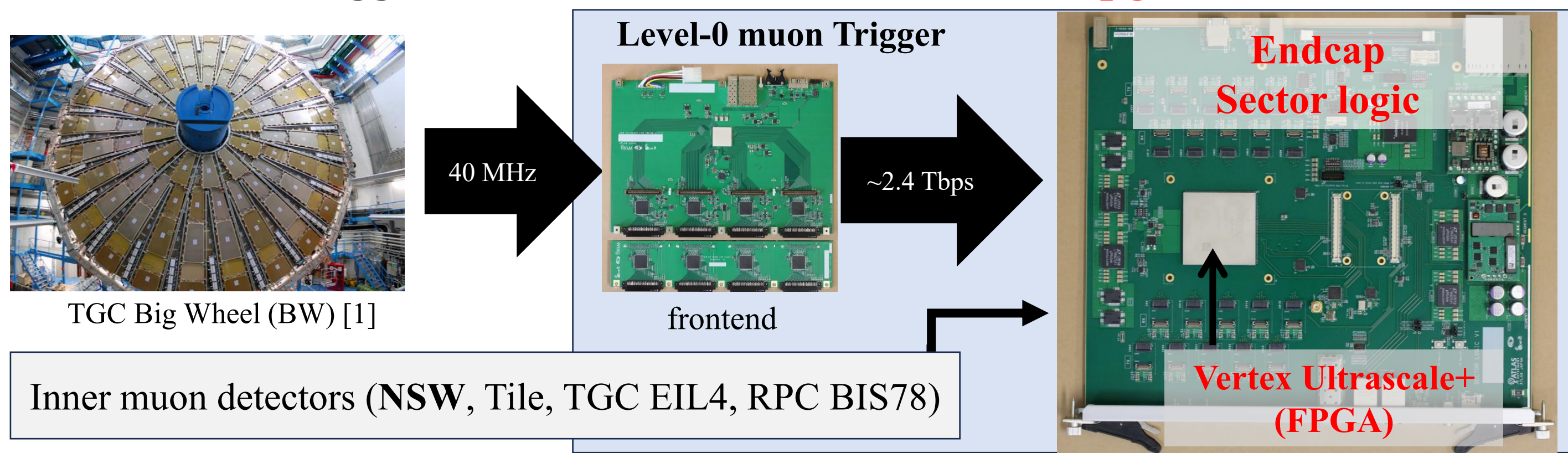


Abstract

This poster presents the design and development status of the ATLAS HL-LHC Level-0 Endcap muon trigger firmware. Muon candidates are reconstructed from TGC hits by firmware implemented on an ATCA blade with an XCVU13P FPGA, applying an inner detector coincidence to reject fake candidates. The Inner Coincidence logic were implemented and validated with detector inputs, and output consistency was confirmed with extensive test patterns. A verification system based on FPGA accelerator cards with similar resources was also developed to enable efficient trigger-logic development. This poster covers the Inner Coincidence concept, design, validation, resource usage, and the accelerator-card-based verification system.

Introduction – Thin Gap Chamber (TGC)

- Provides the Endcap muon trigger with detector hits.
- ~320k readout channels with 25 ns bunch crossing (40 MHz).
- Phase-II (Run 4/HL-LHC) upgrades deliver full-granularity data to the Level-0 trigger.
- **The entire trigger and readout electronics will be upgraded.**



Endcap-Sector Logic (SL)

- An ATCA blade with a large FPGA (Virtex UltraScale+ XCVU13P-class) **executing the L0 Endcap muon trigger.**
- The FPGA runs the trigger logic and hit data readout.
- Receives TGC BW and inner-detector hits and runs TGC BW candidate reconstruction and Inner Coincidence in firmware.
- The firmware for reconstructing muons entering the TGC BW has already been completed.
- **Development of inner coincidence** acquisition system are required.
- The logic must be validated to meet our required specifications.

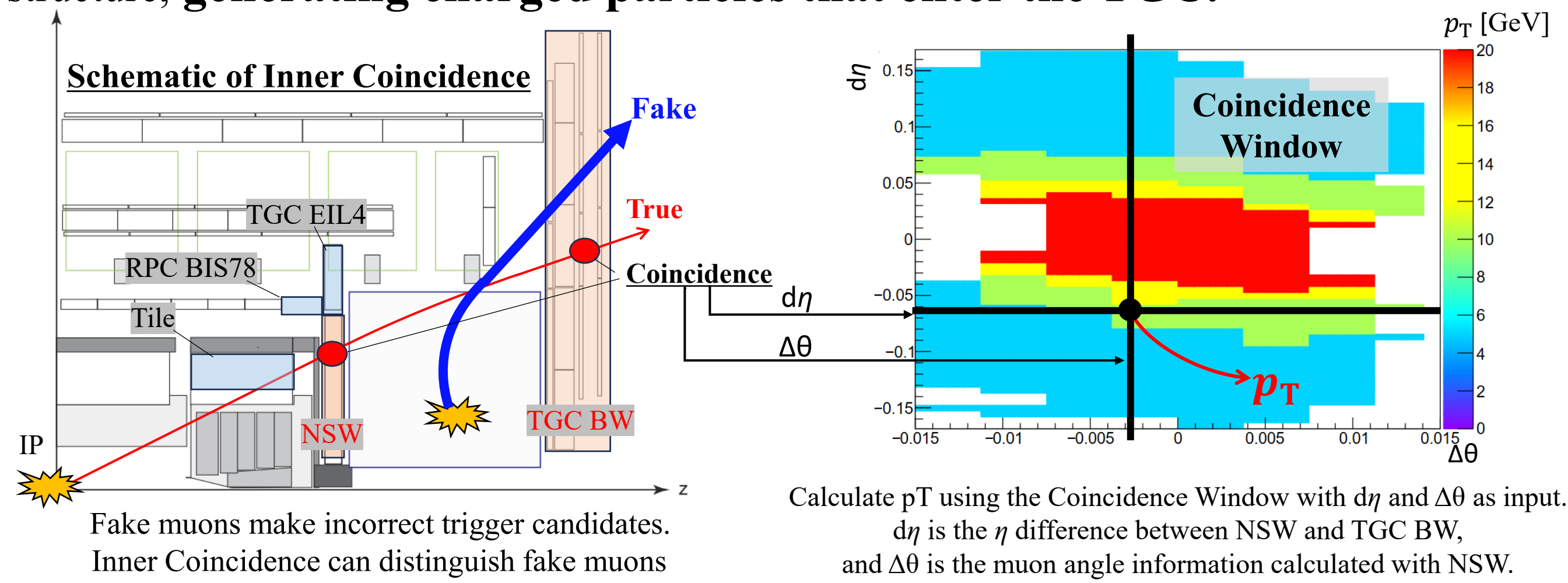
p_T estimate and fake muon

Muon reconstruction with TGC

- TGC trigger candidates are built by taking coincidences across multiple TGC stations/layers in the η and ϕ directions (wire/strip hits).
- From the reconstructed bending in the magnetic field, the candidate p_T is estimated using a precomputed lookup table.

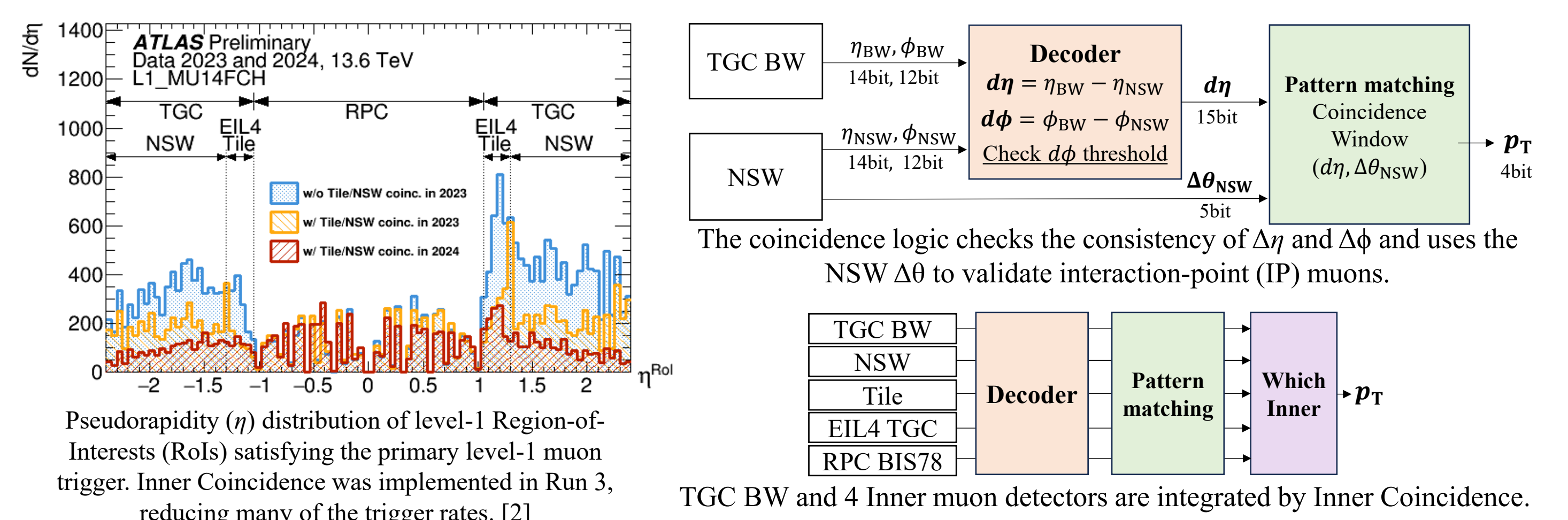
Fake muon

However, neutral particles from the beam pipe cause scattering with the detector's internal structure, **generating charged particles that enter the TGC.**



Inner Coincidence

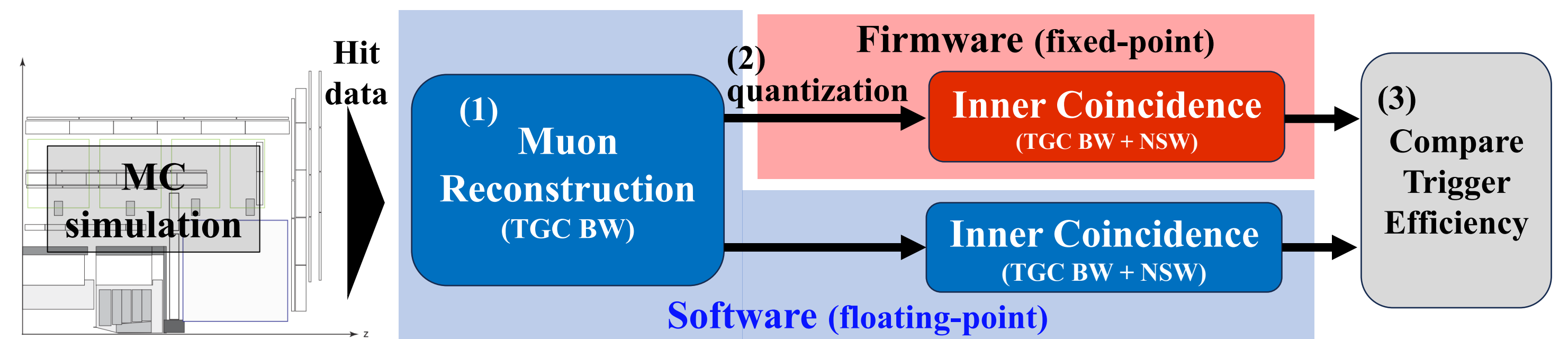
- The Inner Coincidence logic **takes coincidences between the detector inside the magnetic field and the TGC BW.**
- This logic can determine whether muon candidates passing through the TGC originate from the interaction point or are fake muons.
- **In Phase-II (Run 4), we implement a new L0 inner coincidence logic compatible with the upgraded electronics and interfaces.**



Firmware Validation

We **validated the firmware** by comparing its outputs with a floating-point software reference. Consistency was validated by comparing the performance.

- (1) MC events are processed by the floating-point simulator to produce muon position information ($\eta/\phi/\Delta\theta$) for trigger inputs.
- (2) Trigger inputs are quantized and input to the firmware.
- (3) The output results are compared for validation.



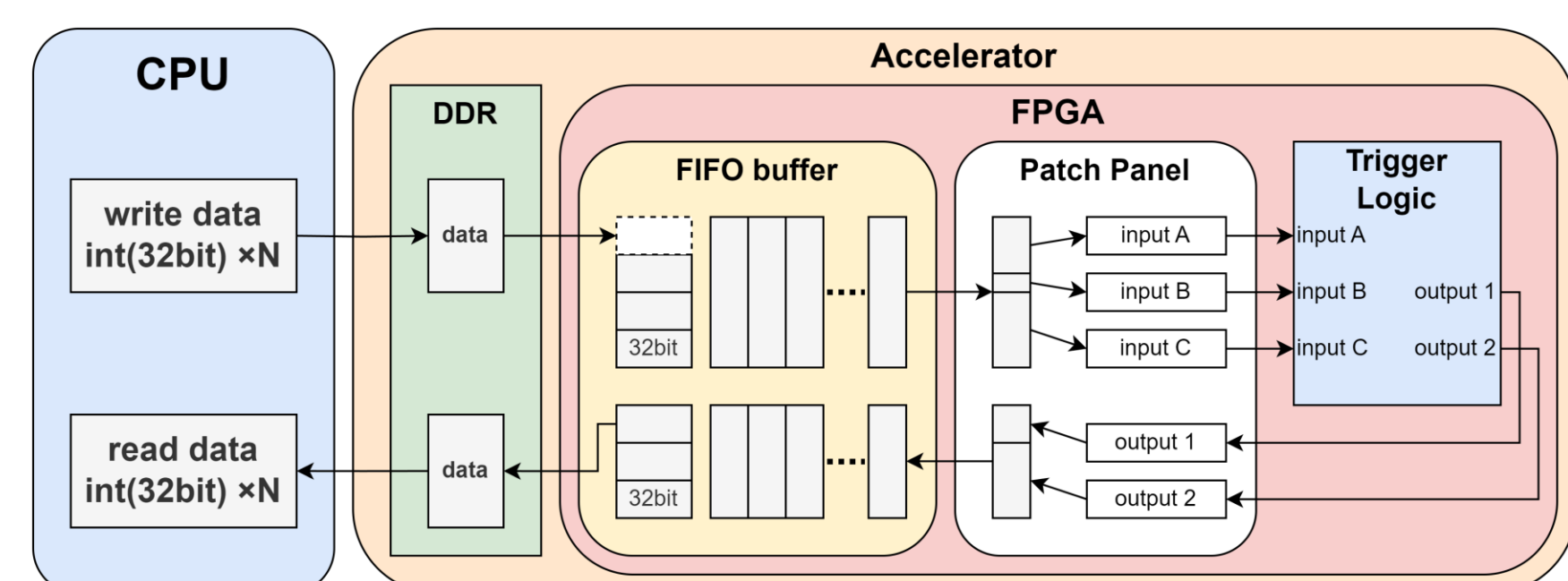
Accelerator-based firmware validation method

We developed a new trigger logic validation system using FPGA accelerators. [3]

- **Problem:** Validating the firmware on the ATCA blade is costly and time-consuming to build and operate.
- **Approach:** Use a PCIe FPGA accelerator card equipped with an UltraScale+ device to run the same firmware and perform hardware-level validation.
- **Performance:** **2.2 ms per validation cycle** (input → output readout → comparison).
- Provides a practical, low-cost, and quickly deployable platform for event-by-event regression tests.



FPGA accelerator connected to PC via PCIe (Alveo 200 ©AMD)



Generate input data on the CPU, send it to the accelerator DDR, feed it to the trigger logic via AXI/FIFO and patch-panel routing, then compute and return the results to the CPU.

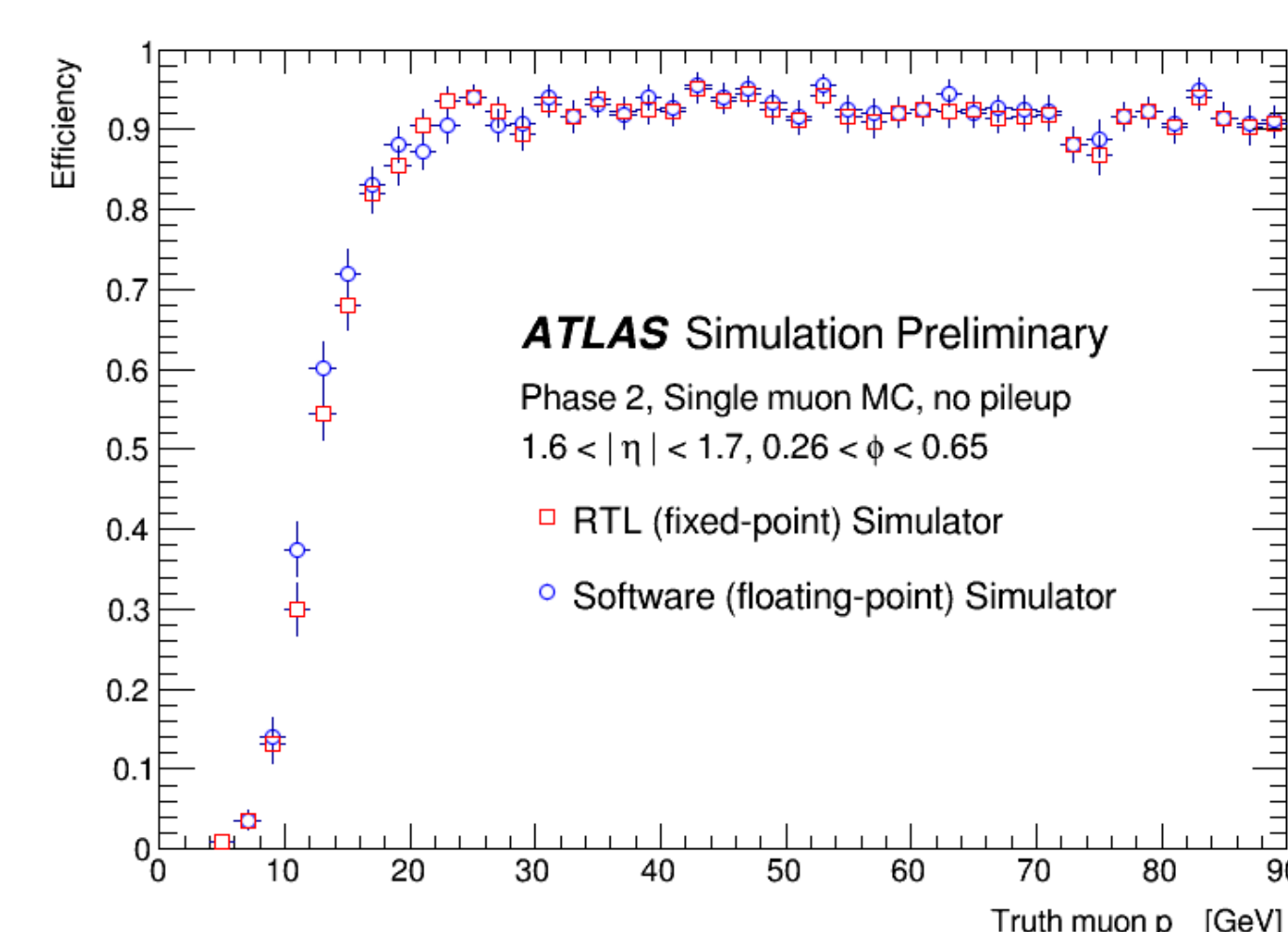
Conclusion

- We developed a new inner coincidence logic for Run 4 and beyond, when the LHC will be increased in luminosity.
- The output from the inner coincidence logic was consistent with the software's floating-point simulator.
- We developed a verification system using an FPGA accelerator. This system makes it possible to run firmware on hardware without the need for a physical machine.

Result

The firmware reproduces the floating-point reference outputs for the tested patterns, **demonstrating a correct implementation of the intended logic.**

Small differences around thresholds can arise from fixed-point quantization of $\eta/\phi/\Delta\theta$.



This figure shows the overall trigger efficiency, including TGC BW candidate reconstruction and the Inner Coincidence, as a function of the truth muon transverse momentum p_T , evaluated with single muon MC simulation. The p_T threshold is 20 GeV. [4]

Resources and timing constraints

This shows FPGA resource usage before and after implementing Inner Coincidence.

Resource usage is within acceptable limits, with ample free space and room for logic expansion.

before	after	LUT	Register
		56%	31%
		BRAM	URAM
		22%	44%

Setup	Hold
Worst Negative Slack (WNS): 0.000 ns	Worst Hold Slack (WHS): 0.001 ns
Total Negative Slack (TNS): 0.000 ns	Total Hold Slack (THS): 0.000 ns
Number of Falling Endpoints: 0	Number of Falling Endpoints: 0
Total Number of Endpoints: 1794556	Total Number of Endpoints: 1766081

List of timing constraints after implementing Inner Coincidence. **No timing violations occurred.**

References

- [1] ATLAS Collaboration, Installation of the first of the big wheels of the ATLAS muon spectrometer, a thin gap chamber (TGC) wheel. [CERN-EX-0609016](https://arxiv.org/abs/1609.09016).
- [2] ATLAS Collaboration, L1Muon Endcap Inner-Coincidence Performance plots in 2024, [PLOT-TDAQ-2024-08](https://arxiv.org/abs/2408.08008).
- [3] R. Mizuhiki, et al., Development of the firmware logic validation system using the FPGA accelerator, JINST 20 (2025) C04011, doi:10.1088/1748-0221/20/04/C04011, arXiv:2503.18357.
- [4] Validation of Level-0 Endcap Muon Trigger Efficiency with RTL Firmware Simulation and Software Reference Model Using TGC Big Wheel Candidate Reconstruction and Inner Coincidence with NSW, TIPP 2026