An FPGA based track finder at Level 1 for CMS at the High Luminosity LHC

Connecting the Dots / Workshop on Intelligent Trackers 2017, LAL Orsay
Alexander David Morton

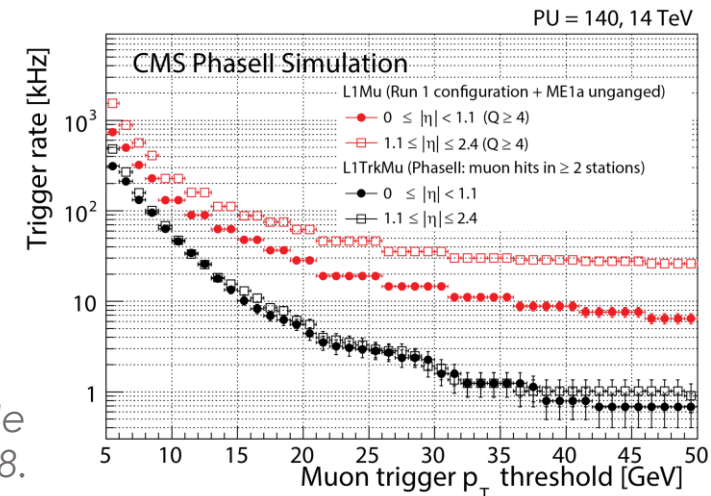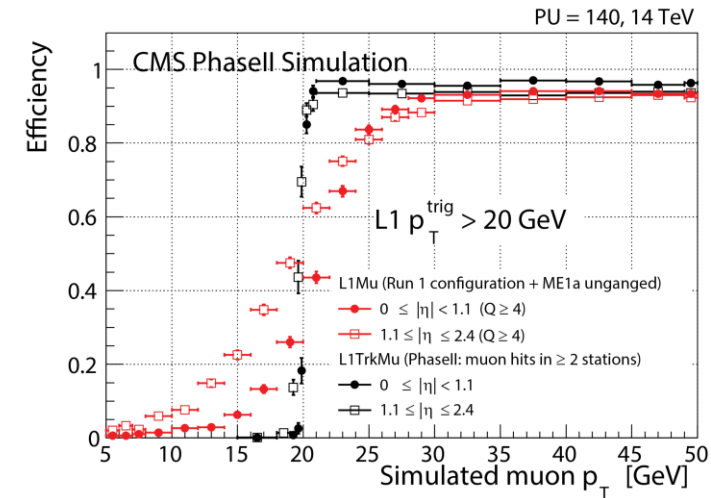# *Background*

Alexander David Morton
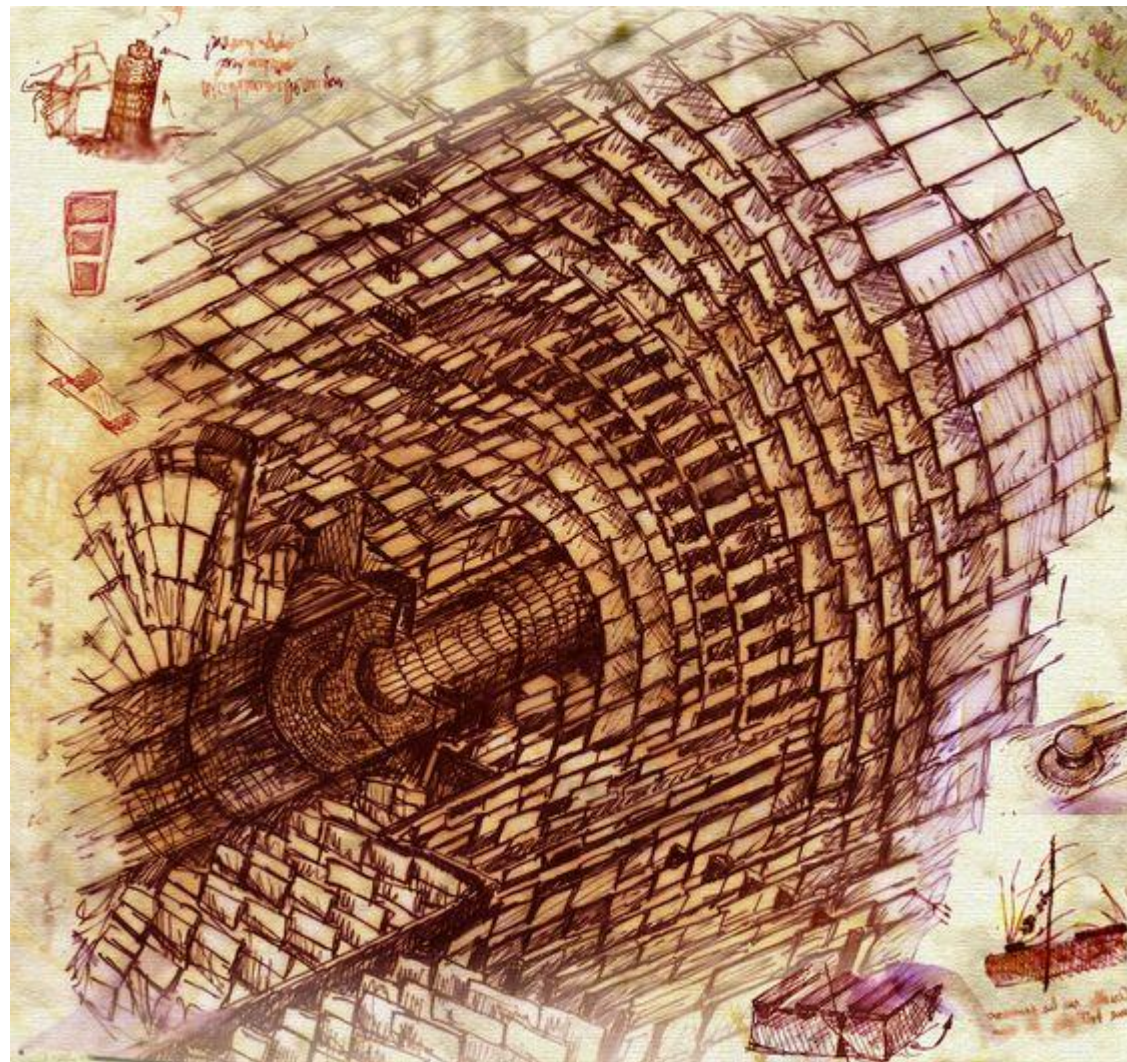
Motivations for replacing the CMS Tracking Detector at the HL-LHC

- Counter existing radiation damage/ cope with increased fluxes.

- Extended tracking acceptance to $|\eta| < 4$.

- Reduced material in the tracking volume.

- Maintain high track reconstruction efficiency under increased pile-up conditions.

- Provide limited information to the Level 1 (L1) hardware trigger system to avoid raising trigger thresholds (and physics).
  - Average number of inelastic collisions per bunch crossing (pileup) will increase up to 140-200!

Tracking Trigger Requirements

- Tracking information for the Level-1 (L1) hardware trigger system in the form of fully reconstructed tracks will be essential for maintaining trigger performance.
  - Current L1 trigger menu at pileup 200 would require 4000 kHz!
  - Adding tracking information to L1 trigger objects would reduce rates down to 500 kHz.
- Total L1 latency limited to 12.5 μs:
  - 4 μs allowed for track reconstruction following readout electronics generation, packaging and transmission from the Data, Trigger and Control (DTC) system.

Plots from CMS Collaboration, "Technical Proposal for the Phase-II Upgrade of the CMS Detector", Technical Report CERN-LHCC-2015-010. LHCC-P-008. CMS-TDR-15-02, Geneva, June 2015.

PU = 140, 14 TeV

CMS PhaseII Simulation

Efficiency

L1 p_T^{trig} > 20 GeV

L1Mu (Run 1 configuration + ME1a unganged)
- $0 \le |\eta| < 1.1$ (Q ≥ 4)
- $1.1 \le |\eta| \le 2.4$ (Q ≥ 4)

L1TrkMu (PhaseII: muon hits in ≥ 2 stations)
- $0 \le |\eta| < 1.1$
- $1.1 \le |\eta| \le 2.4$

Simulated muon p_T [GeV]

PU = 140, 14 TeV

CMS PhaseII Simulation

Trigger rate [kHz]

L1Mu (Run 1 configuration + ME1a unganged)
- $0 \le |\eta| < 1.1$ (Q ≥ 4)
- $1.1 \le |\eta| \le 2.4$ (Q ≥ 4)

L1TrkMu (PhaseII: muon hits in ≥ 2 stations)
- $0 \le |\eta| < 1.1$
- $1.1 \le |\eta| \le 2.4$

Muon trigger p_T threshold [GeV]

CMS Phase-2 Outer Tracker

- Outer Tracker provides tracking information through modules of two closely space silicon sensors.
  - Charged particles produce pairs of hits (a "stub") in these two sensors.
- Relative position of the two hits determines the track p_T (assuming beam-line origin):
  - On-detector electronics only transmit off stubs consistent with p_T > 2 - 3 GeV/c.
   - Reduces rate by factor ~ 10.
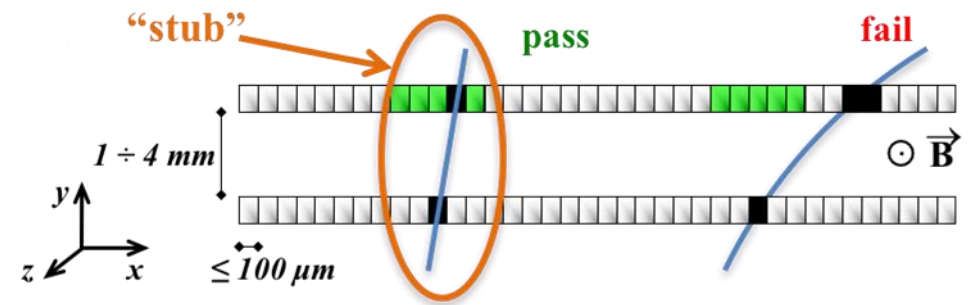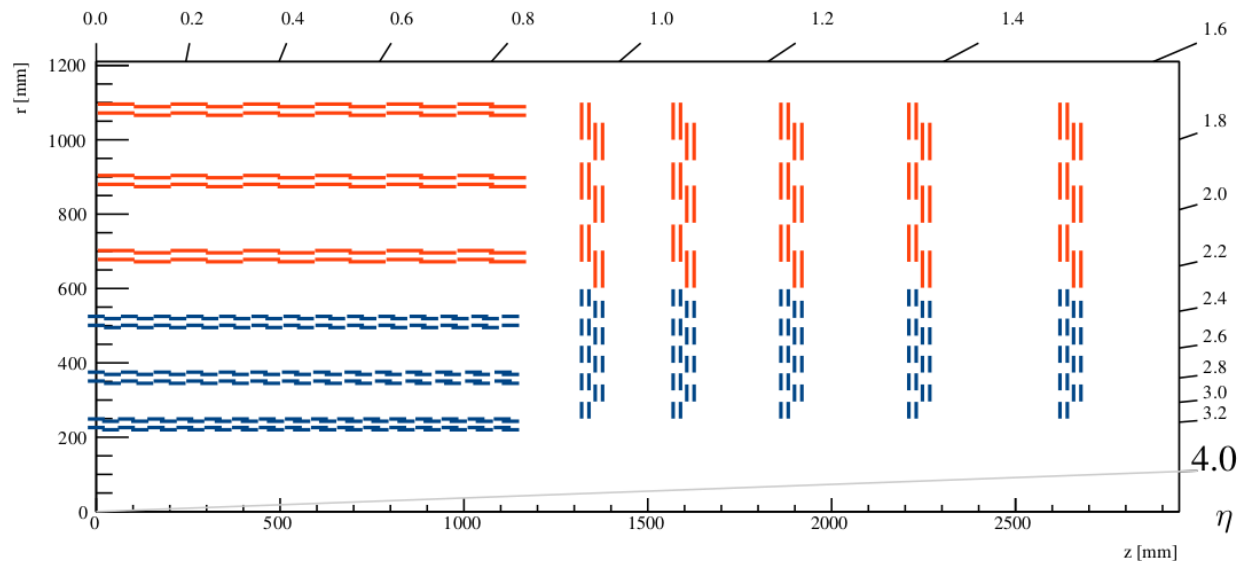
Outer Tracker Layout (left):
Blue: modules with a pixel and strip layer
Red: modules with two strip layers

A Time-Multiplexed Track Trigger

- One of three proposals being investigated in CMS.

- Concept:
  - Scalable, configurable, and redundant system architecture.
  - Track Finding Processor (TFP) implemented using FPGAs:
    - Off the shelf components.
    - Flexibility to modify tracking algorithm based on LHC conditions or new ideas.
  - Fully Time-Multiplexed Design.

- Constraints:
  - How the tracker is cabled to the Data, Trigger and Control (DTC) system.
  - Number of high speed links available for DTC and Track Processing boards.
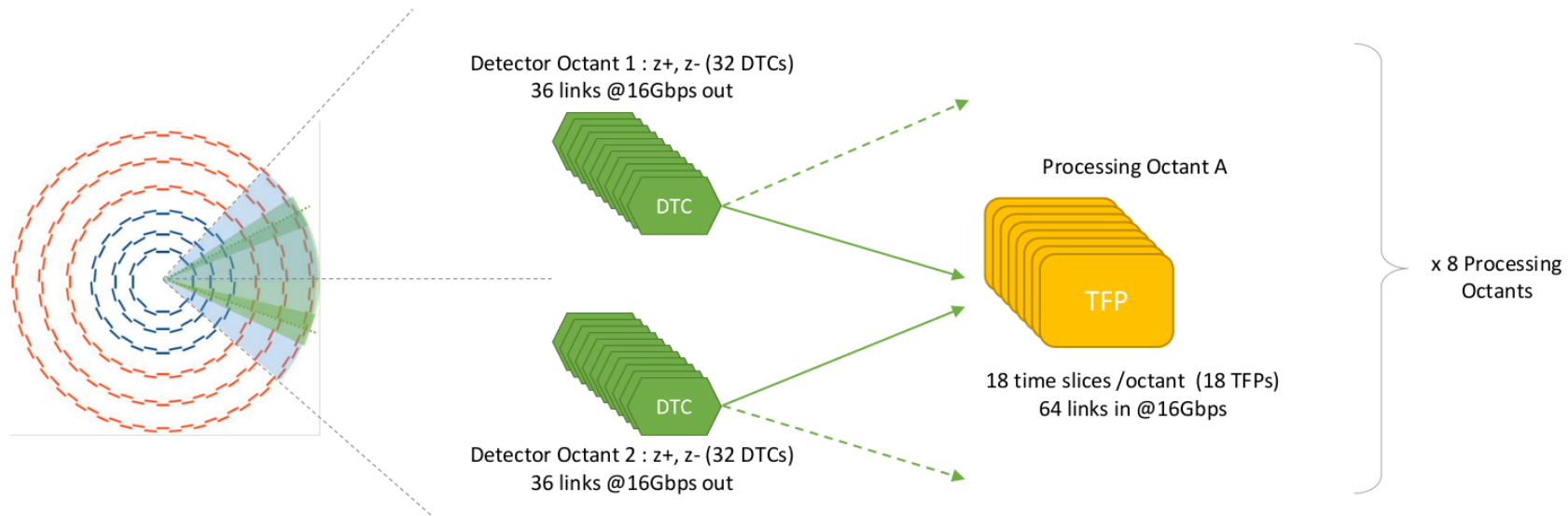
Time-Multiplexing Crash Course

- N identical processors, each processor processing 1/N events.

- Advantages:
  - Fully pipelined (no sideways connections).
    - Synchronisation required only within each node.
  - Allows demonstration of final system with one TFP.
  - A TFP failure results in loss of 1 bunch crossing in N instead of loss of physical region.
  - Spare TFPs in final system allow for online recovery in case of failure or parasitic testing of new algorithms during LHC runtime.

- Time-Multiplexing is successfully used in current CMS trigger.
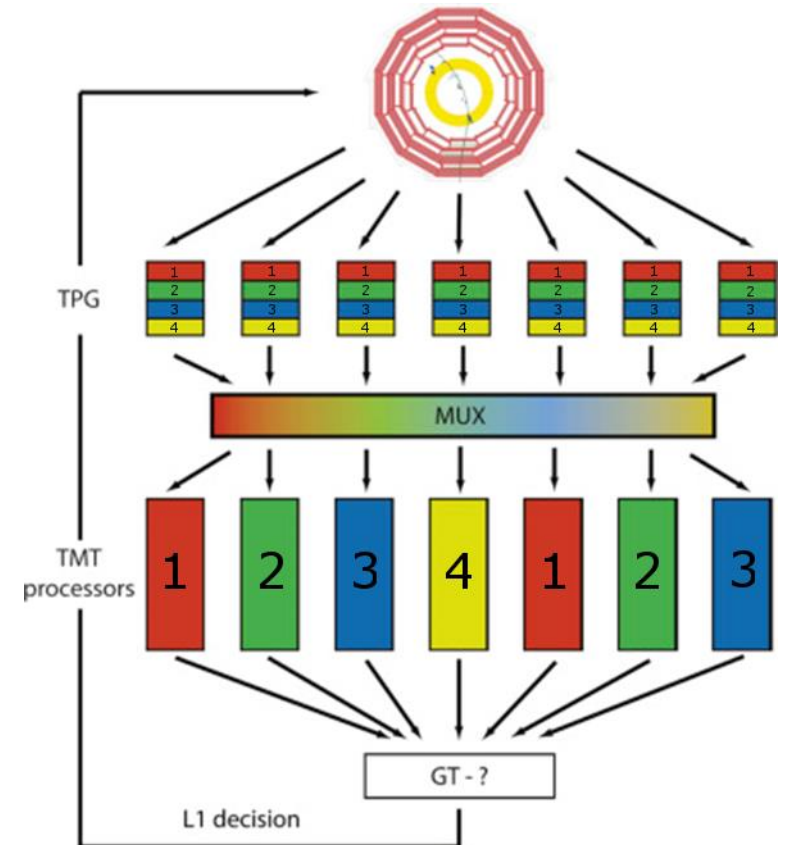
TFPs = TM Period

System Overview

- Tracker detector is ~ divided into φ octants.
- Track Finding "processing octants" are offset by ½ an octant compared to the "detector octants".
  - "Processing octant" never needs stubs from more than two "detector octants" to reconstruct tracks.
  - Perform track finding independently in each sector in parallel.


Detector Octant 1 : z+, z- (32 DTCs)
36 links @16Gbps out

DTC

Processing Octant A

TFP

x 8 Processing Octants

18 time slices /octant (18 TFPs)
64 links in @16Gbps

DTC

Detector Octant 2 : z+, z- (32 DTCs)
36 links @16Gbps out

Track Finding Processor

- Track Finding Processor divided into four self-contained logical blocks:
  - Geometric Processor (GP): Pre-processes stubs and subdivides detector octants into sectors.
  - Hough Transform (HT): Highly parallelised initial coarse track finding in the r-φ plane.
  - Track Fitter (TF): Cleans tracks and precisely fits helix parameters and removes fake tracks.
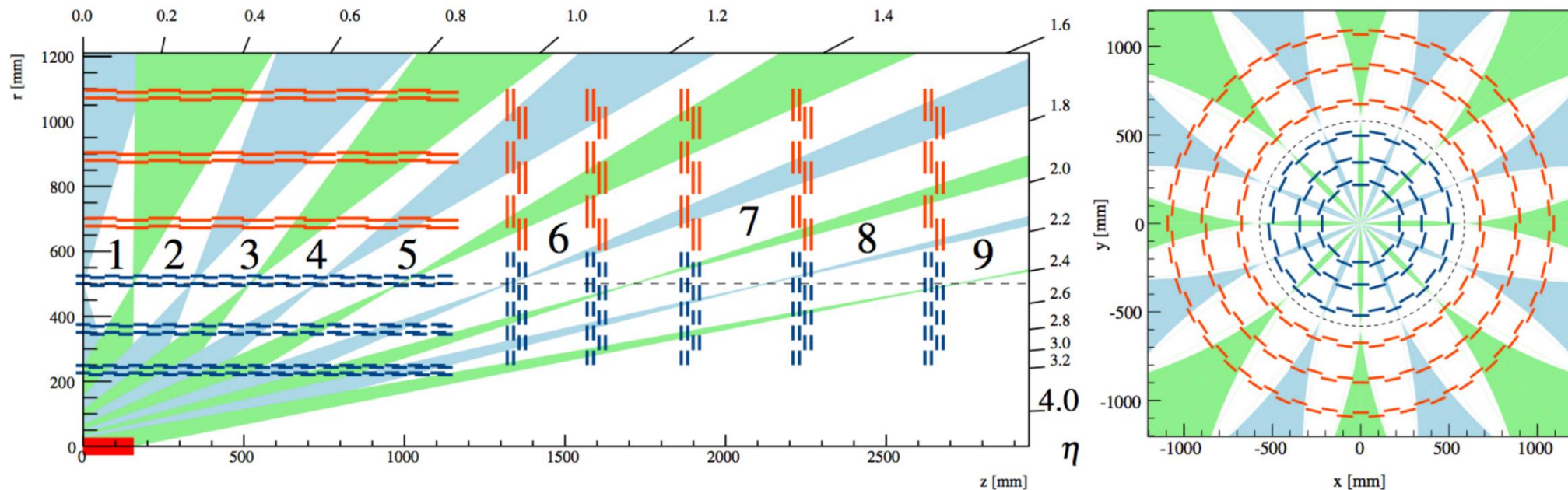 - Duplicate Removal (DP): Final pass filter to remove duplicates generated by the HT.
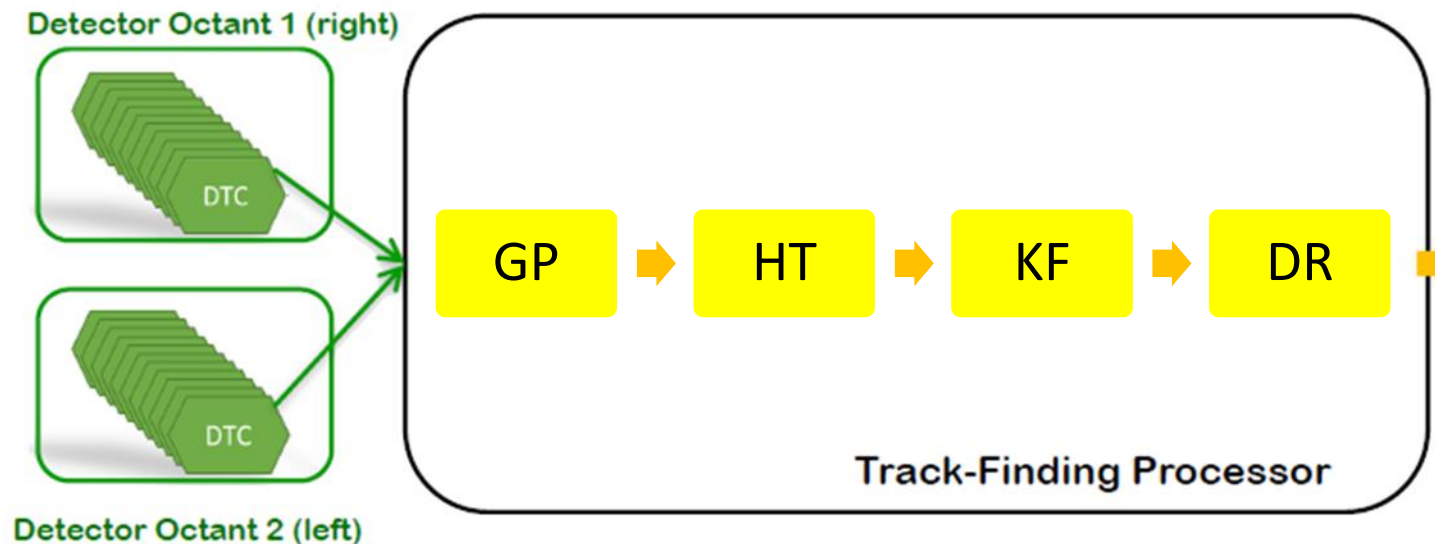
Geometric Processor
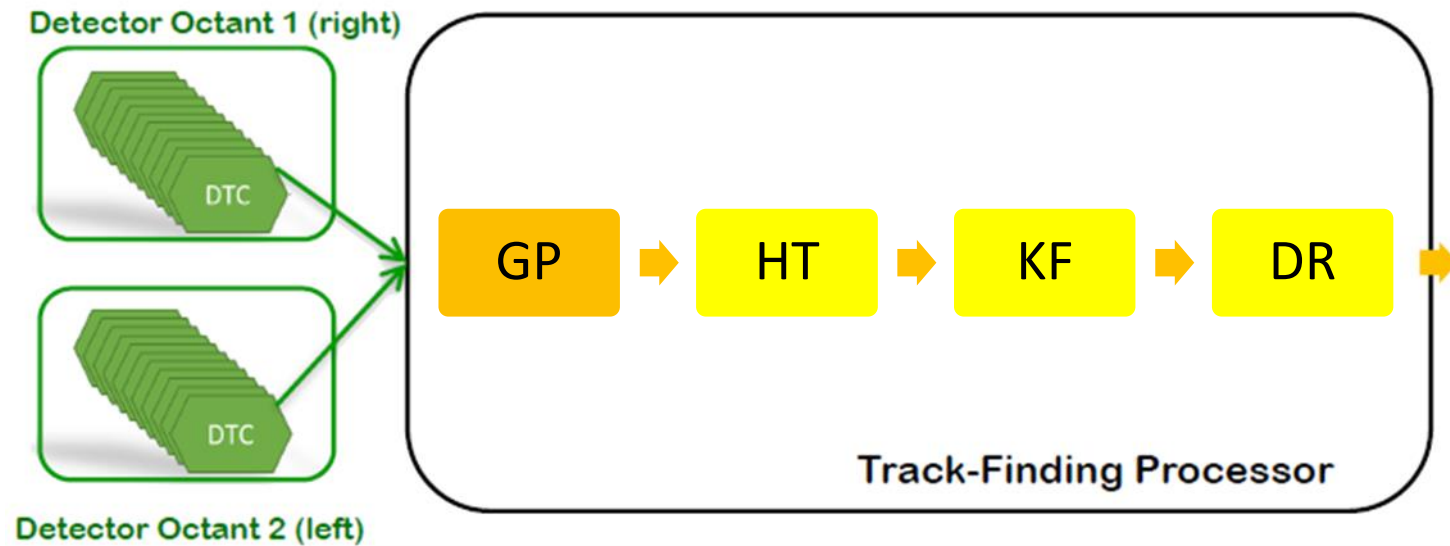
Geometric Processor
Divide et impera

- Detector is divided into φ octants, inspired by the proposed cabling scheme in φ octants.
  - Each octant is subdivided into 2φ x 18η sectors.
 - Independent track-finding occurs in parallel in each processing sector

Geometric Processor
How it works

- Geometric Processor:
  - Assigns stubs to sectors and transmits from each sector along dedicated link to next stage.
    - Ensure tracks found are consistent with line in r-z plane.
  - Duplicates stubs if consistent with more than one sector due to track curvature.
  - Formats stub data for more convenient use downstream.

- Pre-processing and assignment:
  - Converts 48-bit DTC stubs into a 64-bit extended format.
  - Assign stubs to geometric sub-sectors.

- Routing:
  - 72 inputs (one per DTC) routed to any of 36 outputs (one per sector).
  - Happens in three steps:
    - rough η sorting (6 bins) $\rightarrow$ fine η sorting (3 bins) $\rightarrow$ ϕ sorting (2 bins).

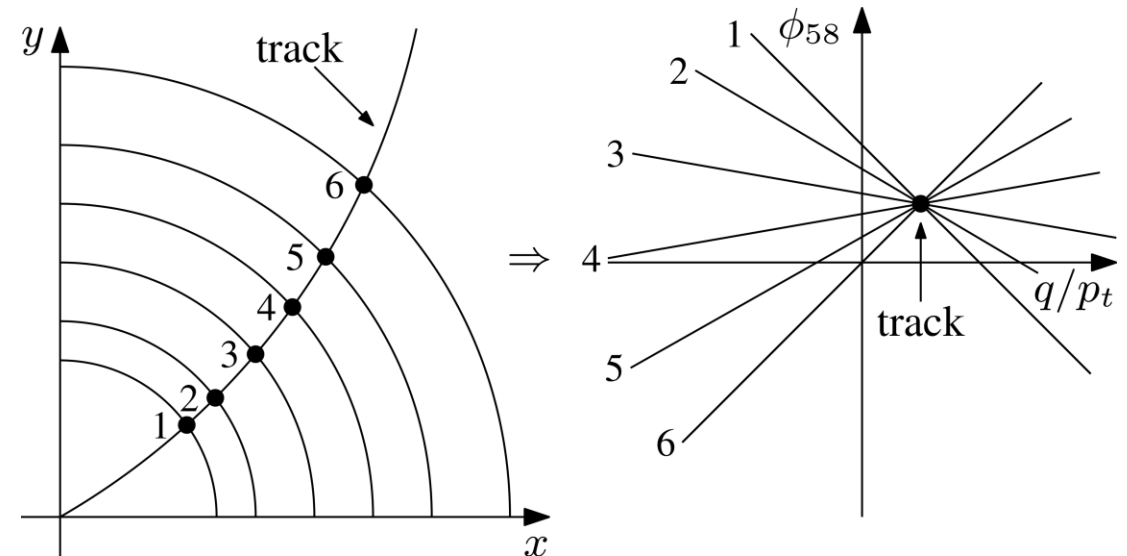# *Hough Transform*

Alexander David Morton
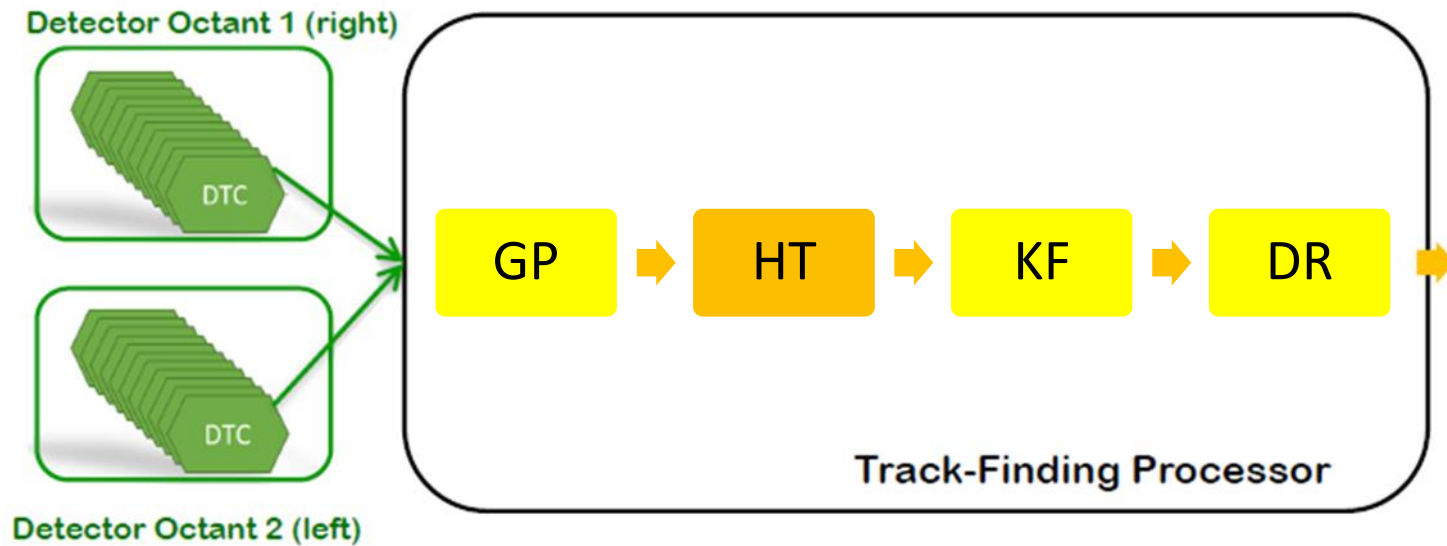
Hough Transform
Theory

- Use Hough Transform (HT) for primary fast search for tracks in r-φ plane.
  - Points in real space = lines in Hough Space/Points in Hough space = real space line.

- For high p_T charged tracks in uniform magnetic field along the z (beam-line) axis:
  - Trajectory in r-φ plane $\cong$ circular within the tracking volume.
  - (Assuming no energy loss).

- Stub radius $r_{58} = r - 58cm$ is used as:
  - In Hough Space, line gradients are given by $r \rightarrow$ always positive.
  - Transform to utilise larger phase-space $\rightarrow$ fewer fake/duplicated tracks.

Hough Transform
Algorithm

1. Calculate ϕ_{58} (ϕ at r_{58}) for each $^q/_{p_T}$.

2. Fill the stub into appropriate cells in a 32x64 array in $^q/_{p_T}$ x ϕ_{58}

3. Ignore $^q/_{p_T}$ values inconsistent with a stub's bend information (rough p_T estimate).

4. Define cells with stubs in at least 4 or 5 layers as track candidates.

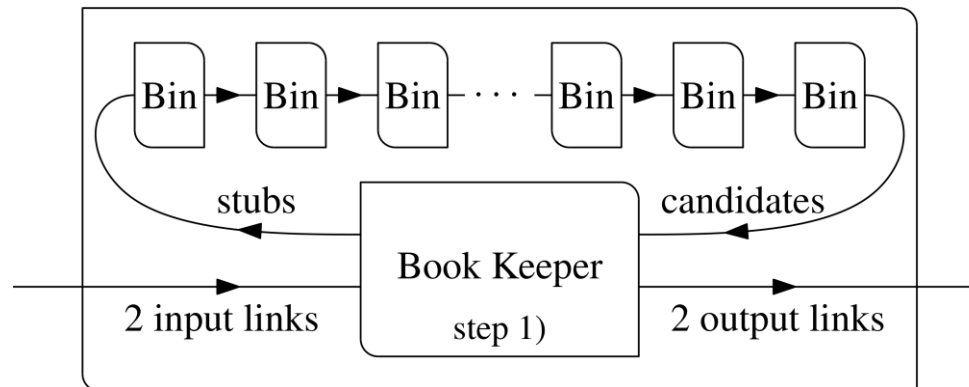   i.   4 layer threshold used to cope with barrel-endcap transition region or dead layers

Algorithm's simplicity → good for FPGAs

Hough Transform
Firmware Implementation - Overview

- Processes one stub per 240MHz clock in a two step fully pipelined design, :
  - Step 1: Filling of the HT array with stubs.
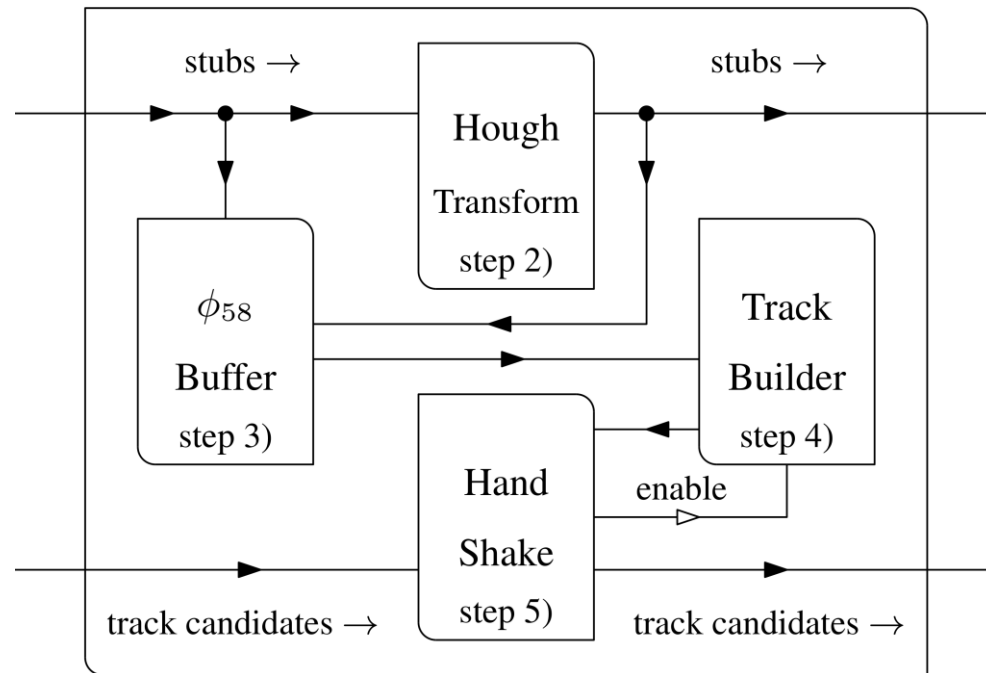 - Step 2: Readout of candidates.

- Book Keeper unpacks stub data from input links → propagates stubs to each Bin in turn.
- Track candidates propagated back to Book Keeper → transmits downstream.

Hough Transform
Firmware Implementation - Bin

- Each bin represents a q/p_T column in the HT array

- Hough Transform:
  - Gets ϕ_{58} at left boundary
  - Calculates ϕ_{58} at right boundary
- ϕ_{58} Buffer:
  - Duplicates stubs if it belongs to two cells.
- Track Builder:
  - Sorts stubs in ϕ_{58} cells.
  - Marks ϕ_{58} cells with stubs in at least 4/5 layers.
- Hand Shake:
 - Controls read-out of candidates

Hough Transform
Firmware Implementation: Truncation

- One HT array processes one new stub per clock cycle (240 MHz):
    - Implies many arrays working in parallel → per octant: $2\phi \times 18\eta = 36$ arrays

- 1025 ns max processing time allowed (determined by Time-Multiplexing period).
    - For 36 BX → process up to 216 stubs

- Local fluctuations are predominately from collimated high energy jets.
    - Load balancing required!
    - Output bandwidth increased by splitting chain of bins → 6 chains of bins.
    - Interleave chains of three different, non-neighbouring, non opposite η sectors.

- Stubs loss due to truncation in $t\bar{t}$ + 200 PU measured at per mille level.
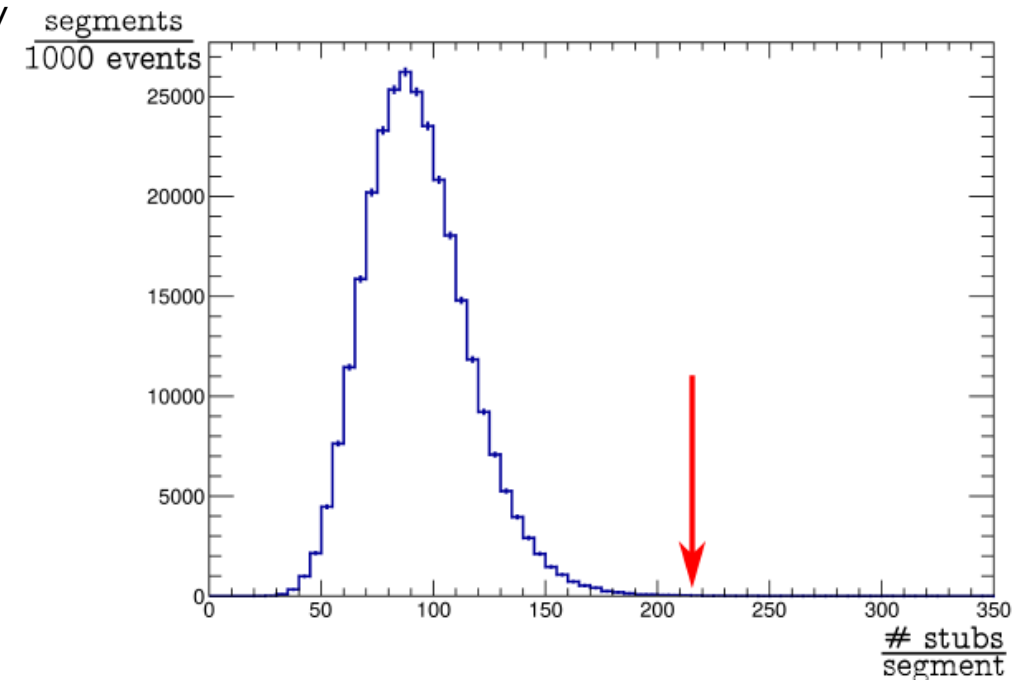
# Track Fitting with a Kalman Filter
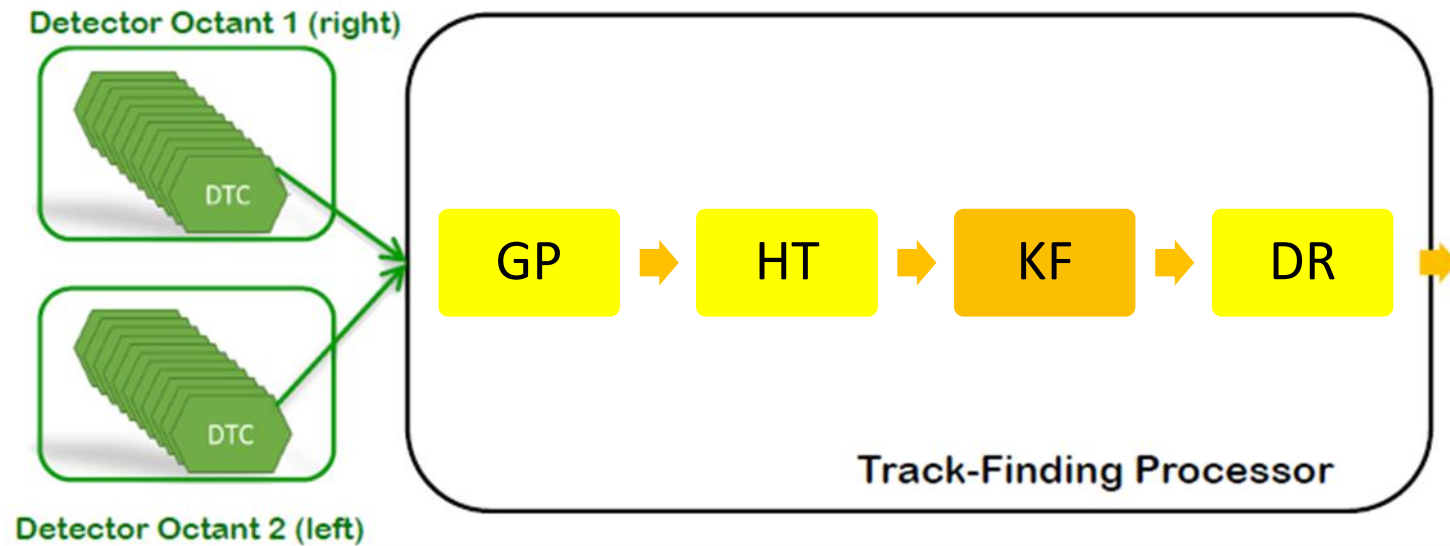
Kalman Filter
Context

- Coarse r-φ helix parameters out of the Hough Transform are used as initial variables for track finding.
    - Segment assignment also provides a good initial seed value.

- Kalman Filters have been widely used for offline reconstruction for many years, for many experiments …

Possible in online reconstruction in ~ μs on an FPGA?

Clue: See "Online track reconstruction using Kalman Filters on FPGAs" , Sioni Paris Summers, 1230, 7th March 2016, CTD/WIT 2017, LAL Orsay

Kalman Filter
Theory

- Initial coarse estimate of the track parameters are made from the HT candidate.

- Stubs are used to update the state following Kalman's formalism.
  - Decreases uncertainty in the state at each iteration.

- Weighting from relative uncertainties in the state and measurement control parameter adjustment.

- Layers can be skipped due to missing stubs or incorrect stubs.

- Multiple stubs on a layer are propagated and ranked.
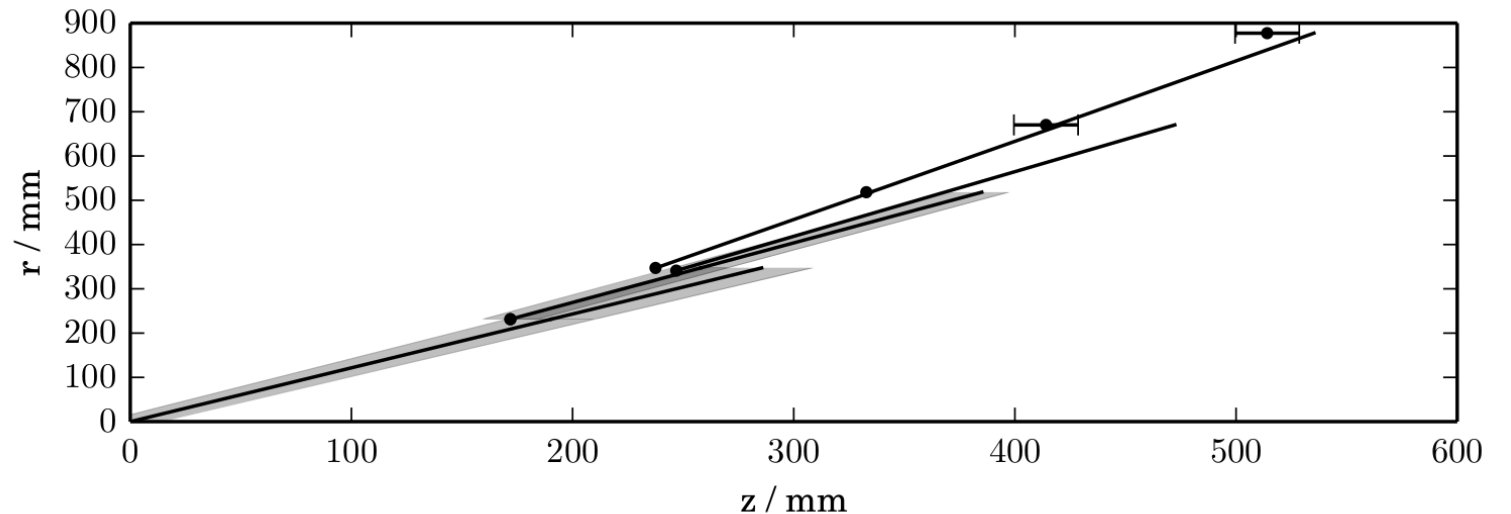
*For a greater treatment, see R. Frühwirth,
"Application of Kalman filtering to track and vertex fitting",
Nucl.Instrum.Meth. A262 (1987) 444-450.*

Kalman Filter
Motivations for use

- Over half of candidates found by the HT are not genuine tracks or have at least one incorrect stub.

Kalman Filter is capable of removing bad stubs …

… but is it suitable for an FPGA? Yes!

Matrices used are:
- *Small.*
- *Size independent of number of measurements.*
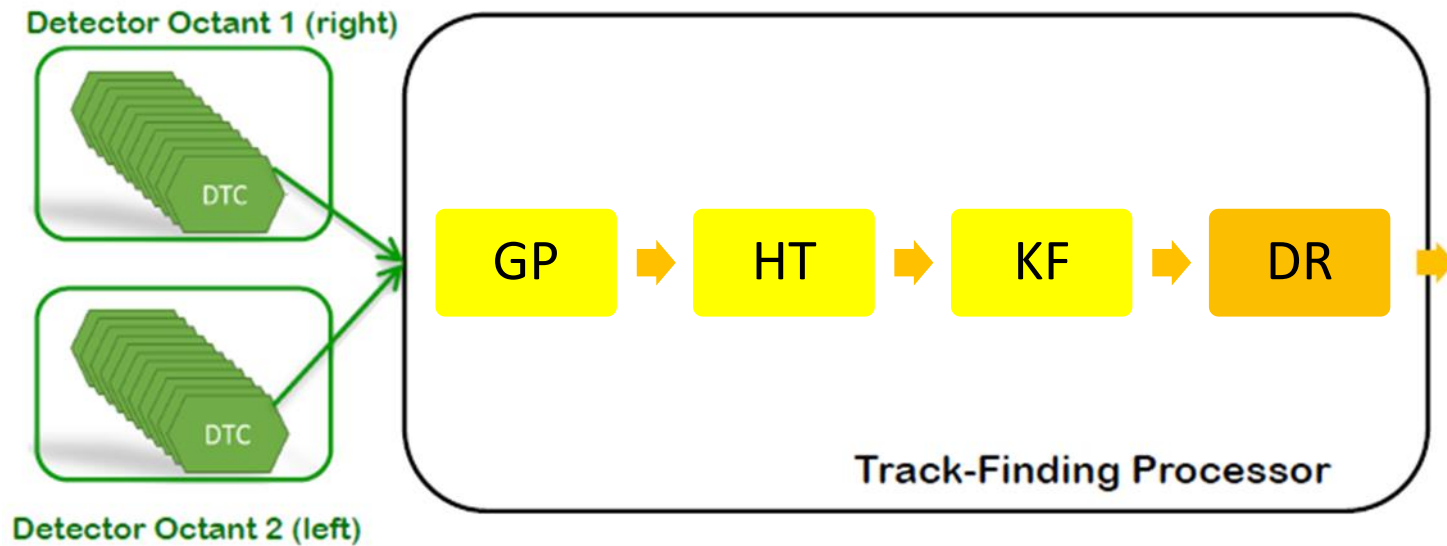- *Matrix inversion used is of a small matrix.*

Kalman Filter
Firmware Implementation

- Starting with the seed state, stubs are used to update the state iteratively.
- Kalman Filter performed on input state using a stub.
  - Up to four best states are kept → surviving states presented to final state selector.
- Final fit is always performed after a fixed period:
  - No truncation.
  - Allows readout of partially filtered states (dense jets with many candidates and stubs per candidate).

From HT → **Stub RAM** → **State/Stub Pairing** → **Kalman Filter** → **State Filter** → **To DP**

State FIFO
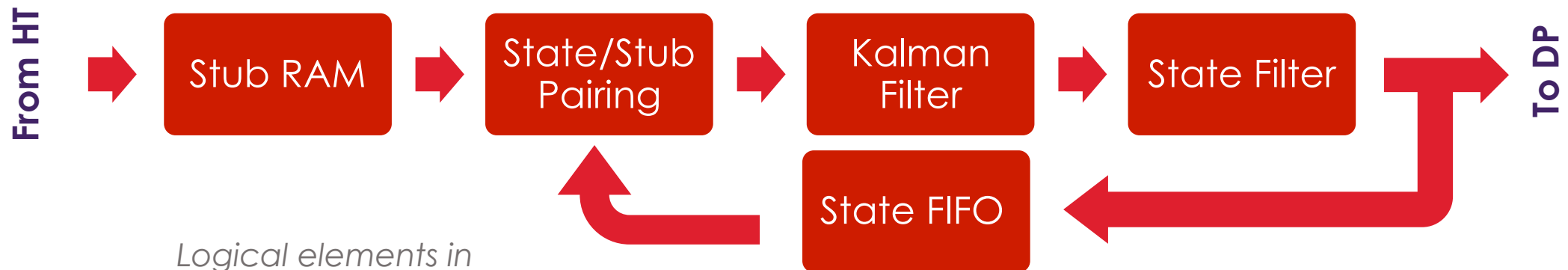
Logical elements in Kalman Filter node

Alexander David Morton

Duplicate Removal

Duplicate Removal

- Hough Transform sometimes reconstructs single particles as several duplicate tracks.

- Naively one would expect the need to compare pairs of tracks to see if they are the same as each other.

- Understanding how the Hough Transform produces fakes provided the basis for more elegant and subtle duplicate removal algorithm …

- Duplicates can also originate from sector segmentation:
  - These can be removed by cleaning the segment boundaries.

Duplicate Removal
Algorithm

- Best to illustrate by example:

Outermost stub

Middle stub

Innermost stub

Cells in Hough Space
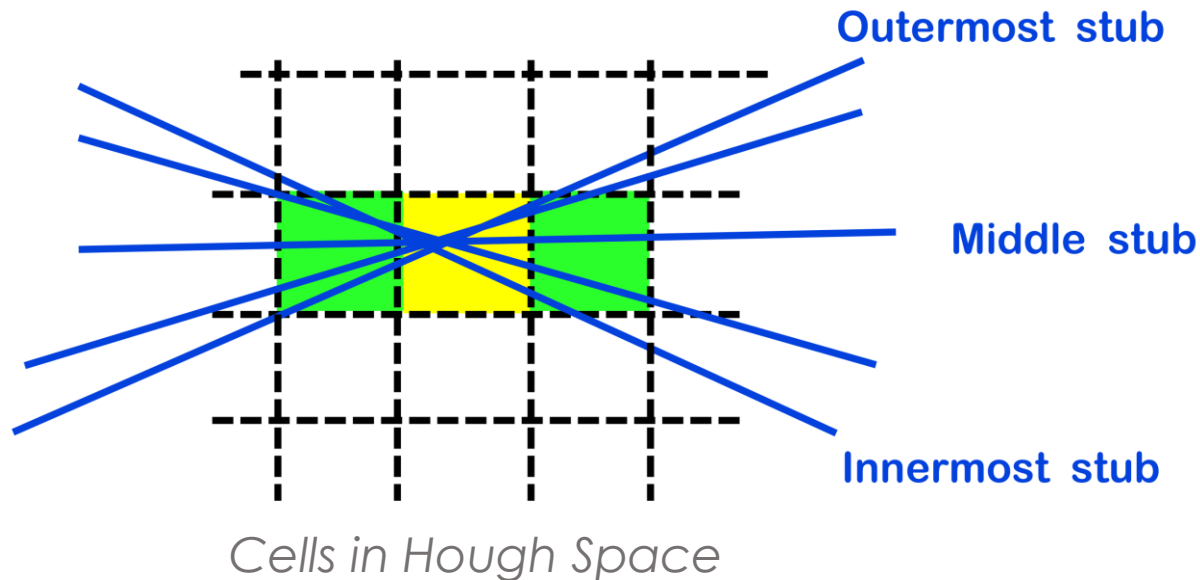
- HT: 5 stubs from single particle produce 3 candidates (in the green and yellow cells).

- 3 candidates contain same stubs – fitted with identical helix parameters.

- Fitted helix parameters give greater resolution:
  - All hits fit into yellow cell regardless of original HT cell.
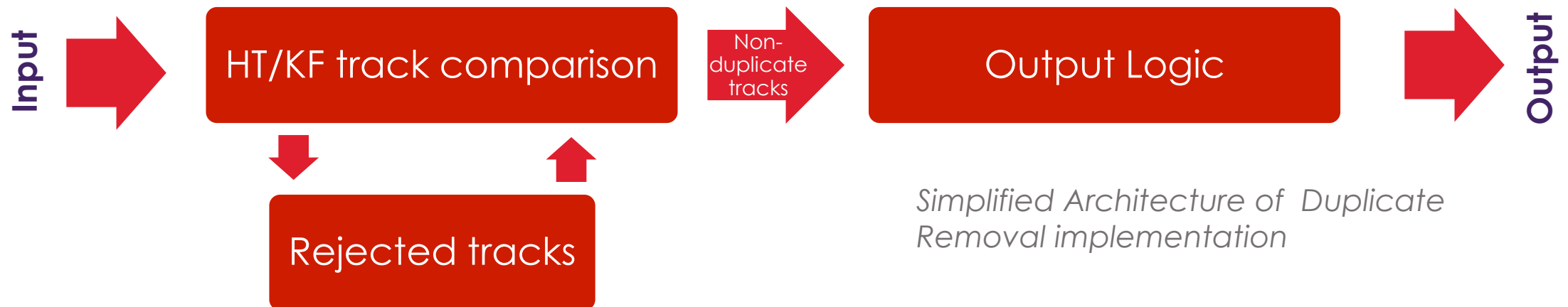
- Subtleties arise from resolution effects.

- i.e. HT and precise KF are compared → inconsistencies removed.

No need to compare pairs of tracks!

Duplicate Removal
Firmware Implementation

i.   Fitted helix parameters are compared to HT helix parameters.
  • Tracks consistent in both the HT and KF are marked and forwarded to the output.
  • Inconsistent tracks rejected.
ii. Incorrectly rejected tracks recovered (not shown on diagram – further details in backup).

The minimal resource usage strategy used has resulted in firmware that can be integrated in the same board as the KF.

Input → **HT/KF track comparison** → Non-duplicate tracks → **Output Logic** → **Output**

Rejected tracks

Simplified Architecture of Duplicate Removal implementation

Demonstrator System
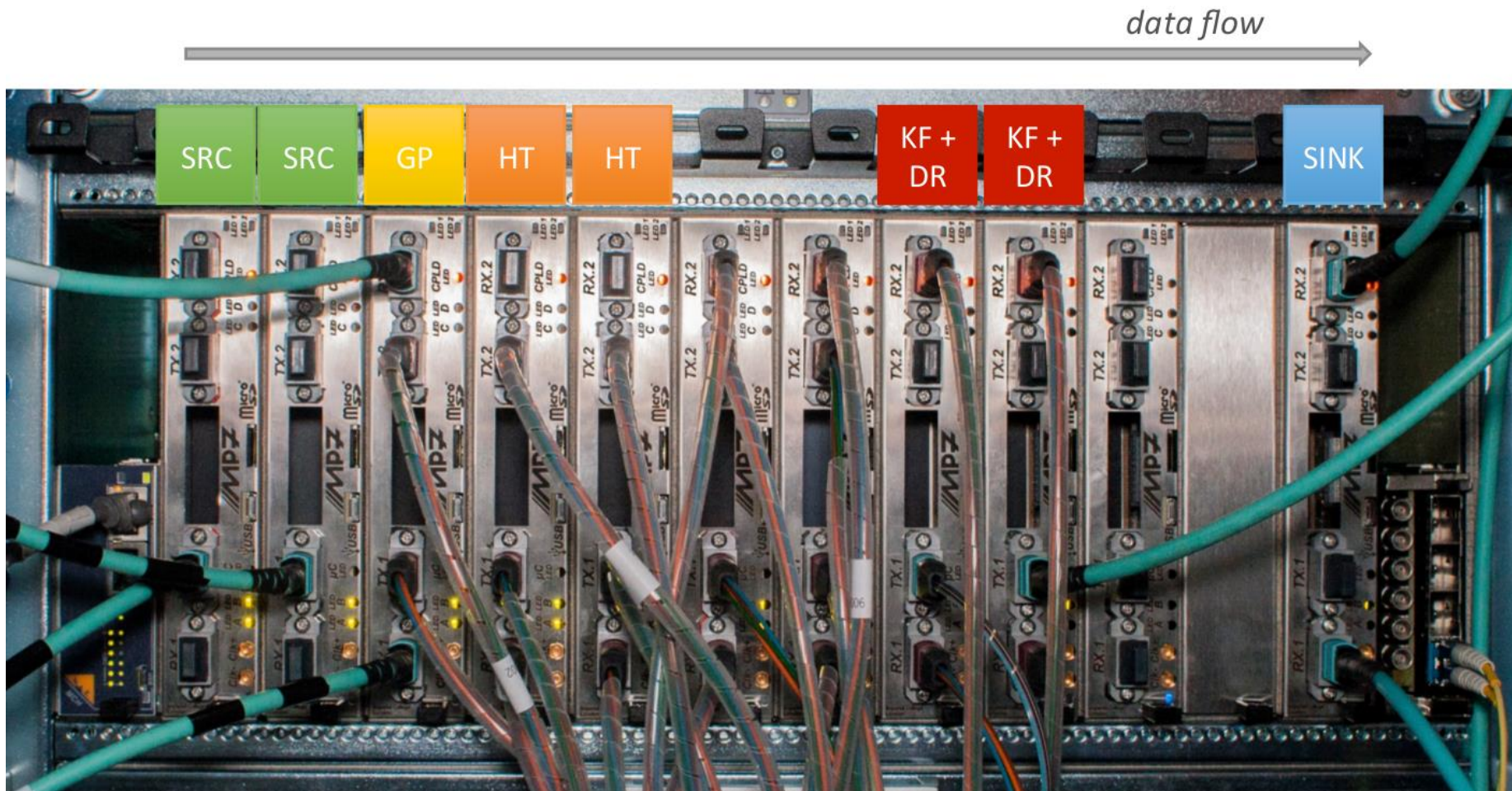
Demonstrator system at Tracker Integration Facility, B186, CERN

Demonstrator System
Hardware platform

- Hardware demonstrator corresponds to a single TFP:
    - Reconstructs tracks in a φ octant ($\frac{1}{8}^{th}$ of tracker solid angle).
    - Reconstructs tracks for 1 LHC event in 36 (TMUX factor)

- Implemented on the Imperial Master Processor, Virtex-7 (MP7):
    - High performance FPGA (Xilinx Virtex-7)
    - 72 in/out optical bandwidth @ 10 Gbps
    - Phase-1 L1 Calorimeter platform:
        - Well tested.
        - Existing infrastructure.
    - Algorithm isolation inside firmware:
        - Generic algorithm interface.
        - Easy swapping of algorithms → reduced development time.

IMPERIAL MP7

Demonstrator System
Hardware Setup
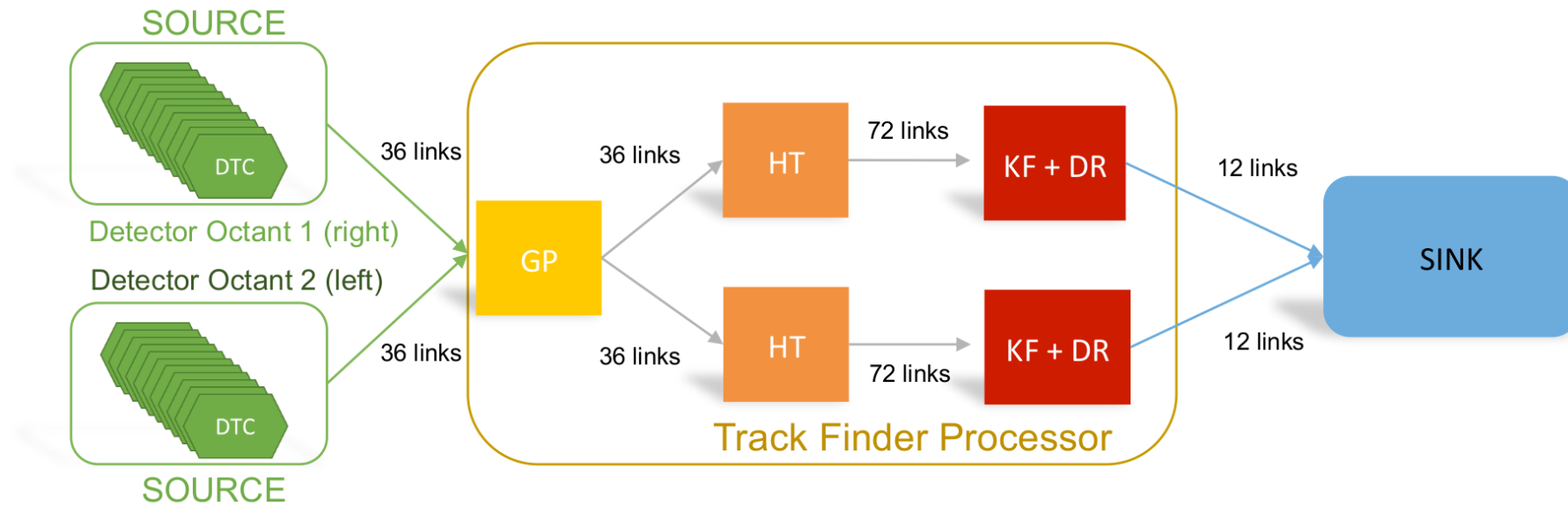
- Each TFP is comprised of logical units self-contained on a single board → 5 boards required.
- Available resources are not a limit to scale/performance of what we want to implement.
- Can extrapolate future board resources, allowing final system demonstration with current technology.
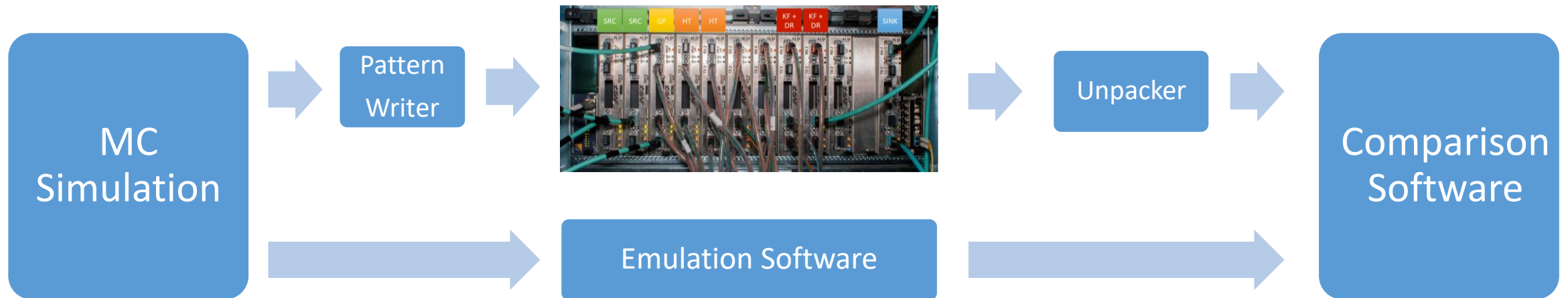
Demonstrator System
Source and Sink

- Two MP7 boards used for a detector octant each – injects 30 consecutive events into TFP.

- One MP7 board used for a sink – capture and store output (up to 30 events before readout).

- Eight boards (five for TFP + two sources + one sink) required to demonstrate complete system.
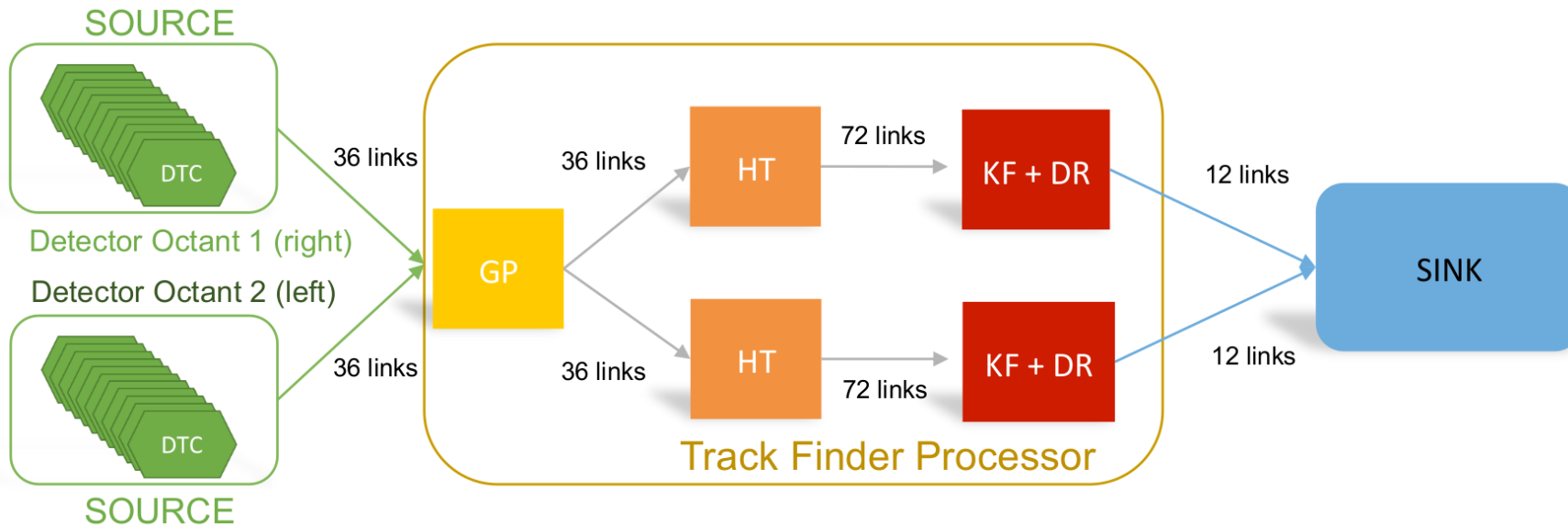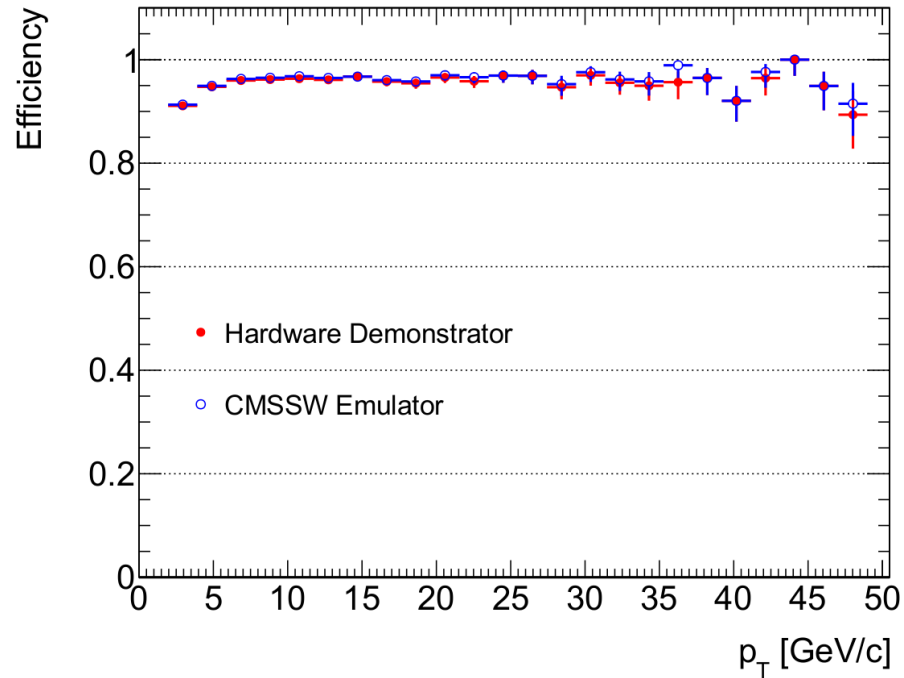
Demonstrator System
Overview

- Stratagem: Compare hardware output directly with software
  - Directly measure hardware performance!

- Both hardware and emulator receive stubs as input and output tracks
- Run on MC at up to pileup of 200.

Demonstrator System
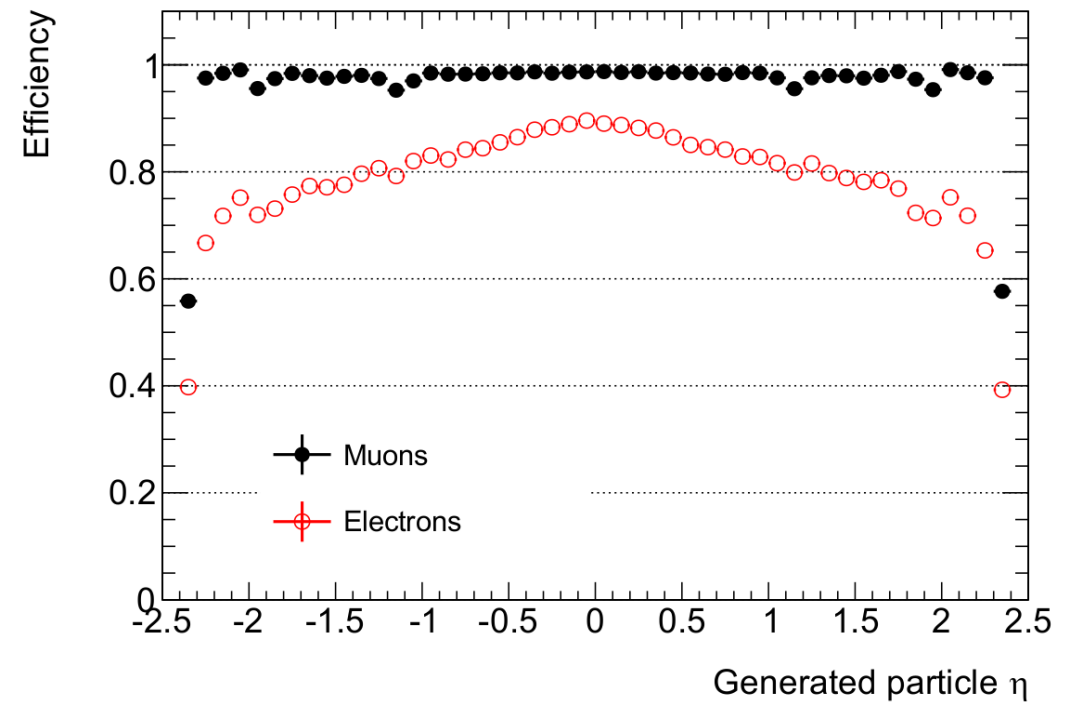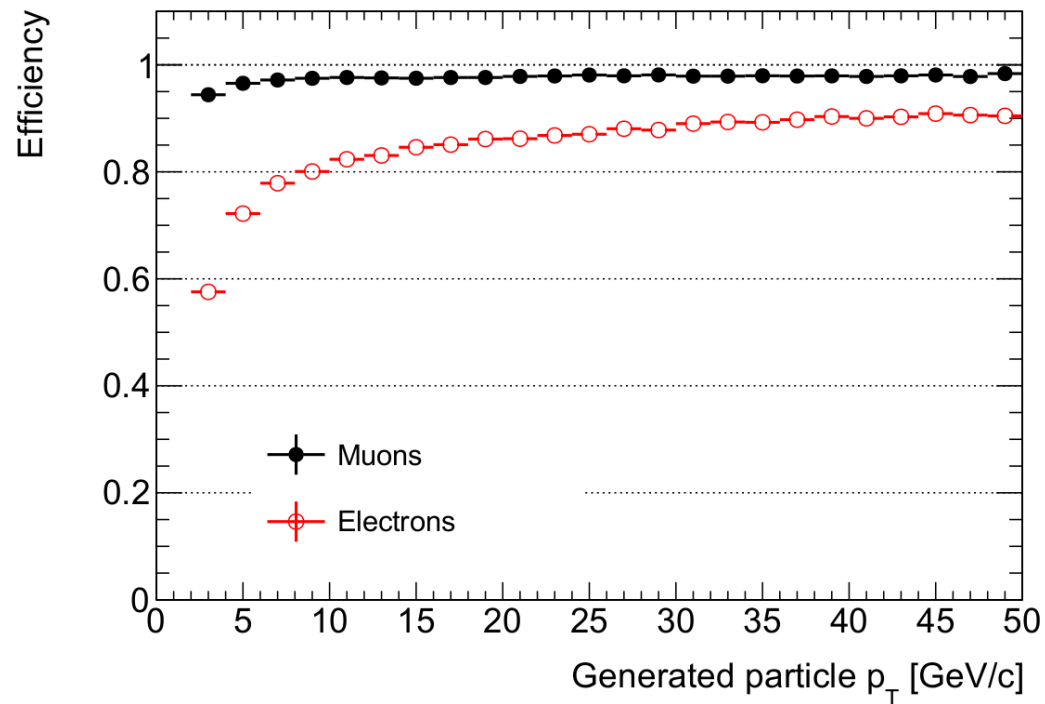## Results – Track Finding Efficiency
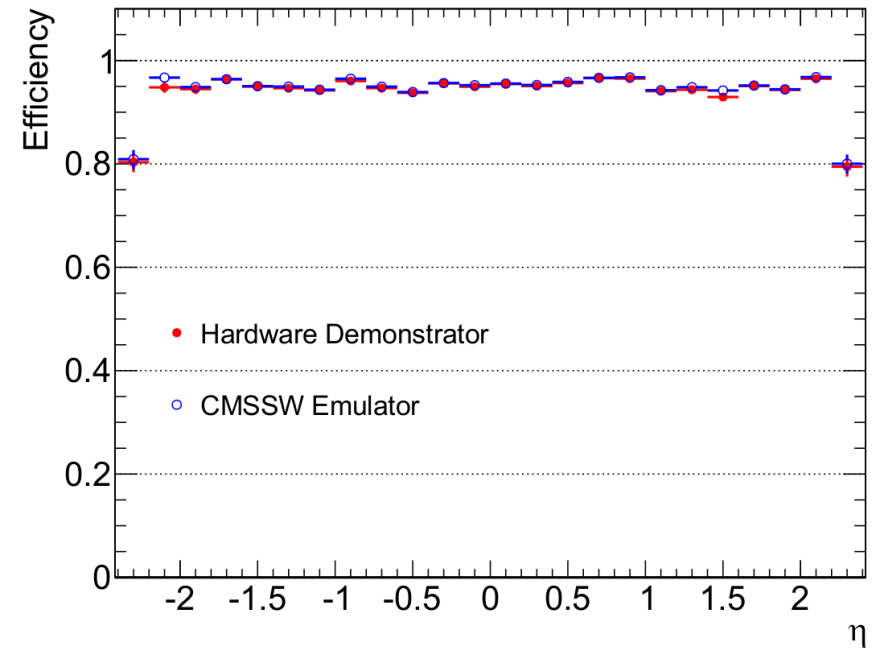
Track finding efficiency for $t\bar{t} + 200\ PU$ (1800 events) – hardware and emulation

	Efficiency (%)	Avg. Rate	Matched Tracks
HW	94.5	76.5	98.7
SW	94.8	79.4	

Demonstrator System
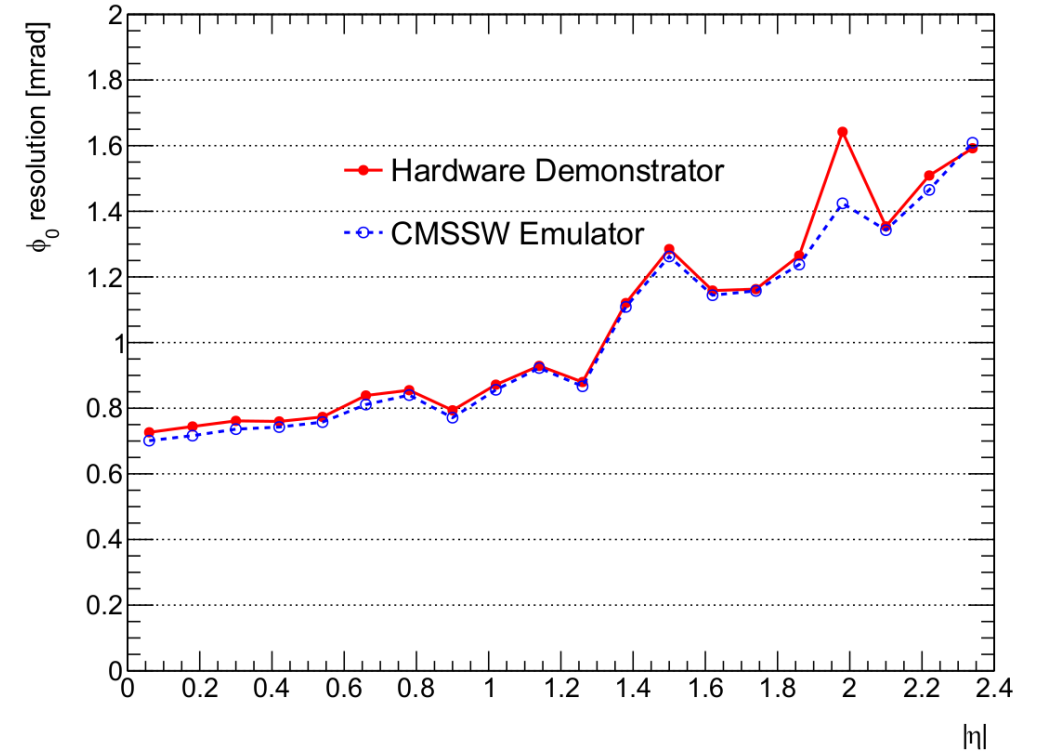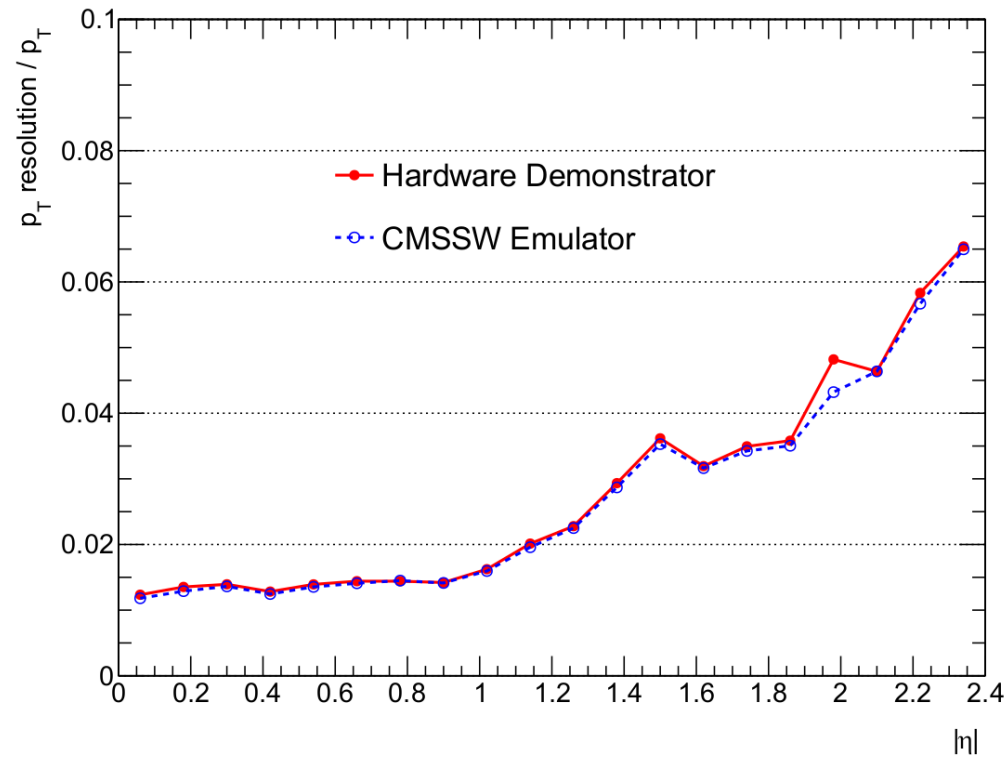## Results – Track Finding Efficiency

Track finding efficiency for electrons and muons from $t\bar{t} + 200\ PU$ (1800 events) – emulation

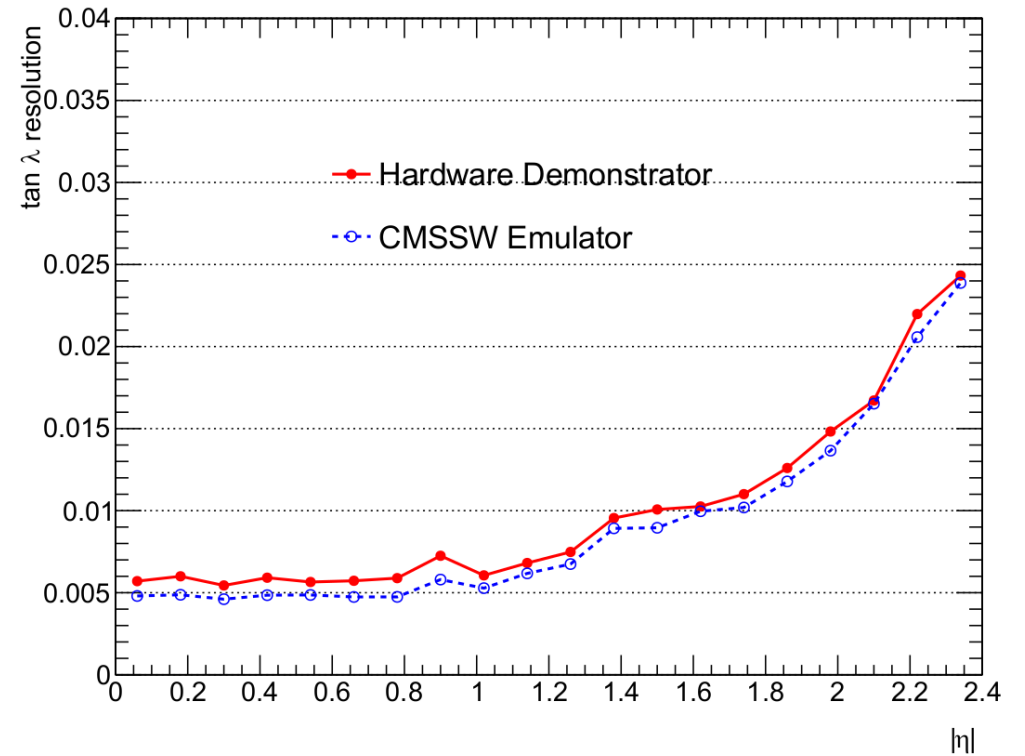# Demonstrator System
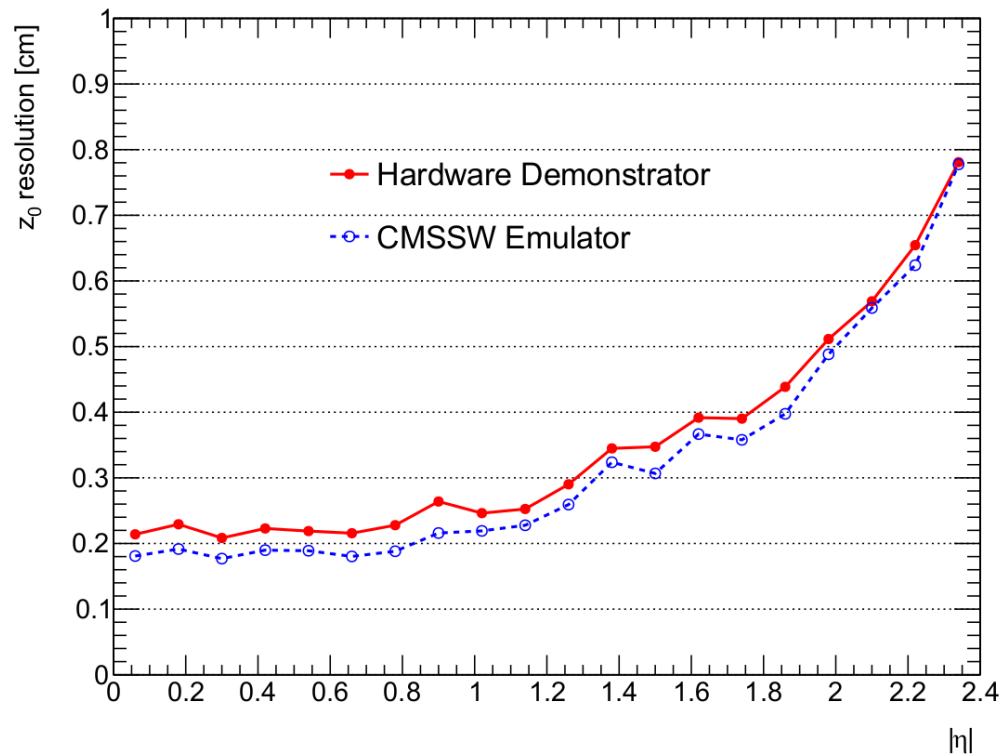## Results – Track Parameters

Helix parameters from $t\bar{t} + 200\ PU$ (1800 events) – hardware and floating point simulation

Demonstrator System
Results – Track Parameters

Helix parameters from $t\bar{t}$ + 200 PU (1800 events) – hardware and floating point simulation

Demonstrator System
Results – Resource Usage

	LUTs [10^3]	FFs [10^3]	BRAM 36	DSP 48
GP	128	228	198	1560
HT	284	378	1368	0
KF + DP	382	794	1750	5040
Demonstrator Total (exc. infra.)	794	1400	3316	6600
Virtex 7 690	433	866	1470	3600
Kintex Ultrascale 115	633	1266	2160	5520
Virtex Ultrascale+ 11P	1296	2592	1970	9216

- GP + HT, and the KF + DR could each fit inside Ultrascale and Ultrascale+ generation chips.
- Resource usage of the system is smaller than five Virtex 7s (used to demonstrate 1 TFP):
  - Highly over-engineered system to eliminate truncation effects → wasted processing bandwidth.
  - Anticipated that a factor of 4 reduction in resources is realistic.

Demonstrator System
Results – Latency

- Latency measurements of full demonstrator chain have been measured:
  - Each block independently.
  - Total chain.
  - Both give identical results.

- Measurements include optical link traversal time and serialisation/de-serialisation (SerDes).

- Latency is fixed, regardless of:
 - Pileup.
  - Event occupancy.

System Latency	Latency [ns]
SerDes & optical length 1	143
GP	310
SerDes & optical length 2	144
HT	1025
SerDes & optical length 3	129
KF + DP	1658
SerDes & optical length 4	129
Total: First out – First in	3538
Last out – First out	225
Total: Last out – First in	3763

Demonstrator System
Results – Cooling Loop Failure

- Simulated cooling loop failures in software:
  - Efficiency loss recovered by locally reducing number of tracker layers from four to five.

- Modest increase in data rate:

No module loss	With Cooling Loop Failure	
	Pre-recovery	Post-recovery
337	304	347
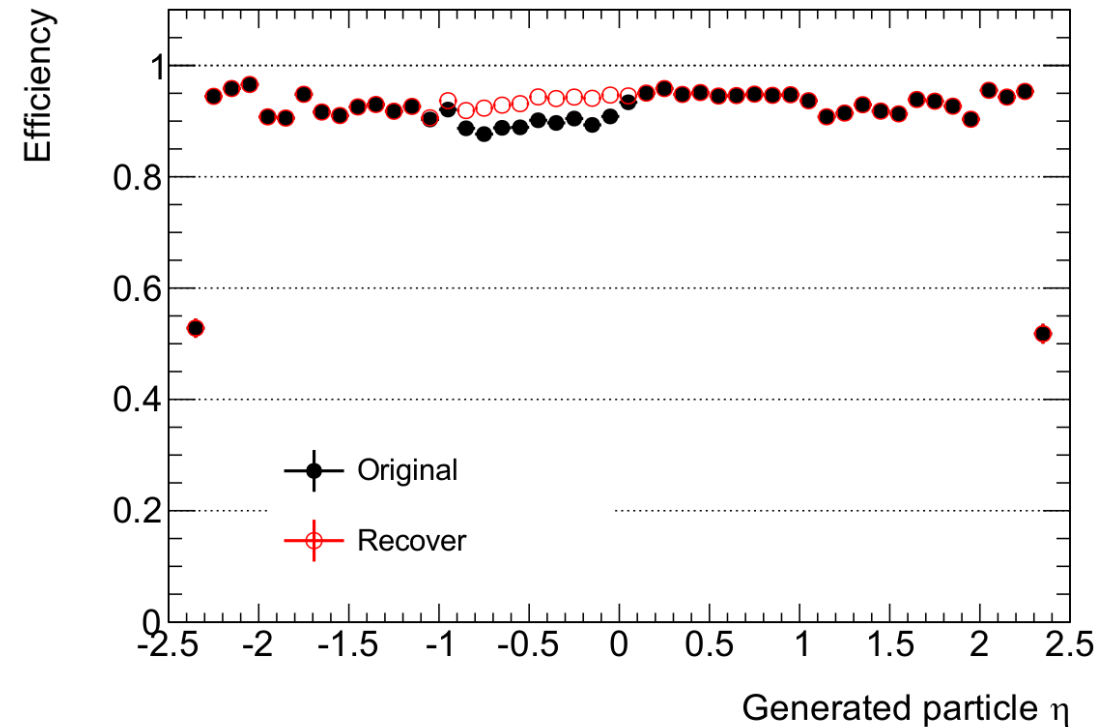
Conclusions

- An ambitious design for a track-finding system for the CMS Phase-2 Outer Tracker at the HL-LHC has been developed using time-multiplexing and FPGA technology.

- Successfully met CMS requirements with today's technology in our highly flexible and scalable demonstrator system:
  - Hardware measured performance with excellent results up to 200 pile-up.
  - Total latency < 4.0 μs demonstrated.

- Confident that algorithms will continue to evolve and develop in collaboration in the coming years.

Time-Multiplexed Track Trigger Team

R. Aggleton, L. Ardila-Perez, F. A. Ball, M. Balzer, J. Brooke, M. Caselle, L. Calligaris,

D. Cieri, E. J. Clement, G. Hall, K. Harder, P. R. Hobson, G. M. Iles, T. James, K. Manolopoulos,

T. Matsushita, A. D. Morton, D. Newbold, S. Paramesvaran, M. Pesaresi, I. D. Reid, A. W. Rose,

O. Sander, C. Shepherd-Themistocleous, A. Shtipliyski, T. Schuh, S. P. Summers,

A. Tapper, I. Tomalin, K. Uchida, P. Vichoudis, and M. Weber

Thanks to all involved!

BACKUP:

Alexander David Morton
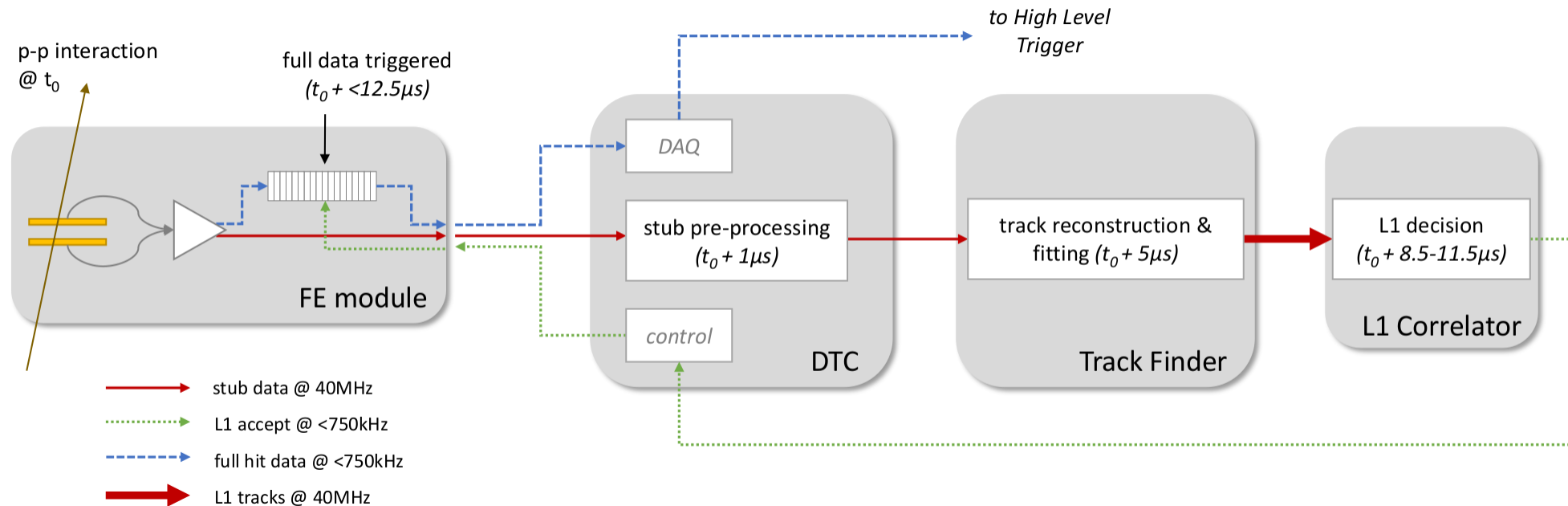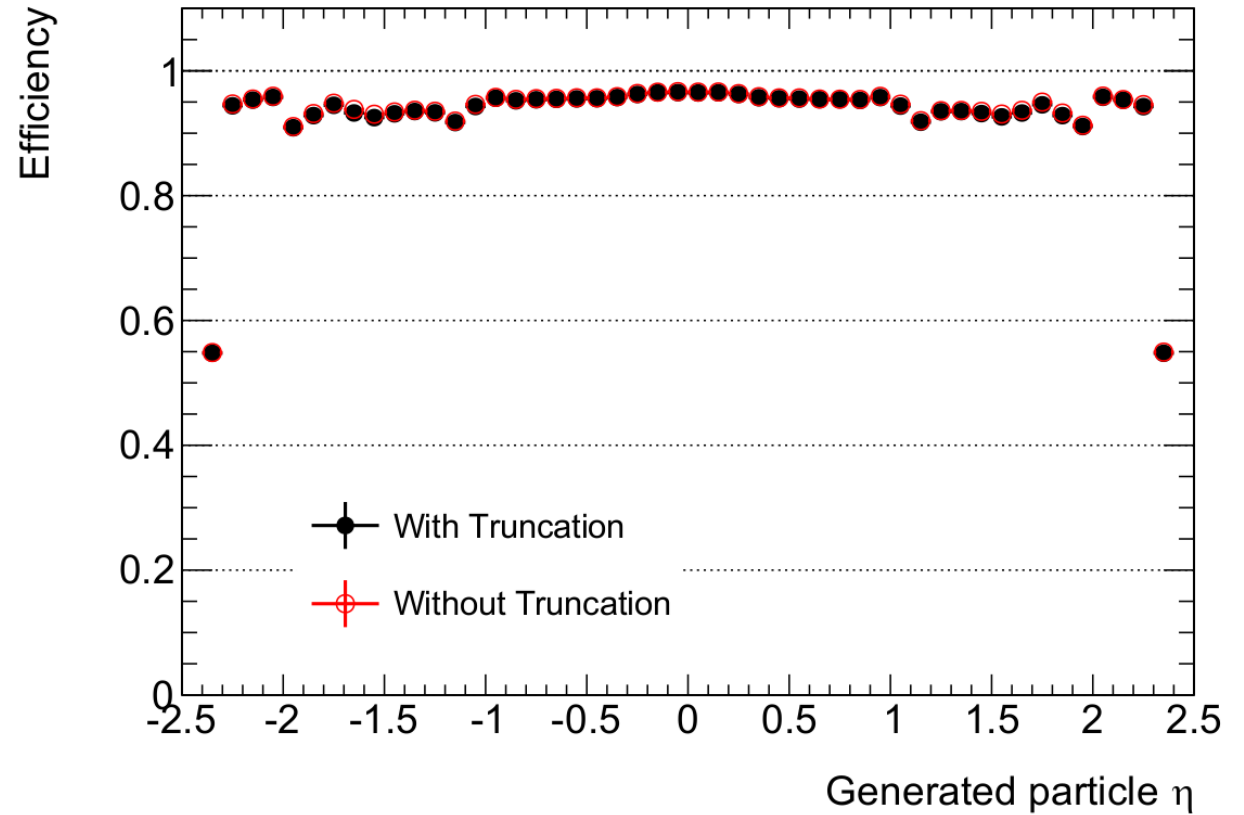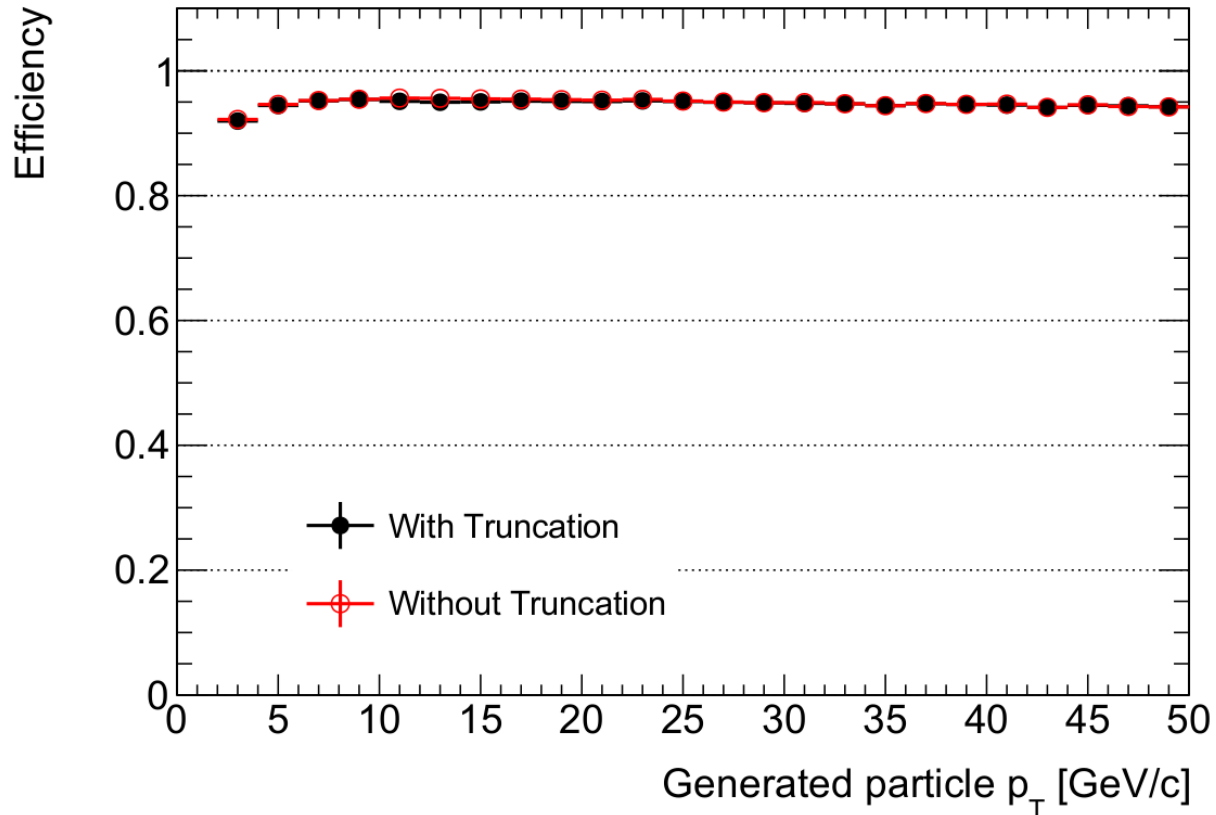
Dataflow and Latency Requirements Illustration

Illustration of dataflow and latency requirements from pT-modules through to the off-detector electronics dedicated to forming the L1 trigger decision.

Demonstrator System Results – Truncation

# Duplicate Removal Algorithm (Postscript)

- Resolution effects cause a loss of a few percent of efficiency.
- 2nd pass through rejected tracks recovers this:
  - Tracks with helix parameters not corresponding to the HT cell of an accepted track from the 1st pass are unlikely to be duplicates.
  - These tracks are rescued.

Alexander David Morton
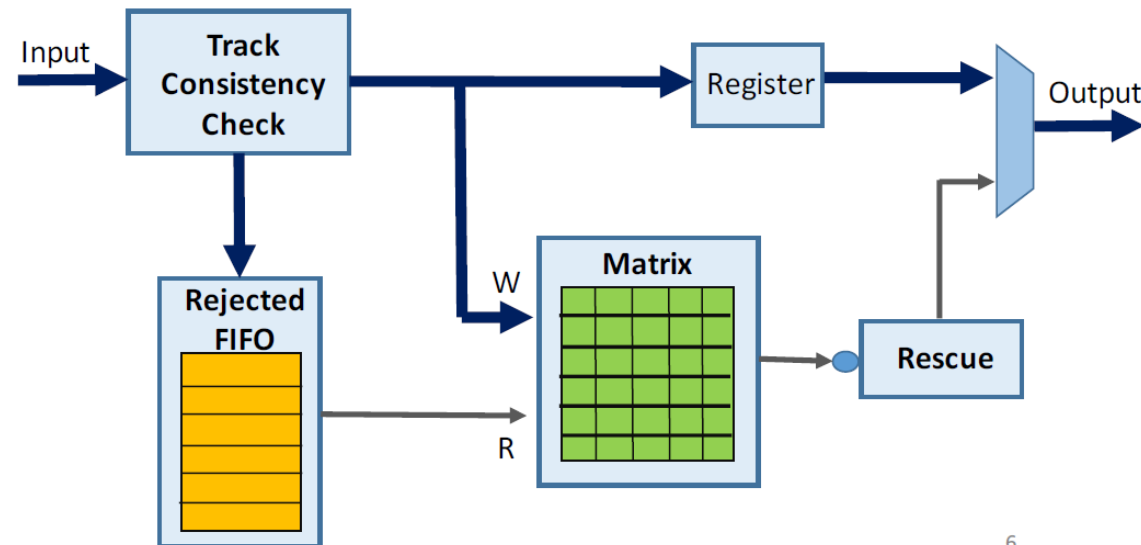
Track finding in the Barrel-Endcap Transition Region

- Originally HT required stubs in at > 4 layers.
  - → efficiency loss in barrel-endcap transition region.

- Relaxing layer requirement from a minimum of 5 to 4 layers in these sectors recovers the efficiency previously lost.
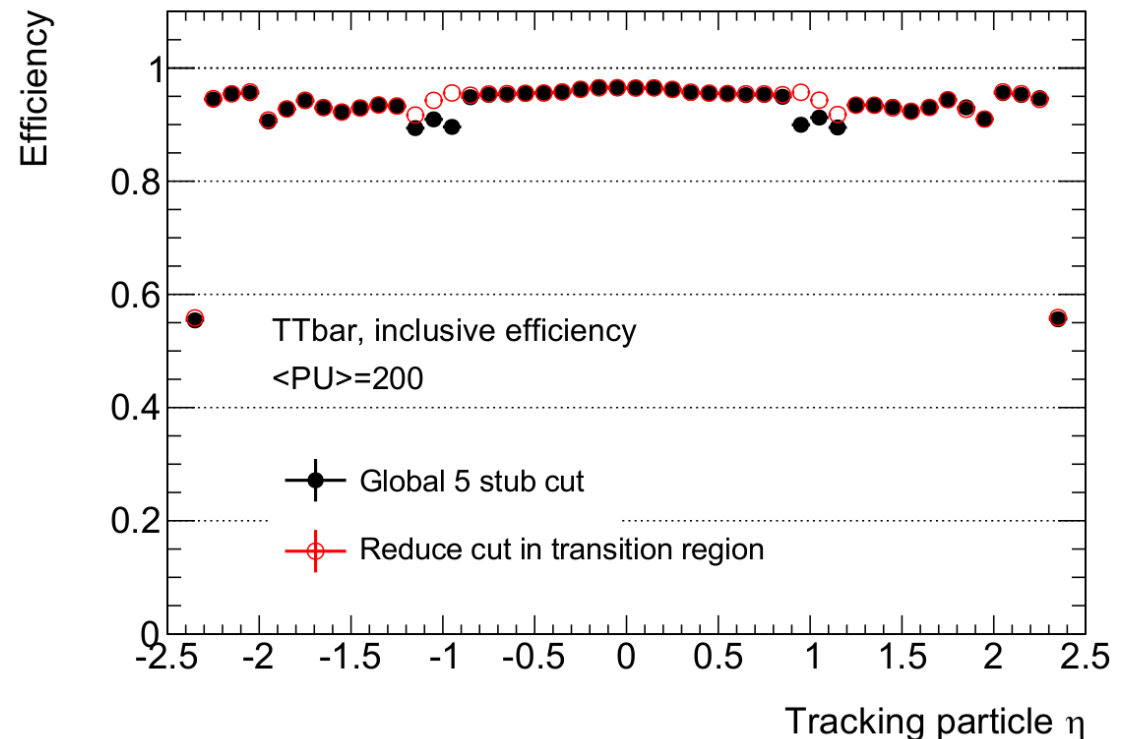
Track Parameters
z_0 resolution improvements

- z_0 resolution not as precise as the expected online performance of single muons at PU of 140 in the Technical Proposal for the Phase-II Upgrade of the CMS.

- Result of choosing 12-bit and 10-bit encoding of r and z coordinates of stubs.
  - Implementation rather than design flaw.

- Simulation shows adding 2 extra bits for r and z in the barrel and endcaps respectively recovers optimal resolution.
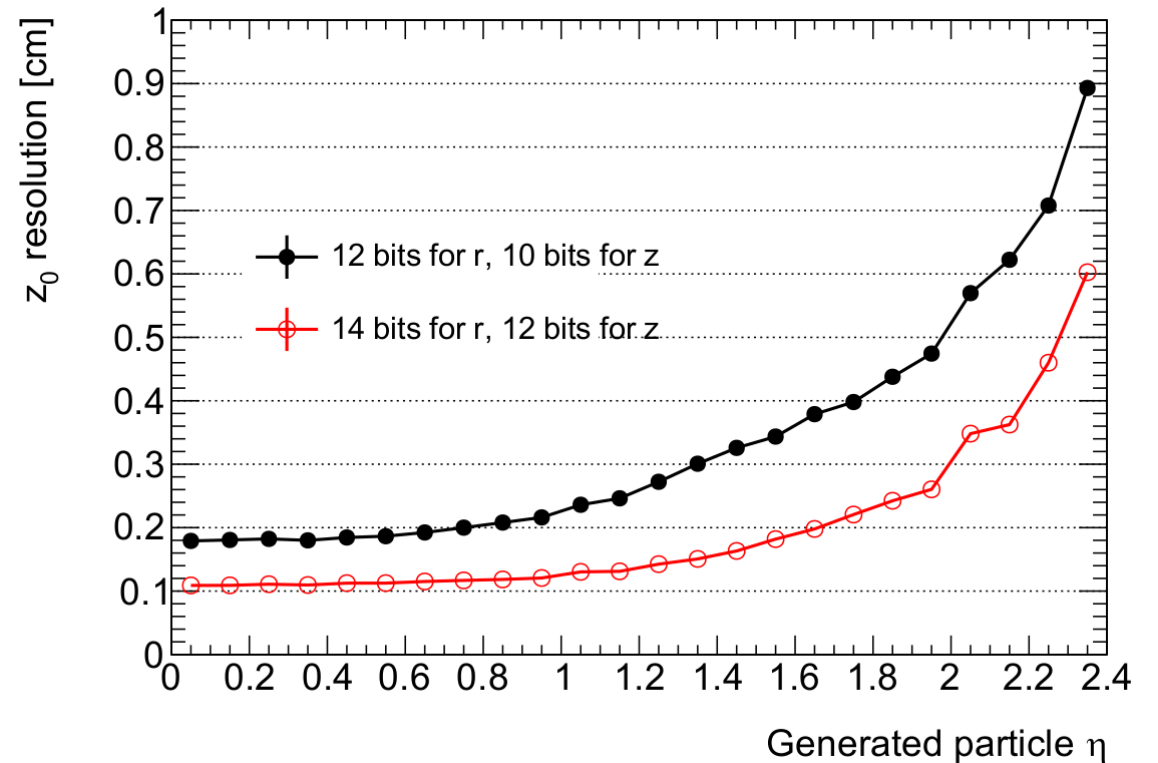
Track finding at 2 GeV/c

- q/p_T range is factor 1.5 times larger, so use factor 1.5 more cells on q/p_T axis of HT.
  - Increases output rate by factor 2.2 and requires 50% more resources in hardware.

- Some efficiency loss at low pT caused by multiple scattering:
  - Some stubs outside of expected HT cell → Merge neighbouring cells at high q/p_T.
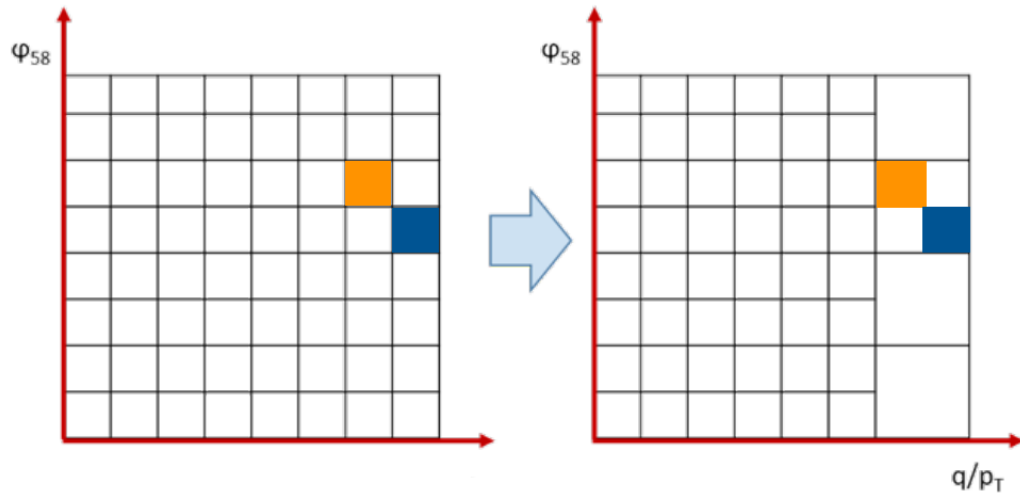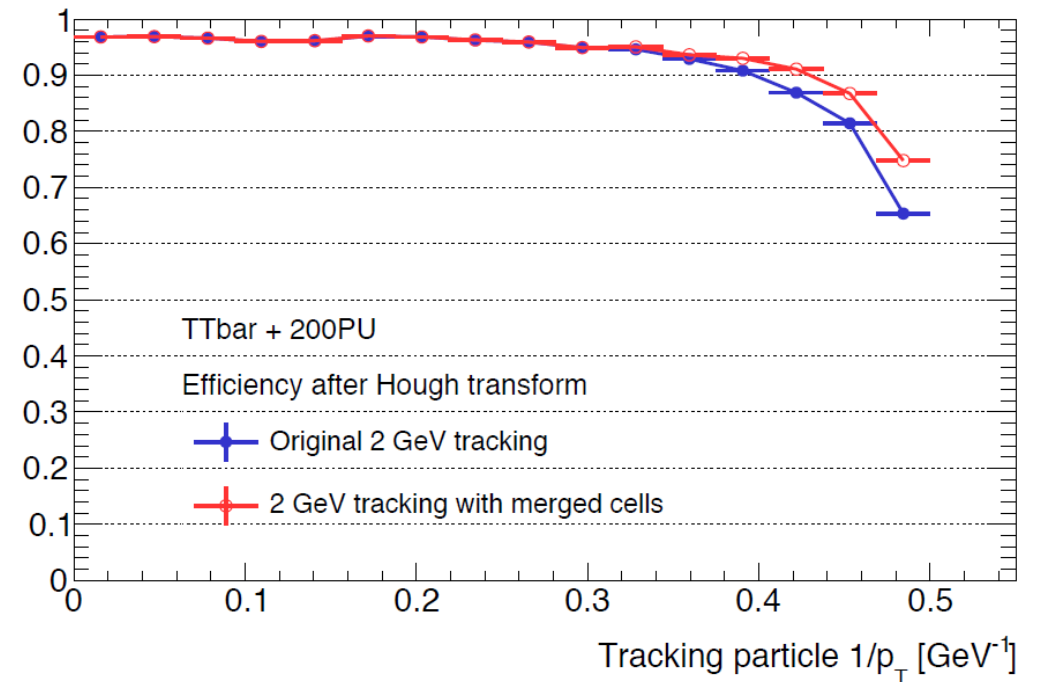   - Further ate increase of 10%.

Illustration of high q/p_T cell merging.

System Scaling Baseline Full System

- Scale up demonstrator on following assumptions:
  - 16 Gbps links (66/64b encoding) → scales TMUX period to 18 BX.
  - 3x Kintex Ultrascale 115 FPGAs per TFP or equivalent.
  - 144 ATAC boards (8 octants x 18 time slices).
  - 16 ATAC shelves, 8 racks.

- 18 BX architecture:
  - Same volume of data at twice the rate.
  - Naively need double demonstrator's resources → more intelligent solutions possible.

- Using current pricing/quotes or past experience:
  - Board costs: ~ 3.5 MCHF (per board: FPGA ~ 10.9k, Optics~ 4.8k, PCB/components ~ 2.6k)
  - Infrastructure costs: ~ 0.8 MCHF
  - Costs include: 85% yield and 10% spares, shelves, hub cards, patch panels, PSU, fibres and PCs
  - Total Cost: ~ 4.3 MCHF