Mu2e-II TDAQ group report

G. Pezzullo (Yale), A. Gioiosa (Pisa)

Intro

We have produced several design proposals and the mini-workshop gave us a good insights from other experiments experience

Now is time to start more specific studies for understanding the feasibility of the proposed solutions and their limits

For each study, the goal is to come up with a good understanding of: (1) resources needed, (2) R&D timeline and (3) expected (preliminary) performance

Expected rates

- ~2x more detector channels, and ~5x more pulses on target, for ~10x higher data rate (if background remains the same).
  - Current expected Mu2e-I data rate from front-ends is 40GBps
- More detector channels and more background implies bigger event sizes (maybe ~3x?)
  - Mu2e-I expected event size is 200KB
- Tape capacity for Mu2e-I is 7PB/year
  - Might assume 2x increase for Mu2e-II to 14PB/year
- Necessary rejection for Mu2e-II is ~3000:1
  - 600KB events @ 3MHz 560MB

Expected rates -> Implications

- Reduced OFF Spill periods (to no OFF Spill time?) implies less advantage for large front-end buffers streaming data
  - In Mu2e-I, have second of downtime to play catch up
  - In Mu2e-II, steady event rate (could buffer just to handle event to event variation, not large accelerator time structures)
- No large front-end buffers at CRV would imply need for low-latency trigger decision for CRV
  - Low latency trigger decision implies an FPGA trigger layer
- Consider the cost of these scenarios:
  - Large CRV buffers and software trigger
  - Small CRV buffers and hardware trigger

Proposals under discussion

- FPGA
  - L1 Trigger (clustering and pattern reconstruction using HLS programming)
- GPU/CPU
  - HLT for evolute Track-Fit
- Backup solution: Scale up Mu2e architecture

Planning

We are planning to organize the work into two separate groups:

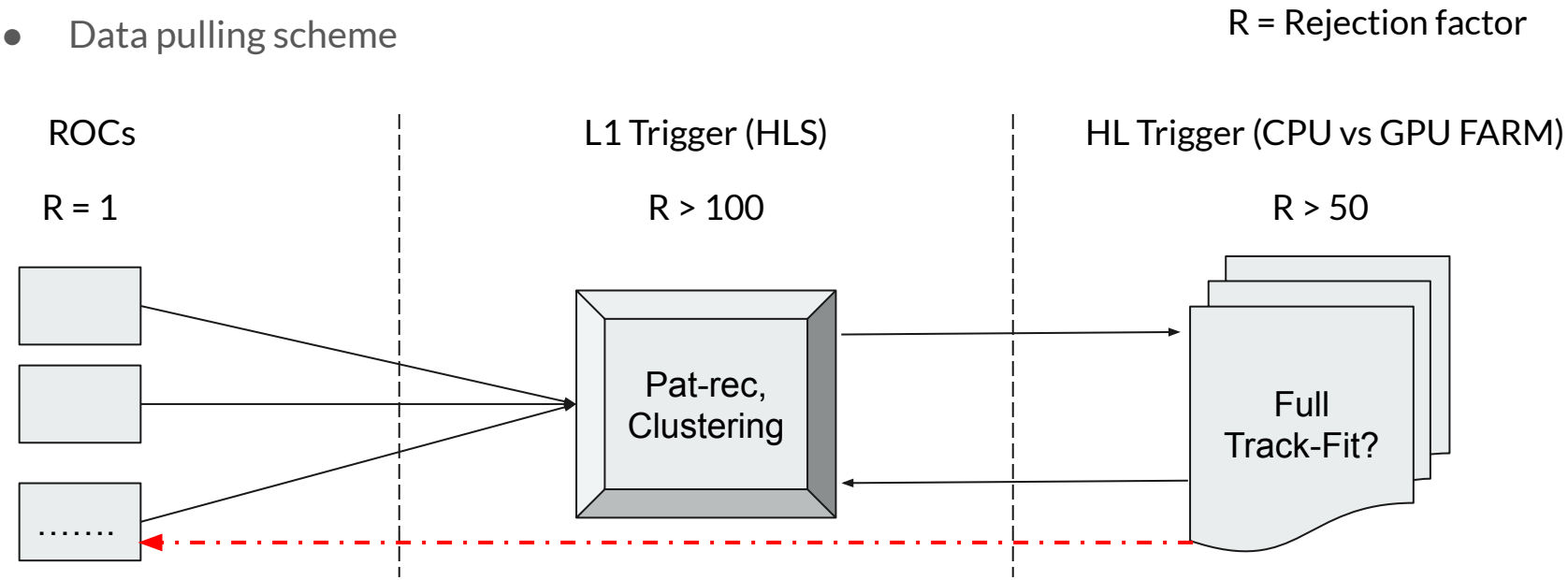
- FPGA
  - L1 Trigger: HLS simulation for pattern reconstruction and clustering
  -
- GPU-based
  - GPU farm w/o L1
  - GPU farm for HLT + L1

Open questions

- Requirements:
  - 10 vs 100 kW: use parasitically simulation from sensitivity study
- GPU-based
  - exploring more in detail the evolute Track-Fit algorithm
  - Interface with artdaq? **G. Cerati sent us a paper**
  - Interface with a L1 layer? **Discussing it with G. Lamanna this Friday**
- FPGA-HLS
  - How difficult is to import our pattern reconstruction and clustering algorithms? (Step 0 implement simplified version of pat-rec)

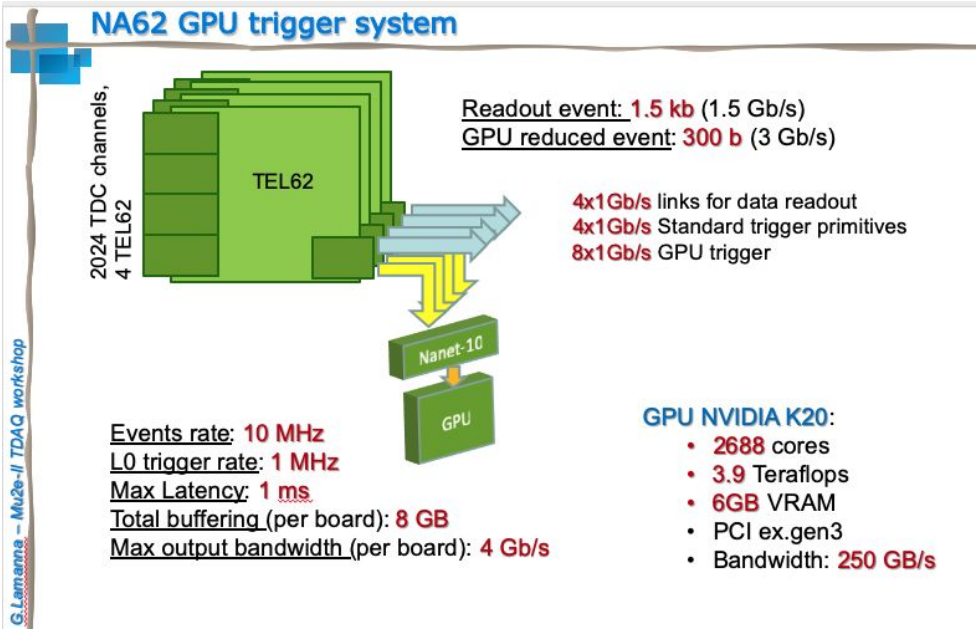
Open questions

- Data pulling scheme



Hybrid solution (maybe there is hope?)

- G .Lamanna (Pisa) showed here a similar solution
- We are trying to get him involved



Working groups

HLS:

- G. Pezzullo and R. Bonventre will start soon learning HLS
  - Started chatting with Ryan who has been helping with the doc and in setting up the test-stand

GPU:

- A. Gioiosa + (Lamanna ?)

References about parallelized tracking algs

* Alice HLT tracking: https://arxiv.org/abs/1712.09430

* LHCb HLT tracking: https://arxiv.org/abs/1912.09161

* CMS pixel tracking: https://cds.cern.ch/record/2293435?ln=en, https://indico.cern.ch/event/742793/contributions/3274390/

* ExaTrkX GNN tracking: https://arxiv.org/abs/2007.00149, https://exatrkx.github.io/

* CMS tracking (multicore CPU): https://arxiv.org/abs/2006.00071

* CBM tracking: https://indico.cern.ch/event/397113/contributions/1837757/attachments/1212483/1768975/Kisel_ACAT2016.pdf

Thanks to Giuseppe Cerri
(Comp-F1 co-convener)