.).

Drivers

Depending on which CAEN hardware is in the setup, and the operating system used, the User must install the correct driver required by the communication link he/she intends to use and compliant to the OS (Windows or Linux).

CAEN provides the drivers for all the supported types of physical communication channels:

- **USB 2.0** is used by Desktop and NIM Digitizers and the V1718/VX1718 or V3718/VX3718 Bridges in case of VME64/VME64X Digitizers.
- **USB 3.0** is used by the A4818, adapting this link to the CONET Optical Link.
- **CONET Optical Link** is supported by the A2818 PCI, the A3818 PCIe and the A4818 Controllers and supported directly by all form factors (Desktop, NIM, VME64 and VME64X Digitizers).
- **VME bus** is restricted to VME64 and VME64X Digitizers and accessed by the V1718, V2718 and V3718 Bridges.

Some possible hardware connection options are described in **Tab. 2.1** with the relevant required drivers.

HW connection chain	Comm. Link	Drivers
PC –> **USB** –> DT57xx/N67xx	USB	Driver for DT/NIM digitizers
PC –> **USB** –> V1718 –> **VME** –> V17xx	USB	Driver for V1718 Bridge
PC –> **USB** –> V3718 –> **VME** –> V17xx	USB	Driver for V3718 Bridge
PC –> **PCI** –> A2818 –> **CONET** –> DT57xx/N67xx/V17xx	OPT	Driver for A2818 Controller
PC –> **PCIe** –> A3818 –> **CONET** –> DT57xx/N67xx/V17xx	OPT	Driver for A3818 Controller
PC –> **USB** –> A4818 –> **CONET** –> DT57xx/N67xx/V17xx	USB_A4818	Driver for A4818 Controller
PC –> **PCI** –> A2818 –> **CONET** –>V2718 –> **VME** –> V17xx	OPT	Driver for A2818 Controller
PC –> **PCIe** –> A3818 –> **CONET** –>V2718 –> **VME** –> V17xx	OPT	Driver for A3818 Controller
PC –> **USB** –> A4818 –> **CONET** –>V2718 –> **VME** –> V17xx	USB_A4818	Driver for A4818 Controller
PC –> **PCI** –> A2818 –> **CONET** –>V3718 –> **VME** –> V17xx	OPT	Driver for A2818 Controller
PC –> **PCIe** –> A3818 –> **CONET** –>V3718 –> **VME** –> V17xx	OPT	Driver for A3818 Controller
PC –> **USB** –> A4818 –> **CONET** –>V2718 –> **VME** –> V17xx	USB_A4818	Driver for A4818 Controller

Tab. 2.1: Examples of hardware setups and driver requirements

CAEN drivers are available for free download (**login is required**) on CAEN website at the web page of the related Digitizer, Bridge and Controller.

Driver installation instructions can be found in the V1718, V3718, A2818, A3818, A4818 User Manuals and Data Sheets, while a Guide **[RD1]** is dedicated to the USB driver installation for Desktop and NIM digitizers.

Software

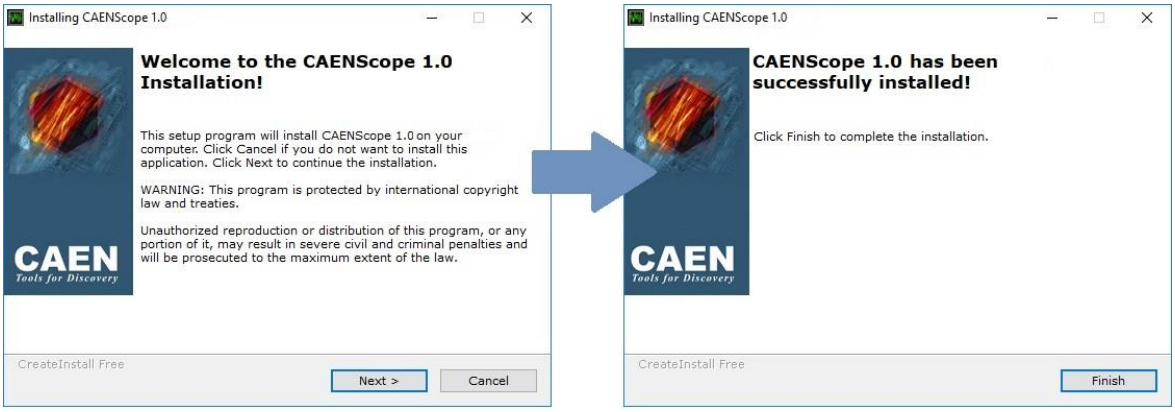
Download the CAENSCOPE installation package for your Operating System from the Downloads -> Software area of CAENScope web page.

For **WINDOWS** Users:

Unpack the compressed file to your host PC and run the program installer following the installation wizard.

Inside the CAENScope destination folder (default path is "*C:\Program Files\CAEN\CAENScope*"), the CBinaryIn.h and CBinaryIn.cpp files are also present and can be used for raw binary output files processing (see Chap. **13**).

Run CAENSCOPE.exe from the destination folder (default path is *C:\Program Files\CAEN\CAENScope*) or shortcut, if generated.



For **LINUX** Users:

Unpack the compressed file using the command "tar xzf name-of-downloaded-file"; the "bin" and "lib" directories are generated.

Inside the "lib" folder, the CBinaryIn.h and CBinaryIn.cpp files are present and can be used for raw binary data files processing (see Chap. **13**).

Issue "./scope" from the "bin" directory to run the software.

3 Uninstallation

In **WINDOWS** OS: select CAENSCOPE in the “Program and Functionalities” list in the *Control Panel*, then execute “uninstall”.

In **LINUX** OS: delete the directory tree in which the compressed file was unpacked.

4 Initialization

System Initialization

Upon launching the CAENSCOPE software, the following display will appear, consisting of an upper Menu Bar and of a graphical area display (as the program starts it will display the CAEN logo splash screen).

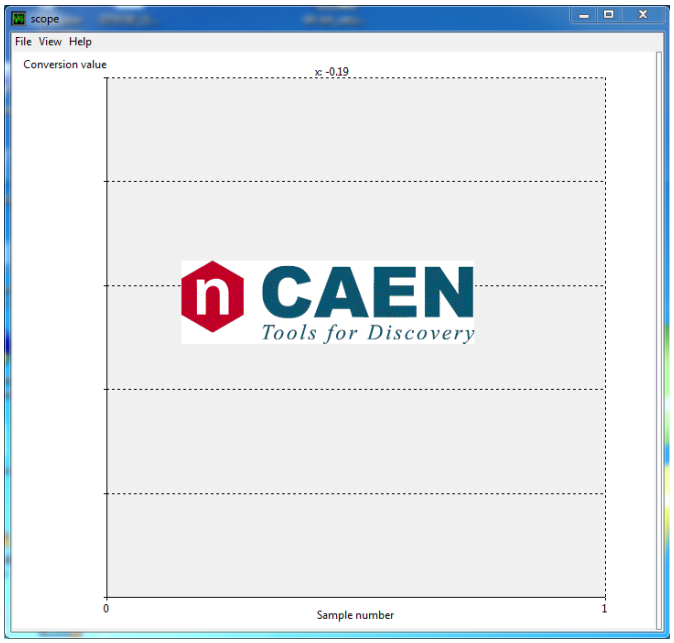


Fig. 4.1: CAENSCOPE Start up Screen

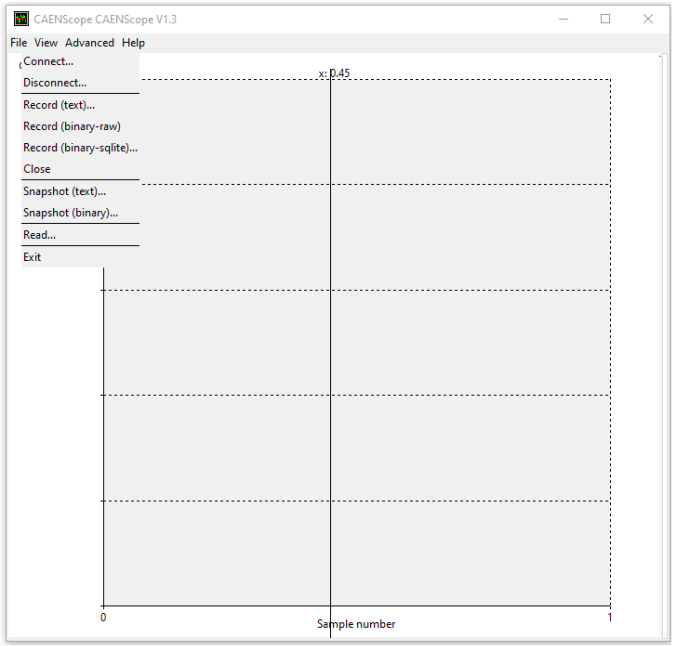


Fig. 4.2: CAENSCOPE Start-up Screen with File Pull Down Menu

Menu Bar

The Menu bar is always present at the top of the CAENSCOPE main window, just below the window's title bar. The User can access the menu items, as shown in **Fig. 4.3** below.

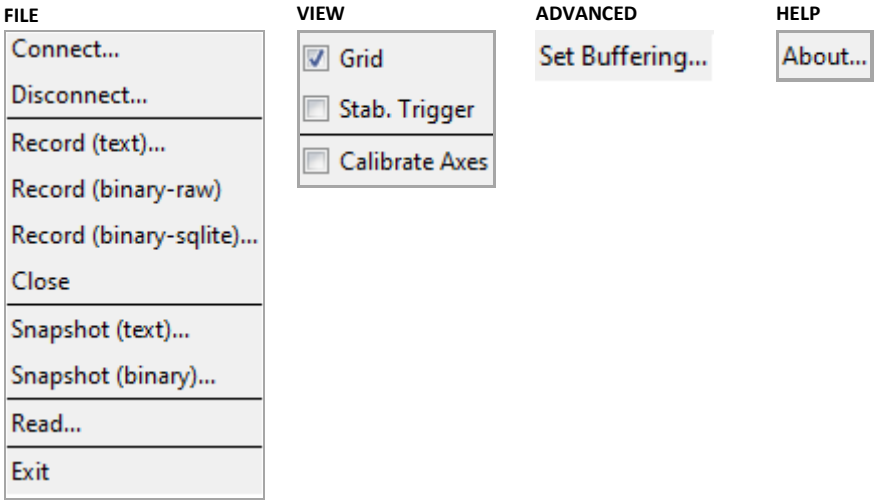


Fig. 4.3: CAENSCOPE Menu bar items and relevant sub-menu

Menu Bar Items: File / View / Advanced / Help

By left clicking on the Menu Bar items, Pull-down Menus are displayed, indicating the availability of the following functions:

1. "**File**" Pull-down Menu:

   Connect / Disconnect / Record (text)/ Record (binary-raw) / Record (binary-sqlite) / Close/ Snapshot (text) / Snapshot (binary) / Read/ Exit

   - **Connect**: Allows the User to select the digitizer CAENScope will use and connect it to the program.
   - **Disconnect**: Disconnects the connected digitizer from CAENScope.
   - **Record (text)**: Enables the recording of waveforms - being acquired by the digitizer - into a text file. The resulting file is in XML format and must have the file type **.xml**.
   - **Record (binary-raw)**: Initiates the recording of waveforms - being acquired by the digitizer - into a raw binary file. This data saving mode supports high speed output of trace and metadata. The resulting file must have the file type **.bin**.
   - **Record (binary-sqlite)**: Initiates the recording of waveforms - being acquired by the digitizer - into a binary SQLite3 database file (see https://www.sqlite3.org for more information). These files must have the file type **.db**.
   - **Snapshot (text)**: Opens a pop-up dialog directory allowing the User to choose a file into which the currently displayed waveforms will be saved. The resulting file is in XML format, identical to the format used by the **Record (text)** menu item.
   - **Snapshot (binary)**: Opens a pop-up dialog directory allowing the User to choose a file into which the currently displayed waveforms will be saved. The resulting file is an SQLite3 database file with a schema identical to that created by **Record (binary-sqlite)** menu item.
   - **Close**: If any of the Record options has been chosen, closes the event file and ends recording.
   - **Read**: Allows the User to open a waveform file and display waveforms in the file. Note that files not in SQLite3 format will be converted into a temporary SQLite3 database file first. A pop-up dialog prompts for the file to view. The file type (extension) determines the format of the file.
   - **Exit**: Exits the CAENSCOPE program.

2. "**View**" Pull-down Menu:

   Grid / Stab.Trigger / Calibrate Axes

   - **Grid**: Enables the display of grid lines on the graphic area.

- **Stab.Trigger**: Enables/disables the trigger stabilization by software. Since the trigger is clocked in with the FPGA clock which is generally not as fast as the digitizer clock, trigger positions will jitter in time. When enabled, **Stab. Trigger** stabilizes the trigger position by shifting the trigger position to the position determined by the Post Trigger parameter (see Sec. **Post-trigger**). This is done by locating the threshold crossing on a waveform-by-waveform basis.
- **Calibrate Axes**: When selected, this command changes the X-axis units from "Samples" to "nanoSeconds". The corresponding graph indications and knob descriptive units will be toggled from uncalibrated (raw digital) units to calibrated (engineering) units.

3. "**Advanced**" Pull-down Menu:

Set Buffering…

- **Set Buffering…**: A pop-up dialog box displays the value set for the 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 1 to 1023.

4. "**Help**" Pull-down Menu:

About…

- **About…**: A pop-up dialog box displays the CAENSCOPE software Release Version, author and copyright information.

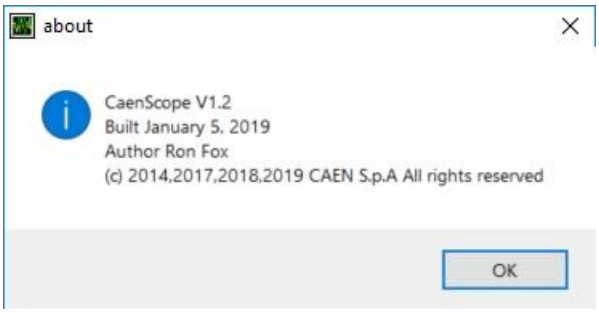


Fig. 4.4: the "About"pop-up dialog box

Acquisition System Set Up

To select a digitizer, left click on the "**File**" command in the menu bar and select "**Connect**" by left clicking it in the pull-down menu.

The following dialog window will appear:

Fig. 4.5: Digitizer to Host Computer Link USB (default) selection and connection prompt box

Select the connection type from the drop-down window on the left.

To use the **USB link**, select "usb" from the Connection pull down menu. The dialog box will appear as in **Fig. 4.5**. Then, type the Link address of the digitizer ("0" is the default value). For VME digitizers, type the board's VME base address in the VME Base Address Box only if you're accessing the digitizer through the V1718/V3718 CAEN Bridge. Please, refer to the digitizer's User Manual to know how to retrieve the VME base address information. Note that the value type will be interpreted as a hexadecimal value and the User does not have to prepend "0x".

To use the **Optical link**, select "Optical" in the Connection pull down menu. The dialog box will appear as in **Fig. 4.6**.

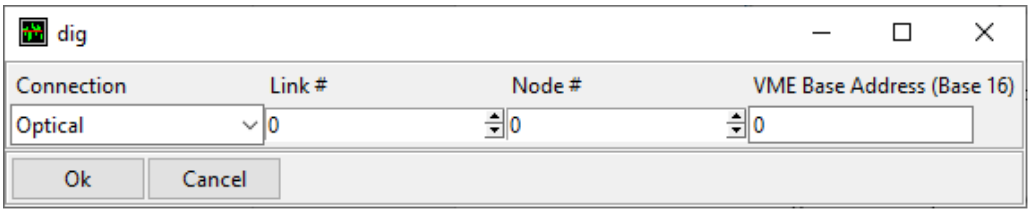


Fig. 4.6: Digitizer to Host Computer Link Optical Link selection and connection prompt box

Type the CONET link number to which the digitizer or VME bus bridge is attached ("0" is the default value). Set the Node Address ("0" is the default value) of the digitizer or bus bridge on the CONET daisy-chain. For VME digitizers only, type the board's VME base address in the VME Base Address Box only if you're accessing the digitizer through the V2718/V3718 CAEN Bridge. Please, refer to the digitizer's User Manual to know how to retrieve the VME base address information. Note that the typed value will be interpreted as an hexadecimal value and the User should not prepend "0x".

To use the connection via an **A4818** USB 3.0 to CONET adapter, several options are available and are listed in **Tab. 4.1** together with the corresponding type of hardware connections.

Connection Option	Type of Hardware Connection
USB_A4818	PC –> **USB** –> A4818 –> **CONET** –> DT57xx/N67xx/V17xx
USB_A4818_V2718	PC –> **USB** –> A4818 –> **CONET** –> V2718 –> **VME** –> V17xx
USB_A4818_V3718	PC –> **USB** –> A4818 –> **CONET** –> V3718 –> **VME** –> V17xx
USB_A4818_V4718 (**COMING SOON**)	PC –> **USB** –> A4818 –> **CONET** –> V4718 –> **VME** –> V17xx

Tab. 4.1: Available Connections involving A4818 link adapter

When selecting one of the options listed for an A4818 connection, the "Link #" field is substituted by the "PID" field. For this parameter, enter the unique A4818 Product Identification number (>10000), which is present in a label affixed to the A4818 adapter itself. In **Fig. 4.7,** an example of connection window when selecting one of the available connections via the A4818 link adapter is shown.

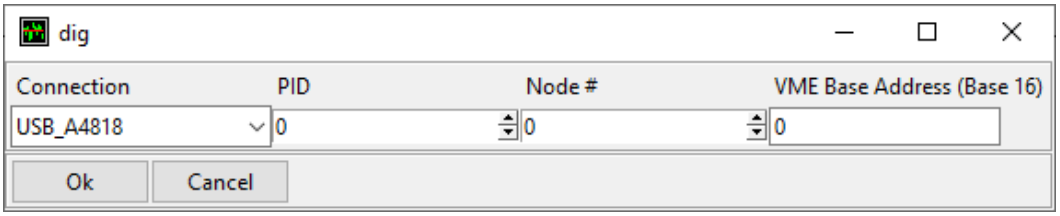


Fig. 4.7: Digitizer to Host Computer Link via A4818 selection and connection prompt box

To disconnect the unit, click on the "Disconnect" item in the File pull down menu. A pop-up dialog box will appear as in **Fig. 4.8.**

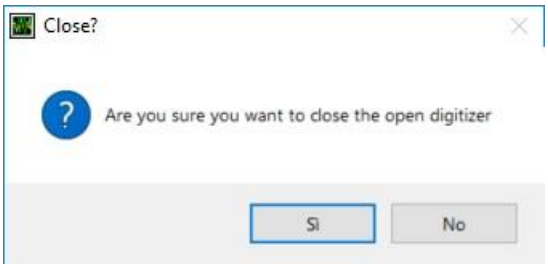


Fig. 4.8: Digitizer Disconnection Dialog Box

Click on the "Yes" key to disconnect the unit. Click on the "No" key to abort the disconnection operation.

Once selected and set the link, click on the button Ok. The CAENSCOPE display will appear as in **Fig. 4.9**, where, by default, the channel Ch0 is enabled. At any time, clicking the Cancel button will cancel this operation.

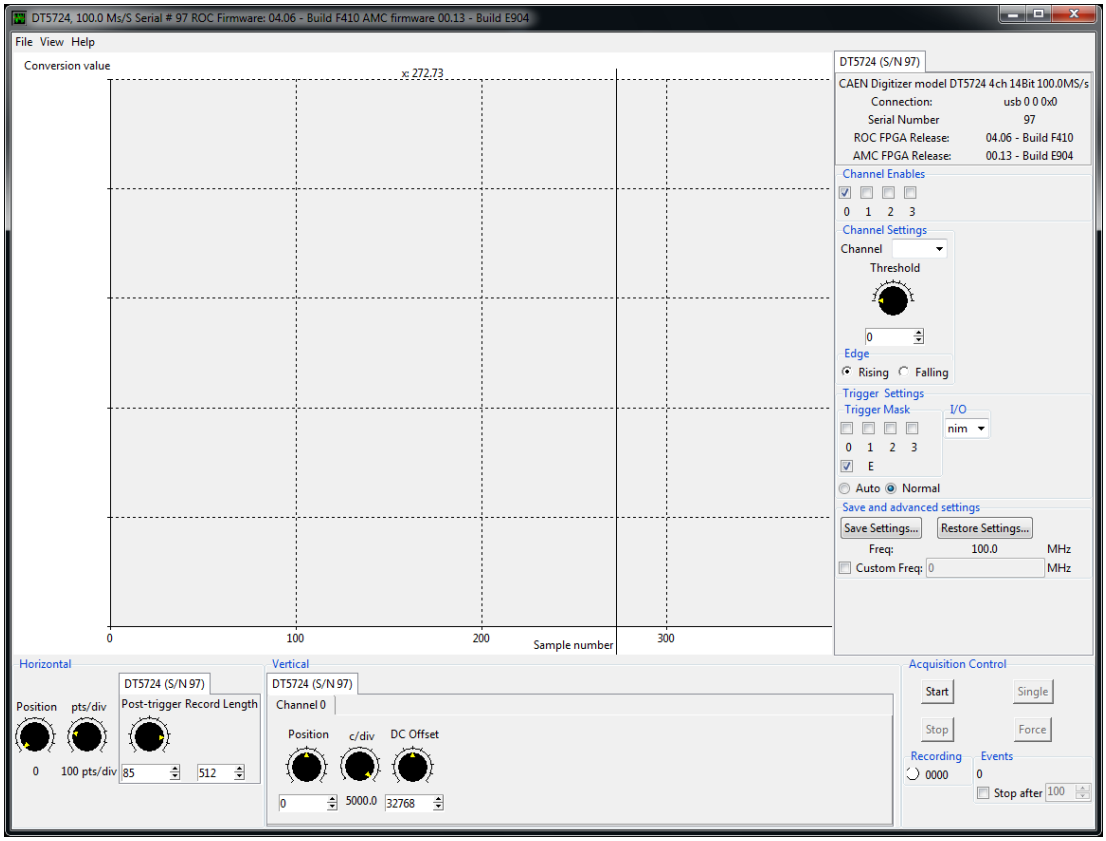


Fig. 4.9: CAENSCOPEMain Screen after connected to a digitizer

Digitizer Info Section

On the top right corner of the screen, a tab is shown. This tab displays the connected digitizer Model Number and Serial Number, along with a text box.

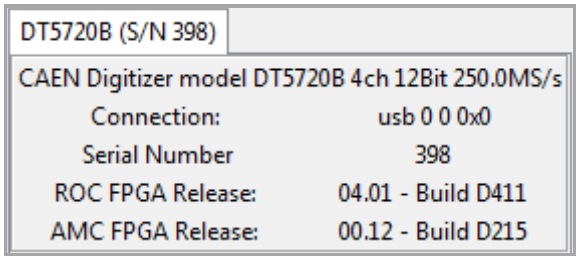


Fig. 4.10: Digitizer info section

The first line is a short description of the digitizer features (number of input channels, resolution and sampling rate). The second line of the box shows the connection type and the other connection parameters. The third line shows the digitizer serial number. The fourth line of the box shows the ROC FPGA release and build date (e.g. D411 stands for April 11th, 2014). The fifth line of the box shows the AMC FPGA release and build date (e.g. D215 stands for February 15th, 2014).

5 Commands

The Knob, Entry & Up/Down Controls

Many of the controls use a knob widget that looks like the picture below:

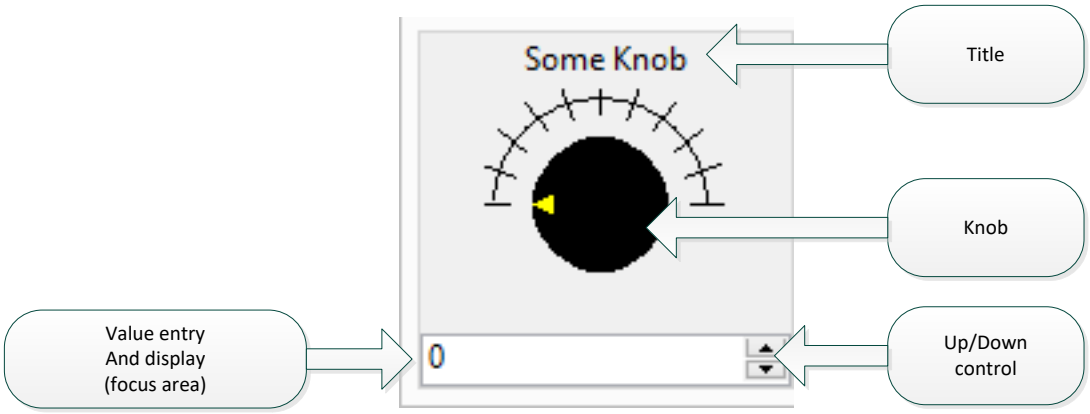


Fig. 5.1: Control knob widget for continuous variables

The knob **Title** describes the function being controlled.

To activate a control, left click on the focus area (**Value entry And display** rectangle in **Fig. 5.1**) of the control you want to operate.

To quit operating the last activated control, left click on a different control focus area or mark or unmark any check box.

The **Knob** can be fast adjusted by holding down the left mouse button and moving the pointer inside the rectangle that holds the control. The pointer on the knob (yellow triangle in **Fig. 5.1**) will point towards the mouse pointer. A finer control can be obtained by using the mouse wheel. Rotating the mouse wheel forward will turn the knob clockwise and rotating the mouse wheel backwards will turn the knob counterclockwise.

The knob for continuous variables features also:

- The **Up/Down control** can be clicked on to increment or decrement the controlled value by a preset amount. This can also be accomplished via the up/down arrow keys on the keyboard, or by rotating the mouse wheel. Rotating the mouse wheel forward will increase the control value and rotating the mouse wheel backwards will decrease it.
- The **Value entry And display** part of the control displays the current value selected by the knob. You can also type a new value and move the knob to that value by hitting the “Enter” key, or by shifting keyboard focus out of the entry widget (quitting the control).

The knob for discrete variables (see **Fig. 5.2**) is neither provided with Up/Down control nor Value entry And display, but it can only go to the position of its tick-marks. The actual value of the relevant parameter is displayed accordingly in a dynamic label placed underneath the knob. With the keyboard focus active, the up and down arrow keys can increase/decrease the knob value by one tick mark.

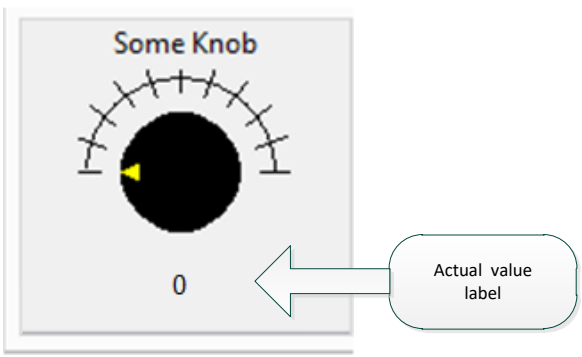


Fig. 5.2: Control knob widget for discrete variables

6 Horizontal Section

This section is always present on the CAENSCOPE display once a digitizer unit has been connected to the host computer.

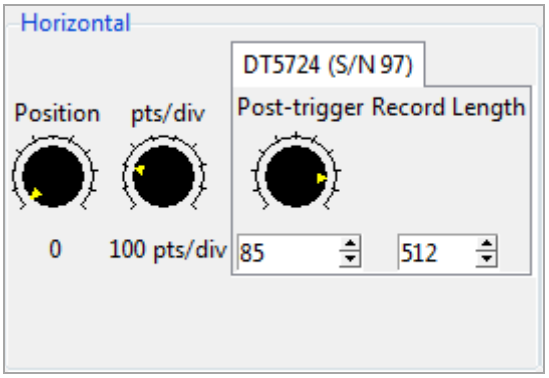


Fig. 6.1: Horizontal settings section

X-Axis Display Options

Horizontal Position (Graphical)

The **Position** knob is a display feature which allows the current trace to be scrolled horizontally to the left or to the right on the grid. The **Position** graphical function can be operated by rotating the relevant knob. This control ranges from "0" up to "10" in steps of 1, i.e. 500 points. This is a pure software scroll.

Horizontal Scale (Zoom)

This knob controls the time expansion of the traces. It is a pure software zoom.

Without the X-axis calibration: the time base is scaled in pts/div, indicating the sampled points/division. The left most position of the knob represents a 10 points/div scale value. Taking the 724 digitizer series as reference, the scale ranges from 10 to 5000 pts/div in 10 steps.

With the X-axis calibration: the time base is scaled in time-unit/division. Taking the 724 digitizer family as reference, the horizontal scale can be selected with the following values: 1,5,10,50,100,500 ns/div, 1,5,10,50,100,500 µs/div, 1 ms/div.

Post-trigger

The **Post-trigger** setting controls the hardware post-trigger value, which is common to all enabled channels of a digitizer. The Post-trigger is set as the percentage of the waveform (i.e. % of the total **Record Length**) the User wants to capture after the trigger point. This is done by delaying the end of a trigger's acquisition window from the trigger time. The delay corresponds to the post-trigger value, i.e. the percentage of the full waveform recorded after the trigger point. If the **Post-Trigger** is = 0, then the waveform is acquired till the trigger point occurs. If the **Post-trigger** is = 100, then the point sampled at Trigger time is the first point of the record.

This control ranges from 0 (%) to 100 (%) of **Record Length** and can be set in a fine way through the relevant knob or typing the value in the text box (in steps of 1 % of the **Record Length**). The up/down arrows next to the text box are in steps of 5 %. The post-trigger function works on both predefined and custom defined record lengths.

Record Length

The **Record Length** Value Box entry controls the hardware record length setting, which is common to all channels of a digitizer. This determines the number of samples each digitizer channel acquire for each trigger. The up/down arrows move between the pre-defined record lengths defined for the digitizer in its buffer organization register (refer to the relevant digitizer User Manual for details). The User can, however, use a custom record length by simply typing a new value in the entry to the left of the up/down arrows and confirming the entry with the "Enter" keyboard command. If the Value Box entry has input focus, the up/down keyboard arrows act as alternative to the up/down arrows on the right of the Value Box. The minimum and maximum record length depend, in general, on the specific digitizer model and on the amount of buffer memory it has.

7 Vertical/Acquisition Section

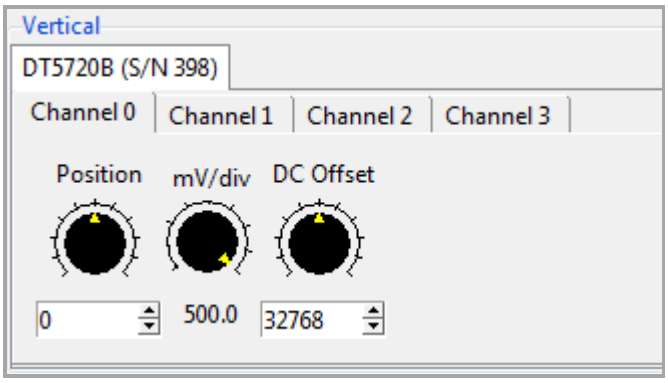


Fig. 7.1: Vertical Settings Section

Y-Axis Display Options

To enable the vertical section, one or more of the "Channel Enables" checkboxes in the Trigger section must be checked (see **Fig. 8.2**). When a "Channel Enables" checkbox is selected, it adds a Tab to the vertical section labeled with the corresponding enabled channel number. Each tab allows the Vertical Position, Scale (Vertical Zoom) and DC Offset to be controlled for the specified channel.

Vertical Position (Graphical)

The **Vertical Position** knob/entry adds a constant software offset to the vertical position of a channel trace. This allows for positioning traces vertically without any risk of truncating the signal (as it might happen if the **DC Offset** control is used). Vertical position range goes from -100 to +100 in steps of 5 (arbitrary unit).

Vertical Scale (Zoom)

The **Vertical Scale** knob/entry changes the effective volts or counts per vertical division of the trace. This is a purely software expansion. This parameter ranges from 0.5 to 5000 c/division (with **Calibrated Axis** disabled) or from 0.05 to 500 mV/division (with **Calibrated Axis** enabled).

DC Offset

This control affects the hardware. Each channel of the digitizer is provided with a 16-bit DAC that sets a DC offset for that channel (refer to the digitizer User Manual for a more detailed description). This knob adjusts the DC offset added to the channel being set (range: 0 up to 65535 counts).

8 Channel & Trigger Panel

This Panel includes a section allowing the User to select which channels will acquire waveforms as a result of a trigger (*Channel Enables*). The trigger threshold for each channel can be set by selecting it from the drop-down box in the "Channel Settings" section of the panel and adjusting the threshold knob. The Edge radio buttons determine if the trigger threshold is for a rising or falling signal.

The trigger setting mask checkboxes allow the User to select which channels participate in the trigger. The E checkbox allows the external trigger to participate in the trigger. The I/O section of the panel sets the logic level of the digitizer's GPIO connectors. The Auto/Normal radio buttons allow for a periodic software trigger (Auto) or purely hardware triggers (Normal).

The Save Settings and Restore Settings buttons allow the User to save the current digitizer settings to file and later restore them to the digitizer. The information about the nominal sampling frequency of the digitizer is also reported and a dedicated section is provided, where the User can type the sampling frequency value to be used exclusively for the X-axis calibration in case the hardware has been programmed to work at a customized frequency value (NOT YET IMPLEMENTED).

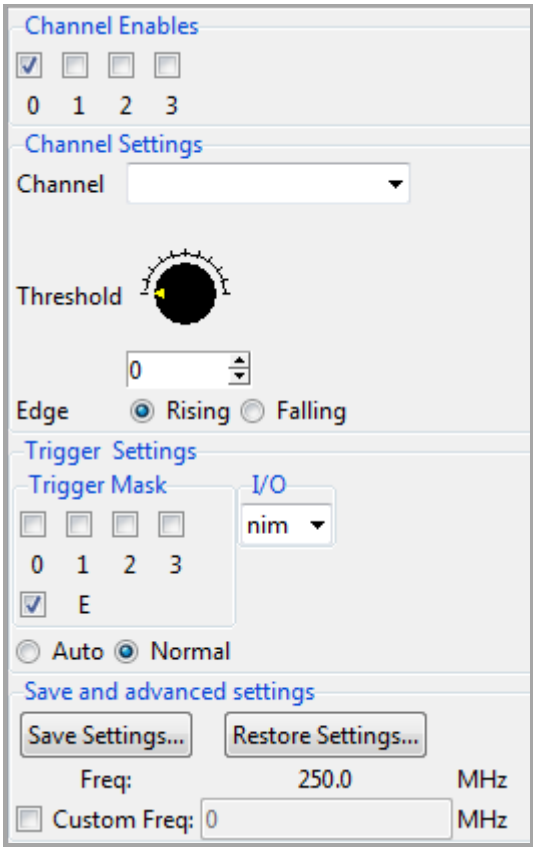


Fig. 8.1: Channel and Trigger Settings sections

Channel Enables Section

The "**Channel Enables"** Section includes as many checkboxes as the available acquisition channels of the digitizer. By marking a channel check box, the corresponding channel will be enabled to record data. On the display, each enabled channel trace is assigned a different identifying color.

Channel (Normal Trigger) Settings

This Section is part of the Trigger Section. It is operative only when the Normal Trigger operation is selected in the Trigger Settings section (see below) and is related to the Trigger Mask enabled channels.

When Trigger is set to Normal, Trigger conditions can be set for each triggering channel. To modify the trigger settings for a channel enabled in the trigger mask, the User can select that channel from the Channel pull down menu. Two parameters can be set, the trigger edge and the trigger threshold.

Edge Setting

The User can select the triggering condition to be on the “Rising“ or “Falling” edge of the signal, by selecting the correspondent button above the channel check boxes line. The Edge setting is common to all channels. This setting determines whether the trigger occurs as the signal rises above the threshold value (rising) or drops below it (falling).

Threshold Setting

In combination with the Edge setting, the threshold determines when a channel triggers memorization of a waveform. This value can be modified by left clicking in the Threshold entry box or on the knob widget. The value can then be typed in the Threshold Entry box, or adjusted thru the knob widget, or increased and decreased with the up/down arrows or with the mouse wheel.

The trigger channel being set is the one selected in the pull-down menu.

A small caret will show the trigger level for each channel on the plot display. The caret will have the same color as the color assigned to display that channel.

Note: For digitizers whose trigger management is based on groups of channels (such as 740 series), the Threshold is common to all the channel in the group.

Trigger Settings

The “**Trigger Settings**” section allows the User to:

- Select which channels can trigger data taking.
- Enable/disable the External trigger.
- Select the type of logic signal (NIM or TTL) accepted as External Trigger (on the front panel relevant connector) or presented at the Trigger output of the digitizer (on the front panel relevant connector).
- Select the trigger mode *Auto* or *Normal*.

Trigger Sources

A row of check boxes numbered according to the digitizer channels allows the User to select which corresponding acquisition channel will be the trigger source. A check box marked “*E*” is available for the External Trigger selection. Each checked channel is available in the channel settings pull-down menu.

Trigger Mask

The Trigger Channels are enabled by marking the desired channel check box and disabled by unmarking it. Every enabled Trigger Channel contributes as an OR source to the Digitizer Trigger Mask, i.e. the digitizer will be triggered by the trigger signal generated by the first of the enabled trigger channels receiving a valid trigger event.

The channels participating in the trigger will be shown in the upper left of the plot display in the form > SN(*serial*).*ch* where *serial* is the digitizer’s serial number and *ch* identifies the channel. A unique color is assigned to each channel that participates in the trigger. Carets showing the trigger thresholds of each channel participating in the trigger are displayed to the left of the Y axis of the plot.

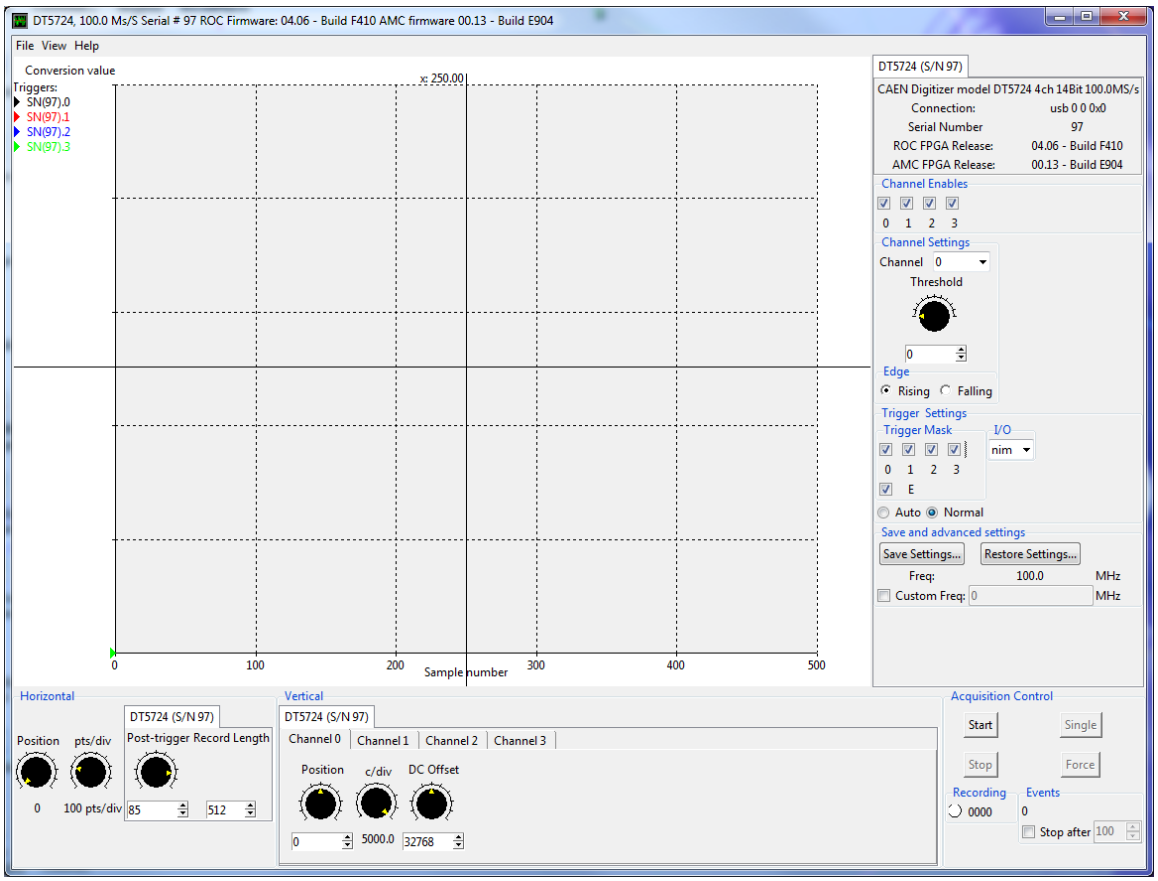


Fig. 8.2: CAENSCOPE Main Screen with Channel Enabled and Trigger Mask Enabled Channel

Trigger Type

- **Normal:** In Normal mode, only an OR of the hardware triggers initiates the recording of an event.
- **Auto:** In Auto mode, when the digitizer is running, periodic software triggers are provided if no hardware triggers are received. This allows the User to observe the baseline level of the digitizer so that the DC offset and trigger level can be better set.
- **External:** The selection of the "External" (E) Trigger check box ORs the External Trigger signal received at the TRG IN input of the digitizer into the trigger condition. The accepted External Trigger Logic levels can be selected as NIM or TTL in the "*I/O*" pull down menu.

Save and Advanced Settings

In this section, by clicking on the "Save Settings…" button, the User can select a file path to save the current settings of the Acquisition & Trigger configuration to an XML file.

By clicking on the "Restore Settings…" button, the User can recall a previously recorded setting file in XML format to the Acquisition and Trigger configuration.

A CAENSCOPE Settings XML file example is shown below:

```
<?xml version="1.0"?>
<caendigitizer>
<digitizer serial="19" version="17" family="xx724" id="9D80864ED42BCEE672B65F496FAC53CA">
<channels value="4"/>
<resolution bits="14"/>
<frequency hz="100000000.0"/>
<maxsamples maxsamples="524288"/>
<channelgroups capable="0"/>
```

```
<zerosuppression capable="0"/>
<inspection capable="0"/>
<dualedge capable="1"/>
<voltagerange hi="5.0" low="-5.0"/>
<windows>
<window size="256"/>
<window size="512"/>
<window size="1024"/>
<window size="2048"/>
<window size="4096"/>
<window size="8192"/>
<window size="16384"/>
<window size="32768"/>
<window size="65536"/>
<window size="131072"/>
<window size="262144"/>
</windows>
</digitizer>
<settings id="2" digitizer="9D80864ED42BCEE672B65F496FAC53CA">
<dcoffsets>
<dcoffset value="32768" channel="0"/>
<dcoffset value="32768" channel="1"/>
<dcoffset value="32768" channel="2"/>
<dcoffset value="32768" channel="3"/>
</dcoffsets>
<trigger external="disabled" mask="12" direction="rising">
<level value="9091" channel="0"/>
<level value="8250" channel="1"/>
<level value="2150" channel="2"/>
<level value="360" channel="3"/>
</trigger>
<window size="1024"/>
<posttrigger value="34.0%"/>
<channels mask="3"/>
</settings>
</caendigitizer>
```

Acquisition Control Settings

The "**Acquisition Control**" section is located at the bottom of the CAENSCOPE screen, to the right of the Horizontal and Vertical Sections. It contains the following four keys:

- **"Start"**: this button starts the data taking in the mode selected in the Trigger Settings tab.
- **"Stop"**: this button stops the data acquisition on all channels.
- "**Single"**: this trigger button takes the digitizer out of data taking mode after the next trigger. If the digitizer is already stopped, it is started and again runs until the next trigger occurs.

- **"Force"**: this key performs a single software trigger of the module. This function is useful to help the User to locate the trace when the trigger conditions have not yet been set properly.

At the bottom left of the section there is a "Recording" status LED indicator. The LED turns green when the Recording function is enabled (see Sec. **Data Recording**). In this condition, every valid event will be recorded to the currently selected record file.

To the right of the LED an event counter indicates the number of waveforms that were recorded to the Record file since the "Record" was given in the "File" pulldown menu. The LED stays green indicating that all incoming records will be written to the selected file until the "Close" command will be clicked in the "File" pull-down menu. The file event counter will remain frozen to its last value upon recording closure. It will be reset to zero if the User changes the destination Record File name or if the User exits the CAENSCOPE program. To the right of the Recording Event Counter a "current acquisition" counter indicates the number of waveforms (events) that were recorded since the last "Start" command was given. Starting the acquisition resets the current acquisition event counter to 0.

A check box marked "Stop after" allows the User (by checking it) to enable a data acquisition stop condition. This is possible by setting the number of events to be recorded before stopping the acquisition in the "Stop after" text box (press Enter on keyboard to activate the typed value). Note that, due to the fact that several events are read from the digitizer at a time, the User may get more events than those selected via the "Stop After" parameter.

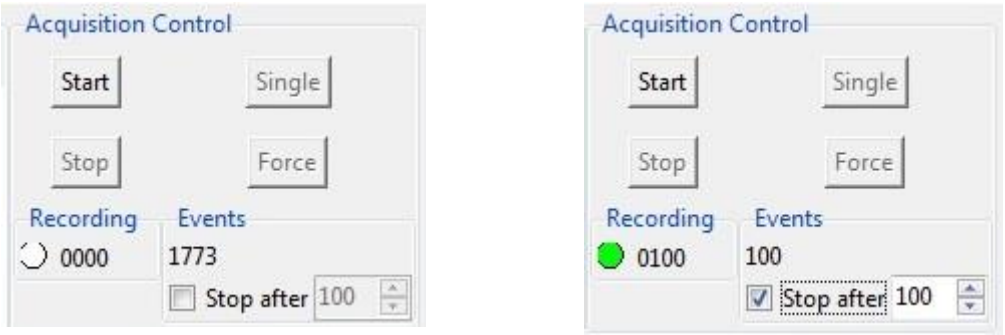


Fig. 8.3: Acquisition Control section and active event-driven Recording

9 Data Acquisition and Recording

Data Acquisition

To understand the Data Acquisition and Recording procedures, it is important to remember that the internal digital memory of each channel of the digitizer operates as a circular buffer. To display a waveform on the screen, the selected number of acquisition points is read and displayed, then the data are overwritten upon the successive trigger by the next waveform sampling. Thus, the waveforms displayed on the screen are not recorded to a file unless the Data Recording function is enabled.

While all hardware channels are digitizing data upon a valid trigger reception, only the Enabled Channels Data are sent to the CAENSCOPE program and made available for display and for recording.

Thus, if data must be recorded, a Record File must be created or enabled thru the process indicated into the next section.

Data plotted are a sample of the data acquired. The formula for determining if an event is plotted is as follows: events are plotted if at least 30 ms have passed since the last plot but, if the time to plot the last set of traces was longer than 30 ms, 2x the time required to plot must pass before the next event is plotted.

Data Recording

In order to start recording events, select one of the Record options from the File menu as shown in **Fig. 9.1**. Record (text) starts recording data in an XML file (.xml file type). Record (binary-raw) starts recording data in a simple binary formatted file (.bin file type). Record (binary-sqlite) starts recording data into an Sqlite3 database. Data from all enabled channels are recorded. Metadata describing digitizer settings are recorded as well and the event data can be correlated to the settings active at the time each event was acquired. Recording into a file will continue whenever data acquisition is active until either another Record selection is made from the File menu, or the File->Close menu item is selected. When one of the "Record" options is selected, the "Recording" LED turns on green.

The "Close" option in the "File" pull-down menu allows then to disable the Data Recording setting ("Recording" LED lights off) and closes the generated output file making it available to be read.

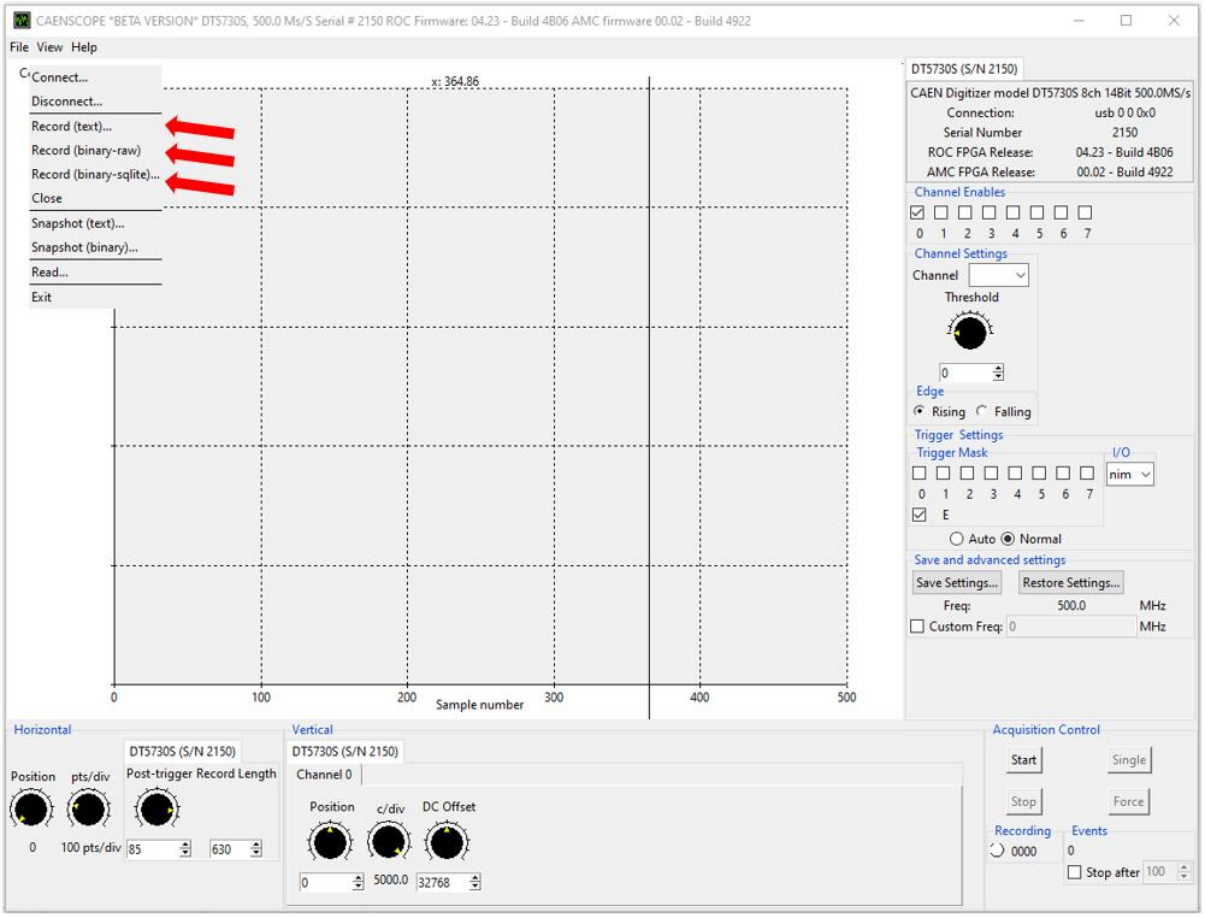


Fig. 9.1: CAENSCOPE Recording activation

SQLite Data Base Recording

In the Menu Bar, select “File”, then “Record (binary - sqlite)” as indicated in **Fig. 9.1**. A Windows Directory window will open, allowing the User to create an SQLite .db record file into the desired folder. By creating the file, the data recording is enabled. Events will be appended to the file as they are recorded by the digitizer and numbered sequentially. Simultaneous event records from different channels will be tagged with the same sequential number, in each channel.

Start the data acquisition with the “Start” button in the Acquisition Control section. If the .db file is created while the acquisition is already running, the CAENSCOPE will start recording already from the first available complete event record successive to the file creation.

Stop the acquisition by using the “Stop” key in the Acquisition Control section.

The display will show the last current signals recorded as in the picture below:

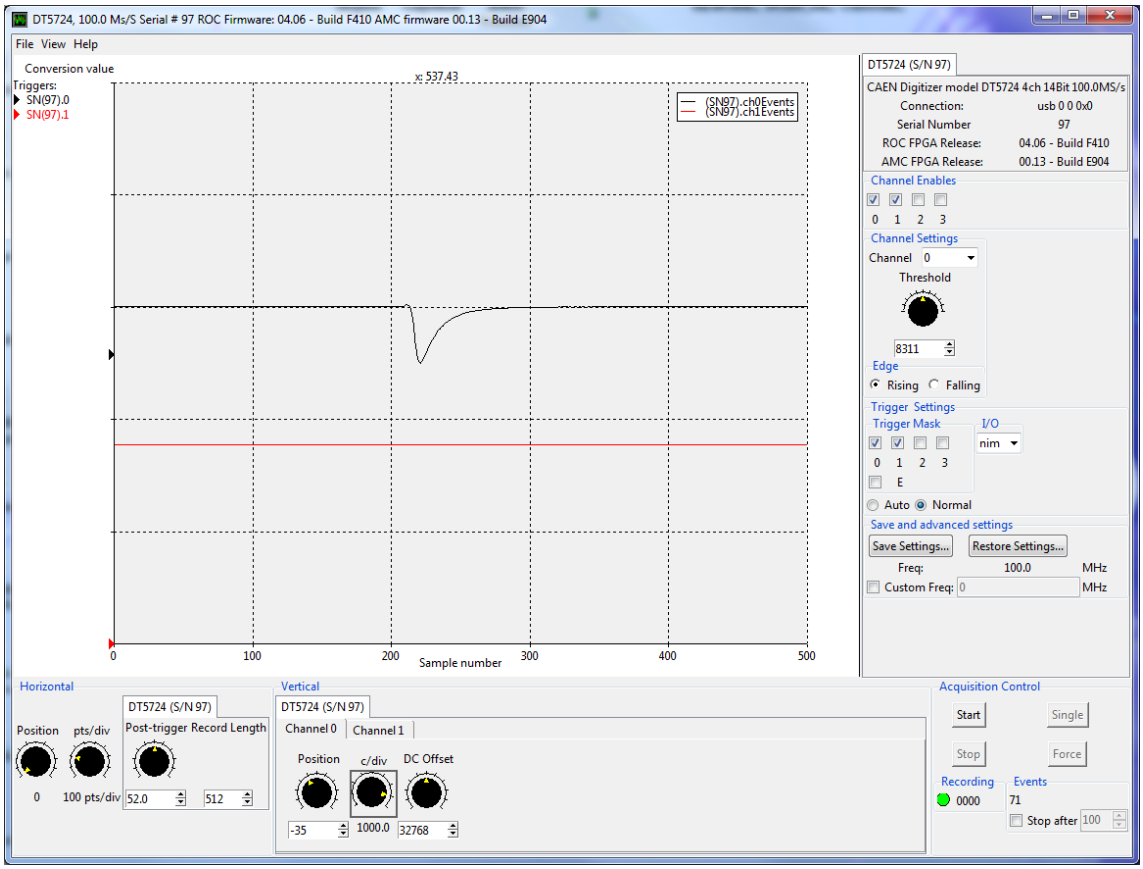


Fig. 9.2: CAENSCOPE Main Screen with live signals on CH0 and CH1

XML File Recording

The Menu Bar “File” -> “Record (text)” is used to initiate recording into an XML file. The User will be prompted for a file into which data will be recorded. Unlike database files, an existing file will be overwritten by new data. Data are recorded exactly as described in Sec. **SQLite Data Base Recording**.

Binary File Recording

CAENScope raw binary format supports high speed output of trace and metadata. The output files written in this way can be read back in with CAENScope or read by a custom user-written application. In Chap. **13**, the C++ library used in the CAENScope software for raw binary file processing is described, while the code containing it is present in the CAENScope folder and can be used in custom user applications.

In the Menu Bar, select “File”, then “Record (binary-raw)” as indicated in **Fig. 9.1**. The User will then be prompted for a file into which record data. The mechanics of recording data are identical to that described in Sec. **SQLite Data Base Recording**.

Playing Back Recorded Data

From the Menu Bar, select the “File” item, then the “Read” item. The user is then prompted for a recorded data file to play back. This data file can be any of the data file types that can be recorded (.db, .xml or .bin). Note that XML and raw binary files will be converted by CAENScope to a temporary database file first. A Recorded Events File Tab will appear to the right upper corner of the screen. Click on the tab to access the recorded data display as in **Fig. 9.3**.

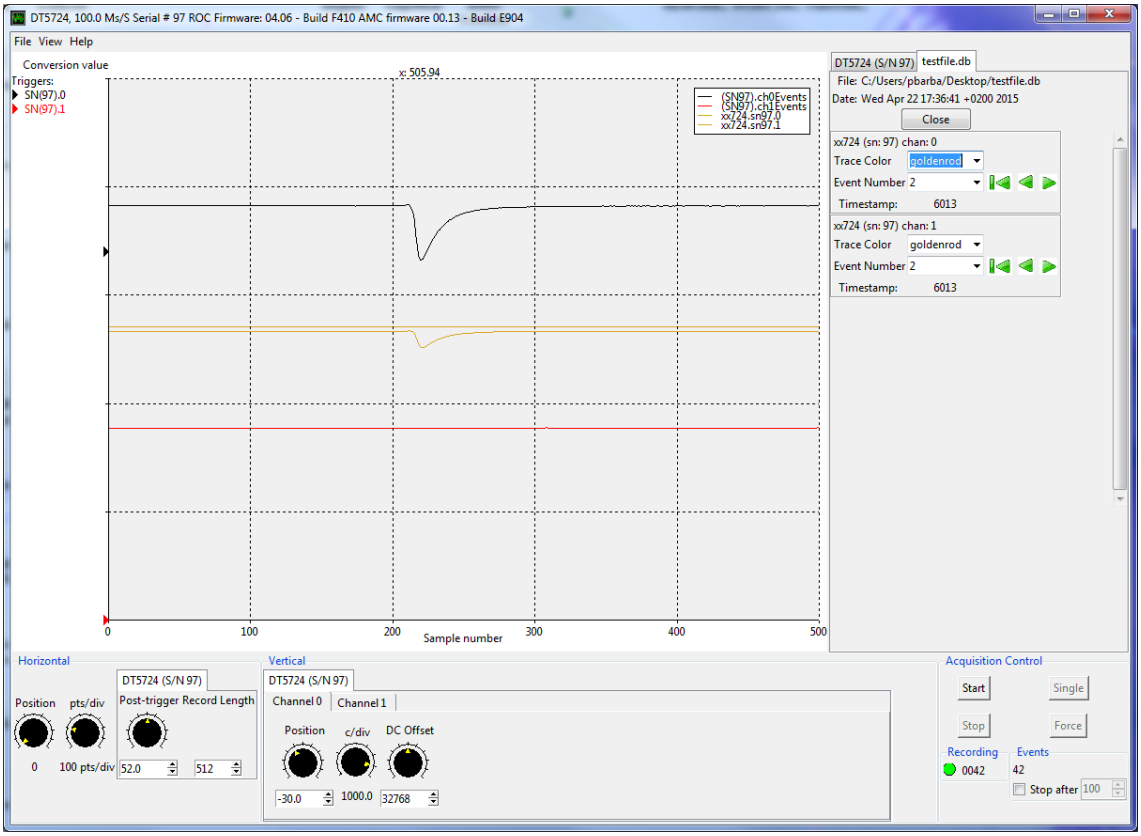


Fig. 9.3: CAENSCOPE Main Screen with recalled recorded signals from CH0 and CH1

The Recorded Events Tab displays the *filename.db* (or *.xml* or *.bin*) record path and the date the Acquisition File was created. It presents as many channel sections as the number of channels which were recorded simultaneously.

The User can scroll each channel independently and visualize the corresponding event on the display. Each section is tagged with the Digitizer Model Number, Serial Number and ID Number of the channel. Each channel trace is displayed with a trace color the user can select from the corresponding pull-down menu.

Events from different channels with the same event number have been recorded at the same trigger time.

By clicking on the Digitizer Tab, it is possible to restart the data acquisition while still displaying the selected recorded events (see **Fig. 9.4**).

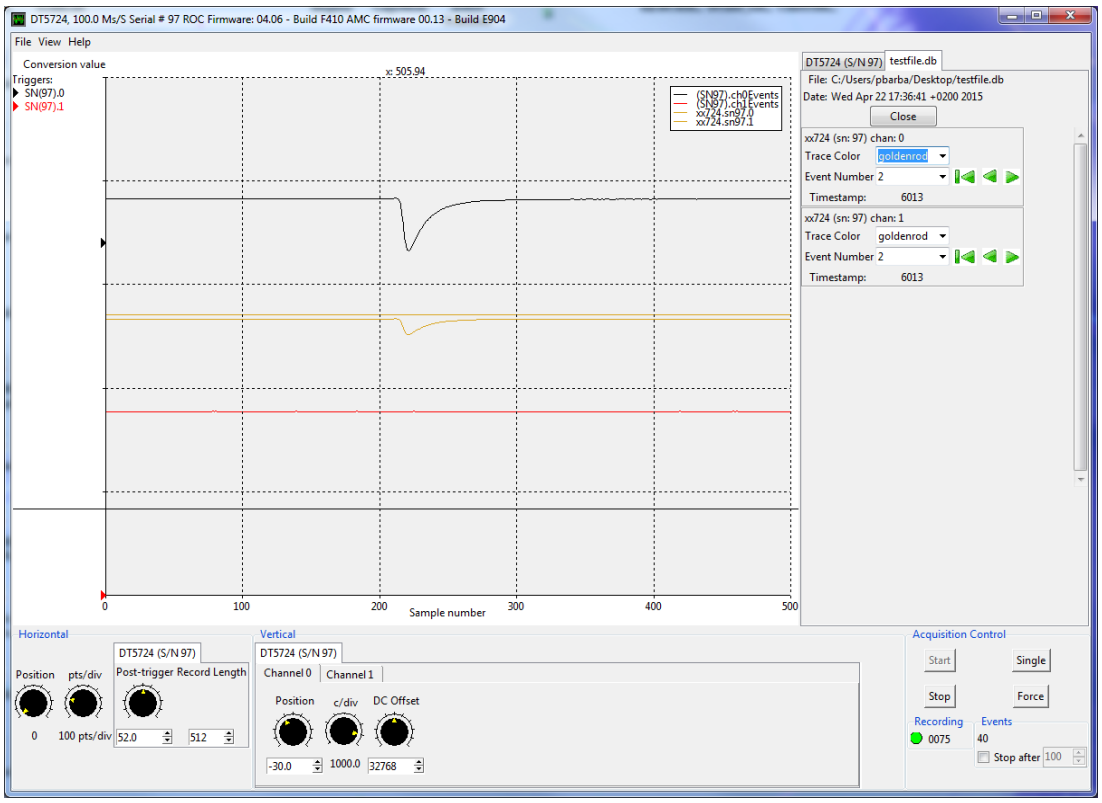


Fig. 9.4: CAENSCOPE Main Screen with live signals on CH0 and CH1 and selected recorded events from db file

The Recorded Event File can be read also when no digitizer is connected. Upon initialization of the CAENSCOPE program as in Chap. **4**, the User can recall the Event files by accessing the Menu Bar item “File” and then “Read” (see Fig. 9.5: **CAENSCOPE Main Screen with recorded events from db file with no digitizer connectedFig. 9.5**).

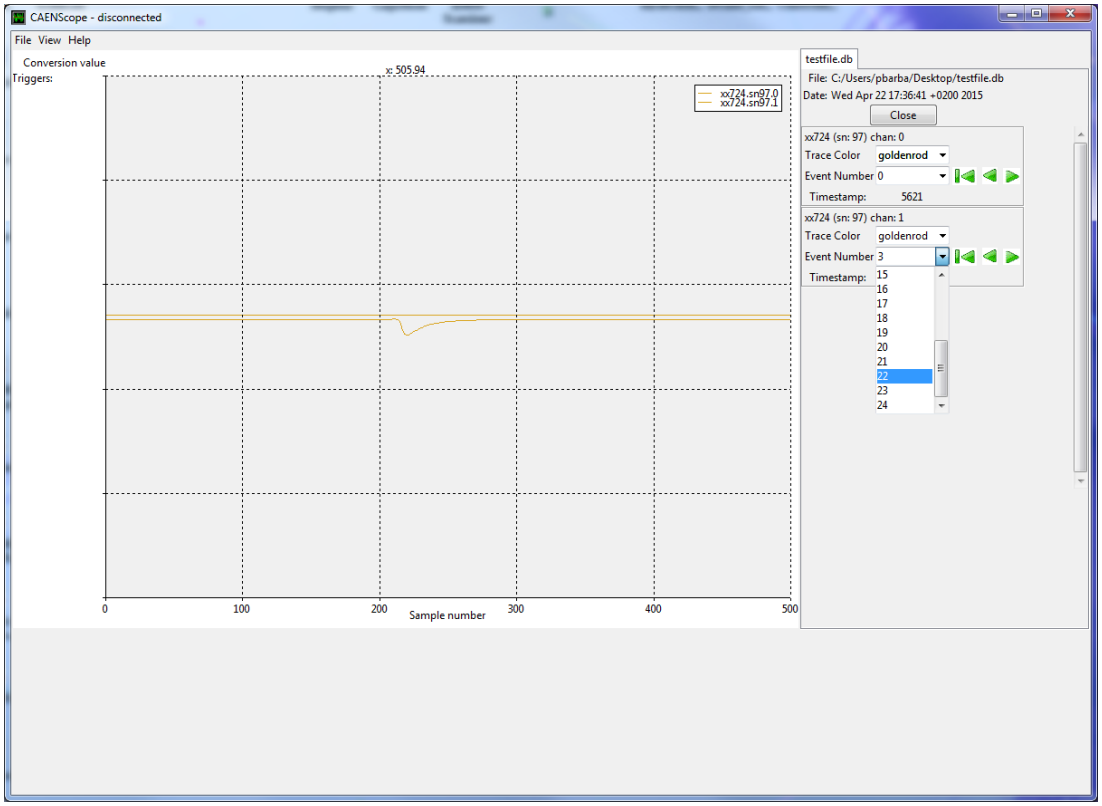


Fig. 9.5: CAENSCOPE Main Screen with recorded events from db file with no digitizer connected

Multiple Recorded Event Tabs can be recalled, both in Live and Recorded Sessions, each one will be identified by the *filename.db* (or *.bin* or *.xml*) indication on the corresponding Tab, as well as by the Date of the creation of the Record File.

To close the Recorded Events Tab, use the “Close” button.

10 Display Utilities

Cursors and Region of Interest (ROI)

The CAENSCOPE plot displays a cross hair cursor that can be moved by the mouse. The x-axis value is dynamically displayed at the top of the plot area (see **Fig. 10.1**).

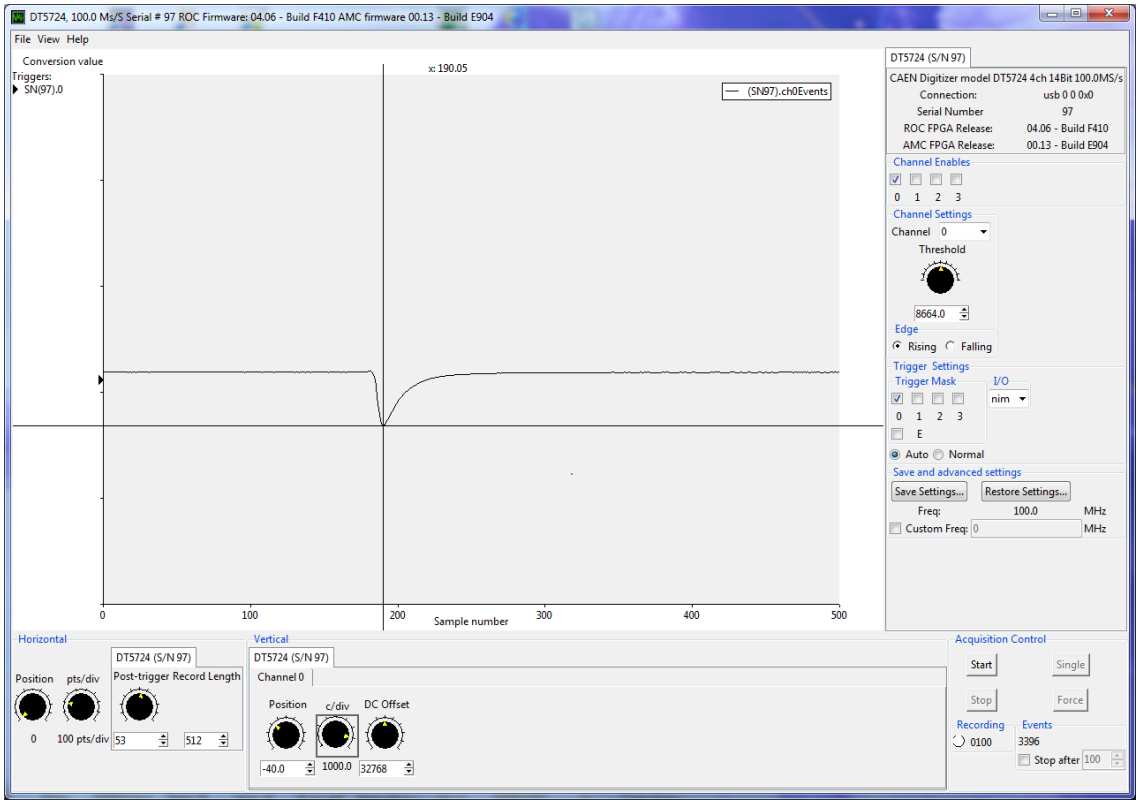


Fig. 10.1: CAENSCOPE plot cursors

It is further possible to select a Region Of Interest (ROI) on the plot. Just hold down the left mouse button and drag the pointer to define the desired ROI (see **Fig. 10.2**). At the top of the plot area, the ROI sample number range and the ROI width (in samples) is given. By right clicking, the y-axis value and range of the ROI are displayed for User convenience (see **Fig. 10.3**).

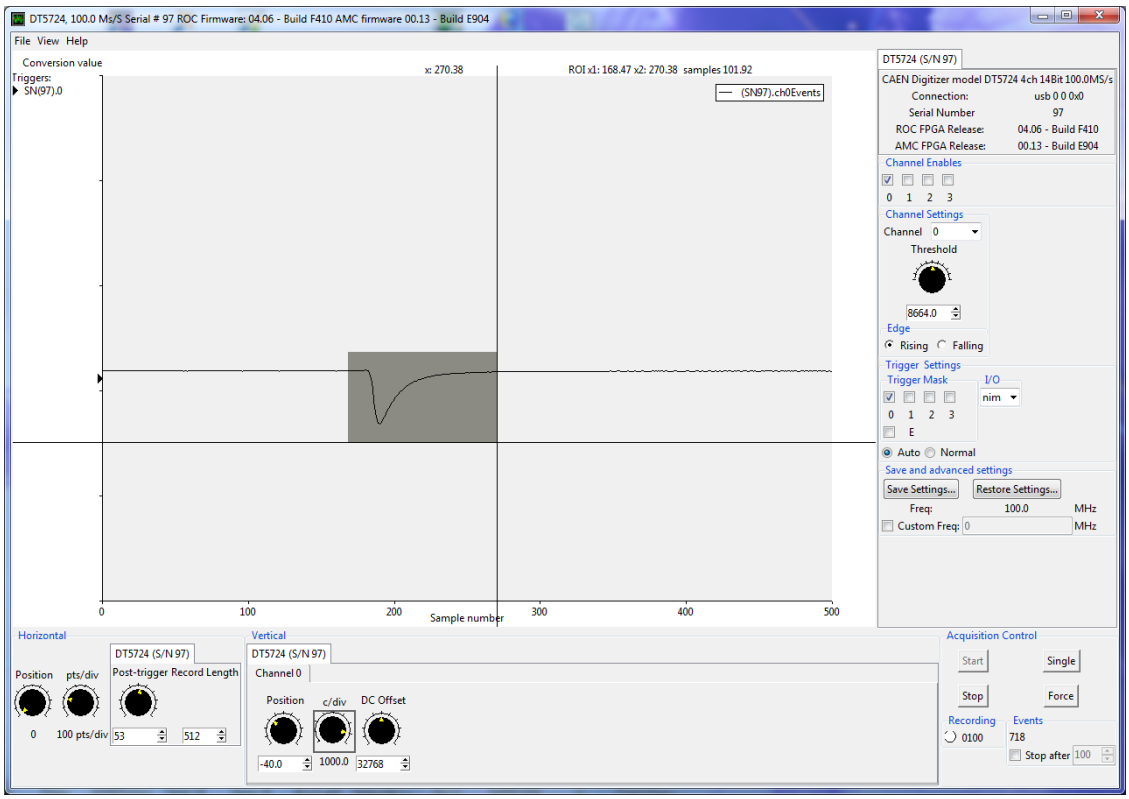


Fig. 10.2: CAENSCOPE ROI selection

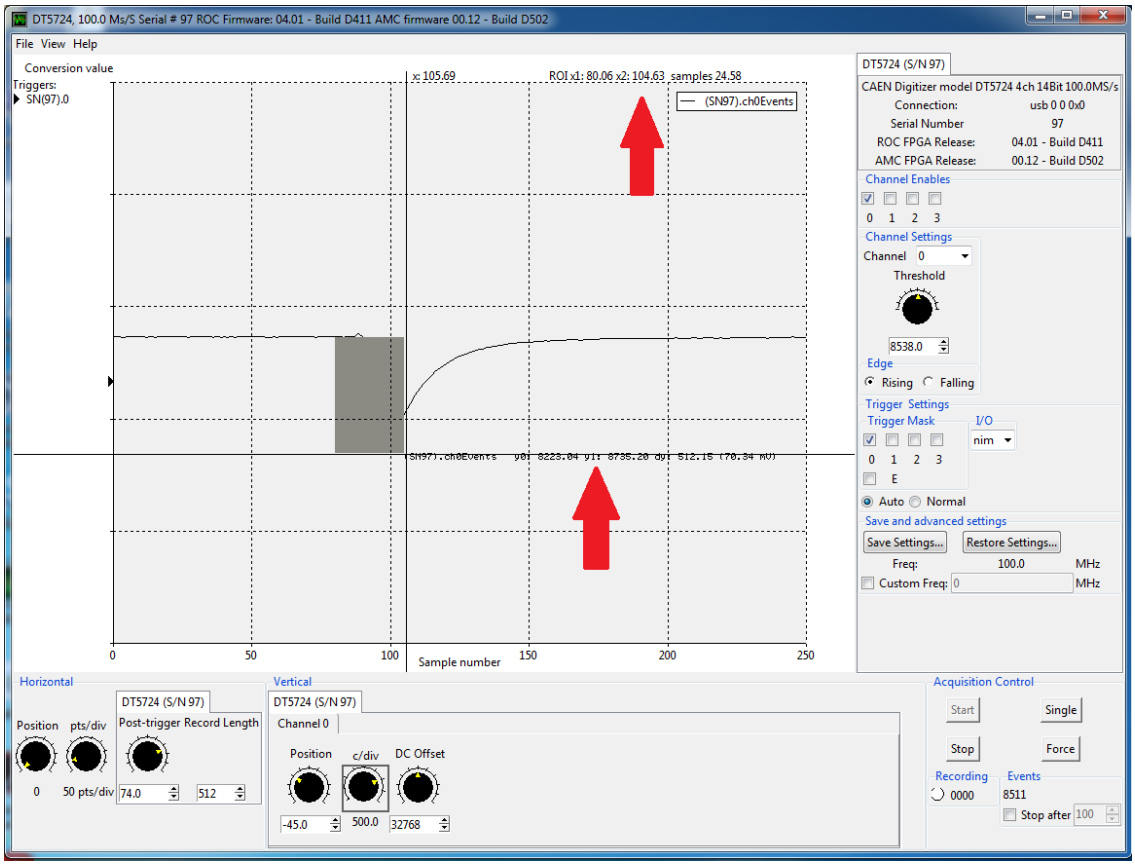


Fig. 10.3: CAENSCOPE ROI information

ROI selection is possible with traces read from an event file as well.

11 Exiting the CAENSCOPE

To Exit the CAENSCOPE, from the Menu Bar access the item “File” then the Item “Exit”. It is also possible to use the Standard close button in the upper right corner of the window (see **Fig. 11.1**).

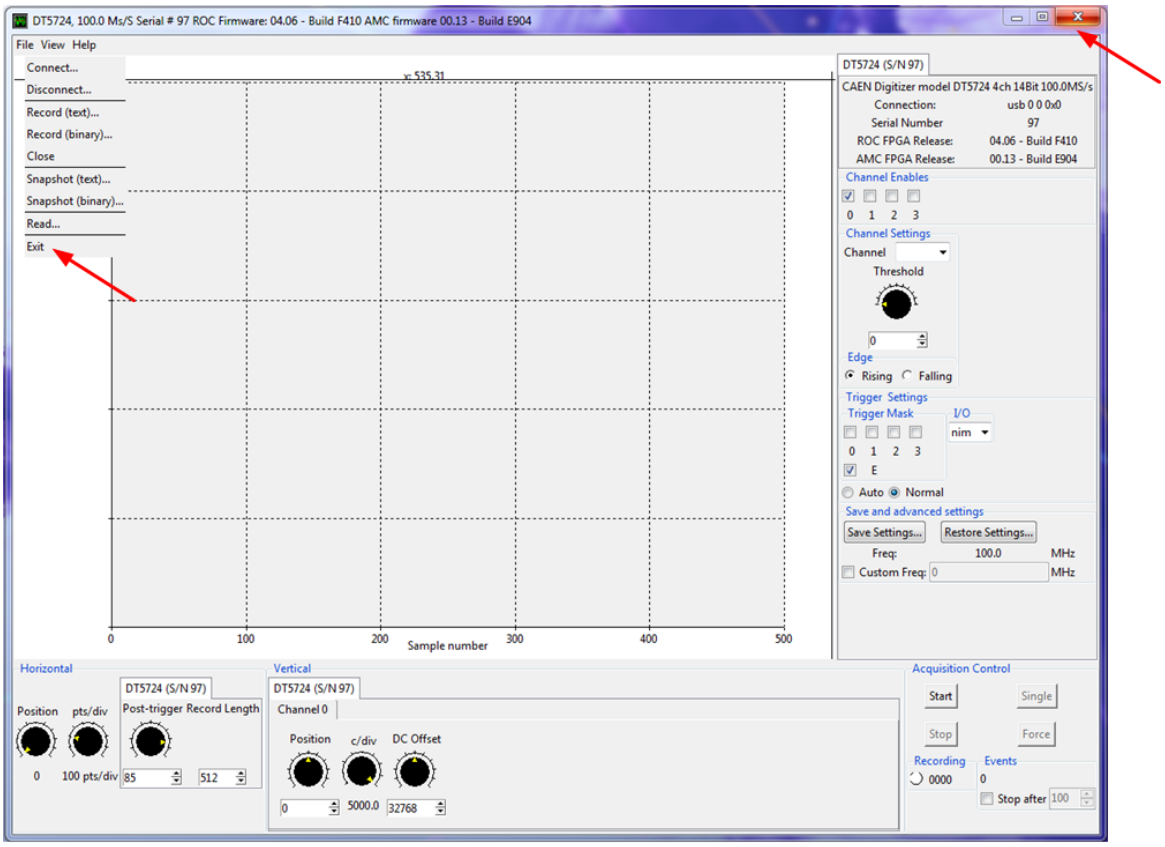


Fig. 11.1: CAENSCOPE Exit options

12 CAENSCOPE Output Files

CAENSCOPE Output File Formats

CAENSCOPE can produce several types of output files:

- **Saved settings files**
- **Text event log files**
- **Binary (Raw) event log files**
- **Binary (SQLite3) event log files**

This chapter describes the format of these files.

Saved Settings Files

These files capture the settings of a digitizer at an instant in time. These files are textual and are in XML format. At present, there is no DTD against which a validating parser can validate these files. The root tag of the document is <caendigitizers>. All data in this file are encapsulated in a single top level <caendigitizers> element.

Within the <caendigitizer> element there are two elements:

- <digitizer> which describes the digitizer and its capabilities
- <settings> which capture the settings of a digitizer at an instant in time.

The file can have several <digitizer> tags and several <settings> tags. The <settings> tags have an attribute **digitizer** that refers to the **id** attribute of the <digitizer> tag to which it applies. This format naturally allows for several digitizers each with several settings snapshots to coexist in a single settings file.

`<digitizer>`

The contents of a <digitizer> tag describe a digitizer and its capabilities. The tag itself has the following attributes:

- **id –** Identifies the digitizer. The **id** is chosen to be a hash that is unique and reproducible for a specific digitizer module.
- **family –** Identifies the family of the digitizer, e.g. x724 is used for the V1724, N6724 and the DT5724 modules.
- **Version –** Contains the board version number.
- **serial –** Contains the module serial number.

The body of the <digitizer> tag contains the following elements:

`<channels>`

This element has no body. It has the attribute value whose **value** is the number of input channels the digitizer has. Note that for the SCA digitizers this value does not count the TR0 input.

`<resolution>`

This element has no body. It has the attribute **bits** whose value is the number of bits of resolution the digitizer offers. For example: <resolution bits="12" /> means the digitizer sample values are in the range [0,4095].

`<frequency>`

This element has no body. It has the attribute **hz** whose value is the sampling frequency of the digitizer.

Note: in the future this may need to be expanded to correctly handle multi-sampling frequency devices like the SCA digitizers.

`<maxsamples>`

This element has no body. Its attribute **maxsamples** gives the maximum number of samples the digitizer can acquire in one event for one channel**.**

`<channelgroups>`

This element has no body. The attribute **capable** describes whether digitizer groups its channels or not. If 0, the digitizer does not group channels. If nonzero, this is the number of channels in a channel group.

<zerosuppression>

This element has no body. If the **capable** attribute is non-empty, the digitizer can return waveforms in a zero-suppressed format.

<inspection>

This element has no body. If the **capable** attribute is non-empty, the digitizer has an inspection output to which signals can be routed.

<dualedge>

Empty element whose **capable** attribute indicates whether the digitizer can be switched into the frequency doubling dual edge sampling mode or not.

<voltagerange>

Empty element with the following attributes:

- **Low,** the lowest input voltage accepted by the digitizer.
- **High,** the highest input voltage accepted by the digitizer.

Note that for CAENSCOPE, these values are input by the User for digitizers that have multiple input range options. This implies that these data are only as good as the User's input. Once entered for a digitizer, the value is memorized.

<windows>

This tag contains one or more `<window>` tags. Each of those has an attribute **size** whose value is the number of samples per channel in an event for the pre-defined buffer organizations**.**

<settings>

This tag contains elements that describe a digitizer setting at a point in time. The following attributes are present in the `<settings>` tag:

- **id –** a unique setting id. No two settings will have the same id value.
- **digitizer –** the value of the **id** of the digitizer these settings are for.

The `<settings>` tag contains the following elements:

<dcoffsets>

This element contains a `<dcoffset>` element for each channel of the digitizer. The attributes of the `<dcoffset>` tag are:

- **channel –** the ID number of a channel (channels number from 0).
- **value –** The value of the DC offset register for that channel.

<trigger>

This tag has the following attributes:

- **direction –** has the value rising or falling indicating the trigger edge.
- **mask –** has a mask of channels with bits set for each channel that participates in the trigger.
- **external** contains the state of the external trigger. The available values can be one between `acq` if the EXT input forces an acquisition, `both` if the external trigger both forces a trigger and outputs on the GPO output, `disabled` if the external trigger is disabled.

The `<trigger>` tag contains several `<level>` tags. Each of those has the attributes:

- **channel –** a channel number (numbered from 0).
- **value –** the threshold value for that channel.

<window>

Has the attribute **size** whose value is the number of samples acquired per trigger in each channel.

<posttrigger>

Has the attribute **value** which is of the form *nnn%* where *nnn* is the percentage of the window size in the post-trigger.

<channels>

Has the attribute **mask** which contains the mask of channels enabled to take data.

A Sample of Settings File

```
<caendigitizer>
<digitizer id="9641E17EA872CAED059B06194120AE02" family="xx740" version="80"
serial="174">
 <channels value="32"></channels>
 <resolution bits="12"></resolution>
 <frequency hz="65000000.0"></frequency>
 <maxsamples maxsamples="196608"></maxsamples>
 <channelgroups capable="8"></channelgroups>
 <zerosuppression capable=""></zerosuppression>
 <inspection capable=""></inspection>
 <dualedge capable=""></dualedge>
 <voltagerange low="0.0" hi="2.0"></voltagerange>
 <windows>
  <window size="192"></window>
  <window size="384"></window>
  <window size="768"></window>
  <window size="1536"></window>
  <window size="3072"></window>
  <window size="6144"></window>
  <window size="12288"></window>
  <window size="24576"></window>
  <window size="49152"></window>
  <window size="98304"></window>
  <window size="196608"></window>
 </windows>
</digitizer>
<settings id="1" digitizer="9641E17EA872CAED059B06194120AE02">
 <dcoffsets>
  <dcoffset channel="0" value="32768"></dcoffset>
  <dcoffset channel="1" value="32768"></dcoffset>
  <dcoffset channel="2" value="32768"></dcoffset>
  <dcoffset channel="3" value="32768"></dcoffset>
 </dcoffsets>
 <trigger direction="rising" mask="0" external="acq">
  <level channel="0" value="0"></level>
  <level channel="1" value="0"></level>
  <level channel="2" value="0"></level>
  <level channel="3" value="0"></level>
 </trigger>
 <window size="192"></window>
 <posttrigger value="5.0%"></posttrigger>
 <channels mask="0"></channels>
</settings>
```

```
</caendigitizer>
```

Text Event Log Files

Text event log files are also XML files. As with settings files, the top-level tag is a <caendigitizer> tag.

Similarly, there are one or more <digitizer> tags and one or more <settings> tags for the digitizers and settings in effect when data from them are recorded. If settings are modified during data taking, new settings records are written as needed.

For each event, there will be also an <event> tag. The <event> tag has the following attributes:

- **id** – a unique id for the event.
- **settings** – the id of the settings in effect when the event was triggered.
- **digitizer** – the id of the digitizer from which this event comes.
- **timestamp** – the trigger time stamp information for that event read out from the digitizer and expressed in units of the Trigger Clock (refer to the User Manual of the specific digitizer model).
- **clocktime** – external time of day clock expressed in seconds since 1970-01-01 00:00:00 +0000 (UTC); particularly useful when, determining the number of accepted triggers over an interval of clock time, and assuming the actual number of triggers is known, an approximate live-time can be computed as:

  triggers-accepted / triggers-total

The <event> tag has the following contents:

<triggershift>

Has the attribute **samples**. If this is non-zero, then the trigger stabilization was being computed when the event was triggered and the value of this variable correspond to the number of samples to shift the waveform by to make the trigger condition lie at the position indicated by the post-trigger value.

<trace>

An event will have several <trace> tags. The <trace> tag will have the **channel** attribute indicating which channel the trace comes from. The body of the <trace> tag is CDATA, containing a set of space separated integers that are the sample values in ADC counts.

A Sample of Text Event Log File

```
<caendigitizer>
<digitizer id="0A708183A6AB2BF6A3D944461802EAA8" family="xx725" version="240"
serial="912">
 <channels value="8"></channels>
 <resolution bits="14"></resolution>
 <frequency hz="256000000.0"></frequency>
 <maxsamples maxsamples="655360"></maxsamples>
 <channelgroups capable="0"></channelgroups>
 <zerosuppression capable=""></zerosuppression>
 <inspection capable=""></inspection>
 <dualedge capable=""></dualedge>
 <voltagerange low="0.0" hi="2.0"></voltagerange>
 <windows>
  <window size="655350"></window>
  <window size="327660"></window>
  <window size="163830"></window>
  <window size="81910"></window>
  <window size="40950"></window>
  <window size="20470"></window>
  <window size="10230"></window>
  <window size="5110"></window>
```

```
  <window size="2550"></window>
  <window size="1270"></window>
  <window size="630"></window>
 </windows>
</digitizer>
<settings id="2" digitizer="0A708183A6AB2BF6A3D944461802EAA8">
 <dcoffsets>
  <dcoffset channel="0" value="32768"></dcoffset>
  <dcoffset channel="1" value="32768"></dcoffset>
  <dcoffset channel="2" value="32768"></dcoffset>
  <dcoffset channel="3" value="32768"></dcoffset>
  <dcoffset channel="4" value="32768"></dcoffset>
  <dcoffset channel="5" value="32768"></dcoffset>
  <dcoffset channel="6" value="32768"></dcoffset>
  <dcoffset channel="7" value="32768"></dcoffset>
 </dcoffsets>
 <trigger direction="falling" mask="1" external="disabled">
  <level channel="0" value="2417"></level>
  <level channel="1" value="0"></level>
  <level channel="2" value="0"></level>
  <level channel="3" value="0"></level>
  <level channel="4" value="0"></level>
  <level channel="5" value="0"></level>
  <level channel="6" value="0"></level>
  <level channel="7" value="0"></level>
 </trigger>
 <window size="630"></window>
 <posttrigger value="49.0%"></posttrigger>
 <channels mask="1"></channels>
</settings>
<event id="1" settings="2" digitizer="0A708183A6AB2BF6A3D944461802EAA8"
timestamp="21109891" clocktime="1558428702">
 <triggershift samples="0"></triggershift>
 <trace channel="0">8144 8142 8148 8141 8142 8144 8136 8151 8149 8140 8148 8149 8142 8143
8148 8148 8141 8150 8150 8152 8143 8140 8148 8148 8144 8144 8142 8140 8143 8154 8146 8138
8148 8150 8148 8146 8150 8144 8148 8146 8146 8144 8136 8136 8148 8142 8144 8148 8148 8144
8144 8142 8146 8144 8144 8148 8142 8154 8144 8154 8138 8150 8140 8149 8147 8144 8143 8141
8148 8148 8141 8157 8144 8146 8150 8149 8146 8149 8142 8156 8148 8148 8158 8147 8148 8148
8144 8141 8142 8148 8150 8142 8144 8140 8140 8142 8144 8160 8154 8140 8142 8138 8152 8137
8150 8148 8144 8144 8147 8150 8140 8150 8146 8140 8148 8150 8148 8146 8140 8150 8148 8149
8142 8144 8148 8138 8146 8148 8146 8149 8144 8141 8150 8144 8149 8143 8144 8142 8146 8144
8149 8140 8156 8143 8149 8141 8148 8144 8144 8151 8140 8138 8148 8148 8140 8148 8141 8144
8144 8150 8138 8152 8148 8146 8142 8144 8142 8148 8147 8149 8149 8154 8148 8146 8143 8148
8140 8144 8138 8148 8148 8142 8152 8148 8152 8154 8152 8148 8144 8138 8146 8146 8149 8144
8150 8149 8138 8148 8144 8138 8142 8140 8146 8144 8150 8143 8140 8150 8141 8149 8144 8148
8148 8144 8154 8148 8148 8149 8142 8146 8144 8146 8142 8142 8144 8140 8148 8150 8151 8146
8148 8144 8148 8142 8152 8148 8152 8148 8148 8143 8143 8150 8152 8142 8146 8146 8140 8147
8144 8146 8144 8148 8144 8144 8148 8146 8138 8146 8144 8152 8149 8154 8150 8138 8138 8146
8142 8152 8148 8144 8148 8154 8144 8137 8140 8148 8148 8148 8014 4510 771 404 391 372 375
346 332 346 340 335 338 331 328 336 329 332 332 338 332 341 332 334 332 324 341 336 330
328 326 323 328 333 336 340 336 334 332 332 330 330 332 332 326 322 331 324 333 326 328
```

```
325 324 322 324 324 325 328 324 323 324 324 320 328 322 322 316 322 320 320 320 325 319
318 316 317 324 312 326 312 323 316 324 318 320 320 320 314 316 316 312 314 324 320 314
318 310 316 317 320 316 314 320 322 318 314 316 322 312 308 308 314 310 313 310 322 314
318 322 312 320 316 310 316 314 314 319 324 304 319 316 317 324 314 311 314 312 312 308
316 316 310 322 312 316 310 310 320 323 314 318 314 308 318 310 314 322 314 320 312 312
314 302 310 317 318 314 316 314 318 320 308 314 310 314 318 312 314 316 310 314 316 326
319 308 310 316 308 318 322 320 316 318 312 309 320 322 320 322 308 311 316 308 317 311
312 316 320 323 314 314 314 316 309 308 320 313 312 310 325 300 312 318 316 316 308 314
310 318 314 312 318 312 318 314 320 312 312 316 314 324 316 314 314 322 308 312 308 316
325 317 316 309 312 312 320 310 320 314 316 310 320 322 320 312 320 312 316 314 315 322
317 322 320 320 313 311 313 324 314 320 328 312 314 311 310 314 319 320 318 324 312 314
314 318 318 324 317 324 324 328 317 318 324 322 324 316 312 317 318 314 323 312 320 324
320 314 316 324 318 314 326 320 318 312 316 320 314 316 312 326 322 320 326 322 316 318
326 316 318 320 318 328 322 318 308 324 325 317 320 318 324
```


```
 </trace>
</event>
<event id="2" settings="2" digitizer="0A708183A6AB2BF6A3D944461802EAA8"
timestamp="146109465" clocktime="1558428703">
 <triggershift samples="0"></triggershift>
```


```
 <trace channel="0">8146 8146 8142 8139 8148 8144 8140 8150 8150 8142 8154 8148 8147 8149
8150 8138 8142 8148 8142 8150 8150 8146 8142 8143 8149 8144 8139 8138 8139 8143 8150 8142
8140 8138 8136 8144 8140 8138 8142 8148 8146 8149 8152 8146 8142 8142 8150 8143 8144 8142
8146 8144 8146 8138 8144 8148 8142 8148 8140 8152 8148 8148 8144 8144 8148 8142 8144 8146
8144 8152 8144 8148 8142 8143 8148 8148 8144 8139 8138 8144 8148 8138 8144 8142 8144 8143
8148 8140 8146 8140 8148 8138 8154 8146 8148 8148 8144 8142 8152 8144 8149 8148 8144 8149
8144 8148 8143 8148 8141 8148 8142 8140 8154 8148 8140 8142 8136 8138 8144 8146 8150 8138
8140 8148 8143 8138 8136 8148 8144 8140 8150 8136 8142 8136 8140 8142 8151 8154 8150 8142
8148 8149 8144 8144 8155 8148 8146 8144 8148 8144 8151 8147 8148 8137 8136 8146 8140 8148
8140 8148 8138 8146 8148 8144 8148 8138 8138 8136 8146 8150 8144 8136 8144 8141 8136 8148
8137 8148 8152 8152 8152 8136 8144 8142 8150 8143 8144 8141 8148 8144 8148 8143 8146 8147
8146 8144 8138 8146 8142 8148 8154 8138 8140 8139 8136 8144 8138 8150 8137 8140 8146 8148
8148 8144 8144 8150 8142 8144 8148 8146 8146 8141 8148 8144 8144 8148 8146 8151 8148 8146
8154 8139 8146 8148 8148 8142 8147 8149 8148 8146 8152 8144 8144 8146 8148 8143 8144 8146
8143 8144 8148 8140 8148 8140 8152 8141 8134 8148 8146 8142 8148 8146 8150 8146 8147 8146
8144 8148 8144 8149 8142 8140 8152 8144 8139 8149 8147 8144 8134 6095 1378 420 402 390
380 355 344 342 330 326 336 330 340 334 334 332 336 346 329 336 328 324 338 332 334 334
332 328 326 326 326 322 342 336 330 334 328 335 330 336 334 328 326 330 334 324 330 322
329 326 328 328 325 325 330 322 322 326 324 320 334 320 324 322 318 324 320 330 320 326
312 316 316 326 320 312 319 308 316 310 322 323 318 320 320 320 314 318 324 317 322 314
323 320 314 320 317 320 320 320 312 320 318 312 313 312 314 316 312 324 314 314 320 314
320 320 324 320 324 312 317 316 316 316 316 314 316 308 320 320 319 310 316 314 320 310
309 314 316 310 313 320 324 318 316 312 314 314 316 320 310 312 314 312 314 324 324 310
318 320 312 316 316 320 318 318 312 318 323 318 320 314 311 309 313 314 314 308 322 324
316 314 318 309 320 318 314 316 314 318 314 316 316 322 316 308 320 314 314 318 322 318
312 316 320 324 312 314 320 312 326 318 318 318 308 320 320 316 309 309 320 316 314 310
318 312 312 311 316 322 318 312 318 310 322 328 314 302 288 322 346 329 296 296 304 312
332 326 293 298 324 316 302 330 316 324 323 310 298 318 332 316 312 314 314 317 312 320
318 320 317 316 316 318 326 310 320 316 319 324 312 316 314 318 316 314 320 318 328 316
320 310 318 326 309 318 324 314 316 314 316 312 324 320 320 320 320 318 310 320 312 318
324 314 318 318 312 309 322 313 314 320 317 320 318 320 320 318 322 320 318 320 318 318
322 320 314 320 320 320 324 318 323 316 322 324 320 318 312 314
 </trace>
</event>
</caendigitizer>
```

Binary (Raw) Event File Format

This section describes the format of the data recorded to raw binary files. The files consist of a sequence of variable length, typed, records. Each record consists of a header that has a common format followed by a body whose format is determined by the record type.

Header Format

The header of each consecutive record in the file consists of two 32-bit words shown in **Tab. 12.1**.

Header Format
Record size in bytes (includes this 32-bit word)
Record type code

Tab. 12.1: Raw binary header format

The first 32-bit word in the header is the size of the entire record in 8-bit bytes. The size value takes into account the full header size (including the size word itself).

The record type code determines the structure and contents of the corresponding body. Valid types are currently:

Type Code	Corresponding Record Body Description
1	*Digitizer descriptor*. This record body describes a digitizer whose data appears in the output file. In the current version of CAENScope, there will only be one of these records.
2	*Digitizer Settings*. For each output file, there will always be an initial digitizer settings record that describes the digitizer settings at the time the file was created. If the settings are subsequently modified, while recording data, additional digitizer settings entries will be added so that, for any event, the User knows the digitizer settings in effect at that time.
3	*Waveform data*. This record body contains the data from one channel of the digitizer for an event. Several contiguous records of this type constitute a single event, that is the data that came across all enabled channels because of a trigger.

Tab. 12.2: Valid Record Type codes

Digitizer Descriptor Record Body

Digitizer descriptor records describe a digitizer from which data was recorded. The body is fixed length and has the following format:

Field Name	Data Type	Contains
Digitizer Id	Uint32_t	Unique (within the output file) identifier of the digitizer. Other records may contain digitizer ids that reference this digitizer, indicating they belong to this digitizer.
Family Code	Uint32_t	The digitizer family code. For example, for a DT5730 digitizer this will be 730 decimal. For a V1725 digitizer, this will be 725 decimal.
ROC Version	20 chars	ROC Firmware version string. This is left justified, zero filled.
AMC Version	20 chars	AMC Firmware version string. Left justified, zero filled.
Serial Number	Uint32_t	Device unique (per model) serial number.
Board Version	Uint32_t	Board Version number.
NChans	Uint32_t	Number of channels that populate the board.
Bits	Uint32_t	Number of bits of resolution the digitizer has.
Hz	Uint64_t	Digitizer sampling frequency in Hz.
Max Samples	Uint32_t	The longest trace in samples a channel can record.

Capflags	Uint32_t	Digitizer Capability flags. This is a bitwise OR of the following bits: 0 – Set if the digitizer has channel groups. 1 – Set if the digitizer supports zero suppression. (720 and 724 family only) 2 – Set if the digitizer has the capability to provide waveform inspection outputs (V1724 digitizer only). 3 – Set if the digitizer is dual edge sampling capable. (751 family only) For instance, for a DT5725 digitizer the value associated to this variable will have the 4 least significant bits set to 0001 since this digitizer only has channel groups.
Vlow	Uint32_t	Channel input voltage range low value in mV.
Vhigh	Uint32_t	Channel input voltage range high value in mV.

Tab. 12.3: Digitizer Descriptor Record body

Digitizer Settings Record Body

The Digitizer Settings are variable length records that describe the settings of a digitizer at a specific point in time. All subsequent events from the referenced digitizer will have been taken with the settings described by this record until a subsequent digitizer settings record.

The body is variable sized, as some items in the body are individual channel settings. The structure of the body is:

Field Name	Data Type	Contains
Settings Id	Uint32_t	An integer that is unique over the settings in this file. Event records will contain this value so that the User knows which settings were in effect when a particular event was recorded.
Digitizer Id	Uint32_t	ID of the digitizer for which the settings are recorded. This is the same ID number present in the Digitizer Descriptor body.
Trigger Mask	Uint32_t	Mask of the enabled channel self-triggers. Channel 0 is the least significant bit etc.
External Trigger Code	Uint32_t	Describes the use of the external trigger with the following legal values represented via the 2 least significant bits of the field: 0 – External output 1 – Acquire Data on external trigger. 2 – Acquire data on external or channel trigger. 3 – Disabled. If bit 2 is set to 1, the trigger will be on the rising edge otherwise it'll be on the falling edge. As an example, if the 3 least significant bits of this field are set to 101, it means that the data are acquired on the external trigger and the trigger will occur on the rising edge crossing.
Window Size	Uint32_t	Number of samples of data that will be taken.
Post Trigger	Uint32_t	Post trigger value in 100*percent. For example, if the post trigger is set to 64 %, the value of this field will be 6400.
Channel Mask	Uint32_t	Bit mask indicating which channels are enabled.
NChans	Uint32_t	The number of channels the digitizer has. This information will be followed by the DC offsets and Trigger levels for each digitizer channel. For example, if the digitizer has 16 channels, this value will be followed by 16 DCoffset values and then 16 Trigger threshold level values.
DCOffsets	Uint32_t*nchan	DC Offset in DAC value for each channel.
Trigger Levels	UInt32_t*nchan	Trigger thresholds for each channel.

Tab. 12.4: Digitizer Settings Record body

Waveform Record Body

These types of record contain the data from a single channel in an event. An event consists of all the data that are acquired and recorded for a single trigger. Since these records only contain the data from a single channel, several of these records may be written for each event. The records associated with a single event and corresponding to different channels will be contiguous in the file.

Waveform record bodies are variable length. The format and contents of this record are:

Field Name	Data Type	Contains
Event Id	Uint32_t	Unique number that identifies an event. All waveform records with the same event ID belong to the same event.
Digitizer Id	Uint32_t	Identifies the digitizer from which this waveform was acquired.
Settings Id	Uint32_t	Identifies the settings in effect when the waveform was acquired.
Trigger Time	Uint32_t	The digitizer trigger time tag expressed in units of trigger clock cycles. The period of the trigger clock depends on the digitizer family (see **Tab. 12.6**).
Time Stamp	Uint64_t	Time stamp (e.g. value of the time function) at the time the waveform was received by the CAENScope software. The time stamp is expressed as the number of seconds since 1 January 1970, 00:00 UTC.
Tshift	Uint32_t	If the Trigger Stabilization is in effect, the value of this variable represents the number of samples needed to shift the waveform to place its trigger position in the position indicated by the post trigger settings. This is a signed value. The trace stabilization accounts for the fact that the digitizer clock and FPGA trigger clocks are at different frequencies, thus making the position of the trigger sample jitter in time on the plot.
Channel	Uint32_t	Channel number from which the waveform was acquired.
Samples	Uint32_t	Number of samples in the waveform.
Waveform	Uint16_t*Samples	Waveform data, one Uint16_t value per sample.

Tab. 12.5: Waveform Record body

Digitizer Family	Trigger Clock Period
720	8 ns
724	10 ns
725	8 ns
730	8 ns
751	8 ns

Tab. 12.6: Trigger clock period for the different digitizer families supported by CAENScope.

Binary (SQLite3) Database Schema

CAENScope allows the User to save binary files are SQLite3 databases. For more information regarding SQLite3, the User is suggested to see http://www.sqlite.org. To summarize, SQLite3 is an SQL database in a file. This section will describe the schema of this database. SQLite has bindings to many programming languages including C/C++, Python and Java. There is a very clean mapping between the XML formats described earlier and the database tables used by the binary format.

The root table is the digitizer table and its contents capture most of the information in the `<digitizer>` tag for the XML file formats.

Field Name	Data Type	Contains
id	VARCHAR(32)	Contains a unique digitizer id (the same as the id in the <digitizer> tag). This is the primary key for this table.
family	VARCHAR(32)	Family name, e.g. xx724
version	INTEGER	Board version
serial	INTEGER	Module serial number
channels	INTEGER	Number of channels in the module
resolution	INTEGER	Number of bits of resolution the samples have.
frequency	INTEGER	Sampling frequency of the digitizer.
maxsamples	INTEGER	Maximum number of samples a channel can have for each trace.
channelgroups	INTEGER	0 or 1 if the digitizer does not have grouped channels, otherwise this is the number of channels in a channel group.
zerosuppress	INTEGER	Non-zero if the digitizer has zero suppressed waveform formats (720 and 724 family only).
inspectable	INTEGER	Non-zero if the digitizer has an inspection output (724 family only).
dualedge	INTEGER	Non-zero if the digitizer supports dual edge sampling (751 family only).
vlow	REAL	Low limit on the input voltage.
vhigh	REAL	High limit on the input voltage.

Tab. 12.7: *digitizers* table fields

The User may recall that the XML format has a mechanism for storing the set of pre-defined buffer sizes for a digitizer. In the SQLite database format this is captured by the window sizes table which has the following fields:

Field Name	Data Type	Contains
id	INTEGER	Sequentially assigned integer. This is the table PRIMARY KEY.
digitizer_id	VARCHAR(32)	Foreign key pointing back to the record in the digitizer table that describes the digitizer for which this window size is valid.
samples	INTEGER	The number of samples in a predefined buffer size.

Tab. 12.8: *window_sizes* fields

Note that in general there will be several entries in the window sizes table for each entry in the digitizer table.

The information in the `<settings>` tag is captured in a set of tables that have as a root the settings root table. This table has the following fields:

Field Name	Data Type	Contains
id	INTEGER	Sequentially assigned unique id for the record. This is the PRIMARY KEY of the record.
digitizer_id	VARCHAR(32)	Id of the digitizer in the digitizer table for which these settings are valid. This is a FOREIGN Key to the digitizers table.
trigger_dir	VARCHAR(7)	Either 'rising' or 'falling' indicating the trigger edge.
ext_trigger	VARCHAR(10)	How the external trigger input is handled. The available values can be one between `acq` if the EXT input forces an acquisition, `both` if the external trigger both forces a trigger and outputs on the GPO output, `disabled` if the external trigger is disabled.
window_size	INTEGER	Number of samples that will be acquired for each channel on a trigger.
post_trigger	REAL	Percent of the window size that will be kept after the trigger.
trigger_mask	INTEGER	Mask of channels that are enabled to trigger.
time_stamp	INTEGER	A time stamp in seconds since January 1, 1970 00:00:00. This time indicates when this setting entry was made in the database

Tab. 12.9: *settings_root* table fields

Information in the <thresholds> tag is maintained in the settings_thresholds table which has the following fields:

Field Name	Data Type	Contains
id	INTEGER	Sequentially assigned unique id for the record. This is the PRIMARY KEY of the record.
settings_id	INTEGER	FOREIGN KEY that contains the id of the record in the settings_root table for which this record is a threshold value.
channel	INTEGER	Channel number of the digitizer.
threshold	INTEGER	Threshold value for the channel.

Tab. 12.10: *settings_thresholds* table fields

Similarly, the DC offsets for settings are maintained in the settings_dcoffsets table:

Field Name	Data Type	Contains
id	INTEGER	Sequentially assigned unique id for the record. This is the PRIMARY KEY of the record.
settings_id	INTEGER	FOREIGN KEY that contains the id of the record in the settings_root table for which this record is a DC offset value.
channel	INTEGER	Channel number of the digitizer.
offset	INTEGER	DC Offset value for the channel.

Tab. 12.11: *settings_dcoffsets* table fields

The events themselves are in a pair of tables. The events table contains an overall description of the event including linkages back to the digitizer and settings applied at the time the event was acquired. The samples table contains the actual samples.

Field Name	Data Type	Contains
id	INTEGER	Unique integer assigned to every event (PRIMARY KEY).
digitizer_id	VARCHAR(32)	The id of the digitizer that created the event (FOREIGN KEY to the digitizers table).
settings_id	INTEGER	Id of the settings that were active when the event was acquired (FOREIGN KEY to the settings_root table).
trigger_shift	INTEGER	If trigger stabilization was enabled, the value of this variable indicates how many samples were necessary to shift the waveform to place the trigger in the position mandated by the Post-trigger value. If trigger stabilization was not enabled, this will be zero.
time_stamp	INTEGER	Time stamp in seconds from January 1, 1970 00:00:00 at which the event was entered in the database. An important note, this is not the timestamp from the digitizer/event but the time at which the event was read by CAENScope.

Tab. 12.12: *events* table fields

Traces are in the sample table which has the following fields:

Field Name	Data Type	Contains
id	INTEGER	Unique integer assigned to the trace (PRIMARY KEY)
event_id	INTEGER	Id of the event metadata in the events table (FOREIGN KEY into the events table).
channel	INTEGER	Channel from which the trace comes.
samples	BLOB	List of integers that make up the trace samples.

Tab. 12.13: *samples* table fields

Given a digitizer with the id 9641E17EA872CAED059B06194120AE02, the following query will therefore return all traces from all channels of that digitizer along with settings information that was current at the time the traces were acquired and information about the digitizer itself. (note other queries are possible as well):

```
SELECT * FROM SAMPLES S
INNER JOIN events E ON E.id = S.event_id
INNER JOIN settings_root SR ON         SR.id = E.settings_id
INNER JOIN settings_thresholds ST ON   ST.settings_id = SR.id
                                   AND ST.channel = S.channel
INNER JOIN settings_dcoffsets SDCO ON SDCO.settings_id = SR.id
                                   AND SDCO.channel = S.channel
INNER JOIN digitizers D on            D.id = E.digitizer_id
```

```
WHERE D.id = '9641E17EA872CAED059B06194120AE02';
```

This method of starting with traces and working backwards through the joins towards the digitizer makes use of the *belongs to* relationships between the child and parent tables. One could equally well work from the digitizer down through the settings and then the events/samples tables which would be making use of the *has* many relationships between parent and child tables. Note how the extra condition on the join with the settings_threshold and settings_dcoffsets picks out only the thresholds and offsets for the trace's channel.

13 C++ Library to Read Raw Binary Files

CAEN makes available inside the software folder a C++ class, named CBinaryIn, and associated header that the Users can incorporate into their own code to read raw binary files (see Sec. **Binary (Raw) Event File Format**). The code contains the same class library used inside CAENScope software. This chapter describes:

- The class header constants
- The class header public data structures
- How to use the class in a practical way, constructing and reading records from the class.

Note that the class source code provided compiles and runs on both Windows and Linux OS. Note that the class is self-contained, relying only on the vector, and string types from the C++ standard template library.

CBinaryIn constants

The CBinaryIn class header defines several public, static constants within the class. These provide symbolic values for the constant values described in the sections that describe the file structure.

These constants fall into three categories:

Record Type constants

Name	Data Type	Description
CBinaryIn::tp_DigitizerDescription	Uint32_t	Record type for digitizer descriptions.
CBinaryIn::tp_DigitizerSettings	Uint32_t	Record type for digitizer Settings.
CBinaryIn::tp_TraceData	Uint32_t	Record type for waveform data.

Tab. 13.1: CBinaryIn Record type constant definitions

Digitizer Capability Flag constants

These values are symbolic values for the bits in the digitizer description capability flag field:

Name	Data Type	Description
CBinaryIn::cap_hasGroups	Uint32_t	Digitizer has channel groups.
CBinaryIn::cap_canZsuppress	Uint32_t	Digitizer can zero suppress traces.
CBinaryIn::cap_canInspect	Uint32_t	Digitizer has inspection test points.
CBinaryIn::cap_DualEdgeClock	Uint32_t	Digitizer can acquire on both edges of the digitizer clock.

Tab. 13.2: CBinaryIn Digitizer Capability flag definitions

External Trigger Usage Codes

These constants provide the codes in the external trigger field of the settings record:

Name	Data Type	Description
CBinaryIn::trg_TriggerOnExtTrg	Uint32_t	Trigger on external trigger only.
CBinaryIn::trg_TriggerOnBoth	Uint32_t	Trigger on external and channel self-triggers.
CBinaryIn::trg_ExtTrigDisabled	Uint32_t	External Trigger disabled (only trigger on channel self-trigger).
CBinaryIn::trg_Rising	Uint32_t	If this bit is set to 1, the trigger fires on the signal rising over the threshold otherwise it fires on the signal dropping below the threshold.

Tab. 13.3: External Trigger codes

CBinaryIn Class Data Structures

The CBinaryIn class pulls data from the file into data structures that are different with respect to the internal file structures. For example, variable sized data (e.g. channel thresholds) are pulled into std::vector objects. This supports simplified User code to access these data. Each data structure is represented as a typedef. In the sections that will follow, we will use the typedef name. In addition to the name used (e.g. `DigitizerDescription`), there is always a typedef pointer that has the same name prefixed with a p (e.g. `pDigitizerDescription`).

CBinaryIn::header

This struct describes a record header and contains the following fields:

Name	Data Type	Description
s_size	Uint32_t	Size of the entire record in bytes including this header.
s_type	Uint32_t	Type of the record. This will be one of the type constants described in **Tab. 13.1** above.

Tab. 13.4: CBinaryIn::header fields

CBinaryIn::DigitizerDescriptor

This struct describes a digitizer. The body of a digitizer descriptor record (described in Sec. **Digitizer Descriptor Record Body**) is read into this struct. It has the following fields:

Name	Data Type	Description
s_id	Uint32_t	Digitizer id. A unique value within the file, assigned to the digitizer described by this struct.
s_familyCode	Uint32_t	Integer part of a digitizer family code. Normally, family codes are of the form xxnnn where xx are the literal characters "xx" and nnn is a three digit number. This field contains the integer part of this code. For example, a V1725 has a family code xx725. This will appear in this field as 725.
s_ROCVersion	20 chars	Text string containing the ROC firmware version.
s_AMCVersion	20 chars	Text string containing the AMC firmware version.
s_SerialNumber	Uint32_t	The serial number of the digitizer module.
s_boardVersion	Uint32_t	The version number of the board.
s_nChans	Uint32_t	The number of input channels on the board. Different actual members of the same family may have different channel counts, for example a V1725 and DT5725 are both xx725 family modules but the V1725 has 16 channels while the DT5725 has only 8.
s_bits	Uint32_t	Number of bits in each digitized sample.
s_Hz	Uint64_t	Digitization frequency in Hz. For example, the 725 family will have 250000000 in this field.
s_maxSamples	Uint32_t	Largest number of samples that can be in a waveform acquired by this module.
s_caplags	Uint32_t	The module capabilities expressed as a bitwise OR of the values described in **Tab. 13.2.**
s_vlow	Uint32_t	Low end of the input range in mV. Note that this value is not described by the digitizer. When a module is connected having several possible voltage ranges, the User must indicate the module's voltage range. This is subject to error when the voltage ranges are set in the hardware but entered by the User.
s_vhigh	Uint32_t	High end of the input range in mV.

Tab. 13.5: DigitizerDescriptor structure fields

CBinaryIn::DigitizerSettings

The digitizer settings record body on file is composed of a fixed header followed by a variable length chunk whose size depends on the number of channels of the digitizer. To capture that, and simplify reading this record from file, the fixed part is represented by a CBinaryIn::DigitizerSettingsFixedHeader struct and the record as a whole is represented by a CBinaryIn::DigitizerSettings struct which includes the CBinaryIn::DigitizerSettingsFixedHeader as a member.

The members of the CBinaryIn::DigitizerSettingsFixedHeader struct are:

Name	Data Type	Description
s_id	Uint32_t	Settings id. This is a unique value assigned to each setting record.
s_did	Uint32_t	Digitizer id. Identifies the digitizer for which these settings apply.
s_TriggerMask	Uint32_t	Mask of trigger enable bits. Each bit represents a channel. If the bit is 1 that channel's self-trigger is enabled.
s_triggerCode	Uint32_t	Triggering code. See **Tab. 13.3** for a description of the values this field can take.
s_windowSize	Uint32_t	Number of waveform samples that will be captured as a result of a trigger.
s_postTrigger	Uint32_t	Post trigger length of the window size in percent multiplied by 100.
s_channelMask	Uint32_t	Contains a bit for each channel. If the channel's bit is set, the channel will be captured in the event of a trigger.
s_nChannels	Uint32_t	Number of channels in the digitizer.

Tab. 13.6: Fields in CBinaryIn::DigitizerSettingsFixedHeader struct

The CBinaryIn::DigitizerSettings struct has the following fields:

Name	Data Type	Description
s_header	CBinaryIn::DigitizerSettingsFixedHeader	The fixed part of the settings.
s_DCOffsets	std::vector<uint32_t>	Per channel DC offsets.
s_TriggerLevels	std::vector<uint32_t>	Per channels trigger threshold values.

Tab. 13.7: Fields in the CBinaryIn::DigitizerSettings struct

CBinaryIn::WaveformData

The data in CBinaryIn::WaveformData are variable length. As with CBinaryIn::DigitizerSettings, this struct is divided into a fixed length header, i.e. CBinaryIn::WaveformFixedHeader, which is then placed in the overall CBinaryIn::WaveformData struct.

Name	Data Type	Description
s_eventId	Uint32_t	A unique value assigned to the event. All waveform records with the same s_eventId are part of the same event.
s_did	Uint32_t	Id of the digitizer from which the event was taken.
s_sid	Uint32_t	Id of the settings that were in effect at the time the data were taken.
s_triggerTag	Uint32_t	The high-resolution time stamp that indicates when the trigger occurred. The units of this time stamp are digitizer dependent (see **Tab. 12.6**)
s_todStamp	Uint64_t	The low-resolution date/time stamp at which the event was acquired. The units of this time are seconds with a base of 1 January 1970, 00:00 UTC.
s_shift	Uint32_t	The number of samples to shift the raw waveform by to make its threshold crossing (trigger time) consistent with the post-trigger percent. Note that this stabilizes the trigger position against the jitter occurring due to the difference in trigger and sampling clock frequencies. If trigger stabilization is not enabled, this will be 0.
s_channel	Uint32_t	The channel number from which these data were taken.
s_nSamples	Uint32_t	The number of waveform samples acquired.

Tab. 13.8. Fields of the CBinaryIn::WaveformFixedHeader struct

The variable length array that contains the waveform samples is then captured in a std::vector object:

Name	Data Type	Description
s_header	CBinaryIn::WaveformFixedHeader	Fixed header. See **Tab. 13.8** above.
s_trace	std::vector<uint16_t>	The waveform samples.

Tab. 13.9: Fields of the CBinaryIn::WaveformData struct

Methods of CBinaryIn

CBinaryIn.h provides:

- Construction
- Destruction
- The ability to read a record header
- The ability to read specific types of record bodies

The suggested flow for a program using the library is to construct a CBinaryIn object connected to the raw binary data file the User's application will read and then alternately read headers and the appropriate body type (based on the record type in the header) and then, when the file has been completely read, destroy the used CBinaryIn object.

Construction

The constructor of the CBinaryIn class has the following signature:

```
CBinaryIn(const char* filename)
```

The filename parameter is a pointer to a null terminated string that contains the path to the file to open. If the file cannot be opened, the constructor will throw a std::system_error exception that describes why this was the case.

readHeader

The method reads a record header into the User supplied buffer. The `readHeader` method of the CBinaryIn class has the following signature:

```
int
CBinaryIn::readHeader(CBinaryIn::header& buffer)
```

The `buffer` parameter is passed by reference and will be filled in with a header from the file. Note that this method does not know how to skip to the next header from any arbitrary file position. It's up to the caller to ensure that the file position is such that a read of the appropriate size will return a header.

The method returns an integer which is:

Value	Meaning
>0	The number of bytes read from the file. This should be `sizeof(CBinaryIn::header)` if used in a properly formatted file.
0	End of file has been encountered.
<0	Some error occurred when reading the file and the reason for the error can be retrieved from `errno`

Tab. 13.10: Return values from CBinaryIn::readHeader

readDigitizerDescriptor

This method reads a digitizer descriptor record body. The `readDigitizerDescriptor` record has the following signature:

```
int
CBinaryIn::readDigitizerDescriptor(
    CBinaryIn::DigitizerDescriptor& buffer
)
```

e

The `buffer` parameter references a `CBinaryIn::DigitizerDescriptor` struct that will be filled in with the data from the file. It is up to the caller to ensure that the file is positioned such that the data following the file position is a valid `CBinaryIn::DigitizerDescriptor` record.

Return values from this method are as follows:

Value	Meaning
>0	Number of bytes read from the file. This should be `sizeof(CBinaryIn::DigitizerDescriptor)` if this method is used in a properly formatted file.
0	There is no more data in the file. This should never happen on a properly formatted file as it indicates a header was read that did not have a body.
<0	An error was encountered reading the file. The reason for the error is in `errno`.

Tab. 13.11: Return values from CBinaryIn::readDigitizerDescriptor

readDigitizerSettings

This method reads a digitizer settings record body. The `readDigitizserSettings` method has the following signature:

```
int
CBinaryIn::readDigitizerSettings(
    CBinaryIn::DigitizerSettings& buffer
)
```

The `buffer` parameter is a `CBinaryIn::DigitizerSettings` reference to a buffer that will be filled in from the file. The caller must ensure the file is positioned to read a Digitizer settings record. The return values from this method are:

Value	Meaning
>0	The number of bytes read from file. Note that due to the structure of `CBinaryIn::DigitizerSettings` this value will, in general, be larger than `sizeof(CBinaryIn::DigitizerSettings)`.
0	An end of file was encountered. This should never happen on a properly formatted file as it indicates a header without a body.
<0	An error was encountered and reported in `errno`.

Tab. 13.12: Return values from CBinaryIn::DigitizerSettings

readWaveform

The `CBinaryIn::readWaveform` method reads a waveform record body from the file. The file must be positioned at a waveform record body. The signature of this method is:

```
int CBinaryIn::readWaveform(CBinaryIn::WaveformData& buffer)
```

The `buffer` parameter is a reference to a `CBinaryIn::WaveformData` struct that will receive the contents of the read. The return values are:

Value	Meaning
>0	The number of bytes read from file. Note that due to the structure of `CBinaryIn::WaveformData` this value will, in general, be larger than `sizeof(CBinaryIn::WaveformData)`.
0	An end of file was encountered. This should never happen on a properly formatted file as it indicates a header without a body.
<0	An error was encountered and reported in `errno`.

Tab. 13.13: Return values from CBinaryIn::readWaveform

14 Troubleshooting

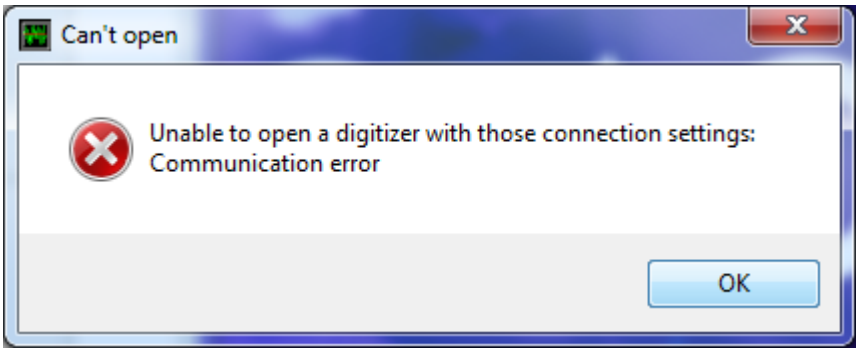


Fig. 14.1: Communication error message

If CAENSCOPE displays the message window shown in **Fig. 14.1**, the User could be in one of the cases listed below:

- The digitizer is powered off or the communication cable is not plugged in, improperly plugged or defective.
- The driver for the communication link is not installed or not correctly installed.
- Incorrect connection parameters have been set by the User in the Connection Dialog window (see Sec. **Acquisition System Set Up**).
- The digitizer or, if a VME digitizer is used, the bridge, is hung. Cycle power on the digitzer/crate to recover from this.

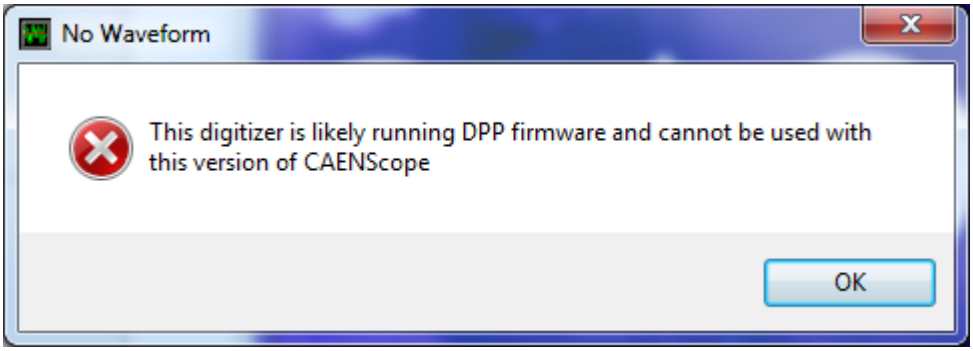


Fig. 14.2: FW compatibility error message

The message shown in **Fig. 14.2**, instead, is displayed when the digitizer runs a DPP firmware, which is not supported by CAENSCOPE. Refer to Sec. **Firmware Compatibility** for details.

15 Technical support

CAEN 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:

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

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

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

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