

## Implementation of Neural Networks on FPGAs with Memristors for Optimizing Data Processing in ATLAS

D. FIACCO ON BEHALF OF THE ATLAS COLLABORATION

*Dipartimento di Fisica, Università di Roma La Sapienza & INFN, Sezione di Roma*

**Summary.** — The High Luminosity LHC (HL-LHC) upgrade [1] will increase the instantaneous luminosity of the LHC by an order of magnitude, posing new challenges for the ATLAS [2] trigger and data acquisition systems. To cope with elevated particle flux and background, the ATLAS muon spectrometer is undergoing major Phase-II upgrades [3], including new detector layers and a redesigned Level-0 (L0) trigger architecture capable of running advanced algorithms on FPGAs. Among the key algorithmic developments is muon momentum regression at the trigger level, aimed at reducing false triggers from low- $p_T$  muons. The primary focus of this work is on the exploration of hardware technologies that can accelerate such inference tasks. In particular, we investigate the use of memristors [4] as analog, in-memory computing elements capable of performing highly parallel matrix-vector operations with extreme energy efficiency. We outline the principles and potential advantages of memristor-based acceleration in the context of trigger-level inference and present a conceptual integration into the ATLAS DAQ architecture. Finally, we report experimental results using Known self-directed channel memristors [5], demonstrating the ability to iteratively program their conductance to target values with a few-percent precision, a critical requirement for storing neural network weights reliably. These findings support the feasibility of hybrid analog-digital architectures for future trigger systems in high-energy physics.

### 1. – Introduction: HL-LHC and ATLAS Muon Spectrometer Upgrades

The high-luminosity phase of LHC [8] operations (HL-LHC), will feature a large increase in simultaneous proton-proton interactions per bunch crossing up to 200, compared with a typical leveling target of 64 in Run 3. This increase in collision rate and density of overlapping events (pile-up) will deliver unprecedented integrated luminosity for physics but also imposes extreme requirements on the detectors and trigger systems. To cope with the high event rates, the ATLAS experiment [2] is implementing a Phase-II upgrade of all its sub-detectors and trigger/data-acquisition (TDAQ) systems. In the muon spectrometer, hardware improvements—such as new endcap stations and an additional barrel layer of thin-gap Resistive Plate Chambers (RPCs), and the replacement of some of the Monitored Drift Tube (MDT) with small-diameter MDT—aim to maintain performance

under high particle flux [3]. Along with detector hardware upgrades, the trigger architecture will be significantly enhanced. For HL-LHC, ATLAS will deploy a new first-level hardware trigger (often called Level-0, or L0) that sends full-granularity muon hit data off-detector to modern FPGA-based trigger processors [10]. This upgrade allows more sophisticated trigger algorithms to run with a latency budget of up to  $\sim 10 \mu\text{s}$  before triggering a readout. The L0 muon trigger will incorporate inputs from the new inner barrel RPC layer and improved endcap stations, and it will operate at the full 40 MHz collision rate, accepting events at up to 1 MHz to be passed to the high-level trigger. The large, high-speed FPGAs in the trigger processors provide the flexibility to implement advanced algorithms, including machine-learning-based methods, to enhance muon identification and momentum measurement in real time.

## 2. – Muon Momentum Regression Problem in the Trigger

The task of quickly estimating a muon’s momentum from limited detector information is critical for the trigger system. In the current ATLAS barrel, the first hardware level of trigger logic uses the pattern of RPC hits in multiple layers to assign a muon candidate to a  $p_T$  threshold. The conventional trigger algorithm implements a coincidence window scheme [9]: a hit in the intermediate RPC layer seeds a road; the algorithm looks for corresponding hits in the subsequent layers within a narrow window that is inversely related to the  $p_T$  threshold (higher  $p_T$  muons bend less in the magnetic field, so their hits align within a smaller window). With the harsher HL-LHC environment the muon trigger upgrade not only adds new detector layers but also calls for more sophisticated trigger algorithms that can estimate the muon’s  $p_T$  (or equivalently its curvature in the magnetic field) with improved precision using the available trigger detector data.

## 3. – Neural Networks approach and their FPGA Implementation in the Trigger

The L0 trigger must rely only on fast detectors (e.g. RPCs), and the momentum regression problem thus becomes a pattern recognition task based on a handful of discrete detector hits. Traditional algorithms struggle to optimally exploit this information, especially in presence of multiple hits (from multiple muons or noise) and complex detector geometry. This motivates the exploration of machine learning approaches that can learn the mapping from trigger hit patterns to muon  $p_T$  from simulation data. E.g., one approach treats the trigger detector data as a coarse “image” and employs a Convolutional Neural Network CNNs model the spatial pattern of RPC hits as binary images, effectively learning local and global correlations to regress  $p_T$  [6]. Regardless of the specific neural network architecture, implementing them on FPGAs for the Level-0 trigger requires meeting stringent timing and resource constraints. These models are made hardware-efficient using quantization and compression techniques [11]. The `hls4ml` framework [12] facilitates conversion from trained models to FPGA firmware, allowing inference within hundreds of nanoseconds and fitting within available LUT and DSP resources.

## 4. – Memristor Technology: Analog Compute-in-Memory for Neural Networks

While FPGAs and ASICs provide powerful digital platforms for trigger algorithms, memristors [4] — a class of emerging, non-volatile analog devices — offer a promising complementary technology for accelerating neural inference. The memristor (short for

“memory resistor”), first theorized by Chua in 1971, is a two-terminal device whose conductance reflects the integral history of applied voltage or current. Crucially, memristors retain their resistance state without power and can be programmed to a continuum of analog values, making them ideal for representing neural network weights directly in hardware. Fabricated in dense crossbar arrays, memristors enable analog in-memory computing: input voltages applied to rows result in output currents on columns via Ohm’s and Kirchhoff’s laws, effectively computing a matrix-vector multiplication in a single step. As shown in left side of Figure 1, each device  $G_{ij}$  contributes  $I_j = V_i G_{ij}$ , inherently realizing a weighted sum. Memristors offer several compelling advantages, including non-volatile weight storage, which eliminates repeated memory fetches; analog computation, allowing massively parallel and energy-efficient multiply-accumulate (MAC) operations; and high integration density, with feature sizes down to tens of nanometers and potential for 3D stacking. Studies report memristive MACs operating at femtojoules per operation—orders of magnitude lower than CMOS-based logic (e.g., an analog ReRAM-based accelerator achieves  $\sim 11$  fJ/MAC, 270 times better than digital ReRAM and 430 times better than SRAM) [13]. However, the technology faces critical challenges. Memristors are still under development and lack standardization. Various materials are under investigation (e.g., oxide-based ReRAM, PCM, CBRAM), each with unique trade-offs. Devices often suffer from conductance drift, programming variability, and susceptibility to temperature and noise. Furthermore, their radiation tolerance remains inadequately characterized for high-energy physics environments like ATLAS. For reliable inference, weights must be precise and stable over time, which is non-trivial with analog devices. Calibration protocols, hybrid digital-analog schemes, and variation-aware training are active research areas addressing these concerns [14].

## 5. – Integration of Memristors into the ATLAS DAQ System

Considering the potential of memristor crossbar arrays to accelerate matrix operations, it is intriguing to imagine their integration into the ATLAS Level-0 trigger and DAQ chain. A proposed architecture involves a hybrid board within an ATCA crate, combining FPGAs for control with a memristor chip for analog matrix-vector multiplication (m-MM). FPGAs would digitize inputs (e.g., muon segment parameters), convert them to analog voltages via DACs, apply them to the memristor array, and then read back currents (weighted sums) via ADCs, as described by the schema on the right side of the Figure 1.

Given that analog computation is nearly instantaneous, the main latency arises from I/O conversions. For compact networks, inference within microseconds is achievable. Energy consumption would also scale favorably with network size, especially compared to FPGA DSPs. Such accelerators could be deployed per sector or trigger tower, leveraging memristors’ inherent parallelism. Still, to ensure robustness, periodic recalibration may be required (e.g., between LHC fills), using known test vectors. Built-in FPGA checks could monitor output deviations and disable malfunctioning regions to avoid incorrect trigger decisions.

Substantial R&D is needed to ensure long-term stability, radiation tolerance, and system-level integration. Nonetheless, if these hurdles are addressed, memristor-based inference could significantly enhance the efficiency and scalability of real-time processing in ATLAS.

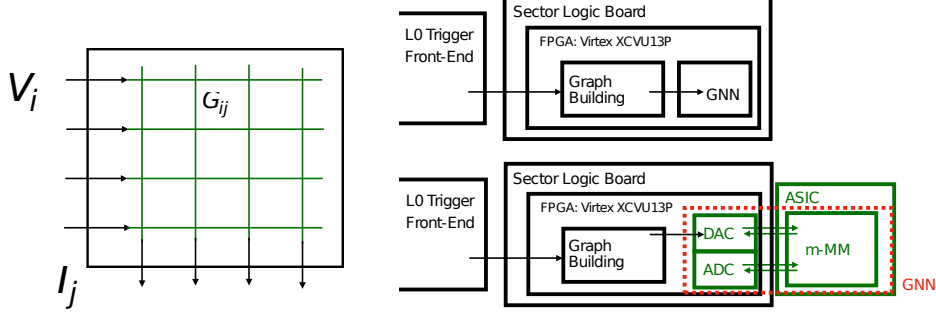


Fig. 1: In the left side there is a schematic of a memristor crossbar array performing an  $M \times N$  matrix multiplication. Input voltages  $V_i$  are applied to word lines (rows), and output currents  $I_j$  are measured at bit lines (columns). Each crosspoint has a memristor of conductance  $G_{ij}$ , so the current through that device is  $I_{ij} = V_i G_{ij}$ . By Kirchhoff's Law, the column current  $I_j = \sum_i V_i G_{ij}$ , computing the weighted sum of inputs for output  $j$  in a single step. The right side of the figure gives a schematic of a possible integration of the memristor crossbar matrix (m-MM) in the hipotesis of an implemented GNN in the FPGAs of the L0 trigger of the Muon Spectrometer of ATLAS. The FPGA would digitize the relevant inputs (e.g. a set of preprocessed hit coordinates or segment parameters), convert them to analog voltages via DACs, and apply them to the memristor array input lines. The currents summed at the output lines (representing weighted sums) would be converted back to digital via ADCs and fed to the FPGA logic, which would apply activation functions or further processing.

## 6. – Memristor Programming Experiments and Results

To enable neural inference with memristors, each device must be programmed to a target conductance corresponding to a trained network weight. Due to inherent variability and non-ideal switching, a simple one-shot voltage pulse is usually insufficient. Instead, an iterative *read-program-verify* loop is typically used [5].

We performed programming experiments using  $1 \times 16$  matrix of self-directed channel (SCD) memristors from Knowm Inc., which offer analog conductance tuning. Devices (a few microns wide) were programmed to a range of target values across the dynamic range. Each cycle involved: (1) reading current conductance  $G$ , (2) comparing with the target  $G^*$ , (3) applying incremental set or reset pulses as needed, and (4) re-reading and iterating until within an acceptable error or max iterations. Substantial device-to-device variability was observed. Certain memristors exhibited a threshold-like response, saturating near  $80 \mu S$ , while others were capable of reaching values as high as  $200 \mu S$ . After a few initial “forming” cycles, 4 memristors within the 16 with the more similar range of conductance variability were selected.

On the 4 selected memristors we tested an adaptive programming scheme proportional to the error  $G^* - G$  and typically in 5–15 pulses the target range has been achieved for each of them, as shown in Table I. The Figure 2 shows on the left the subdivision in 8 WPs of the conductance range, achieving the goal to use each memristor as a 3-bit object and using as working point definition the interval given by  $\pm 10\%$  of  $G^*$  (left) and the

TABLE I.: The table reports the target conductance values  $G^*$  for the four selected Known memristors, alongside the final programmed values obtained after the iterative tuning loop. Each  $G^*$  was reached within 5–15 iterations using an adaptive programming strategy. The programmed conductances remained stable within the defined range for at least 5 minutes. However, after 20 minutes, some devices exhibited a measurable decrease in conductance, indicating possible retention degradation that requires further investigation.

Memristor Label	Target $G^*$ WP	Set $G$ WP	$G$ WP after 5 min	$G$ WP after 20 min
A	7	7	7	7
B	5	5	5	5
C	6	6	6	4
D	3	3	3	1

programming loop evolution of the memristor C toward the target WP 6 (right).

Short-term retention was stable over 5-minute intervals at room temperature, though longer-term tests are ongoing. Potential drift or environmental sensitivity may require occasional recalibration or refresh.

Programming time per memristor was in the millisecond range due to long pulses and read delays. However, massive parallelism is feasible in crossbar arrays, allowing simultaneous programming of entire rows or columns. In practice, weight programming would occur only during network deployment or periodic recalibration (e.g., between LHC fills).

## 7. – Conclusion

The HL-LHC presents formidable challenges in detector readout and real-time data processing. To sustain high performance under extreme pile-up and background conditions, the ATLAS muon spectrometer is being enhanced with new detector layers and a

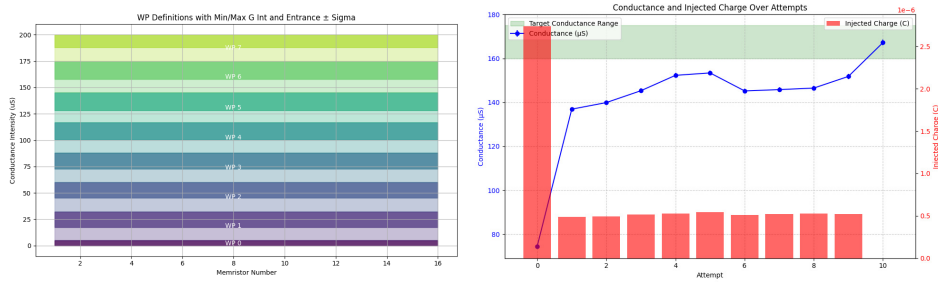


Fig. 2: In the left side are represented the 8 defined WPs in the available conductance for the selected memristors. The intervals achieve the goal to use each memristor as a 3-bit object, using as working point definition the interval given by  $\pm 10\%$  of  $G^*$ . The right side of the figure shows the conductance and the estimated charge injection of the memristor C in function of the programming loop index.

Level-0 trigger capable of executing advanced algorithms on FPGA hardware. AI-based methods can be deployed to perform fast and accurate muon momentum regression, significantly reducing fake triggers from low- $p_T$  backgrounds while maintaining sensitivity to high- $p_T$  physics.

Looking further ahead, analog in-memory computing using memristor crossbar arrays offers a compelling long-term strategy to complement digital inference. Memristors promise ultra-efficient matrix-vector operations with high density and non-volatility, enabling potentially lower-latency, energy-saving neural inference directly at the trigger level. Our experimental studies with Knowm memristors demonstrated the feasibility of programming conductance values with a 10% percent precision using adaptive iterative strategies—an essential step toward using these devices as analog weight stores in neural networks.

Although memristors are not yet ready for deployment in the 2027–2029 ATLAS Phase-II trigger system, continued research in this direction could enable hybrid digital-analog solutions for future HEP experiments. Such architectures could push the frontiers of trigger processing, paving the way for next-generation accelerators beyond HL-LHC.

\* \* \*

The author acknowledges V. Bocci, G. Gustavino, F. Iacoangeli, V. Ippolito and S. Giagu for their help. This work was partially funded by the PRIN project 2022XXMTCH

## REFERENCES

- [1] O. ABERLE ET AL., *High-Luminosity Large Hadron Collider (HL-LHC): Technical design report*, CERN-2020-010, CERN, Geneva (2020), DOI:10.23731/CYRM-2020-0010
- [2] THE ATLAS COLLABORATION, *The ATLAS Experiment at the CERN Large Hadron Collider*, JINST 3 (2008) S08003;
- [3] EVANGELOS N. GAZIS, *ATLAS Muon Spectrometer Upgrade for the HL-LHC Era's Challenges*, Symmetry 2024, 16(8), 1035;
- [4] L. CHUA, *Memristor-The missing circuit element*, IEEE Transactions on Circuit Theory, vol. 18, no. 5, pp. 507-519, September 1971, doi: 10.1109/TCT.1971.1083337
- [5] KRISTY A. CAMPBELL, *Self-directed channel memristor for high temperature operation*, Microelectronics Journal vol. 59, pp. 10-14 (2017)
- [6] FRANCESCATO, S., GIAGU, S., RITI, F. ET AL., *Model compression and simplification pipelines for fast deep neural network inference in FPGAs in HEP*, Eur. Phys. J. C 81, 969 (2021)
- [7] A. ELABD ET AL., *Graph Neural Networks for Charged Particle Tracking on FPGAs*, Front. Big Data 5, 828666 (2022).
- [8] LYNDON EVANS and PHILIP BRYANT, *LHC machine*, JINST 3 (2008) S08001;
- [9] THE ATLAS COLLABORATION, *Performance of the ATLAS muon triggers in Run 2*, JINST 15(2020)P09015
- [10] STEFANO GIAGU, ON BEHALF OF THE ATLAS COLLABORATION, *Fast and resource-efficient Deep Neural Network on FPGA for the Phase-II Level-0 muon barrel trigger of the ATLAS experiment*, EPJ Web of Conferences 245, 01021 (2020)
- [11] R. OSPANOV ET AL., *Development of a resource-efficient FPGA-based neural network regression model for the ATLAS muon trigger upgrades*, Eur. Phys. J. C 82, 576 (2022)
- [12] J. DUARTE ET AL., *Fast inference of deep neural networks in FPGAs for particle physics*, JINST 13 (2018) no.07, P07027.
- [13] J. MARINELLA ET AL., *Multiscale Co-Design Analysis of Energy, Latency, Area, and Accuracy of a ReRAM Analog Neural Training Accelerator*, doi: 10.1109/JETCAS.2018.2796379

- [14] AGUIRRE, F., SEBASTIAN, A., LE GALLO, M. ET AL., *Hardware implementation of memristor-based artificial neural networks*, Nat Commun 15, 1974 (2024).