



## Simulation of the Upgraded Digital Signal Processing Firmware Using High-Level Synthesis on the Real-Time Trigger Path of the ATLAS Liquid Argon Calorimeters

**Lucca Oliveira Facio Viccini<sup>1</sup>** - lucca.viccini@engenharia.ufjf.br

**Melissa Santos Aguiar<sup>1</sup>** - melissa.aguiar@engenharia.ufjf.br

**Marcos Vinícius Silva Oliveira<sup>2</sup>** - mo@bnl.gov

**Huacheng Cai<sup>3</sup>** - huacheng.cai@cern.ch

**Nick Fritzsch<sup>4</sup>** - nick.fritzsche@cern.ch

**Paolo Rondot<sup>5</sup>** - prondot@student.ethz.ch

**Luciano Manhães de Andrade Filho<sup>1</sup>** - luciano.andrade@engenharia.ufjf.br

<sup>1</sup> Federal University of Juiz de Fora - Juiz de Fora, MG, Brazil

<sup>2</sup>Brookhaven National Laboratory - Upton, New Jersey, United States of America

<sup>3</sup>University of Pittsburgh - Pittsburgh, Pennsylvania, United States of America

<sup>4</sup>European Organization for Nuclear Research (CERN) - Meyrin, Geneva, Switzerland

<sup>5</sup>ETH Zurich - Zurich, Switzerland

**Abstract.** Precision and speed in instrumentation systems are essential in high-energy physics experiments to study fundamental particles. The Large Hadron Collider (LHC) at CERN relies on advanced systems like the ATLAS detector. The ATLAS Liquid Argon Calorimeter (LAr) has suffered upgrades to enhance its data processing capabilities, including the introduction of LATOME boards for back-end signal processing in Run 3. This paper focuses on the functional simulation steps taken to verify latest firmware upgrades for the LATOME boards, which incorporate High-Level Synthesis (HLS) components. A multi-layered simulation approach is presented to ensure the new firmware's reliability and performance. The validation process includes core functionality tests, integration with serializers and deserializers, and full system-level simulations.

**Keywords:** High Energy Physics, FPGA, High-Level Synthesis, Functional Simulation, Low latency

### 1. INTRODUCTION

High-energy physics experiments investigate the fundamental components of matter and their interactions, using particle accelerators to collide beams at near-light speeds (Perkins, 2000). These collisions produce rare events that require sophisticated, high-speed, and precise instrumentation (Barletta et al., 2014; Brüning & Rossi, 2019).

The Large Hadron Collider (LHC), currently the world's most powerful accelerator, operates in a 27 km underground tunnel (Evans & Bryant, 2008), with four major experiments including ALICE (ALICE, 2008), LHCb (LHCb, 2008), CMS (CMS, 2008), and ATLAS (ATLAS, 2008).

The ATLAS experiment, illustrated in Figure 1, is the largest and most complex detector from LHC, featuring a 25-meter tall, 44-meter long cylindrical structure with three main sub-detectors: the Muon Spectrometer, the Inner Tracking Detector and the calorimeters.

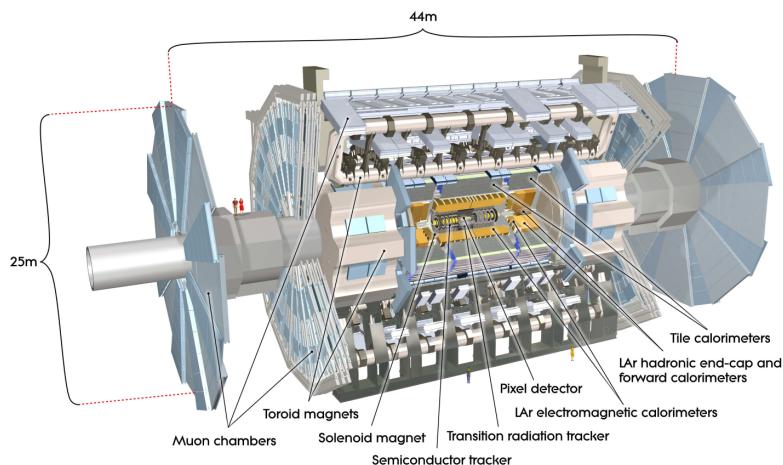


Figure 1: Diagram of ATLAS and its subsystems extracted from (ATLAS, 2008).

The calorimeters measure particle energy through showers generated in their material. The Tile Hadronic Calorimeter (TileCal) and the Liquid Argon Calorimeter (LAr) in ATLAS handle hadronic and electromagnetic showers, respectively. The LAr calorimeter detects electrons, photons, and hadrons, using layers of metal and ionized liquid argon to produce signals, which are digitized to determine the original particle's energy and position.

Given the high collision rate of 40 MHz at the LHC, ATLAS generates vast amounts of data per second. The Level-1 (L1) Trigger system filters this data in real-time, relying on energy measurements from the Liquid Argon (LAr) calorimeter to make rapid decisions on which events to retain for further analysis, ensuring that only the relevant data is preserved.

To handle the increased data rates and improve the trigger decision process during Run 3 of the LHC, the LAr calorimeter has undergone a significant upgrade. This upgrade extends the legacy system with new front-end components, sending the Super Cells' (SC) digitized data to new back-end components (Aad et al., 2022). Super Cells are aggregations of multiple calorimeter cells that provide finer granularity and more detailed information about the energy deposited by particles. As part of the upgrade to the back-end electronics, the LATOME (LAr Trigger prOcessing MEzzanine) boards play an important role in pre-processing data for the L1 trigger system.

The LATOME firmware has undergone a series of upgrades, the most recent being the replacement of two blocks with new ones developed using the High-Level Synthesis (HLS) tool, Catapult by Siemens. This approach using HLS offers several benefits, including improved performance, reduced development time, and increased maintainability. This paper focuses on the functional simulation of the latest LATOME firmware using these new HLS components. The functional verification is performed using a novel multi-layered approach to mitigate long simulation run times and to test different classes of issues separately for each of the verification steps. More details are described in the subsequent chapters.

## 2. LATOME

As part of the upgrade, the LAr Trigger Digitizer Boards (LTDBs) now digitize the analog signals from the LAr calorimeter's Super Cells at 40 MHz. These analog-to-digital converted (ADC) signals are sent to the back-end electronics, specifically the LAr Digital Processing Blades (LDPBs), which calculate the SCs transverse energy and transmit this data to the new Level-1 (L1) Trigger system. This new trigger system comprises a set of Feature EXtractors (FEX) with three subsystems targeting electromagnetic (eFEX), jet (jFEX) and global (gFEX) features (Aad et al., 2022).

The LDPB consists of four Advanced Mezzanine Cards (AMCs), called LATOMEs, integrated into the LAr Carrier (LArC) ATCA board, as shown in Figure 2. The LAr calorimeter uses 116 LATOMEs to cover the entire detector. Each LATOME features an Intel Arria 10 FPGA with a heat sink (to the left of the picture) and reads data from the LTDBs via 48 optical links through MTP-12 connectors (to the right of the picture). The processed data is then transmitted via 48 optical links through an MTP-48 connector to the FEXes. The primary function of the LATOME is to calculate the energy deposited in the detector using ADC values from the LTDB.

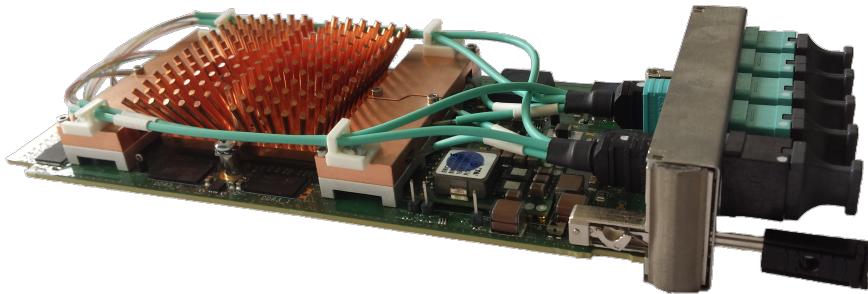


Figure 2: LAr Trigger prOcessing MEzzanine (LATOME) board (Aad et al., 2022)

### 2.1 LATOME Firmware

The LATOME firmware is responsible for four key functions: managing up to 48 high-speed input links from one or more LTDBs, corresponding to 320 Super Cells; applying a digital filtering algorithm to reconstruct the transverse energy of the Super Cells every 25 ns, while also determining the bunch crossing identifier (BCID) to pinpoint the exact time of energy deposition; outputting the reconstructed energy data to the Level-1 calorimeter trigger system (FEX) every 25 ns; and processing and buffering data for delivery to the TDAQ readout chain and local monitoring systems, triggered by events such as a Level-1 Accept or random triggers.

Figure 3 presents a block diagram of the LATOME firmware, highlighting both the main data path and the clock organization within the system. The firmware is built around the Low-Level Interface (LLI), which controls the hardware components. The main data path, from LTDB to FEX, is divided into four blocks: input stage (IS), configurable remapping (REMAP), user code (UC), and output summing (OSUM). In addition to these, the firmware includes: TDAQ/monitoring (MON) for data transfer to TDAQ and local monitoring, Internet Protocol Bus controller (IPBus) for the slow-control interface, and a Timing, Trigger, and Control (TTC) module for decoding and providing signals from the TTC system (Taylor, 1998). The main data path blocks are detailed as follows:

- **Input Stage (IS):** receives and decodes the LTDB data stream at 320MHz, and aligns the data from different fibers to make sure they are synchronized to the same BCID.
- **Configurable Remapping (REMAP):** reorganizes data from the IS received at 320MHz based on the region of the detector for each bunch crossing. It groups Super Cells data into Trigger Towers and sends them to the User Code block at 240MHz. Since the data arrangement varies across the detector, the block is configured for each of the 116 LATOMEs at startup.
- **User Code (UC):** processes Super Cell data from the REMAP block at 240MHz. It synchronously outputs the reconstructed transverse energy and data quality bits to the OSUM block, maintaining a fixed latency for each corresponding bunch crossing.
- **Output Summing (OSUM):** groups the data received from the UC at 240MHz and calculates sums over specific regions of the detector, encode the results and send them to the FEX output fibers at 280MHz.

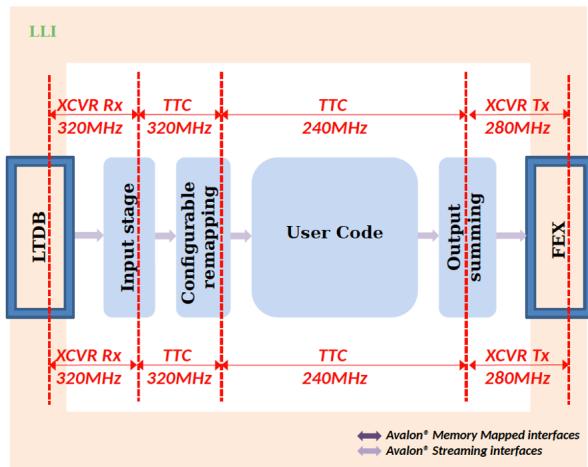


Figure 3: LATOME firmware clock domains diagram.

The latest upgrade to the LATOME firmware simplifies several time-division multiplexing components by replacing them with more straightforward, parallel logic units. For instance, the on-the-fly routing flexibility needed to handle all 116 detector mappings is now implemented in parallel using well-known circuits like switch matrices. Resource sharing is optimized through a case-by-case analysis, where the reduction in area is weighed against any penalties in latency and timing. When area reduction is substantial, resource sharing is automatically implemented using High-Level Synthesis technology. Throughout the process, trade-offs between area, latency, and maximum clock frequency are made to meet project requirements and target hardware specifications, while maintaining the original functionality of the source code.

## 2.2 HLS Implementation in LATOME Firmware

The LATOME firmware is being upgraded via a complete replacement of REMAP and OSUM blocks. Previously, both blocks routed detector information using time-multiplexed interfaces running at three different clock frequencies introducing challenges on the timing

closure step of the design flow. The new approach implemented the REMAP block with a single processing step so-called Input Switch Matrix (ISM) that routes the detector information using a sparse switch matrix with all the multiplexers implemented in parallel. The ISM acts as a routing hub, where the Super Cells values are efficiently directed to their respective data processing lines without the overhead of serial data processing.

The OSUM has been redesigned, transitioning from a serial data processing approach to parallel processing, similar to the improvements made in the REMAP block. The redesigned OSUM block is composed of several key components, all developed using High-Level Synthesis (HLS): (i) *Masking*, used for disabling noisy Super Cells; (ii) *EMEC Adapter*, which distributes the energy of 6 Super Cells into 4 Super Cells while conserving the total energy while addressing particularities of the EMEC calorimeter region (Aad et al., 2022); (iii) *OSM* (Output Switch Matrix), which centralizes and streamlines the routing of processed data, enhancing data routing flexibility; (iv) *eFEX MLE*, responsible for the compression the energy value using a multi-linear encoding scheme; (v) *Frame Encoder*, which encodes the energy values into a predefined data output format; and (vi) *CRC*, which adds a checksum into the output data that are later checked by the FEX system.

### 3. FUNCTIONAL SIMULATION

This section outlines the methodology employed for validating the latest LATOME firmware with HLS blocks before deploying to the underground ATLAS computing room. The validation process emphasizes the use of multiple simulation layers of increasing levels of complexity. Initially, unit tests are used to verify the functionality of each of the blocks separately. Then, functional simulation checks against errors all the HLS blocks connected together still in a single clock domain. These preliminary simulations are followed by a more complete simulation incorporating SerDes (serializers and deserializers) and three clock domains. Subsequently, an entire system-level simulation is performed to evaluate the interactions between the HLS blocks and other components of the firmware. This multi-layered simulation approach provides a thorough validation, identifying and addressing potential issues at each stage to minimize the chance of finding errors during operation.

#### 3.1 Simulation Structure

The simulation structure leverages a coroutine-based cosimulation environment, Cocotb (Cocotb, 2024), in conjunction with QuestaSim (Questa, 2024) for functional simulation and debugging. In this setup, coroutines are employed to drive and monitor signals from the device under test (DUT) and compare them against expected outputs generated by the firmware simulation model.

These asynchronous coroutines receive pseudo-random data from identically seeded independent data generators, ensuring consistency across all checks. In this work, two firmware models were created to generate the expected outputs, which are described in the following section.

The simulation and verification process is divided into layers, each responsible for handling different levels of complexity and ensuring comprehensive validation of the firmware. By simulating 10,000 bunch crossings (BCs) for all 116 detector mappings, the structure allows for the collection of extensive statistical data, summing in total more than 1 million BCs, which is crucial for validating the firmware's performance under a wide range of conditions. In the less

complex layers, such as Layer 0, this information can be obtained much more quickly, enabling efficient debugging and validation early in the development process. Figure 4 presents a general block diagram of the simulation structure for Layers 1, 2 and 3, highlighting how the DUT and coroutines adapt according to the level of complexity being tested. Across all layers, the DUT is consistently simulated using the Questa Advanced Simulator.

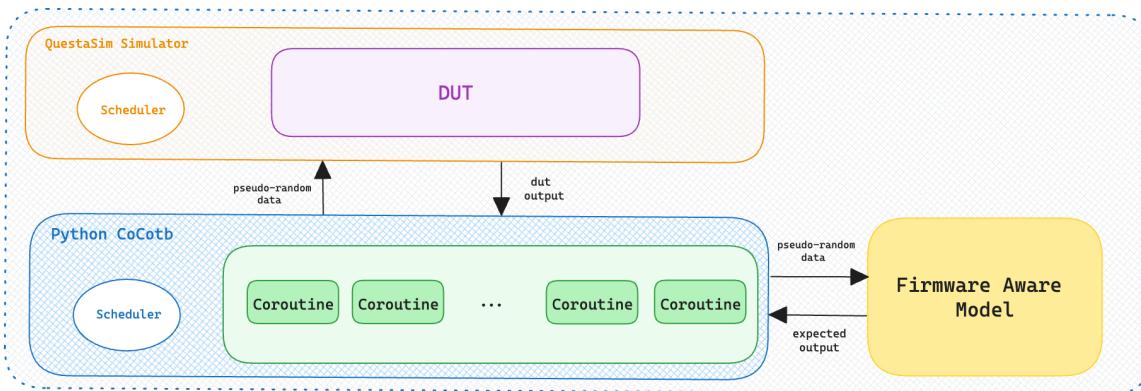


Figure 4: Block diagram of the simulation structure

### 3.2 Firmware Models

In order to create the expected values for comparison with the DUT outputs, two simulation models were developed: the firmware agnostic model and the firmware aware model. Figure 5 provides a block diagram of how the models are cross-checked. The firmware agnostic model uses only the existing input and output configurations of the mapping of the detector to simulate the expected behavior, without detailed knowledge of the internal firmware blocks. This model provides a high-level verification of the overall logic and data paths.

In contrast, the firmware aware model incorporates information from the designed firmware, mimicking the functionality of each hardware block. It includes all internal configurations and processes, excluding the input and output mappings used by the agnostic model. This approach ensures validation of each of the intermediate steps of the internal logic separately enhancing debugging capabilities of the internal logic and detailed configurations, complementing the high-level checks performed by the agnostic model.

Both models are used to validate each other by feeding them the same pseudo-random input and comparing their outputs. Once they are cross-validated, the aware model is utilized in all simulation layers to validate the HLS-generated VHDL code via cocotb. This approach minimizes the chances of propagating errors and ensures robust verification of the firmware.

### 3.3 Simulation Layers

**Layer 0 (L0):** This layer simulates each one of the new blocks, individually, using a C++ testbench. This stage is divided in two types of simulations:

- C++: Performed in software, compiled with gcc. The output from the HLS synthesizable C++ code is compared against the output from the testbench code (non-synthesizable).

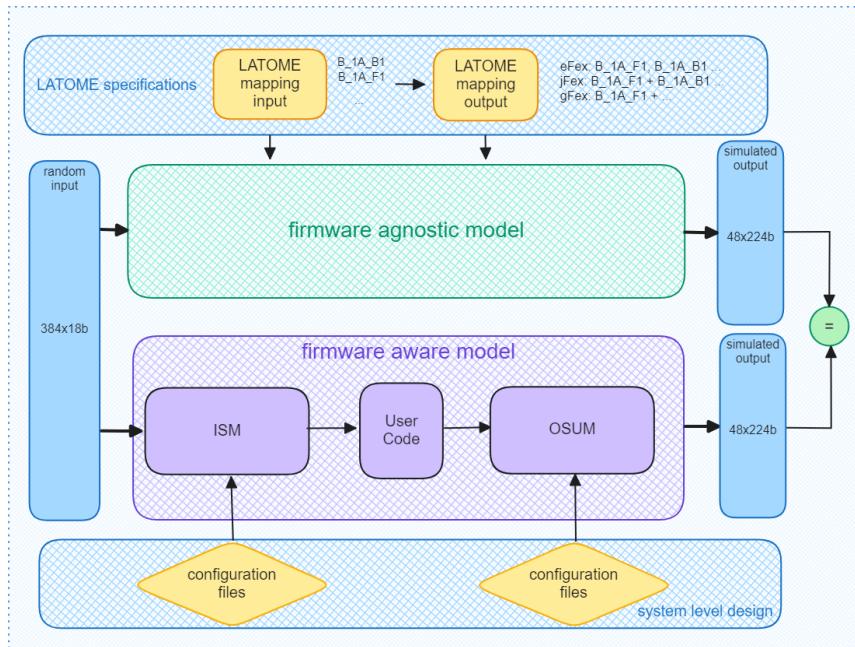


Figure 5: Block diagram of the LATOME's firmware models

- RTL: C++ and VHDL co-simulation using Siemens Questa Advanced Simulator. The C++ testbench interfaces with the HLS-generated VHDL code using an automatically generated SystemC adaptor.

Thanks to Siemens SCVerify technology, the same testbench is used for both simulations. Table 1 shows the simulation running time for most of the HLS submodules. Each of the blocks takes a maximum of 3 seconds to have their functionality verified using the C++ HLS simulation. The C++ and VHDL co-simulation can be hundredes of times slower because their model the generated circuit at the register transfer level.

Table 1: L0 - Time in seconds for 10 000 bunch crossings for 1 detector mapping

	<b>ISM</b>	<b>CRC</b>	<b>EMEC Adapter</b>	<b>OSM</b>	<b>eFEX MLE</b>
<b>C++</b>	2,1309	1,5292	0,1735	0,1108	0,0054
<b>RTL</b>	62,3421	19,2092	1,7950	61,3009	0,6379

**Layer 1 (L1):** The goal of this layer of simulation is to validate the functionality of the High-Level Synthesis (HLS) generated blocks, highlighted in blue in Figure 6. This initial phase aims to identify and resolve any bugs related to the HLS code. The focus is on simulating the Input Switch Matrix (ISM) and Output Summing (OSUM) with parallel interfaces to ensure that the logic and data paths are correctly implemented. These two blocks were written using C++ and subsequently synthesized to RTL using the Catapult HLS tool. Simulations are performed within a single clock domain, operating at 240 MHz. Figure 6 provides a block diagram of the DUT used in this simulation. On the left, the ISM handles the remapping function, while the OSUM manages the output summing function. This approach ensures that the functionality of the firmware is checked against errors before advancing to more complex layers.

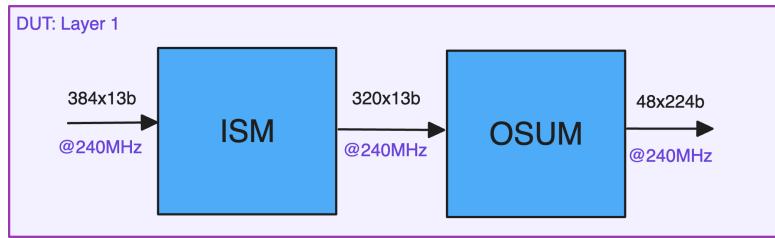


Figure 6: Block diagram of Layer 1 - single clock domain

**Layer 2 (L2):** Figure 7 illustrates how Layer 2 builds upon the foundation established in Layer 1 by incorporating time division multiplexing a feature that is required to interface the new blocks to the existing ones. This is achieved by adding serializers and deserializers to the DUT, alongside the ISM and OSUM HLS-generated blocks. The objective of this layer is to validate the SerDes blocks and their interaction with the HLS modules, ensuring clock synchronization and data integrity across the three different clock domains. This layer manages four conversions between parallel and serial data formats, with two involving data transfer between related clock frequencies. It manages conversions between parallel and serial data formats, converting from 320 MHz serial to 240 MHz parallel before the ISM, and from 240 MHz parallel to 280 MHz serial after the OSUM, ensuring clock synchronization and data integrity. These steps enable the new firmware architecture while minimizing changes to existing blocks, which remain intact. The layer also verifies that these conversions do not introduce errors in the data processed by the High-Level Synthesis (HLS) blocks. Additionally, this layer introduces a block called the user code bypass. This block mimics the actual user code when its FIR coefficients are configured to produce a unitary output, i.e. bypassing all the user code input to their respective outputs.

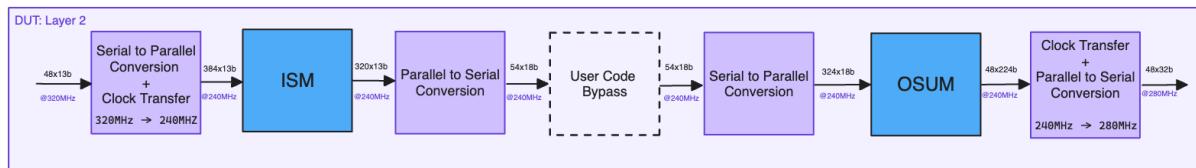


Figure 7: Block diagram of Layer 2 - multiple clock domains

**Layer 3 (L3) - System-Level Simulation:** The final layer handles the complete LATOME system-level simulation and validation, covering both firmware and software components. This stage aims to test the entire system, including the TTC block, IPBus control, Monitoring, Input Stage, and LLI, as shown in Figure 8, in addition to the components tested in Layers 0, 1 and 2. Layer 3 tests a DUT which integrates all elements validating their interactions and ensuring overall system reliability and performance.

Layer 3 utilizes a DUT that instantiates the actual User Code instead of a user code bypass. The User Code is configured to pass data from the REMAP stage to the OSUM stage without any manipulation, ensuring that the output of the REMAP is the input of the OSUM. The configuration applies FIR filter coefficients for unity processing (input = output), sets the pedestal value to zero, activates fixed-point calculations, and turns off the baseline correction.

Additionally, Layer 3 tests part of the software infrastructure used in operation. Testing low-level software as part of the FPGA functional simulation is not often implemented as part

of the verification flow and it is considered one of the novel contributions implemented as part of this work. This includes scripts capable of controlling and monitoring the registers for both the REMAP and OSUM blocks as well as the other blocks included. This allows reproducible system verification without the need for physical hardware, which leads to a rapid and efficient firmware design and verification process.

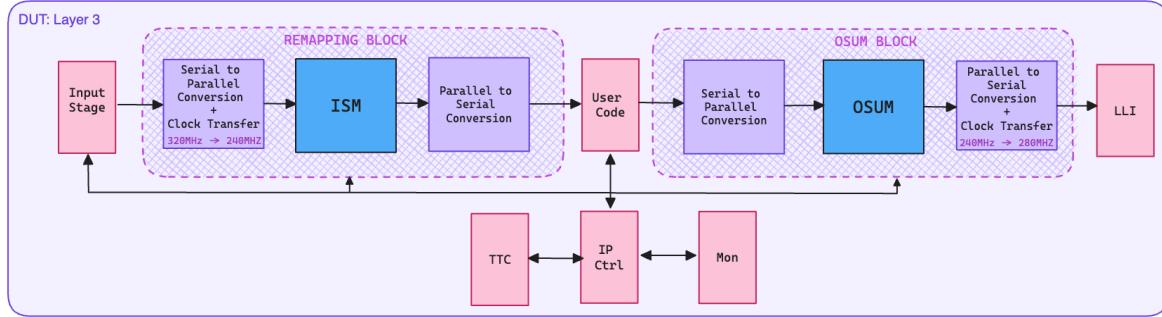


Figure 8: Block Diagram of Layer 3 - Complete Infrastructure

By following this structured approach across three layers of simulation, we ensure that the upgraded firmware is thoroughly validated and integrated, ready to meet the stringent requirements of the ATLAS Liquid Argon Calorimeters in the high-luminosity LHC environment.

Table 2 summarizes the simulation times for Layers 1, 2, and 3 to process 10,000 bunch crossings for detector mapping. Layers 1 and 2 have similar simulation duration's, each taking approximately 32 minutes. In contrast, Layer 3, which involves a full system-level integration and validation, requires significantly more time—around 75 minutes. This extended duration is a reflection of the increased complexity and the comprehensive nature of the system-level testing necessary to ensure that the firmware meets the performance requirements in ATLAS.

Table 2: L1, L2, and L3 Simulation time in minutes for 10,000 bunch crossings for 1 detector mapping

	Simulation Time (min)
<b>L1</b>	32
<b>L2</b>	32
<b>L3</b>	75

#### 4. CONCLUSIONS

The decision to divide the simulation process into multiple layers has enhanced our ability to isolate issues early in the design cycle, thereby accelerating the overall development process. The upgrade to the LATOME firmware, which involved integrating new features and enhancing performance, was rigorously tested using this multi-layered simulation strategy. By implementing stand-alone simulations in Layer 0, we were able to rapidly accumulate statistics which helped us validate the firmware within a short period of time.

The simulation times across the different layers demonstrated the efficiency of the approach presented in this work. Layers 1 and 2, focusing on the validation of interconnected blocks,

each required approximately 32 minutes to process 10,000 bunch crossings for one mapping of the detector. However, the more comprehensive system-level simulation in Layer 3 took significantly longer—around 75 minutes—reflecting the added complexity.

A key advantage of using High-Level Synthesis (HLS) in this process is the potential for simulations to be hundreds of times faster than traditional RTL simulations. This significant speedup allowed for more iterations and extensive testing within a shorter time frame, contributing to the overall success of the firmware development.

Currently, the upgraded firmware is undergoing testing at Point 1, the location of the ATLAS detector at CERN, where it is integrated with the hardware and operated under real-world conditions. To date, no firmware issues have been detected during these tests, thanks to the effectiveness of our multi-layered simulation approach, which successfully identified and resolved potential problems throughout the development process. This successful outcome underscores the strength of our simulation strategy in preparing the firmware for the demanding environment of the ATLAS experiment.

### **Acknowledgements**

This work was supported by CNPq. We would also like to extend our gratitude to the ATLAS Experiment, especially the Liquid Argon (LAr) Calorimeter team, for their support in the development of this work.

## **REFERENCES**

- ALICE Collaboration. (2008), “The ALICE experiment at the CERN LHC”, *Journal of Instrumentation*, IOP Publishing, vol. 3, no. 08, p. S08002.
- Aad, G., et al. (2022), “The Phase-I trigger readout electronics upgrade of the ATLAS Liquid Argon calorimeters”, *Journal of Instrumentation*, vol. 17, May 2022, P05024. DOI: 10.1088/1748-0221/17/05/P05024.
- ATLAS Collaboration. (2008), “The ATLAS experiment at the CERN Large Hadron Collider”, *Journal of Instrumentation*, vol. 3, no. 08, p. S08003.
- Barletta, W., Battaglia, M., Klute, M., Mangano, M., Prestemon, S., Rossi, L., and Skands, P. (2014), “Future hadron colliders: From physics perspectives to technology R&D”, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 764, pp. 352–368.
- Brüning, O., and Rossi, L. (2019), “The high-luminosity large hadron collider”, *Nature Reviews Physics*, vol. 1, no. 4, pp. 241–243.
- Cocotb contributors. (2024), “cocotb: Coroutine-based co-simulation testbench for verifying RTL”, *cocotb*. Available at: <https://www.cocotb.org/>.
- CMS Collaboration. (2008), “The CMS experiment at the CERN LHC”, *JInst*, Citeseer, vol. 3, no. 08, p. S08004.
- Evans, L., and Bryant, P. (2008), “LHC Machine”, *Journal of Instrumentation*, IOP Publishing, vol. 3, no. 08, p. S08001.
- LHCb Collaboration. (2008), “The LHCb detector at the LHC”, *Journal of Instrumentation*, IOP Publishing, vol. 3, no. 08, p. S08005.
- Perkins, D.H. (2000), “Introduction to High Energy Physics”, *Cambridge University Press*.
- Siemens EDA. (2024), “Questa Advanced Simulator”, *Siemens EDA*. Available at: <https://eda.sw.siemens.com/en-US/ic/questa/simulation/advanced-simulator/>.
- Taylor, B.G. (1998), “TTC distribution for LHC detectors”, *IEEE Transactions on Nuclear Science*, vol. 45, no. 3, pp. 821–828. Available at: <https://doi.org/10.1109/23.682644>.