XXVII ENMC
ENCONTRO NACIONAL DE MODELAGEM COMPUTACIONAL
XV ECTM
ENCONTRO DE CIÊNCIA E TECNOLOGIA DE MATERIAIS
2024

REAL-TIME FPGA-BASED SIMULATOR FOR THE TILE CALORIMETER READOUT SYSTEM IN THE ATLAS EXPERIMENT

Fábio Cardani Luna[1] - fabio.luna@engenharia.ufjf.br
Ualison Ferreira Dias[1] - ualison.dias@engenharia.ufjf.br
Pedro Henrique Braga Lisboa[2] - pedro.lisboa@lps.ufrj.br
Thiago Campos Acácio Paschoalin[1,3] - thiago.paschoalin@cefetmg.br
Tiago Motta Quirino[4] - tiago.quirino@eng.uerj.br
Luciano de Manhães Andrade Filho[1] - luciano.andrade@engenharia.ufjf.br
[1]Universidade Federal de Juiz de Fora - Juiz de Fora, JF, Brazil
[2]Universidade Federal do Rio de Janeiro - Rio de Janeiro, RJ, Brazil
[3]Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG), Unidade Leopoldina, MG, Brasil
[4]Universidade do Estado do Rio de Janeiro - Rio de Janeiro, RJ, Brazil

Abstract. *This work proposes a real-time simulator for the readout system of the Tile Calorimeter (TileCal) in the ATLAS experiment, utilizing Field-Programmable Gate Arrays (FPGAs). This simulator models the TileCal readout chain, including the dynamic sensor and signal conditioning circuits, generating physics data at a rate of 40 MHz, synchronized with the LHC collision rate. The choice of FPGAs is driven by the need for parallel processing and high performance. The implemented simulator occupies only 3% of the FPGA resources, offering real-time processing and allowing for dynamic adjustments, despite the challenges of hardware design and resource optimization.*

Keywords: *Real-time processing, High-energy physics, FPGA*

1. INTRODUCTION

Numerous laboratories worldwide are at the forefront of technological innovation, especially in research, development, and application. CERN (European Organization for Nuclear Research) stands out, primarily focusing on particle physics. However, its research is not limited to this field alone, extending to solid-state physics, medicine, biology, cosmology, astrophysics, and computing (Scientific American, 2024). CERN has been responsible for numerous discoveries and innovations, such a remarkable discovery is the confirmation of the Higgs Boson in 2012 by the Large Hadron Collider (LHC) (ATLAS Collaboration, 2012).

Brazil beginning in the mid-1980s and expanding since then, culminating in becoming an associate member state on March 13, 2024 (ALVES Jr. et al., 2014), and collaborates intensively is in software and hardware, developing everything from systems that enhance communication

to boards aimed at instrumentation. These contributions apply models of signal processing, applied statistics, artificial intelligence, and computational methods (GOOSSENS et al., 2010). The electronic component is crucial for research on detectors. Without advances in this field, many experiments and discoveries would be impossible, given the use of various sensors and detectors that capture data from collisions (ATLAS Collaboration, 2013). The particles collide with each other at the center of the experiments installed in the LHC accelerator. ATLAS is a general-purpose experiment consisting of several layers of sensors that measure different characteristics of the collision byproducts. The detector of interest in this work is the Tile Calorimeter (TileCal), which forms one of the layers of the experiment and is responsible for measuring the energy of the hadrons released from the collisions (ATLAS Collaboration, 2008).

As the entire system requires flexibility, parallel processing capability, and high performance, using FPGAs (Field-Programmable Gate Arrays) is essential, especially in real-time data processing, as seen in the first-level trigger filters of the experiment (WALLON et al., 2015). These filters select interesting events in real time during collisions. Due to the enormous volume of data generated, this filtering phase is crucial for storing and later analyzing only the relevant data. In summary, FPGAs play a vital role in many LHC applications, from data acquisition and processing to the control and synchronization of the equipment used (BENNIS et al., 2017).

Real-time simulation on FPGAs has found widespread application across various domains, including power electronics (Slater et al., 1998; Parma and Dinavahi, 2007; Herrera and Wang, 2013; Gaitán Cubides et al., 2022) and, more recently, particle physics (Neu et al., 2024). The work presented in (Herrera and Wang, 2013) focuses on developing a real-time simulation platform that leverages Field Programmable Gate Arrays (FPGAs) for modeling power converters operating on electric vehicles. This approach facilitates highly precise and fast simulations, which are crucial for real-time applications and hardware in the loop testing. In Gaitán Cubides et al. (2022), the authors provide a comprehensive overview of Real-Time Digital Simulators (RTDS) and their applications, particularly in the context of global energy transition. Finally, in (Neu et al., 2024), the authors compare several hardware-accelerated, graph-based preprocessing methods for the Belle II experiment (Abe et al., 2010). These techniques have been extensively used to prepare data for Artificial Neural Network models, especially in the context of trigger systems within particle physics detectors. Traditionally, these algorithms are tested in software environments, where graph features are pre-calculated from detector data. In this work, however, the authors implement and evaluate these preprocessing methods on FPGA, enabling the algorithms to be tested in conditions closer to real-world system operation. Their results demonstrate that the proposed preprocessing approaches successfully meet the tight performance requirements for both throughput and latency.

Implementing a real-time simulator is extremely important in scenarios where response time is critical, such as in the first-level filters of the LHC experiments. These filters need to process large volumes of data in real-time, selecting relevant events that will be stored for future analysis. A real-time simulator can replicate this environment, allowing new technologies to be tested under near-real conditions before being deployed in the experiment (Neu et al., 2024).

The main advantage of an FPGA-based simulator compared to an offline software-based simulator is the ability to process data in real-time, providing immediate feedback and allowing dynamic adjustments. However, while offline software is easier to develop, debug, and update, it cannot offer the same level of performance in applications where latency and processing speed are critical (Neu et al., 2024).

Despite its advantages, implementing a real-time simulator on FPGA poses significant

challenges. The complexity of hardware design, the need for resource optimization, and development time are obstacles that can slow down and increase the cost of the process. Furthermore, FPGA development requires a high level of technical expertise, which may limit the number of teams capable of contributing to the project (Slater et al., 1998).

To achieve the stated goals, we propose the implementation of a real-time simulator using FPGAs, which is crucial for simulating data at speeds comparable to actual collisions in the LHC. This approach will enable more effective validation and development of new technologies. Our work is motivated by the need to model a simulator that can be used to train and develop these technologies, facilitating their integration into the collider environment. The simulator will generate physics data at a speed equal to or close to that of real collisions, allowing other teams to utilize the generated results for their projects (KHAN et al., 2020).

The text is structured as follows: Section 2 provides an overview of the TileCal readout system model; which we aim to simulate. Section 3 discusses the hardware implementation for the real-time digital simulator; Finally, Section 4 presents the conclusions of this work.

2. TILECAL READOUT SYSTEM MODEL

The readout process in the TileCal is represented by the block diagram in Figure 1. The process begins with the interactions of hadronic particles with scintillators, which produce light that is transmitted via optical fibers to a photomultiplier tube (PMT), converting it into a negative electrical current pulse. The generated signal is then processed by an electronic conditioning circuit, where it is stretched in time to match the frequency response of the readout channel and amplified 64 times in high-gain mode. Finally, the signal is digitized by 10-bit ADCs at a sample rate of 40 MHz, due to the synchronization across the entire LHC readout system and the collisions, which cannot be altered (Anderson et al., 1998; ATLAS Collaboration, 2013).

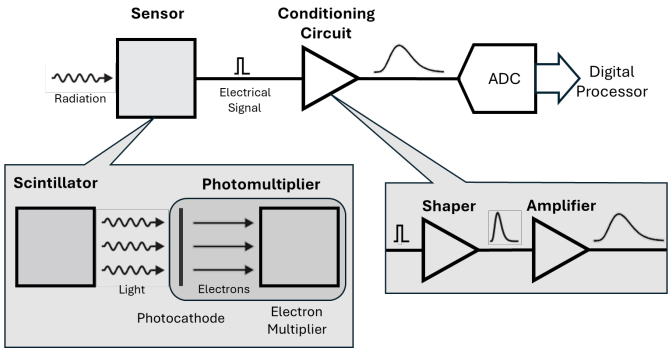

Figure 1: Block diagram of detector readout chain.

The interaction events of particles with the readout channels occur within a time interval comparable to the data acquisition rate. This allows the energy depositions to be interpreted as discrete impulses with amplitudes corresponding to the deposited energy. This interpretation is further strengthened when the readout channel is understood as a convolutional model, where all the dynamic characteristics of the TileCal's instrumentation system are concentrated into a characteristic pulse ($h[n]$), and the energy deposition is instantaneous, modeled by a randomly generated energy deposition signal ($s[n]$). The readout of the measurement channel ($r[n]$) is therefore given by convolution operation ($r[n] = h[n] * s[n]$).

There are approximately 10,000 measurement channels in the TileCal, distributed across different regions of the detector and designed to be identical to each other. This uniformity ensures that all channels share the same characteristic pulse $h[n]$. However, each scintillator is interleaved with steel plates that absorb a fraction of the particle's energy, leading to variations in event rate and energy depending on the channel's location, which in turn modifies the signal $s[n]$.

A readout channel simulator is proposed using a convolutional model, as represented in the flowchart in Figure 2. To generate the signal $s[n]$, two random numbers must be provided, derived from the parameters occupancy ($0 \leq \rho \leq 1$) and average pulse amplitude ($1/\lambda > 0$), both of which depend on the channel's location. Additionally, the characteristic pulse $h[n]$ must be known to produce the readout signal $r[n]$.

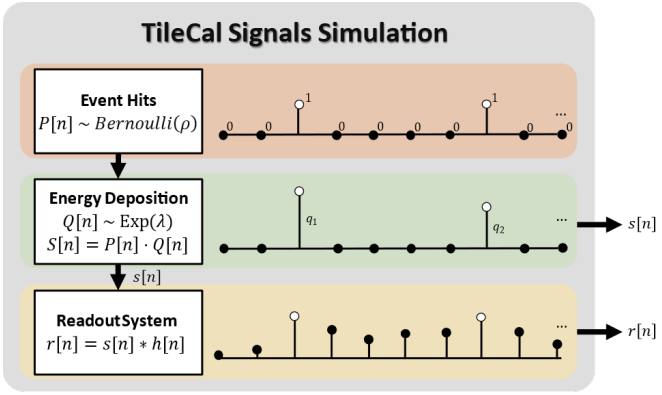

Figure 2: TileCal Signals Simulation.

The occupancy ρ represents the event rate in the channel, or the probability of an event, modeled by a Bernoulli distribution to generate a Bernoulli sequence. The model assumes a signal sampled at a 40 MHz rate, where each sample is independent of the others and signifies an event occurrence, as shown in the following equation:

$$P[n] = \begin{cases} 1 & with\ probability\ \rho \\ 0 & with\ probability\ 1-\rho \end{cases} \tag{1}$$

In the context of an exponential random variable $Q[n]$ representing the amplitude of energy deposition, provided by the analysis of particle interactions in the detector (ATLAS Collaboration, 2017). λ is the rate parameter of the exponential distribution and it reciprocal $1/\lambda$, represents the expected value or mean of the distribution. This mean is the average amplitude of energy deposition per event.

By the product of the random variables $P[n]$ and $Q[n]$, the random variable $S[n]$ is obtained, representing the energy deposition signal, which, when realized, generates the signal $s[n]$.

The definition of $h[n]$ is derived from the analysis of the sensors and the readout electronics, specifically the shaper circuit.

Scintillator detectors exhibit a significant temporal response in the formation of emitted light when interacting with radiation. The scintillation process is not instantaneous; it is characterized by a rapid rise in light intensity followed by an exponential decay. This temporal profile is further influenced by the capacitive properties of PMTs, which are also modeled using

exponential functions to describe the processes of charge accumulation and discharge (Wright, 2017).

The signal from sensors can be accurately modeled using a multi-exponential function, which captures both the rapid rise and subsequent decay phases of the light pulse, as well as the associated electrical charging and discharging processes. Despite this complexity, the electrical output pulse from a scintillation detector is often effectively approximated by a bi-exponential function ($e^{-t/\tau_D} - e^{-t/\tau_R}$) (Wright, 2017).

By analyzing the bi-exponential pulse provided by the PMT, as Spieler (2002), it is possible to model the entire sensor assembly as a current source followed by a capacitor. This model is coupled to the shaper circuit, as shown in the figure 3, and it is noted that the readout chain is then modeled by an RLC circuit. The discrete energy deposition impulses are applied to the source $i_s(t)$ and can, therefore, be represented by $s[n]$.

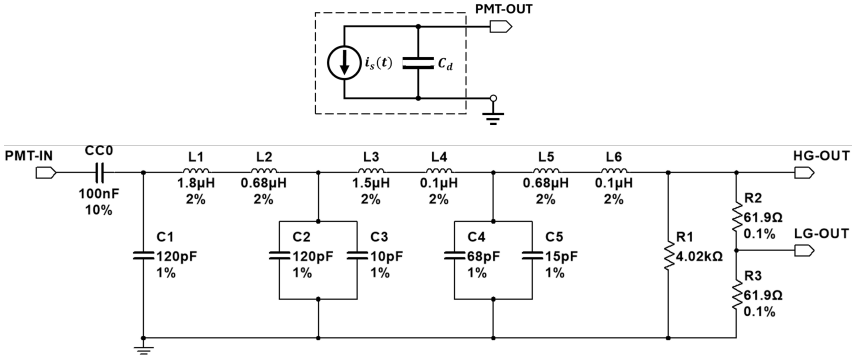

Figure 3: Readout circuit model.

The transfer function $H(s)$ of the readout chain circuit is characterized by 7 poles, which can be decomposed into first and second-order modes. This decomposition results in 5 components. Applying the inverse Laplace transform yields $h(t)$, which can be digitized to $h[n]$ by considering $n = t/T_s$, with $T_s = 25 \cdot 10^-9$.

$$h(t) = 2.3 \cdot 10^7 e^{-4.0 \cdot 10^7 t} - 7.7 \cdot 10^2 e^{-1{,}1 \cdot 10^9 t} - 4.4 \cdot 10^7 e^{-3.6 \cdot 10^7 t} cos(6.2 \cdot 10^7 \cdot t - 45.0°) - \\ - 3.5 \cdot 10^4 e^{-8.8 \cdot 10^4 t} - 4.4 \cdot 10^7 e^{-3.6 \cdot 10^7} cos(6.2e^7 \cdot t + 27.0°) \tag{2}$$

According to the figure, the characteristic pulse $h[n]$ is convolved with $s[n]$ to generate the final signal $r[n]$ in the TileCal readout simulation.

3. HARDWARE IMPLEMENTATION

The paper proposition is to develop an online calorimeter simulator running at FPGA. The simulator will be a very important tool for test online techniques of energy estimation and particle detection for TileCal.

To do so, each part of the process was separated in small tasks to facilitate the comprehension. The system has to produce sample every 25 ns because the collision rate is 40 MHz at the calorimeter. The simulator needs to work at different occupancy levels.

The solution will generate random positions for the collisions and random energy values for this collisions. The channel response needs to be considered, to complete the final simulated dataset.

3.1 Random Number Generator

To begin developing the code for the simulator is necessary to develop a digital circuit capable of generating random numbers that will be important for determining when collisions occur and the energy of this event. It was choosen a 7-bit output random number for to accomplish the task.

The method used was the LCG (Linear Congruential Generator), which is a pseudo-random number generator based on the formula described at Eq. 3.

$$X_{n+1} = (a \cdot X_n + c) \,\%\, m \tag{3}$$

Where a is the multiplier, c is the increment and m is the factor used for the Modulus operation associated to the method. The X_n represents the actual random number generated and the X_{n+1} is the next output number. For this project, the parameters chosen were: a = 75, c = 74 e m = $2^{16} + 1$.

Describing the development, the parameters were defined as previously mentioned. As inputs, the block has a 40 MHz clock to control the registers and the reset signal to return at initial values. The output in is only the 7-bit random value founded with Eq. 3.

A 32-bit register was also created, which is a shift register designed to hold the data shown in the LCG equation. The high number of bits is justified due to the fixed points operations related. Implementing the logic, at each clock edge, the formula is applied to update the value of the X_n register.

Finally, the module's output receives a truncated value from the 32 -bit register, picking the 7 Most Significant Bits, to accomplish the 128 random states. The figure 4.

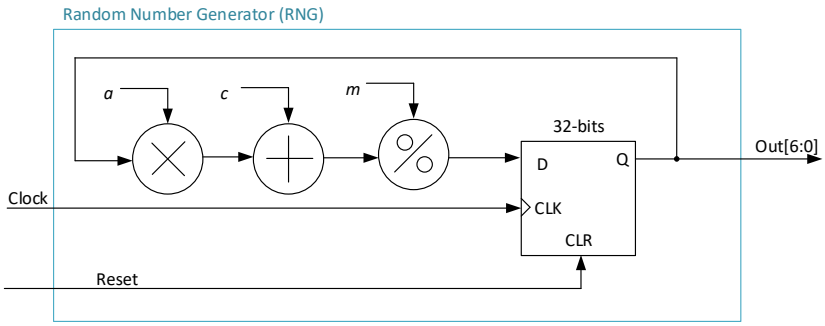

Figure 4: Random Number Generator

3.2 Determination of Collision Positions

To assign collision positions, the generated random number is used. So, it is important to receive this information as input from the module described at subsection 3.1. Other important input pin is the the occupancy level of the dataset that will determine the number of events present at an observation time. This factor also has 7 bits to relate the number received.

This block has a simple task. The random number received is compared to the occupancy set of the simulated cell, and if this number is lower than the chosen level, the block informs that occurs a collision, indicating an unitary value. Otherwise, a null value is set to informs that is no event of interest in that sample. This collision information is conducted to the exterior by a single bit output pin.

The occupancy parameter mentioned represents the percentage of collision positions relative to the total number of samples. In this case, the value of the occupancy it is actually found indirectly. For example, a 50% occupancy, the input number set to be compared would be 64. This can be described by the the Eq. 4.

$$occ = p(\%) \cdot \frac{128}{100} \tag{4}$$

Where p is the desired occupancy rate in percentage. The module that accomplish the goal is represented with the Fig. 5

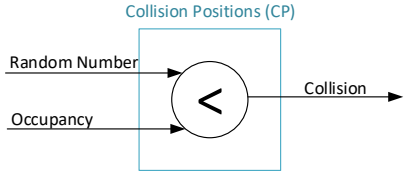

Figure 5: Collision Positions

3.3 Assigning Energy to Collisions

Other important task is to assign energy to the points where collisions occurred. To achieve this, a second random number generator is used.

The energy of the collisions is defined by a exponential distribution function and has a heavy operation effort for a FPGA hardware. To do so, other approximated approach was developed. The idea is to generate the values of the wanted distribution with an offline algorithm along the inputs, with fixed steps between then, and save to a memory at FPGA. With the input value provided by the random number generator as the addrress of the device, the memory is capable to the distribution output associated to the entry in a single clock.

In this case, a memory with 128 positions with 10 bits each one was defined for this proposition because they seems to be enough. The 128 positions are the steps described before and the 10-bits registers is important to carry all necessary information that would be lost with a fixed point operation.

The module has the clock, reset input pins and an 7-bit input that receives the random number. The output data is a 10-bit that represents the energy of the colision.

To summarize the operation block, The random number generator assigned to the input pin is attached to the memory address to find the correct value of the exponential distribution previously calculated. The memory product is the resultant energy of the collision and is connected to the output pin. The Fig. 6 represents the block procedure described.

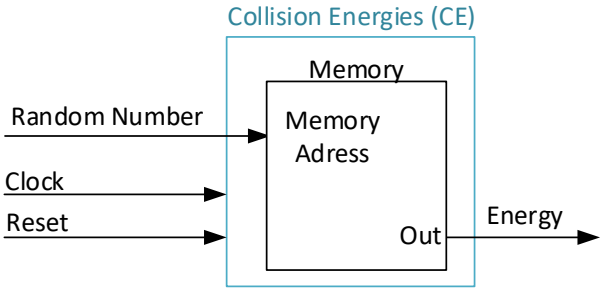

Figure 6: Energy of Collisions

3.4 Simulation of the Calorimeter

With the collisions positions and their energy defined, is possible to complete the final step of the simulated calorimeter. Until then, the impulsive energy value is knew but the calorimeter cell reading is an pulse that is spread in more samples.

To achieve this, the impulse response of the calorimeter is crucial. The output samples of the reading is the convolution product between this impulse response and the energy amplitude.

A simple way to do a convolutional product at FPGA is using a FIR (Finite Impulse Response) filter. The input samples are delayed with a determined parameter and each of them are multiplied by a weight. The sum of this multiplications results is the convolution product. To find the calorimeter output, the weights of the FIR filters are set with their impulse response parameters.

It is important to emphasize that the weight of the FIR filter needs to be quantized. Using the rounded value of then will not preserve the correct response of the real signal, so it is necessary to multiply by a certain factor. This solution a light implementation because the float point emulated operations demands a lot of resources.

For this specific calorimeter simulation, the input signals are the clock, 10-bit signal with samples sequences containing the collision informations. The output pin has 21 bits to bear the resultant convolution product without data loss. The calorimeter impulse response has 7 parameters. This block is represented by the Fig. 7

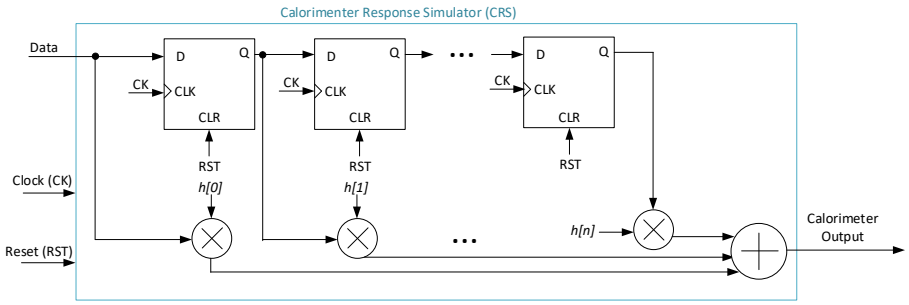

Figure 7: Calorimeter Response

With this step done, an online simulator is done and the parts described before are all linked up. The Fig. 8 shows a simplified version how they are connect to accomplish the task.

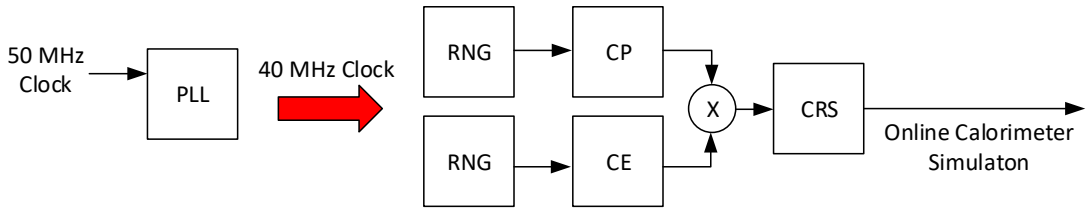

Figure 8: Online Calorimeter Simulator

In Fig. 9 , the result of the simulator in the ModelSim hardware simulation environment can be seen. The first signal is the 40 MHz clock that synchronizes the entire system. The second signal represents the generation of energy depositions, which is the combination of the CP and CE Block in Fig. 8. Finally, the calorimeter response is added, generating the simulator's output signal. The last signal in this figure is the output of an FIR deconvolution filter, aimed at recovering the energy deposition information. Thanks to this simulator, the reconstruction filter is tested in real time at the correct data acquisition rate of TileCal.

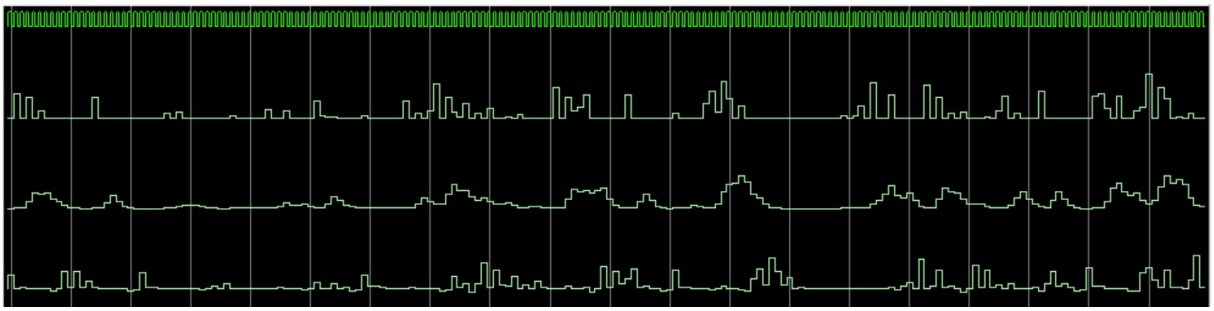

Figure 9: Online Calorimeter Simulator

The system was tested on the FPGA of the mentioned kit, and its correct real-time operation was observed. The simulator occupies only 3% of the FPGA's logic resources, and its maximum operating frequency is 100.96 MHz, well above the usage frequency, which is 40 MHz.

4. CONCLUSIONS

A simulator was designed by considering the interaction parameters of particles with the detectors, depending on the location of the readout channel within the detector. It is recognized that both the event rate and the average pulse amplitude differ across channels. Based on these parameters, random numbers were generated, respecting the varying characteristics of the channels. Additionally, the entire dynamics of the readout chain were combined into a convolutional model, represented by a characteristic pulse for the readout channel, thereby forming the channel's measurement signal. This approach enabled an approximate reproduction of the operational conditions of the TileCal.

It was demonstrated that the FPGA implementation was feasible at a rate of 40 MHz, with the simulator occupying only 3% of the FPGA's resources, allowing for the potential implementation of more complex or additional systems on the same hardware in the future.

For future work, a statistical analysis of the simulated signals is proposed with sufficient rigor to ensure the reliability of the simulators and their application in the development of readout systems for particle physics experiments.

REFERENCES

Abe, T. et al. (2010). Belle II Technical Design Report.

ALVES Jr., A. A., Carvalho, M. M., Sousa, D. B., and Pereira, R. J. (2014). Brazil's contribution to the large hadron collider. *The European Physical Journal C*, 74:2882.

Anderson, K., Pilcher, J., Sanders, H., Tang, F., Berglund, S., Bohm, C., Holmgren, S., Jon, K., Blanchot, G., and Cavalli-Sforza, M. (1998). Front-end electronics for the atlas tile calorimeter. In *Proceedings of Fourth Workshop on Electronics for LHC Experiments, Rome*, page p239.

ATLAS Collaboration (2008). The atlas experiment at the cern large hadron collider. *Journal of Instrumentation*, 3:S08003.

ATLAS Collaboration (2012). Observation of a new particle in the search for the standard model higgs boson with the atlas detector at the lhc. *Physics Letters B*, 716(1):1–29.

ATLAS Collaboration (2013). Cern and the higgs boson discovery. *Nature*, 498:302–303.

ATLAS Collaboration (2017). Technical design report for the phase-ii upgrade of the atlas tile calorimeter. Tech. Rep. CERN-LHCC-2017-019, ATLAS-TDR-028, CERN.

BENNIS, L., Haddad, Y., Othman, K., and Tran, D. (2017). Advances in fpga technology for high-energy physics. *Journal of High Energy Physics*, 2017(6):25.

Gaitán Cubides, L. F., González Sánchez, J. W., and Giraldo Velazquez, L. A. (2022). A review of real time digital simulations: Theory and applications for the energy transition. *IEEE Latin America Transactions*, 20(10):2295–2307.

GOOSSENS, M., Ma, L., Xu, Q., and Zhang, H. (2010). Application of fpgas in high-energy physics experiments. *IEEE Transactions on Nuclear Science*, 57(2):490–497.

Herrera, L. and Wang, J. (2013). Fpga based detailed real-time simulation of power converters and electric machines for ev hil applications. In *2013 IEEE Energy Conversion Congress and Exposition*, pages 1759–1764.

KHAN, R., Lee, T., Gomes, P., and Santos, L. (2020). Modeling simulators for lhc data analysis. *IEEE Computing in Science Engineering*, 22(4):45–52.

Neu, M., Becker, J., Dorwarth, P., et al. (2024). Real-time graph building on fpgas for machine learning trigger applications in particle physics. *Computing and Software for Big Science*, 8:8.

Parma, G. G. and Dinavahi, V. (2007). Real-time digital hardware simulation of power electronics and drives. *IEEE Transactions on Power Delivery*, 22(2):1235–1246.

Scientific American (2024). Cern: Expanding research beyond particle physics. *Scientific American*. Accessed on: 14 Aug. 2024.

Slater, H. J. et al. (1998). Real-time emulation for power equipment development. part 2: The virtual machine. *IEE Proceedings - Electric Power Applications*, 145(3):153.

Spieler, H. (2002). Pulse processing and analysis. In *IEEE Nuclear Science Symposium Short Course*, pages 52–90, San Francisco. IEEE.

WALLON, S., Dubois, L., Martin, C., and Smith, J. (2015). Fpga-based data processing in lhc experiments. *IEEE Transactions on Nuclear Science*, 62(3):1203–1210.

Wright, A. (2017). *The photomultiplier handbook*. Oxford University Press, Oxford.