Rev. 7 - July 16th, 2024

# R5560/R5560SE

128 Channel 14 bit 125 MS/s Open FPGA Digitizer

CAEN
Tools for Discovery

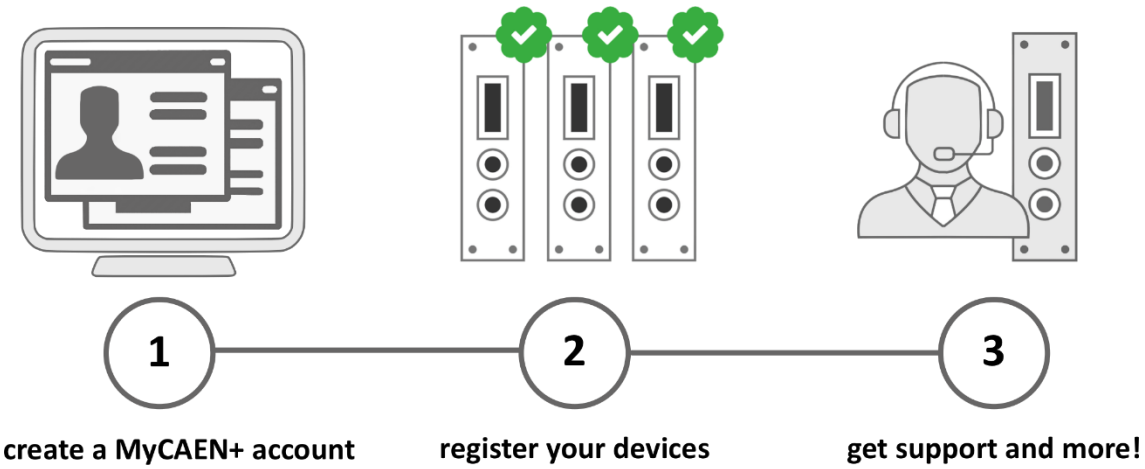
**User Manual**

UM6952

# 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 User Manual

This User Manual contains the full description of the R5560SE 128-Channel 14-bit @125 MS/s Open FPGA Digitizer, of its web interface and a brief guide to the Open Hardware Readout Software.

# Change Document Record

| Date | Revision | Changes |
|---|---|---|
| June 5th, 2020 | 00 | Initial release |
| December 17th, 2020 | 01 | Revision of a deleted section |
| March 19th, 2021 | 02 | Revised **Functional Description** |
| May 27th, 2021 | 03 | Minor revision of Technical Specifications |
| October 15th, 2021 | 04 | Revised **Safety Notices**<br>Minor revision of Technical Specifications |
| February 2nd, 2022 | 05 | General revision.<br>Added new update procedure<br>Added Chap. **Touchscreen Display Guide** |
| June 20th, 2023 | 06 | Unification of R5560 and R5560SE User Manuals and complete revision of contents |
| July 16th, 2024 | 07 | Rectified pinout of R5560 analog input connector |

# Symbols, abbreviated terms and notation

| | |
|---|---|
| ADC | Analog to Digital Converter |
| ASIC | Application Specific Integrated Circuit |
| DAQ | Data Acquisition |
| FIFO | First In First Out |
| FPGA | Field Programmable Gate Array |
| FTP | File Transfer Protocol |
| OS | Operating system |
| SDK | Software Development Kit |

# Reference Document

[RD1] GD6520 - SCI-Compiler Quick Start Guide

[RD2] UM6519 - SCI-Compiler User Manual

# Manufacturer contact

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



# Limitation of Responsibility

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

# Disclaimer

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

# Made in Italy

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

---

# Table of contents

# List of Figures

# List of Tables

# 1 Introduction

The R5560/R5560SE is a 19" rack-mount 128 Channels 14-bit 125MS/s **Open FPGA** Digitizer with **Differential** or **Single-Ended** inputs, designed to attain programmable data processing capabilities.

The R5560SE is an optimal solution for **large experiments,** usually requiring fast digitization of analog signals and usage of several digital lines to interface with external systems. The board supports **multi-board synchronization** through a single CAT5e cable, with the possibility scale up to thousands of channels**.** Moreover, the **rack–mount** form factor simplifies the experimental setup in case of multi-board systems, where an effective space management is often a constraint.

Thanks to its programmability, the board can fit several applications**.** In fact, it is possible to take advantage of the powerful SoC mounted onboard (Xilinx Zynq®-7030 or 7035 models) to write a **custom pulse processing algorithm** on the open FPGA as well as build a middleware/software that fits the needs of the application of interest.

The device is designed to operate in laboratory environment under the supervision of skilled technicians.

The R5560 features differential analog input on RJ45 connectors while the R5560SE features single-ended analog inputs on MCX connectors with an **advanced and versatile analog frontend**: it is possible to configure the gain, input impedance, offset on each single channel and also integrate a programmable shaper, making the device an optimum choice for gamma-rays applications.

The R5560/R5560SE is capable to manage simultaneously **digital** and **analog** signals to implement many functionalities required by physics experiments: signal digitization, complex trigger logic, Pulse Height Analysis with MCA capabilities, Time Tagging, Pulse Shape Discrimination, Scaler, Counters and so on.

Thanks to the Open FPGA and programmable analog frontend, the R5560SE can be used for a wide range of applications:

- Charge integration for the readout of SiPM, Silicon Detectors, PMTs
- Neutron detectors readout
- Trapezoidal filter PHA for HPGe readout
- Pulse shape discrimination
- Continuous reset or transistor reset preamplifier readout

The R5560/R5560SE is fully supported by **SCI-Compiler,** a Windows-based graphical development system for **easy FPGA programming**. This tool allows to develop and compile the firmware code using graphical blocks which represents the functionalities needed for firmware implementation (for example oscilloscope, TDC, MCA, charge integration, etc). SCI-Compiler automatically generates the VHDL firmware code starting only from logic blocks and virtual instruments that can be connected together in the GUI and, moreover, it allows usage of the compatible Sci-SDK in Windows, Linux (refer to **[RD1]** for more details).

A complete, ready-to-use default firmware is provided for free and open source. The default firmware manages the basic waveform digitization and pulse height analysis, and it is preloaded on the board. The user can open the default firmware project in SCI-Compiler and modify it in order to customize, for example, the trigger logic, the data online processing or integrate it in a larger system.

Available board models and accessories are listed below.

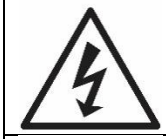
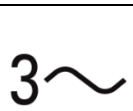
| Ordering options | | |
|---|---|---|
| R5560A | R5560A – 128 Ch. 14 bit 125 MS/s Digitizer-7030 | WR5560AXAAAA |
| R5560B | R5560B – 128 Ch. 14 bit 125 MS/s Digitizer-7035 | WR5560BXAAAA |
| R5560SE | R5560SE – 128 Ch. 14 bit 125 MS/s Digitizer single-ended – 7030 | WR5560SEXAAA |
| R5560SEB | R5560SEB – 128 Ch. 14 bit 125 MS/s Digitizer single-ended - 7035 | WR5560SEBXAA |
| **Accessories** | | |
| SW55 | SCI-Compiler User Firmware Generator | WSW555XAAAAA |
| RCS1Y | 1 year remote customization service + upgrade for Sci-Compiler | WSW55RCSXAAA |
| RCS5Y | 5 years remote customization service + upgrade for Sci-Compiler | WSW55RCSX5YA |

**Table 1.1:** table of available board models and accessories.

# 2 Safety Notices

**N.B. Read carefully the "Precautions for Handling, Storage and Installation" document provided with the product before starting any operation.**

The following HAZARD SYMBOLS may be reported on the unit:

| | |
|---|---|
|  | Caution, refer to product manual |
|  | Caution, risk of electrical shock |
|  | Protective conductor terminal |
|  | Earth (Ground) Terminal |
|  | Alternating Current |
|  | Three-Phase Alternating Current |

The following symbol may be reported in the present manual:

| | |
|---|---|
|  | General warning statement |

The symbol could be followed by the following terms:

- **DANGER**: indicates a hazardous situation which, if not avoided, will result in serious injury or death.
- **WARNING**: indicates a hazardous situation which, if not avoided, could result in death or serious injury.
- **CAUTION**: indicates a situation or condition that, if not avoided, could cause physical injury or damage the product and / or its environment.

To avoid potential hazards, use the product only as specified. Only qualified personnel should perform service procedures.
**Avoid Electric Overload.** To avoid electric shock or fire hazard, do not power a load outside of its specified range.
**Avoid Electric Shock.** To avoid injury or loss of life, do not connect or disconnect cables while they are connected to a voltage source.

**Do Not Operate without Covers.** To avoid electric shock or fire hazard, do not operate this product with covers or panels removed.
**Do Not Operate in Wet/Damp Conditions.** To avoid electric shock, do not operate this product in wet or damp conditions.
**Do Not Operate in an Explosive Atmosphere.** To avoid injury or fire hazard, do not operate this product in an explosive atmosphere.
**Do Not Operate with Suspected Failures.** If you suspect this product to be damaged, please contact Technical Support.

The following operating limits must be respected:

| Net class | Connector | Unit | Min | Max |
|---|---|---|---|---|
| Power | / | Voltage | 90 $V_{ac}$ | 250 $V_{ac}$ |
| USER IO | LEMO | Voltage | -0.1 V | 3.6 V |
| Digital lines | Digital 2.5V | Voltage (3.3 V I/O bank selected) | 0 V | 2.7 V |
| | | Current | | 10 mA |
| | LVDS | Voltage | 0.5 V | 3 V |
| | | Current | | 10 mA |
| | | Common Mode | 0.9 V | 1.75 V |
| | I2C/Serial | Voltage | 0 | 3.5 V |
| | RS485/RS422 | Voltage | 0V | +5V |
| Analog Differential Input | RJ45 | Voltage | | 2 Vpp |
| | | Common Mode | 0.5 V | 3.8 V |
| | | V+,V- Absolute Range | 0.5 V | 3.8 V |
| Analog SE Input | MCX | Voltage | | 1.5 $V_{pp}$ |
| | | Common Mode | 0.5 V | 3.8 V |
| | | V+,V- Absolute Range | 0.5 V | 3.8 V |

**Table 2.1:** operating limits for R5560SE connectors.

**WARNING:** the digital lines are directly connected to the FPGA I/Os. Violation in maximum absolute rating illustrated in this document will likely destroy the FPGA. There is no buffer or protection on this line. That is necessary because we want to preserve the possibility to operate at different voltages and with both single ended and differential signals.

**THIS DEVICE SHOULD BE INSTALLED AND USED BY SKILLED TECHNICIANS ONLY OR UNDER THEIR SUPERVISION**

# 3 Block Diagram

The R5560 is based on Xilinx Zynq-7000, a powerful System-on-Chip that uses a large programmable logic for massive real-time signal processing and a Dual Core 1GHz ARM processor for Ethernet communication and data post-processing. The board core is organized in 4 sections with **DAQ capabilities** and **Digital Interface**, each reading out 32 analog channels and hosting 32 configurable digital LVDS I/Os on VHDCI connector.

The R5560/R5560SE is organized as shown in **Figure 3.1**:

- **4 DAQ sections**: the DAQ section is the core of the system and it implements the analog-to-digital conversion, running the custom firmware generated by the SCI-Compiler. It is based on **Xilinx Zynq-7000**, a powerful System-on-Chip that uses a large programmable logic for massive real-time signal processing and a Dual Core 1GHz ARM processor for data post-processing. The DAQ section hosts readout interface (Optical Link, 1Gb Ethernet) and a JTAG USB connector, that allows for direct access to the FPGA for on-the-fly firmware upload and debug. The JTAG also gives access to the integrated USB-to-UART converter, which is useful to interface with a Linux console.

- **8 AFE sections (R5560SE only)**: each DAQ is connected to two **analog frontend boards (AFE)**. The first AFE manages the analog input channels from 0 to 15 while the second the channels from 16 to 31. The AFE's function is to amplify the signal, regulate the offset, select the impedance, the input division and the shaping of the signal.

- the **baseboard** is used to interconnect all the board sections. It features a system manager to route digital signals from the DAQ to the SYNC connectors and USER IOs (and vice versa). It hosts the USB3.0 and 1Gbps Ethernet ports for slow control and cumulative readout of the 4 DAQ sections.

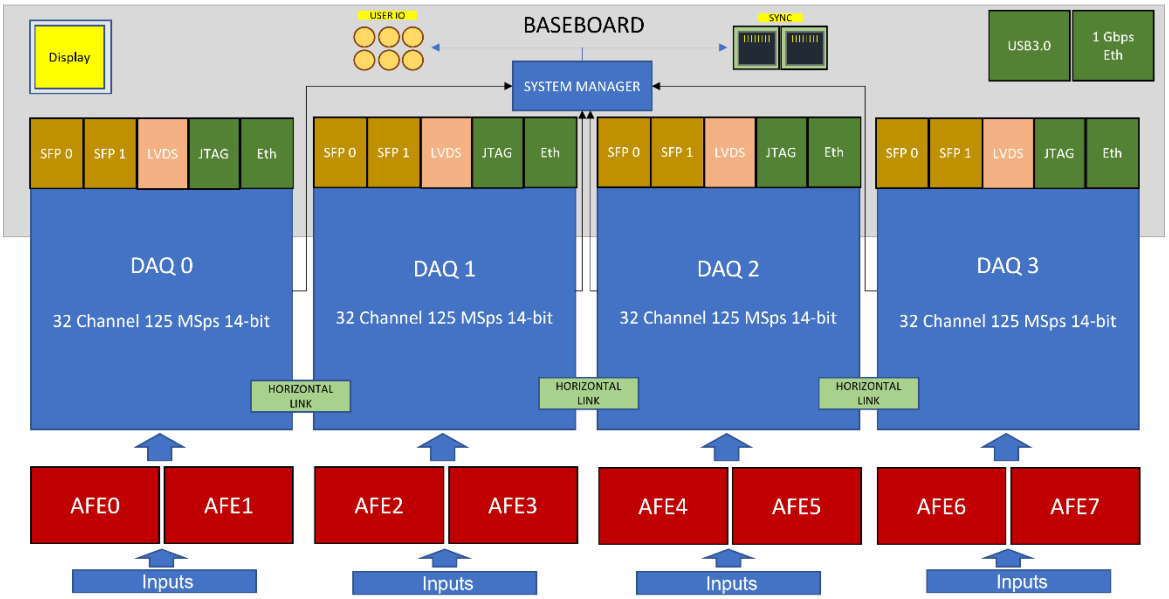



**Figure 3.1:** block diagram of the R5560/R5560SE. AFE sections are implemented in the R5560SE version only.

# 4 Technical Specifications

| TECHNICAL | | | |
|---|---|---|---|
| **POWER REQUIREMENTS** | **Voltage:** 100-240 $V_{ac}$ ± 10%<br>**Frequency:** 50/60 Hz<br>**Typ. Power consumption**: 0.8 A @ 220 $V_{ac}$<br>**Fuse**: 2A, 250V – 5 mm x 20 mm | | |
| **ANALOG INPUT – R5560** | **Channels**<br>128 Differential inputs<br><br>**Impedance**<br>$Z_{diff}$ = 100 Ω | **Connector**<br>32x CAT5e RJ45<br><br>**Full Scale Range**<br>2 $V_{pp}$ | **Bandwidth**<br>125 MHz |
| **ANALOG INPUT – R5560SE** | **Channels**<br>128 single-ended inputs on MCX<br><br>**Impedance**<br>50 Ω/1 kΩ programmable<br><br>**Bandwidth**<br>60 MHz<br><br>Programmable DC offset adjustment on each input in the full scale range | **Analog Coarse Gain**<br>[x1:x100] | **Full Scale Range**<br>[0.015 $V_{pp}$: 1.5 $V_{pp}$] |
| **DIGITAL I/Os** | **Channels**<br>4 x 32 Differential<br><br>**Impedance**<br>$Z_{diff}$ = 100 Ω | **Connector**<br>4x VHDCI<br><br>**Coupling**<br>DC | **Signal Type**<br>LVCMOS 2.5V<br>LVDS<br>BLVDS |
| **USER I/Os** | **Impedance**<br>50 Ω | **Coupling**<br>DC | |
| **DIGITAL CONVERSION** | **Resolution**<br>14 bits | **Sampling Rate**<br>125 MS/s Simultaneously on each channel | |
| **CLOCK GENERATION** | **125 MHz ADC clock**<br>Clock sources: internal/external<br>• Internal 25 MHz oscillator<br>• External 25 MHz – USER IN 0 or SYNC connector | | |
| **TRIGGER** | **Trigger Source**<br>*Internal/External*: managed by the default firmware<br>*Complex trigger logic*: implementable by the user on the open FPGA | **Trigger Propagation**<br>Through USER I/Os and Sync Connector<br><br>**Trigger Time Stamp**<br>*Default FW*: 32-bit counter, 8 ns resolution, 26-day range;<br>*Custom FW*: defined by the firmware design | |
| **SYNCHRONIZATION** | **Clock Propagation**<br>USER I/Os connectors<br>SYNC Connector | **Acquisition Synchronization**<br>Through programmable LEMO<br>Through dedicated SYNC Connector<br><br>**Sync connector allows to cascade multiple units and synchronize them with a single standard CAT5e cable** | |
| **FPGA** | **Open FPGA**<br>4x Xilinx Zynq-7000 SoC:<br>Z-7030 (R5560A, R5560SEA)<br>Z-7035 model (R5560B, R5560SEB) | | |
| **MEMORY** | 1 GByte of memory for list readout on each SoC<br><br>Up to 8kS/ch for simultaneous waveform readout | | |
| **COMMUNICATION INTERFACE** | **Ethernet (readout)**<br>4 x 1Gbps (4 Gbps cumulative speed)<br><br>**Ethernet (slow control)**<br>1 Gbps (slow control is an additional port, not required if the fast readout ethernet is used)<br><br>**Optical Link** | All readout interfaces allow to perform the same task at different speed and using different media. They can be used independently or simultaneously.<br><br>The different readout interface allows to integrate the R5560SE in existing experimental environment. | |

| | | |
|---|---|---|
| | Slots for 8 x 10Gbps SFP+ transceivers (communication protocol not implemented by default)<br><br>**USB 3.0**<br>1x USB 3.0 readout<br><br>**PARALLEL TTL/LVDS Readout**<br>Readout through the VHDCI digital I/O on custom protocol | |
| **FIRMWARE** | **Default**<br>-Waveform recording and Pulse Height Analysis<br>- Ethernet communication | **Custom**<br>Use SCI-Compiler to develop your own firmware!<br><br>Firmware can be upgraded via Ethernet or USB 2.0 debugger (on-fly) |
| **SOFTWARE** | • **Open Hardware Readout Software** to manage the default firmware<br>• **SCI-Compiler** for custom firmware development | |
| **MECHANICAL** | | |
| **FORM FACTOR** | R5560 - 19", 2U Rack-mount<br>R5560SE - 19", 3U Rack-mount | |
| **DIMENSIONS (H/W/L)** | R5560 - 88/482.0/367.0 (396 with handles) mm<br>R5560SE - 132.5/482.0/366.0 (399.8 with handles) mm | |
| **ENVIRONMENTAL** | | |
| **ENVIRONMENTAL** | Indoor use | |
| **OPERATING TEMPERATURE** | Operating Temperature -20 °C ÷ 50 °C | |
| **OPERATING HUMIDITY** | 25% ÷ 95% RH non condensing | |
| **STORAGE TEMPERATURE** | -30 °C ÷ +80 °C | |
| **STORAGE HUMIDITY** | 5% ÷ 90% RH non condensing | |
| **ALTITUDE** | ≤2000 m | |
| **POLLUTION DEGREE** | 2 | |
| **OVERVOLTAGE CATEGORY** | II | |
| **EMC ENVIRONMENT** | Commercial and light industrial | |
| **IP DEGREE** | IPX0 enclosure, not for wet location | |
| **REGULATORY** | | |
| **COMPLIANCE** | • EMC: CE 2014/30/EU Electromagnetic compatibility Directive<br>• Safety: CE 2014/35/EU Low Voltage Directive | |

**Table 4.1:** technical specifications for the R5560 and R5560SE.

# 5 Packaging and compliancy

The R5560/R5560SE is available as 19” rackmount module housed in an aluminium case.

The unit is inspected by CAEN before the shipment, and it is guaranteed to leave the factory free of mechanical or electrical defects.

When receiving the unit, the user is strictly recommended to inspect for any damage which may have occurred during transportation. Particularly, inspect for exterior damages like broken connectors and check that the panel is not scratched or cracked.

All packing material should be held on until the inspection has been completed. If damage is detected, the user must file a claim with the carrier immediately and notify CAEN.

Before installing the unit, make sure to read thoroughly the safety rules and installation requirements (Sec. **Safety Notices**), then place the package content onto your bench.

The content of the delivered package standardly consists of the part list shown in the table below (

**Table** 5.1**)**. All the official documentation, firmware updates, software tools, and accessories are available on *www.caen.it* at the product web page.

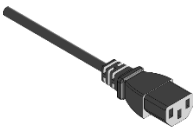
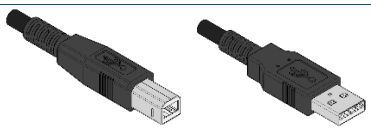
| | Part | Description | Qty |
|---|---|---|---|
|  | R5560/R5560SE | 128 Ch. 14 bit 125 MS/s Digitizer | x1 |
|  | Power supply cable | Standard C13 power supply chord | X1 |
|  | USB cable | USB 2.0 141/5HS type A-B I/O cable L=2MT | x1 |
|  | Ethernet cable | ETHERNET CAT6 cable L=2MT | X4 |
|  | SCI-Compiler | USB Dongle for SCI-Compiler software **(if included in your board purchase order)** | X1 |
|  | User guide | UM6952 – R5560/R5560SE User Manual | x1 |

**Table 5.1:** delivered kit.

**CAUTION:** to manage the product, consult the operating instructions provided.

It is recommended to:

- Inspect containers for damage during shipment. Report any damage to the freight carrier for possible insurance claims.
- Check that all the components received match those listed on the enclosed packing list. (CAEN cannot accept responsibility for missing items unless we are notified promptly of any discrepancies.)
- Open shipping containers; be careful not to damage contents.

- Inspect contents and report any damage. The inspection should confirm that there is no exterior damage to the unit such as broken knobs or connectors and that the front panel and display face are not scratched or cracked. Keep all packing material until the inspection has been completed.
- If damage is detected, file a claim with carrier immediately and notify CAEN service.
- If equipment must be returned for any reason, carefully repack equipment in the original shipping container with original packing materials if possible. Please contact CAEN service.
- If equipment is to be installed later, place equipment in original shipping container and store in a safe place until ready to install

**DO NOT SUBJECT THE ITEM TO UNDUE SHOCK OF VIBRATIONS**

**DO NOT BUMP, DROP OR SLIDE SHIPPING CONTAINERS**

**DO NOT LEAVE ITEMS OR SHIPPING CONTAINERS UNSUPERVISED IN AREAS WHERE UNTRAINED PERSONNEL MAY MISHANDLE THE ITEMS**

**USE ONLY ACCESSORIES WICH MEET THE MANUFACTURER SPECIFICATIONS**

# SCI-Compiler License

R5560SE is compatible with SCI-Compiler ([RD1][RD2]), the Windows-based CAEN ***firmware generator and compiler*** for ***easy FPGA programming.*** It is an **automatic code generator** that, starting from a graphical block diagram, generates a VHDL piece of code that implements the required function. The software uses a prebuilt library set containing macroblocks implementing complex functions (MCA, Oscilloscope, Digitizer, TDC) that the user can connect one with each other. SCI-Compiler is also able to generate C libraries and drivers to be used in **Windows**, **Linux** and **macOS** for DAQ software implementation.

**Note:** SCI-Compiler full version works upon the purchase of a license, contact CAEN at info@caen.it for more information.

**Note:** the board needs a runtime license to run SCI-Compiler-based firmware, contact CAEN at info@caen.it for more information.

**Note:** a 30D trial version of SCI-Compiler is available for free at www.sci-compiler.com.

Full version of SCI-Compiler is accessible after activation of the **software license** on CAEN website using the SERIAL NUMBER and an ACTIVATION key provided together with the USB Dongle. The USB Dongle must be plugged into the PC to run SCI-Compiler full version. The generated firmware can be uploaded on each compatible board that has been activated with a **SCI-Compiler Runtime license** on CAEN website. Refer to **[RD1]** and **[RD2]** for more details.


**Figure 5.1:** SCI-Compiler USB Dongle and keys for license activation.

**Note:** user is not allowed to use the code generated by the SCI-Compiler on boards different from R5560SE. Even using a small part of the code generate by the SCI-Compiler on a custom design board or other products is an explicit violation of the license terms and it is an offense against CAEN S.p.A and Nuclear Instruments S.R.L.

# 6 PID (Product Identifier)

PID is the CAEN product identifier, an incremental number greater than 10000 that is unique for each product. The PID is on a label affixed to the product (**Fig. 6.1**) and it is even stored in an on-board non-volatile memory readable via touchscreen, Web Interface or Readout Software (see Chap. **Touchscreen Display Guide**, **Web Interface** and **Open Hardware Readout** Software).


**Fig. 6.1:**PID location taking a CAEN rack unit as an example (the number in the picture and the device model are purely indicative)

# 7 Power Requirements

The R5560/R5560SE is powered by an internal 100/240V-12V AC/DC stabilized power supply.

The power supply chord is included in the delivered kit and it is connected to equipment using an IEC C14 connector on the back panel. The power chord is a standard wall IEC C13.

# 8 Cooling Management

The R5560/R5560SE board can operate in the temperature range -20 °C ÷ 50 °C.

Air flow fans are installed onboard. The user must take in care to provide a proper cooling to the board with external fan if the board is used in an enclosure or if the board is installed in a setup with poor air flow.
Excessive temperature will, in first instance, reduce the performance and the quality of the measurements and can also damage the board.

The unit dissipates almost 100W. Please do not close the rear fan holes to avoid unit overheating. If in a single rack tower, multiple units are installed, please consider external fans or rack mounted air conditioning system.

If the board is stored in cold environmental, please check for water condensation before power on.

The board has not been tested for radiation hardness. High energy particle can be source of soft error and can damage the FPGA. If used in strong proton or neutron beams, arrange proper shielding or remote the sensor with a custom cable.

## Cleaning the air vents

CAEN recommends to occasionally clean the air vents on all vented sides of the board. Lint, dust, and other foreign matter can block the vents and limit the airflow. Be sure to unplug the board before cleaning the air vents and follow the general cleaning safety precautions.

# 9 Installing the device

The device should be installed in 19” rack as detailed in the following steps:

1. Ensure you have the correct bolts needed to secure the device in your rack (bolts not included)
2. Carefully slide the system into an available 3U space in your 19-inch rack.
3. Screw the four locking bolts
4. Connect power supply chord to the AC power supply C14 inlet connector on the rear panel

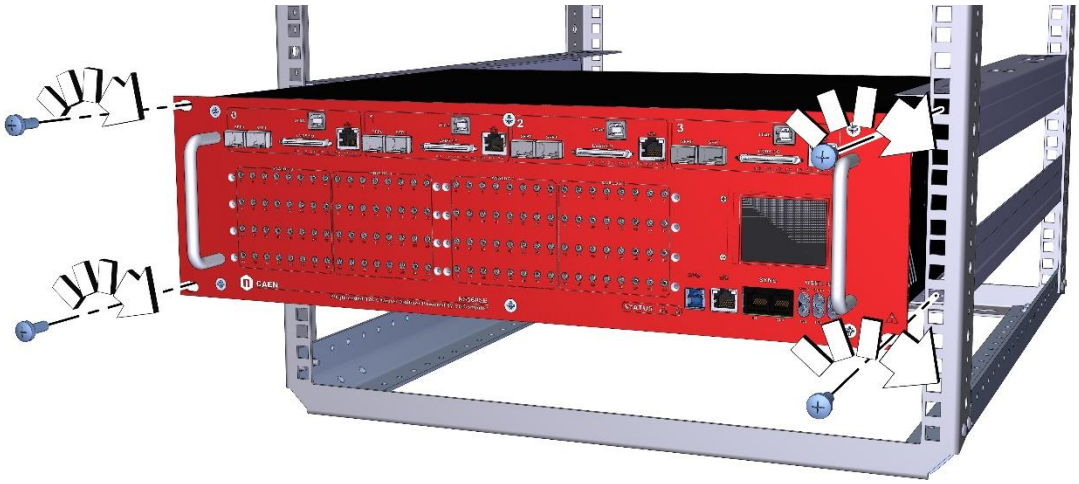


**Figure 9.1**: installing the device into a 19” rack

**ONLY QUALIFIED PERSONNEL SHOULD PERFORM INSTALLATION OPERATIONS**

**DO NOT INSTALL THE EQUIPMENT SO THAT IT IS DIFFICULT TO ACCESS THE SWITCH AND POWER CONNECTION ON THE BACK PANEL**

**IT IS RECOMMENDED THAT A CIRCUIT-BREAKER IS NEAR THE EQUIPMENT**

**THE SAFETY OF ANY SYSTEM THAT INCORPORATES THE DEVICE IS UNDER THE RESPONSIBILITY OF THE ASSEMBLER OF THE SYSTEM**

Do not use the device and contact technical support if one of these situations is verified:

- Enclosure integrity is compromised
- Insulation of HV chord is damaged (if present)
- The indication led or display is not performing as required (e.g. led not working, display with incorrect graphic)
- Fans are not working (if present)

# 10 Panels Description

## R5560 dimensions

The R5560 unit is housed in a standard 2U 19” rackmount aluminium box.

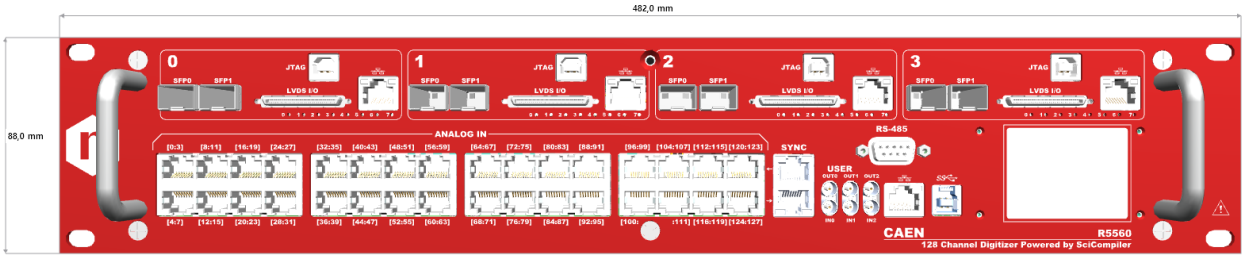



**Figure 10.1**: R5560 front panel view and dimensions.


**Figure 10.2**: R5560 rear panel view and dimensions.

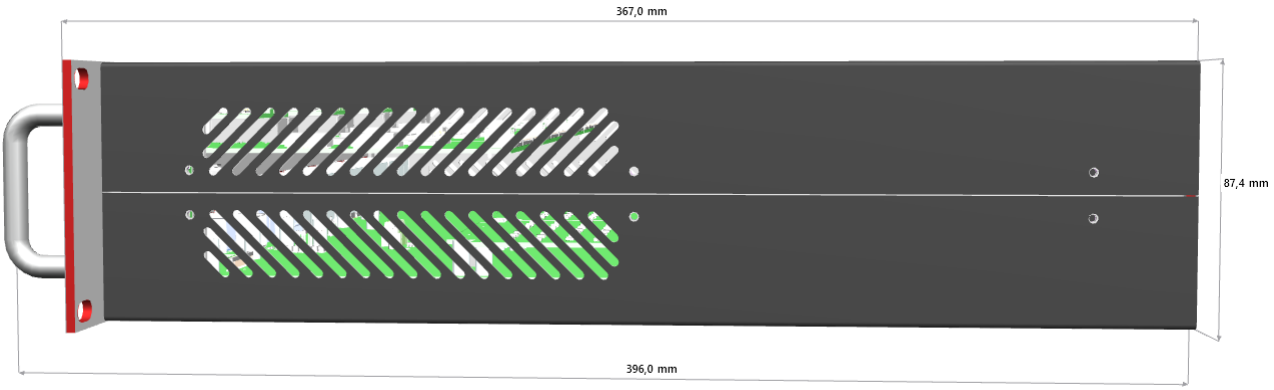



**Figure 10.3**: R5560 side view and dimensions.

# R5560SE dimensions

The R5560SE unit is housed in a standard 3U 19” rackmount aluminium box.

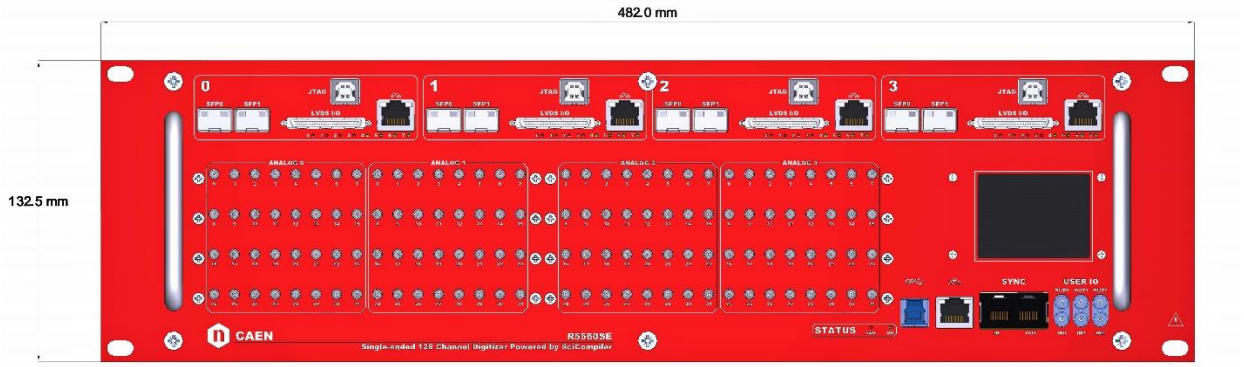



**Figure 10.4**: R5560SE Front panel view and dimensions.

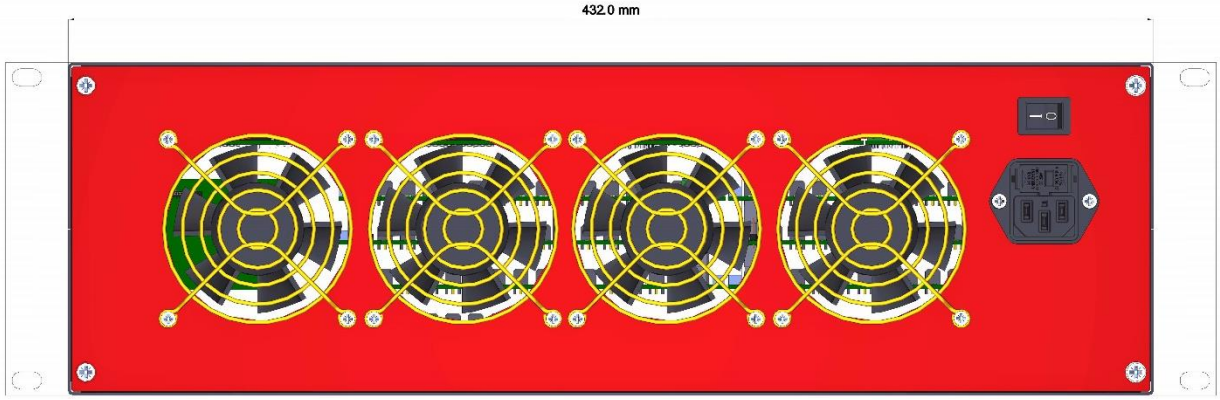



**Figure 10.5:** R5560SE Rear panel view and dimensions.

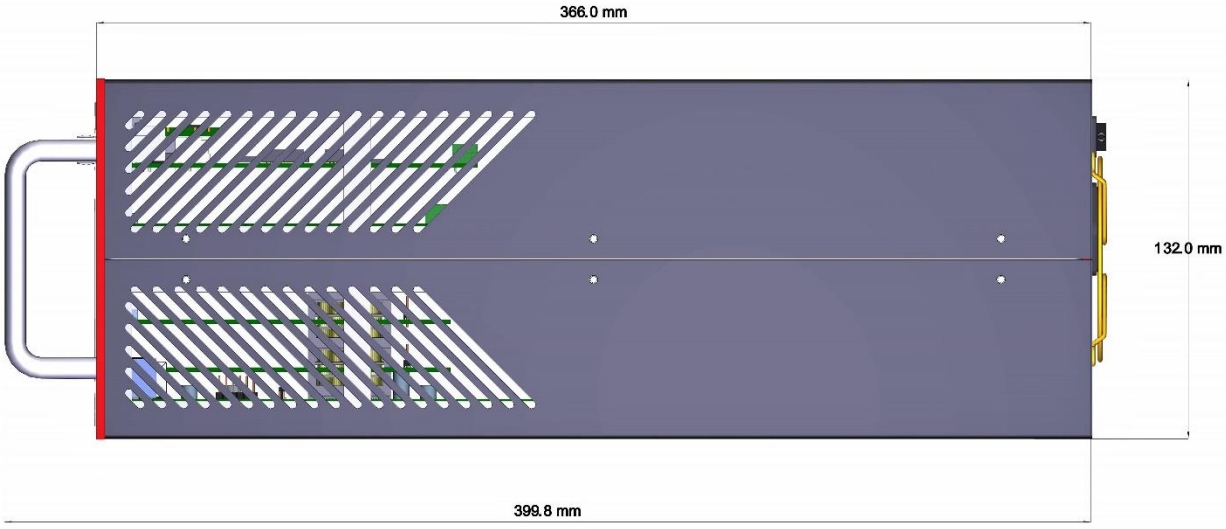
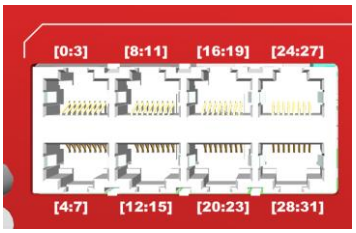



**Figure 10.6:** R5560SE side view and dimensions.

# Front Panel

## DIFFERENTIAL ANALOG INPUT – R5560 only




**FUNCTION**

8 x RJ45 connectors carrying 32 analog differential signals. Compatible with standard Ethernet cable CAT5e with standard ethernet pinout

**ELECTRICAL SPECS**

Sign. type: differential (Analog 0...3V Common Mode, 2Vpp differential signal).

Coupling: DC.

$Z_{diff}$: 100 Ω.

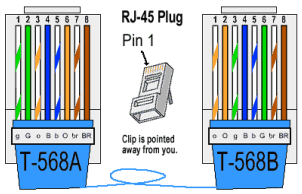
**MECHANICAL SPECS**

Series: RJ45.

Suggested plug: Ethernet crimp plug

Suggested cable: Cat5e or Cat6 ethernet cable

**PINOUT**




## SINGLE-ENDED ANALOG INPUT – R5560SE only


**FUNCTION**

Input connectors from CH0 to CH31 on four different sections. They receive the single-ended input analog signals.

**ELECTRICAL SPECS**

Input dynamics: [1.5 : 0.015] $V_{pp}$

Input impedance: 50Ω / 1kΩ

**MECHANICAL SPECS**

Series: MCX

## JTAG


**FUNCTION**

Connector for direct FPGA access in order to perform fast firmware download and FPGA signals monitor and probing

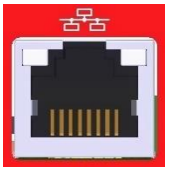
**ELECTRICAL SPECS**

Standard: compliant with USB 2.0

**MECHANICAL SPECS**

Series: USB 2.0 - B

## ETHERNET



**FUNCTION**

1 Gbps Ethernet interface

- 4 ports for high-speed readout, one for each section
- 1 port for slow control

⚠ Slow control communication is not implemented in the default firmware

**ELECTRICAL SPECS**

*Not available*

**MECHANICAL SPECS**

Series: RJ45

## OPTICAL LINK

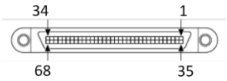
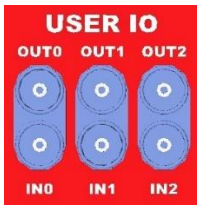
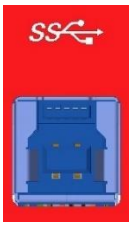
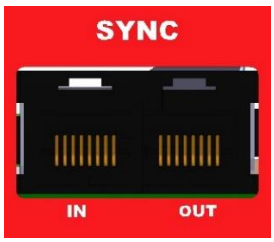

**FUNCTION**

Slot for SFP+ connector. 10 Gbps optical transceiver can be inserted here

⚠ SFP+ transceivers are not provided with the board kit

**MECHANICAL SPECS**

Series SFP+ socket

| | | |
|---|---|---|
| | ⚠ Optical link communication is not implemented in the default firmware | |
| **LVDS I/O** | | |
|  | **FUNCTION**<br>68-pin connector carrying 32/64 differential input digital signals<br>**ELECTRICAL SPECS**<br>Sign. type: differential (LVCMOS 2.5V, LVDS, BLVDS).<br>$Z_{diff}$: 100 Ω | **MECHANICAL SPECS**<br>Series: VHDCI connector.<br>Type: 71430-0008<br>Manufacturer: TE Connectivity<br>Suggested plug: 68-pin VHDCI<br>Suggested cable: SCSI-5 type.<br>**PINOUT**<br>  |
| **LEDs** | | |
|  | **FUNCTION**<br>8 user-configurable LEDs<br>**ELECTRICAL SPECS**<br>*Not available* | **MECHANICAL SPECS**<br>Not available |
| **USER IOs** | | |
|  | **FUNCTION**<br>General purpose digital I/O connectors. Their function is defined at firmware level<br>**ELECTRICAL SPECS**<br>NIM/TTL | **MECHANICAL SPECS**<br>Series LEMO |
| **USB 3.0** | | |
|  | **FUNCTION**<br>USB 3.0 connector for data readout<br>⚠ USB 3.0 communication is not implemented in the default firmware<br>**ELECTRICAL SPECS**<br>Standard: compliant with USB 3.0 | **MECHANICAL SPECS**<br>Series USB 3.0 - B |
| **SYNC** | | |
|  | **FUNCTION**<br>2 x RJ45 connector carrying 3 sync (CLOCK, T0, RUN) differential signals for each connector (IN/OUT).<br>**ELECTRICAL SPECS**<br>Signal type: differential (LVDS)<br>Coupling: DC<br>$Z_{diff}$ : 100 Ω | **MECHANICAL SPECS**<br>Series: RJ45.<br>Suggested plug: Ethernet crimp plug<br>Suggested cable: Cat5e or Cat6 ethernet cable |
| **RS-485 preamplifier box (R5560 only)** | | |
|  | **FUNCTION**<br>RS485 interface to remote control the pre-amplifier gain/shaping time<br>⚠ RS4585 communication is not implemented in the default firmware | **MECHANICAL SPECS**<br>Series: D-SUB9. |

**ELECTRICAL SPECS**

*Not available*

**STATUS LEDs**


**FUNCTION**

FAIL: if ON, firmware not correctly uploaded onboard

OK: if ON, board correctly operating

**MECHANICAL SPECS**

Not available

**DISPLAY**

**FUNCTION**

2.4” Touchscreen Display for board control and monitoring.

**ELECTRICAL SPECS**

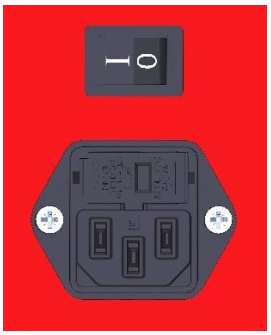
*Not available*

**MECHANICAL SPECS**

*Not available*

# Rear Panel

**AC 220V and ON/OFF switch**




**FUNCTION**

Input connector for the R5560SE main power supply (100-240 V) and ON/OFF switch

**ELECTRICAL SPECS**

*Not available*

**MECHANICAL SPECS**

Series: AC power supply C14 inlet connector

# 11 Functional Description

The R5560/R5560SE is a flexible board meant to be used as a fully programmable readout system for different applications. In the following we give a detailed description of the board hardware and its operation.

## System Architecture

The R5560/R5560SE integrates four System-on-chip with configurable FPGA and dual core ARM-A9 processors. The FPGA of each chip is devoted to the signal acquisition and real-time processing while the ARM processor can run C/Python applications that operate on the data pre-processed by the FPGA. This allows to execute complex real-time algorithm that requires to operate on less data but with several operations. For these purposes, a large quantity of memory is available to store list mode data (up to 1 Gbyte per processor).
For example, it is possible to implement in the FPGA triggers and pulse shape discrimination algorithm Neutron/Gamma discrimination. On the processor side, it is possible to implement, for instance, a Neural Network that performs Neutron/Gamma discrimination with a separation capability much higher than the algorithm operating in real-time in the FPGA. For sure, the Neural Network will never be able to sustain MHz rates, while the FPGA will easily manage this amount of data. When the FPGA is unable to distinguish a pulse, it is possible, for example, to transfer that particular pulse to the ARM processor for a more advanced analysis. The 90% of the events will be processed in real-time by the FPGA while the 10% of them will be queued in a buffer for further investigation by the ARM processor.

Each ARM processor controls one dedicated 1 GBps ethernet link, allowing extremely easy development of custom communication interface. The development of the FPGA processing algorithm can be done with SCI-Compiler software, while the development of the ARM part of the firmware can be done in Eclipse using the available tools provided with the Virtual Machine.

The R5560SE is a built basing on several sub-boards:

- **Base board**: it hosts all connectors, the main power supply circuits, the system monitor and the logic for synchronization. A SoC manages the slow control, the interface between the DAQ modules, the generation and distribution of the clock and sync signals. The board controls also a touch screen display with a simple GUI to configure and control the system.
- **4 DAQ boards**: this is the programmable core of the system. Each one digitizes and processes in real-time 32 independent channels. Each section hosts a large and powerful SoC based on Zynq processor (7030 or 7035 model) 14-bit 125 MSPS ADCs, 1 Gbps Ethernet link, two optical link, LVDS I/Os and an integrated a USB 2.0 JTAG.
- **8 AFE boards (R5560SE only)**: this board perform signal conditioning in order to convert the single-ended inputs into differential ones. It controls the signal shaping, gain, offset and input impedance

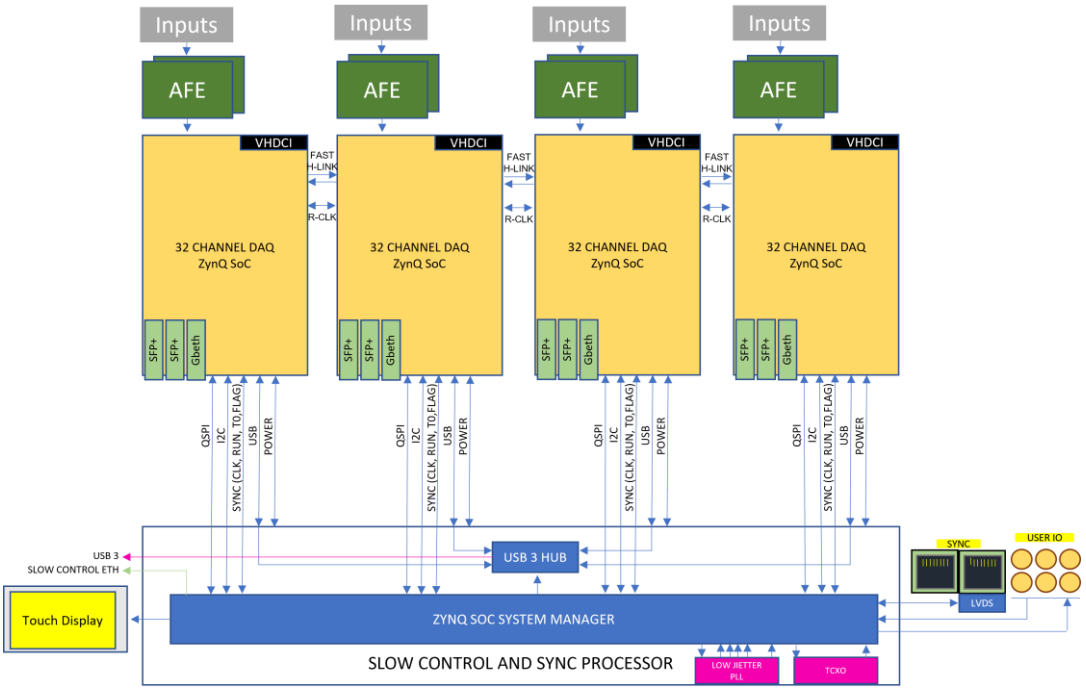



**Figure 11.1:** the complete block diagram of R5560SE. The R5560 works the same without the AFE boards.

**Figure 11.1** shows the full scheme of the R5560SE. Each DAQ board is an independent unit (with independent firmware) processing up to 32 analog channels. The analog signals come from the AFE board as a differential signal with $2V_{pp}$ dynamic and 1 V common mode. The ADCs convert independently each channel and provide the digital streams as input to the SoC. The SoC manages the real-time processing and high-speed communication between external facility through Gigabit Ethernet or Optical Link. Each DAQ section can exchange data with nearby sections (DX, SX) with a dedicated horizontal link (H-LINK). The bidirectional H-LINK link offers 12.5 Gbps bandwidth and allows a significant calculation load reduction in the readout facility.

**Note:** please find more details about the H-LINK management at **https://github.com/NuclearInstruments/r5560-daq-horizontal-interconnections**

The Slow Control and Sync Processor (SCSP) is a SoC that manage the synchronization between multiple R5560SE and an external system. Moreover, the SCSP is connected with slow speed bus to all modules (I2C, 100Mbps QSPI bus). This link allows the SCSP to collect data from each of the four DAQ sections, configure the memory-mapped registers and read all memory-mapped FIFO or memories instantiated in the firmware. The SCSP has an Ethernet connector and exposes a server that allows to readout data and set/get any register from a single connection. The bandwidth is limited to 100Mbps even if the link is a 1Gbps link, and it is meant to be used for debug purposes.

A USB 3.0 hub interconnect together all the SoCs offering a 400Mbit/s readout link for section on a single USB 3.0 connector.

A touch screen display allows to monitor the status of the system, environmental parameters, set and read ethernet configuration of all DAQ board and of the SCSP board.

# DAQ section

The R5560/R5560SE is organized in four DAQ sections, each one featuring 32 analog inputs for a total 128 readout channels. Each section hosts 4 ADCs operating in simultaneous sampling at 125 MS/S - 14 bit.
The frontal panel clearly identifies the analog channels, grouping them with the relative digital signals. The different sections can easily communicate each other thanks to fast horizontal links.

Each DAQ board hosts a Xilinx Zynq-7000 (7030 or 7035 model), a powerful System-on-Chip that uses a large programmable logic for massive real-time signal processing and a Dual Core 1GHz ARM processor for data post-processing. The user can implement here the needed custom firmware, eventually exploiting SCI-Compiler tool. The firmware is stored on the 1GB flash memory while 1GB DDR3 works as a primary memory for Linux and as cache for the data to be transferred from the FPGA to the LAN. The ARM microcontroller features all the software required to control and configure the system and to implement Ethernet communication.

On each DAQ board, there are also 2 optical links, capable to reach up to 10Gbit/s data transfer. The optical link protocol can be configured by the user to interface the unit with external readout systems. The DAQ section also hosts a USB JTAG connector, in order to allow for on-the-fly firmware uploading and USB-to-UART bridge access. In this way it is possible to directly access the Linux console of the Zynq. The Zynq also has a USB interface connected to a USB connector (that is on the base board, in the bottom part of the instrument) that emulates a LAN over USB, in order to be able to read out and configure the R5560/R5560SE.

Each DAQ section is connected to the baseboard through 6 digital signals: 3-input sync-in and 3 output sync-out. These lines connect the FPGA of the Zynq to the FPGA on the baseboard. The user can configure the Zynq signals to be connected to LEMO I/Os or to the SYNC signal connectors.

The DAQ section is the part of the device in charge of the analog-to-digital conversion. The ADCs accept the analog signals coming from the input connectors. In the case of the R5560, the differential analog input directly feed the ADCs, while in the R5560SE, each differential input of the ADCs is driven by the output of the analog frontend (AFE) section – see Par. **Analog Frontend - R5560SE –** that, in turns, accepts single-ended analog input signals.

The ADCs receive the clock or from the on-board generator or from the baseboard. The baseboard allows selecting different clock sources, for example, from a LEMO connector, dedicated SYNC connector or internal baseboard clock.

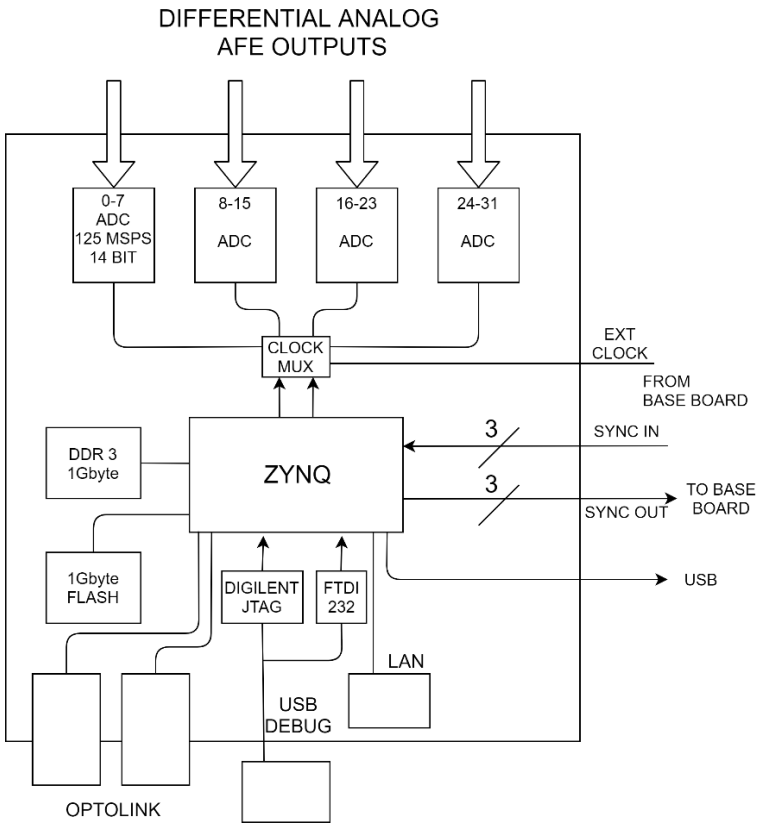
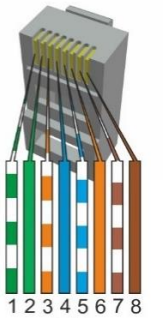



**Figure 11.2:** block diagram of a single DAQ section of the R5560/R5560SE. Signals interconnecting the DAQ with the baseboard and AFE boards are shown.

## R5560 Differential Inputs

The R5560 has four groups of 8x RJ45 shielded connectors to carry analog signals, for a total of 128 differential analog inputs. Differential signals are used to minimize the crosstalk between channels and the noise, even with long cables carrying the analog inputs from the detectors. The pinout of the analog connector is shown in **Figure 11.3**. This allows to use standard CAT5e pre-crimped cables to connect differential signals to the board.




| PIN CABLE | COLOR | Board Ch/Polarity |
|---|---|---|
| 1 | White-Green | 2+ |
| 2 | Green | 2- |
| 3 | White-Orange | 4+ |
| 4 | Blue | 3+ |
| 5 | White-Blue | 3- |
| 6 | Orange | 4- |
| 7 | White-Brown | 1+ |
| 8 | Brown | 1- |

**Figure 11.3**: RJ45 analog connector pinout: The pinout for the first group of channels [1:4] is shown. The same ordering can be applied to the subsequent groups.

The R5560 inputs have the following characteristics:

- Bandwidth: 60 MHz
- Gain: 1 V/V
- Differential Input Dynamic: 2 $V_{pp}$
- Any common mode voltage is accepted if within the absolute maximum rating.
- Absolute Input Dynamic: 0.5 V … 3.8 V
- Differential Input Impedance: 100 Ω
- Single Ended Impedance: 50 Ω
- 50µV integrated noise

As previously said, the R5560 accepts differential analog input signals, which have several advantages:

- Immunity to common mode noise
- Immunity to channel cross-talk
- Immunity to RF noise
- Immunity to ground-loops
- Immunity to ADC clock coupling
- With limited signal swing, it is possible to achieve a larger input dynamic.

Differential signals can be directly connected to the R5560 RJ45 input connectors, according to the input parameters reported in **Table 11.1**. Both V_P and V_N must be connected.

| SYMBOL | DESCRIPTION | MIN | MAX |
|---|---|---|---|
| VCM | Input common mode | 0.5 | 3.5 |
| VDIFF | Input Differential signal | 0 | 2Vpp |
| V_P | Input positive terminal | 0.5 | 3.5 |
| V_N | Input negative terminal | 0.5 | 3.5 |
| RIN | Input impedance | | 100Ω |

**Table 11.1:** differential parameters for the R5560 analog input.

## Analog Frontend - R5560SE

The R5560SE features an advanced programmable analog frontend. Onboard, there are 8 AFE sections (two for each DAQ section) managing the single-ended analog inputs and routing the signals to the ADCs of the DAQ section.

The analog frontend converts the input single-ended analog signal to a differential one, that fits at best the input dynamic range of the ADC converter. There are several stages inside the AFE board, as shown in **Figure 11.4**.

- The first block allows the selection of the **input impedance** between 50 Ohm and 1 kOhm
- The second block allows dividing the signal by 5, if needed
- The third block allows to **shape** the input signal selecting among 1, 10, 30 µs shaping time or to bypass the shaper using DC coupling. The output of the shaper is connected to the offset block.
- The **offset** block allows to add or subtract from the input signal baseline any value between -2 V and +2 V
- The **gain** stage allows selecting the gain to apply to the input signal between 1 and 100. Possible gain values are distributed on a logarithmic scale and can be selected via software. The output of the gain stage is a differential signal that is connected to the ADC.

The analog input channels are grouped in couples. Each pair of channels shares the configuration of the input impedance, division and gain, while the offset can be selected independently on each channel. The shaper can be configured at instrument level, meaning that the shaping time is the same on all channels.

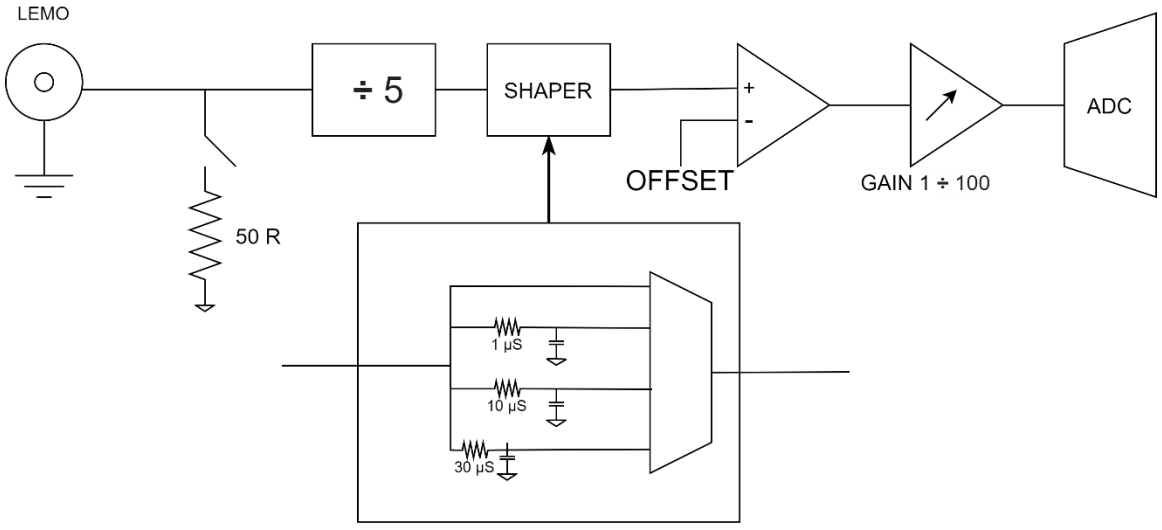



**Figure 11.4:** block diagram of the AFE section of the R5560SE.

# Digital Stage

The digital stage is available on VHDCI connectors (LVDS I/Os) on the front panel or LEMO I/Os.

Each VHDCI connector carries:

- 32 differential lines supporting the standards LVDS, BLVDS (or PSEUDO LVDS)
- 58 single ended CMOS 2.5V from DATA1_P/_N to DATA29_P/_N
- 6 single ended CMOS 1.8V from DATA30_P/_N to DATA32_P/_N
- 4x ground pin

The pinout of the digital VHDCI connector is shown below.

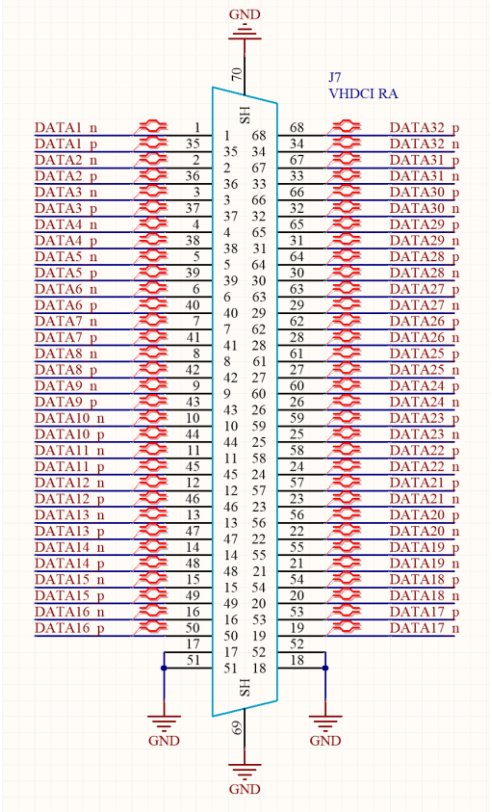



**Figure 11.5:** pinout of the LVDS digital connectors of the R5560SE.

Digital I/Os on the VHDCI connector can be used for several applications, for example:

- Fast Parallel readout (protocol supported by SCI-COMPILER)
- Input for triggers/veto
- Input from photon counting detector / ASIC
- Readout of digital ASIC both with serialized output or parallel output
- Control of the readout process of analog ASIC

Using SCI-Compiler, is possible to select the I/O standard and the pin direction (IN/OUT). Pin direction and signal standard must be selected at configuration time and cannot be changed in real-time.

**WARNING:** To ensure fast communication (up to 200MHz DDR, 12.8Gbps) no buffer or over-voltage protection are present between FPGA and I/O. Short circuits or over-voltage will damage the FPGA.

**WARNING:** In single ended mode the last 6 signals (DATA_30, 31 and 32) are 1.8V voltage ONLY. CMOS2.5 or CMOS3.3V will damage the FPGA.

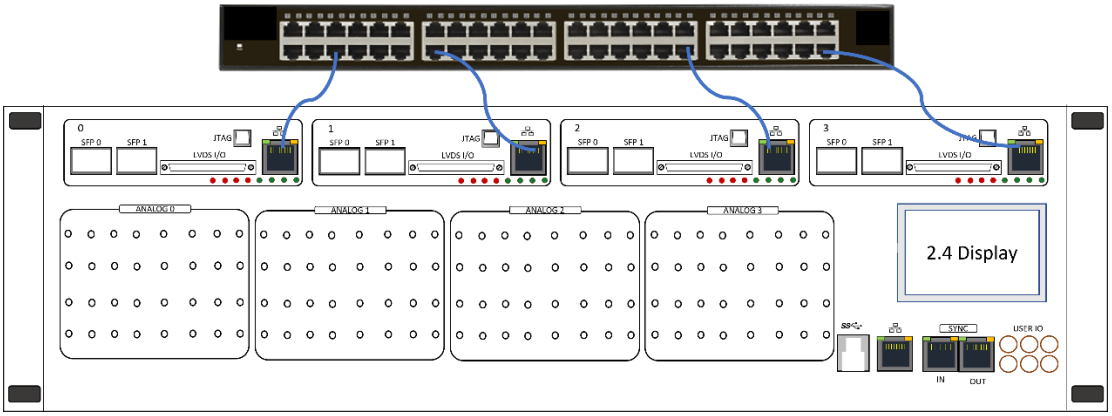
Moreover, it is possible to use the general purpose digital I/O LEMO connectors to drive NIM/TLL signals to be used for different purposes. The function of the LEMO I/Os is defined at firmware level and can be customized by the user by exploiting open FPGA features via SCI-Compiler software.

# Data readout

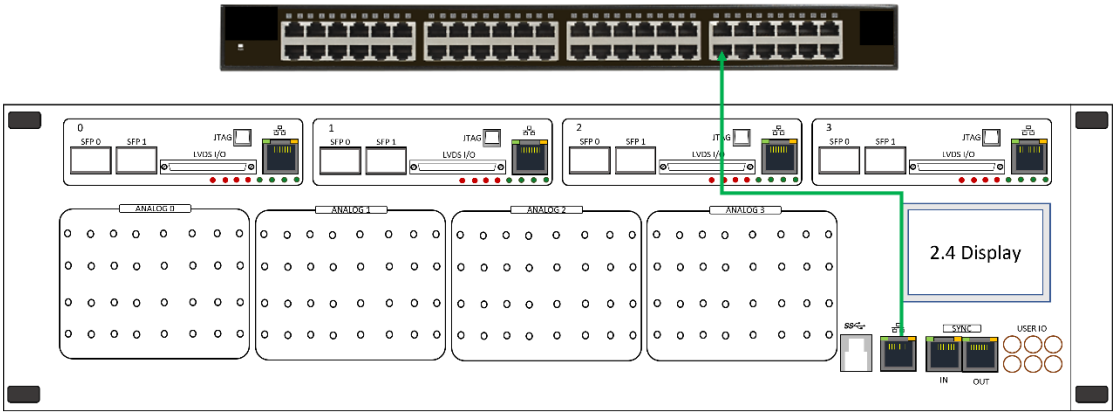
The R5560/R5560SE is designed to provide the **maximum flexibility in experiments requiring hundreds of readout channels.** Both the wide range of available communication protocols and the scalable architecture are designed to help users in building a multi-board system and in **integrating it in existing facilities**.

There are several ways to readout data from the R5560/R5560SE:

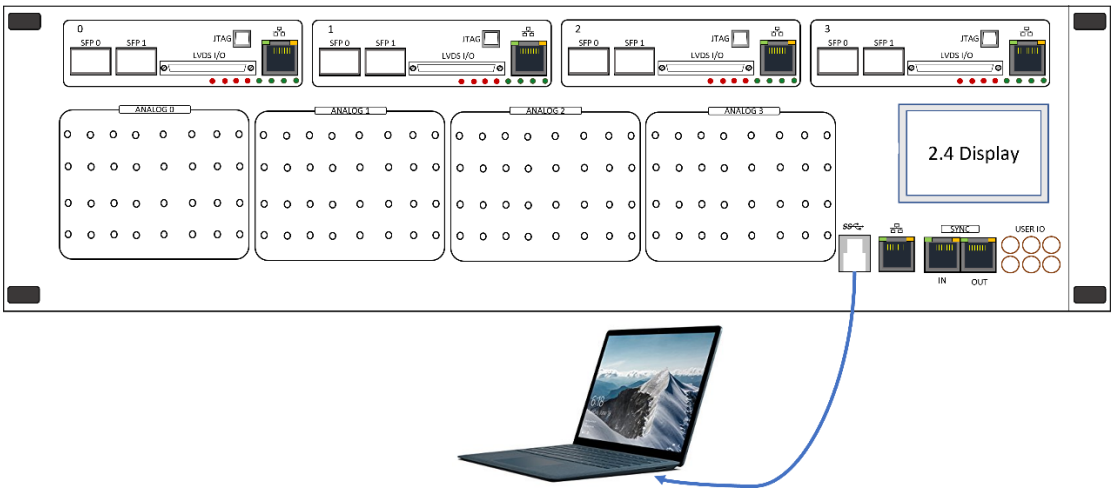
1) READOUT USING DAQ SECTION ETHERNET PORT
   Each DAQ section has a dedicated ethernet link connected to the PS (processor side) of the SoC. This Ethernet port is fully managed by the user application that implements a custom protocol using TCP/IP or UDP sockets. A basic application is provided as example to customize the protocol (see **Appendix** for more details)
   A multi-port 10Gbps or an Ethernet switch is required in order to perform the R5560SE readout.




2) READOUT USING SLOW CONTROL ETHERNET PORT
   The slow control ethernet port allows to implement a low-speed readout. With a single ethernet port it is possible to readout data from all channels and set/get registers. Please consider that everything can be readout using this system, except the fast FIFO managed by the DMA.



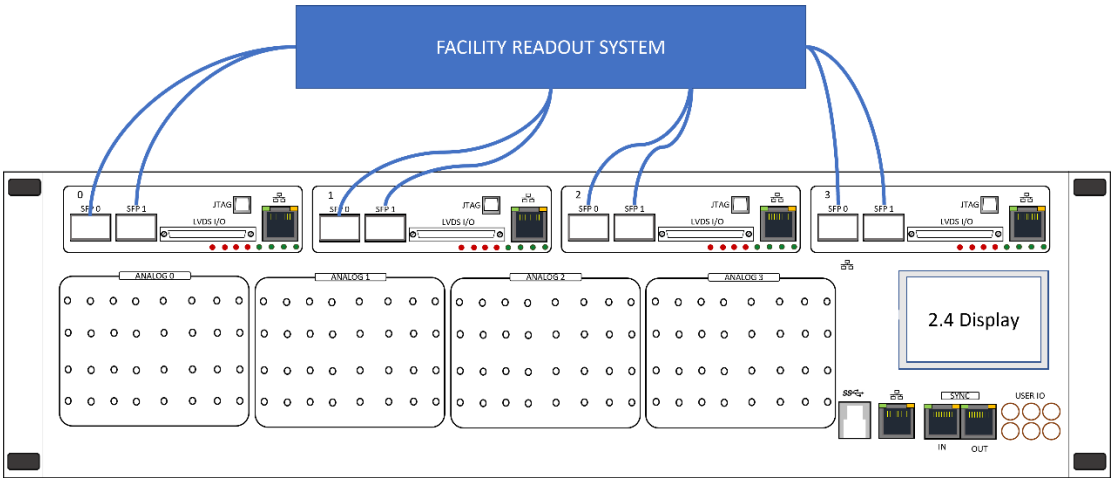

3) READOUT USING THE USB 3.0
   The SoC exposes a virtual ethernet card on the USB 3.0 connector using RNDIS protocol. Connecting the USB 3.0 cable the user will enumerate 5 new ethernet interfaces on the computer. Each interface is a virtual ethernet card with an IP address (configurable on the display). Max speed 300 Mbit/s per section.




4) READOUT USING OPTICAL LINK

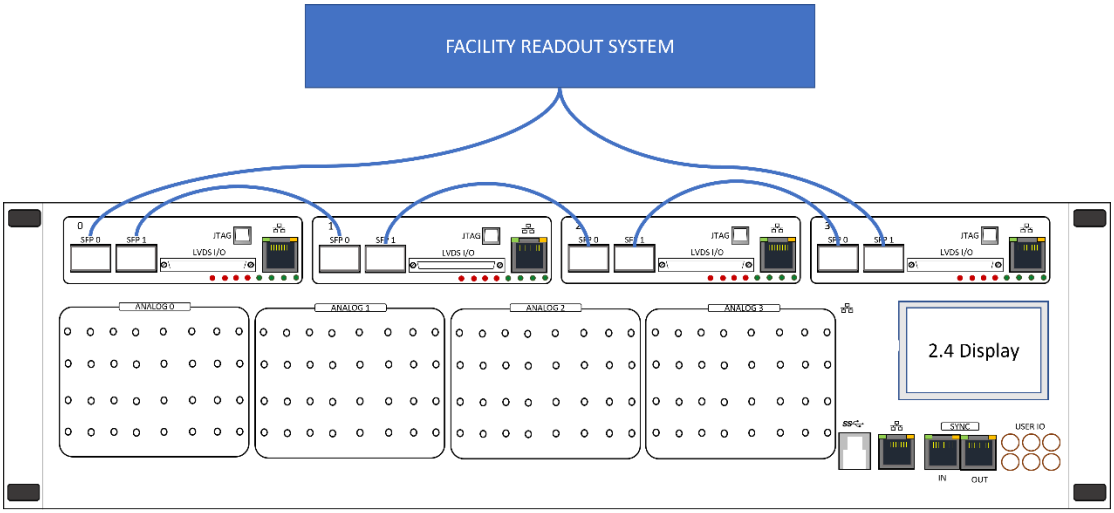
The optical link allows to obtain the maximum bandwidth from the R5560SE. Aggregating the 8 links at 12.5 Gbps, it is possible to reach a maximum readout speed of 100Gbps.

A free open-source protocol is available and can be used to interface with external facilities (see **Appendix** for more details).



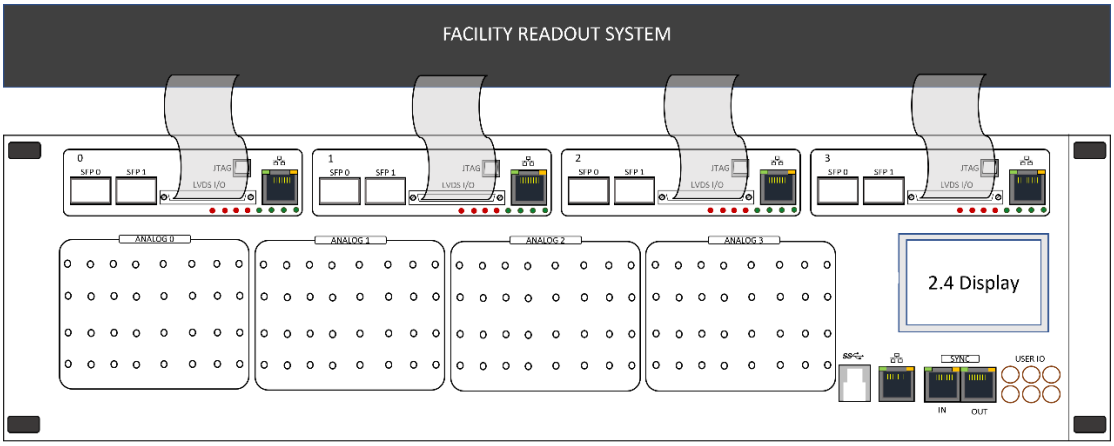

5) OPTICAL DAISY CHAIN

Multiple R5560SE units can be daisy-chained in order to reduce the number of optical links. In daisy-chain mode the total bandwidth of the chain must be less than 12.5 Gbps.




6) PARALLEL READOUT

The R5560SE can be readout in parallel using the LVDS interface. The LVDS interfaces offers a 32-bit wide bus. User must develop its own protocol using the available SDK.




# Clock Distribution

The R5560/R5560SE has a **very versatile clocking network** (see **Figure 11.6**). It is possible to use different clock sources for the ADCs:

- the internal clock, generated on the base.
- the internal clock, generated on the DAQ board.
- external clock

The DAQ board selects the 125 MHz clock to be provided to the ADCs between its internal clock and the clock provided by the baseboard. This is done by a selector (SEL) controlled by the FPGA. The FPGA is driven by the firmware loaded in the Zynq, overwriting any setting that has been configured via the web interface or via the display. The SEL signal can be configured using the SCI-Compiler. When SEL is 0, the default value of the clock is provided to the ADC by the Zynq itself. When SEL is 1, the Zynq is bypassed, and the clock is provided directly by the baseboard.

The baseboard accepts and route to the DAQ a clock signal, following these available options:

- 25 MHz internal oscillator
- LEMO IN 0 connector
- pin A1/A2 of the SYNC IN connector (see **Figure 11.7**)

If the clock is provided to the baseboard externally by the LEMO 0 or the SYNC connector, the frequency must be 25 ± 0.5 MHz, using a different frequency requires an external VCO/PLL.
The clock signal fed into the baseboard, is converted by the PLL passing from 25MHz single-ended into a 125MHZ LVDS clock.

- The 25 MHz input clock can be routed to the LEMO OUT 0 or to the pin A1/A2 of the SYNC OUT connector (see **Figure 11.7**) on the front panel of the R5560SE

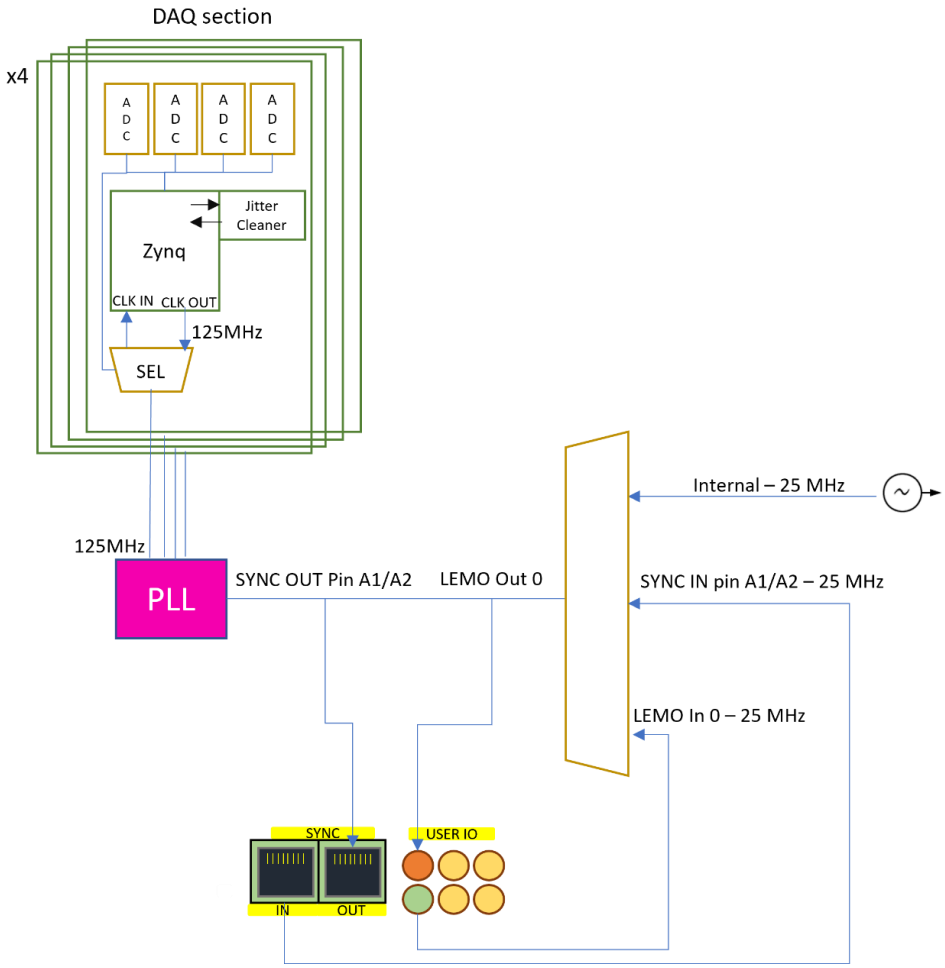



**Figure 11.6:** scheme of the clock distribution on the R5560/R5560SE

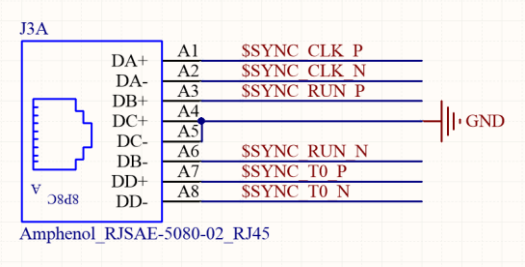



**Figure 11.7:** pinout of the SYNC RJ45 connector of the R5560SE

# Synchronization among multiple boards

The R5560/R5560SE has been designed to operate in large readout systems. Multiple R5560/R5560SE can be connected in such a way that they can operate in **simultaneous sampling** with a common time reference to be able to accurate timestamp events.

Synchronization operates at two levels:

- ONLINE: a system manger time stamping logic with a common clock and reset
- OFFLINE: the event builder software in the DAQ uses the time stamp to merge events coming from different board in a single event.

Synchronization requires the following signals:

- **Common Clock**: ensures that all ADCs operate in phase exactly at the same frequency. It also guarantees that internal time counter and TDC will be clocked at the same frequency, and they will be synchronous.

- **Run**: is a common signal generated by the master board and indicate the start of a run. Global timestamp and global event counter will be reset on the rising edge of this signal. All events will be timestamped with counters those zero by this signal. Event builder in the readout system will merge different events coming from multiple R5560SE using this unique timestamp.
- **T0/flag** is a user signal that can be used to reset secondary timestamp counter or to mark an event. This signal is optional, and it is used to make a measurement with respect to a periodic zero signal (I.E., a beam signal). The function of T0 is configurable in Sci-Compiler

**The synchronization between boards can work using a single RJ45 cable that connect in daisy chain multiple instruments through the SYNC IN/OUT port on the front panel.** The cable carries all synchronization signals including: CLK (25MHz), RUN and T0 on LVDS differential signals. A standard ethernet CAT5e cable can be used for board-to-board synchronization. In order to set the boards to properly operate using this single-cable synchronization, refer to Chap. **Touchscreen Display** Guide and **Web Interface**.

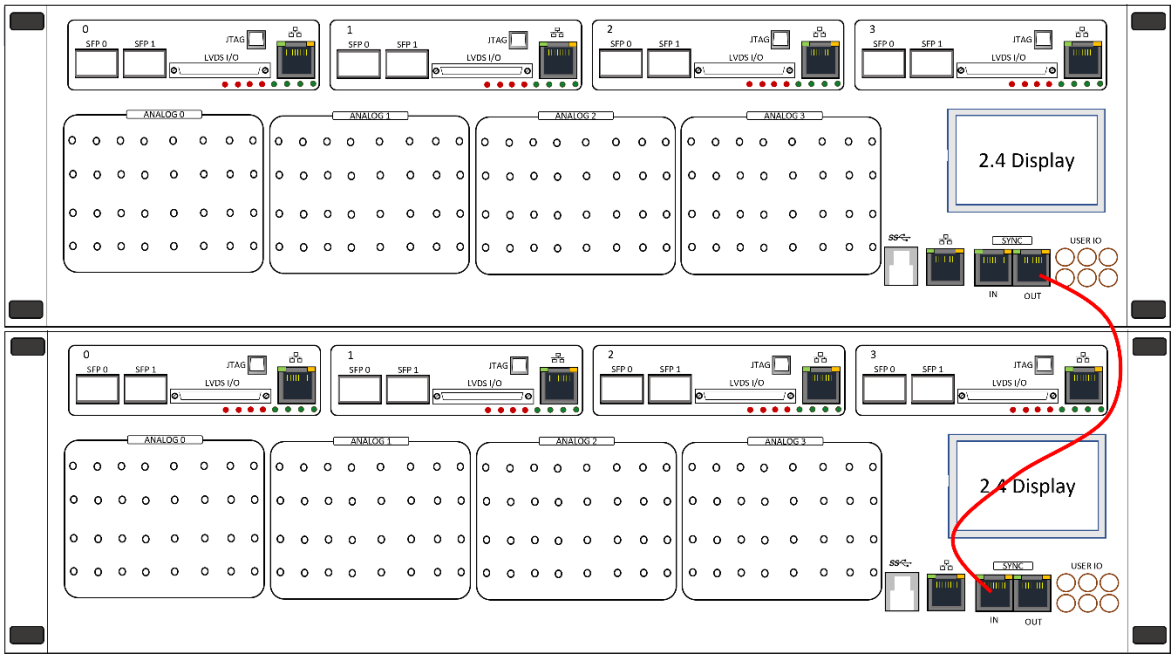



**Figure 11.8:** synchronization connection between two R5560SE boards using a single Ethernet cable. Slave board (bottom) is receiving synchronization signals on SYNC IN connector from the master board (top) SYNC OUT.

As an alternative to distribution of needed signals through the SYNC IN/OUT connectors, on the frontal panel there are also three USER IN and three USERS OUT on LEMO connectors. The function can be selected by the configuration web page of the SCSP among:

- Clock reference
- T0
- Start of run
- Flag/Beam
- Next Frame (useful in imaging application, where is necessary to accumulate counting for a fixed period)

# DAQ Datapath

R5560/R5560SE boards can be readout using multiple interface: Ethernet, USB, Optical Link and H-LINK. All these connection interfaces should be able to access to the same data in the PL. To do this, a router is used to commit a R/W request from one of the interfaces to the PL and complete the R/W operation.

SCI-Compiler works with the logic of memory mapped resources. Every endpoint in the FPGA (register, digitizer, oscilloscope, frame transfer, list transfer, ….) is mapped at a particular address of the PL in an address space 32 bit wide. Each endpoint has a memory size multiple of a power of two: a register for example has a memory size of 1, a FIFO of 1, an oscilloscope of 0xFFFF.
The DAQ is always a slave of a master system. Master system could be the PC client that connect via ethernet to the server running on the CPU (PS) of the Zynq, or the facility readout system in more complex systems. The master starts a transfer operation with a short packet addressing the salve board and specify operation, address in the slave and, for

write operation, the payload. The router catches the packet and check the board address. If the board address matches with board id or with the general call (0xFFF) the packet is processed. If the packet comes from an optical link and does not match the board ID or the packet is a general call (0xFFF) the packet is forwarded to the TX optical link and in the H-LINK.

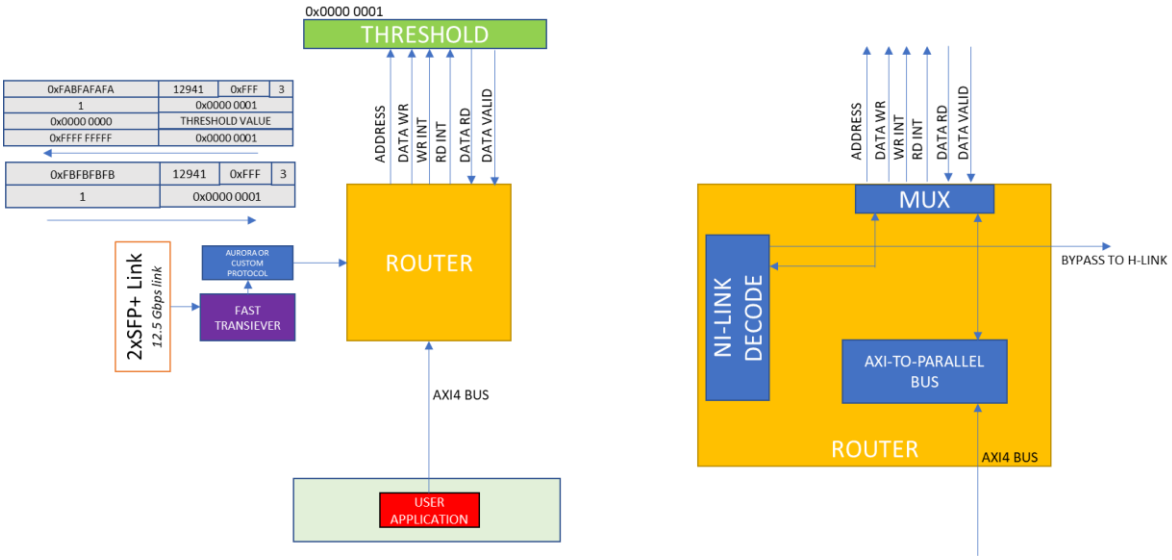



**Figure 11.9:** example of the router operation for data transfer.

The router also manages the H-LINK. H-LINK appears in the User Programmable Logic as a four-FIFO (two TX and two RX). User has a TX/RX couple of FIFOs to send/receive data from SX side and a couple for the RX side. The user can just push data in the correct FIFO and the framework will transfer data to the nearby DAQ board.

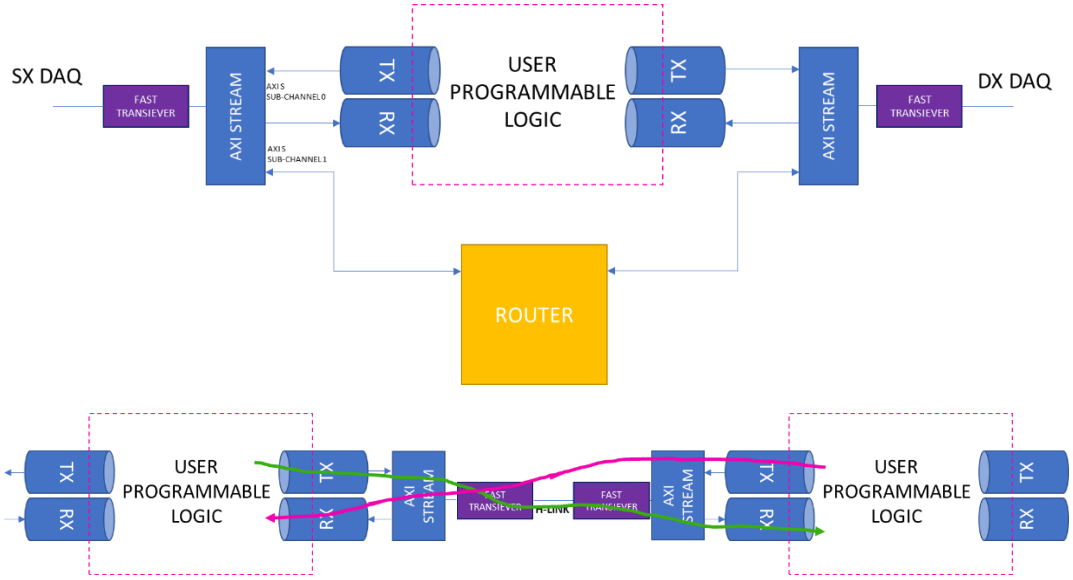



**Figure 11.10:** data transferring of two R5560SE sections, using the router or the H-LINK.

# 12 Touchscreen Display Guide

The R5560/R5560SE features a 2.4” Touchscreen display for basic board configuration. In the following, screenshots are taken from a R5560SE as a reference, but the given description is valid for R5560 too.

**DO NOT USE SHARP METAL OBJECTS, SUCH AS TWEEZERS OR SCREWDRIVERS, OR POINTED OBJECTS, SUCH AS PENS OR PENCILS, TO TOUCH THE DISPLAY**

The main window (see **Figure 12.1**) shows the PID (Product Identification Number) of the board and the current app version installed.

By tapping on the SETTINGS button, it is possible to access a series of tabs where the user can configure the IP address of the board and select the routing configuration for signals like the Clock source and SYNC signals.


**Figure 12.1:** the main page of the Touchscreen display.

## Ethernet settings

The ETHERNET tab allows the user to configure the IP address of the baseboard and DAQ sections. To change the IP configuration:

- Tap on SETTINGS from the main page and use the arrows on the top bar to surf to the ETHERNET tab. Then tap on BASE or the DAQ section number to set the IP address of the baseboard and of the DAQ sections respectively.
- Assign an IP, Gateway (GW), Netmask (NM) and DNS in a valid range for your network to be able to reach the instrument from your computer. For example, if your computer IP address is a 10.128.1.1, type an IP that ranges from 10.128.1.2 to 10.128.1.254 and the mask (NM) must be 255.255.255.0
- It is possible to enable the DHCP to automatically obtain the IP from the DHCP server in the network or it is possible to assign manually the IP by clicking on the keyboard icon on the right of each field.
- Select the firmware of interest and press “Save” button in the bottom right corner of the screen.
- The user will be notified if the IP has been successfully applied. Reboot of the board is needed for the changes to become effective.

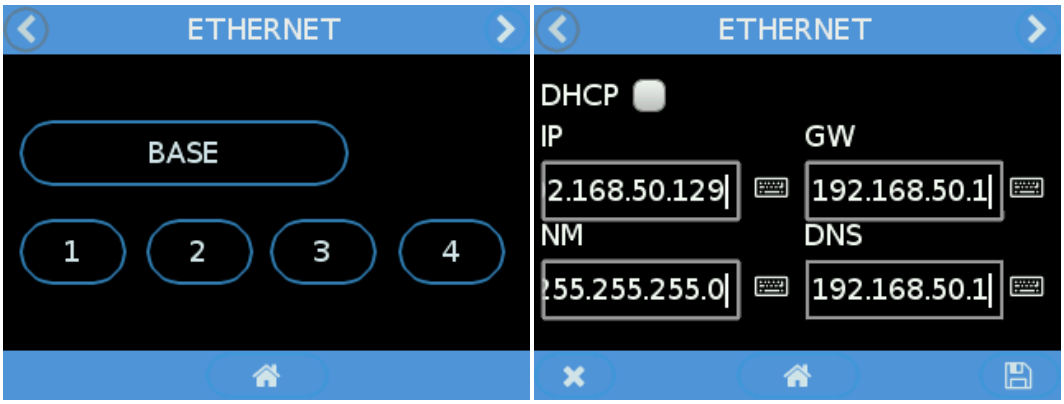



**Figure 12.2:** the Ethernet tabs on the touchscreen display.

# Signal routing settings

In the SETTINGS menu, it is also possible to configure the routing of some interal digital signals of the board, in order to configure clock distribution and synchronization:

- Tap on SETTINGS from the main page and use the arrows on the top bar to surf to the CLOCK INPUT, DAQ SYNC, SYNC OUT, LEMO OUT or PULSER tabs.

- In the CLOCK INPUT tab, it is possible to select the **clocking source** choosing among INTERNAL, LEMO IN 0 or SYNC CONNECTOR. If LEMO IN 0 is selected, the clock must be a 25 MHZ TTL signal. If SYNC CONNECTOR is selected it must be a 25 MHz LVDS signal connected to pin A1/A2 of the SYNC IN RJ45 connector. Refer to Par. **Clock Distribution** of this manual to better understand the clocking strategy of the board. . For standard synchronization using SYNC IN/OUT connectors daisy chain (see Par. **Synchronization among multiple boards**), please choose **SYNC CONNECTOR** option for each of the slave boards

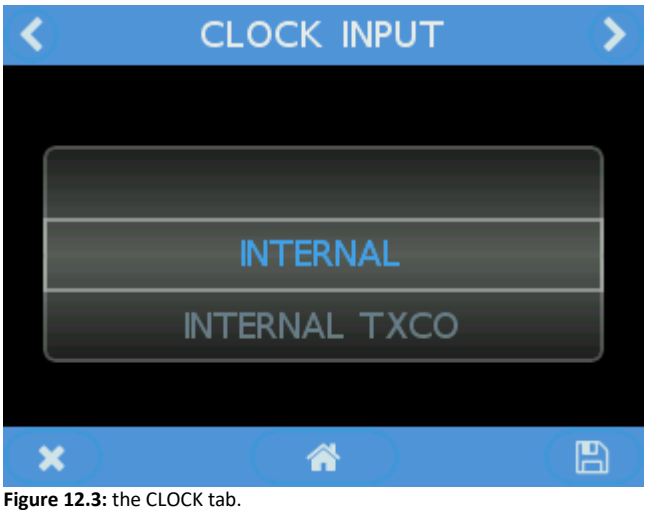



**Figure 12.3:** the CLOCK tab.

- The DAQ SYNC tab allows to select the source for the synchronization signals used by the board. Synchronization needs three independent signals to be routed to the DAQ board (CLOCK, RUN, T0). It is possible to select between LEMO IN, SYNC IN and Internal Pulser source for each of the three available lines. For standard synchronization using SYNC IN/OUT connectors daisy chain (see Par. **Synchronization among multiple boards**), please choose **SYNC IN** option for each of the slave boards. If using Internal Pulser source for one of the three signals, the remaining two ones can be driven via register using the option SYNC REG

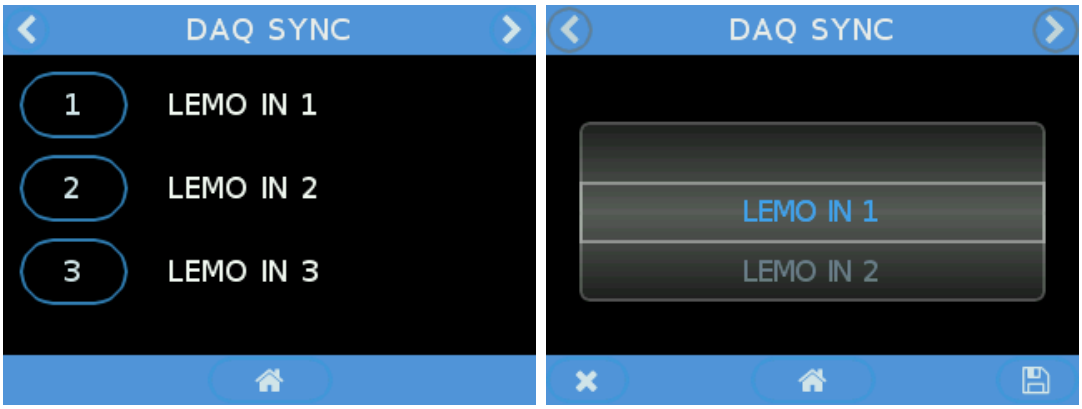



**Figure 12.4:** the DAQ SYNC tab.

- SYNC OUT tab allows to set the signals carried on the RJ45 SYNC OUT connector placed on the front panel of the instrument. This connector can carry three independent signals that allows the synchronisation of two or more boards. It is possible to select between LEMO IN, SYNC IN, and INTERNAL PULSER for each of the three available lines. For standard synchronization using SYNC IN/OUT connectors daisy chain (see Par. **Synchronization among multiple boards**), please choose SYNC-IN option.

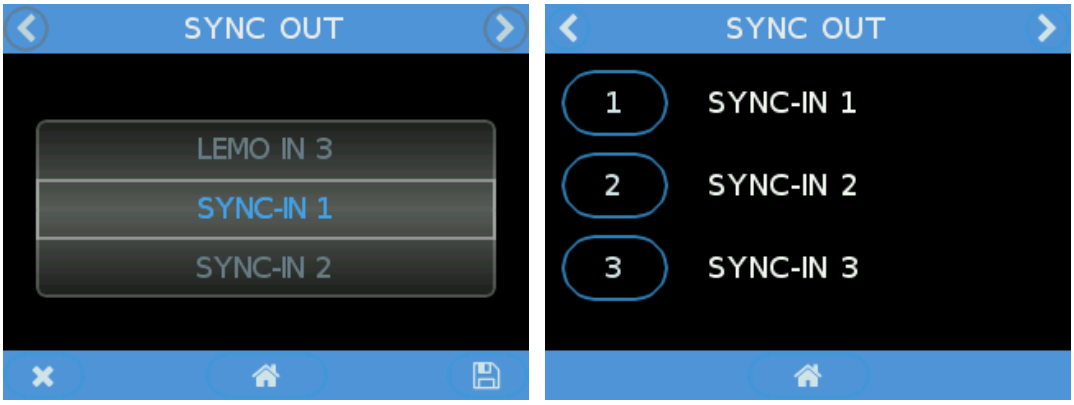



**Figure 12.5:** the SYNC OUT tab.

- LEMO OUT tab allows to select which signals are to be routed on the front panel three USER Outputs; it is possible to select between LEMO IN, SYNC IN, INTERNAL PULSER and SYNC REG

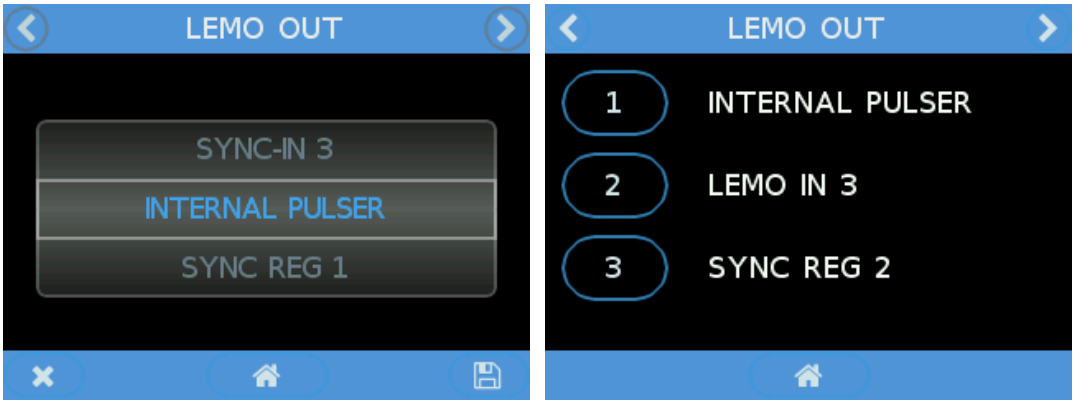



**Figure 12.6:** the LEMO OUT tab.

- PULSER tab allows to set the internal pulser to be used in combination with CLOCK INPUT = Internal. Frequency and width of the pulser can be chosen in the dedicated fields.

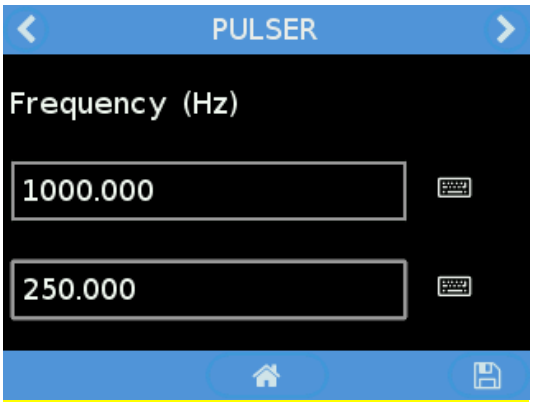



**Figure 12.7:** the LEMO OUT tab.

# Info and status

From the TEMPERATURE, DAQ LAN STATUS and DAQ INFO tabs, it is possible to check:

- the DAQ boards and baseboard temperature
- the DAQ boards and baseboard IP address
- the  name of the firmware loaded on each DAQ boards


**Figure 12.8:** the TEMPERATURE, DAQ LAN STATUS and DAQ INFO tabs.

# 13 Web Interface

The R5560/R5560SE comes with an embedded web interface that allows the user to configure the analog front end parameters, upload new firmware and upgrade the instrument OS.

To connect to the web interface:

- Connect the instrument to the PC via Ethernet or USB. It is possible to connect the baseboard or the single DAQ boards individually using the dedicated connection ports of each section.
- If using Ethernet, set your PC to be in the same network of the instrument.
- Open a web browser and type the IP or USB address as shown on the instrument display, according to the chosen physical connection (IP refers to Ethernet connection)

## Baseboard Web Interface

### Connection

To access the specific Web Interface of a DAQ section, connect to the board by using the Ethernet port as shown below.

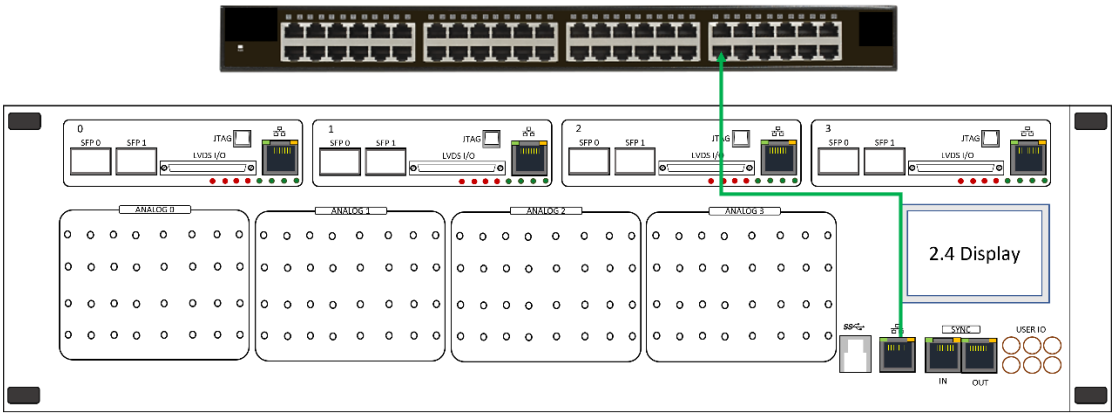



**Figure 13.1:** connection scheme to access the baseboard Web interface. The baseboard port can be connected to the network or directly to a PC Ethernet port.

### Device Status and firmware upgrade

In the Device Status (Home) window it is possible to monitor:

- the board IP address
- the board PID
- the temperature of the base board
- the baseboard firmware release (i.e., the OS of the system)
- the IP of each DAQ board
- the firmware loaded on each DAQ board (name and release), i.e., the pulse processing algorithm.
- the temperature of the DAQ board
- the temperature of the analog boards (AFE section of the R5560SE)

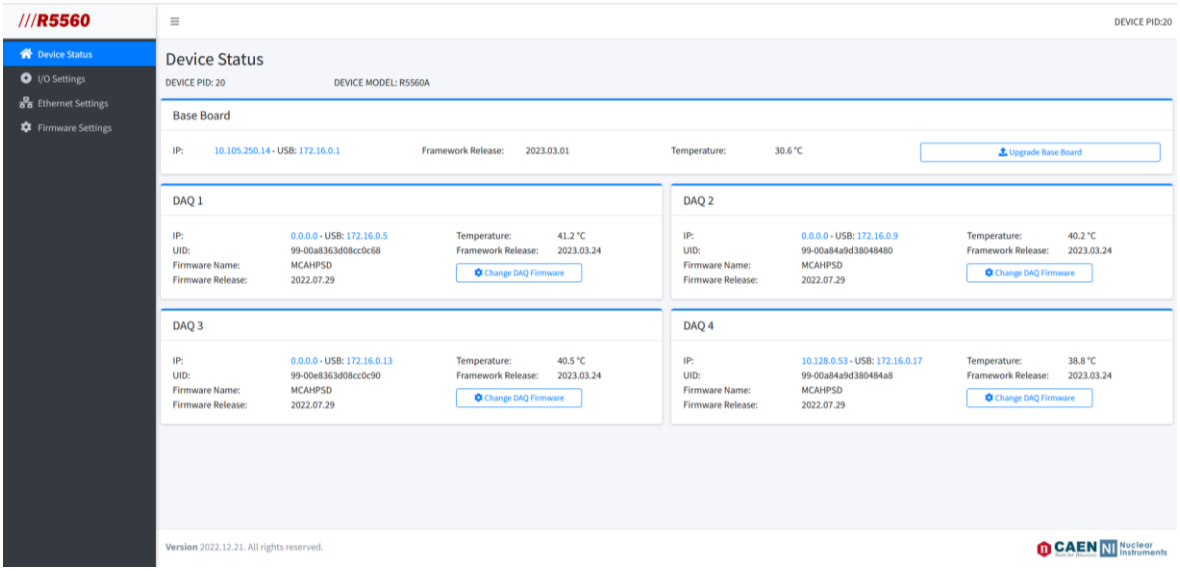



**Figure 13.2:** the Home page of the baseboard Web interface.

If a newer version of the OS running on the ARM process is released on the CAEN website, it is possible to upgrade the device by clicking on *Upgrade Base Board*. The opening page will ask the user to select a **.bsu file** previously downloaded from the CAEN website. Press Upgrade button to perform the loading of the new OS version and follow the given instructions.

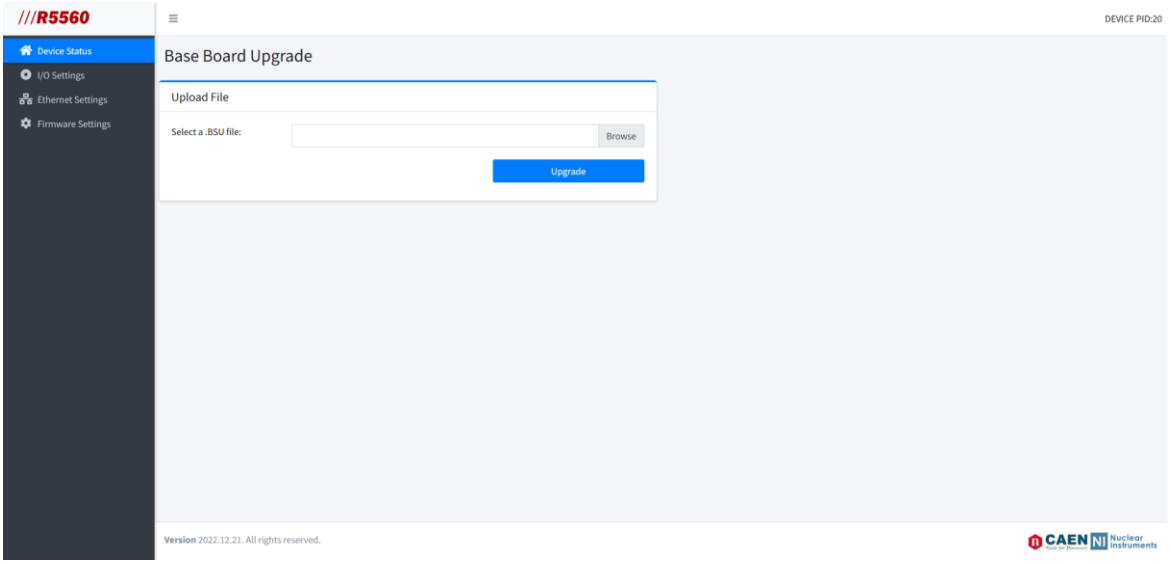



**Figure 13.3:** upgrade of the OS of the board from the baseboard Web interface.

## Analog settings

The Analog Settings window allows to configure the relevant parameters for the analog input stage of each DAQ section. According to the description of the analog stage given in Par. **Analog Frontend**, it is possible to configure the following parameters:

- analog coarse gain (hereinafter Gain)
- offset
- input impedance (50 / 1K)
- Division by 5
- Shaper

Gain, impedance, and division can be configured for each couple of channels, the offset independently for each single channel (odd or even channel of the couple), while the shaper is common for all channels of DAQ1-2 and DAQ3-4 sections separately. Always press Apply button to store the settings and check if a successful info message appears.

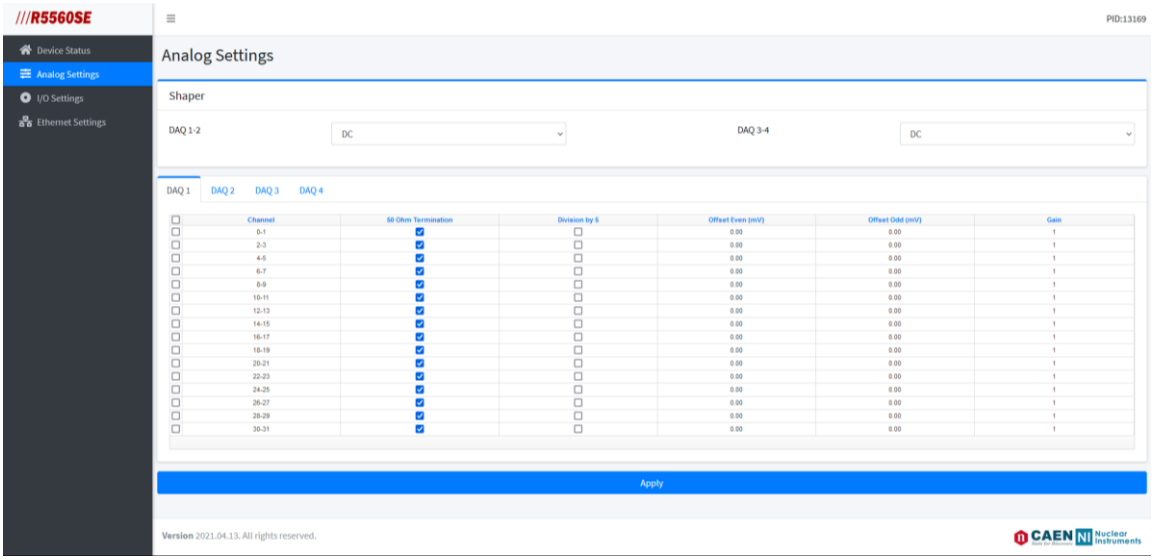



**Figure 13.4:** the Analog Settings page of the baseboard Web Interface.

## I/O settings

The I/O settings page allows to configure the clock and synchronization signals routing. It is possible to set:

- Clock source (**CLOCK INPUT**) choosing among INTERNAL, LEMO IN 0 or SYNC CONNECTOR. If LEMO IN 0 is selected, the clock must be a 25 MHZ TTL signal. If SYNC CONNECTOR is selected it must be a 25 MHz LVDS signal connected to pin A1/A2 of the SYNC IN RJ45 connector. Refer to Par. **Clock Distribution** of this manual to better understand the clocking strategy of the board. . For standard synchronization using SYNC IN/OUT connectors daisy chain (see Par. **Synchronization among multiple boards**), please choose **SYNC CONNECTOR** option for each of the slave boards.

- Internal pulser to be used in combination with CLOCK INPUT = Internal : frequency and width of the pulser can be chosen in the dedicated fields.

- The source for the synchronization signals used by the board (**DAQ SYNC IN**). Synchronization needs three independent signals to be routed to the DAQ board (CLOCK, RUN, T0). It is possible to select between LEMO IN, SYNC IN and Internal Pulser source for each of the three available lines. For standard synchronization using SYNC IN/OUT connectors daisy chain (see Par. **Synchronization among multiple boards**), please choose **SYNC IN** option for each of the slave boards. If using Internal Pulser source for one of the three signals, the remaining two ones can be driven via register using the option SYNC REG

- The signals carried on the RJ45 SYNC OUT connector placed on the front panel of the instrument. This connector can carry three independent signals that allows the synchronisation of two or more boards. It is possible to select between LEMO IN, SYNC IN, and INTERNAL PULSER for each of the three available lines. For standard synchronization using SYNC IN/OUT connectors daisy chain (see Par. **Synchronization among multiple boards**), please choose SYNC-IN option.

- The signals to be routed on the three front panel **LEMO OUT**; it is possible to select between LEMO IN, SYNC IN, INTERNAL PULSER and SYNC REG

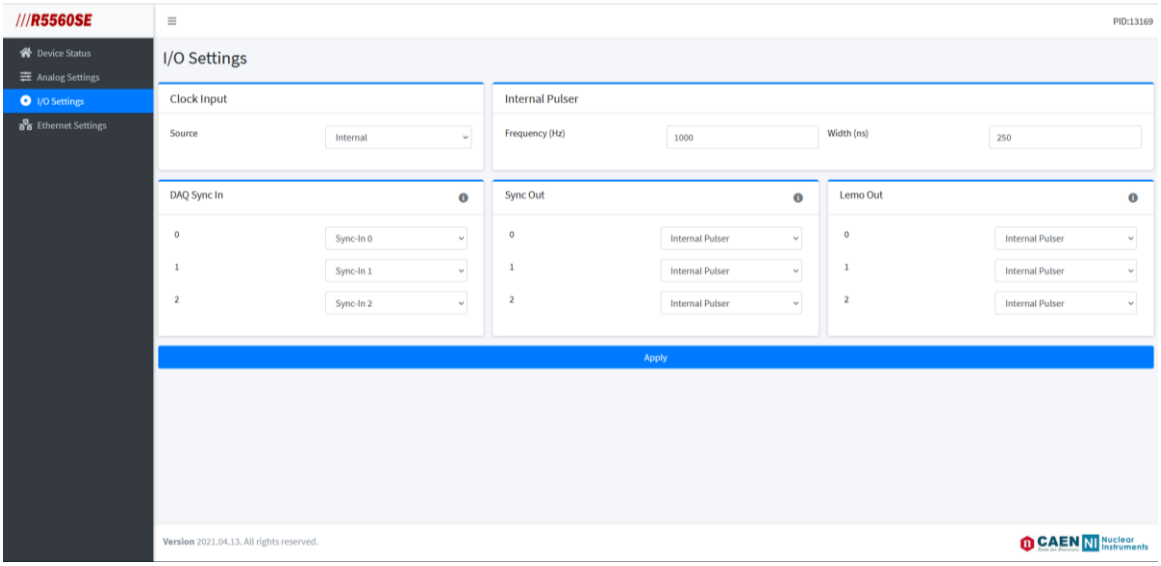



**Figure 13.5:** the I/O Settings page of the baseboard Web interface.

## Ethernet settings

The Ethernet Settings window allows the user to configure the IP address of the baseboard and of each DAQ section. It is possible to enable the DHCP in order to automatically obtain the IP from the DHCP server in the network or it is possible to assign manually the IP by clicking on the keyboard icon on the right.

Assign an IP in a valid range for your network to be able to reach the instrument from your computer. For example, if your computer IP address is a 10.128.1.1, type an IP that ranges from 10.128.1.2 to 10.128.1.254 and the mask (NM) must be 255.255.255.0

After modifying the Ethernet settings, press Apply button and follow the instructions given in the opening window.

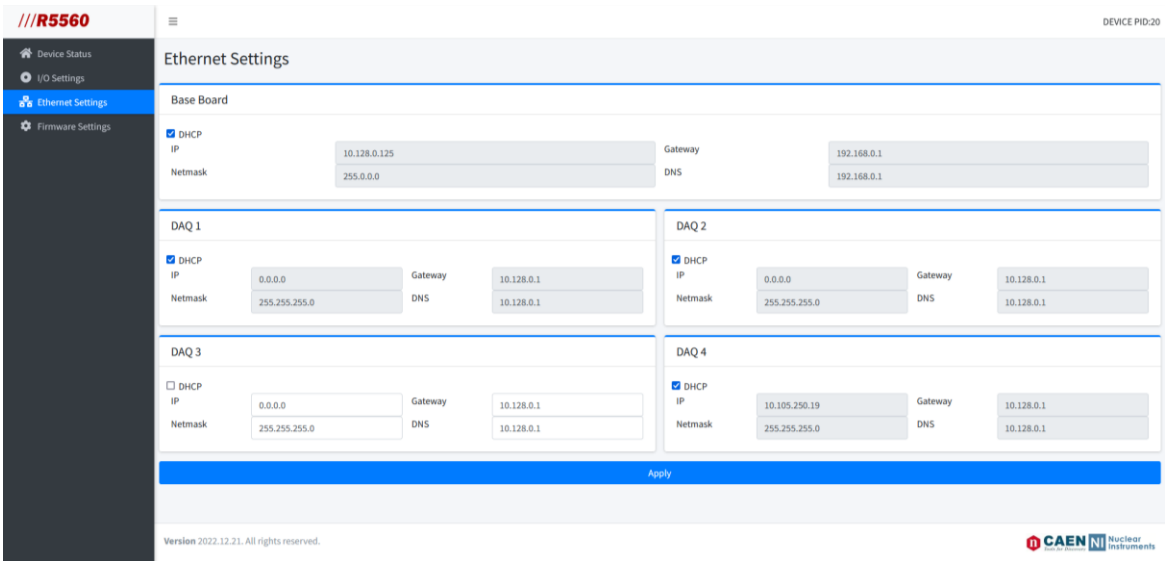



**Figure 13.6:** the Ethernet Settings page of the baseboard Web interface.

## Firmware Settings

The Firmware Settings window allows the user to manage the **FPGA firmware** to be loaded into the DAQs sections. It is possible to load the same firmware for all sections or different files for each section. Once the DAQs of interest are selected, it is possible to load firmware already installed onboard (choose from the drop-down menu and press *Install*) or to upload a new .**niu file**, for instance a firmware generated by Sci-Compiler.

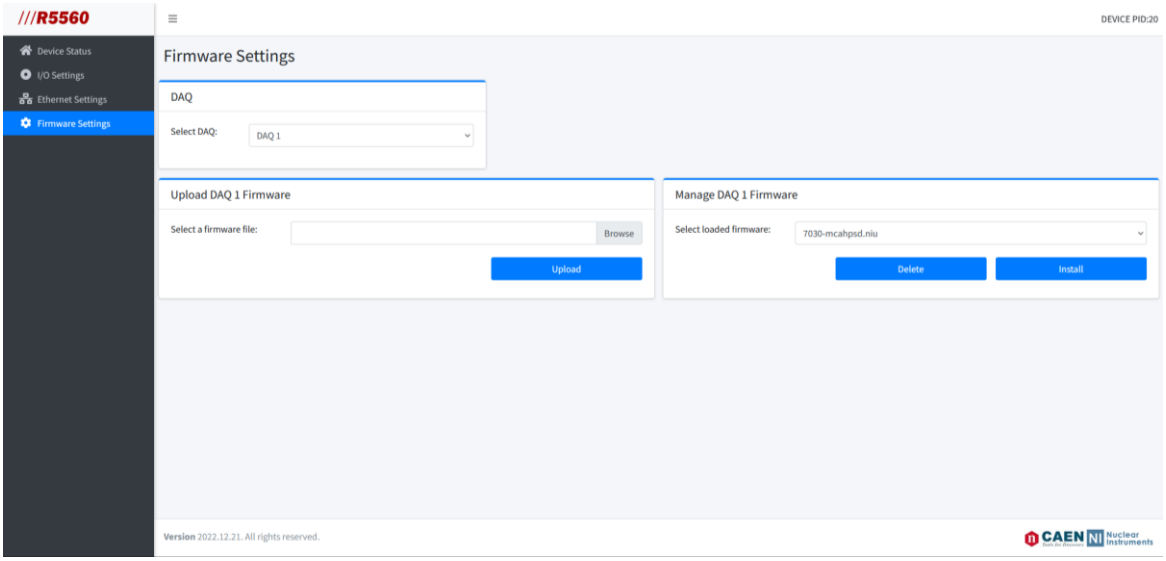



**Figure 13.7:** the Ethernet Settings page of the baseboard Web interface.

# DAQ section Web Interface

## Connection

To access the specific Web Interface of a DAQ section, connect to the board by using the Ethernet ports as shown below.

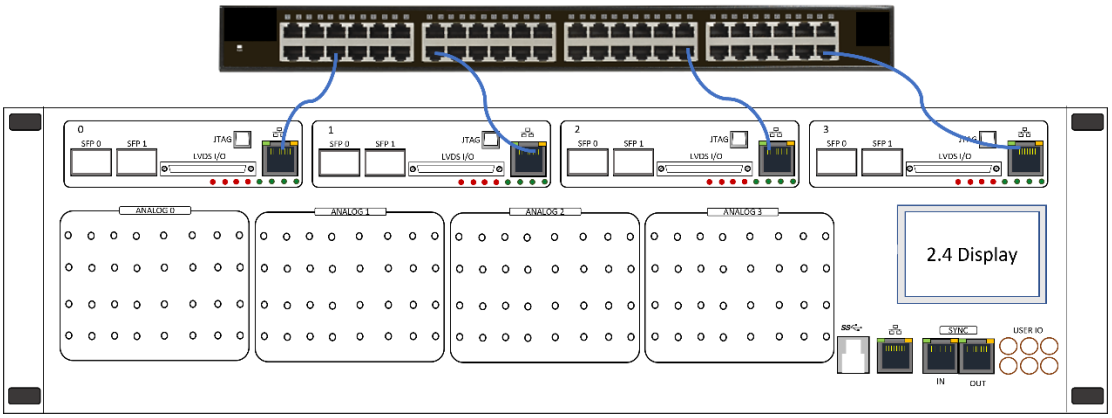



**Figure 13.8:** connection scheme to access the DAQ section Web interface. Each port can be connected independently to the network or directly to a PC Ethernet port**.**

## Device Status and firmware upgrade

In the Device Status (Home) window it is possible to monitor:

- the DAQ board IP address
- the FPGA UID
- the temperature of the DAQ board
- the Framework release (i.e., the OS of the system)
- the firmware loaded on the DAQ board (name and release), i.e., the pulse processing algorithm
- Resource Explorer status
- Connection information for base and all DAQs

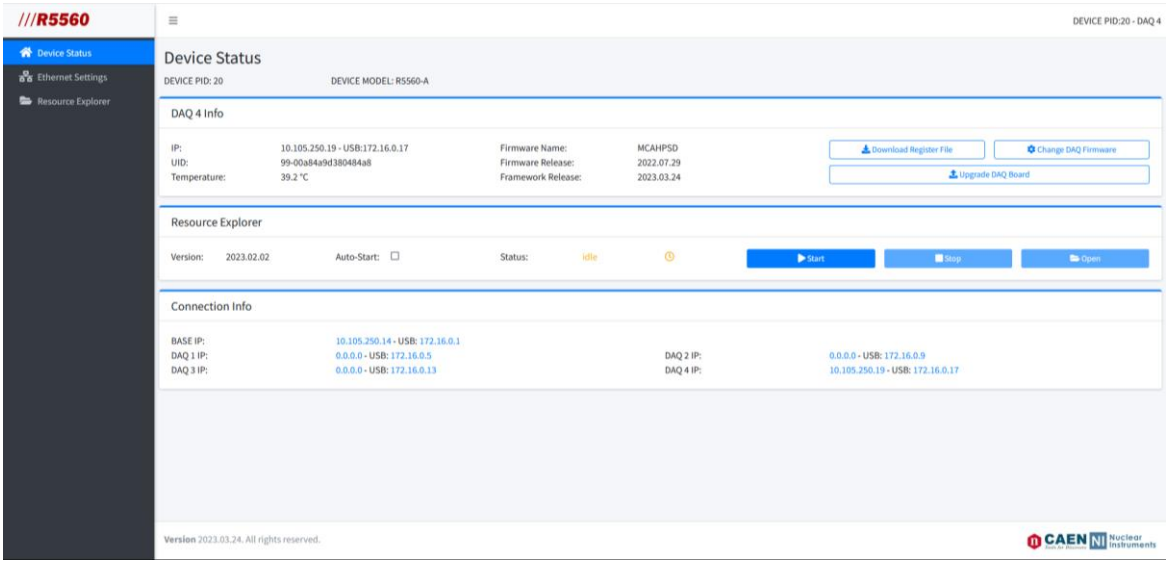



**Figure 13.9:** the Home page of the DAQ section Web interface.

If a newer version of the OS running on the ARM process is released on the CAEN website, it is possible to upgrade the device by clicking on *Upgrade DAQ Board*. The opening page will ask the user to select the **.dqsu file** previously downloaded from the CAEN website. Press Upgrade button to perform the loading of the new OS version and follow the given instructions.

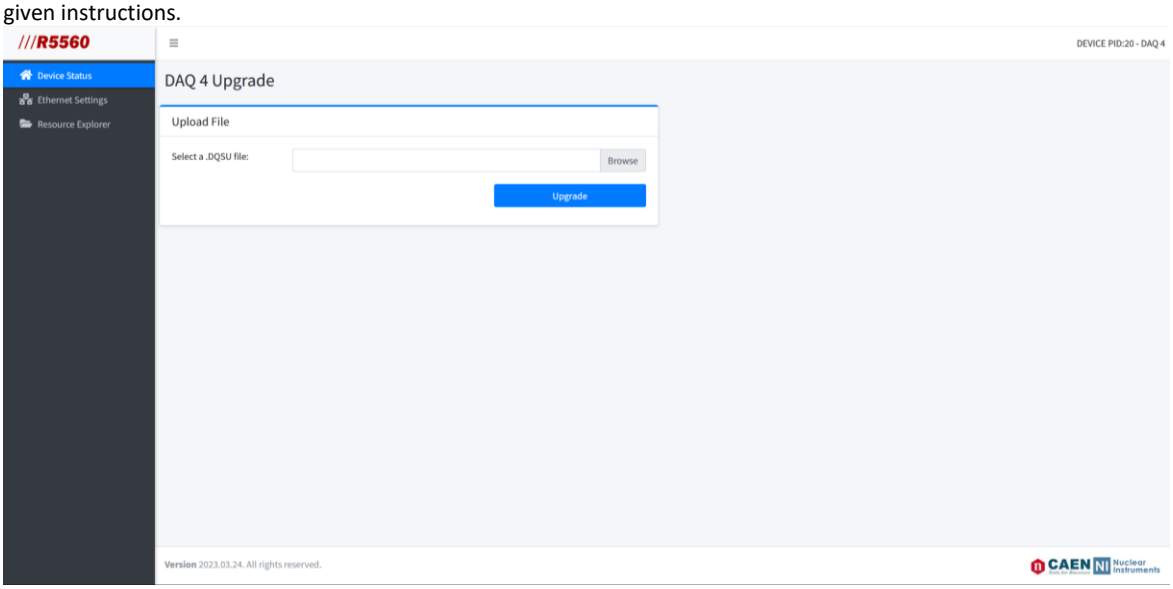



**Figure 13.10:** upgrade of the OS of the board from the DAQ Web interface.

It is also possible to change and upload a new FPGA firmware, for instance one that has been generated by Sci-Compiler. To upload a new firmware, press *Change DAQ Firmware* button in the DAQ section. In the opening page, it is possible to load a firmware already installed onboard (choose from the drop-down menu and press *Install*) or to upload a new **.niu file**, for instance a firmware generated by Sci-Compiler. The firmware will be copied in the persistent memory of the R5560/R5560SE.

**Note:** if upgrading the firmware via USB, follow the procedure below to avoid Windows USB peripheral recognition problems:

- Change the firmware and let the unit reboot automatically as per the standard process.
- Disconnect the USB cable.
- Reboot the module manually.
- Reconnect the USB and the unit should be correctly seen in the Device Manager as Remote NDIS Compatible Device

It is possible to activate the firmware by selecting its name in the Manage Firmware menu and press *Install* button.
It is also possible to remove an existing firmware by selecting its name in the scroll-down menu and press Delete button.
For any of this operation, follow the instructions given in the subsequent opening windows and always check if a successful info message appear.

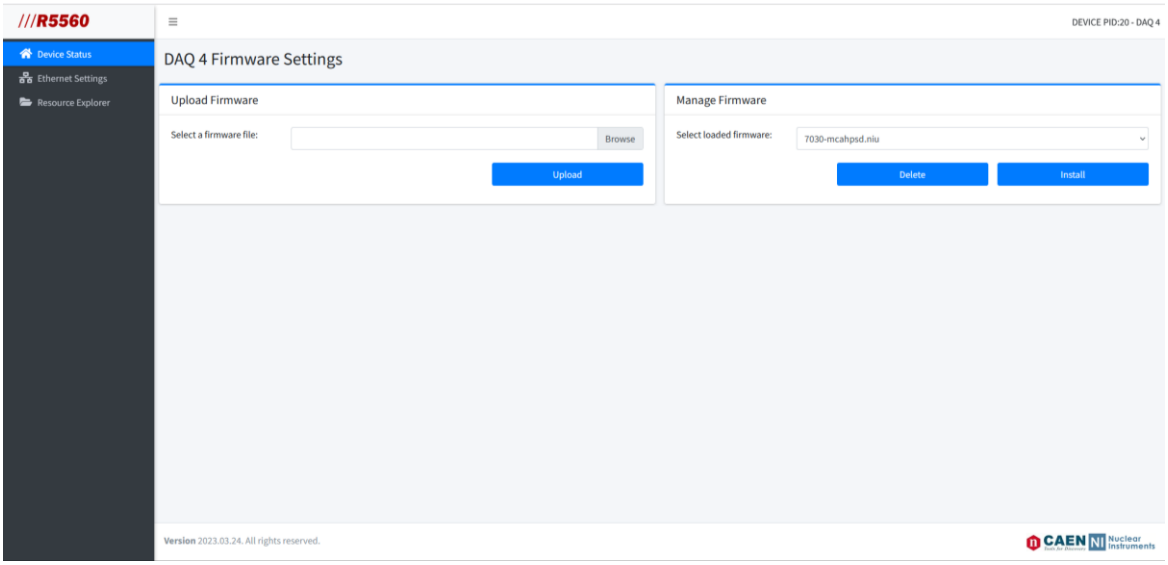



**Figure 13.11:** upgrade of the firmware from the DAQ section Web Interface

## Analog settings

The Analog Settings window allows to configure the relevant parameters for the analog input stage of each DAQ section. According to the description of the analog stage given in Par. **Analog Frontend**, it is possible to configure the following parameters:

- analog coarse gain (hereinafter Gain)
- offset
- input impedance (50 / 1K)
- Division by 5
- Shaper

Gain, impedance and division can be configured for each couple of channels, the offset independently for each single channel (odd or even channel of the couple), while the shaper is common for all channels of the DAQ section. Always press Apply button to store the settings and check if a successful info message appears.

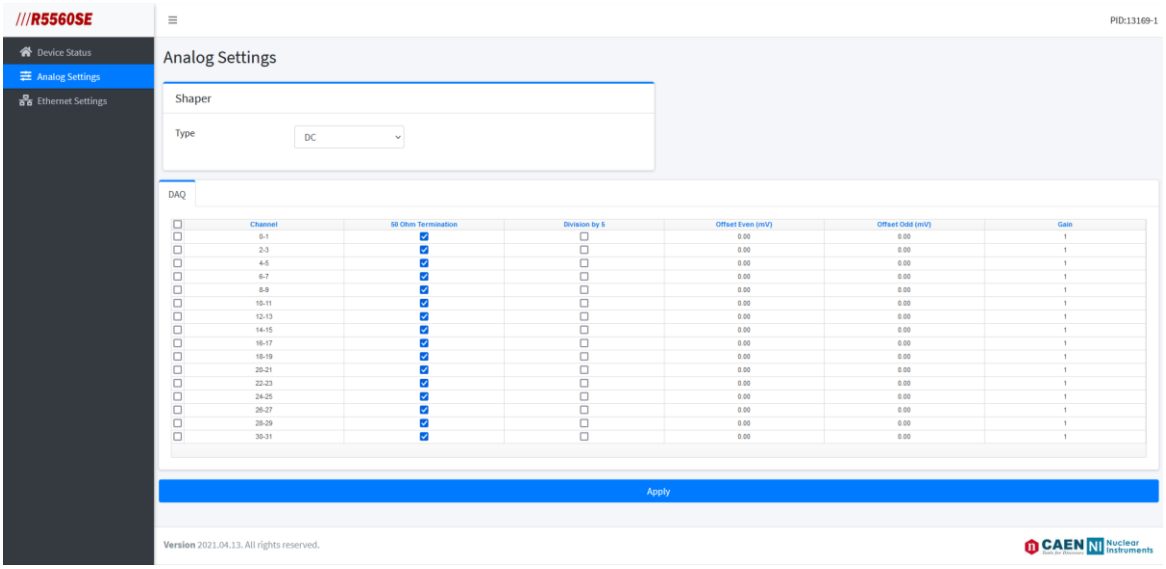



**Figure 13.12:** the Analog Settings page of the DAQ section Web Interface.

## Ethernet settings

The Ethernet Settings window allows the user to configure the IP address of the DAQ section. It is possible to enable the DHCP in order to automatically obtain the IP from the DHCP server in the network or it is possible to assign manually the IP by clicking on the keyboard icon on the right.

Assign an IP in a valid range for your network in order to be able to reach the instrument from your computer. For example, if your computer IP address is a 10.128.1.1, type an IP that ranges from 10.128.1.2 to 10.128.1.254 and the mask (NM) must be 255.255.255.0

After modifying the Ethernet settings, press Apply button and follow the instructions given in the opening window.

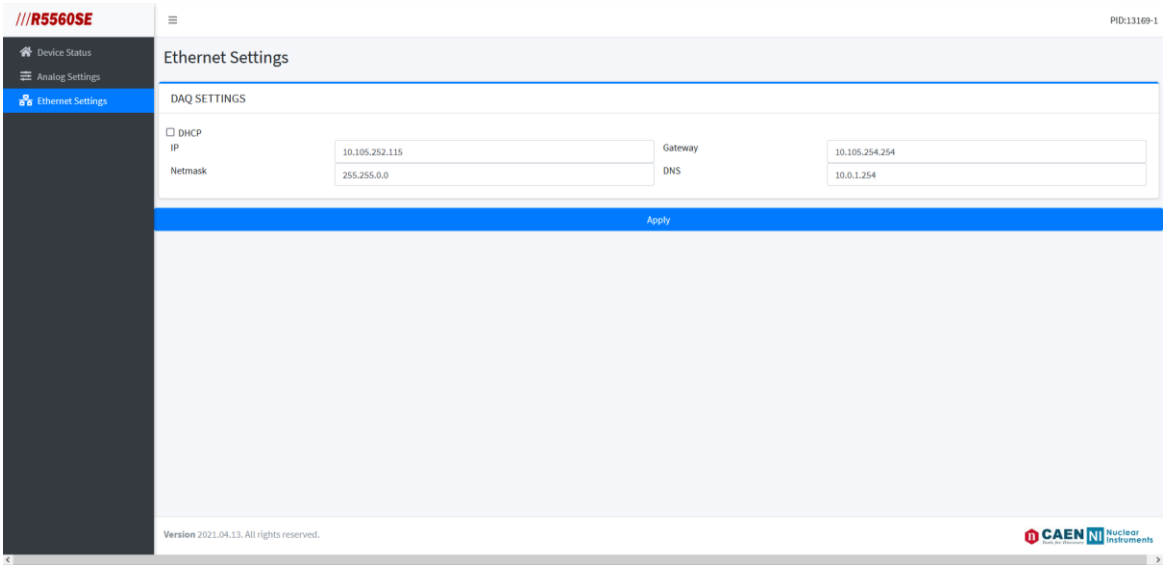



**Figure 13.13:** the Ethernet Settings page of the DAQ section Web interface.

## Resource Explorer (beta)

The Resource Explorer tab replicates the same functionalities of the Resource Explorer tool available in Sci-Compiler.

Sci-Compiler offers a simple built-in tool, called *Resource Explorer,* to connect one of the supported boards and test the features of the FPGA firmware with no need to write any software code. This tool allows to manage all **Local Bus** readout blocks placed in the firmware diagram therefore it gives the possibility to read and write the configuration registers (i.e. Signal Processing parameters) and shows the readout instruments like *Spectrum* , *Oscilloscope*, etc. in a GUI , so that it is possible to change the signal processing parameters and test the results of acquisition live, with no need to write any software code. In practice, the Resource Explorer is a first-use debug interface to test the firmware and check the behaviour of the Processing algorithm specified through the block diagram.

Refer to **[RD1]** for more details.

# 14 Firmware Developing

The R5560/R5560SE is a programmable platform and it is designed in order to encourage the user to develop its own custom firmware using **SCI-Compiler** to generate and compile the firmware code.

A full working **default firmware** is provided: this is a fully featured solution and it is developed not as a basic example to start developing with SCI-Compiler but as a full DAQ readout system firmware. The R5560SE comes with the default firmware already uploaded. In any case the firmware can be downloaded from CAEN website and easily installed on the board with the OpenHardware – Firmware Upgrader tool.

The **default firmware** implements the typical features of Waveform Recording Digitizers and QDC algorithms. It is fully managed by the free-downloadable and open-source Open Hardware Readout Software. In more details, the default firmware implements the following pulse processing features:

- **Waveform digitization** of all analog channels
- Leading edge or derivative trigger
- 128 independent channels **trapezoidal filter** for energy calculation
- Time stamping with 8 ns resolution
- Different readout modes: independent channels, frame trigger (OR of all channels trigger), external trigger
- External veto and trigger on programmable LEMO GPIO
- List readout mode
- Rate meter with dead time calculation on each channel

**Note:** The default firmware supports readout via Ethernet or USB 2.0. No support to optical link is included in the default firmware.

## Firmware design: the processing core

The R5560SE processing core is a Xilinx Kintex7 FPGA. The FPGA can be programmed in several ways:

- **VHDL/Verilog**: this is the typical language for developing processing system in FPGAs. It is based on basic operation (logic/arithmetic/conditional processing/sequential element) connected each other to build component. Component are then connected together in order to create more complex designs.

- **C**: a C/C++ program is automatically converted in a VHDL design by a software tool.

- **Design with high level blocks**: the user connects together a series of pre-designed building block in order to obtain the desired processing system.

**SCI-Compiler is a set of very high-level blocks which implement the functionalities of the most common instrumentation used in physics experiment: digitizer, MCA, TDC, Time Over Threshold, trigger, scaler, etc.**

Sci-Compiler is designed to work at best with R5560SE and CAEN suggests using this firmware generator and compiler tool, since it focuses the attention only on the functional blocks of the application to be implemented and does not require a deep knowledge of the device in use.
The user is free to develop a firmware in VHDL without using the SCI-Compiler. We suggest using the default framework to correctly initialize the board components. Framework source code is partially encrypted in order to protect our IP and avoid unauthorized copy of the board. The framework code is available inside the installation folder of SCI-Compiler.

SCI-Compiler generates VHDL code and Vivado Project. If the user wants to add custom functionalities that are not implementable with SCI-Compiler, it is possible to develop some parts of the design in VHDL and then compile as a standard Vivado Project.

The R5560SE can be used for different applications, requiring different firmware (for example multichannel analog digitization, readout of mixed signal ASICs, direct readout of shaped signals).

In the following, we describe the default firmware of the R5560SE.

# Default firmware

The R5560/R5560SE has a default firmware pre-programmed from factory. The default firmware implements 32-channels for each DAQ submodule digitizer with **waveform recording** and **Pulse Height Analysis.**

The default firmware is designed to work with Open Hardware Readout Software. It is fully developed in SCI-Compiler (see **Figure 14.1**) and can be easily modified by the user. The default firmware project is available in the SCI-Compiler example section.

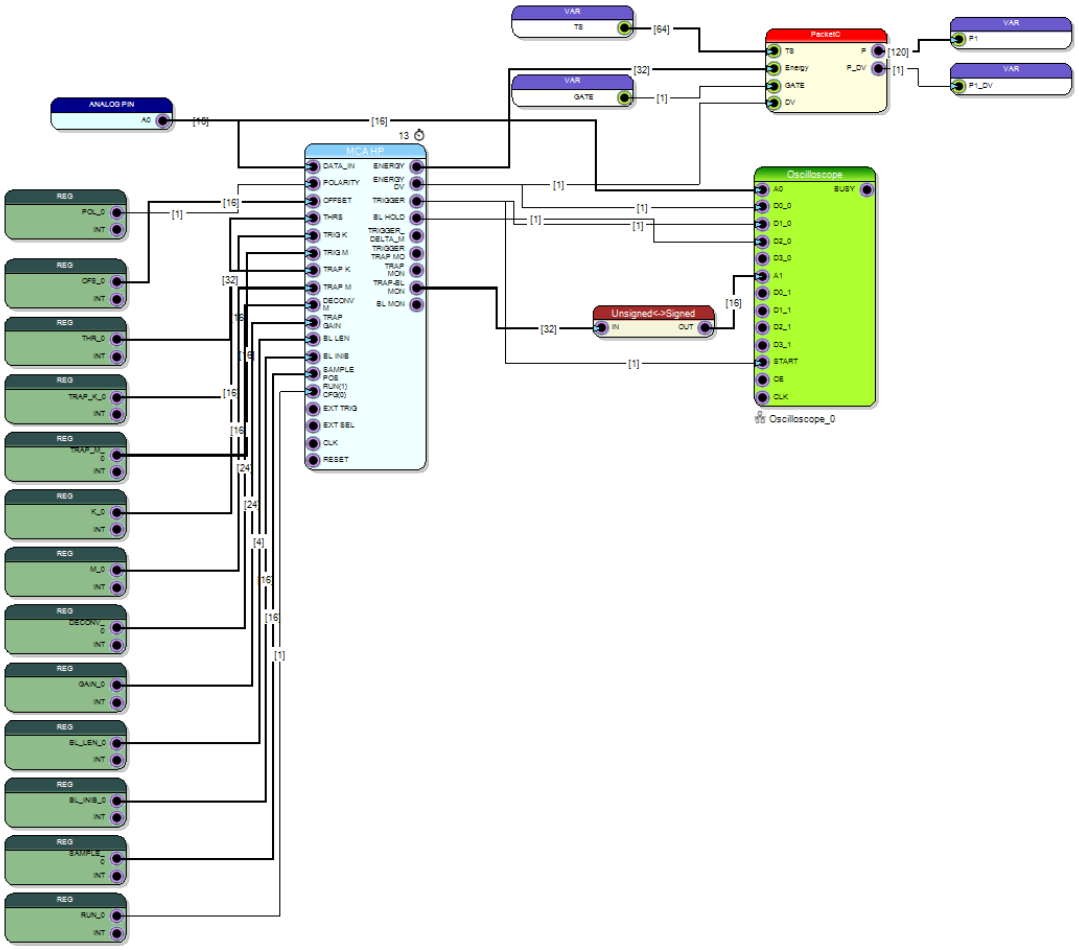



**Figure 14.1:** SCI-Compiler scheme for single channel data processing in the default firmware.

The default firmware implements an independent trapezoidal energy filter on each channel for energy calculation. It is based on a derivative trigger followed by a trapezoidal energy filter and gives in output a packet containing the timestamp and the energy of the input signal. The trapezoidal filter is a filter able to transform the typical exponential decay signal generated by a charge sensitive preamplifier into a trapezoid whose flat top height is proportional to the amplitude of the input pulse (that is to the energy released by the particle in the detector) (see **Figure 14.2**). The trapezoid plays almost the same role of the shaping amplifier in a traditional analog acquisition system. There is an analogy between the two systems, both have a “shaping time” constant. For both, a long shaping time gives a better resolution but has higher probability of pile-up. Both are AC coupled with respect to the output of the preamplifier whose baseline is hence removed, but both have their own output DC offset and this constitutes another baseline for the peak detection.

The output data packet contains an header, a word for the energy, two words for the timestamp and a word for the channel ID.

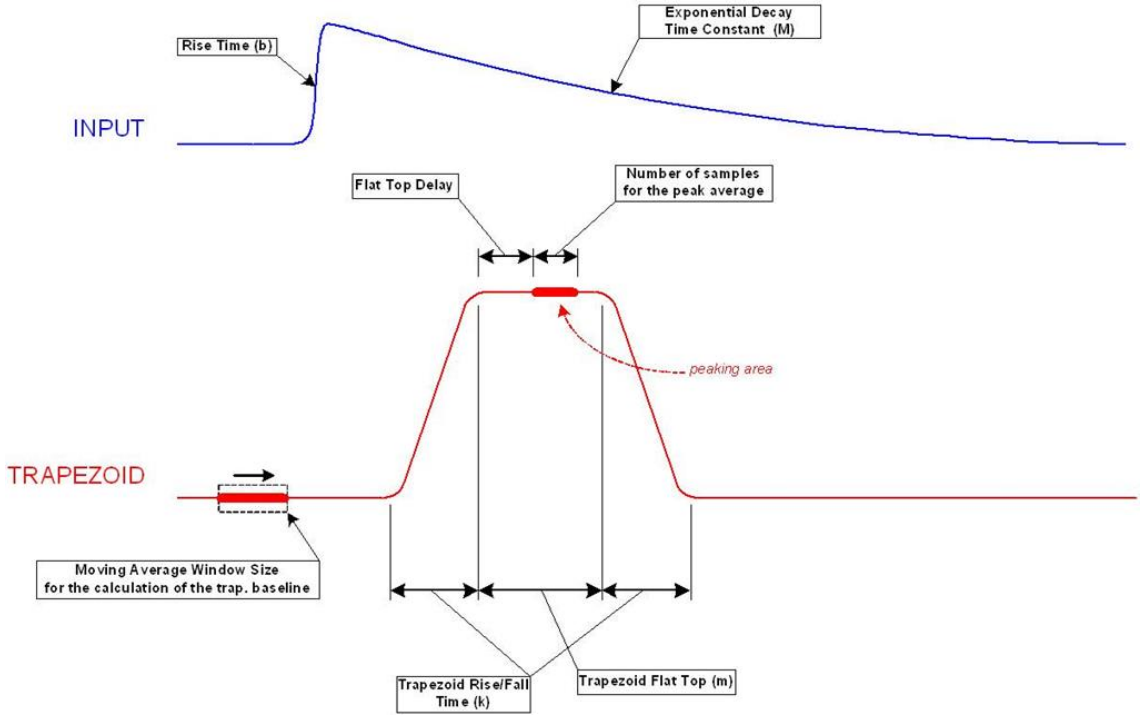



**Figure 14.2:** the trapezoid method used in the board default firmware.

# 15 Drivers & Libraries

The R5560/R5560SE does not require any driver to operate with Ethernet link and it is compatible with any OS. To interface with USB, a RNDIS driver is required in Windows OS, while it will work as is in modern Linux OS.

A library to manage the default firmware and open-source software example code is available at **https://github.com/NuclearInstruments/r5560_sdk**

Moreover, libraries for C/C++/C#/Python are available for custom firmware generated by Sci-Compiler too under the form of a Software Development Kit called Sci-SDK.

The SciSDK is a Software Development Kit installed together with Sci-Compiler setup and compatible with any board supported by Sci-Compiler.

**Note:** for specific documentation of the functions and parameters of the SciSDK, refer to the Sci-Compiler **online Help** or to **https://nuclearinstruments.github.io/SCISDK/**

**Note:** alternative installation methods for the SciSDK are given in the online documentation

## Drivers installation

To install the USB drivers for R5560SE correct operation in Windows OS, follow the step below:

- Connect the USB 3.0 connector of the R5560 to the PC and Windows will locate 6 new devices and a root hub:
  - 5x RNDIS compatible ethernet card
  - 1x Serial Adapter
- The serial adapter does not require any driver.
- Open the Device Manager of your computer. You will find 5 peripherals with yellow exclamation mark label or a RNDIS device or an unknown device.
- Right click on each one → update driver → Browse my computer → Let me pick from a list → Network Adapters. Select Microsoft as manufacturer and Remote NDIS Compatible Device as model. Authorize the installation and the driver should be installed automatically.

## R5560 SDK for default firmware

A library to manage the default firmware is available at **https://github.com/NuclearInstruments/r5560_sdk.**
It contains the functions that allows to start and close the connection with the R5560/R5560SE device, to write and read data to/from registers, to write and read array of data to/from buffers and to read arrays of data from FIFO buffers.
All the functions require a connection handle to specify the board to communicate with. The handle is defined in the following way:

```
typedef struct {
     int Csocket;
     int connected;
     uint32_t __IICBASEADDRESS;
     uint32_t __IICBASEADDRESS_STATUS;
     SOCKET_TYPE socketType;
} tR5560SE_Handle;
```

All the functions return an integer indicating the communication status:

- 0: ok
- -1: no connection
- -2: socket error
- -3: connection drop
- -4: allocation error
- -5: out of limit error
- -6: timeout error

**Start Connection**

Starts the ethernet connection to the R5560SE device. The arguments are the IP address passed as a string, the connection port and the connection handle to the board which will be returned by the function.

```
int R5560SE_ConnectTCP(char *ipaddress, uint32_t port, tR5560SE_Handle *handle);
```

| Argument | Direction | Type |
|---|---|---|
| ipaddress | Input | char |
| port | Input | uint32_t |
| handle | Input/output | tR5560SE_Handle |

example:

```
tR5560SE_Handle handle;
R5560SE_ConnectTCP("192.168.50.150", 8888, &handle);
```

**Close Connection**

Disconnects the R5560SE device specified by the connection handle passed as argument.

```
int NI_CloseConnection(tR5560SE_Handle *handle);
```

| Argument | Direction | Type |
|---|---|---|
| handle | Input/output | tR5560SE_Handle |

example:

```
tR5560SE_Handle handle;
NI_CloseConnection(&handle);
```

**Write Register**

Allows to write a value in a register of the FPGA in the R5560SE device. The arguments are the data to be written in the register, the register address to be written and the board connection handle.

```
int NI_WriteReg(uint32_t data, uint32_t address, tR5560SE_Handle *handle);
```

| Argument | Direction | Type |
|---|---|---|
| data | Input | uint32_t |
| address | Input | uint32_t |
| handle | Input/output | tR5560SE_Handle |

example:

```
tR5560SE_Handle handle;
NI_WriteReg(1, 0xA, &handle);
```

**Read Register**

Allows to read a value stored in a register of the FPGA in the R5560SE device. The arguments are a pointer returning the data read from the register, the register address to be read and the board connection handle.

```
int NI_ReadReg(uint32_t *data, uint32_t address, tR5560SE_Handle *handle);
```

| Argument | Direction | Type |
|---|---|---|
| data | Input/output | uint32_t |
| address | Input | uint32_t |
| handle | Input/output | tR5560SE_Handle |

example:

```
tR5560SE_Handle handle;
```

```
uint32_t data;
NI_ReadReg(&data, 0xA, &handle);
```

**Write Data**

Allows to write an array of values in a register of the FPGA in the R5560SE device. The arguments are the data array to be written in the register, the number of data to be written, the register address to be written, the board connection handle and the number of data written in the register returned by the function.

```
int NI_WriteData(uint32_t *data,
        uint32_t count,
        uint32_t address,
        tR5560SE_Handle *handle,
        uint32_t *written_data);
```

| Argument | Direction | Type |
|---|---|---|
| data | Input/output | uint32_t |
| count | Input | uint32_t |
| address | Input | uint32_t |
| handle | Input/output | tR5560SE_Handle |
| read_data | Input/output | uint32_t |

example:

```
tR5560SE_Handle handle;
uint32_t data[10]= {0, 2, 10, 1, 0, 1, 3, 1, 2, 0};
uint32_t written_data;
NI_WriteData(&data, 10, 0xA, &handle, &written_data);
```

**Read Data**

Allows to read an array of values from a buffer of the FPGA in the R5560SE device. The arguments are the preallocated array of data to be read from the buffer, the number of data to be read, the buffer address to be read, the board connection handle and the number of data read from the buffer returned by the function.

```
int NI_ReadData(uint32_t *data,
       uint32_t count,
       uint32_t address,
       tR5560SE_Handle *handle,
       uint32_t *read_data);
```

| Argument | Direction | Type |
|---|---|---|
| data | Input/output | uint32_t |
| count | Input | uint32_t |
| address | Input | uint32_t |
| handle | Input/output | tR5560SE_Handle |
| written_data | Input/output | uint32_t |

example:

```
tR5560SE_Handle handle;
uint32_t data[100];
uint32_t read_data;
NI_ReadData(&data, 100, 0xA, &handle, &read_data);
```

**Read Fifo**

Allows to read data from a FIFO buffer of the FPGA in the R5560SE device. The arguments are the pre-allocated array of data to be read from the buffer, the number of data to be read, the buffer address to be read, the address of the register containing the FIFO status, the type of bus mode (0 for register access like, 1 for blocking streaming access like and 2 for non-blocking streaming access like), the timeout of the read

operation in milliseconds, the board connection handle and the number of data read from the buffer returned by the function.

```
int NI_ReadFifo(uint32_t *data,
        uint32_t count,
        uint32_t address,
        uint32_t fifo_status_address,
        BUS_MODE bus_mode,
        uint32_t timeout_ms,
        tR5560SE_Handle *handle,
        uint32_t *read_data);
```

| Argument | Direction | Type |
|---|---|---|
| data | Input/output | uint32_t |
| count | Input | uint32_t |
| address | Input | uint32_t |
| fifo_status_address | Input | uint32_t |
| bus_mode | Input | BUS_MODE |
| timeout_ms | Input | uint32_t |
| handle | Input/output | tR5560SE_Handle |
| read_data | Input/output | uint32_t |

example:

```
tR5560SE_Handle handle;
uint32_t data[100];
uint32_t read_data;
NI_ReadFifo(&data, 100, 0xA, 0xB, 1, 1000, &handle, &read_data);
```

# 16 Open Hardware Readout Software

Open Hardware Readout Software is a **free and open-source** software developed for Windows OS to operate **in conjunction with the default firmware** of the R5560/R5560SE, in order to provide a ready-to-use solution.

Moreover, CAEN provides free and open source libraries as a Software Development Kit (see Chap. **Drivers & Libraries** for more details).

The software implements typical features needed to acquire and process data in nuclear spectroscopy and particle physics:

- Waveform monitor for all the channels in free running mode or with leading edge trigger
- List mode readout (energy, time) and data saving with channel independent trigger
- Energy Spectrum plot for all channels
- Bidimensional heatmap visualization for imaging, with configurable detector shape
- Spectrum fitting and energy calibration
- Data saving with waveform dump on file

The software is distributed both as compiled application and as source code. The source code is written in VB.NET and C# and it can be easily customized by the user to adapt to a custom firmware and for any other need. In order to recompile the Open Hardware Readout Software, a free version of Visual Studio .NET 2015 or later must be installed on the user's PC.

The following guide and screenshots refer to Open Hardware Readout Software version 2021.10.1.0 with a single DAQ section of the R5560SE being connected.

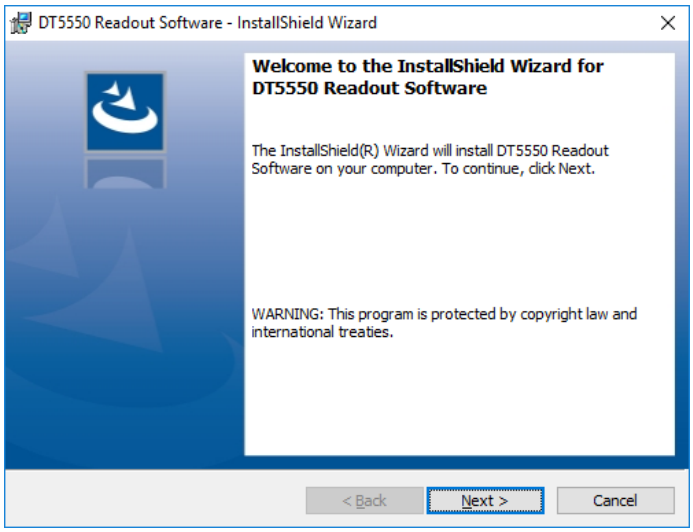
## Software installation

Open Hardware Readout Software is compliant with Windows 10-64bit.

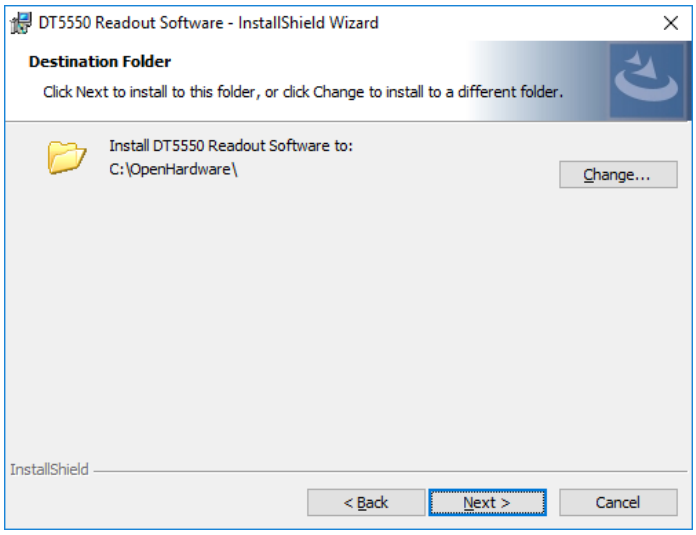
**Note:** The software is standalone and does not require the prior installation of any library. The installation setup also installs the Open Hardware Firmware Upgrader tool.

In order to install the Open Hardware Readout Software and the Open Hardware Firmware Upgrader, follow the steps below:

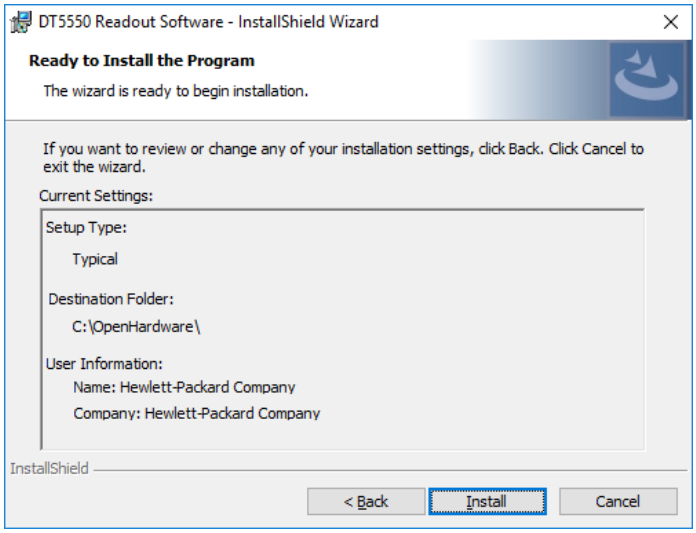
- Download the software package from the R5560SE product page on the CAEN website (**login required**)
- Unzip and run the executable.
- A setup wizard will start. **Press "Next" to continue.**



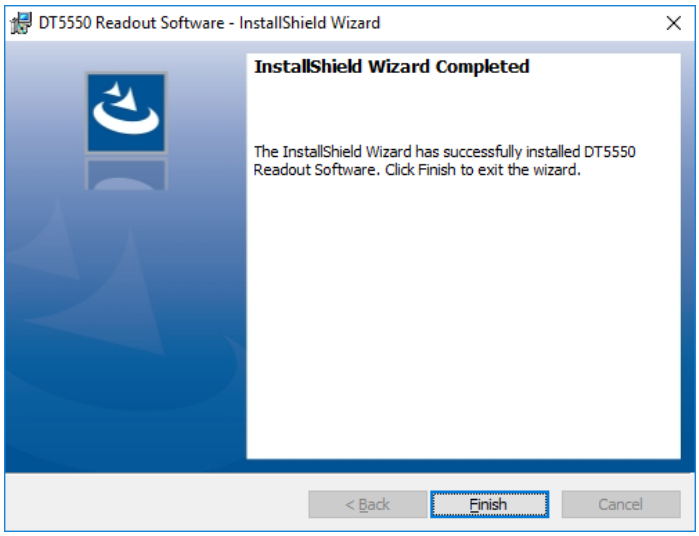

- Choose the destination folder and **press "Next".** By default, the readout software is installed in C:\OpenHardware\R5560SE and the firmware upgrader in C:\OpenHardware\FirmwareUpgrader



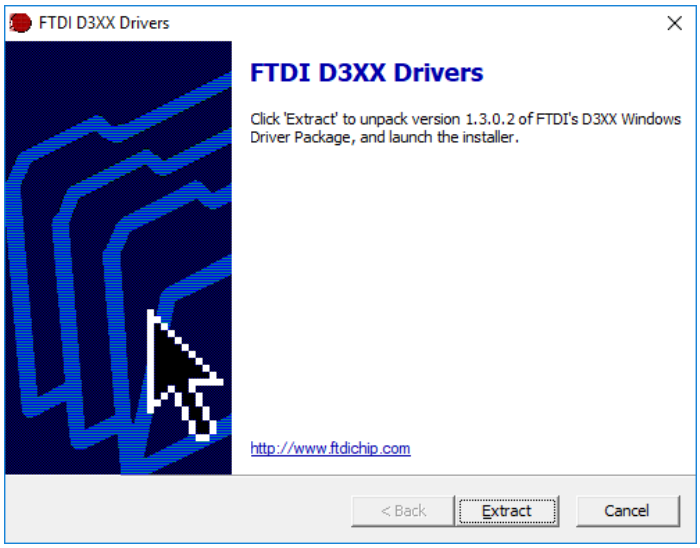

- **Click "Install"** to complete software installation.



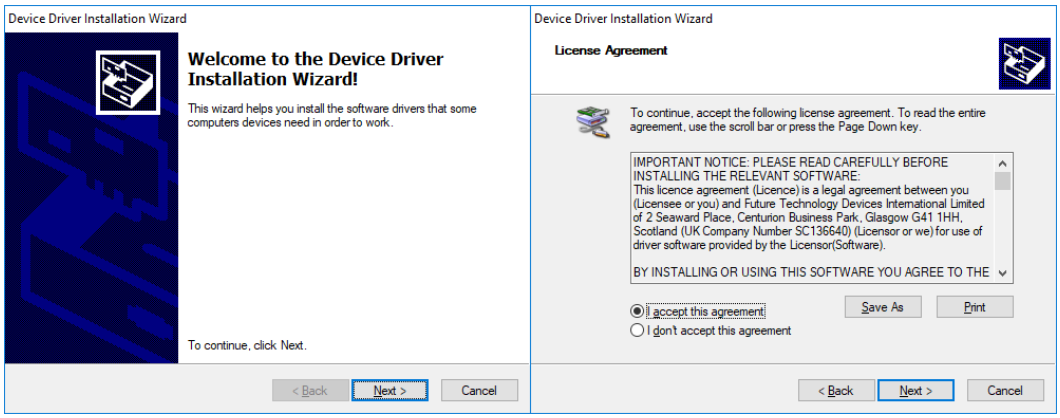

- Wait until installation is completed and press "**Finish**" to complete the setup.




- Once Open Hardware Readout Software installation is complete, the Wizard will ask to extract and install FTDI USB Drivers (needed to operate with other compatible boards). Press "**Extract**" to continue.




- **Press "Next"** in the following window to continue and **accept the License Agreement**.




- The driver will then be installed. **Press "Finish"** to exit the Wizard.
- Now it is possible to launch the software.

**Note:** the software is open source: it is possible to find the software VB.NET source code and the default firmware project (developed with SCI-Compiler) in the installation folder, to allow user customization.

# Board connection

After launching the software, the "Connection" window will open. By selecting the R5560SE tab the user can connect each DAQ of the R5560SE board by clicking on the 'Add' button. In the new row that appears in the table, it is possible to choose the connection type and to insert the IP address (Ethernet or USB) of the selected board type.

- *Connection*: it is possible to choose between 'Ethernet' or 'USB'. Both connections works upon IP address protocol.
- *IP Address*: insert the IP address corresponding to the board to be connected and to the relevant connection interface. Ethernet and USB address are shown in the *DAQ LAN Status* of the Touchscreen display of the board.

After the connection parameters has been set, the user can press the "Connect" button to initialize the software GUI. In the 'Status' field of each board, the connection status is displayed. If the connection cannot be established for one or more of the inserted boards, an error message box will appear.

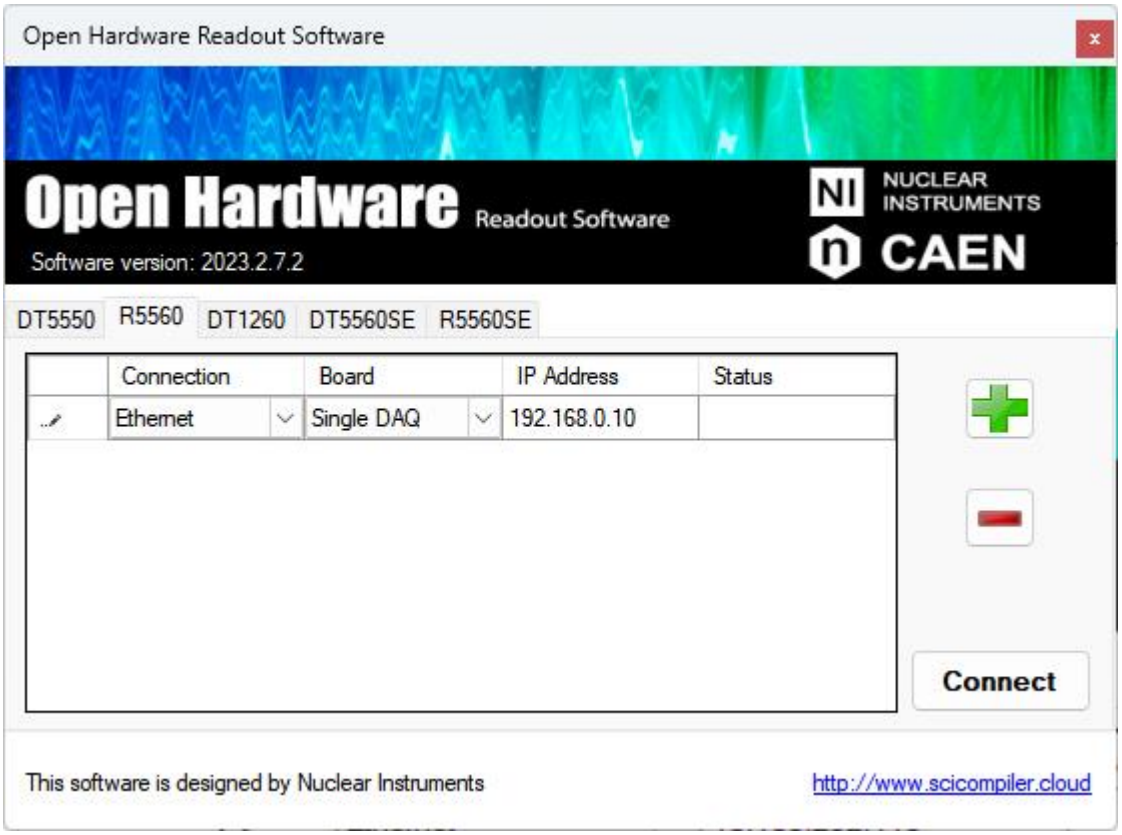



**Figure 16.1:** the "Connection" window at start-up of the Open Hardware Readout Software

The 'Remove'  button deletes the selected row of the table.

# Software GUI Description

After successful connection, the main window will appear. The window is divided in four areas:

- Control bar: here there are all the buttons needed to control the acquisition process.
- Main working area: all tabs like "Settings", "Spectrum" and "Oscilloscope" are shown here.
- Imaging module: the 2D image of the channels energies is reconstructed in the two heatmap (one for the single frame view and the other for the cumulative view).
- Log Area: all messages related to the user actions are displayed in the history log.

All panels can be undocked and rearranged also outside the main window in order to organize the software layout on multiple monitors.

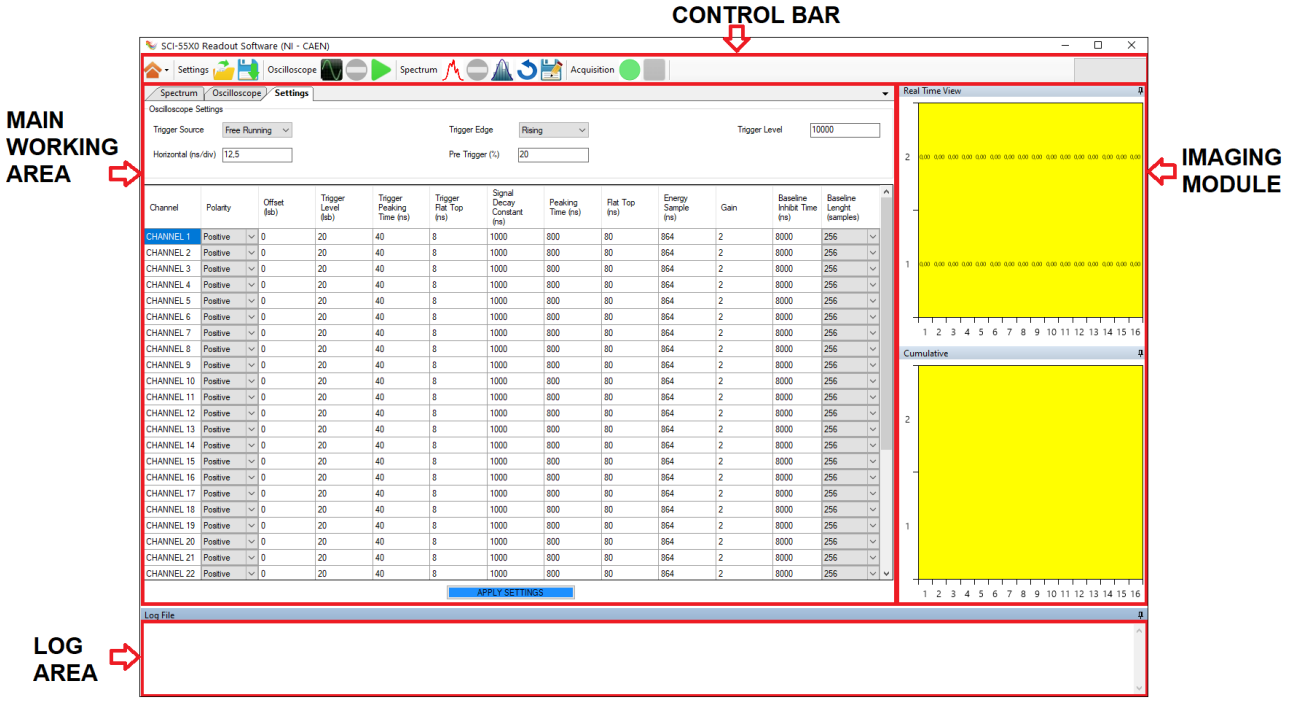



**Figure 16.2:** the main window of the Open Hardware Readout Software. The main areas are highlighted.

## Control Bar

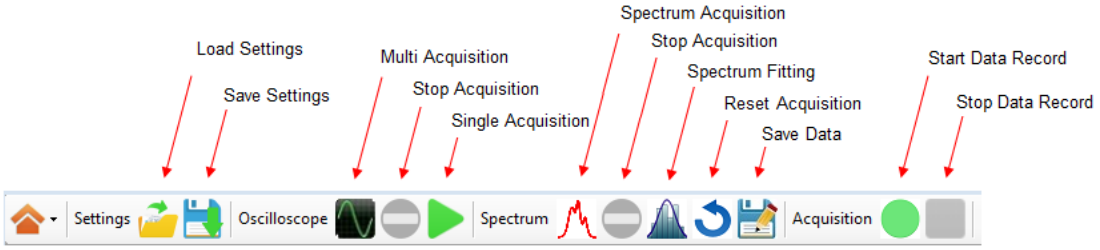
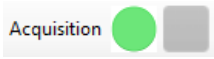



**Figure 16.3:** the Control Bar of the Open Hardware Readout Software.

The "Control Bar" contains the following buttons:

- 
  - **Spectrum**: change the visualization features of the spectrum (semi-logarithmic or linear on Y axis, rebinning on the X axis, graphical plot type)
  - **View**: select the zoom type
  - **File**: export and print plots
-  :
  - **Load** settings from file
  - **Save** Settings to file
- 
  - **Multi Acquisition**: start to acquire waves until stop acquisition button is pressed
  - **Stop Acquisition**: abort waves acquisition
  - **Single Acquisition**: acquire a single waveform every time the button is pressed
- 
  - **Spectrum Acquisition**: start spectrum and image acquisition process
  - **Stop Acquisition**: abort spectrum acquisition
  - **Spectrum Fitting**: open the fitting tool window
  - **Reset Acquisition**: reset spectrum and image
  - **Save Data**: save on disk the spectrum for each channel and the cumulative images

- 

  - ❖ **Start Data Record**: open the data dump window to configure and start saving data (waves or energy) event by event
  - ❖ **Stop Data Record**: stop current data dump process

## Settings Tab

The Settings Tab allows to configure all digitizer parameters. It is divided in two areas, as shown in **Figure 16.4**:

- **Oscilloscope settings (red box)**: it allows to set the parameters to visualize the waveforms on the oscilloscope plot: it can be selected the same trigger source, trigger edge and trigger level for all the oscilloscopes, the time scale on the plot and the length of the pre-trigger window
- **Channel Specific Settings (green box)**: this is divided in two tabs:
  - *Firmware settings* is a table with a row for each channel containing all configuration parameters for that channel, i.e. signal polarity and offset, all the derivative trigger parameters, all the trapezoidal filter parameters and all the baseline calculation parameters.
  - *AFE Settings* is a table with a row for each couple of channels, containing the configuration parameters of the analog frontend stage, i.e. the input impedance, the offset, shaping and analog coarse gain.

The user must push the button **APPLY SETTINGS**, at the bottom of the window, in order to let the settings become effective. It is suggested to stop an ongoing acquisition begore applying new settings.

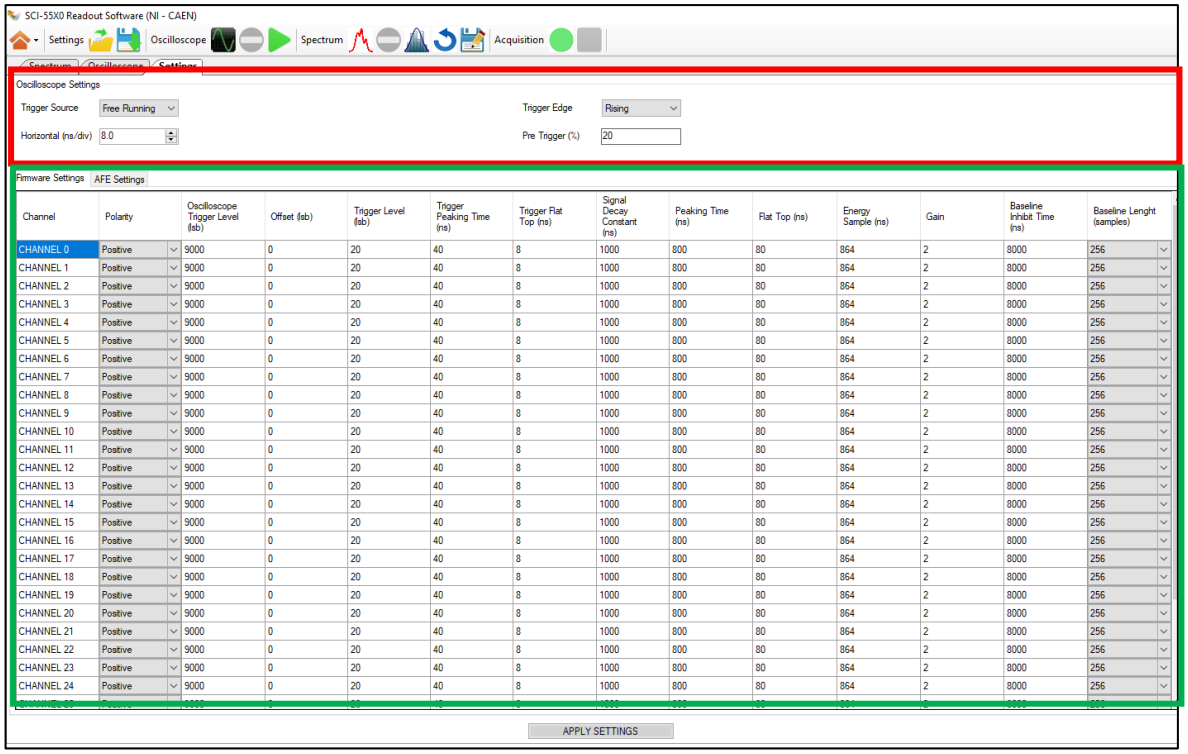



**Figure 16.4:** the Settings Tab of the Open Hardware Readout Software.

### Oscilloscope Settings

These settings (see red box in **Figure 16.4**) allow to configure the same parameters of all the oscilloscopes (i.e. of the waveform recording).

- Trigger Source: Select the trigger source for the oscilloscopes. Options are:
  - **Analog Signal**: a leading-edge trigger on the analog input of each channel is implemented to trigger the correspondent oscilloscope waveform acquisition. The Trigger Edge (*Oscilloscope settings* tab) and the **Oscilloscope Trigger Level** (*Firmware settings* tab) must be properly set.

- **MCA Trigger**: the derivative trigger of each channel is used as a trigger for the correspondent oscilloscope waveform acquisition. The threshold to be set is **Trigger level (lsb)** in the *Firmware Settings* tab.
- **Free running**: each oscilloscope operates without trigger and acquires a waveform as soon as the previous transfer is completed

- Trigger Edge: it selects if each oscilloscope triggers on the rising or falling edge of the input signal. It works only in conjunction with *Trigger Source = Analog Signal*
- Horizontal (ns/div): Set the time base for each oscilloscope in ns per division
- Pre-Trigger (%): Specify the portion of the acquisition window dedicated to the pre-trigger (signal sampled before the trigger)

### Firmware Settings

These settings (see green box in **Figure 16.4**) allow to configure the specific parameters of each channel, used by the MCA firmware component:

- Polarity: Select between positive and negative polarity according to the input channel
- Oscilloscope Trigger level (lsb): set the absolute trigger threshold taken into consideration by the oscilloscope acquisition if *Trigger Source = Analog Signal* in the *Oscilloscope Settings* tab.
- Offset: Set the analog offset added to the analog input channel after the polarity inversion
- Trigger Level (lsb): Set the derivative trigger level threshold. It is taken into consideration always for Spectrum acquisition, while for Oscilloscope acquisition only if *Trigger Source = MCA Trigger*
- Trigger Peaking Time (ns): Set the peaking time of the trapezoidal trigger signal (between 16 ns and 800 ns)
- Trigger Flat Top (ns): Set the flat top of the trapezoidal trigger signal
- Signal Decay Constant (ns): Set the trigger inhibition time after a trigger
- Peaking Time (ns): Set the peaking time of the energy trapezoidal filter (between 16 ns and 4 us)
- Flat Top (ns): Set the flat top of the energy trapezoidal filter
- Energy Sample (ns): Set the position where the value of the energy trapezoidal filter is sampled to measure the signal energy (should be peaking time + 0.8*Flat Top).
- Gain: Multiplication factor for the energy trapezoidal filter result. This allow to recover resolution avoiding trashing away the least significant bits of the energy filter. A low gain reduces the resolution of the measure while a too high gain will saturate the spectrum dynamic.
- Baseline Inhibit Time (ns): Inhibition time for the baseline calculation after a trigger signal. It avoids that the tail of an event is measured as part of the baseline (should be greater than 2*Decay Constant).
- Baseline Length (samples): Length in samples of the window for baseline calculation.

**Note:** the sum of the peaking time and of the flat top could not exceed 4096 ns. In this case the fields related to these parameters become red in the table and the 'Apply Settings' button is disabled.

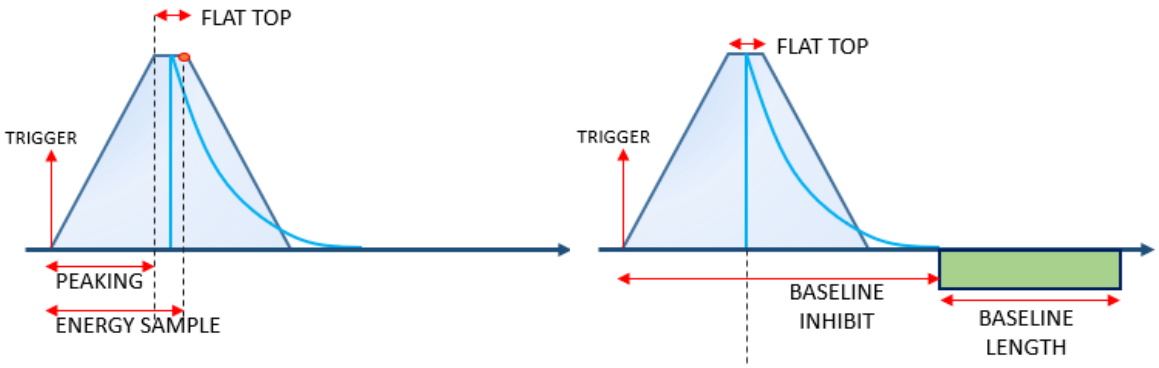



**Figure 16.5:** scheme of the trapezoidal energy filter parameters to be set in the Open Hardware Readout Software.

### AFE Settings – R5560SE only

These settings (see **Figure 16.6**) allow to configure the analog frontend parameters for each channel of the board. According to the the description of the analog stage given in Par. **Analog Frontend,** it is possible to configure the following parameters:

- Shaper: Select the shaper type and its timing among DC, AC 1us, AC 10us, AC 30us. This parameter is common for all channels.
- Termination: it is possible to choose between 50Ω and 1k Ω input termination
- Division by 5: enable/disable the analog input signal voltage divider (÷ 5)
- Offset Even (mV): set the offset of the even channel of the correspondent couple indicated in the *Channel* column
- Offset Odd (mV): set the offset of the odd channel of the correspondent couple indicated in the *Channel* column
- Gain: set the analog coarse gain for the relevant couple of channels among the listed possible values.

…

**Note:** Gain, Impedance and Division by 5 can be configured for each couple of channels, the Offset independently for each single channel, while the shaper is common for all channels.

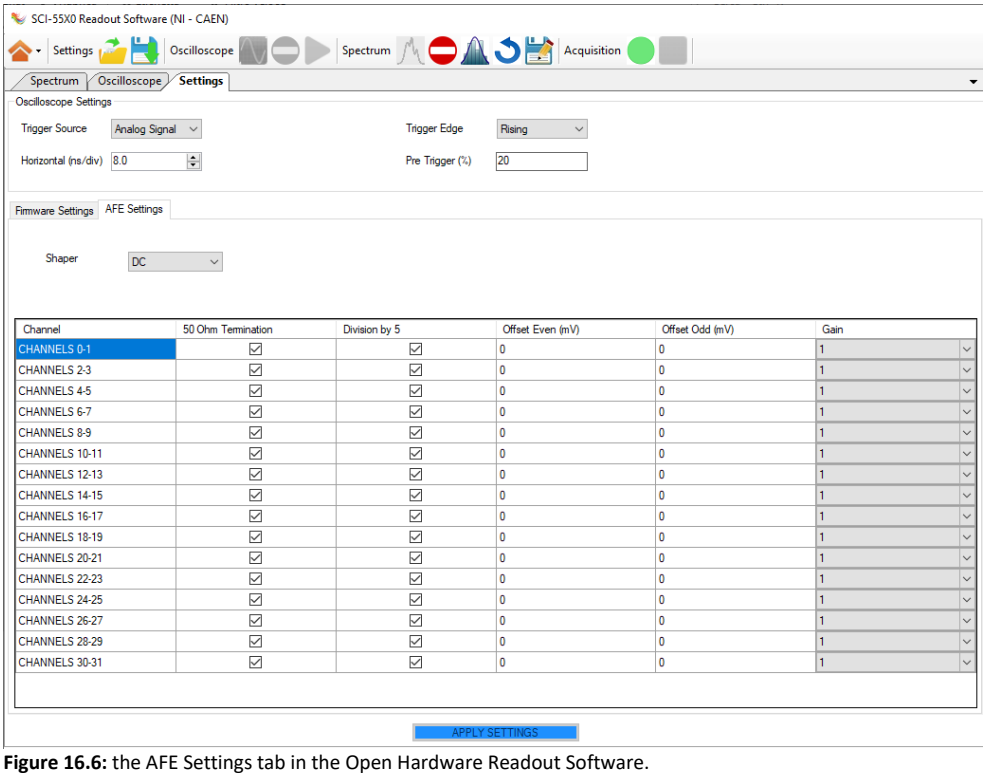



**Figure 16.6:** the AFE Settings tab in the Open Hardware Readout Software.

## Oscilloscope Tab

The Oscilloscope Tab allows to monitor on plot all analog input signals and digital probes:

- The ANALOG trace displays the waveform sampled by the ADC
- The TRAPEZOIDAL trace displays the analog signal after the trapezoidal energy filter and the baseline subtraction
- The ENERGY SAMPLE digital signal indicates the position at which the value of the trapezoidal filter is sampled for the energy calculation
- The TRIGGER digital signal toggles when the trapezoidal trigger identifies a signal
- The BASELINE HOLD digital signal displays the state of the baseline restorer. When low, the baseline restorer is running, when high is holding

Multiple input channels can be captured and displayed on the same plot by checking the correspondent checkbox from the channels list on the left of the Oscilloscope Tab. All oscilloscope configuration parameters can be set in the *Settings* Tab.

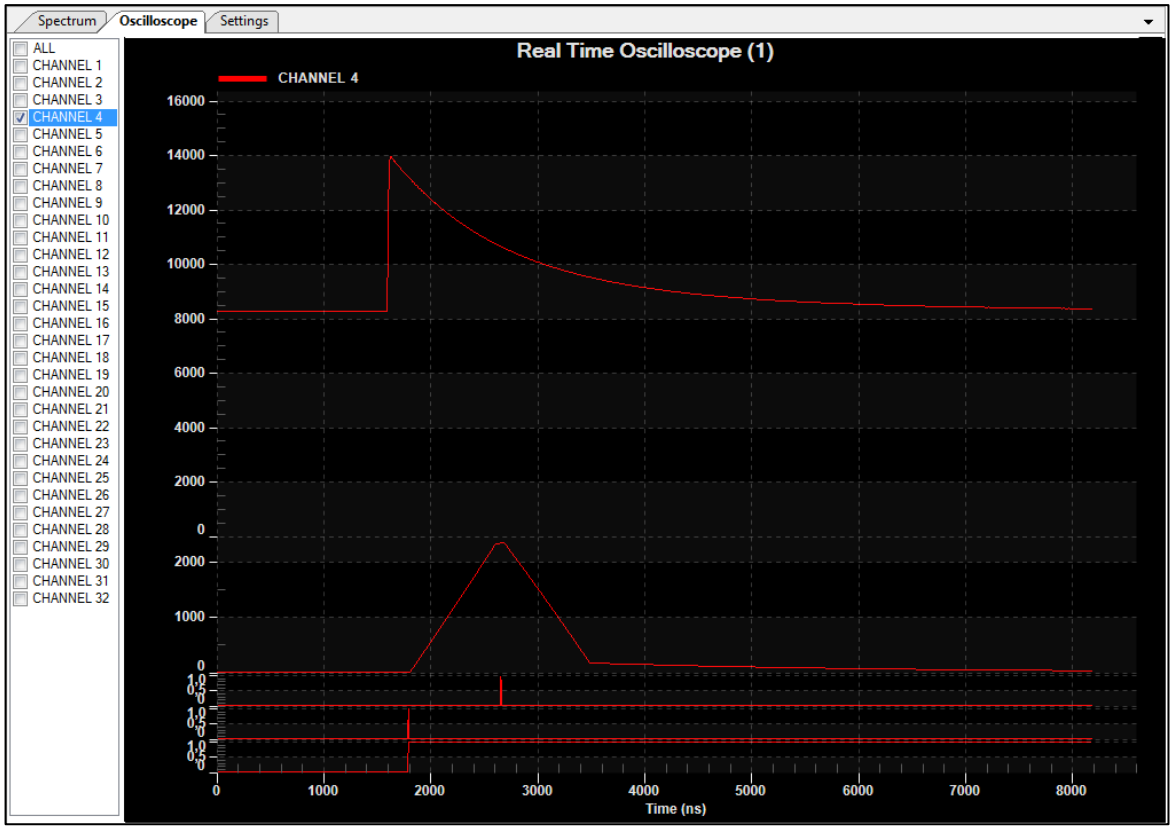



**Figure 16.7:** the Oscilloscope Tab of the Open Hardware Readout Software. Here only signals for channel 4 are displayed.

The value indicated between the parenthesis in the Oscilloscope plot title represents the current total number of waveform acquisitions.

In order to zoom the waveforms, three zoom modes are available. The zoom mode can be selected pressing a key on your keyboard. The same operation can be done from Menu → View. Options are:

- Area (press Z): enables the zoom on both axes. Drag the mouse to zoom a rectangular area.
- Horizontal Zoom (Press H): enables horizontal zoom. Drag the mouse to zoom on the X-axis.
- Vertical Zoom (Press V): enables vertical zoom. Drag the mouse to zoom on the Y-axis.
- Unzoom (press U): restores the full view of the plot.

The user can also:

- print the current view of the plot: Menu → File → Print → Oscilloscope (or press P).
- export the current view of the plot: Menu → File → Export → Oscilloscope (alternatively press X).

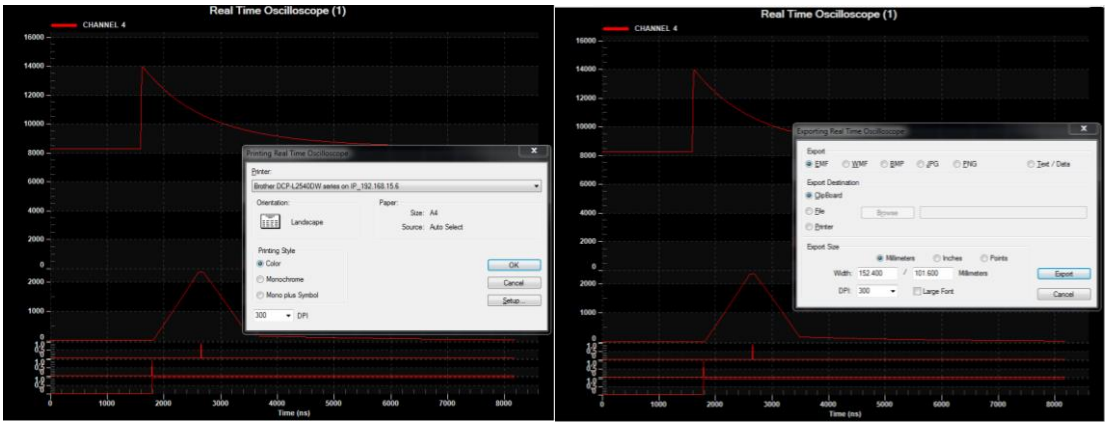



**Figure 16.8:** printing (left) and saving to file (right) the current view of the Oscilloscope Tab

## Spectrum Tab

The Spectrum Tab allows to display the cumulative spectrum of each channel calculated in real-time by the board. The trapezoidal energy filter of the default firmware of the R5560SE generates a 64k bin spectrum for each channel. The rebinning feature can be selected from the Menu → Spectrum → Rebin. The rebinning is applied to all channels.

The spectrum of multiple channels can be displayed on the same plot by checking the relative checkboxes in the channels list on the left of the Spectrum Tab. Each time a channel spectrum is added to the display area, a new label will appear in the legend and a color will be automatically selected for the new spectrum. The plot window is divided in two areas. The main area displays the current zoom of the spectrum. On the bottom area is shown the full view of the spectrum.

In order to zoom the spectrum, three zoom modes are available. The zoom mode can be selected pressing a key on your keyboard. The same operation can be done from Menu → View

- Area (press Z): enables the zoom on both axes. Drag the mouse to zoom a rectangular area.
- Horizontal Zoom (Press H): enables horizontal zoom. Drag the mouse to zoom on the X-axis.
- Vertical Zoom (Press V): enables vertical zoom. Drag the mouse to zoom on the Y-axis.
- Unzoom (press U): restores the full view of the plot.

The spectrum can be displayed in both linear and semi-logarithmic mode. It is possible to switch between Lin/Log mode: Menu → Spectrum → Lin/Log (alternatively Press L).

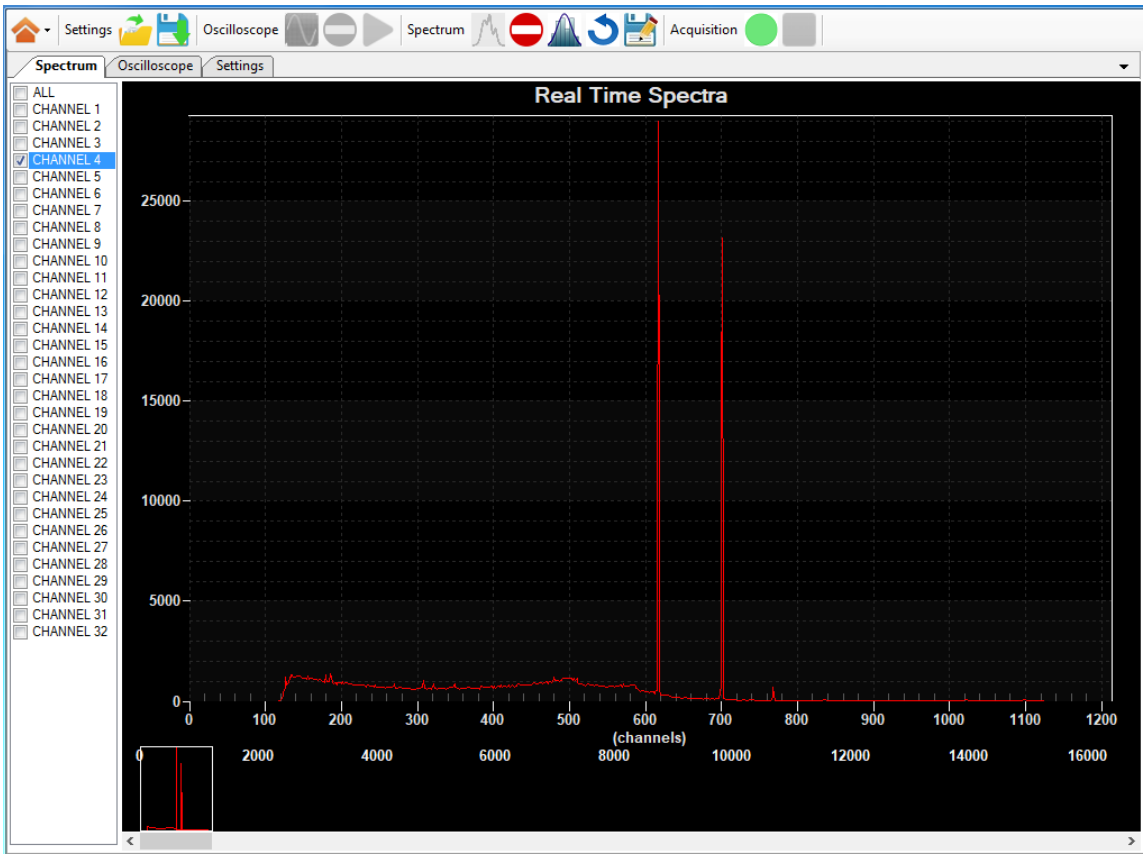



**Figure 16.9:** the Spectrum Tab of the Open Hardware Readout Software. Here only the spectrum for channel 4 is displayed.

It is also possible to change the plot type: Menu → Spectrum → Plot Mode (alternatively press O to cycle between plot modes). Available plot modes are:

- Step
- Line
- Line with interpolation
- Bar
- Area
- Area with interpolation
- Dot
- Dot with Line
- Dot with interpolation

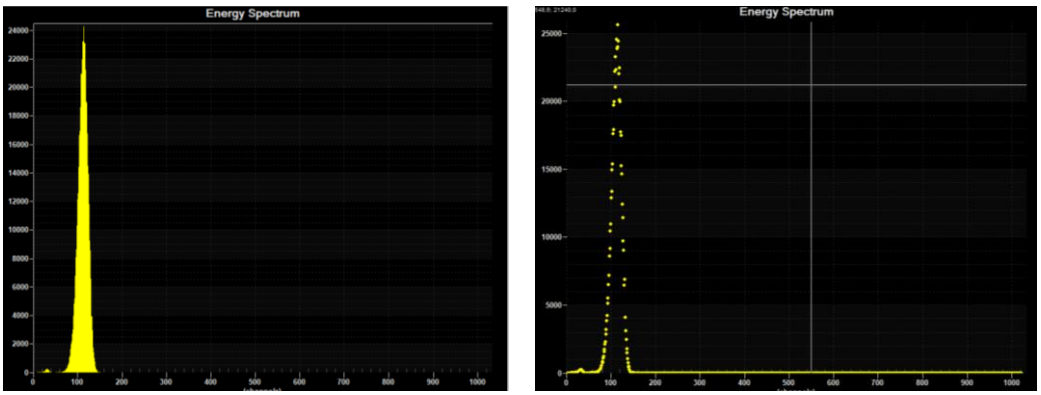



**Figure 16.10:** spectra shown in different plot modes (left = Area, right = Dot).

The user can:

- print the current view of the plot: Menu → File → Print → Spectrum (alternatively press P).
- save the current view of the plot: Menu → File → Export → Spectrum (alternatively press X).

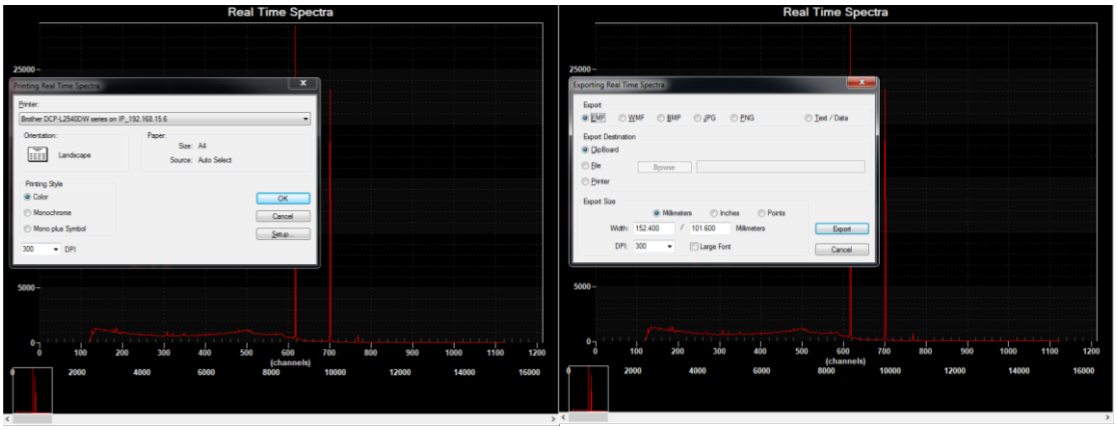
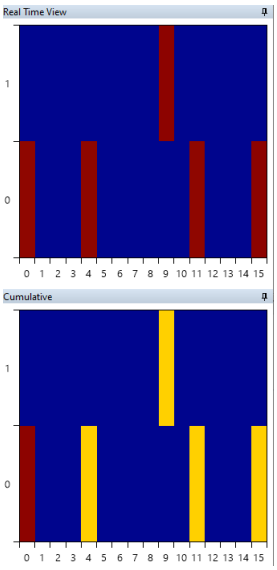



**Figure 16.11:** printing (left) and saving to file (right) the current view of the Spectrum Tab.

## Imaging Module




The Imaging Module displays the real-time and the cumulative 2D heatmaps acquired by the R5560SE, basing on the measured energies for each channel. Each pixel of the image is associated to a channel. The user can rearrange the pixels to create a different shape.

The last captured event is displayed on the **real-time view** (top part of the display area). As the Custom Packet component transfers only one triggered event at a time, only one channel energy value will be displayed.
The bottom part of the display area shows the **cumulative image**.

The acquisition is controlled by the “Start Spectrum” and “Stop Spectrum” buttons in the Control Bar. The “Reset Acquisition” button for spectra in the Control Bar reinitializes also the acquisition of the heatmaps.
The “Save Data” button in the Control Bar allows to save the cumulative energies of all channels in a .csv file whose saving path can be chosen by the user.

**Figure 16.12:** the Imaging Module of the Open Hardware Readout Software, where it is possible to see 5 channels acquiring energy data.

# How to Perform a Fit

The fitting tool of the Open Hardware Readout Software allows to perform gaussian fits of the peaks in the spectrum. In order to create a new fit, follow the instructions below:

- a channel should be checked in the list of channels on the left of the Spectrum Tab.
- press the “Spectrum Fitting” button in the Control Bar to open the fitting tool at the bottom of the Spectrum Tab.
- insert a value in the Cursor 1 and in the Cursor 2 columns of the table to specify respectively the left and the right boundaries of the fit. On the spectrum, the fit boundaries are displayed with white vertical lines and identified with a number corresponding to the table row number. The fit area is shown in green.
- The fitting tool uses the spectrum data between the two cursors to calculate the Mean, the standard deviation (STD) and the Area of the selected region. For each fitted area the results extracted from the gaussian fit are also reported: the mean (Mean Fit), the standard deviation (STD Fit), the full width at half maximum (FWHM), the Resolution (R) and the fitting area (Area Fit).
- A fit can be deleted by selecting the correspondent row in the table and pressing ‘Del’ on your keyboard.

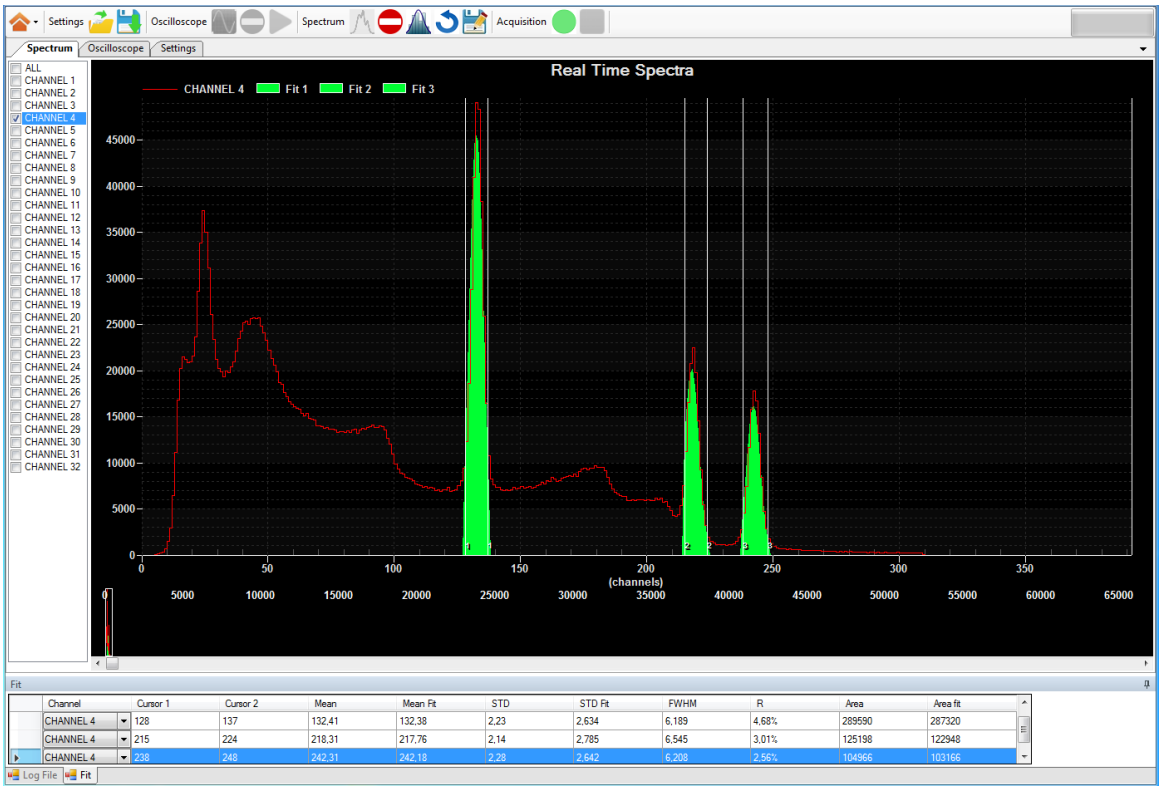



**Figure 16.13:** the fitting tool of the Open Hardware Readout Software.

# How to Save Data

The "**Save Data**" button in the Control Bar allows to save both the spectrum and the cumulative image for the enabled channels on the Spectrum Tab. When the button is pressed a first dialog window is visualized to choose the file path for the spectrum data saving. A second dialog window is shown to set the file path for the cumulative image saving.

- The .csv spectrum file is made of several rows equal to the number of bins in the spectrum. Each row contains the number of events of the correspondent bin for all the channels visualized in the plot.
- The .csv cumulative image file contains the total energy information for all the channels.

The "**Start/Stop Data Record**" buttons in the Control Bar allow to store on file data from the board.
The "Data Record Configuration" window appears after pressing the "Start Data Record" button. The saved file path can be set by clicking the BROWSE button and the channels data to be saved can be set by checking the correspondent checkboxes in the channel list. The "Data Type" option can be used to choose the type of data to be stored:

- Oscilloscope: Every time the oscilloscope triggers, the downloaded waveforms from selected channels is dumped on disk (analog and digital)
- List Mode: The buffered events consisting of energy and time information from the single channel of the board that event by event fires the trigger are stored on the disk.

The "Target Mode" option allows to set how to stop data recording:

- Free: record data until "Stop Data Record" button is pressed
- Time: record data for a specified amount of time (not available for the Oscilloscope Data Type)
- Events: record data until a specified number of events is reached.

The amount of time or the number of events in the respective recording mode can be set in the "Target Value" field.

To start the Oscilloscope or the Frame dumping press the START button.

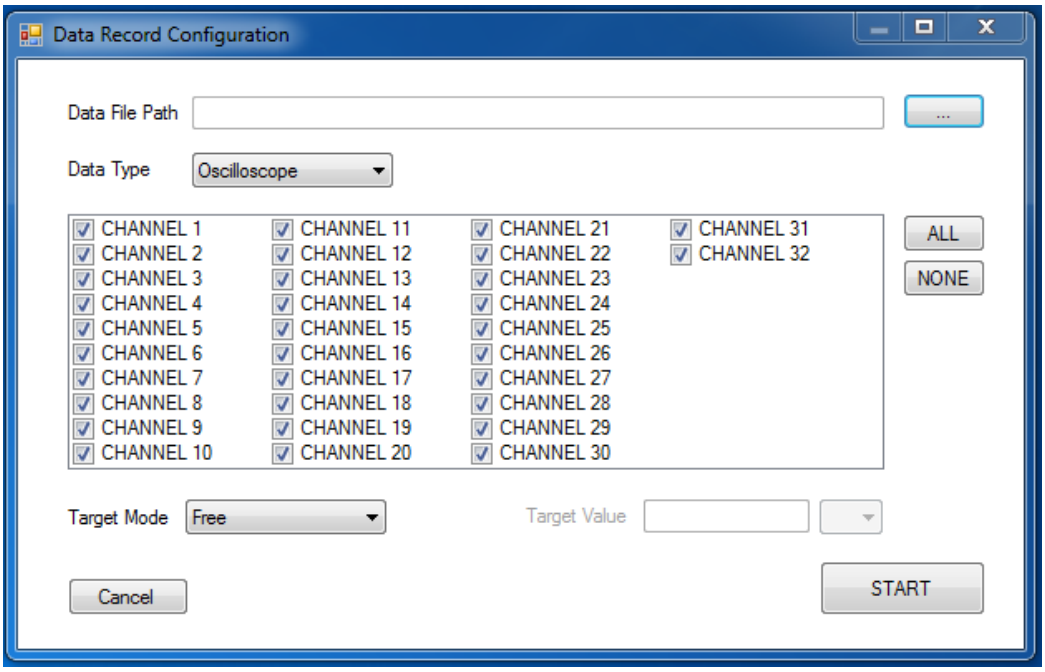



**Figure 16.14:** the Data Record Configuration window.

Each line of the .csv file for the Oscilloscope Mode contains the following information:

- Event number;
- Channel number;
- Number of waveform samples;
- Number of waveforms (5: two analog and three digital);
- Waveform values of the five signals.

Each line of the .csv file for the List Mode contains the following information:

- Event channel;
- Event time tag;
- Event energy.

# 17 Appendix

In this Appendix we give more details about Ethernet and Optical communication protocols available for R5560SE boards. We take as example the R5560SE board: the description of the protocols can be extended to the other board models.

## Ethernet Protocol

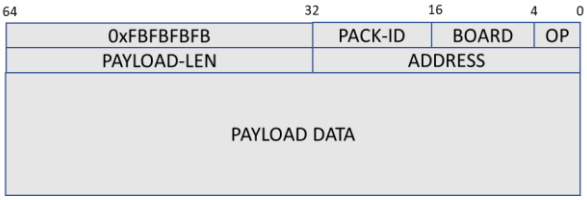
The standard application for R5560SE implements three different servers:

- a TCP web server listening on the port 8899 that supports binary communication
- a TCP web server listening on ports 8900, 8901, 8902 and 8903 that are directly connected to the DMA managed FIFO. This socket is a monodirectional link that stream any data that the FPGA push in the DMA-managed FIFO to the end user client
- a web socket application based on textual JSON messages

The ethernet protocol is the same for the Gigabit LAN connections and for the USB 2.0 connection. USB 2.0 emulates an ethernet card on the host client.

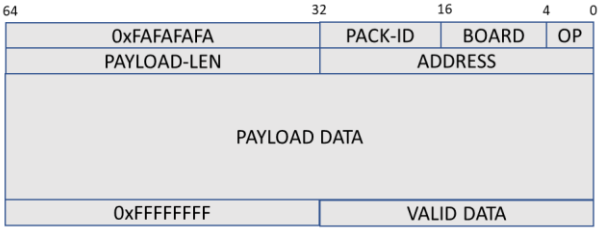
### TCP WEB SERVER

The TCP web server listens on the port 8899 for messages. These messages are then converted by the software application in read/write to the FPGA. This server runs on all DAQ board and on the SCSP board (on slow control ethernet port).
A packet with the following format is sent from the client (PC) to the server (DAQ/SCSP) application:




PACK-ID: incremental number generated by the CB that identify a particular packet.
BOARD: always 0xFFF if direct connected to the DAQ. 0x000 (SCSP), 0x001 (DAQ 1), 0x002 (DAQ 2), 0x003 (DAQ 3), 0x004 (DAQ 4) to identify the sub DAQ board if the client is connected to the Slow Control Board.
OP: identify the operation
0000: enumerate board. Return a json data that enumerate the board and all specs
0001: write with address incrementation
0010: write without address incrementation
0011: read with address incrementation blocking
0100: read without address incrementation blocking
0101: read with address incrementation non-blocking
0110: read without address incrementation non-blocking
PAYLOAD-LEN: number of D-WORD of data to be transferred
ADDRESS: destination address base address
PAYLOAD: data to be transferred to the destination

The DAQ (or the SCSP) answers to the operation with a packet like the following:




PACK-ID: copy the PACK-ID of the requesting packet from the CB
BOARD: always 0xFFF if direct connected to the DAQ. 0x000 (SCSP), 0x001 (DAQ 1), 0x002 (DAQ 2), 0x003 (DAQ 3), 0x004 (DAQ 4) to identify the sub DAQ board if the client is connected to the Slow Control Board.

OP: copy the CB request OP
PAYLOAD-LEN: copy the CB request PAYLOAD
ADDRESS: copy the CB request ADDRESS
PAYLOAD: data transferred from DT55560 to CB
VALID-DATA: number of D-WORD effectively transferred in the payload. If the payload is longer than valid data, data after the first VALID-DATA will be discarded.
ERROR CODE: 0xFFFFFFFF means no error, otherwise an error code is returned

### FIFO SERVER

If DMA-based FIFO is instantiated in the firmware, for each FIFO a new TCP server is created starting from the port 8900. These web servers run only on DAQ board and are not accessible from SCSP. There is no protocol on this socket. What is pushed in the DMA FIFO on FPGA side, will appear as is on the end user PC. FIFO are 32-bit wide and data should be casted to uint32.

### JSON SERVER

The JSON server lists on port 8000 for WebSocket connection. WebSocket is a textual protocol supported by all modern browser, JavaScript and all development tools. The performance on WebSocket is lower that using raw socket but is human readable and produce structured results. All slow control and DAQ configuration are managed through this web socket and all operation listed below can be performed:

**Set Register** *{"command":"set_register", "name":"<register_name>", "value":int_value}*

**Get Register** *{"command":"get_register", "name":"<register_name>", blocking:true/false}*

**Write Memory** *{"command":"write_mem", "name":"<memory_block_name>", "data":[,,,,,,]}*

**Read Memory** *{"command":"read_mem", "name":"<memory_block_name>", blocking:true/false, increment:true/false }*

**Set timeout** *{"command":"set_timeout", int: milliseconds}*

Bloking indicates if the command stalls until all data are available or timeout.
Increment indicates if address should be increment during the read.

Others configuration command can be issued:

**Change Eth CFG** *{"command":"set_eth0", "dhcp":true/false, "ip":"x.x.x.x", "netmask":"x.x.x.x", "gateway":"x.x.x.x"}*

**Change board id** *{"command":"set_id", "value":xxx}*

**Set ADC clock source** *{"command":"set_adc_clock", "value":"internal/system/optolink/hlink"}*

**Set H-LINK clock source** *{"command":"set_adc_clock", "value":"internal/system/optolink"}*

**Get Board temperature** *{"command":"get_temp"}*

**Get monitor voltages** *{"command":"get_volts"}*

**Reboot** *{"command":"reboot"}*

**Save Settings** *{"command":"save"}*

The answer to the command will be:

*{"command":"<command>", "result":true/false, "reason":"…", "data":[,,,,,]}*

Data field will contain the read data as uint32 in 10-base digits.

# Optical Link Protocol

The R5560SE has **8 SFP+ connector that can host optical transceivers operating at a bitrate up to 12.5Gbps**. Each link is bidirectional and can be used for point to point connection or in daisy chain mode. Each DAQ section manages two independent links (two SFP+ connector).
The R5560SE can be easily interfaced with large readout infrastructure using the SDK and developing in VHDL a custom protocol with daisy chain capabilities.

The connection between optical link and the FPGA are different in function of the model of the FPGA installed in the R5560SE. The main differences in the connection scheme is that the Zynq 7030 version (R5560SE) does not allows the propagation of the clock on the optical link in daisy-chained system, while 7035 model (R5560SEB) allows for it.
In order to propagate the clock in a daisy chain of boards, it is mandatory to recover the clock from the gear box of the GTx (RX), clean it with an external jitter cleaner and provide this clock as a clock reference for the TX GTx.
The RX channel needs a reference clock for correct operation. This should be a clean, low jitter clock generated locally on the RX channel. This clock cannot be recovered from the link. In order to propagate the clock, it is necessary to recover the clock and feed it as a reference clock to the TX.
Xilinx FPGAs have just one QPLL per Quad-block (block of four transceivers). The QPLL is the component that generates all high-speed frequencies that are necessary to serialize the data in the transceiver. If the QPLL is already used for the RX, it cannot also handle the clock for the TX. The TX clock (recovered from RX circuit) must be propagated providing it as a reference clock of the second Quad-Block of transceiver.

Different models of the Zynq SoC, are based on different connection schemes for the optical links operation:

- **R5560/R5560SE**
  The Zynq 7030 has just 4 GTx transceivers (in just one quad-block). It is indeed impossible (due to Xilinx FPGA limitation) to propagate the clock in a daisy chain. All four GTx are used, two managing the SFP+ connector while the other two are used for the internal H-LINK. Internal H-LINK and external SFP+ will operate with the same reference clock (it does not mean that will have the same bit rate; each transceiver has a CPLL that allows to change the link speed starting from a common external reference clock).
  The recovered clock is cleaned with the jitter cleaner and forwarded back to the FPGA logic. It is used for event timestamping, ADC clocking and propagated on the H-LINK to nearby DAQ sections.

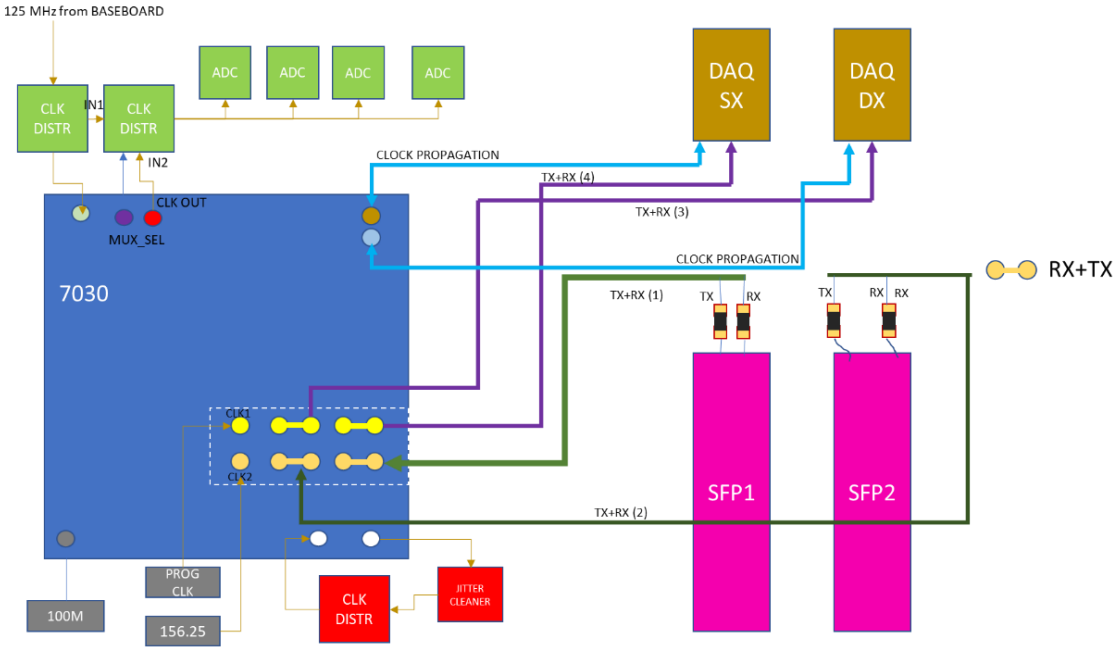



**Figure 17.1:** scheme for optical links operation for R5560SE

- **R5560B/R5560SEB**
  The Zynq 7035 has 8 GTx transceivers (in two quad-blocks). So, it possible to propagate the clock in a daisy chain. The SFP+ connector can be connected to GTX2 of the first quad-block (clock propagation is not possible) or to the GTX1 of the second quad-block (clock propagation is possible). The clock recovered from the RX channel in the first quad-block is cleaned and provided back to the FPGA and to the TX GTX on the second quad-block. The TX will be clocked with the cleaned version of the RX clock, allowing to propagate down the clock in the chain. The recovered clock provided to the FPGA logic can be used for clock timestamping logic, ADCs and it is propagated to nearby DAQ sections.

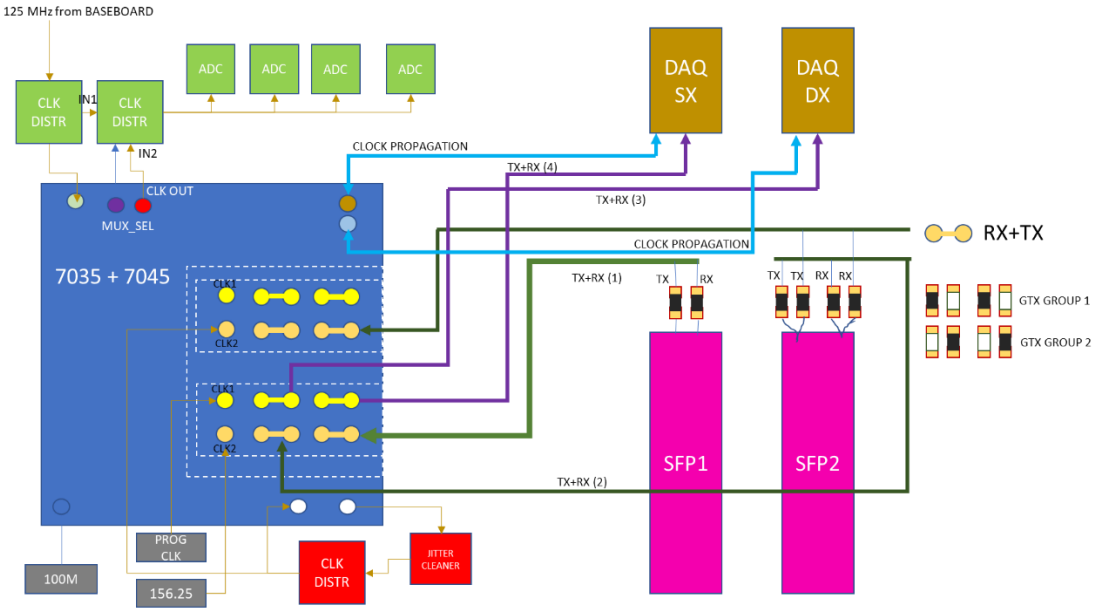



**Figure 17.2:** scheme for optical links operation for R5560SEB

# Libraries usage examples

The following example in C shows how to connect to the R5560SE through the IP address, to set the parameters of the Oscilloscope component, to start a data acquisition, to read the Oscilloscope buffer status and download the data array.

```
//define and initialize all the variables
tR5560SE_HANDLE handle;         //board connection handle
uint32_t decimator = 0;         //no decimation of oscilloscope data
uint32_t pre_trigger = 100;     //number of samples between the first data and the
trigger
uint32_t trigger_level = 10000;//threshold in lsb to be exceeded to fire the
trigger
uint32_t trigger_mode = 1;      //trigger on the rising edge of the analog signal
uint32_t status_osc = 0;        //status of the oscilloscope data buffer
uint32_t position = 0;          //number of samples indicating the trigger position
uint32_t data_osc[1024];        //oscilloscope data array
uint32_t size_osc = 1024;       //size of the oscilloscope data array
uint32_t read_data_osc;         //number of oscilloscope data read
int Osc_Events = 10;            //number of oscilloscope data array to be
downloaded
int e = 0;                      //iteration variable
#define REG_CONFIG_DECIMATOR 0x00000001
#define REG_CONFIG_PRETRIGGER 0x00000002
#define REG_CONFIG_TRIGGER_LEVEL 0x00000003
#define REG_CONFIG_TRIGGER_MODE 0x00000004
#define REG_CONFIG_ARM 0x00000005
#define REG_READ_STATUS 0x00000006
#define REG_READ_POSITION 0x00000007
#define REG_DATA_ADDRESS 0x00000008

//connect to the R5560SE board
if(R5560SE_ConnectTCP("192.168.50.150", 8888, &handle) != 0)
       {printf("Unable to connect to the board!\n"); return (-1); };

while (e<Osc_Events)
{
       //set oscilloscope parameters
       if (NI_WriteReg(decimator, REG_CONFIG_DECIMATOR, &handle)!= 0)
       printf("Set Decimator Error");
```

```
      if (NI_WriteReg(pre_trigger, REG_ CONFIG_PRETRIGGER, &handle)!= 0)
      printf("Set Pre Trigger Error");
   if (NI_WriteReg(trigger_level, REG_CONFIG_TRIGGER_LEVEL, &handle)!= 0)
      printf("Set Trigger Level Error");
   if (NI_WriteReg(trigger_mode, REG_CONFIG_TRIGGER_MODE, &handle)!= 0)
      printf("Set Trigger Mode Error");

   //start oscilloscope acquisition
   if (NI_WriteReg(0, REG_CONFIG_ARM, &handle)!= 0)
      printf("Set Config Error");
   if (NI_WriteReg(1, REG_CONFIG_ARM, &handle)!= 0)
      printf("Set Config Error");

   //wait until the oscilloscope status indicates that data are available
   while (status_osc != 1)
         if (NI_ReadReg(&status_osc, REG_READ_STATUS, &handle) != 0)
      printf("Status Error");

   //read the position of the trigger
   if (NI_ReadReg(&position, REG_READ_POSITION, &handle) != 0)
printf("Position Error");

   //read the oscilloscope data
      if (NI_ReadData(&data_osc, size_osc, REG_DATA_ADDRESS, &handle,
&read_data_osc) != 0)
         printf("Get Data Error");
   e++;
} printf("Download Finished");
```

The following example in C shows how to connect to the R5560SE through the IP address, to start a data acquisition from a FIFO buffer provided by the Custom Packet component, to read its buffer status and download the data array. Data are organized inside the Custom Packet with a layout defined by the user during the component creation.

```
//define and initialize all the variables
tR5560SE_HANDLE handle;        //board connection handle
uint32_t status = 0;           //status of the custom packet data buffer
uint32_t data[100000];         //custom packet downloaded data array
uint32_t N_Packet = 100;       //number of data array packets to be downloaded
uint32_t N_CP = 4;             //number of custom packet word of the data layout
uint32_t bus_mode = 1;         //blocking streaming access to the fifo buffer
uint32_t timeout = 1000;       //read data timeout expressed in milliseconds
uint32_t read_data;            //number of custom packet data read
uint32_t N_Total_Events = 1000;//total number of data to be downloaded
uint32_t ReadDataNumber = 0;   //iteration variable
#define REG_CONFIG 0x00000001
#define REG_READ_STATUS 0x00000002
#define REG_DATA_ADDRESS 0x00000003

//connect to the R5560SE board
if(R5560SE_ConnectTCP("192.168.50.150", 8888, &handle) != 0)
      {printf("Unable to connect to the board!\n"); return (-1); };

//start custom packet data acquisition
if (NI_WriteReg(0, REG_CONFIG, &handle) != 0)
      printf("Reset Error");
if (NI_WriteReg(1, REG_CONFIG, &handle) != 0)
      printf("Start Error");

//check if the custom packet status indicates that data are available
if (NI_ReadReg(&status, REG_READ_STATUS, &handle) != 0)
      printf("Status Error");
```

```
if (status >0)
{
    while (ReadDataNumber <N_Total_Events)
    {
           //read the data from the custom packet fifo buffer
if (NI_ReadFifo(&data, N_Packet * (N_CP + 3), REG_DATA_ADDRESS,
                REG_READ_STATUS, bus_mode, timeout, &handle, &read_data) != 0)
      printf("Data Download Error");

           ReadDataNumber = ReadDataNumber+ N_Packet;
    }
    printf("Download completed");
}
else printf("Status Error");
```

# 18 Instructions for Cleaning

The equipment may be cleaned with isopropyl alcohol or deionized water and air dried. Clean the exterior of the product only.
Do not apply cleaner directly to the items or allow liquids to enter or spill on the product.

## Cleaning the Touchscreen

In order to clean the touchscreen (if present), wipe the screen with a towelette designed for cleaning monitors or with a clean cloth moistened with water.
Do not use sprays or aerosols directly on the screen; the liquid may seep into the housing and damage a component. Never use solvents or flammable liquids on the screen.

## Cleaning the air vents

It is recommended to occasionally clean the air vents (if present) on all vented sides of the board. Lint, dust, and other foreign matter can block the vents and limit the airflow. Be sure to unplug the board before cleaning the air vents and follow the general cleaning safety precautions.

## General cleaning safety precautions

CAEN recommends cleaning the device using the following precautions:

1) Never use solvents or flammable solutions to clean the board.
2) Never immerse any parts in water or cleaning solutions; apply any liquids to a clean cloth and then use the cloth on the component.
3) Always unplug the board when cleaning with liquids or damp cloths.
4) Always unplug the board before cleaning the air vents.
5) Wear safety glasses equipped with side shields when cleaning the board

# 19 Disposal

The disposal of the equipment must be managed in accordance with Directive 2012/19 / EU on waste electrical and electronic equipment (WEEE).

The crossed bin symbol indicates that the device shall not be disposed with regular residual waste.

# 20 Technical Support

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

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

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


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

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

*https://www.caen.it/wp-content/uploads/2022/11/Safety_information_Product_support_W.pdf*

**CAEN GmbH**
Eckehardweg 10
42653 - Solingen
Germany
Phone +49 212 254 40 77
Fax +49 212 254 40 79
info@caen-de.com
**www.caen-de.com**

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

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

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

*UM6952 - R5560/R5560SE rev. 7 - July 16th, 2024 00000-00-R5560-MUTX*