An FPGA-based tracking system for accelerating event reconstruction at LHCb

Francesco Terzuoli *et al.* on behalf of the LHCb-Retina group

3rd Workshop Italiano sulla Fisica ad Alta Intensità - WIFAI
Young Scientist Forum – Nov 13th 2024

Event reconstruction at LHCb

- LHCb is a forward spectrometer tailored for charm and beauty physics studies

- High cross section of interesting events [CERN-LHCC-2014-016]:
    - → triggering on simple quantities is not possible
    - → event fully reconstructed online by LHCb
        at the LHC average rate (~30 MHz)

- Heterogeneous solution in Run 3 (2022-2026):
    - → heterogeneous trigger:
        GPU (HLT1) + CPU (HLT2)

- What about HL-LHC?
    - → Luminosity up to 1.5×10^{34} cm^{-2} s^{-1} (up to x7.5 w.r.t. Run 3)
    - → Increase in luminosity translates in higher computational power

- LHCb established a Coprocessor TestBed:
    - → testing new **heterogeneous computing solutions** with realistic conditions provided

LHCb plan for U2

courtesy of A. Cerri

2
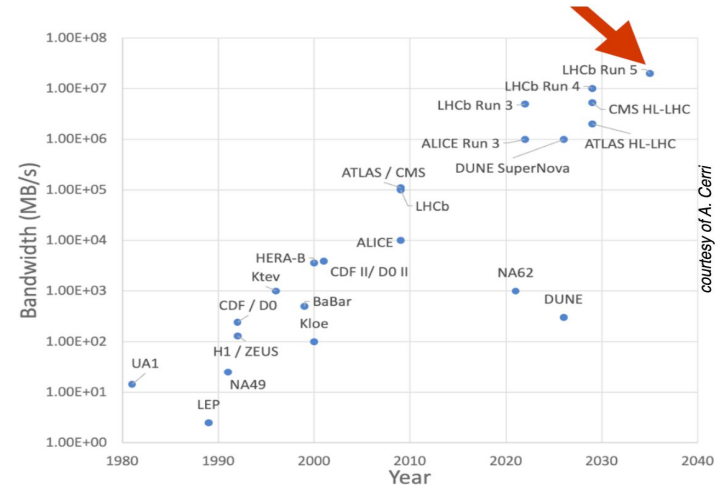
Introducing "primitives"
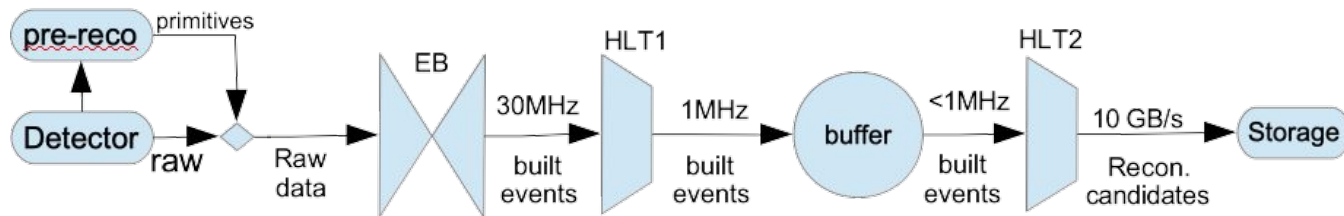
- After detector readout and <u>before</u> event building **primitives** can be created

- Inject primitives (i.e. clusters, track segments) as raw data in the early stages of DAQ
  - Accelerate following stages (off-loading)
  - Possible bandwidth reduction

- Constraints:
  - Located at readout level: required event throughput 30 MHz (LHC bunch crossing rate)
  - Before event building → Constrained latency → can't rely on time-multiplexing

- FPGAs with their low latency and high throughput are good device candidates for this task
  - The "Artificial Retina" is a highly-parallel architecture conceived for this scenario [G.Punzi - Vetex2019]
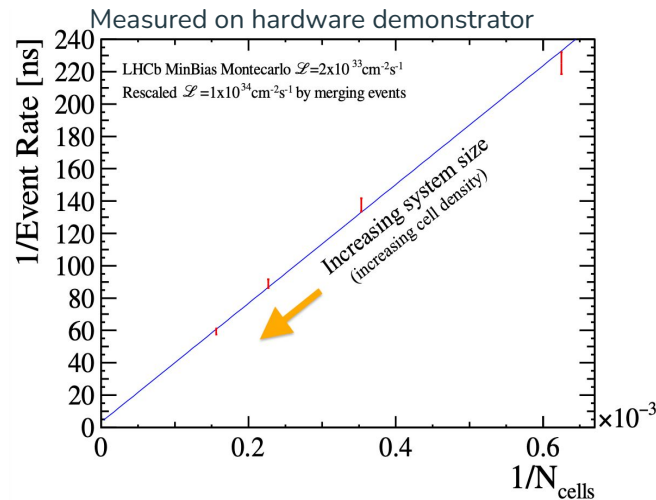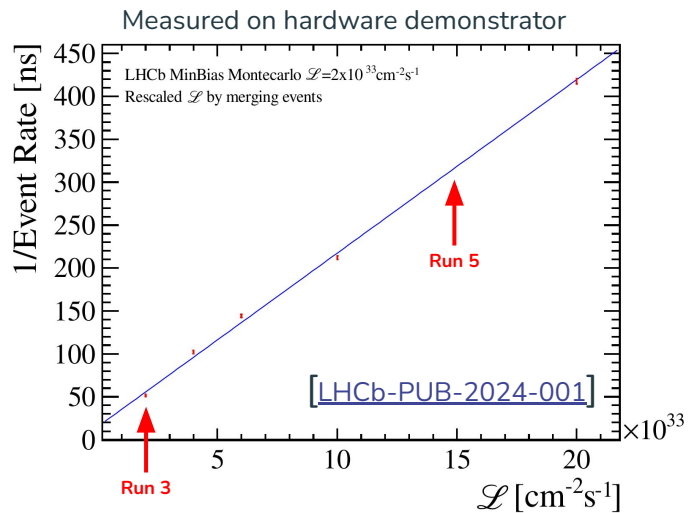
The "Artificial Retina" architecture
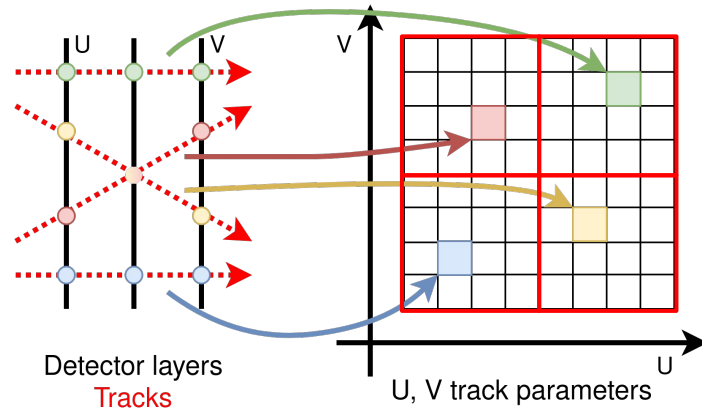
Generalised numerical application of the Hough transform

Detector layers
Tracks

U, V track parameters

- Track parameter space represented by a matrix of processing units (cells)

- Each cell computes a weighted sum of hits near the reference track

- Reconstructed tracks identified as local maxima in the cells matrix response
  - Interpolating responses of nearby cells for obtaining real tracks parameters

Cells work in a fully parallel way for reaching high-throughput and low-latencies

Overcoming FPGA size limitations (without increasing latency) with ***cells spread over several chips***

4

Is Retina advantageous at HL regimes?

- Merging events from Run 3 conditions simulation -> emulating higher luminosities

Measured on hardware demonstrator

LHCb MinBias Montecarlo $\mathscr{L}=2\times10^{33}cm^{-2}s^{-1}$
Rescaled $\mathscr{L}$ by merging events

Run 5

[LHCb-PUB-2024-001]

Run 3

1/Event Rate [ns] vs $\mathscr{L}$ [cm^{-2}s^{-1}]

Measured on hardware demonstrator

LHCb MinBias Montecarlo $\mathscr{L}=2\times10^{33}cm^{-2}s^{-1}$
Rescaled $\mathscr{L}=1\times10^{34}cm^{-2}s^{-1}$ by merging events

Increasing system size
(increasing cell density)

1/Event Rate [ns] vs $1/N_{cells}$

- Similarly, a **bigger system** can be emulated increasing the cell density of the demonstrator

- We can maintain the throughput by linearly increasing the system size

- Where and how much Retina can accelerate LHCb event reconstruction?

5

Tracking at LHCb [CERN-LHCb-PUB-2021-005]

- Velo tracks: hits from VELO (VErtex LOcator - pixel)

- T tracks: hits from SciFi (Scintillating Fibres)

- Long tracks: hits on at VELO-(UT)-SciFi
  - The most used in analysis

- Downstream tracks: hits on UT and SciFi
  - Most interesting for studying:
    Neutral kaons and lambdas ($D^0 \to K_S K_S$, $K_S \to \mu \mu$, etc.)
    Lifetime-unbiased $D^0 \to K_S \pi \pi$
    Exotics LLPs

  - Recently implemented in HLT1 [J. Zhuo - CHEP 2024]

- Downstream tracks are reconstructed starting from T tracks

- Long tracks can be reconstructed starting from T tracks
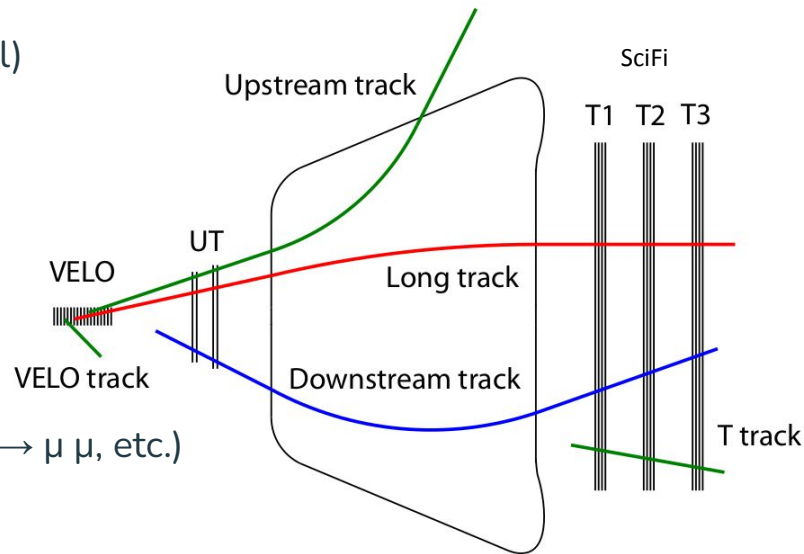
The matching sequence

- One of the possible HLT1 reconstruction sequence at LHCb

- VELO and SciFi independently reconstructed

  - T-tracks + VELO tracks -> Long Tracks
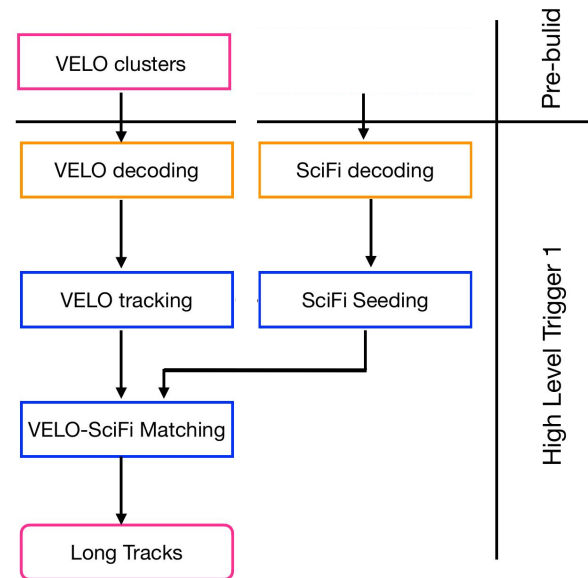
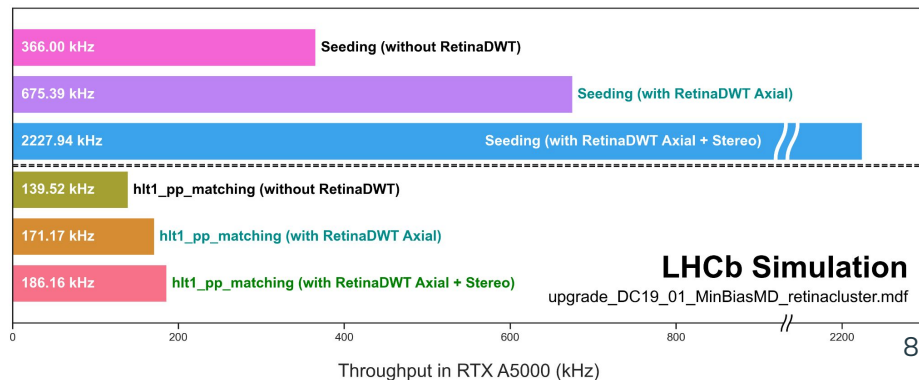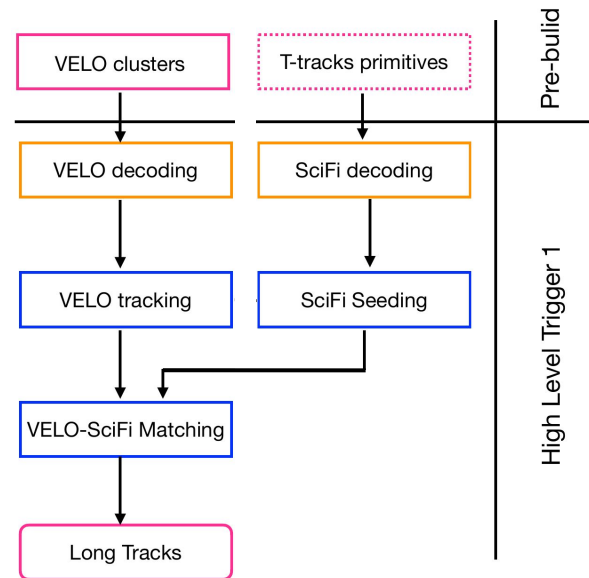- Requires **7.2 µs** per event: **1.5 µs** only for Seeding

7

The matching sequence

- One of the possible HLT1 reconstruction sequence at LHCb

- VELO and SciFi independently reconstructed then matched

  - T-tracks + VELO tracks -> Long Tracks

- Requires **7.2 μs** per event: **1.5 μs** only for Seeding

How much can we accelerate it using *primitives*?

- Seeding -> primitive decoding and refitting (test with HLT1 sw)

- Execution time:
  - Total: **5.4 μs**
    - Decoding and refitting: **0.06 μs**
    - Negligible overhead

- **Saved time >100% replaced algorithm**
  due to memory off-loading

LHCb Simulation
upgrade_DC19_01_MinBiasMD_retinacluster.mdf

8

The Downstream Tracker

- LHCb plans to build a device (DWT) for reconstructing T track primitives using the "artificial retina" architecture for **Run 4** [LHCB-TDR-025]

- System overview

  - ~100 FPGAs boards (new LHCb readout boards)

  - 24 hosts servers (separated cluster from DAQ servers)

  - Infiniband connection to current DAQ

- DWT as a mean for accelerating HLT, **not substituting it**

  - Retina -> combinatorial side of the task

  - HLT can refine the primitives with further ghost removal, clone killing and employ more sophisticated tools (e.g. ML)

Approved by LHCC

Performance of the DWT via simulation

- C++ Retina emulator employed (exact bitwise adherence to FPGA firmware [Terzuoli - CDT2023])

- 2-step reconstruction:
  - Axial pattern recognition (Retina) + Ghost removal (χ^2 fit)
  - Stereo pattern recognition (Retina) + Ghost removal (χ^2 fit)

χ^2 scatter plot
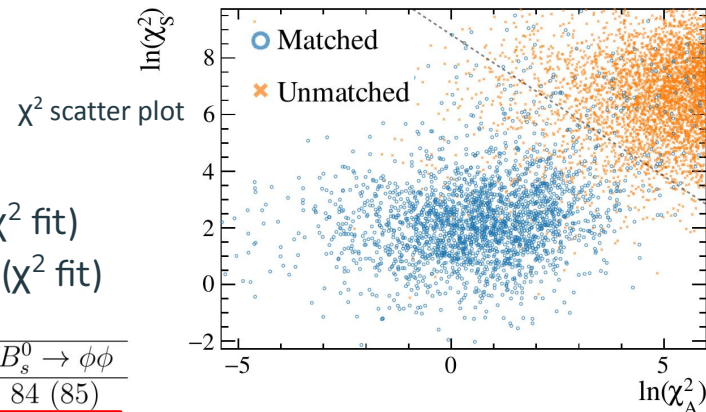
Track type	MinBias	$D^0 \to K_S^0 \pi^+ \pi^-$	$B_s^0 \to \phi\phi$
Long, $p > 3\,\mathrm{GeV}/c$	85 (86)	83 (84)	84 (85)
Long, $p > 5\,\mathrm{GeV}/c$	90 (91)	89 (90)	89 (89)
Long from B not $e^\pm$, $p > 3\,\mathrm{GeV}/c$	-	-	88 (87)
Long from B not $e^\pm$, $p > 5\,\mathrm{GeV}/c$	-	-	90 (90)
Down, $p > 3\,\mathrm{GeV}/c$	84 (85)	83 (84)	83 (84)
Down, $p > 5\,\mathrm{GeV}/c$	89 (91)	88 (89)	88 (89)
Down from strange not $e^\pm$, $p > 3\,\mathrm{GeV}/c$	-	83 (83)	-
Down from strange not $e^\pm$, $p > 5\,\mathrm{GeV}/c$	-	88 (88)	-
Down from strange not long not $e^\pm$, $p > 3\,\mathrm{GeV}/c$	-	83 (83)	-
Down from strange not long not $e^\pm$, $p > 5\,\mathrm{GeV}/c$	-	88 (89)	-
ghost rate	16 (10)	17 (12)	17 (13)
ghost rate / (1 - ghost rate)	0.2 (0.1)	0.2 (0.1)	0.2 (0.1)

- $\chi^2_A < 60$
  $\ln(\chi^2_A) + \ln(\chi^2_S) < 8.5$

- Fiducial requirements:
 $p_T > 200$ MeV/c; $2 < \eta < 5$

Conclusions

- **Heterogeneous computing** solution is becoming a must as we move to the high luminosity frontier
  - Pattern recognition tasks will greatly benefit from this

- FPGAs offer as the target device due to high throughput and low latency, also **greener solution**

- Primitives as intermediate reconstruction step at readout
  - Accelerating the following DAQ chain -> Off-loading -> More complex tasks available

- **Retina** reconstructed primitives will become a reality in LHCb Run 4

  - DWT tracker -> **Seeding+Matching sequence** can be **accelerated by 33%** when primitives are included in the main LHCb reconstruction sequence

- Good tracking performance -> fine tuning in evaluation (balancing Retina/HLT1 refinement)

- Plan to build a **vertical slice of the DWT before Run 4** for extensive integration tests

- **Gain of knowledge and experience in view of the highly challenging HL-LHCb (Run 5)**

Thanks for your attention!

Backup

Are we capable of building this?

- Architecture tested during several years of R&D

- **HW Demonstrator** installed LHCb TestBed facility (Point 8)
  - Receiving data from LHCb DAQ in real time

- Implemented on 8 FPGA Stratix 10

- VELO quadrant reconstruction

- Tested on LHCb Montecarlo data:
  - Nominal luminosity (2×10^{33} cm^{-2}s^{-1})
  - Longest uninterrupted run: 27 days
  - **Event rate: 19.6 MHz**
  - Power consumption: 550 W

- Real data: good qualitatively accord
between Retina and HLT2 reconstructed tracks (no ghost/clone killing in Demonstrator)
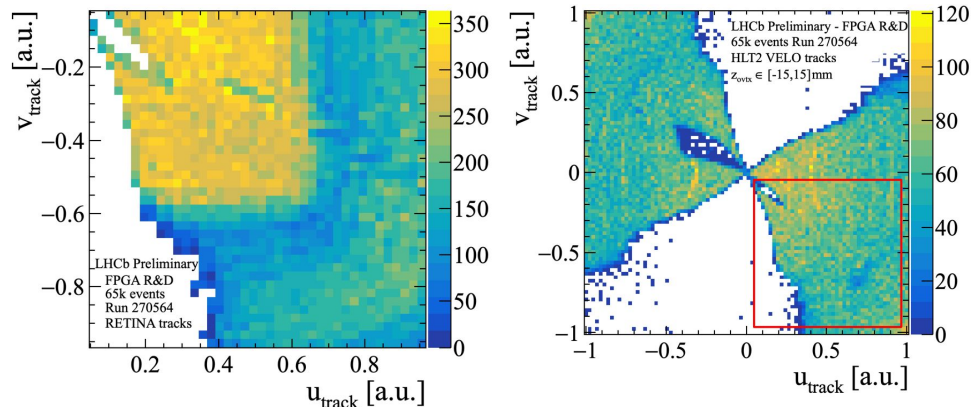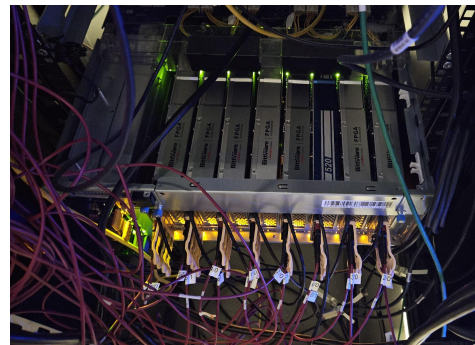
Why heterogeneous solutions?

Where does this lead us?

- Architecture-aware programming is a must. In particular, memory will play a major role to designing our future algorithms

From "Computer Architecture"
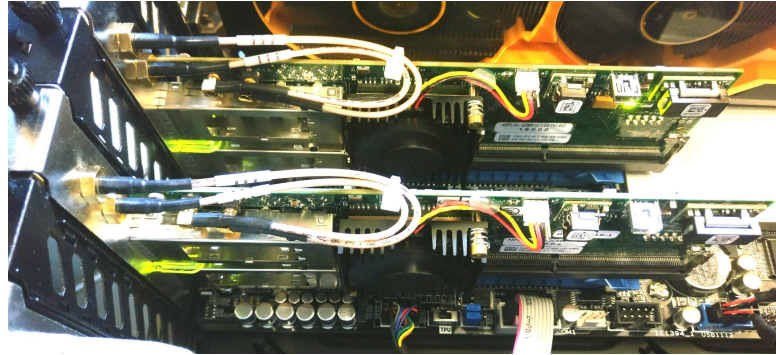
[D.Campora, 6th Ull workshop]

Daniel Cámpora

13
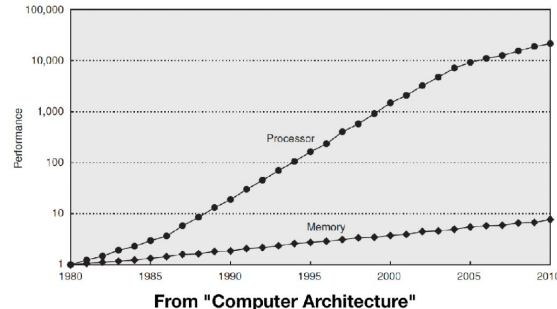
- Tasks that require non-contiguous data (e.g. pattern recognition) need many memory accesses

- More suitable devices can help us with these type of tasks in a more efficient way
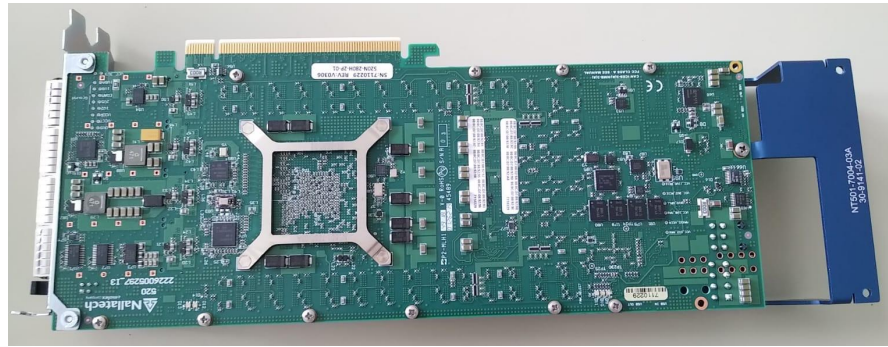 → HLT can dedicate freed resources to more complex trigger selections

Hardware

- Prototyping board,
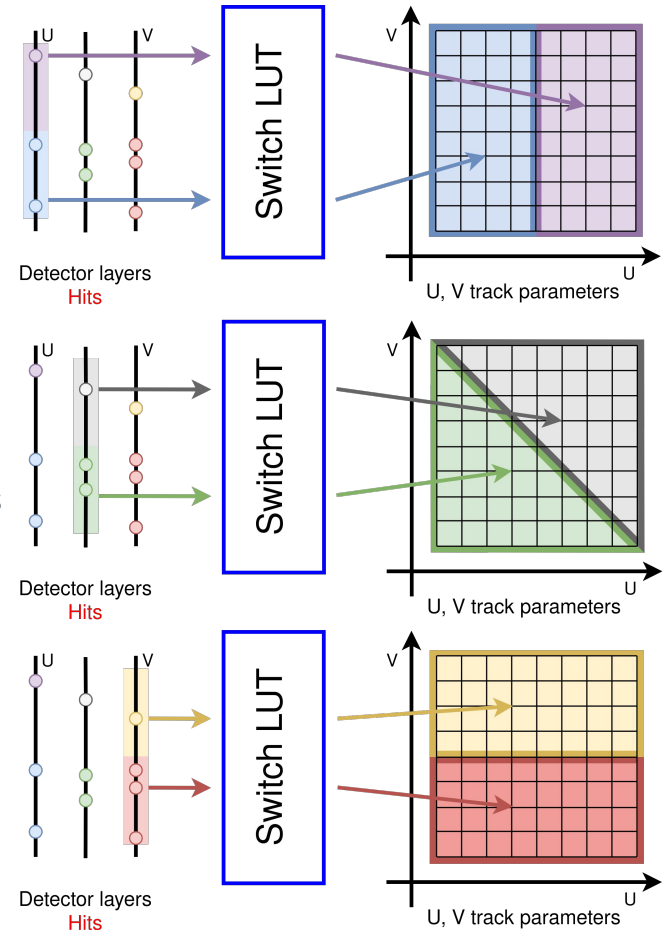 2 Intel Stratix V FPGAs,
  96 optical links

- PCIe 8x board, 1 Intel Arria V GX FPGA, 8 optical links

- PCIe 16x board, 1 Intel Stratix 10 FPGA, 16 optical links

The Distribution Network

- Hits are provided to different Tracking boards arranged by sub-detector DAQ board.

- A custom distribution network rearranges the hits by track parameters coordinates (similar to a "change of reference system").

- Using Lookup Tables (LUTs), the Distribution Network delivers to each cell only hits close to the parametrized track, enabling large system throughput.

- The Distribution Network is a single entity transversal to all the Tracking boards.

- We designed a modular Distribution Network spread over the same array of FPGAs performing the tracking.

Detector layers
Hits

Switch LUT

U, V track parameters

Detector layers
Hits

Switch LUT

U, V track parameters

Detector layers
Hits

Switch LUT
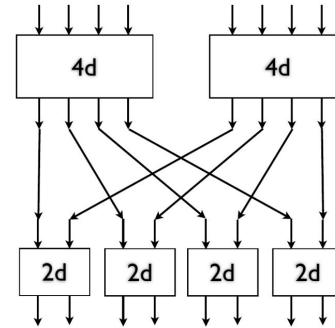
U, V track parameters

Switch

- 2-way dispatcher (2d): 2 splitters (1 input - 2 outputs) and 2 mergers (2 inputs - 1 output).

- Combining 2-way dispatchers is possible to build a switch with the desired number of lanes:

  - Switch with $N = 2^n$ lanes requires M 2-way dispatchers: $\begin{cases} M(0) = 0 \\ M(n) = 2M(n-1) + 2^{n-1} \end{cases}$

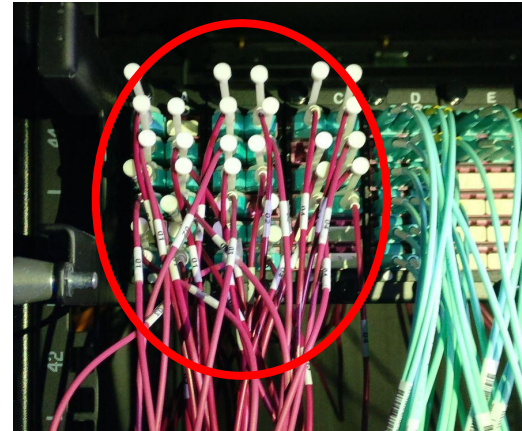- We can implement any 2^n lanes switch changing a single parameter.
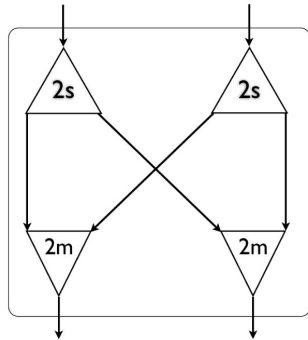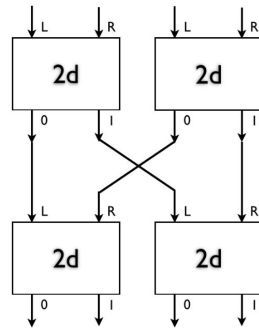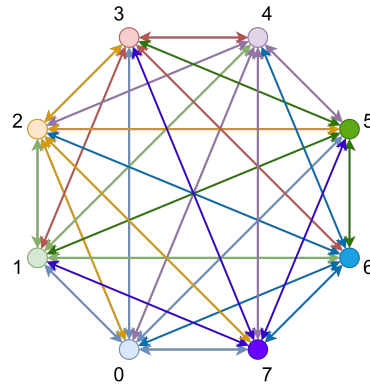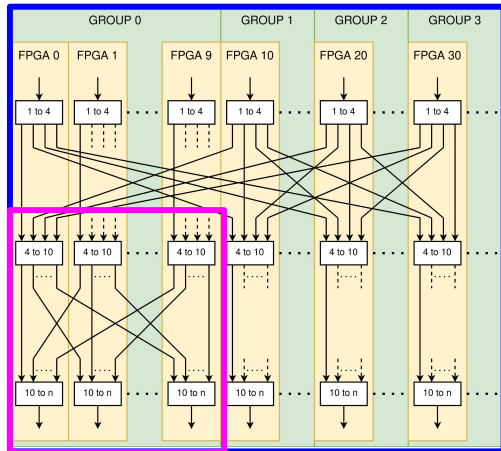

2-way dispatcher (2d)   4-way dispatcher (4d)   8-way dispatcher (8d)

Distribution network

- Portion of the whole VELO distribution network currently implemented
  - 8 nodes full-mesh network
  - 28 full-duplex links at 25.8 Gbps
  - Total bandwidth 1.41 Tbps

Optimising the hits distribution

- The switch (modular design) handles the hits distribution by routing the hits to the correct implemented TPUs using look-up-tables (LUTs)
- Implemented 64 TPUs over 8 boards (8TPUs/board)
- First optimised: last step of the switch (8x8 dispatcher)
  - One switch/chip: 1TPU/Out_line and 2 VELO modules/Input_line
- The optimisation
  - Pairing the TPUs (2 by 2) with highest number of common hits
  - Iterate over the paired TPUs as we move to higher switch levels
    - Move hits duplication towards last switch layers

2-way dispatcher (2d)   4-way dispatcher (4d)   8-way dispatcher (8d)

20
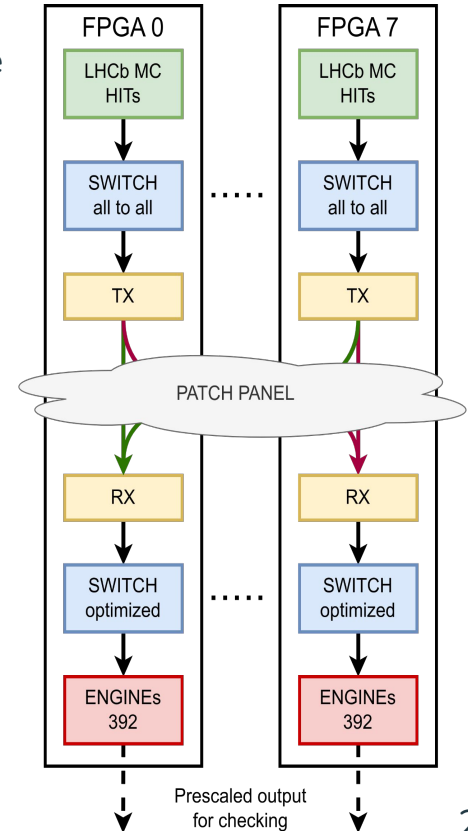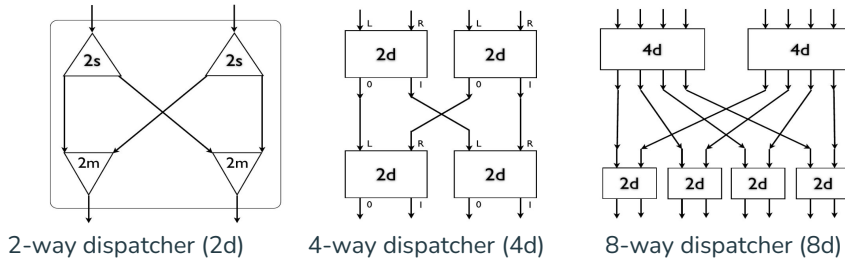
Running live on data from collisions

- Testbed facility is fed with data from the Monitoring Farm (1kHz evts) and they are stored on disk
  - Chunks of RawEvents (1.9MB/chunk)

- In addition to hit clustering, alignment need to be applied
  - Conversion of VELO clusters from local to global coordinates

- Need communication with FPGAs using stock board PCIE driver
  - Loading VELO hits to the boards
  - Reading reconstructed tracks from the boards
  - Checking FPGAs error registers

- Moreover due to the the demonstrator not being integrated in the LHCb online system
  - Decoding incoming RawEvents from Monitoring Farm
  - Selecting the detector sources (VELO modules compatible with the chosen quadrant)