A Data Processing System for Real-Time Pulse Processing and Timing Enhancement for Nuclear Particle Detection Systems

M. Faisal, R.T. Schiffer, M. J. Haling, M. Flaska, S. A. Pozzi, and D. D. Wentzloff

***Abstract*—This paper presents a digital data processing system that allows real-time processing of neutron and gamma-ray pulses. The captured data are processed in real time on a field programmable gate array device. A number of pulse processing algorithms are implemented on this system including correlation-based triggering and on-the-fly pulse shape discrimination. Moreover, a timing enhancement technique using correlations is presented that allows sub-sample timing accuracy of the time-of-arrival of pulses. Finally, simulation and measurement results are presented and discussed in detail to show the system's potential.**

***Index Terms*—Detectors, digital-signal processing, digitizers, liquid scintillation detectors, pulse shape discrimination.**

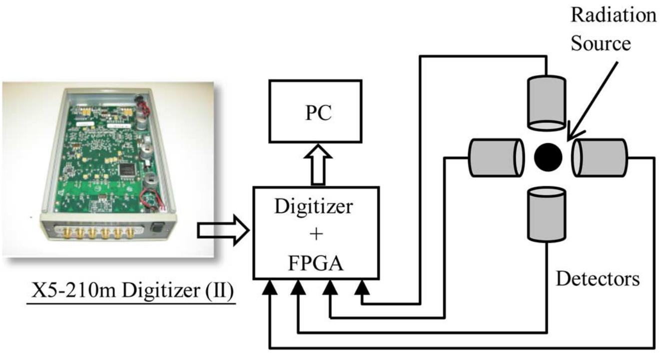


Fig. 1. The overall system block diagram.

I. INTRODUCTION

SIGNIFICANT research efforts are being focused on digital implementations of nuclear-particle detection and data acquisition systems. Typically, nuclear-particle detection systems based on liquid scintillators consist of photomultiplier tubes that convert particle energy into voltage waveforms, which are then digitized using a number of analog-to-digital converters (ADCs) [1], [2]. The digitized waveforms resemble pulses and are stored on a computer for post processing. Various off-line algorithms are often used to extract pulse features and/or to perform pulse shape discrimination (PSD). These measurements are crucial for nuclear non-proliferation and international safeguard applications where fast and efficient radioactive source identification is required [3]–[7].

This paper presents a system that implements real-time processing of captured pulses on a field programmable gate array (FPGA) device directly receiving samples from four ADCs. Specifically, pulse features such as pulse height and time-of-arrival are extracted in real time. Moreover, algorithms such as PSD and correlation-based pulse detection are implemented. The processed results are transferred to a computer over PCIe bus and a custom graphical user interface is used to display the results in real time. Additionally, a correlation-based pulse timing technique which allows sub-sample timing of pulses is proposed.

The structure of the paper is organized as follows. Section II gives a brief overview of the architecture of the data-processing system. Section III describes the algorithms that are implemented on the FPGA. Section IV gives a brief description of the graphical user interface and Section V explains how the enhanced timing technique works. Next, some simulation and measured results are presented in Section VI. Finally, the paper is concluded in Section VII.

II. DATA-ACQUISITION SYSTEM

The overall detection system, as depicted in Fig. 1, consists of four liquid scintillation detectors, a 4-channel digitizer board with embedded FPGA, and a PC. The digitizer board is an X5-210m model from Innovative Integration. The digitizer board has four ADCs for parallel measurements. The maximum sampling frequency of the ADCs is 250 MHz and their resolution is 14 bits (effective number of bits $(\mathrm{ENOB}) = 11.7\ \mathrm{bits}$). Moreover, this digitizer board features a Xilinx Virtex 5 FPGA on which the aforementioned algorithms are implemented. The captured pulses and the results of real-time signal processing performed on the FPGA are then sent to a PC via a PCIe card. In the prototype presented in this paper, both the signal-processing results and the raw waveforms are stored on a PC for verification

III. IMPLEMENTED ALGORITHMS

As mentioned above, a number of real-time algorithms are implemented on the FGPA for real-time processing of the captured pulses. Real-time processing significantly reduces the

---

Manuscript received June 14, 2012; revised February 12, 2013; accepted March 11, 2013.

M. Faisal, R. T. Schiffer, M. J. Haling, and D. D. Wentzloff are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: mufaisal@umich.edu).

M. Flaska and S. A. Pozzi are with the Department of Nuclear Engineering and Radiological Sciences, University of Michigan, Ann Arbor, MI 48109 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNS.2013.2253124

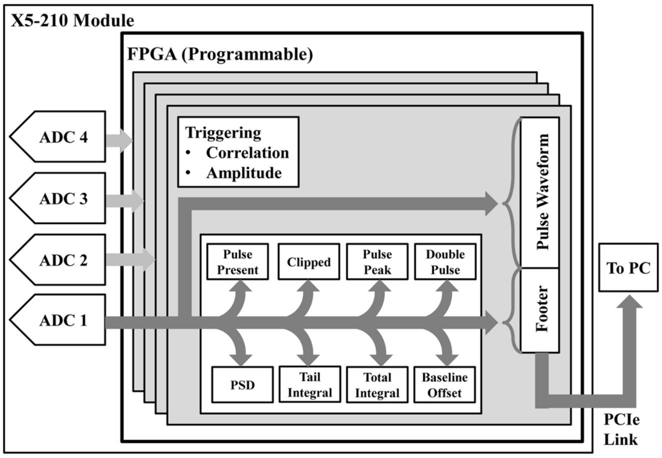


Fig. 2. The implemented algorithms and the data structure.

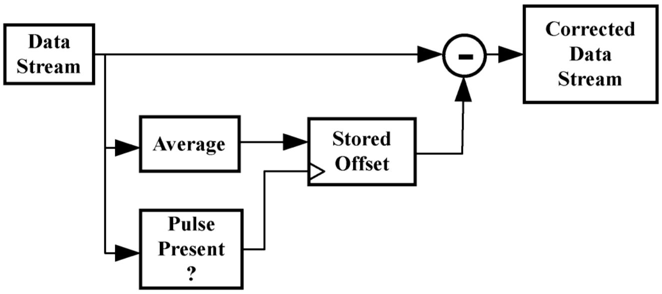


Fig. 3. Baseline correction algorithm.

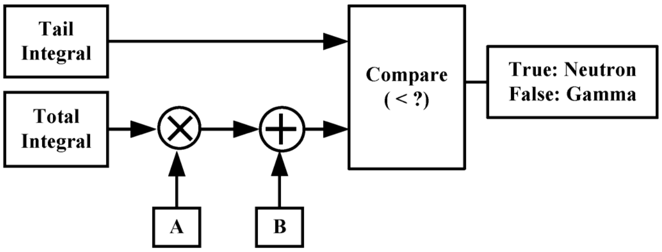


Fig. 4. Pulse shape discrimination algorithm.

amount of data to be stored on a computer and allows rapid source identification which is required for non-proliferation applications. The algorithms implemented on the FPGA for pulse processing are detailed in the subsections below. The results of these algorithms are appended to the captured pulse data and sent to the computer. Fig. 2 shows a block diagram of the algorithms implemented on the FPGA as well as the data structure of the captured waveforms which are sent to the computer.

A. Triggering

In typical, data-acquisition systems, a trigger signal determines when to start the data acquisition. In this prototype, two types of triggering mechanisms are implemented in order to detect the presence of a pulse and start capturing the data. The first type of trigger logic simply monitors the output of the ADC and starts capturing data when the output of any ADC exceeds a user-defined threshold. The trigger threshold is programmable via the software on the PC, and is typically set well above the noise level of the detectors in order to control the number of "false" trigger events due to noise crossing the threshold.

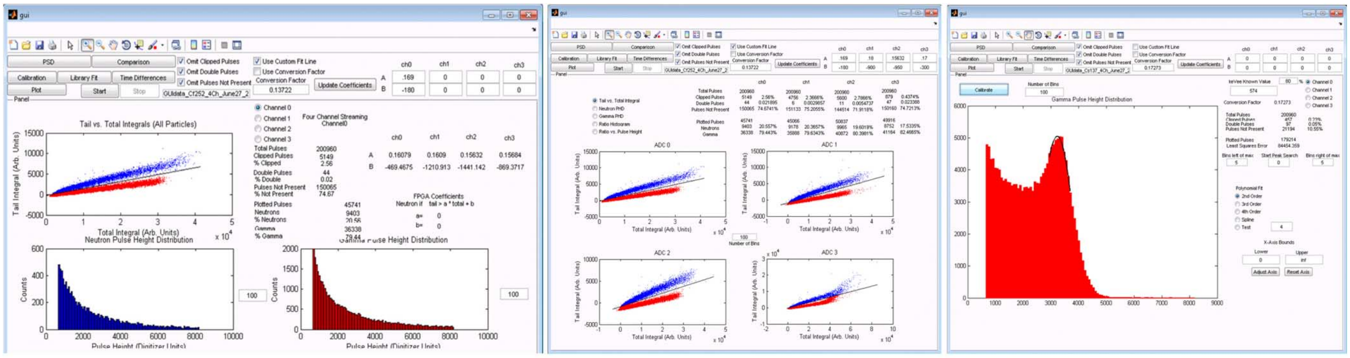
The amplitude thresholding technique works well for high amplitude pulses. However, when low energy pulses are desired, the amplitude thresholding technique is severely limited by the thermal noise, quantization noise, and detector noise [2]. Therefore, a second triggering technique is implemented which significantly improves the probability of correct pulse detection at low energies. This is a correlation-based triggering technique which enhances the detector sensitivity. As previously presented in [2], the sensitivity of a detector can be significantly improved by detecting the arrival of a pulse using a correlation-based algorithm that suppresses noise and amplifies the signal. This allows detection of pulses with amplitudes as low as the noise floor. The correlation algorithm is implemented on the FPGA in order to improve the sensitivity of the detector. This detection technique is particularly suitable for detection of shielded materials.

B. Double-Pulse and Clipped-Pulse Detection

Double pulses (i.e., pile up) and clipped pulses are typically considered corrupted data for PSD algorithms. Therefore, identifying and discarding these pulses in real time saves on disc space and bandwidth, while also improving the accuracy of the PSD. A double-pulse detection algorithm has been implemented on the FPGA of the digitizer board which detects the presence of multiple pulses within one capture window. This kind of scenario typically corrupts various measurements such as PSD and must be discarded prior to post processing. Moreover, if the amplitude of the incoming pulse is too large, it saturates the range of the ADC and appears as a clipped pulse. The saturated pulses are also flagged and can be discarded in real time.

C. Baseline-Offset Measurement

There is an inherent offset voltage in the recorded signals due to non-ideal electronics in the data-capturing system. This offset often drifts to around 1.5 mV and can introduce errors into the real-time, pulse-processing algorithms, in particular the triggering and charge-integration-based PSD. In order to compensate for this offset and align the baseline of all captured pulses to zero, an algorithm was implemented which calculates the inherent offset present at the input of the digitizer and then subtracts this offset from the data stream. What is unique about this system is this offset is recalculated prior to the capture of every pulse, and therefore can track wandering offsets sometime present in detectors. The offset is calculated from a 64-sample average of the signal just prior to every pulse arriving. In order to ensure that the signal used to calculate the offset does not contain a pulse, the latest offset is compared against the previous offset and only updated if the difference is less than a user-defined threshold. If this difference is greater than the user-determined threshold, the previous offset is used instead; otherwise, the offset is updated to the latest calculated value and subtracted from the data stream. Fig. 3 graphically illustrates this baseline correction algorithm.



(a) **(b)** **(c)**

Fig. 5. Graphical user interface snapshots.

D. Pulse Shape Discrimination

The tail and total integrals of captured pulses are computed in real time on the FGPA in order to classify the pulses as either a gamma or a neutron. The computed tail and total integrals are compared with a line that can be programmed by the user.

If the comparison is true, the pulse is classified as a neutron. Otherwise, the pulse is classified as a gamma ray [3]–[7]. This classification is then stored in the footer of the data packet (see Fig. 2) and sent to the computer. The PSD algorithm is illustrated graphically in Fig. 4.

E. Pulse Height Measurement

The pulse height is computed and stored at the end of the data structure. The results of the above mentioned measurements are appended to the captured pulse data and sent to the computer. The graphicaluser interface (detailed in the following section) displays the results in a tabular manner.

IV. GRAPHICAL USER INTERFACE

Captured pulses and their corresponding appended information are continuously being received by the PC. The software operates on these pulses in batches. First, only selected information is stripped out of the data appended to each pulse, such as the tail and total integrals, and pulse height. This data are then passed to Matlab for plotting. Each time a new batch of pulses is received, the Matlab GUI is updated, providing a real-time interface to pulse data that has been analyzed and classified on the FPGA. The maximum refresh rate of the GUI is two seconds. Fig. 5 shows a snapshot of the GUI which has four tabs for the plotting of different data and options for removing clipped and double pulses. All plots update continuously while the measurement system is acquiring data. Fig. 5(a) shows a plot of the tail versus total integrals located under the PSD tab (integrals were computed on the FPGA in real time). Neutrons are shown as blue data points while gamma rays are shown as red data points. A linear-fit line for PSD purposes is also displayed. Fig. 5(b) shows information about the data displayed by the GUI which reflects the options chosen. Fig. 5(c) is a histogram of pulse heights generated under a separate calibration tab, which is used for setting the gain of the detector amplifiers. The black fit line on the histogram plot is used to find the Compton edge in order to calculate the conversion factor between digitizer units and keVee.

V. PULSE TIMING ENHANCEMENT TECHNIQUE

In nuclear particle detection systems, the time resolution of the data is typically limited by the sampling rate of the analog-digital-converters (ADCs). However, for applications where a finer timing resolution is required to measure the precise time of arrival of particles, post processing is required in order to approximate the time to a sub-sample precision. One such method is to perform leading edge interpolation on the digitized waveform to determine the time of arrival. However, this method is sensitive to noise and can severely skew the results in situations where the energy levels of the pulses are near the noise floor. Moreover, this measurement is typically performed off line on a computer after capturing the data. Other timing algorithms include optimal filter algorithms [8]. We hereby propose an alternative, novel approach for measuring the time of arrival of pulses with a sub-sample precision. In this approach, we propose correlating the incoming data to a predetermined set of pulse templates that each has a sub-sample time shift with respect to each other. The template that results in the peak in correlation determines the sub-sample time-of-arrival of the measured data. Correlation is a widely used technique for determining how similar two signals are, and the computations used (multiplies and additions) are amenable to FPGA implementation at high clock rates.

A. Oversampled Template

When many pulses are captured using a digitizer, one can observe that the pulses are all randomly shifted with respect to the sample clock of the ADC. This means that for some pulses the ADC sample time will exactly fall on the pulse peak, while for other pulses it will not and instead points slightly before and after the peak will be sampled. This is illustrated Fig. 6(a). An oversampled template can be computed from this set of pulses by first determining the random shift between each pulse, and then realigning all of the shifted pulses based on the amount of the shift. As illustrated in Fig. 6(b) for only four pulses, the result of aligning and interleaving the pulses is an oversampled template, in this case effectively 4-times faster sampling. The procedure for computing multiple shifted templates is shown in

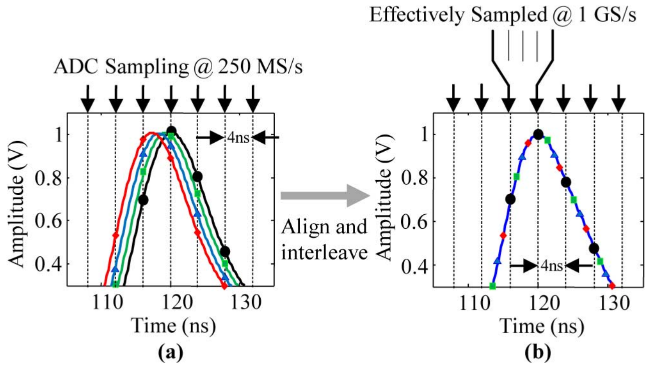


Fig. 6. Procedure for generating oversampled template.

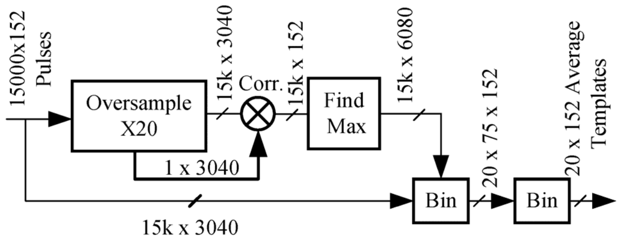


Fig. 7. Procedure for generating multiple templates.

Fig. 7. First, a set of 15,000 pulses was oversampled 20 times. Then an arbitrary pulse is selected as a reference pulse that the rest of the 14,999 pulses are correlated with. The result of the correlation is then binned into 20 time bins. The size of the bin is the sampling period/20 which in this case is $4\ \mathrm{ns}/20 = 200$ ps. Next, the original pulses are tagged with these bin numbers. Now we have the original 15,000 pulses binned according to where their oversampled correlation peaks occur. The pulses in these time bins are then averaged, and the result is twenty average template pulses that are generated from actual measured data, where each template is shifted by 200 ps. This is a one-time process. Once the oversampled templates are generated, we use them as inputs to a correlator to find the time shift that most closely matches each incoming pulse.

B. Proposed Implementation

Fig. 8 shows the proposed implementation for correlation-based timing to achieve sub-sample timing resolution. The incoming data are sent to N correlators in parallel which compute the coefficient of correlation of the incoming data to the predefined templates that were discussed in the previous section. The output of the correlator bank is sent into a peak search block which determines the index $(\mathrm{C_{max\_ind}})$ of the correlator that has the largest coefficient of correlation. In parallel, another block searches for the peak of the pulse coming in order to determine the coarse time of arrival of the incoming pulse. The fine time is then determined using the index of the maximum correlation. The final time of arrival of the pulse is given by $\mathrm{Tpulse\_final} = ((\mathrm{Cmax\_ind})\mathrm{x}(\mathrm{T_s}/\mathrm{N})) + \mathrm{T_{peak\_coarse}}$. Where $\mathrm{T_s}$ is the sampling period of the ADC and N is the number of parallel correlators.

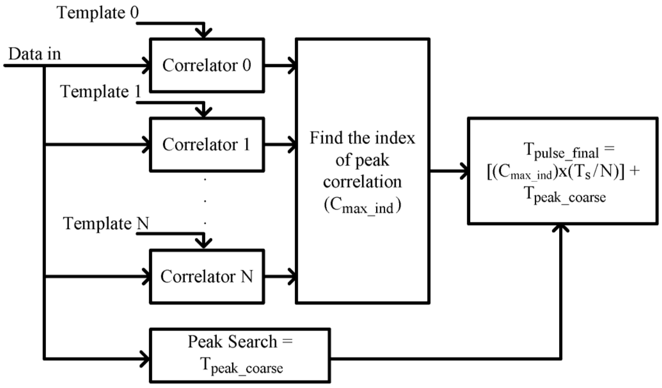


Fig. 8. Proposed implementation for pulse timing enhancement.

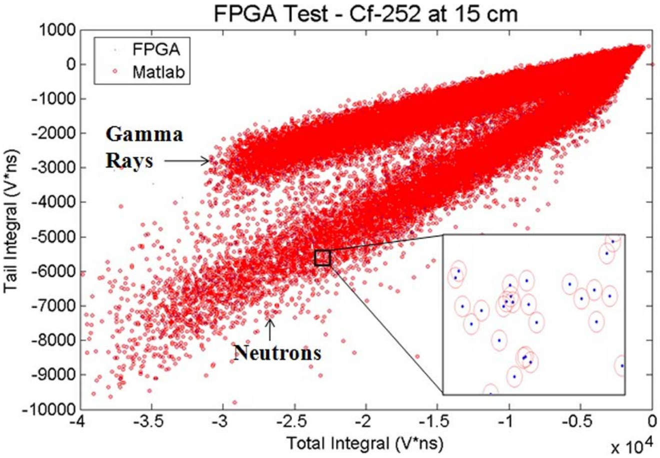


Fig. 9. PSD measurement results.

VI. Simulation and Measurement Results

All of the algorithms mentioned in Section III were implemented on the FPGA. We conducted real measurements in order to verify the correct functionality of the implementations. The raw pulse waveforms were utilized to verify that the PSD logic on the FPGA was working correctly. Specifically, the tail and total integrals of all of the measured pulses were calculated offline with Matlab using the raw pulse data. An experiment utilizing a Cf-252 source located 15 cm from an EJ-309 liquid scintillation detector was conducted and the Matlab-calculated integrals were compared to their corresponding FPGA-calculated integrals. All of the Matlab-calculated integrals matched the FPGA-calculated integrals exactly, as shown in Fig. 9 [1]. This demonstrates that the PSD logic works correctly.

In addition to the verification of the implemented algorithms, the simulation results of the proposed timing enhancement technique are presented in Fig. 10. The simulation results prove that the proposed timing technique is indeed feasible and performs better than the traditional leading edge interpolation technique in a high noise environment. Real measured data were used in this simulation. In order to compare the performance and robustness of the two timing methods, additive white Gaussian noise was added to the pulse and the timing error was measured using both techniques. Fig. 10 shows the timing error as a function of noise level while the pulse amplitude is fixed to 5 keVee

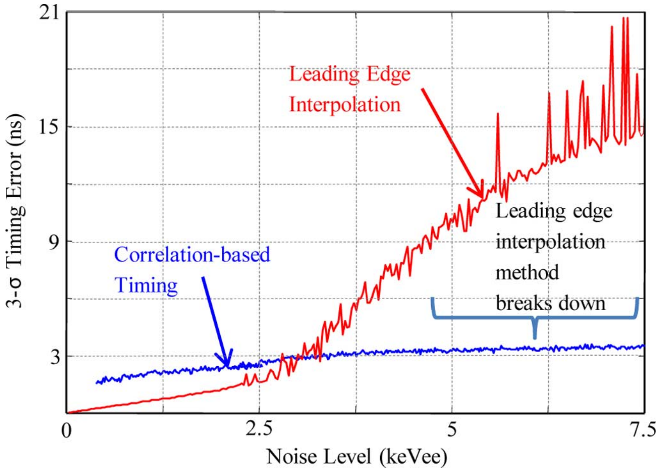


Fig. 10. Comparison of the enhanced timing technique with the leading edge interpolation technique.

(keV electron equivalent). We can see that the 3-sigma of timing error for correlation-based technique is a quarter of that for the leading edge interpolation method in the presence of a 5-keVee noise level. Moreover, the leading edge interpolation method breaks down when there is a large amount of noise present in the system. Therefore, the correlation-based technique significantly improves the timing accuracy for low-energy or shielded particles. This work has direct applications in nuclear nonproliferation.

VII. CONCLUSION

A real-time pulse-shape discrimination and data processing system for nuclear particle detection systems is presented. In addition to PSD, a number of data processing algorithms are implemented which allow pulse feature extraction. Moreover, a graphical user interface is implemented which displays the results of the real-time processing on a computer. The graphical user interface updates the results in real-time which allow the user to observe a real-time portrait of the radioactive source. The real-time processing of pulses on the FPGA significantly reduces the amount of storage and off-line processing required for such systems. Therefore, this system is an important milestone towards miniaturization of nuclear particle detection systems. Moreover, a pulse timing enhancement technique is presented which allows pulse timing at a higher resolution than the ADC sampling rate. Simulation results show that the proposed technique performs significantly better than the traditional leading edge interpolation technique. This technique is particularly appropriate for determining the correct time of arrival of pulses that are on the order of the noise level.

REFERENCES

[1] R. T. Schiffer, M. Flaska, S. A. Pozzi, S. Carney, and D. D. Wentzloff, “A scalable FPGA-based digitizing platform for radiation data acquisition,” *Nucl. Inst. Meth. A*, vol. 652, pp. 491–493, 2011.

[2] M. Faisal, R. T. Schiffer, M. Flaska, S. A. Pozzi, and D. D. Wentzloff, “A correlation-based pulse detection technique for gamma-ray/neutron detectors,” *Nucl. Inst. Meth. A*, vol. 652, pp. 479–482, 2011.

[3] B. Sabbah and A. Suhami, “An accurate pulse-shape discriminator for a wide range of energies,” *Nucl. Instr. Meth. A*, vol. 58, pp. 102–110, 1968.

[4] J. Kalyna and I. J. Taylor, “Pulse-shape discrimination: An investigation of n- γ discrimination with respect to size of liquid scintillator,” *Nucl. Instr. Meth. A*, vol. 88, pp. 277–287, 1970.

[5] P.-A. Söderström, J. Nyberg, and R. Wolters, “Digital pulse-shape discrimination of fast neutrons and γ rays,” *Nucl. Instr. Meth. A*, vol. 594, pp. 79–89, 2008.

[6] B. Esposito *et al.*, “Digital pulse-shape discrimination in organic scintillators for fusion applications,” *Nucl. Instr. Meth. A*, vol. 518, pp. 626–628, 2004.

[7] M. Flaska and S. A. Pozzi, “Identification of shielded neutron sources with the liquid scintillator BC-501A using a digital pulse shape discrimination method,” *Nucl. Instr. Meth. A*, vol. 577, pp. 654–663, 2007.

[8] A. Bousselham, P. Ojala, and C. Bohm, “Digital timing with non-stationary noise optimal filter algorithm for LSO/APD detectors,” in *Proc. IEEE Nucl. Sci. Symp. Conf.*, 2007.