Hardware status and need for 2025 run: L0TP

Ottorino Frezza

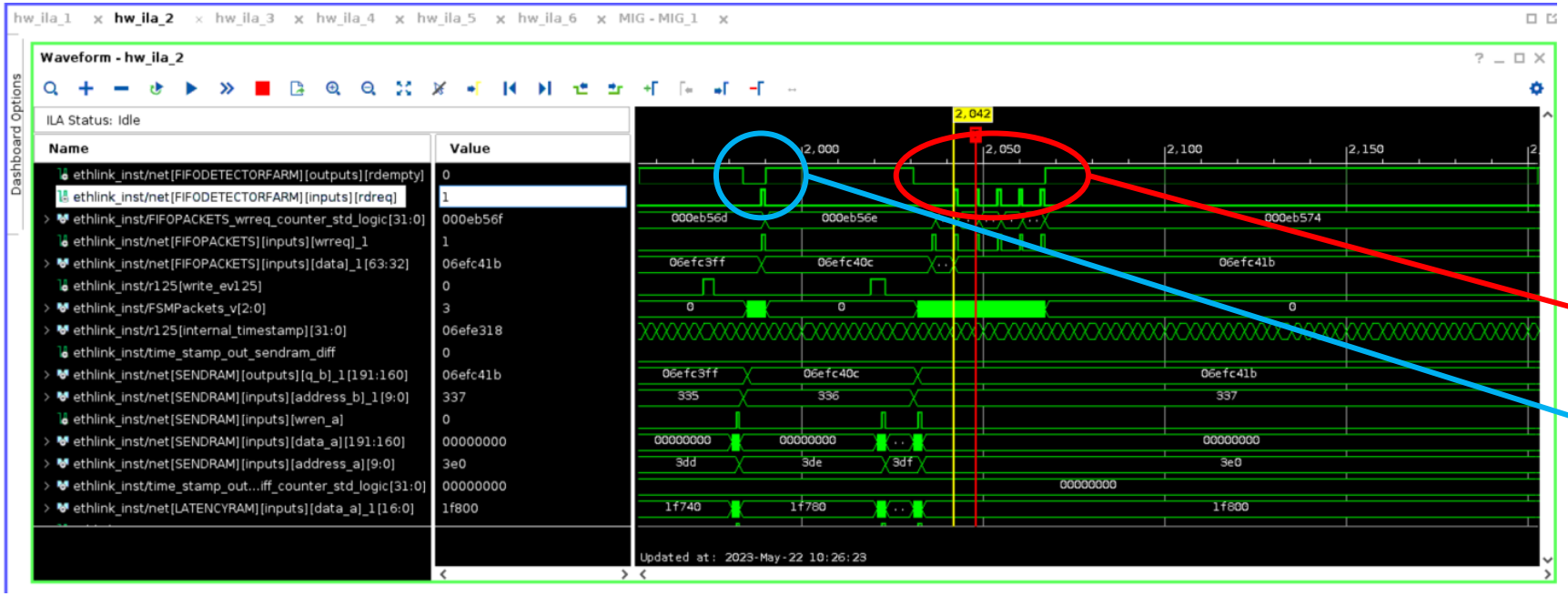
On behalf of L0TP+ Team

Status

- The system is on-line and stable.
- Timestamp mismatch errors (TME) : new version with change on TX logic, to reduce the TME. The number of errors is now negligible.
- Missing EOB : during 2024 run we performed some tests to debug the problem.
- Disconnection of the DIM server : further investigation needed.

Timestamp mismatch bug

- The Main issue with L0TP during 2023 run was the bug on the timestamp reported in the MEP.
- The number of bad timestamp sent from L0TP depended on the intensity and on the quality of the beam. Increasing the intensity increased the number of errors.
- In 2023 we spot this bug adding probes in the firmware.
- The empty signal of asymmetric (40 MHz write clock, 125 Mhz read clock) FIFO FIFODETECTORFARM stays high after read in some cases.
- The Finite State Machine (FSMPACKETS) that reads from that FIFO starts or stops according to this empty signal, and it affected the timestamp field. We changed the logic and the number of TME switched from thousand to tens.



Missing EOB

Slide showed from Joel (Run meeting 2 August). To solve the problem we can start from this preliminary, non exahustive test and implement more sophisticated test in the 2025 (also in Dry Run)

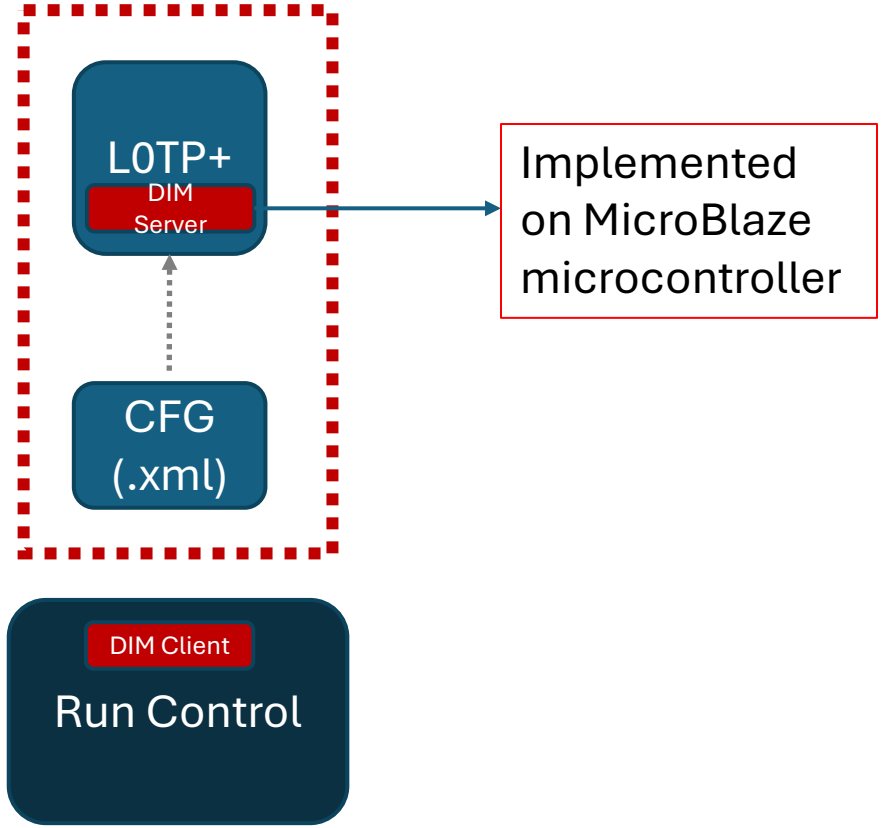
Investigation of missing EOB [ongoing]

[Ottorino & Enrico]

- In technical stop (no beam) we ran triggers with:
  - CTRL=CHOD, read out full frame, zero downscale (maximum noise)
  - + Periodics2 set to period=25000.
  - Giving ~141820 L0 requests (should be equal to L1 in). [Run 14292]
- Found that (sometimes) with 141818 L0 requests, L1 in = 141809. (Missing 9 events)
  - e.g. Burst 136.
  - Appears to be case where:
    - 1 event in last MEP (usually 8 events per MEP) + 1 MEP for EOB.
    - Sent correctly by L0TP+ to farm8 (round robin means these are sent to farm (141818 /8) % 16 = 15 (meaning 15th in the alphabetic list, 1,10,11…16,2…))
    - But farm logs show 'missing fragments' in farm9
- Problem reproducible and symptoms understood but problem not yet solved.

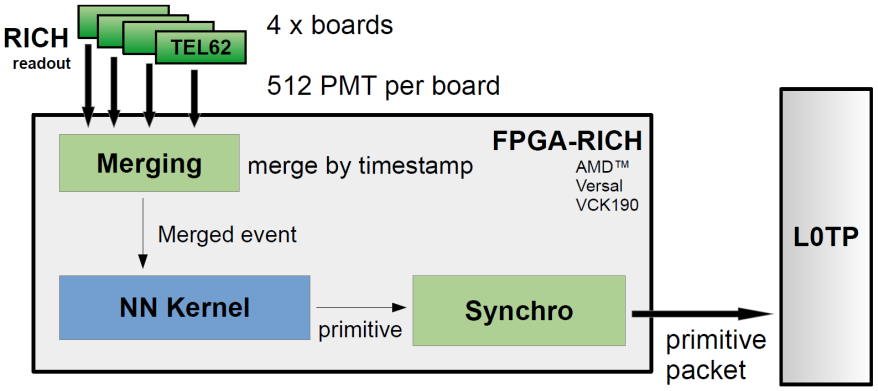
DIM server

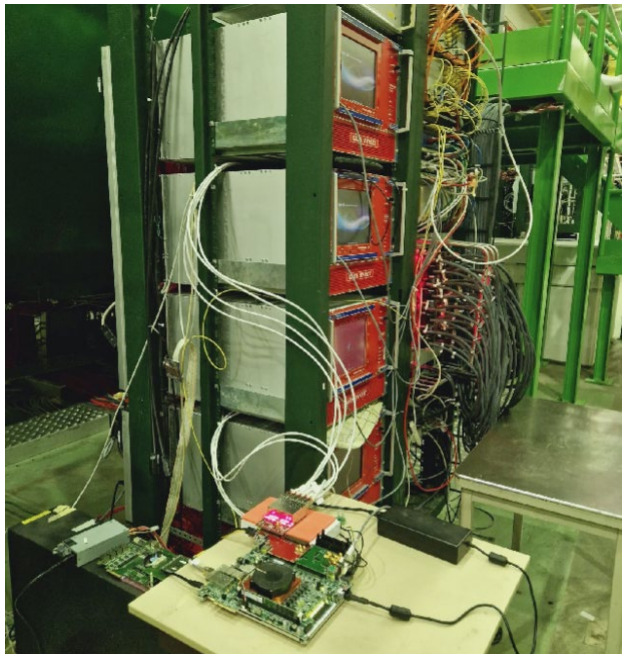
- During the 2023 run we experienced the lost of connection of L0TP microcontroller with the Run Control.
- The problem was due to a high number of requests to the Microblaze (update of the ip address of PC Farm) that froze the system
- Problem fixed at the end of 2023 Run.
- New problem arose during 2024
- We should debug the problem launching the dim from the MicroBlaze shell but : the problem is not so frequent and if the shell crashes the Dim stops to work



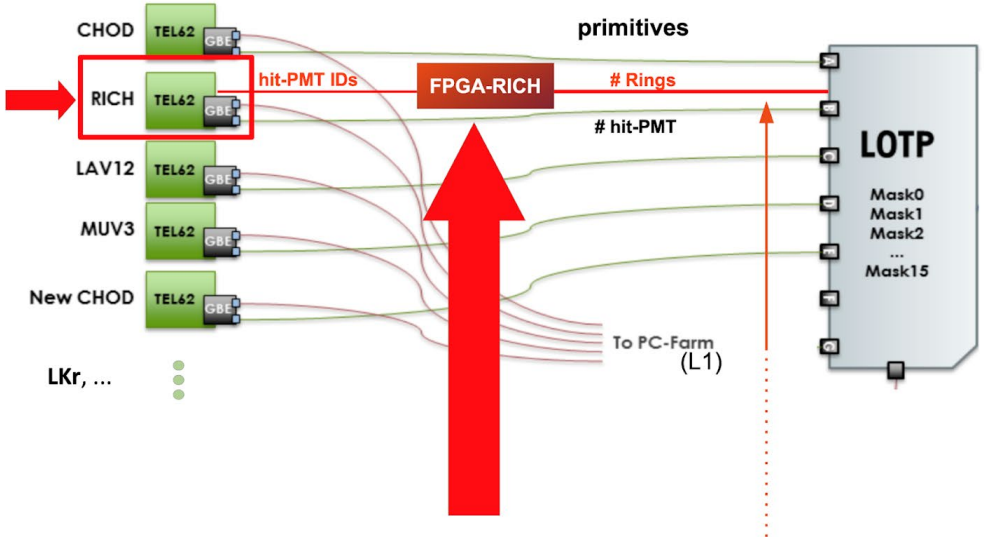
RICH PID using NN on FPGA at L0 Trigger

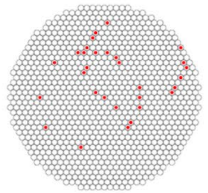
FPGA-RICH : Reconstruction of Cherenkov rings, produced by electrons in the RICH detector, using an AI algorithm on FPGA.



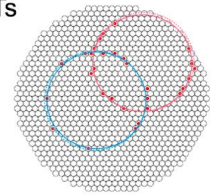


Prototype based on Versal Board





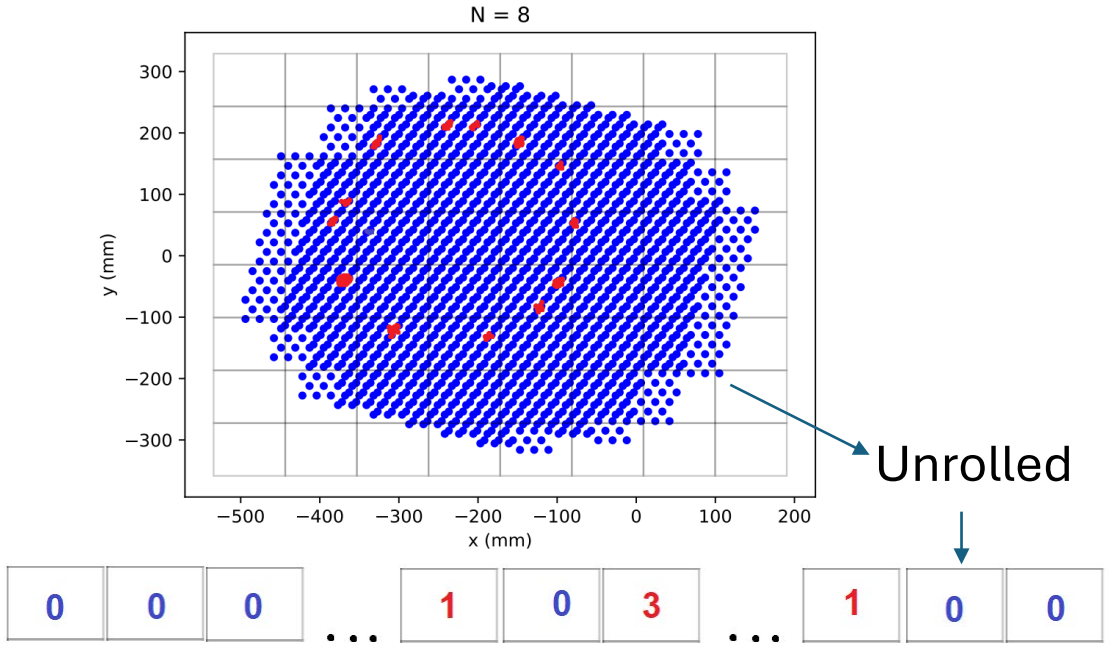
RICH primtives: # of hit-PMTs



FPGA-RICH primtives:
# of rings (0, 1, 2, 3+)
(more in the future, e.g. # of e^-)

RICH PID using NN on FPGA at L0 Trigger

- NN based on dense layers that process an "unrolled" image obtained from mapping the RICH PMTs positions on the RICH disk plane (3 Dense layers: 64 (in) → 64, 64 → 16, 16 → 4 (out))



Classification performance

	Efficiency	Purity
0 Rings	88.88	95.04
1 Rings	88.90	86.48
2 Rings	76.34	72.22
>=3 Rings	77.12	84.62

- **NN KERNEL *average throughput*** (depends on number of hits) **~20 MHz**, ***latency* = 29 clock cycles @**150 MHz (**193 ns**).
- **FPGA-RICH Utilization** LUT = 14%, BRAM = 3%, DSP = 7% FF=6% (VCK190).
- Full pipeline tested with artificial MGP streams in LAB.
- System deployed at the experiment, at the end of the RUN trying to find a workaround to TEL62 fw bugs.

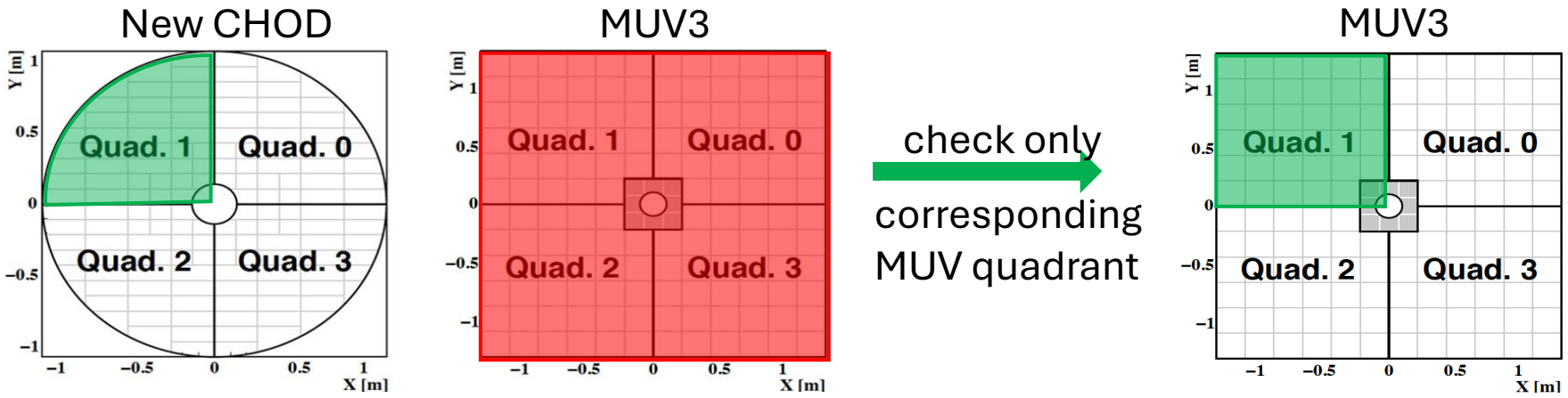
New Trigger algorithm

The presence of halo muon causes random rejection of genuine events in some Trigger Masks

- mask0: RICH3*NewCHOD*!MUV3
- mask1: RICH3*UTMC*!QX*!MUV3*!LKr

The **!MUV3 on all MUV3 surface** can be reduced to quadrants to reduce the over veto probability

Introduce a **new logic signal at L0TP+** level:
Qi!Mi **= QI1*!MQI1+ QI2*!MQI2+ QI3*!MQI3+ QI4*!MQI4**



Try to implement separated **Qi veto conditions quadrants by quadrants** avoiding to veto on halo muons passing into other quadrants

Conclusion

- Major bugs solved, others under test
- The available resources on the new board allow implementation of new features.
- New Trigger algorithm are under test
  - FPGA-RICH on L0TP+
  - New Trigger mask on MUV3 Veto detector

Thank you!