Advanced Digital Spectroscopy Channel for Signal Analysis and Diagnosis Based on SoC FPGA

Jorge Jiménez-Sánchez, Manuel Sánchez-Raya, José María Hinojo-Montero, *Member, IEEE*, Fernando Muñoz-Chavero, Juan Antonio Gómez-Galán, and Ismael Martel

***Abstract*—This article presents the design and implementation of a data acquisition system for a digital spectroscopy channel based on a field-programmable gate array (FPGA) integrated in a system-on-chip (SoC). The system integrates two independent modules: a pulse emulator and a pulse analyzer. The emulator generates signals by synthesizing pulses using Poisson-distributed arrival times and Gaussian-distributed amplitudes, as well as custom configurations to simulate effects such as pile-up. The analyzer captures and processes these signals, extracting key parameters such as pulse height and arrival time. It also includes FPGA hardware processing with customizable and configurable filters, speeding up processing before transferring the results to the SoC for storage and visualization. The system is implemented on a low-cost platform based on an AMD Xilinx Zynq 7010 SoC, with programmable FPGA logic and an ARM Cortex-A9 processor running Linux Ubuntu. This SoC FPGA enables fast signal processing and extraction of key parameters and histogram generation in the FPGA, while the microprocessor manages communication and control tasks. A client–server software model based on a TCP/IP architecture and using standard commands for programmable instruments (SCPI) commands facilitates remote control and data visualization through a Python-based graphical interface. By leveraging the emulator and analyzer in a closed-loop configuration, the system enables self-testing and calibration, allowing fine-tuning of delays and signal losses through comparative analysis of both modules. In addition, the emulator can operate independently as a configurable pulse source for applications in nuclear engineering, particle accelerators, and medical engineering. This setup eliminates the need for radioactive sources, enabling precise and safe analysis in spectroscopy applications. Laboratory tests and validation under real radiation conditions, including comparison with commercial systems, demonstrated the system's accuracy and flexibility, highlighting its usefulness in spectroscopy. The integration of both modules into a single device provides flexibility and cost reduction, with performance competitive with state-of-the-art digital spectroscopy systems reviewed.**

***Index Terms*—Client–server model, digital spectroscopy, field-programmable gate array (FPGA), hardware preprocessing, pulse analyzer, pulse emulator, radiation detectors, signal processing, system on chip (SoC).**

I. Introduction

NUCLEAR spectroscopy, as a branch of nuclear and applied physics, studies the behavior of atomic nuclei and their properties based on the radiation they emit [1]. Its foundation lies in the precise measurement of the energy spectrum associated with different forms of ionizing radiation [2]. This requires the detection and processing of signals generated when nuclei interact with a detector [3]. Several types of detectors are employed in this technique: scintillation detectors, gaseous detectors (such as ionization chambers or proportional counters), solid-state detectors (e.g., diamond or sapphire), and, most widely, semiconductor detectors, among which high-purity germanium (HPGe) and silicon detectors stand out [4].

When a particle interacts with the detector, a current pulse is generated whose height is proportional to the energy deposited in the interaction. To measure this pulse accurately, a signal processing chain is employed [5]. Traditionally, this chain [see Fig. 1(a)] begins with a preamplifier, responsible for adapting and amplifying the detector signal. Subsequently, a shaper or pulse amplifier applies analog filtering to improve the signal-to-noise ratio. Additional modules, such as discriminators and timers, allow the selection and organization of relevant events. The maximum amplitude of the filtered signal is then digitized by an analog-to-digital converter (ADC). Finally, a multichannel analyzer (MCA) receives these digitized values and constructs a histogram that represents the energy spectrum. The analysis of processed pulses makes it possible to extract key information, such as deposited energy, arrival time, or pulse shape [6]. These parameters, translated into physical magnitudes, allow for the identification of different types of particles and the characterization of nuclear processes [7].

Recent technological advances, particularly in high-speed ADCs and field-programmable gate arrays (FPGAs), have enabled the replacement of fully analog processing chains [see Fig. 1(a)] with digital or hybrid approaches [see Fig. 1(b)] [8], [9]. This paradigm shift has allowed spectroscopy—once exclusive to nuclear physics laboratories [10], [11] (e.g., the study of high-energy collisions and radioactive decay)—to be easily extended to multiple scientific and technical fields such as medicine [12], astrophysics [13], industry [14], security and defense [15], archeology [16], agriculture [17], energy [18], and environmental monitoring [19].

In this process of transition and expansion, engineering plays a key role in developing new instrumentation, analysis,

Received 22 December 2025; accepted 4 February 2026. Date of publication 23 February 2026; date of current version 19 March 2026. This work was supported by grants PID2021-127711NBI00 and PID2024-159209NB-C21 funded by MCIN/AEI/10.13039/501100011033 and, as appropriate, by the "European Union NextGenerationEU/PRTR". The Associate Editor coordinating the review process was Dr. Pawel Kwiatkowski. *(Corresponding author: Juan Antonio Gómez-Galán.)*

Jorge Jiménez-Sánchez, José María Hinojo-Montero, and Fernando Muñoz-Chavero are with the Department of Electronic Engineering, University of Seville, 41092 Seville, Spain (e-mail: jjsanchez@us.es; jhinojo@us; fmunoz@us.es).

Manuel Sánchez-Raya and Juan Antonio Gómez-Galán are with the Department of Electronic Engineering, Computers and Automation, University of Huelva, 21007 Huelva, Spain (e-mail: msraya@diesia.uhu.es; jgalan@diesia.uhu.es).

Ismael Martel is with the Department of Applied Physics, University of Huelva, 21007 Huelva, Spain (e-mail: imartel@uhu.es).

Digital Object Identifier 10.1109/TIM.2026.3667282

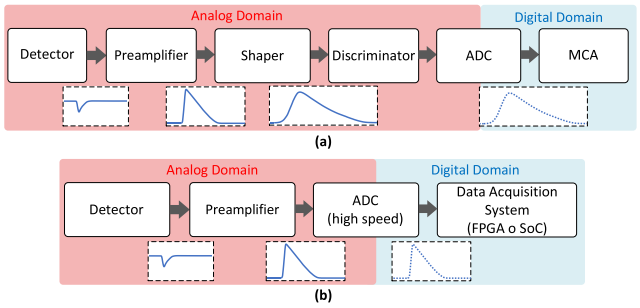


Fig. 1. Comparison of (a) traditional analog and (b) digital spectroscopy channels.

and signal processing techniques [20]. Its extension to new contexts has also favored the use of spectroscopy in the educational field as a tool for teaching radiation concepts and signal processing in scientific and technological training [21].

However, in these new contexts, three main limitations arise: 1) restricted access to specialized laboratories; 2) the high cost of commercial digital systems; and 3) the rigidity of such systems to adapt to specific needs.

First, access to nuclear physics laboratories is limited due to the high operational costs and strict radiological safety requirements, making it difficult to perform experiments with the frequency and flexibility demanded by emerging fields such as engineering or education.

Second, commercial digital equipment used in spectroscopy is costly, which poses a barrier for universities, research centers, and projects with limited resources.

Moreover, these systems are often inflexible due to the use of proprietary software, closed interfaces, and restricted access to hardware reprogramming. This hinders the customization of algorithms and integration with external platforms, limiting their applicability in contexts beyond traditional nuclear physics, such as applied research or the development of innovative solutions.

The objective of this work is to design and implement a data acquisition system for a digital spectroscopy chain based on a low-cost board that integrates an ADC, a digital-to-analog converter (DAC), and a system-on-chip FPGA (SoC-FPGA). In addition, the development of system components relies on free and/or accessible software. The open and reprogrammable nature of the system allows for the dynamic implementation and testing of algorithms in the FPGA, adding new functions without modifying the hardware [22], [23]. In this way, the aim is to provide a flexible and accessible tool for research, engineering, and education, demonstrating that signal analysis can be achieved without the prohibitive cost of high-end commercial devices.

The developed system integrates, in the programmable logic (FPGA) of SoC, both real-time signal analysis and processing capabilities, as well as the generation of synthetic signals for particle impact emulation. It consists of two modules: a pulse emulator and a pulse analyzer. The analyzer is responsible for continuous pulse capture and processing, extracting key parameters such as amplitude and arrival time. It includes a configurable digital filter to optimize the signal-to-noise ratio [24] and allows for the processing of pile-up pulses to maximize performance and minimize dead time [25]. The emulator, on the other hand, reproduces the behavior of preamplifiers, facilitating testing, calibration, and system validation without specialized laboratories or radioactive sources. When connected in a loop with the analyzer, it enables system self-evaluation and calibration of delays and losses, and it can also operate independently as a pulse generator for other systems.

On the SoC's processor, a Linux Ubuntu operating system (OS) runs an application based on a client–server architecture. This application allows both modules to be controlled locally or remotely through a graphical interface. Therefore, this solution aims to exploit the advantages that an FPGA-SoC system provides. The FPGA performs high-speed signal processing, leveraging parallelization and hardware acceleration, while the CPU handles higher-level tasks that are difficult to implement in hardware, such as user communication, data storage, and application control and supervision [26]. Furthermore, the inclusion of an OS increases flexibility and facilitates the development of custom applications directly on the SoC [27]. This allows tasks to be automated, configurations saved, and experiments reproducible.

This data acquisition system is fully integrated into a digital spectroscopy architecture, realizing the advantages of the digital chain over the analog one. The signal is digitized immediately after the detector–preamplifier set, eliminating intermediate analog stages that introduce noise and interference. With this reduction of components, the signal-to-noise ratio and energy resolution are improved [28]. In addition, replacing multiple analog modules with a single compact board reduces size, complexity, and cabling. This facilitates integration and portability while lowering power consumption and mitigating thermal stability issues of analog components [29].

This article is organized as follows: Section II presents the state of the art. Section III describes the system architecture. Section IV details the logic and software implementation. Section V presents the experimental validation and results obtained. Section VI provides a discussion of the results and a comparison with other works, and finally, Section VII outlines the conclusions of the work.

II. State of the Art

Numerous studies in spectroscopy have contributed to the evolution from analog spectroscopy chains to digital architectures, driven by the flexibility of digital processing as a means to overcome the limitations of classical analog systems. In the following, we present recently published spectroscopy solutions, analyzed in relation to the three key limitations identified in this work: accessibility, cost, and flexibility.

Some proposals still include analog blocks to perform preprocessing prior to the digital domain. In [30], a digital analyzer was developed for gamma spectroscopy with scintillation detectors, implementing a pulse detection algorithm based on pulse height. However, the preamplifier signal had to pass through an analog shaper before being digitized with a 12-bit, 500 kS/s ADC. This work was extended in [31], which used the same configuration and hardware for a compact digital system and also aimed at gamma spectroscopy, incorporating coincidence experiments with scintillation detectors. In addition, Choi and Chai [32] presented a digital pulse processing

system on a Virtex-5 FPGA, designed to improve resolution and compensate for effects such as pile-up and ballistic deficit. Nevertheless, it retained an analog prefilter before digitization (14-bit, 62 MS/s ADC) in order to adapt the preamplifier signal and reduce noise and pile-up effects. In all these cases, the systems only partially captured the benefits of the digital paradigm since they preserved elements of the classical analog chain.

Other works replaced the processing traditionally carried out in the analog chain with fully digital FPGA processing. In [33], a real-time digital pulse processing system was designed for gamma spectroscopy at high count rates. It implemented adaptive trapezoidal shaping on a Kintex-7 along with a 14-bit, 250 MS/s ADC. The system achieved high resolution and pile-up rejection, but its reliance on a PXIe chassis made it neither compact nor low-cost, limiting its applicability in new spectroscopy contexts. In contrast, Cao et al. [34] proposed a compact, low-cost digital MCA designed for portable devices in nuclear and radiation spectroscopy applications based on an Intel MAX10 FPGA with a 14-bit, 150 MS/s ADC. Its objectives included energy spectrum generation, pile-up rejection, and gain stabilization against temperature variations. Similarly, Zhang et al. [35] developed a real-time digital processing system for gamma spectroscopy, implemented on a Spartan-6 with a 14-bit, 125 MS/s ADC, designed to adapt to different radiation detectors and generate energy spectra in real time. Both systems implemented trapezoidal shaping and pile-up rejection directly in the FPGA and stood out for their compactness and low cost. However, since they were based on FPGAs without an embedded processor, they required an external device for control and communication, reducing their flexibility compared to SoC-FPGA architectures.

A later line of research sought to overcome these limitations using SoC-FPGA platforms, which increase autonomy, flexibility, and portability. In [36], a digital pulse processing chain was developed for nuclear spectroscopy applications on the PYNQ-Z1 platform (Zynq-7020 SoC). The system used its internal 16-bit, 1 MS/s ADC and demonstrated that such digital processing could improve spectrum quality, optimize real-time detection, and facilitate access to advanced tools through an open-source approach. Similarly, Han et al. [37] proposed a method to improve the performance of nuclear spectroscopy systems under pile-up conditions, implementing a hybrid cusp–flattop filter on a Zynq-7020 SoC with a 12-bit, 65 MS/s ADC. These proposals highlight the advantages of SoC integration, but none incorporate pulse emulation, limiting accessibility and validation without radioactive sources.

At the high-performance end, some works explored solutions based on MPSoC and external digitizers. In [38], a system was developed to discriminate neutrons and gamma rays using advanced pulse shape discrimination (PSD) and time-of-flight (TOF) algorithms on a Zynq-7000 SoC. Acquisition relied on an external CAEN digitizer (500 MS/s, 14-bit), and processing was performed offline, as direct ADC–SoC interfacing was left as future work. In [39], a digital pulse shape analysis system was implemented for real-time processing of signals from liquid scintillation detectors on a Zynq UltraScale+ MPSoC housed in a MicroTCA chassis. The system was designed to operate with a 16-bit, 1 GS/s ADC via a JESD204B interface. However, no real converter was connected; instead, a loopback between transmitter and receiver with pre-digitized signals was used. Both systems demonstrated high computing power and algorithmic flexibility but at the cost of high expense and without offering a compact board or fully autonomous real-time operation.

Taken together, the reviewed works confirm the transition toward fully digital chains although several limitations remain. Approaches retaining analog stages only partially exploit the digital paradigm, while FPGA-based systems achieve good performance and low cost but often lack flexibility due to dependence on external equipment for control and communication. SoC-FPGA platforms offer higher integration and open software environments, yet commonly lack pulse emulation, limiting accessibility. High-performance solutions provide advanced algorithms but are constrained by offline validation, high cost, and reliance on external infrastructures. This context motivates this work: a compact, autonomous, and low-cost system that integrates ADC and DAC, FPGA-based processing algorithms, and pulse emulation within an open and/or accessible software ecosystem, thus providing a comprehensive response to the challenges of accessibility, cost, and flexibility identified in this study.

III. System Overview

The proposed system consists of two independent elements implemented in programmable logic: a pulse analyzer and a pulse emulator. The analyzer processes the signal from the preamplifier and digitizes it with an ADC, while the emulator recreates the output of the preamplifier using a DAC.

In addition, the SoC includes a microprocessor running Linux-Ubuntu OS.

Fig. 2 illustrates the system's block diagram. The software uses a client–server model based on a TCP/IP architecture implemented in the Linux OS for communication with the emulator and analyzer modules of the programmable logic. Each module works with a separate client that, running locally or remotely via an Ethernet connection, sends standard commands for programmable instruments (SCPIs) through an independent graphical interface to the server hosted on the SoC's microprocessor running Linux. The server, in turn, manages communication with programmable logic.

A. Emulator Overview

The emulator is designed to generate synthetic pulses that mimic the behavior of real detector signals, allowing for system testing, calibration, and validation without the need for radioactive sources. It can simulate pulses using a Poisson distribution for arrival times and a normal distribution for pulse heights, or these parameters can be manually defined.

Its operation begins with mode selection and parameter configuration via the client's graphical interface. Once graphically configured, the client transmits the desired settings to the server using SCPI. This communication occurs either remotely via Ethernet if the client runs on a separate PC or locally if it executes on the system's microprocessor. Upon receiving the configuration, the server establishes communication with the emulator module in programmable logic via the advanced

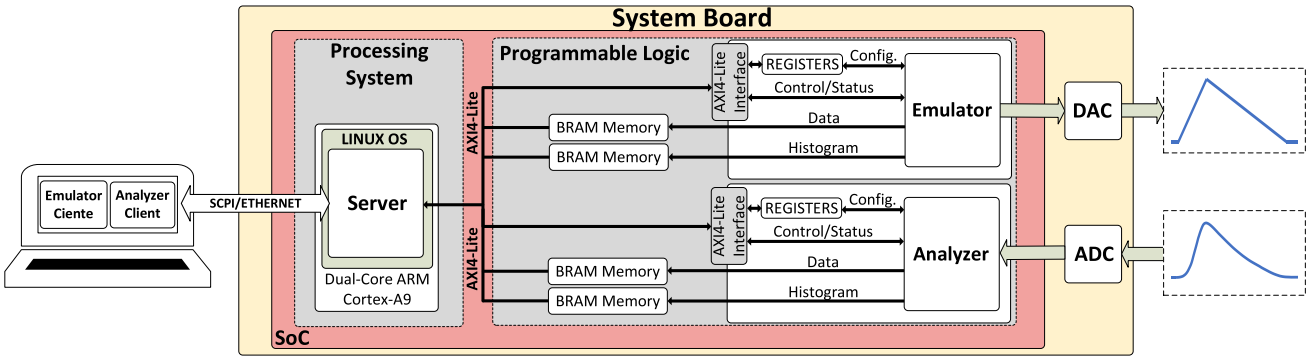


Fig. 2. Block diagram of the data acquisition system of the digital spectroscopy channel system.

extensible interface (AXI) bus. It writes the configuration registers of the module and sends commands to begin its operations.

Once the configuration is complete, the emulator sends the samples to the DAC to reconstruct signals that recreate the preamplifier output after a particle impact. Simultaneously, the generated arrival times and heights are stored in a block random access memory (BRAM), accessible from the server. In addition, the pulse height values are used to generate a histogram of the emulator pulses in the programmable logic and stored in another BRAM.

Finally, the server retrieves the samples and histogram from the BRAMs and forwards them to the client for visualization.

B. Analyzer Overview

The analyzer is responsible for capturing and processing the full signal output from the preamplifier, digitizing it via an ADC. It extracts key parameters such as pulse height and arrival time, enabling precise signal characterization. In addition, it generates a histogram of detected pulses, facilitating further analysis.

Similar to the emulator, the analyzer is configured through the client's graphical interface, where users can define its operating modes and parameters. The client sends these settings to the server via SCPI, which then communicates with the analyzer module through the AXI-Lite bus.

Once configured, the analyzer continuously captures ADC signals and processes them through the dedicated algorithms. Depending on the selected mode, it either stores the entire waveform or extracts relevant features such as pulse height and arrival time. These data are stored in the BRAM. Similarly, a histogram of the received pulses is created and stored in another BRAM.

Finally, data are read by the server and sent to the client via SCPI for visualization through its graphical interface.

IV. Implementation Details

The Red Pitaya STEMlab 125–14 board [40] is built around AMD Xilinx Zynq 7010 SoC [41], which includes a dual-core ARM Cortex-A9 processor [42] running at 667 MHz alongside programmable logic equivalent to an AMD Xilinx Artix-7 FPGA.

It includes a dual-channel ADC with 14-bit resolution, a 125 MS/s sampling rate, and selectable voltage ranges of ± 1 V or ± 20 V. It also includes a 14-bit DAC with a 125 MS/s sampling rate and a ± 1 V output range. In addition, the board features 512 MB of RAM and an SD card slot for hosting the OS. External interfaces include 1-Gbit Ethernet, USB 2.0, Wi-Fi support (via adapter), and multiple digital and analog interfaces.

This board is particularly well-suited for our application compared to other analyzed platforms (e.g., ADALM-Pluto [43], HackRF One [44], ADALM-1000 [45], NI myRIO-1900 [46], Analog Discovery 3 [47], and Ettus USRP [48]), as it integrates all the hardware required for system operation into a single compact, low-cost device, including on-board ADC and DAC, which allows the integration of the pulse emulator and analyzer within an open or accessible software ecosystem.

For this platform, several official applications are available [49], and the community has developed numerous open-source tools [50], [51]. However, none of them fully meet the specifications defined in this work (FPGA reconfigurability, use of a Linux-based OS, a TCP/IP client–server, SCPI commands, a Python-based client, and fully hardware-accelerated processing, including pulse generation, parameter extraction, and histogram computation). In this context, it was decided to develop the complete system using open-source or freely available tools, ensuring full control, as well as the reproducibility and extensibility of the proposed solution.

The implementation of the system's core components built within the programmable logic is detailed: the emulator (Section IV-A) and the analyzer (Section IV-B). Next, the implementation of the Ubuntu-Linux OS is described (Section IV-C), followed by a description of the server software (Section IV-D) and the client software developed for the application (Section IV-E).

A. Emulator Implementation

The emulator module is designed to recreate the preamplifier's output signal after the particle detector. Developed using Vivado 2024.2 software and implemented as a standalone IP module within the FPGA, it can operate either in conjunction with the analyzer for calibration and functionality testing or as a standalone input source in experiments for other spectroscopy channels. The module features an AXI4-Lite interface, enabling communication with the server running on Linux on the ARM processor.

The emulator supports two selectable operating modes:

1) **Statistical Mode**
   In nuclear decay processes, the exact moment of a decay event is unpredictable, but the number of events

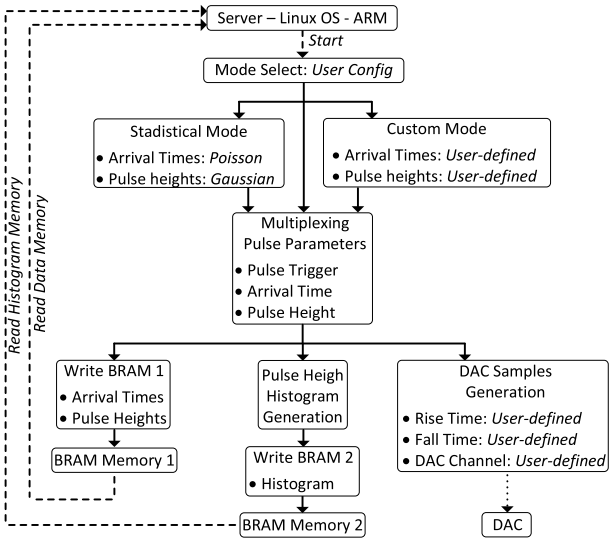


Fig. 3. Operational flow of the emulator.

per second follows a Poisson distribution. In addition, the energy of emitted particles is not always identical due to small variations, which can be approximated by a normal distribution. Therefore, in this mode, the emulator generates pulse arrivals following a Poisson distribution, with pulse heights modeled by a Gaussian distribution centered around a user-defined energy value.

2) **Custom Mode**
This mode allows the user to define multiple pulses by specifying their arrival times and pulse heights. It is particularly useful for studying pile-up effects in the preamplifier. The pulses can be generated cyclically, making it possible to simulate repeated experimental conditions.

Once the arrival times and pulse heights are defined for the selected mode, these values are written to BRAM. A histogram is generated with the resulting pulse heights and saved too to another BRAM. Finally, using the above values and the user-defined.

Fig. 3 summarizes the operational flow of the emulator. Once the emulator operation has been defined, the architecture shown in Fig. 4 is used for its implementation.

The Linux server sends configuration and control parameters to the emulator module via AXI4-Lite. This is the standard way to interact with the emulator, allowing users to set its operating parameters and issue commands remotely. The *AXI4-Lite interface* handles communication with the microprocessor and processes incoming data by identifying the target address range and distinguishing between:

- Configuration parameters, which are stored in the *Registers block* and accessible to all blocks (see Table I).
- Control commands (Start/Restart, Stop, or Reset), which are forwarded to the *Controller block* for emulator operation management.

Upon receiving the Start/Restart command, the *Controller block* activates the corresponding mode (Statistical or Custom) as defined in the *Mode register*. Simultaneously, the *Timer block* (integrated within the *Controller block*) is activated, providing the time reference for the emulator module.

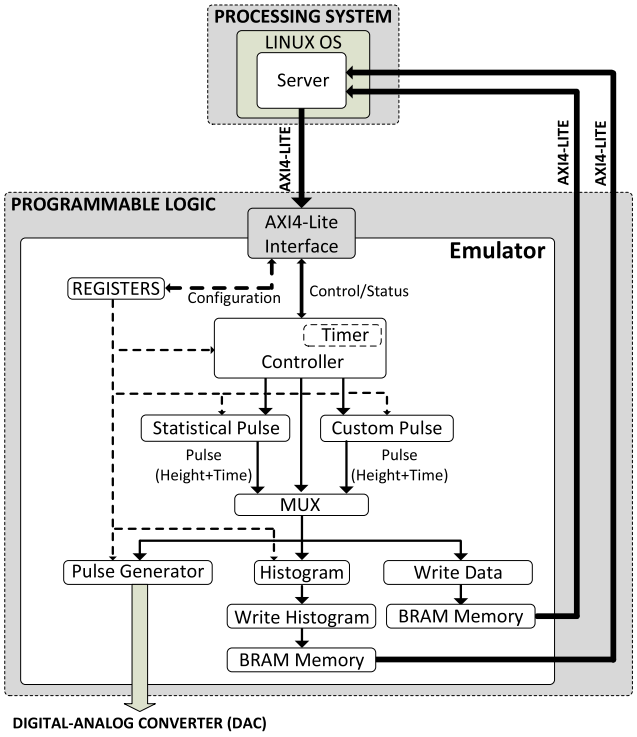


Fig. 4. Architecture of the emulator module.

TABLE I
EMULATOR CONFIGURATION REGISTERS

Command	Mode	Description
Mode	Both	Mode selection
Pulse Rate	Statistical	Defines the ratio for the Bernoulli test in Poisson pulse generation.
Seed Statistical	Statistical	Seed for pseudo-random number generation for the approximate Poisson Distribution
Seed Gaussian	Statistical	Seed for pseudo-random number generation for Gaussian Distribution (32 registers)
LFSR Width	Statistical	Number of bits in the ML-LFSR generator to control Gaussian variance
Mean Gaussian	Statistical	Gaussian Mean of Pulse Heights
Initial Delay	Custom	Delay before the first pulse
Delay between Pulses	Custom	Delay between consecutive pulses (4 registers)
Final Delay	Custom	Delay after the last pulse
Height Pulse	Custom	Height of each pulse (5 registers)
Cycle	Custom	Activation of cyclic sequence
Init Histogram	Both	Initial value for the histogram
Final Histogram	Both	Final value for the histogram
Step Histogram	Both	Width of each histogram interval
Rise Time	Both	Rise time for the generated pulse
Fall Time	Both	Fall time for the generated pulse
DAC Channel	Both	Select the DAC output channel

1) **Statistical Mode Implementation**
In this mode, the *Controller block* activates the *Statistical Pulse block*, which generates pulses at Poisson arrival times and Gaussian-distributed heights.
To determine arrival times, since the Poisson distribution is computationally expensive, an approximation method

based on the Bernoulli trial is implemented using a maximum-length linear feedback shift register (ML-LFSR) to generate pseudorandom numbers [52].

At each clock cycle, the last n bits of the ML-LFSR sequence are compared to an n-bit mask (size is configurable by the *Pulse Rate register*), where all bits are set to 1. A match indicates a successful Bernoulli trial, triggering a pulse event.

To avoid short repeating patterns and ensure accurate Poisson-distributed pulse timing, a 96-bit ML-LFSR is used. The *Seed Statistical register* configures its initial state, and the last 32 bits of the register provide a new 32-bit output every clock cycle [53].

At a clock frequency of 125 MHz, a 32-bit mask (maximum value) produces, on average, one event every 2^{32} cycles (approximately every 34 s). Reducing the mask to 16 bits allows comparing only the last part of the generated sequence, increasing the event rate to an average of every 524 ms.

Pulse heights follow a Gaussian distribution, achieved through an approximation based on the central limit theorem. The method sums the outputs of 32 independent uniform distributions, each generated by an ML-LFSR. These generators use different seeds configured in the *Seed Gaussian registers* to guarantee the independence of the generated sequences.

Each generator produces an n-bit output, where n is adjustable via the *LFSR Width register* (up to 12 bits per generator), allowing control over the final variance of the distribution. Finally, the system normalizes the sum of uniform variables by subtracting the mean and shifting it to the value set in the *Mean Gaussian register*.

2) **Custom Mode Implementation**
In the Custom mode, the *Controller block* activates the *Pulse Custom block*, which generates pulses at user-defined arrival times and pulse heights, configured via dedicated registers. These registers allow the emulation of a cyclic sequence, precisely adjusting the impact times and energy of each pulse within the cycle. The *Initial Delay register* sets the arrival time for the first pulse, while the *Delay between Pulses registers* define the intervals between subsequent pulses. The *Final Delay register* determines the delay after the last pulse before sequence repetition. Pulse heights are individually specified in the *Height Pulse registers*. Cyclic operation is optional. If disabled, pulses are generated only once. When enabled via the *Cycle register*, the sequence repeats indefinitely.

Once the arrival times and pulse heights are generated according to the selected mode, the *Controller block* manages the *Multiplexer*, which selects the appropriate data source according to the operating mode. At its output, the *Multiplexer* asserts a trigger signal and forwards the pulse height (32 bits) and pulse arrival time (40 bits) to different blocks to perform three main functions:

1) **Data Storage in BRAM**
All generated pulse values (heights and arrival times) are stored in BRAM (using the *Write Data block*), ensuring a complete record of the experiment.

2) **Histogram Generation and Storage in BRAM**
The processes pulse data to generate a histogram, providing statistical insights into the experiment. The histogram range and bin size are configurable. The resulting histogram is also stored in another BRAM (using the *Write Histogram block*).

3) **Pulse Signal Generation**
The synthesizes, generating DAC samples, a pulse signal based on the defined arrival time and height. The *Rise Time* and *Fall Time registers* control the transition dynamics. The board features two DAC output channels, with the *DAC Channel register* selecting the active output.

The BRAM memories (data and histogram) are accessible from the Server, allowing external analysis.

B. Analyzer Implementation

The analyzer module processes the output signal from the preamplifier located behind the detector. Like the emulator module, it is developed as an independent IP core on the FPGA using Vivado 2024.2 and integrates a dedicated AXI4-Lite interface for communication with a server hosted on Linux OS.

The analyzer module has two operating modes:

1) **Full Capture Mode**
The analyzer records the entire preamplifier pulse sampled by the ADC, triggered by a user-defined threshold. This mode enables detailed pulse shape analysis, offering insight into particle type and energy beyond just pulse height. By examining pulse shape, mainly rise and fall times, users can distinguish between different interactions and improve event classification. In addition, this mode allows ADC performance calibration by comparing captured signals with controlled pulses from the emulator in the Custom mode.

2) **Height Capture Mode**
The analyzer processes the ADC-sampled signal to extract pulse height and timing information. Due to the steep rising slope of the preamplifier signal and the absence of a shaper, accurately determining the trigger point for height extraction—crucial for impact energy measurement—and precise arrival time is challenging. To address this, the FPGA applies a triangular finite impulse response (FIR) filter [54] to the ADC output, generating a trigger point to capture the values of interest (pulse height and arrival time). Because this algorithm does not capture the preamplifier signal through a threshold, but rather through a triangular filter that provides a trigger when a rising edge occurs, the effect of pile-ups can be efficiently detected when multiple signals overlap. Therefore, unlike threshold-based methods, this approach enhances pile-up detection and improves measurement accuracy.

Fig. 5 shows the operational flow of the analyzer module.

In a first BRAM, different data are stored depending on the active operating mode. In Full Capture mode, the ADC samples are written directly to this memory, enabling full waveform acquisition. In Height Capture mode, the same BRAM stores the pulse arrival time and pulse height. In

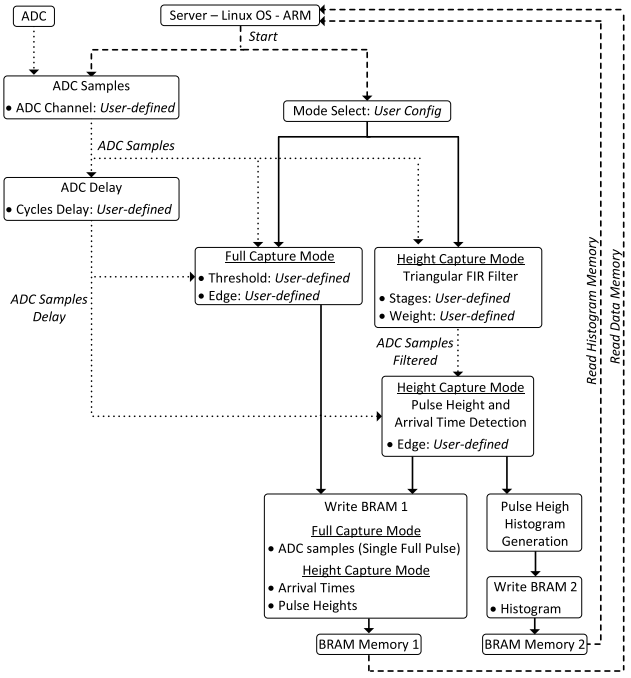


Fig. 5. Operational flow of the analyzer.

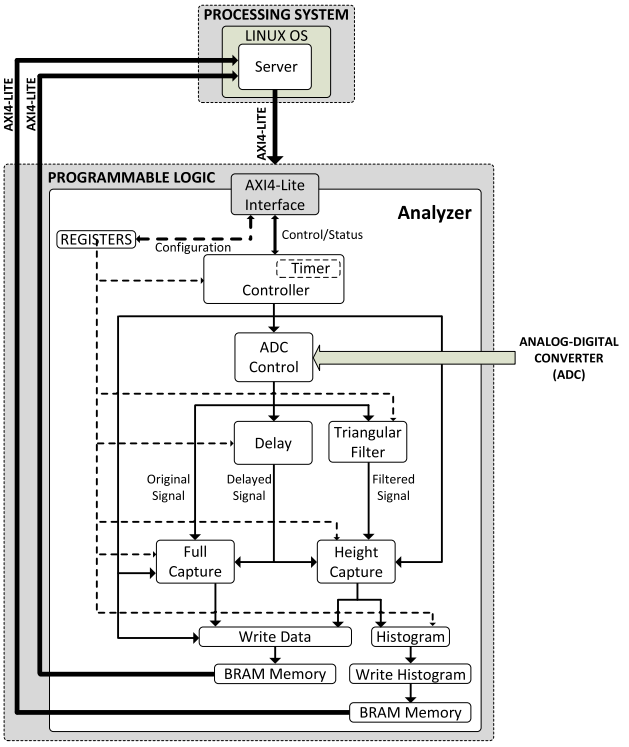


Fig. 6. Architecture of the analyzer module.

addition, when Height Capture mode is active, a second BRAM is used to store the generated pulse height histogram.

After the definition of the analyzer functionality, the architecture in Fig. 6 is adopted to implement this module.

The analyzer module operates similar to the emulator described earlier. The server running on the Linux OS sends parameters to the address range assigned to the analyzer.

The data are received through the independent *AXI4-Lite interface*, which handles communication with the microprocessor and discerns between configuration and control commands:

TABLE II
ANALYZER CONFIGURATION REGISTERS

Command	Mode	Description
Mode	Both	Mode selection
Threshold	Full	Signal capture threshold
Edge	Both	Setting the capture edge
Cycles Delay	Both	Cycles for pre-trigger capture or latency compensation
Stages	Height	Number of stages of the filter
Weight	Height	Weight of the filter FIR (16 registers)
Init Histogram	Height	Initial value for the histogram
Final Histogram	Height	Final value for the histogram
Step Histogram	Height	Histogram interval width
ADC Channel	Both	Select the ADC input channel

- Configuration parameters, enumerated in Table II, are stored in the *Register block* and are accessible from any block within the analyzer.
- Control commands (Start/Restart, Stop, or Reset) are forwarded by the *AXI4-Lite interface* directly to the *Controller block*, which manages the module.

When the Start/Restart command is received, the *ADC Control block* is activated to acquire the sampled signal. A configurable delay is applied by the *Delay block* (FIFO-based) to the samples for pre-threshold capture (Full Capture mode) or filter latency compensation (Height Capture mode). At the same time, the *Controller block* enables the selected capture mode (*Mode register*) and activates its internal *Timer block*, providing a 40-bit time reference with an 8 ns resolution for all processing stages.

1) **Full Capture Mode Implementation**
In this operating mode, the *Controller block* activates the *Full Capture block*, which records the entire waveform of the preamplifier output as sampled by the ADC. A capture is triggered when the ADC signal samples, from the *ADC Control block*, exceed a user-defined threshold set in the *Threshold register*, and it allows capturing complete pulses to analyze their shape, duration, and energy.
The *Edge register* allows users to select whether the capture should be triggered on a rising or falling edge, depending on the signal polarity. Since precise pulse acquisition requires capturing the signal before the threshold crossing, the *Full Capture block* receives delayed samples from the *Delay block*. The delay depth is configurable through the *Cycles Delay register* and ensures that once the threshold is crossed, the system retrieves the delayed samples, thus capturing the pulse from its true initial moment.
Captured samples are stored in BRAM (using the *Write Data block*) until it is full. Once memory is filled, the *Controller block* halts operation, requiring a new *Start/Restart command* to capture another pulse. Given the 14-bit ADC resolution and a 15-bit-deep BRAM with 32-bit words (storing two ADC samples per word), a pulse of up to 524 μs can be stored.
2) **Height Capture Mode Implementation**
In this, the system processes the ADC-sampled signal to determine two key parameters of each pulse:

a) **Arrival time:** The precise moment when the pulse begins.
b) **Pulse height:** The maximum amplitude of the pulse correlates with the impact energy.
The main challenge to face is due to the fact that the ADC samples the raw output of the preamplifier, which has a very fast rising edge and a significantly longer decay time. Without a shaper to smooth the signal, detecting the precise start of the rising edge is challenging, making it difficult to capture the correct pulse height and arrival time.
To optimize FPGA resources, a triangular FIR filter is used, as it requires minimal hardware [55], avoids phase distortion, and accurately detects pulse onset and peak height. This filter smooths the rising edge, improving trigger reliability. Unlike threshold-based triggering, the FIR filter detects rising edges based on the signal's shape, allowing efficient pile-up detection when multiple pulses pile-up.
To explain the Height Capture operating mode, we follow the signal processing flow through its key building blocks:

a) **Signal Delay Compensation:** Since filtering introduces processing latency, the *Delay block* (a configurable FIFO, shared by both analyzer operating modes) stores the original ADC signal (provided by the *ADC Control block*), ensuring proper alignment with the processed signal. The delay depth is set in the *Cycles Delay register* based on the filter's latency.
b) **Triangular FIR Filtering:** The *Triangular Filter block* processes the ADC samples, generating a smoothed signal that enhances edge detection. The number of filter stages (2–16) is configurable via the *Stages register*, and filter coefficients can be adjusted in the *Weight registers* (can also be used for alternative filtering approaches [56]).
c) **Edge Detection and Parameters Extraction:** The *Height Capture block* receives the filtered signal (for precise edge detection) and the delayed ADC signal (to extract accurate pulse heights and arrival times). The block implements a simple algorithm that when a filtered signal rising edge is detected, it captures the arrival time from the delayed ADC signal, and when a filtered signal falling edge is detected, it captures the pulse height from the delayed ADC signal. For signals with different polarities, the capture edges can be inverted using the *Edge register*.
d) **Data Storage and Histogram:** The arrival times and pulse heights are stored in a first BRAM (using the *Write Data block*) for further analysis. This BRAM is shared with the Full Capture mode and operates according to the selected mode. In addition, the *Histogram block* processes the captured heights to generate a statistical distribution. The histogram's initial, final, and step values are configurable via registers, and the histogram is stored in a second BRAM (via the *Write Histogram block*).

C. Operating System

For this project, a Linux OS based on Ubuntu Core 20.04.5 LTS [57] was integrated on the AMD Xilinx Zynq 7010 SoC, allowing both standalone operation—configuring and managing the device directly from the SoC—and remote control via SSH or serial port. This eliminates the need for an external computer while enabling flexible access.

Integration involved the bootloader, device tree blob (DTB), and kernel, generated using PetaLinux 2024.2 [58], importing the output from Vivado 2024.2 of the logic design (hardware configuration and bitstream). Instead of the default PetaLinux rootfs, we chose Ubuntu Core, a lightweight distribution optimized for IoT and embedded ARM-based systems, providing a package manager, development tools, and remote access capabilities.

During boot, the FPGA is configured with the developed hardware logic. Once the board is booted, the OS enables both on-device and remote configuration, management, and development, streamlining system operation.

D. Server

A TCP/IP server was implemented on the SoC's ARM Cortex-A9 to enable independent remote configuration and control of the emulator and analyzer modules. The server, coded in C, uses a reduced set of SCPI instructions over the board's Ethernet interface.

The server supports multiple client connections (via TCP port 5000), either emulator or analyzer clients, and manages the corresponding modules implemented in the SoC's programmable logic. A multi-threaded architecture handles client requests independently so that clients can send SCPI instructions to control the emulator and analyzer, which operate independently.

Multiple clients of the same type (emulator or analyzer) can read or modify the configuration and state of the respective module, with real-time notifications sent to all connected clients upon any state change. The server directly configures module registers and accesses the BRAM memories through the AXI protocol, accessing the system memory map (nmap) in the address range assigned to each module or memory.

This client–server approach allows remote control of the board, eliminating the need for a local keyboard or monitor while minimizing system overhead.

E. Client

A graphical user interface (GUI) has been developed for each module to provide an intuitive way to independently control the analyzer and emulator modules, supporting both local and remote operation:

- *Local Use:* The GUI runs directly on the Linux OS deployed on the SoC's microcontroller, allowing users to interact with the system by connecting a keyboard and monitor to the system board.
- *Remote Use:* The GUI can be executed on an external computer connected via Ethernet.

Each interface communicates with the system by sending SCPI commands over TCP/IP to the server explained in Section

IV-D, allowing users to configure and operate the FPGA-based emulator and analyzer modules efficiently.

Python was chosen for its flexibility and cross-platform support, using PyQt and PyQtGraph for the interface. Since the emulator and analyzer operate independently, each has a dedicated interface:

1) **Emulator Interface**
   The emulator interface includes control buttons, mode selection, and a graph displaying the generated histogram. Three sliders adjust rise time, fall time, and zoom, while a side panel shows system-generated data. Additional settings are available under the Setup tab.
2) **Analyzer Interface**
   The analyzer interface displays the captured signal (Full Capture mode) or a histogram (Height Capture mode). Two sliders adjust zoom and threshold (used only in Full Capture mode), with other settings in the Settings tab.

For both interfaces, once communication is established via the IP address in the Setup menu, module parameters are automatically retrieved and updated in real time. Any parameter change is broadcast to all connected clients, ensuring synchronized operation across multiple locations.

V. Results

A structured experimental methodology was adopted to characterize, validate, and evaluate the designed modules under real conditions.

Initially, the emulator and analyzer modules were tested independently using laboratory equipment. Then, they were evaluated together in a loopback configuration.

Finally, measurements were carried out at the Nuclear Physics Laboratory of the University of Huelva (Spain) under real radiation experimental conditions. The designed system was directly connected to the output of a preamplifier coupled to a silicon detector exposed to a ^{241}Am radiation source. The results were compared with those obtained using a high-precision laboratory oscilloscope and a conventional analog spectroscopy chain.

A. Emulator Testing

The emulator module was first tested by connecting its DAC output to a Keysight MSO9404A oscilloscope to visualize the generated signals in different modes:

1) **Statistical Mode**
   The event generation rate (from 32 ns to 34 s, doubling the event rate at each step), mean (from 0 V to 1 V in 122 μV steps), and variance (from 0 mV to 650 mV, doubling with each step) of pulse height were varied to assess signal generation flexibility.
2) **Custom Mode**
   Pulses with adjustable parameters, such as number of pulses (between 1 and 5), amplitude (from 0 V to 1 V in 122 μV steps), rise and fall times (both individually configurable from 8 ns to 32 768 ns, doubling in each step), and delay between pulses (from 0 ms to 524 ms in 8 ns steps), include pulse pile-up scenarios and tests involving signal inversion (negative pulses).

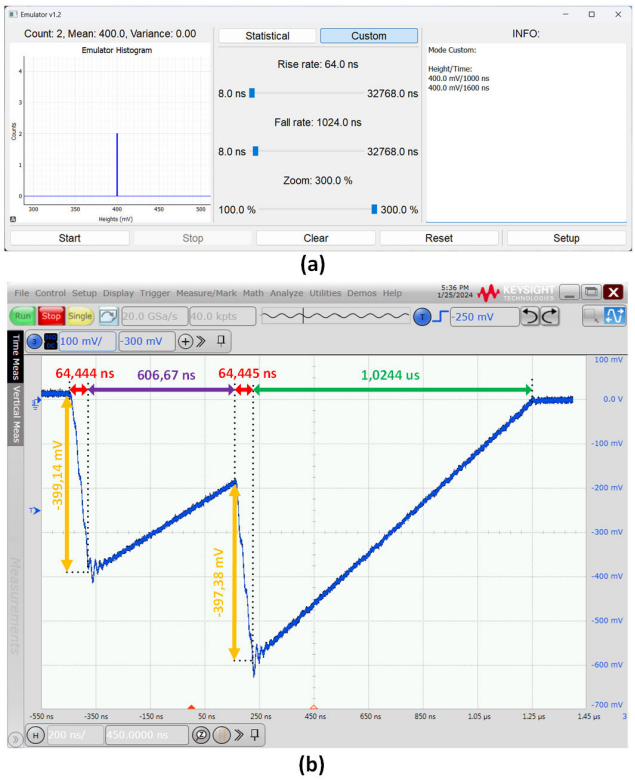


Fig. 7. (a) Emulator client interface set to generate pile-up and (b) DAC output with pile-up observed on the oscilloscope.

As an example, to illustrate the emulator testing, Fig. 7(a) shows the emulator client GUI configured to generate two overlapping pulses (pile-up) with active signal inversion (negative pulses), each with an amplitude of 400 mV, a rise time of 64 ns, a fall time of 1024 ns, and a 600 ns delay between pulses. Fig. 7(b) shows the output waveform of the corresponding DAC, as captured by the oscilloscope. The rise and fall times closely match the configured values, with errors below 1%, and the pulse amplitudes deviate by less than 0.7%. These small discrepancies may be attributed to impedance mismatches and connection losses and could be calibrated within the system.

B. Analyzer Testing

The analyzer module was validated using a Tektronix AFG31252 waveform generator connected to the system's ADC. Pulses with different amplitudes and timing parameters were captured in two available modes:

1) **Full Capture Mode**
   It records the complete signal waveform from a configured threshold (adjustable from -1 V to 1 V in 122 μV steps) and supports configuration for rising or falling edge capture.
2) **Height Capture Mode**
   It extracts only pulse heights and arrival times. The triangular FIR filter options were varied by modifying the number of stages (2–16) with each stage having a configurable weight in a 5-bit individual register.

As an example, Fig. 8 displays the analyzer client GUI capturing a signal in Full Capture mode with a -200 mV

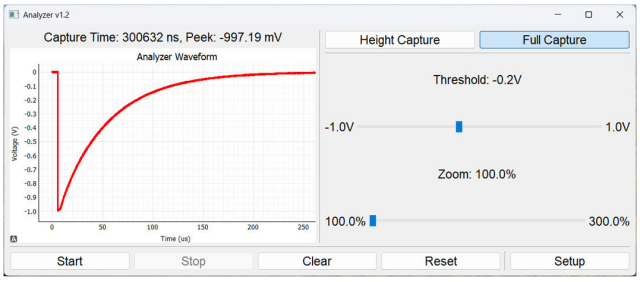


Fig. 8. Analyzer client interface in Full Capture mode to capture a signal generated with the waveform generator.

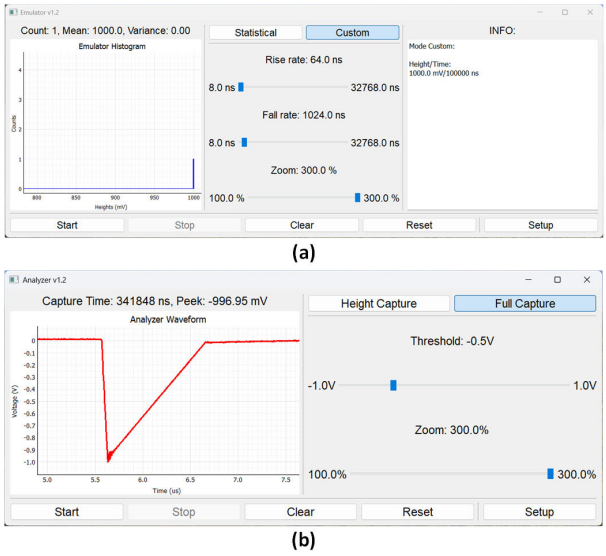


Fig. 9. (a) Emulator and (b) analyzer interfaces for loopback testing (Custom mode and Full Capture mode, respectively).

threshold and falling-edge triggering. The signal, generated by the waveform generator, was configured with an amplitude of 1000 mV (negative pulse), a rise time of 64 ns, and a fall time of 256 μs. The captured waveform is visible on the GUI. The measured rise and fall times exhibit relative errors below 1% and an extracted pulse amplitude of 997.19 mV (corresponding to a relative deviation below 0.3%), confirming the correct operation of the analyzer module.

C. Loopback Testing

To verify system functionality, the emulator was looped with the analyzer, and two types of experiments were conducted:

1) **Single-Pulse Capture**
   Fig. 9 shows both client interfaces: the emulator [see Fig. 9(a)], configured in the Custom mode to generate a 1000 mV (negative pulse) pulse with a 64 ns rise time and 1024 ns fall time, and the analyzer [see Fig. 9(b)], set to Full Capture mode with a −500 mV threshold and falling-edge triggering. The pulse was correctly captured, with only a slight gain loss due to impedance mismatches and connection losses, corresponding to a relative amplitude error below 0.31% (996.95 mV measured), which could be calibrated within the system.
2) **Statistical and Height Mode**
   As the final step in the laboratory tests, Fig. 10 shows the interfaces of the emulator configured in the Statistical mode [see Fig. 10(a)], connected in a loop to the analyzer configured in the Height Capture mode [see Fig. 10(b)].
   In the emulator, the pulse rate is set to 25 (a pulse generated every 269 ms), with a mean value of 500 mV (negative pulses), a 12-bit LFSR (corresponding to an approximate variance of 650 mV), a rise time of 64 ns, and a fall time of 1024 ns.
   In the analyzer, the capture is configured in Height Capture mode, using a 16-stage FIR filter with weights for a Triangular FIR Filter configuration. In this mode, the analyzer threshold is not used.
   As a result, the analyzer captures all 2200 pulses generated by the emulator. The measured mean pulse amplitude is 498.22 mV, with a relative error of below 0.4%, while the measured variance is 647.51 $(\text{mV})^2$, differing from the generated value by about 0.1%.

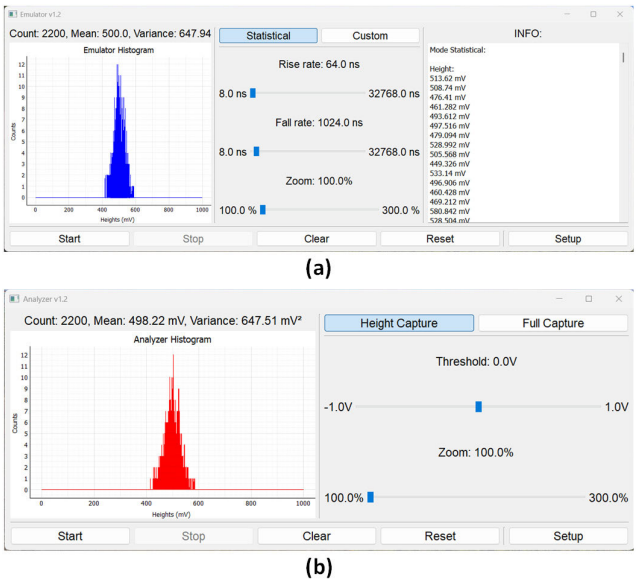


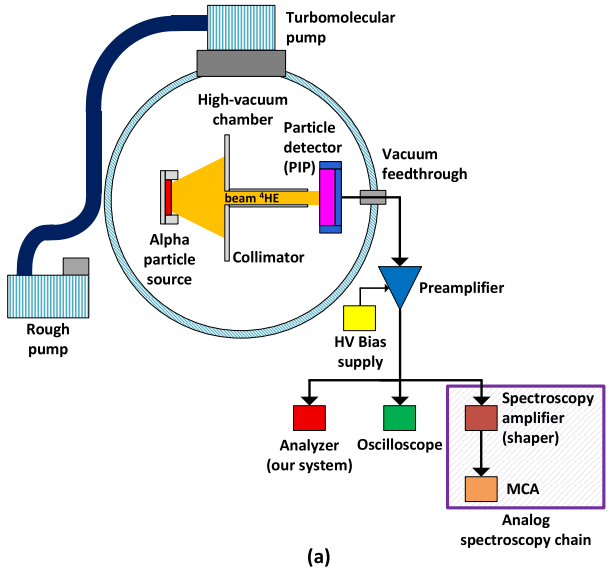
Fig. 10. (a) Emulator in Statistical mode and (b) analyzer in height capture mode during loopback testing.

This result demonstrates the correct operation of the system in a loopback configuration, enabling system self-calibration.

D. Testing Under Real Conditions

After the electronic characterization of the system, its performance was evaluated under real irradiation conditions at the Nuclear Physics Laboratory of the University of Huelva (Spain). The experimental setup is shown in Fig. 11. A collimated ^{241}Am radioactive source was used to provide a beam of alpha particles ^{4}He with a kinetic energy of 5.5 MeV. The ^{4}He ions were implanted into a fully depleted passivated implanted planar silicon (PIPS) detector with a thickness of 500 μm and an active area of 300 mm^2 (Canberra FD 300-17-500 RM).

The source–detector assembly was installed in a high-vacuum chamber and pumped down to 10^{-5} mb for the ^{4}He beam to reach the detector. The vacuum system consisted of a turbopump Leybold TURBOVAC mod. 350iX and a rough-pump Leybold SCROLLVAC SC5D. The detector was



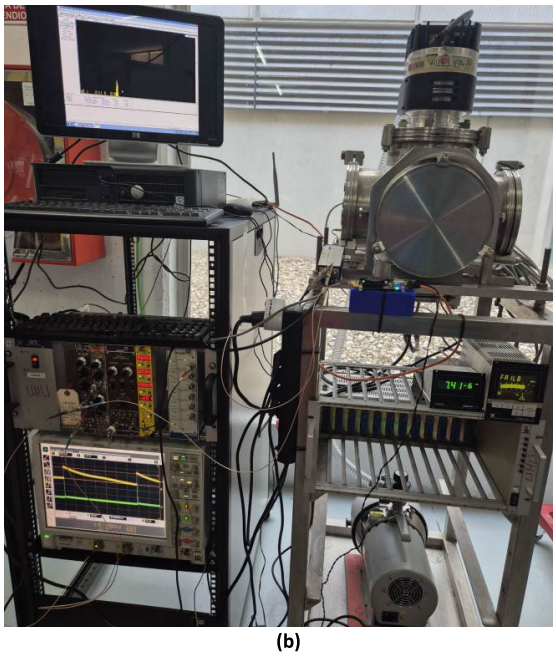


Fig. 11. (a) Sketch of the experimental setup and (b) a photograph of the setup for testing under real conditions.

depleted at the nominal bias of 23 V, obtaining 62 nA reversed-leakage current.

The charge signals generated in the detector by the interaction of alpha particles were extracted from the vacuum chamber through a LEMO high-vacuum feedthrough and routed to a Canberra Model 2003BT silicon detector preamplifier.

To evaluate the system under real experimental conditions, two complementary comparisons were performed. First, waveforms acquired by the analyzer in Full Capture mode were compared with those recorded by a Keysight MSO9404A oscilloscope to assess amplitude accuracy, waveform morphology, and temporal characteristics of signals.

Second, histograms generated by the analyzer in Height Capture mode were compared with those from a conventional spectroscopy chain comprising an ORTEC Model 571 spectroscopy amplifier and a CANBERRA Multiport II MCA,

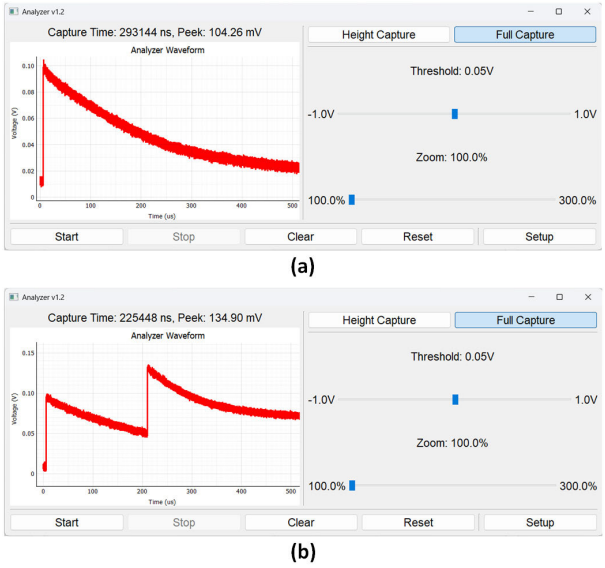


Fig. 12. Analyzer in Full Capture mode with (a) captured pulse and (b) pile-up for testing under real conditions.

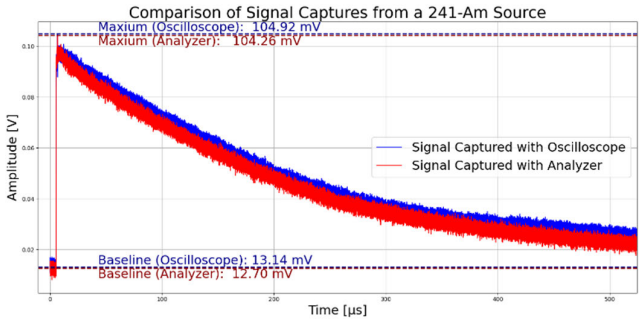


Fig. 13. Comparison of the waveforms recorded by the oscilloscope and analyzer (Full Capture mode).

both powered by a Wiener UEP 15 supply. This comparison benchmarks the digital analyzer against a widely used commercial system.

In a first step, the developed digital analyzer was configured in Full Capture mode, using rising-edge triggering with a trigger level of 50 mV. In parallel, the Keysight MSO9404A oscilloscope was set to acquire the same signals, using the same trigger level, with a vertical scale of 20 mV/div, a time base of 200 μs/div, and a sampling rate of 5 GSa/s.

Under these conditions, pulses were captured and analyzed to evaluate the system response to single events [see Fig. 12(a)], showing a pulse height of 104.26 mV with a rise time of 184 ns and a fall time of approximately 800 μs, and to pulse pile-up [see Fig. 12(b)], where the second peak reaches 134.90 mV about 200 μs after the maximum of the first pulse.

Fig. 13 compares waveforms recorded by the oscilloscope and the digital acquisition system under identical irradiation conditions. The signals show excellent agreement, with deviations below 1 mV in amplitude and baseline, confirming accurate capture of the preamplifier output and preservation of signal integrity, maintaining rise/fall times.

Subsequently, the system was configured to evaluate pulse-height acquisition and histogram generation using a conventional analog spectroscopy chain as reference. The

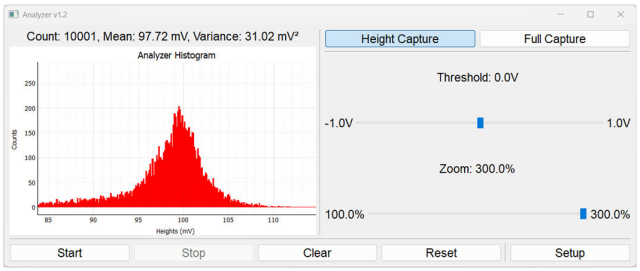


Fig. 14. Analyzer in height capture mode for testing under real conditions.

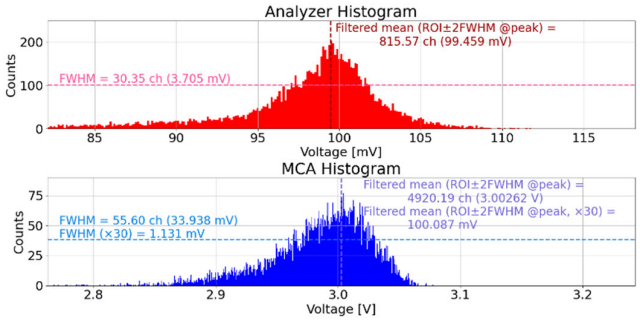


Fig. 15. Comparison of the histograms recorded by the analyzer and the commercial analog spectroscopy chain.

ORTEC 571 spectroscopy amplifier was operated with a gain of 30, while the CANBERRA Multiport II MCA was configured with 16 384 channels over a 0–10 V range and a preset of 10 000 counts. In parallel, the digital acquisition system was set to Height Capture mode using a 16-stage triangular FIR filter.

The digital analyzer acquired the pulse amplitudes generated by the preamplifier and built the corresponding histogram, as shown in Fig. 14, obtaining a mean of 97.72 mV and a variance of 31.02 $(\text{mV})^2$ for 10001 captured pulses.

Finally, Fig. 15 compares the pulse-height histograms obtained with the designed analyzer and the commercial analog spectroscopy chain after compensating for the amplification factor ($\times 30$) of the ORTEC 571 amplifier. The mean pulse amplitude measured by the digital analyzer differs by less than 0.7% from that obtained with the MCA, demonstrating very good agreement between both systems.

VI. Discussion

Regarding the FPGA implementation, the design makes intensive use of logic and memory resources, employing 16 889 LUTs (95.96%) and 56 BRAM blocks (93.33%) to implement the analyzer and emulator modules. Flip-flop utilization is moderate (46.06%), I/O resources reach 65%, one PLL is used, and LUTRAM usage remains below 1%. Timing analysis confirms that all timing constraints are met at the target clock frequency of 125 MHz, with an estimated on-chip power consumption of approximately 1.65 W.

The results obtained under real irradiation conditions validate the correct operation of the proposed system. The close agreement with oscilloscope measurements confirms accurate waveform acquisition, while the histogram results show very good agreement with those from a conventional spectroscopy chain. The proposed analyzer, therefore, provides a good approximation to commercial systems while relying on a low-cost, flexible, and accessible digital platform with reduced infrastructure requirements.

To place these results in context, the proposed system is next compared with the works reviewed in the state of the art using a common set of criteria directly linked to the three main limitations identified in the Introduction: accessibility (pulse-emulation, experimental validation, and comparison with commercial equipment), cost (compact standalone integration and on-board ADCs), and flexibility (use of SoC-FPGAs, communication interfaces, implemented algorithms, pile-up handling, and software ecosystem).

In addition, complementary aspects such as ADC resolution and the paradigm shift from analog to digital processing chains provide further benchmarks to assess overall system performance and technological innovation. The purpose of this framework is to ensure that all systems are evaluated under the same perspective, enabling a rigorous and systematic comparison that positions our proposal within this context and highlights its advantages, limitations, and innovations.

The outcome of this analysis is summarized in Table III, which presents the characteristics associated with the defined criteria across the reviewed works, including our own system.

Our system, compared to [30] and [31], completely eliminates the analog shaping stages, thereby avoiding the inherent restrictions of the classical acquisition chain. In comparison with [32], beyond removing analog prefilters, it integrates a faster ADC and Ethernet communication, overcoming the limitations of the legacy RS-232 interface. With respect to [33], it does not require an external PXIe chassis, which significantly reduces cost and improves portability by integrating all resources into a single compact and autonomous board. Compared to [34] and [35], which rely on stand-alone FPGAs dependent on external PCs for management, our design integrates an SoC with embedded Linux and a full client–server architecture, enhancing flexibility and removing that dependence. In relation to [36] and [37], although implemented on SoC-FPGAs, they lack pulse-emulation capabilities; our system incorporates this key functionality to improve accessibility, enabling validation without radioactive sources and outside specialized laboratories. In comparison with [38], which depends on external digitizers and offline processing, our design ensures real-time acquisition and analysis on fully integrated hardware, reducing both cost and complexity. Finally, compared to [39], which validates only with emulated signals (pre-digitized samples) and within an expensive chassis, our system combines validation with real detectors and internal emulation, guaranteeing accessibility across different environments, low infrastructure cost, and flexibility for future expansion and adaptation.

In view of the results obtained, the digital spectroscopy channel developed has a wide range of potential applications, including the generation of multichannel systems taking advantage of the new systems based on RF SoC FPGA. This will have a major impact on the emulation and/or characterization of high-resolution particle detector arrays for studies of nuclear reactions with radioactive ions, such as the GLObal ReactIon Array (GLORIA) [59].

TABLE III
COMPARATIVE SUMMARY OF OUR SYSTEM AND STATE-OF-THE-ART IMPLEMENTATIONS

	Processing chain / SoC-FPGA	Integration (Compact / On-board ADC)	ADC (Resolution, Rate)	Communication / Software	Algorithms	Pile-up / Emulation	Validation / Commercial comparison
Our Work	Digital / SoC Zynq-7010	Yes / Yes	14-bit, 125 MS/s	Ethernet 1 Gbps / C-Python	Trapezoidal –triangular	Yes/Yes	Yes/Yes
[30]	Analog Shaper / SoC Zynq-7010	Yes / Yes	12-bit, 500 kS/s	USB–Wi-Fi / LabVIEW	Amplitude threshold	No / No	Yes / Yes
[31]	Analog Shaper / SoC Zynq-7010	Yes / Yes	12-bit, 500 kS/s	USB–Wi-Fi / LabVIEW	Amplitude threshold	No / No	Yes / Yes
[32]	Digital (with pre-filter) / FPGA Virtex V	Yes / –	14-bit, 62 MS/s	RS-232 / –	Trapezoidal	Yes / No	Yes / Yes
[33]	Digital / FPGA Kintex-7	No / No	14-bit, 250 MS/s	PXIe / LabVIEW	Trapezoidal	Yes / No	Yes / Yes
[34]	Digital / FPGA Intel MAX 10	Yes / Yes	14-bit, 150 MS/s (operated 100 MHz)	USB / C#	Trapezoidal	Yes / No	Yes / Yes
[35]	Digital / FPGA Spartan-6	Yes / Yes	14-bit, 125 MS/s	USB / –	Trapezoidal	Yes / No	Yes / Yes (specifications)
[36]	Digital / SoC Zynq-7020	Yes / Yes	16-bit, 1 MS/s	– / –	Trapezoidal	Yes / No	Yes / No
[37]	Digital / SoC Zynq-7020	– / –	12-bit, 65 MS/s	– / –	Cusp–Flattop	Yes / No	Yes / No
[38]	Digital / SoC Zynq-7000 Family	– / No	14-bit, 500 MS/s (external, offline)	USB–UART / C	Amplitude threshold	Yes / No	No / Yes (offline)
[39]	– / MPSoC Zynq UltraScale+	No / No	16-bit, 1 GS/s (emulated)	MicroTCA / C++	FIR + CFD	Yes / Yes (samples only)	No / No

VII. CONCLUSION

This article presents a versatile and reconfigurable digital spectroscopy system that integrates emulation and analysis modules into a compact SoC FPGA platform. It covers both the hardware design implemented in the FPGA and the necessary software stack, including Linux OS configuration, server implementation, and client interfaces, making the system a fully spectrometry channel data acquisition system.

By incorporating a fully digital signal processing chain, the system eliminates analog components such as the shaper, enabling direct interfacing between the analyzer and the preamplifier. This approach reduces complexity and cost while enhancing signal integrity and adaptability.

One key advantage of the system is its emulation capability, which allows for precise pulse generation with configurable parameters. This feature facilitates realistic signal testing, including pile-up, without the need for radioactive sources, making it an effective tool for calibration and algorithm validation. In addition, the analyzer module leverages the FPGA's logic fabric to perform real-time, hardware-accelerated processing while remaining reconfigurable to accommodate different experimental conditions.

The client–server architecture enhances usability by enabling both local and remote operation, allowing users to configure and control the system via intuitive graphical interfaces.

Experimental validation was conducted first in a controlled laboratory environment, testing the emulator and analyzer both independently and in a combined setup. The system was then evaluated under real conditions at the Nuclear Physics Laboratory of the University of Huelva (Spain), where it was connected to a radiation detector coupled to a preamplifier. The close agreement with oscilloscope measurements confirms accurate waveform acquisition, while histograms show very good agreement with those obtained using a conventional commercial spectroscopy chain, confirming performance comparable to high-cost commercial and demonstrating the system's applicability in practical scenarios.

Furthermore, the proposed system demonstrates competitive performance compared to other state-of-the-art digital systems reviewed in the literature.

Overall, the proposed system addresses the main limitations identified in the problem statement (restricted accessibility, high cost, and limited flexibility) and offers a low-cost, compact, and adaptable digital alternative to commercial spectroscopy solutions. The system is suitable for nuclear engineering, particle accelerators, and medical imaging.

ACKNOWLEDGMENT

This research was developed under grants PID2021-127711NBI00, PID2024-159209NB-C21 and CPP2022-009904 funded by MCIN/AEI/10.13039/5011000 11033 and, as appropriate, by the "European Union NextGenerationEU/PRTR"; and under the program "Contrato Predoctoral PIF del VI Plan Propio de Investigación y Transferencia, Convocatoria 2020" supported by Universidad de Sevilla (Spain). Funding for open access charge: Universidad de Huelva / CBUA.

REFERENCES

[1] G. F. Knoll, *Radiation Detection and Measurement*, 4th ed., Hoboken, NJ, USA: Wiley, 2010.

[2] P. Andreo, D. T. Burns, A. E. Nahum, J. Seuntjens, and F. H. Attix, *Fundamentals Ionizing Radiat. Dosimetry.* Hoboken, NJ, USA: Wiley, 2017.

[3] N. Tsoulfanidis and S. Landsberger, *Measurement and Detection of Radiation*. Boca Raton, FL, USA: CRC Press, 2021.

[4] R. He et al., "Advances in nuclear detection and readout techniques," *Nucl. Sci. Techn.*, vol. 34, no. 12, p. 205, Dec. 2023, doi: 10.1007/s41365-023-01359-0.

[5] D. McGregor and J. K. Shultis, *Radiation Detection: Concepts, Methods, and Devices*. Boca Raton, FL, USA: CRC Press, 2020.

[6] K. Siegbahn, *Alpha-, Beta- and Gamma-Ray Spectroscopy*. Amsterdam, The Netherlands: North Holland, 2012.

[7] A. Belyaev and D. Ross, "The basics of nuclear and particle physics," in *Undergraduate Texts in Physics*. Cham, Switzerland: Springer, 2021.

[8] R. Grzywacz, "Applications of digital pulse processing in nuclear spectroscopy," *Nucl. Instrum. Methods Phys. Res. Sect. B, Beam Interact. Mater.*, vol. 204, pp. 649–659, May 2003, doi: 10.1016/s0168-583x(02)02146-8.

[9] P. S. Lee, C. S. Lee, and J. H. Lee, "Development of FPGA-based digital signal processing system for radiation spectroscopy," *Radiat. Meas.*, vol. 48, pp. 12–17, Jan. 2013, doi: 10.1016/j.radmeas.2012.11.018.

[10] P. Gandini, "Spectroscopy at LHCb," *Moscow Univ. Phys. Bull.*, vol. 77, no. 2, pp. 199–200, Apr. 2022, doi: 10.3103/s0027134922020357.

[11] S. Chatrchyan et al., "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC," *Phys. Lett. B*, vol. 716, no. 1, pp. 30–61, Sep. 2012, doi: 10.1016/j.physletb.2012.08.021.

[12] M. P. M. Letertre, P. Giraudeau, and P. de Tullio, "Nuclear magnetic resonance spectroscopy in clinical metabolomics and personalized medicine: Current challenges and perspectives," *Frontiers Mol. Biosciences*, vol. 8, Sep. 2021, Art. no. 698337, doi: 10.3389/fmolb.2021.698337.

[13] R. Diehl, "Cosmic gamma-ray spectroscopy," *Astronomical Rev.*, vol. 9, no. 3, pp. 1–54, Jul. 2014, doi: 10.1080/21672857.2014.11519736.

[14] O. V. Pakari et al., "Gamma-ray spectroscopy in low-power nuclear research reactors," *J. Nucl. Eng.*, vol. 5, no. 1, pp. 26–43, Jan. 2024, doi: 10.3390/jne5010003.

[15] R. T. Kouzes, E. R. Siciliano, J. H. Ely, P. E. Keller, and R. J. McConn, "Passive neutron detection for interdiction of nuclear material at borders," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 584, nos. 2–3, pp. 383–400, Jan. 2008, doi: 10.1016/j.nima.2007.10.026.

[16] L. F. García del Moral, A. Morgado, and J. A. Esquivel, "Reflectance spectroscopy in combination with cluster analysis as tools for identifying the provenance of neolithic flint artefacts," *J. Archaeological Sci., Rep.*, vol. 37, Jun. 2021, Art. no. 103041, doi: 10.1016/j.jasrep.2021.103041.

[17] A. E. M. Khater and H. A. Al-Sewaidan, "Radiation exposure due to agricultural uses of phosphate fertilizers," *Radiat. Meas.*, vol. 43, no. 8, pp. 1402–1407, Sep. 2008, doi: 10.1016/j.radmeas.2008.04.084.

[18] M. W. Akram et al., "Defect detection and degradation analysis in photovoltaic modules using thermography, spectroscopy, and current-voltage measurements, and quantitative assessment of their impact," *Energy Technol.*, vol. 8, no. 7, Jul. 2020, Art. no. 2000100, doi: 10.1002/ente.202000100.

[19] J. Klusoň, "In-situ gamma spectrometry in environmental monitoring," *Appl. Radiat. Isot.*, vol. 68, nos. 4–5, pp. 529–535, Apr. 2010, doi: 10.1016/j.apradiso.2009.11.041.

[20] A. Jhingan, E. T. Subramaniam, and S. Venkataramanan, "Detectors and front-end electronics for nuclear physics research at IUAC," *Eur. Phys. J. A*, vol. 61, no. 3, p. 64, Mar. 2025, doi: 10.1140/epja/s10050-025-01526-4.

[21] C. Martin, K. Cockerline, and S. Landsberger, "The appropriate environmental sample to educate novice students in environmental radioactivity measurements using gamma ray spectroscopy," *J. Environ. Radioactivity*, vol. 290, Nov. 2025, Art. no. 107815, doi: 10.1016/j.jenvrad.2025.107815.

[22] A. T. Susanto, P. Prajitno, S. K. Wijaya, S. Soekirno, and I. P. Susila, "A systematic literature reviews of multichannel analyzer based on FPGA for gamma spectroscopy," *J. Phys. Conf. Ser.*, vol. 1528, p. 12016, Apr. 2020, doi: 10.1088/1742-6596/1528/1/012016.

[23] C. S. A. Kumar, P. Hemanth, B. Ganesh, T. Manikya, S. Kanthi, and B. S. Likith, "A survey of FPGA-based techniques for high-performance gamma radiation detection," in *Proc. 4th Int. Conf. Data Eng. Commun. Syst. (ICDECS)*, Bengaluru, India, Mar. 2024, pp. 1–6.

[24] Z. Tang and Z. Liu, "Review of research on nuclear signal pulse shaping," *J. Theory Pract. Eng. Sci.*, vol. 4, no. 4, pp. 17–19, Apr. 2024, doi: 10.53469/jtpes.2024.04(04).03.

[25] H.-P. Pham, V.-H. Le, T.-D. Trang, and N.-D. Nguyen, "Count loss evaluation for accuracy enhancement of a FPGA-based gamma spectroscopy," *Sci. Technol. Nucl. Installations*, vol. 2024, no. 1, Jan. 2024, Art. no. 7076108, doi: 10.1155/2024/7076108.

[26] J. Jiménez-Sánchez, P. Blanco-Carmona, J. M. Hinojo-Montero, F. R. Palomo, R. L. Millán, and F. Muñoz-Chavero, "Advanced system-on-chip field-programmable-gate-array-powered data acquisition system for pixel detectors," *Sensors*, vol. 24, no. 1, p. 218, Dec. 2023, doi: 10.3390/s24010218.

[27] K. Guerrero-Morejón, J. M. Hinojo-Montero, J. Jiménez-Sánchez, C. Rocha-Jácome, R. González-Carvajal, and F. Muñoz-Chavero, "Efficient real-time isotope identification on SoC FPGA," *Sensors*, vol. 25, no. 12, p. 3758, Jun. 2025, doi: 10.3390/s25123758.

[28] International Atomic Energy Agency. (Oct. 2010). *Instrumentation for Digital Nuclear Spectroscopy*. [Online]. Available: https://www-pub.iaea.org/MTCD/Publications/PDF/te_1706_web.pdf

[29] International Atomic Energy Agency. (Nov. 2007). *Signal Processing and Electronics for Nuclear Spectrometry*. [Online]. Available: https://www-pub.iaea.org/MTCD/publications/PDF/te_1634_web.pdf

[30] V. H. Hai, "Development of an FPGA-based pulse-height analyzer for scintillation detectors in gamma energy spectroscopy," *IEEJ Trans. Electr. Electron. Eng.*, vol. 19, no. 10, pp. 1737–1742, Oct. 2024, doi: 10.1002/tee.24131.

[31] V. H. Hai, N. T. T. Phuc, H. T. K. Trang, T. T. H. Loan, and P. L. H. Sang, "Development of FPGA-based nuclear electronics using NI MyRIO hardware for small-scale radiation detector systems," *IEEE Trans. Nucl. Sci.*, vol. 72, no. 3, pp. 380–384, Mar. 2025, doi: 10.1109/TNS.2024.3481408.

[32] H. Choi and J.-S. Chai, "FPGA-based trapezoidal digital pulse shaping in nuclear spectrometry," *J. Korean Phys. Soc.*, vol. 82, no. 3, pp. 236–243, Feb. 2023, doi: 10.1007/s40042-022-00661-1.

[33] S. Saxena and A. I. Hawari, "Investigation of FPGA-based real-time adaptive digital pulse shaping for high-count-rate applications," *IEEE Trans. Nucl. Sci.*, vol. 64, no. 7, pp. 1733–1738, Jul. 2017, doi: 10.1109/TNS.2017.2692219.

[34] V. H. Cao et al., "Flexible and low-cost FPGA-based multichannel analyzer for handheld measurement devices," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 1018, Dec. 2021, Art. no. 165808, doi: 10.1016/j.nima.2021.165808.

[35] J. Zhang, X. Qin, M. Zhao, and X. Chen, "FPGA implementation of a real-time digital pulse processing analysis for radiation detectors," *Appl. Radiat. Isot.*, vol. 176, Oct. 2021, Art. no. 109900, doi: 10.1016/j.apradiso.2021.109900.

[36] F. A. Castaño, L. F. Ramírez, S. Montoya, D. Estrada, and J. H. López, "Development of an open-source nuclear event discrimination pipeline using FPGA-based systems: Comprehensive approach with data acquisition, noise filtering, trapezoidal shaping, and benchmarking with ionization and scintillation detection technology," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 1077, Aug. 2025, Art. no. 170546, doi: 10.1016/j.nima.2025.170546.

[37] X. Han, W. Yang, and F. Li, "FPGA implementation and measurement of cusp-flattop hybrid filter shaping method for nuclear instruments," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 1065, Aug. 2024, Art. no. 169572, doi: 10.1016/j.nima.2024.169572.

[38] B. Wellons, R. S. Kumaran, S. Lee, and S. Prasad, "Development of a signal processing software for scintillation detectors and implementation on an FPGA for fast sensing," *Nucl. Technol.*, vol. 209, no. 1, pp. 69–81, Jan. 2023, doi: 10.1080/00295450.2022.2108686.

[39] C. González et al., "Hardware acceleration of digital pulse shape analysis using FPGAs," *Sensors*, vol. 24, no. 9, p. 2724, Apr. 2024, doi: 10.3390/s24092724.

[40] Red Pitaya. *STEMlab 125-14—Red Pitaya,* . Accessed: Dec. 20, 2025. [Online]. Available: https://redpitaya.com/stemlab-125-14/?srsltid=AfmBOoozqv4qVwTOaXEbvxy9SUdeqcftNLiisGkvhKPbx2wiTZrGFR9G

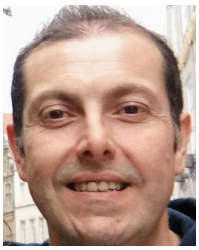
[41] *Zynq-7000 SoC Data Sheet: Overview (DS190)*. Accessed: Dec. 20, 2025. [Online]. Available: https://docs.amd.com/v/u/en-U.S./ds190-Zynq-7000-Overview

[42] *ARM Cortex-A9 Technical Reference Manual*, AMD. Accessed: Dec. 20, 2025. [Online]. Available: https://documentation-service.arm.com/static/5f0370fccafe527e86f5bfb2?token=

[43] Analog Devices. *ADALM-Pluto Active Learning Module (PlutoSDR)*. Accessed: Dec. 20, 2025. [Online]. Available: https://www.analog.com/en/resources/evaluation-hardware-and-software/evaluation-boards-kits/adalm-pluto.html
[44] Great Scott Gadgets. *HackRF One Software Defined Radio*. Accessed: Dec. 20, 2025. [Online]. Available: https://greatscottgadgets.com/hackrf/one/
[45] Analog Devices. *ADALM1000 Active Learning Module*. Accessed: Dec. 20, 2025. [Online]. Available: https://www.analog.com/en/resources/evaluation-hardware-and-software/evaluation-boards-kits/adalm1000.html
[46] National Instruments. *NI MyRIO-1900*. Accessed: Dec. 20, 2025. [Online]. Available: https://www.ni.com/docs/en-U.S./bundle/myrio-1900-getting-started/
[47] Digilent. *Analog Discovery 3*. Accessed: Dec. 20, 2025. [Online]. Available: https://digilent.com/reference/test-and-measurement/analog-discovery-3/
[48] Ettus Research. *Universal Software Radio Peripheral (USRP) SDR Products*. Accessed: Dec. 20, 2025. [Online]. Available: https://www.ettus.com/products/
[49] Red Pitaya. *Official Applications Red Pitaya*. Accessed: Dec. 20, 2025. [Online]. Available: https://redpitaya.readthedocs.io/en/latest/appsFeatures/applications/apps-featured.html
[50] Pavel Demin Red Pitaya Notes. *Vector Network Analyzer*. Accessed: Dec. 20, 2025. [Online]. Available: https://pavel-demin.github.io/red-pitaya-notes/vna/
[51] Pavel Demin Red Pitaya Notes. *Multichannel Pulse Height Analyzer*. Accessed: Dec. 20, 2025. [Online]. Available: https://pavel-demin.github.io/red-pitaya-notes/mcpha/
[52] A. K. Panda, P. Rajput, and B. Shukla, "FPGA implementation of 8, 16 and 32 bit LFSR with maximum length feedback polynomial using VHDL," in *Proc. Int. Conf. Commun. Syst. Netw. Technol.*, May 2012, pp. 769–773, doi: 10.1109/CSNT.2012.168.
[53] A. Veiga and E. Spinelli, "A pulse generator with Poisson-exponential distribution for emulation of radioactive decay events," in *Proc. IEEE 7th Latin Amer. Symp. Circuits Syst. (LASCAS)*, Feb. 2016, pp. 31–34, doi: 10.1109/LASCAS.2016.7451002.
[54] A. Farsoni, "Digital filters in radiation detection and spectroscopy," Oregon State Univ., Corvallis, OR, USA, Tech. Rep. NE/RHP 537, 2009.
[55] H. Jiang and L. Zhao, "Trapezoidal shaping algorithm based on FPGA," in *Proc. 2nd Int. Conf. Electron. Inf. Eng. Comput. Technol. (EIECT)*, Oct. 2022, pp. 341–344, doi: 10.1109/eiect58010.2022.00074.
[56] D. Alberto, E. Falletti, L. Ferrero, R. Garello, M. Greco, and M. Maggiora, "FPGA implementation of digital filters for nuclear detectors," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 611, no. 1, pp. 99–104, Nov. 2009, doi: 10.1016/j.nima.2009.09.049.
[57] U. C. Ltd. *Ubuntu 20.04.5 LTS (Focal Fossa)*. Accessed: Dec. 20, 2025. [Online]. Available: https://cdimage.ubuntu.com/releases/focal/release/
[58] *PetaLinux Tools Documentation: Reference Guide (UG1144)*. Accessed: Dec. 20, 2025. [Online]. Available: https://docs.amd.com/r/en-U.S./ug1144-petalinux-tools-reference-guide/Overview
[59] G. Marquínez-Durán et al., "GLORIA: A compact detector system for studying heavy ion reactions using radioactive beams," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 755, pp. 69–77, Aug. 2014, doi: 10.1016/j.nima.2014.04.002.

Jorge Jiménez-Sánchez was born in Alcalá del Río, Seville, Spain. He received the Telecommunications Engineering and M.Sc. degrees from the University of Seville, Seville, in 2018 and 2025, respectively, where he is currently pursuing the Ph.D. degree.

Since 2016, he has been with the Department of Electronic Engineering, School of Engineering, University of Seville. In 2024, he was an Invited Researcher at the Department of Experimental Physics, CERN, Geneva, Switzerland. His research interests include data acquisition systems for nuclear instrumentation, digital design with a focus on implementing hardware accelerators, and embedded systems.



Manuel Sánchez-Raya was born in Málaga, Spain, in 1971. He received the degree in electronic engineering from the University of Málaga, Málaga, Spain, in 1992, and the degree in computer science from the University of Málaga, in 1996. and the Ph.D. degree in electronic engineering from the University of Huelva, Huelva, Spain, in 2013.

He is a Professor at the Department of Electronic Engineering, Computers and Automation, University of Huelva. His research focuses on nuclear instrumentation, embedded systems, digital design, and real-time applications.

José María Hinojo-Montero (Member, IEEE) was born in Seville, Spain. He received the Telecommunication Engineering and Ph.D. degrees from the University of Seville, Seville, in 2010 and 2016, respectively.

From 2007 to 2024, he was a Researcher at the University of Seville. From 2024 to 2025, he was an Assistant Professor at the University of Huelva, Huelva, Spain. Since 2025, he has been an Assistant Professor at the Department of Electronic Engineering, University of Seville. He was an Invited Researcher with the Klipsch School of Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM, USA, in 2014, and the Imperial College London, London, U.K., in 2015. His research interests include power management, design of low-power low-voltage circuits, and mixed-signal processing.

Fernando Muñoz-Chavero was born in El Saucejo, Seville, Spain. He received the Telecommunications Engineering and Ph.D. degrees from the University of Seville, Seville, in 1998 and 2002, respectively.

Since 1997, he has been with the Department of Electronic Engineering, School of Engineering, University of Seville, where he has been a Professor since 1999. In 2000 and 2002, he was a Visiting Researcher at Natlab, Philips Research, Eindhoven, The Netherlands, and the Klipsch School of Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM, USA, in 2003. His research interests are related to low-voltage low-power analog circuit design, analog–digital and digital–analog conversion, and analog and mixed-signal processing.

Juan Antonio Gómez-Galán was born in Alosno, Huelva, Spain. He received the Electronic Engineering degree from the University of Granada, Granada, Spain, in 1999, and the Ph.D. degree (Hons.) from the School of Engineering, University of Seville, Seville, Spain, in 2003.

He was an Invited Researcher at the Klipsch School of Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM, USA, in the summer of 2004. He is currently a Full Professor with the Department of Electronic Engineering, Computers and Automation, University of Huelva, Huelva. His current research interests are related to analog and mixed signal processing with emphasis on low-voltage, low-power implementations, nuclear electronics for particle detectors, and electronic instrumentation in renewable energy systems.

Ismael Martel was born in Seville, Spain. He is a Professor of nuclear physics at the University of Huelva, Huelva, Spain, and the Head of the Nuclear Physics Group. He is currently teaching nuclear technology to students in the degree of Energy Engineering. His research activity focuses on the study of nuclear structure and dynamics in accelerator facilities worldwide. At the radioactive beam facility ISOLDE at CERN, Geneva, Switzerland, he is a spokesperson for several experiments and research and development projects for particle detector development, such as the ISRS spectrometer at CERN (www.uhu.es/isrs). He also led research and development projects on particle accelerator design for cancer therapy and radioisotope production.