Open Source Remote Diagnostics Platform for Custom Instrumentation in Nuclear Applications

Iván René Morales[1(✉)], Maria Liz Crespo[1], and Sergio Carrato[2]

[1] Multidisciplinary Laboratory (MLab), STI Unit, The Abdus Salam International Centre for Theoretical Physics (ICTP), Strada Costiera, 11, 34151 Trieste, Italy
{imorales,mcrespo}@ictp.it

[2] Dipartimento di Ingegneria e Architettura (DIA), Universitá degli Studi di Trieste, Piazzale Europa, 1, 34127 Trieste, Italy
ivanrene.moralesargueta@phd.units.it, carrato@units.it

Abstract. An open source remote diagnostics platform for custom electronics in experimental acquisition setups has been developed, targeting nuclear and high-energy physics (HEP) applications. We aim at enabling remote access to instrumentation hardware prototypes located in radiation-controlled areas by using existent network infrastructures. The platform relies on two components: a graphical user interface (GUI) developed in GNURadio and a remote hardware bridge (HB). The GUI was designed using the GNURadio's optimized building blocks, ensuring a fluid visualization of real-time pulse traces and energy spectrum. No third-party libraries are used, turning our solution into an easy drop-in tool for instrumentation diagnostics in HEP and radiation-related experiments. Reliable communication between the remote instrument and the GUI is accomplished using TCP/IP, carried out by the HB in case of remote instruments without network interface. A Raspberry PI Zero-W is tested as the HB to demonstrate the few computational resources required for this end.

Keywords: Nuclear instrumentation · GNURadio · GUI · Remote diagnostics · Hardware prototypes

1 Introduction

Testing custom hardware prototypes for acquisition systems in experiments related to nuclear or particle physics instrumentation usually requires scheduling time slots for measurements during a beam run [1]. Thus, having a graphical user interface (GUI) to assess the behavior of the detectors in real time is crucial to quickly verify the system response under controlled stimuli. Since no physical access to radiation-controlled areas is allowed during the beam operation, personnel are forced to be located in a control room away from the detectors.

Commercial solutions exist for both remote oscilloscope and multichannel analyzer (MCA) applications [2,3], which are the most commonly used tools for this testing stage [4]; however, the compatibility is limited to the vendor-specific products and no APIs are available to create wrappers for custom hardware solutions. Moreover, many of them require a dedicated computer to be placed in the radiation-controlled area to act as a bridge with the instrument prototype under test to enable the connectivity via remote desktop.

Our proposal addresses both features in a single solution: a GUI for real-time diagnostics and a hardware bridge (HB) to enable remote access for instrument prototypes without network connectivity. The open source GUI was developed using the GNURadio software development kit, taking advantage of its building blocks, which are already optimized for fluid visualization and real-time multi-core processing, regardless of the operating system [5]. Hence, the intensive computation tasks are carried out in the computer where the GUI is being executed, making possible to use low-end and low-power devices as the remote HB. Moreover, no third-party libraries are required, enabling a flawless deployment to easily start testing data acquisition system (DAQ) prototypes. To ensure the reliability of the transmitted data, the communication between the DAQ and the GUI is carried out using the transport control protocol (TCP).

In terms of alternative communication protocols, some commercial solutions rely on dedicated high-level solutions built atop TCP, such as LAN eXtensions for Instrumentation (LXI) [6], which consumes significant computational resources at the DAQ side and increase the development time for diagnostics purposes. Other protocol alternatives exist to address real-time remote data transmission: Padded Jittering Operative Network (PJON) designed to exploit noisy or long transmission distances on wireless interfaces [7], or Lab Streaming Layer (LSL), designed for multiple channel brain signals [8]; however, they require third-party or custom libraries to provide GNURadio compatibility and may hinder the deployment process with missing dependency or version problems [9]. Our platform overcomes these issues by using TCP transactions managed by GNURadio built-in blocks, leveraging existing cabled or wireless network infrastructures. Moreover, the GUI is compatible with multiple operating systems and the HB can be deployed in many different models of inexpensive off-the-shelf low-power single-board computers (SBC). These features provide flexibility to use the available hardware at the laboratories without compromising the fluid oscilloscope and MCA visualization.

2 Methodology

We chose a custom DAQ under development as the instrument to be diagnosed. In our tests, the DAQ was connected to the proposed HB via universal serial bus (USB) emulating a universal asynchronous receiver-transmitter (UART) interface. Three different gamma detectors were tested to assess the flexibility of the platform under diverse signal types: a sodium-iodide (NaI:Tl) crystal with a silicon photomultiplier (SiPM), a NaI:Tl crystal with a photomultiplier tube (PMT), and a CLYC (Cs2LiYCl6:Ce) crystal with a SiPM.

Albeit the HB can be deployed using several Linux-compatible SBC models, we opted for a low-end and low-power Raspberry PI Zero W (version 1.1) to demonstrate the few computational requirements needed for its operation and the flexibility to power it from any available USB port or phone charger. This SBC features only 512 MB of DDR2 RAM and a single-core 1 GHz ARMv6 CPU: humble specifications compared to other models available nowadays. We measured the power consumption during one hour of heavy CPU load and continuous WiFi transmission, using a 5 W USB adapter as power supply.

The GUI was tested in a desktop computer (located in the control room) with 16 GB of DDR4 RAM and a Core i7-8700 processor running GNURadio version 3.10.5.1 on Linux Ubuntu 20.04.6. The DAQ and detectors were located about ten meters away, in the radiation-controlled area. Figure 1 shows the proposed remote diagnostics platform block diagram, including all the elements of the system. It is worth noting that the HB acts as an auxiliary element in case the DAQ under development is not provided with network connectivity; otherwise, the computer running the GUI may be directly connected to the DAQ.

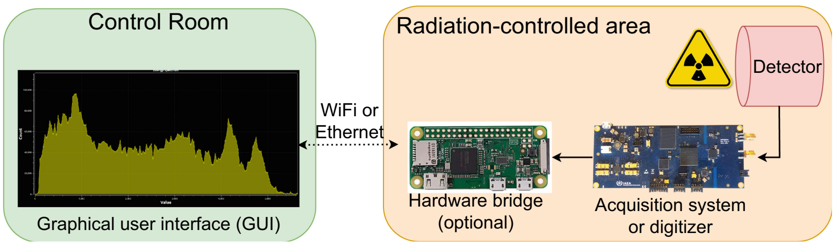

Fig. 1. Block diagram of the system components distribution, including the hardware bridge between the remote digitizer and the network infrastructure.

3 Results

The oscilloscope and MCA were tested with a remote DAQ connected to multiple detectors (one at a time) stimulated with a Cobalt-60 (Co-60) gamma source. The source was located close to the scintillators and served as a reference for pulse shape visualization and energy spectrum computation. The HB was able to transmit the data using its wireless interface between the DAQ (at the radiation-controlled area) and the computer executing the GUI (in the control room). A sustained throughput close to 2.1 Mbps was measured, while the system drew an average of 934 mW from the power supply. Moreover, rates close to 27 Mbps were reached with a cabled Ethernet connection. The open source GUI source files and the script for the HB are available at the online repository [10].

3.1 Graphical User Interface

Since the GUI was developed in GNURadio, it is also compatible with multiple operating systems (Linux, MacOS, Windows). Moreover, a GNURadio port for Android (currently under development) may also enable its execution on a smartphone or tablet [11]. Such flexibility provides access to the DAQ or detector under diagnostics from mostly any available device in the control room of the experiment. The GUI features some of the most common tools required for debugging and diagnostics of nuclear and particle physics detectors and digitizers, i.e. oscilloscope (real-time trace visualization) and MCA (energy spectrum) [12]. Stability of both GUI elements during long-term measurements was tested in 24-hours continuous sessions with no issues reported or detected.

The following settings are required to be defined prior to the first execution of the application: remote IP address and port number (either from the HB or the remote DAQ), sampling rate of the DAQ, and trace length (in sample count). After setting these values up the GUI can be executed directly from GNURadio, leading to a window with two tabs (oscilloscope and MCA), described next.

Oscilloscope Based on GNURadio's oscilloscope block, provides the user the basic tools required for the real-time diagnostics of scintillation detectors and the DAQ prototypes attached to them, such as baseline, pulse shape and pulse amplitude distribution. Figure 2 shows the oscilloscope tab featuring the pulse shape of a NaI:Tl scintillator coupled to a SiPM and digitized by the remote DAQ under test. During run-time, this tool provides the following features and settings: cross-level trigger threshold with pulse polarity selection, multiple trigger sources (signal, peak detector), baseline restorer (BLR), peak detector and pulse height analysis (PHA), and recording of individual traces in binary format. An update rate close to 15 traces per second was measured with short trace lengths (128 samples) in cabled Ethernet mode.

Multichannel Analyzer Based on GNURadio's histogram block, displays the estimated deposited energy spectrum in the detector. The individual energy of each event is computed as the peak amplitude (using the PHA) of the pulse minus its baseline (from the BLR value). Flexibility to the user is provided by tuning the following settings during execution: number of histogram bins, autoscale and window zooming, limited number of events or accumulative spectrum. Figure 3 shows the MCA tab with experimental data built upon a Co-60 γ source located close to the NaI:Tl detector with SiPM, shown in ADC channels. The two characteristic photopeaks at 1.17 MeV (above channel 4000) and 1.33 MeV (below channel 5000) as well as the Compton back-scatter peak (210 keV) and the Compton edge (936 keV) evidence the expected behavior of the system.

3.2 Hardware Bridge with SBC

The tested HB demonstrated consistent remote data transmission using the embedded WiFi 802.11n peripheral. Configuring the SBC simply required

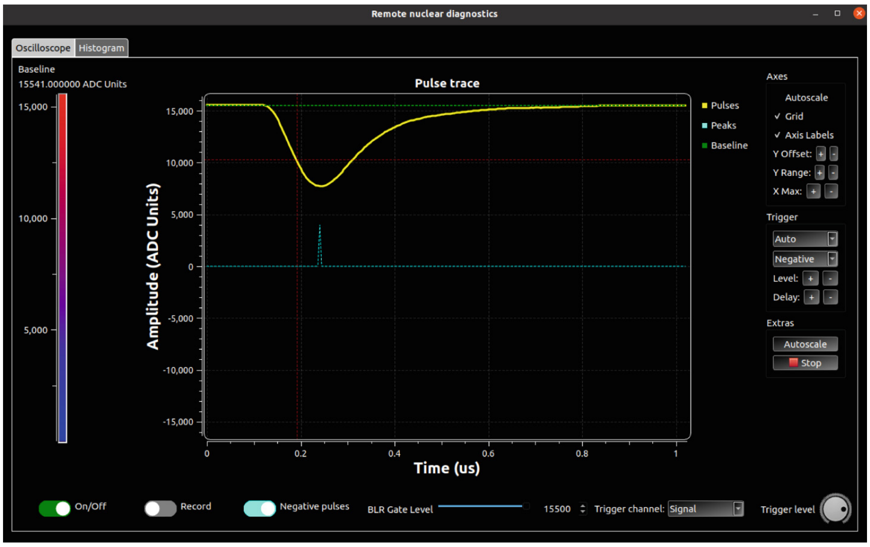

Fig. 2. Oscilloscope tab for real-time visualization of the triggered traces showing experimental data from a Co-60 gamma source close to a NaI:Tl SiPM detector.

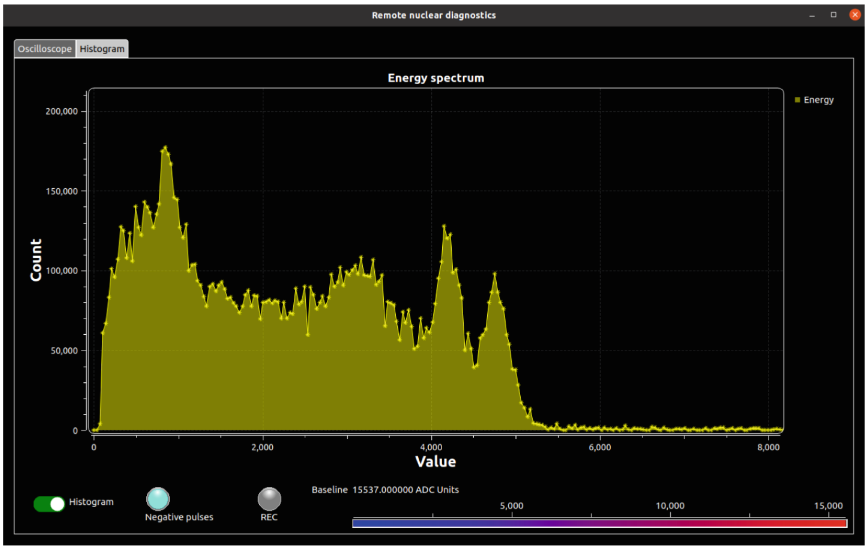

Fig. 3. Multichannel analyzer (MCA) tab. The histogram represents the accumulated energy spectrum of measured pulses from the NaI:Tl crystal with SiPM when placed close to the Co-60 γ source. The spectrum matches the expected detector response.

installing the default operating system (Raspberry PI OS, release 20230503) into a 8 GB micro-SD card and setting up the WiFi access-point configuration. Taking the hardware bridge script up and running only required to download the source code from the provided public repository and execute the Python script.

4 Conclusions

The idea of this platform aroused from the necessity of deploying a fast diagnostics system for custom instrumentation hardware under experimental conditions. An open source platform for remote DAQ diagnostics has been consequently developed and tested with diverse detectors. The system enables an easy setup to test instrumentation prototypes with an oscilloscope and multichannel analyzer from remote locations taking advantage of existent network infrastructures. The open source GUI is optimized for multi-core processing to provide a fluid visualization of the data transmitted from the DAQ under test and no third-party libraries are required to execute it, apart from installing GNURadio. A remote hardware bridge has also been developed to enable connectivity for remote instruments that lack of network capabilities. The proposed platform provides a drop-in solution for fast debugging and diagnostics of custom electronics in nuclear and HEP experiments.

References

1. Diener R et al (2019) The DESY II test beam facility. Nucl Instrum Methods Phys Res Sect A: Accel Spectrometers Detect Assoc Equip 922:265–286. https://doi.org/10.1016/j.nima.2018.11.133
2. CAEN: CAENScope (2023). https://www.caen.it/products/caenscope/. Accessed 15 June 2023
3. AMETEK: MAESTRO multichannel analyzer emulation software (2023). https://www.ortec-online.com/products/application-software/maestro-mca. Accessed 15 June 2023
4. Crespo ML et al (2021) Remote laboratory for e-learning of systems on chip and their applications to nuclear and scientific instrumentation. Electronics 10:2191. https://doi.org/10.3390/electronics10182191
5. Valerio D (2008) Open source software-defined radio: a survey on GNUradio and its applications
6. Mumford R (2007) The LXI standard: past, present and future. Microwave J 50
7. Mitolo GB (2022) PJON (Padded Jittering operative network). https://github.com/gioblu/PJON. Accessed 22 June 2023
8. Stenner T et al (2022). Lab streaming layer project. https://doi.org/10.5281/zenodo.6387090, github.com/sccn/liblsl/tree/v1.16.0
9. USRP (2020) LSL toolkit for GNURadio/USRP data. https://github.com/lwa-project/usrp. Accessed 22 June 2023
10. Morales I (2023) Remote diagnostics platform for custom instrumentation in nuclear applications—code repository. https://doi.org/10.5281/zenodo.8082581. Accessed 10 July 2023

11. Bloessl B, Baumgärtner L, Hollick M (2020) Hardware-accelerated real-time stream data processing on android with GNU radio. ACM 103–109. https://doi.org/10.1145/3411276.3412184
12. Yang J, Xiong M, Zhao X (2020) Improved low power multichannel pulse amplitude analyzer. OALib 07:1–7. https://doi.org/10.4236/oalib.1106629