Tracklet Approach to Level-1 Tracking for CMS at the HL-LHC

Margaret Zientek (Cornell University)

On behalf of the CMS Collaboration

CTD/WIT – March 2017

Introduction

- High Luminosity LHC (HL-LHC) upgrade planned for 2025
  - Peak luminosity 7.5 x 10^{34} $cm^{-2}s^{-1}$
  - Average pileup (PU) of 140-200

Huge amount of data → challenging environment for CMS

- New CMS tracker will have triggering capability → **L1 Tracking**
- L1 (hardware trigger) tracking helps deal with amount of data
  - Enhanced lepton ID, vertexing, track isolation

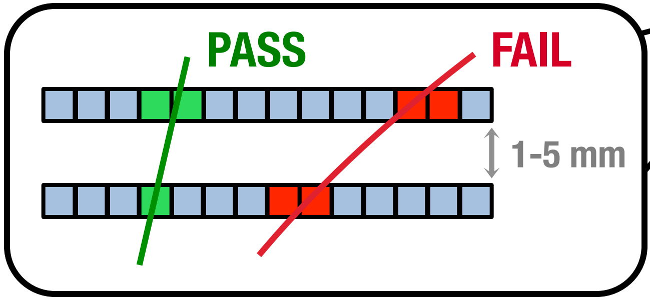
- Three CMS approaches to L1 Tracking: AM, TMTT, Tracklet
- This presentation: **Tracklet approach**
  - Overview of the algorithm
  - Performance results from simulations
  - Firmware implementation & hardware demonstrator results
  - Projections for full system

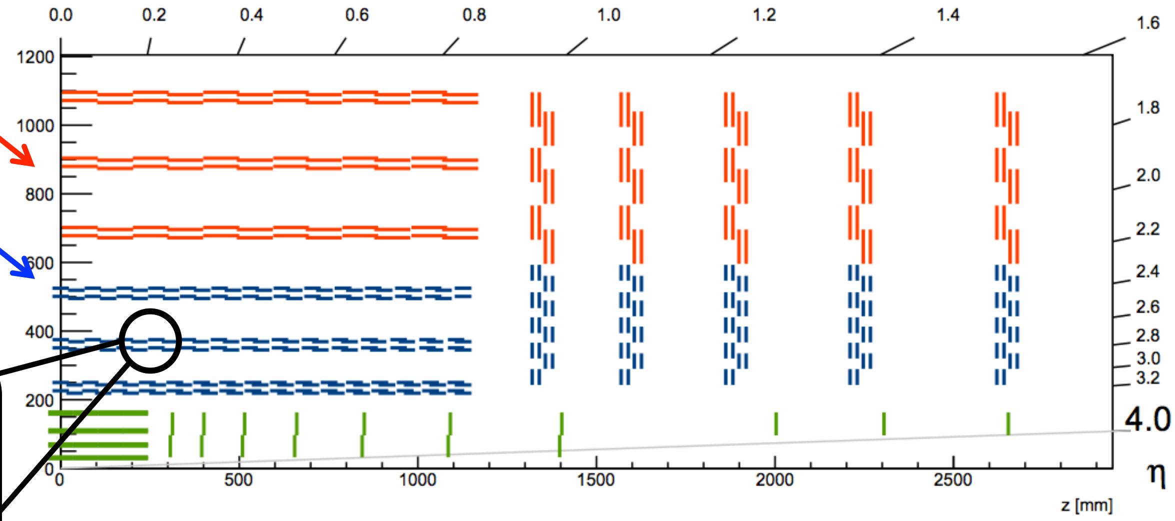
CMS Tracker of the HL-LHC

- Tracklet results with flat barrel (tech. proposal) geometry

- Two types of p_T modules:
  - **2S** (strip-strip)
  - **PS** (pixel-strip)

Stub





- Correlated pair of clusters
- Consistent with p_T > 2 GeV tracks

Introduction to the Tracklet Approach

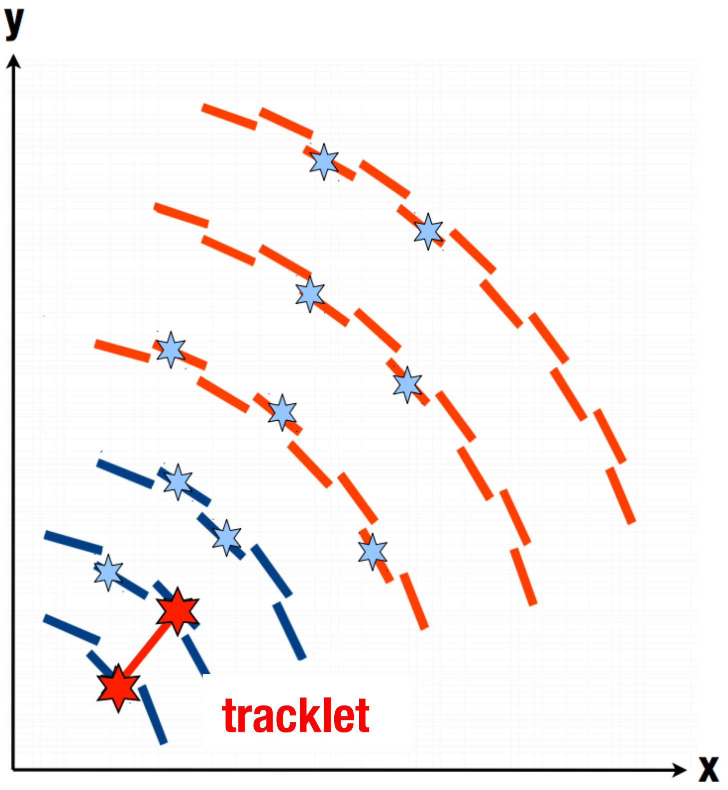
- Minimal hardware system based on **commercial FPGAs**
  - FPGAs are ideal for fast tracking…
    - Increasing capabilities
    - Programming flexibility

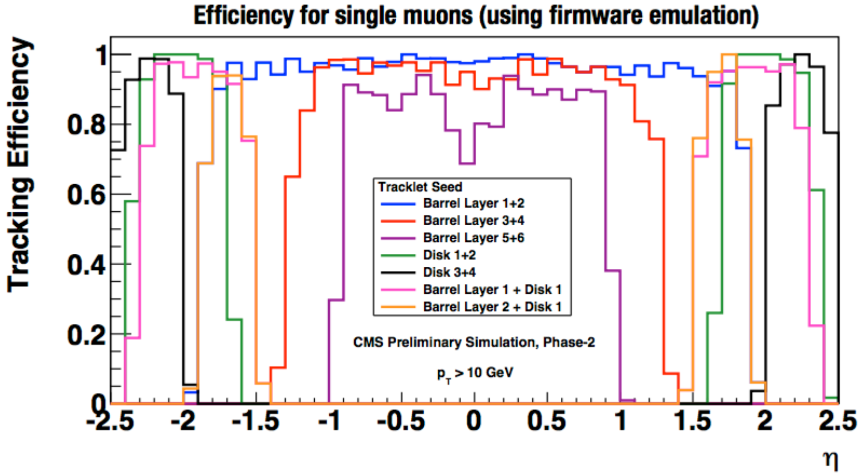
- Tracklet algorithm
  - **Road search algorithm**
  - Few (simple) calculations
  - **Pipelined algorithm** works naturally with FPGAs
  - **Parallelized** processing (in space & time multiplexing)
  - Operates at a **fixed latency** → **truncate** if needed

Seeding

- Seed by forming a **tracklet** from pairs of **stubs** in adjacent layers (or disks)
  - Initial track parameters from stubs + IP constraint
  - Tracklets must be consistent with $p_T > 2$ GeV, $|z_0| < 15$ cm

- Seed in multiple layer combinations for good coverage & redundancy
  - Barrel: **L1+L2**, **L3+L4**, **L5+L6**
  - Disk: **D1+D2**, **D3+D4**
  - Overlap: **L1+D1**, **L2+D1**
  - *Adaptable*

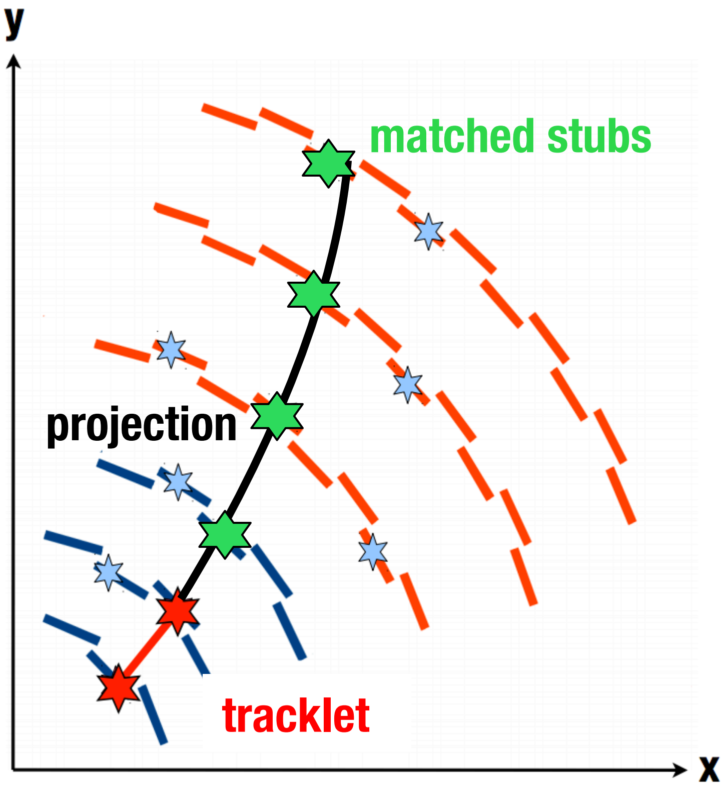




Project & Match Stubs

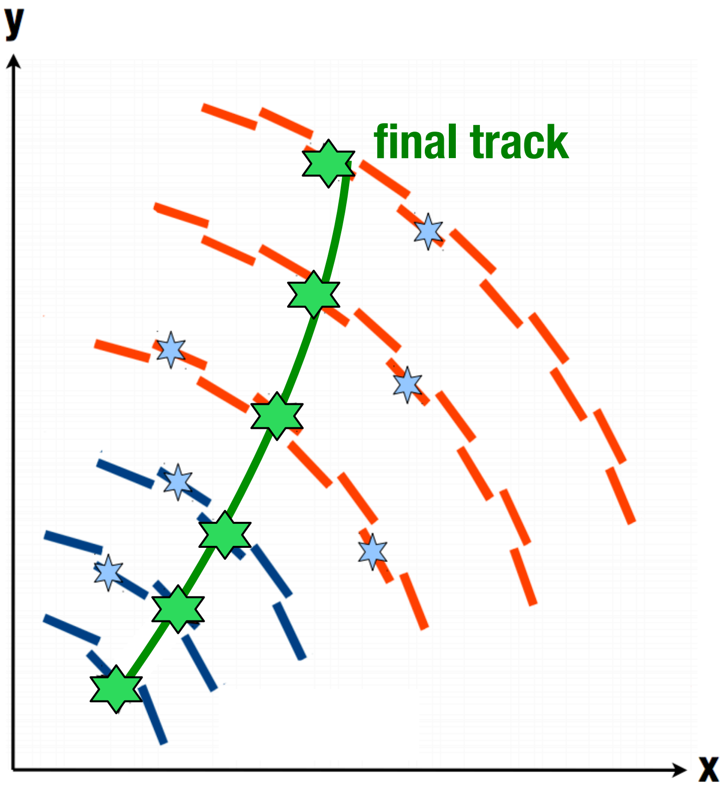
- Use tracklet to **project** to other layers and the disks
- Project both inwards and outwards
- All projections done simultaneously in parallel

- Look for **matched stubs** within a window around the projected track
- Stub with smallest residual kept for fitting stage

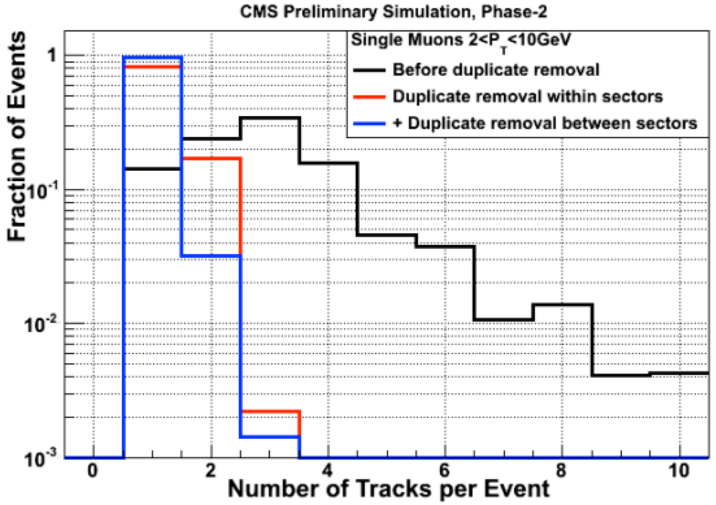


Fit & Duplicate Removal

- Use original and matched stubs
  - Require at least 4 stubs for valid track
- Refit to get the **final track**
- Linearized χ^2 fit
- Gives final track parameters
  - p_T, η, ϕ_0, z_0
  - Optional d_0 (5-parameter fit)

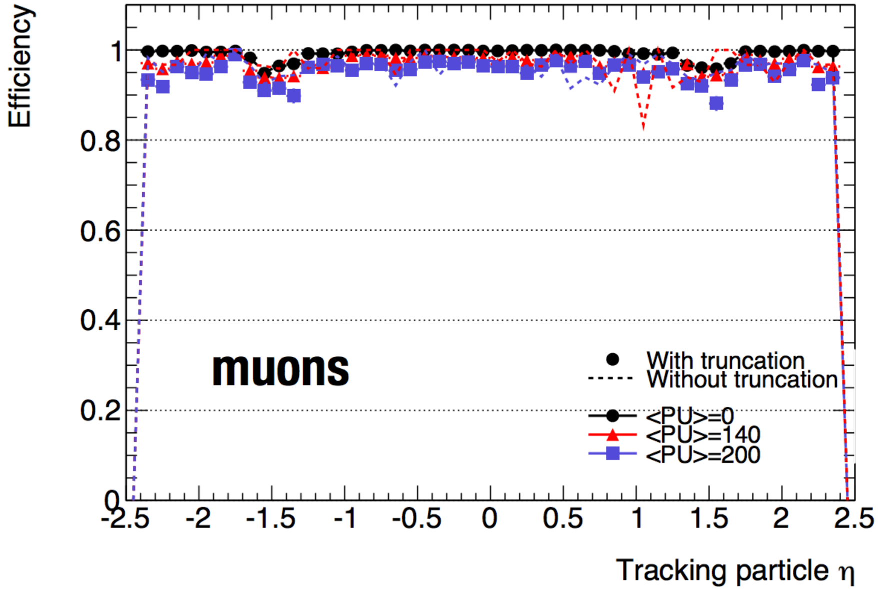


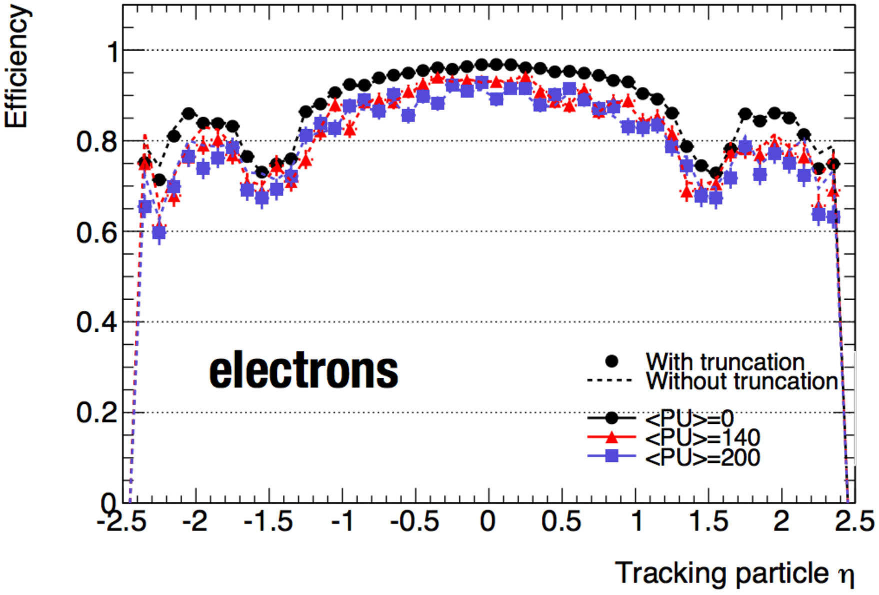
- Given track may be found many times (multiple seeding combinations)
- **Remove duplicate tracks** – tracks that have fewer than 3 unique stubs

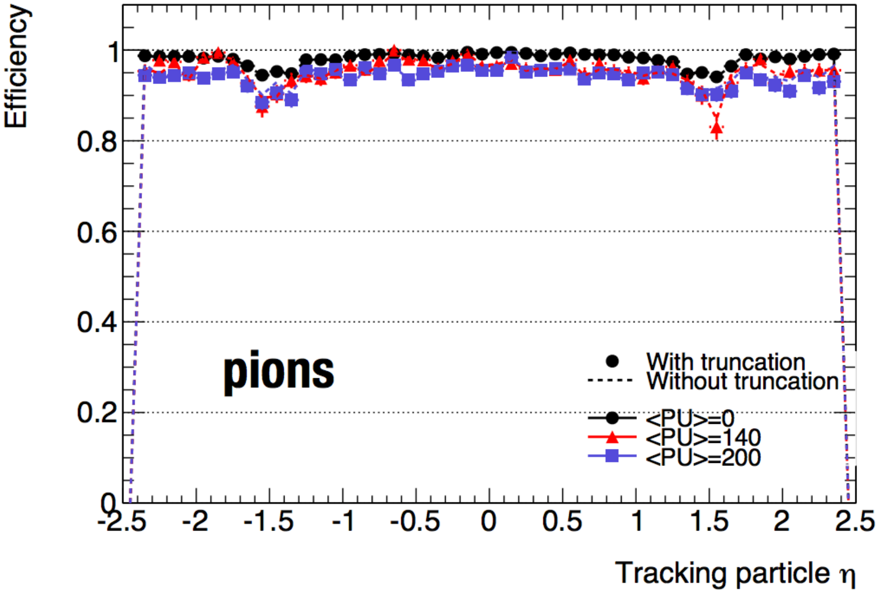


Tracking Performance

- Efficiency as a function of η
- For a single object (e,μ,π), high efficiency achieved
- Effect of truncation is minimal

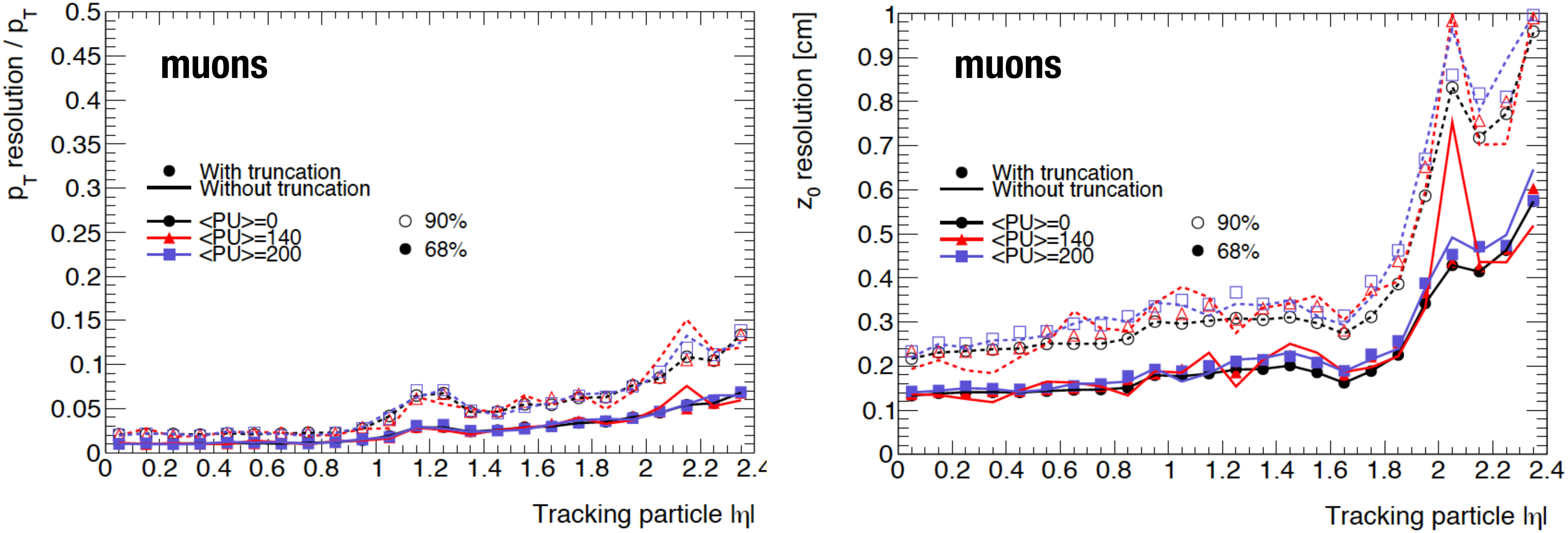






Tracking Performance

- Track parameter resolutions



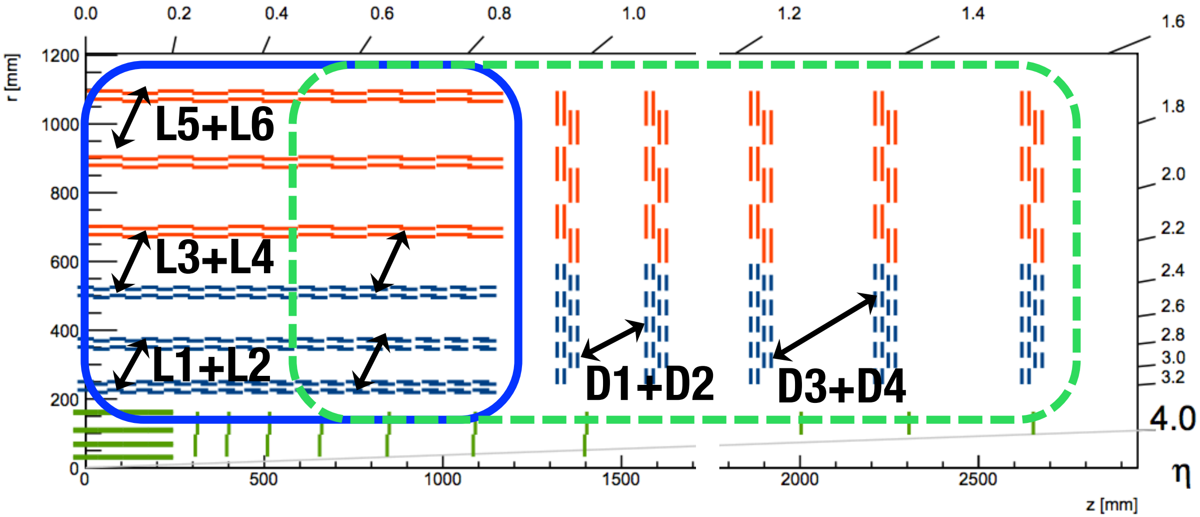
- Already good enough resolution for trigger
- Known degradation from using too few bins in certain points of the calculation → can be corrected

Firmware Overview

- **Pipelined** algorithm which operates with a **fixed latency**
  - Few hand-optimized processing modules
  - Processing modules read from, and write to, memories (BRAMs)
  - Wiring of modules & memories automated via python scripts
  - Each processing step has a fixed time to produce its first output
  - Pipelined design produces output for new event every TMUX*25ns
  - After that, move to next event → **truncate if needed**

Currently implemented (+η) as two FW projects:

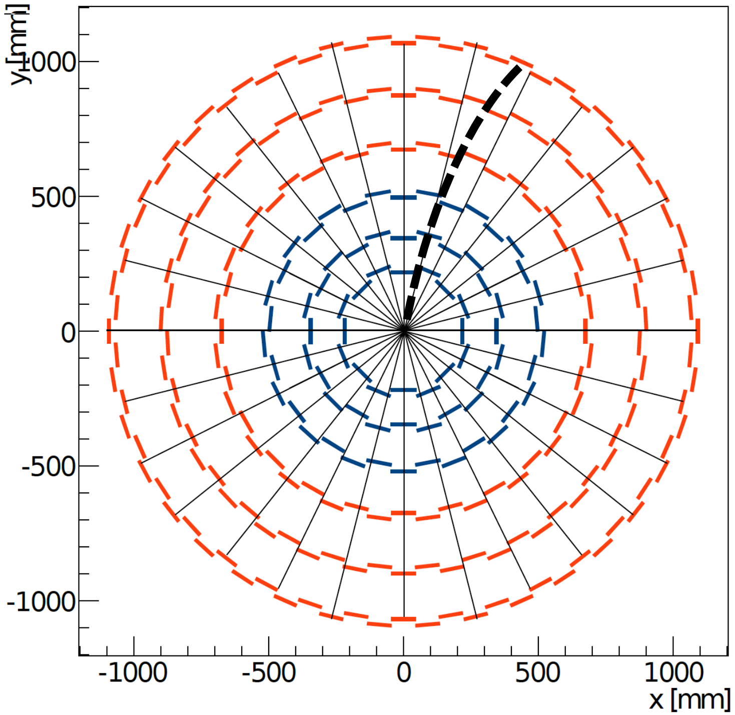
- **Half barrel**
- **Hybrid + disks**



Hardware Configuration

- Deal with combinatorics by **parallel data processing**

- **Time multiplex** by x4-8 *(Adaptable)*
  - TMUX=6 → new event every 150 ns
- Divide detector into ϕ **sectors** *(Adaptable)*
  - 28 sectors → 2 GeV track spans max two sectors
  - Each sector is a processing board



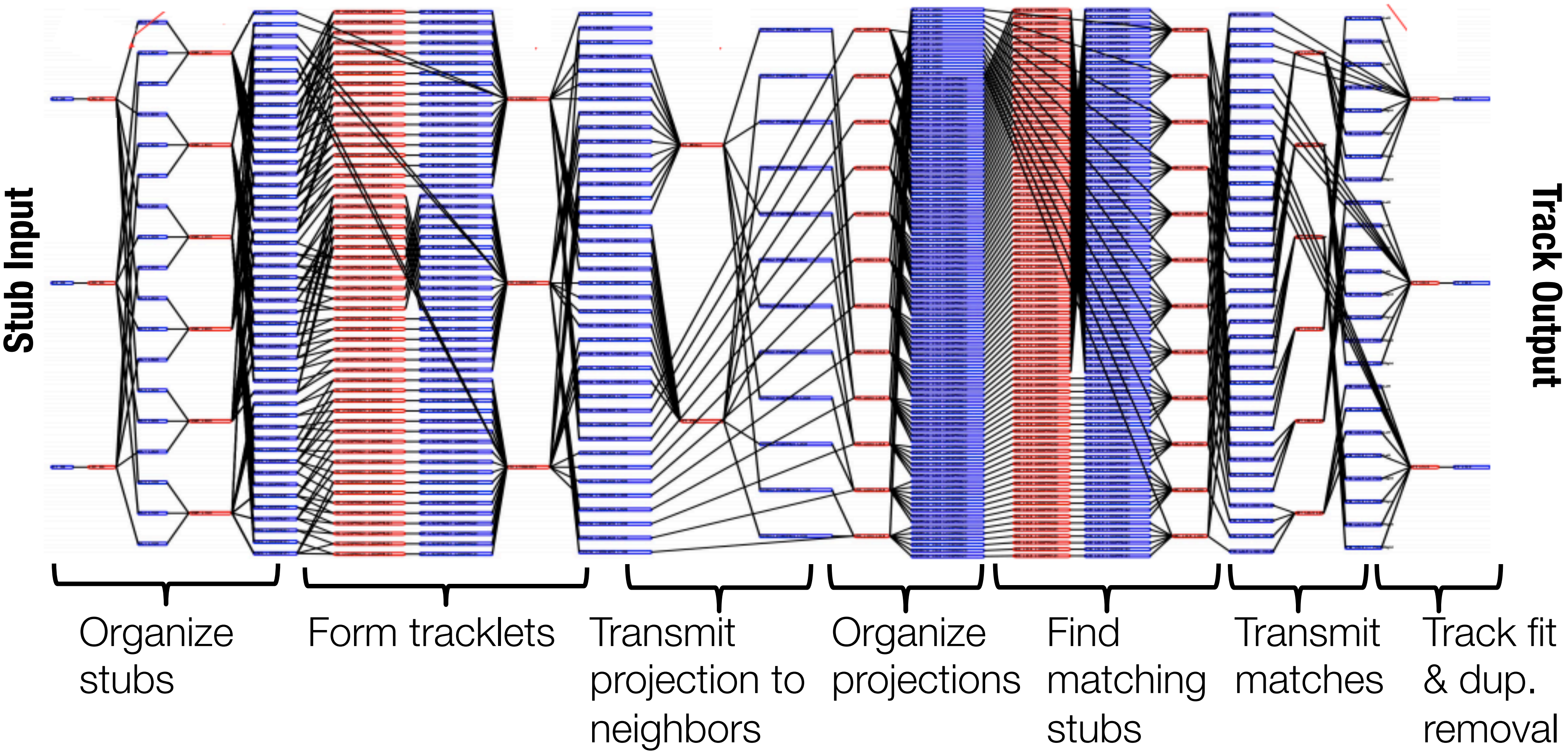
- Tracklets are formed within a sector
  - Can project to its adjacent sectors
  - Tracklet then needs to be sent to neighbor for stub matching → need inter-board communication

Project Overview

Memories

Processing modules

8 processing + 2 transmission steps used to implement algorithm



Stub Input

Track Output

Organize stubs

Form tracklets

Transmit projection to neighbors

Organize projections

Find matching stubs

Transmit matches

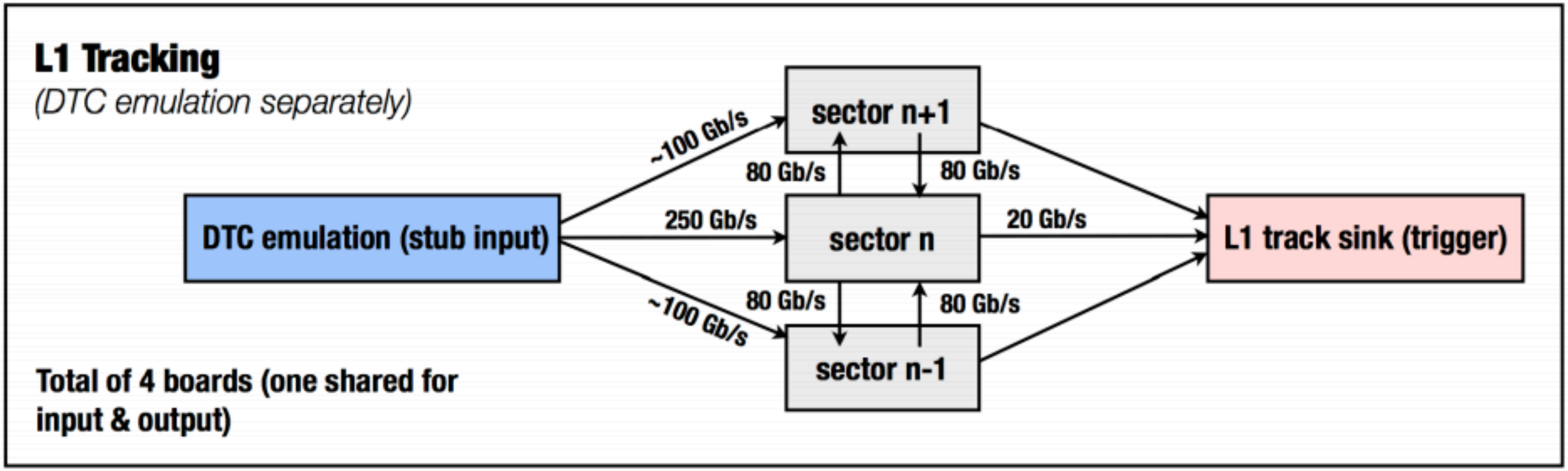
Track fit & dup. removal

Demonstrator

Test stand made of **4 CTP7s** (with **Virtex-7 690T FPGA**) and **AMC13** card for clock/synchronization

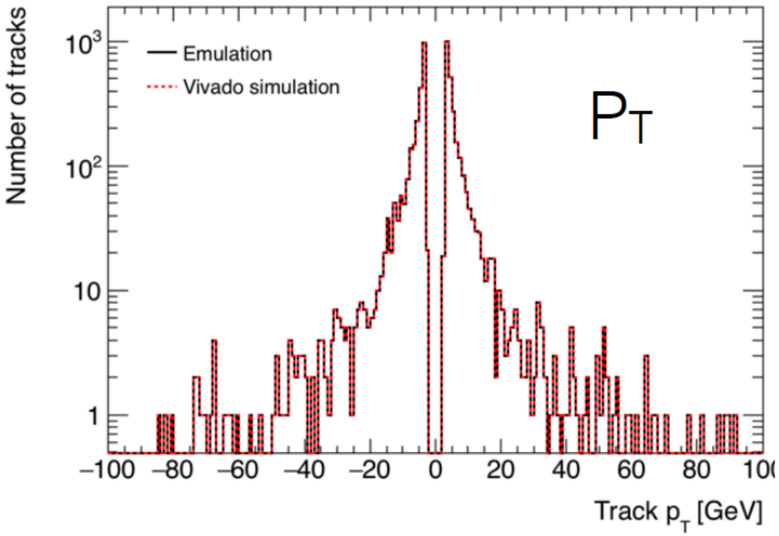
- Used for full scale testing (including inter-board communication)
- Validate performance v. emulation
- Latency measurements

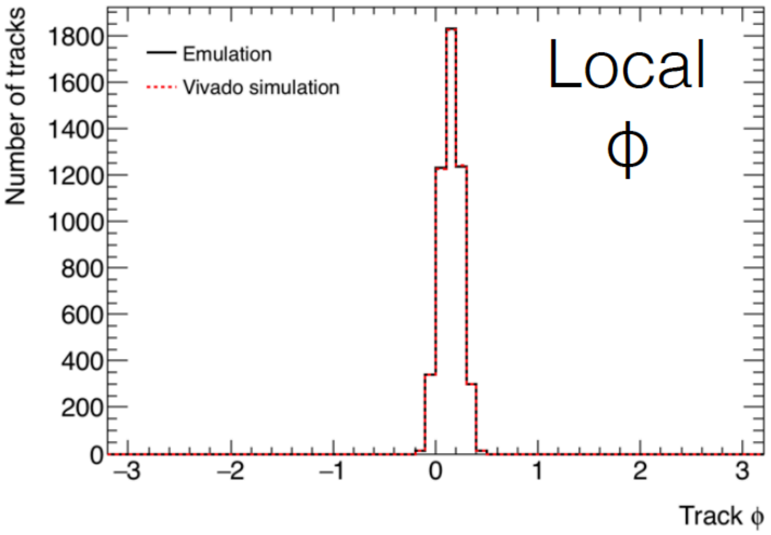
CTP7 Boards from Univ. of Wisconsin
Currently used in CMS L1 trigger



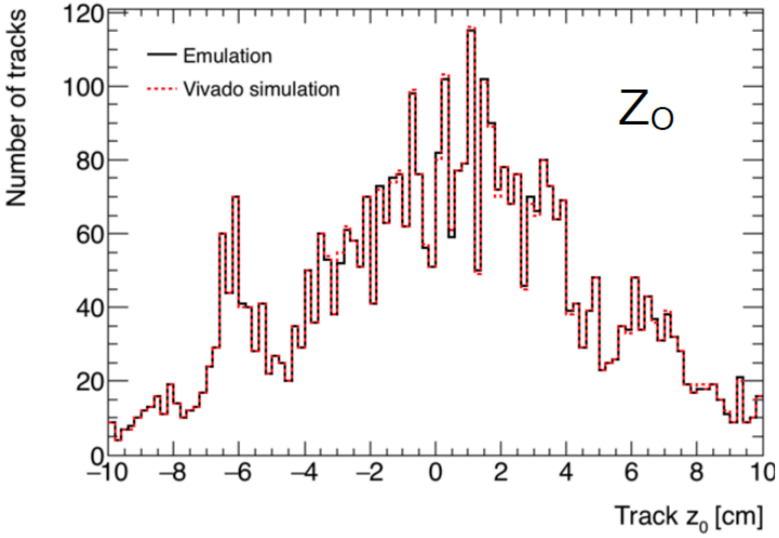
Performance v. Emulation

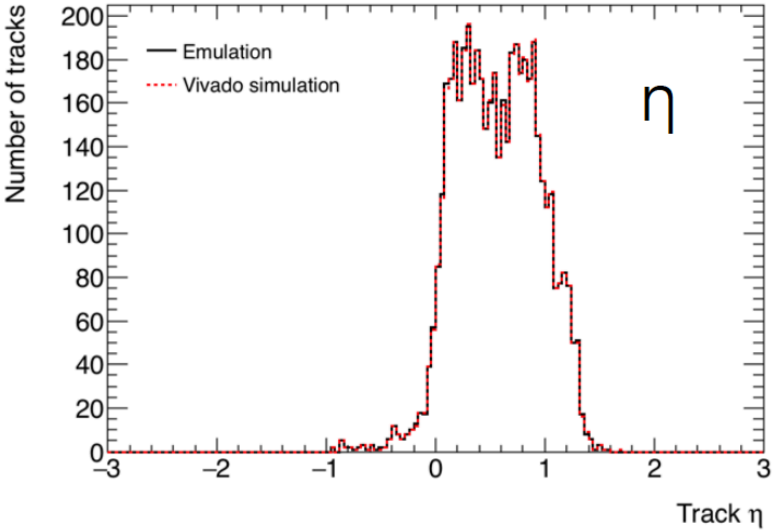
Single muon : 100% agreement
ttbar+200 PU: >99% agreement





Half barrel project - ttbar + 200 PU

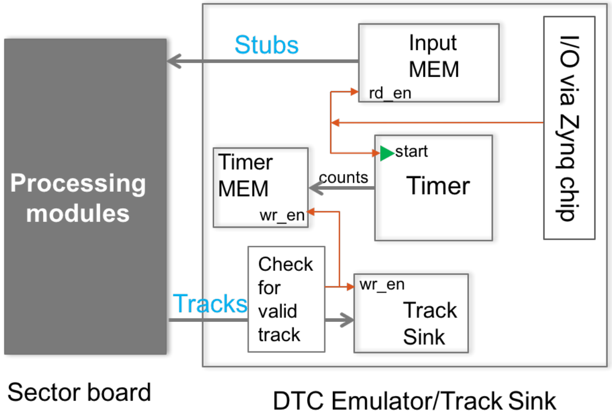




Latency Measurement & Model

- L1 trigger decision at 12.5 µs – goal: tracks before 4 µs

- A full **end-to-end latency measurement** done with clock counter
  - 240 MHz clock (same as processing clock)
  - Implemented on DTC emulator board
  - First track out latency: 800 clks = **3.33 µs**
  - Verified latency for
    - Half barrel and hybrid + disk projects
    - Single muon and ttbar+200PU events



- Compare this to **latency model**
  - Each processing step has a fixed latency → predict latency
  - Model latency: **3.35 µs** which agrees well with measured one (3 clks or 0.38% difference)

Improvements to the Latency

- L1 trigger decision at 12.5 μs – goal: tracks before 4 μs
- Current latency: tracks at 3.33 μs → **Can we get faster?** Yes… there are some very obvious candidates to reduce latency

- Algorithmic improvements:
  - Remove redundant “layer router” (~150 ns)
  - Considerable latency from **inter-board communication (~1μs)**
    - Optimize the transmission protocol
    - **Duplicate data** from neighboring sector → **remove sector-to-sector communication** in projection and match finding steps

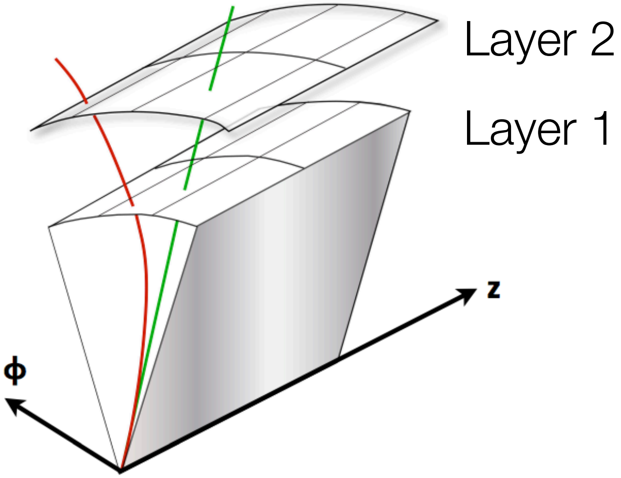
- General improvements:
  - Run with higher clock speed
  - Different clock domains for different processing modules

Progress Towards a Full System

Better Load Balancing

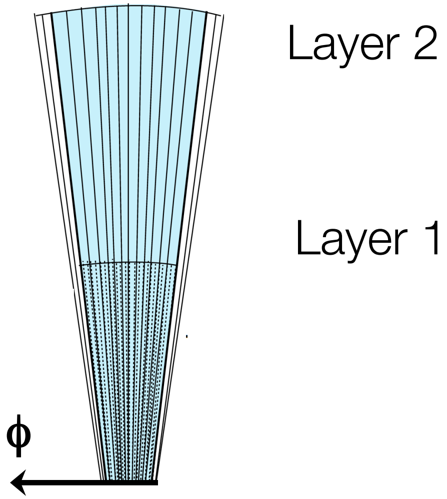
- Main challenge for tracklet approach: combinatorics when forming tracklets & matching tracklet projections to stubs
- Subdivide ϕ sector into Virtual Modules (VMs) for parallel processing
- Future: better load balancing by using thinner (in ϕ) VMs

Current configuration:



Layer 1: 8 z * 3 ϕ = 24 VM
Layer 2: 8 z * 4 ϕ = 32 VM
768 pairs
Only 96 of these can form tracklets

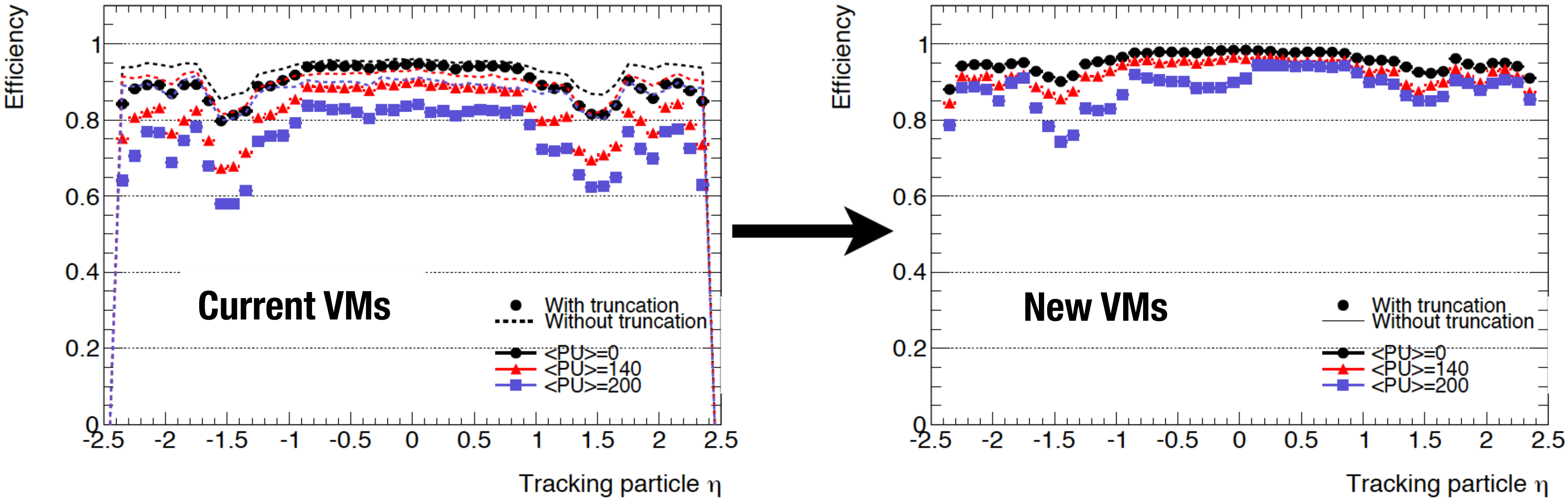
Future configuration:



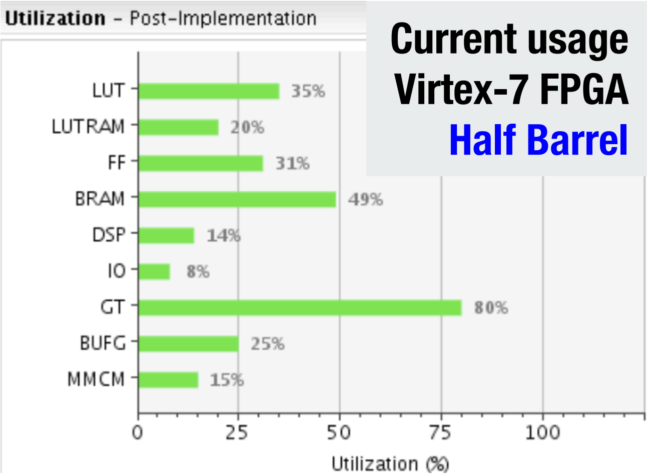
Layer 1: 24 VM in ϕ (full length in z)
Layer 2: 16 VM in ϕ (full length in z)
Total of 120 pairs can form tracklets

Tracking in Jets (ttbar)

- Compared to current VMs, improved load balancing (the thinner ϕ partitions)
  - No additional resources
  - Significant performance improvement for tracks in jets
  - Minimizes the impact of truncation



FPGA Resource Usage Projections



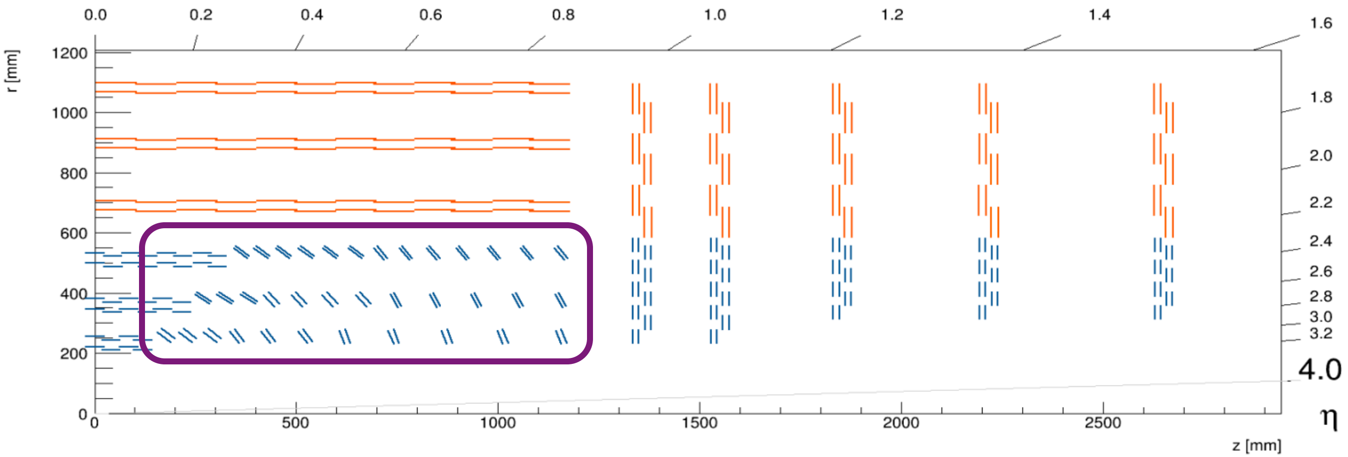
Current usage Virtex-7 FPGA Half Barrel

- Goal: One processing board for a "full sector" (one ϕ sector, full η range)
- Given current resource usage → estimate what is needed for processing a full sector
- Assume 25Gbps links needed from DTC → Compare with resources of Ultrascale+ FPGAs

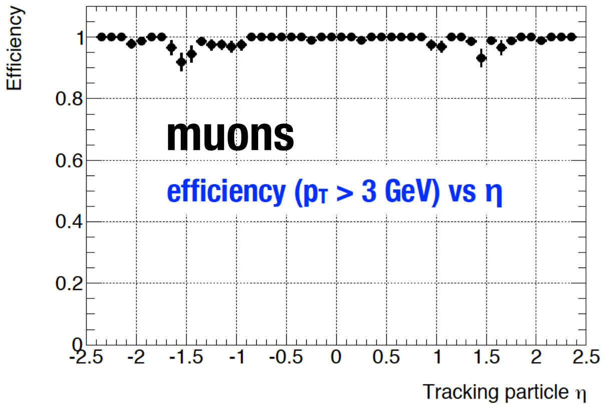
	LUT Logic	LUT Memory	BRAM	DSP
Full sector	279733	151191	2721.5	1818
Virtex-7	65%	87%	185%	51%
VU3P	32%	81%	85%	80%
→ **VU5P**	21%	53%	58%	52%
VU7P	16%	40%	42%	40%
VU9P	11%	27%	28%	27%
VU11P	10%	27%	29%	20%
VU13P	7%	20%	22%	15%

Moving to Tilted Barrel Geometry

- Tracker as it actually will be built



- Ported floating-point and bitwise emulations to tilted barrel geometry
  - Minor changes in geometric constraints
  - Firmware will be easy to adapt
  - Efficiencies remain high
  - z_0 resolution slightly worsens in transition region



Conclusions

- L1 tracking crucial to HL-LHC physics goals
- **Tracklet approach** to L1 tracking
  - Road search algorithm using **commercial FPGAs**
  - Fully implemented as floating-point & integer-based algorithm

- **Demonstrated feasibility** of the Tracklet approach
  - Half barrel & hybrid + disks projects running on Virtex-7 FPGAs
  - Excellent agreement (>99%) between firmware & emulation
  - Time from stubs in to tracks out: **3.33 μs latency**
  - Design seems **scalable** to UltraScale+ FPGAs

- Ongoing improvements:
  - Reduce latency
  - Better load balancing in the VMs by changing partitioning
  - Migrate to tilted barrel geometry

BACKUP

Latency Model

TMUX 6, 240 MHz CLK

Step	Processing time (ns)	Latency (clk)	Latency (ns)	Transmission Latency (ns)	Total (ns)
Input link	0.0	1	4.2	316.7	320.8
Layer Router	150.0	1	4.2	0.0	154.2
VM Router	150.0	4	16.7	0.0	166.7
Tracklet Engine	150.0	5	20.8	0.0	170.8
Tracklet Calculator	150.0	43	179.2	0.0	329.2
Projection Transceiver	150.0	13	54.2	316.7	520.8
Projection Routing	150.0	5	20.8	0.0	170.8
Match Engine	150.0	6	25.0	0.0	175.0
Match Calculator	150.0	16	66.7	0.0	216.7
Match Transceiver	150.0	12	50.0	316.7	516.7
Track Fit	150.0	26	108.3	0.0	258.3
Duplicate Removal	0.0	6	25.0	0.0	25.0
Track Link	0.0	1	4.2	316.7	320.8
Total	**1500.0**	**139**	**579.2**	**1266.7**	**3345.8**

Estimated Total Latency

Latency Model

TMUX 6, 240 MHz CLK

Step	Processing time (ns)	Latency (clk)	Latency (ns)	Transmission Latency (ns)	Total (ns)
Input link	0.0	1	4.2	316.7	320.8
Layer Router	150.0	1	4.2	0.0	154.2
VM Router	150.0	4	16.7	0.0	166.7
Tracklet Engine	150.0	5	20.8	0.0	170.8
Tracklet Calculator	150.0	43	179.2	0.0	329.2
Projection Transceiver	150.0	13	54.2	316.7	520.8
Projection Routing	150.0	5	20.8	0.0	170.8
Match Engine	150.0	6	25.0	[illegible]	[illegible]
Match Calculator	150.0	16	66.7	0.0	216.7
Match Transceiver	150.0	12	50.0	316.7	516.7
Track Fit	150.0	26	108.3	0.0	258.3
Duplicate Removal	0.0	6	25.0	0.0	25.0
Track Link	0.0	1	4.2	316.7	320.8
Total	**1500.0**	**139**	**579.2**	**1266.7**	**3345.8**

Overhead in each processing module

Estimated Total Latency

Latency Model

TMUX 6, 240 MHz CLK

Step	Processing time (ns)	Latency (clk)	Latency (ns)	Transmission Latency (ns)	Total (ns)
Input link	0.0	1	4.2	316.7	320.8
Layer Router	150.0	1	4.2	0.0	154.2
VM Router	150.0	4	16.7	0.0	166.7
Tracklet Engine	150.0	5	20.8	0.0	170.8
Tracklet Calculator	150.0	43	179.2	0.0	329.2
Projection Transceiver	150.0	[illegible]	[illegible]	[illegible]	520.8
Projection Routing	150.0	[illegible]	[illegible]	[illegible]	170.8
Match Engine	150.0	6	25.0	0.0	175.0
Match Calculator	150.0	16	66.7	0.0	216.7
Match Transceiver	150.0	12	50.0	316.7	516.7
Track Fit	150.0	26	108.3	0.0	258.3
Duplicate Removal	0.0	6	25.0	0.0	25.0
Track Link	0.0	1	4.2	316.7	320.8
Total	**1500.0**	**139**	**579.2**	**1266.7**	**3345.8**

Processing time of each module before moving to the next event (TMUX = 6)

Estimated Total Latency

Latency Model

TMUX 6, 240 MHz CLK

Step	Processing time (ns)	Latency (clk)	Latency (ns)	Transmission Latency (ns)	Total (ns)
Input link	0.0	1	4.2	316.7	320.8
Layer Router	150.0	1	4.2	0.0	154.2
VM Router	150.0	4	16.7	0.0	166.7
Tracklet Engine	150.0	5	20.8	0.0	170.8
Tr				0.0	329.2
Pr				316.7	520.8
Pr				0.0	170.8
Match Engine	150.0	6	25.0	0.0	175.0
Match Calculator	150.0	16	66.7	0.0	216.7
Match Transceiver	150.0	12	50.0	316.7	516.7
Track Fit	150.0	26	108.3	0.0	258.3
Duplicate Removal	0.0	6	25.0	0.0	25.0
Track Link	0.0	1	4.2	316.7	320.8
Total	**1500.0**	**139**	**579.2**	**1266.7**	**3345.8**

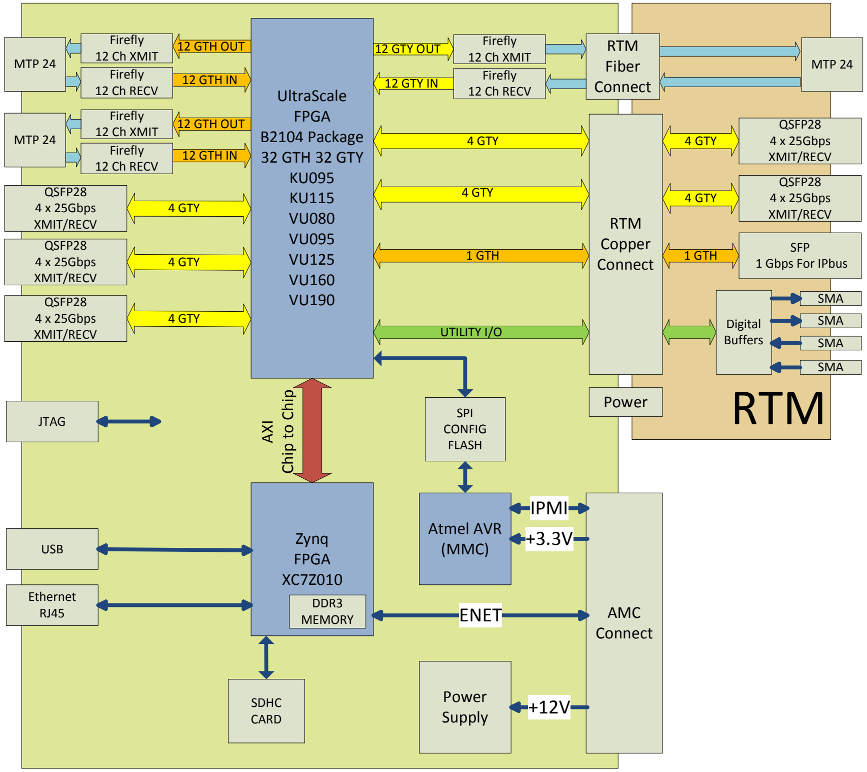
Inter-board communication latency:

- Transmission protocol for stub inputs, projections, matches and track outputs
- 76 clk (240MHz) measured with ChipScope

Estimated Total Latency

Prototype Board for High Speed Links

- Explore different 25Gbps technologies
  - Links
  - Connectors
  - Layout
  - Fiber RTM
  - Copper RTM
- Ultrascale FPGA
  - KU115 for processing
  - VU080 for I/O capabilities



SERIAL LINK DEVELOPMENT BOARD

- Based on existing g-2 project
- Board is in design now