Fast segment finding in the ATLAS MDTs using a Legendre transform algorithm

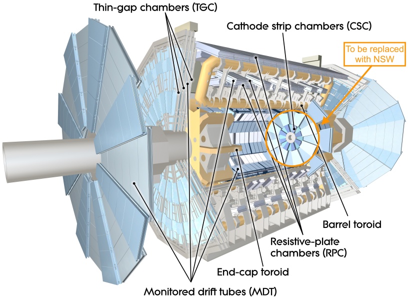
Kostas Ntekas
on behalf of the ATLAS TDAQ Collaboration

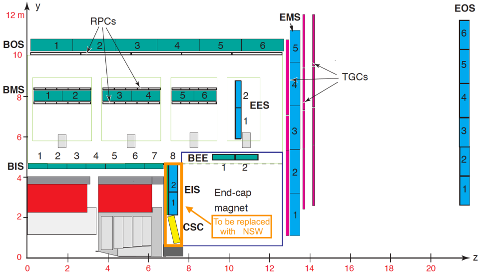
University of California Irvine

Connecting the Dots 2018, Seattle (US)
March 17, 2018

1. **The ATLAS First-level Muon Trigger**
2. **Using the MDTs in ATLAS First-level Muon Trigger for HL-LHC**
3. **The Legendre Transform algorithm for MDT segment reconstruction**
4. **Hardware implementation of the Legendre Transform algorithm**

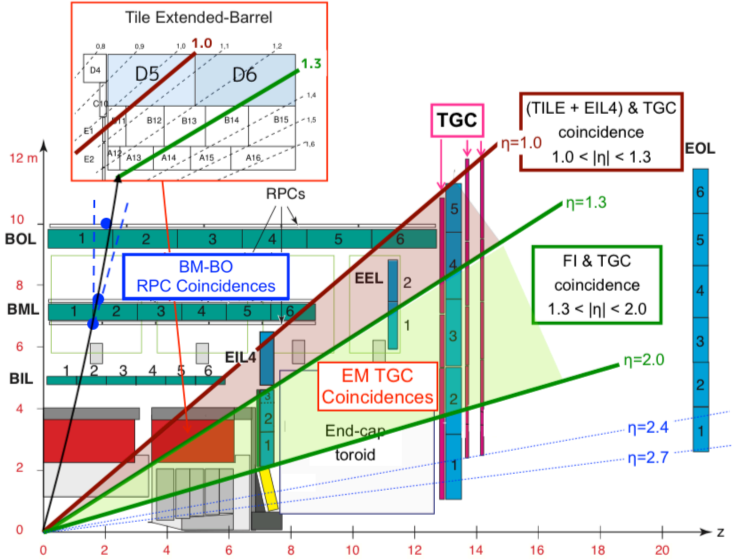
The ATLAS Muon Spectrometer





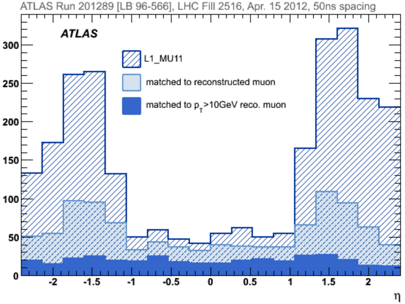
- RPC/TGC (fast response with BCID capability) are used in the muon trigger decision
- CSC/MDT (excellent spatial resolution) are the precise tracking detectors
- The inner wheels will be replaced with new detector technologies (Micromegas & sTGC) in Phase-I that will contribute both in tracking and triggering

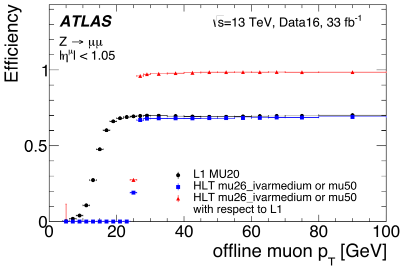
ATLAS First-level Muon Trigger Evolution (Pre-Phase-I)



Limitations in the current ATLAS First-level Muon Trigger

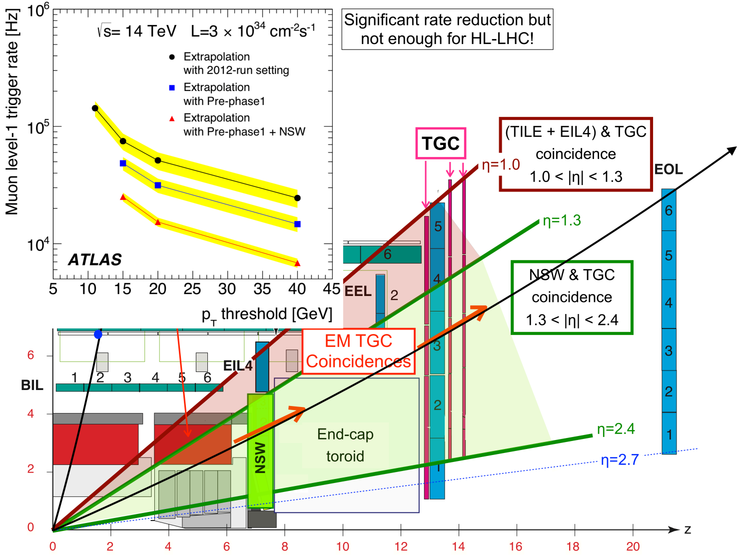
- Endcap muon trigger rate dominated by fakes (charge particles not originating from the IP)
- Low ($\sim$70%) barrel trigger efficiency due to geometrical acceptance of the RPC chambers
- The current trigger chambers RPC & TGC have only moderate p_T resolution (limited spatial resolution)$\rightarrow$ shallow efficiency turn-on curve



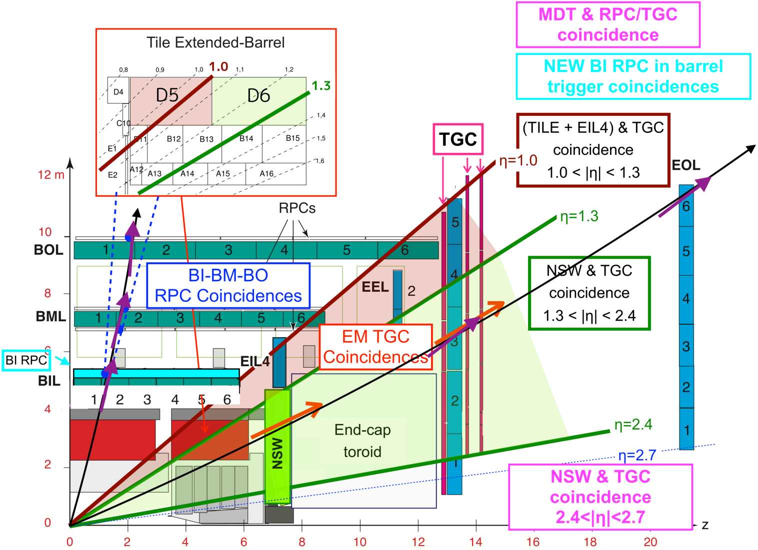


- HL-LHC physics program necessitates maintaining single muon trigger $p_T > 20\,\text{GeV}$ and efficient di-lepton triggers with lower p_T threshold $\rightarrow$ upgrade first-level muon trigger

ATLAS First-level Muon Trigger Evolution (Phase-I)

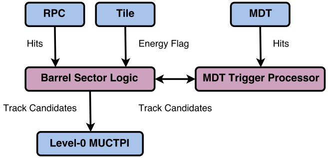


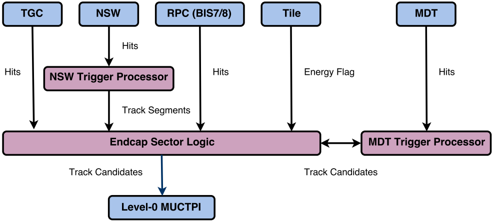
ATLAS First-level Muon Trigger Evolution (Phase-II)



ATLAS Phase-II First-level Muon Trigger Architecture

- First level trigger decision is issued at 1 MHz rate (overall muon trigger latency $< 5.4\,\mu s$)
- MDTs included in the muon trigger → improve p_T measurement and suppression of fake coincidences
- RPC/TGC[2] provide pre-trigger candidates
  - Reference time and position vector for matching the MDT hits in space and time[1]
  - 2^{nd} coordinate measurement (along non-bending plane)

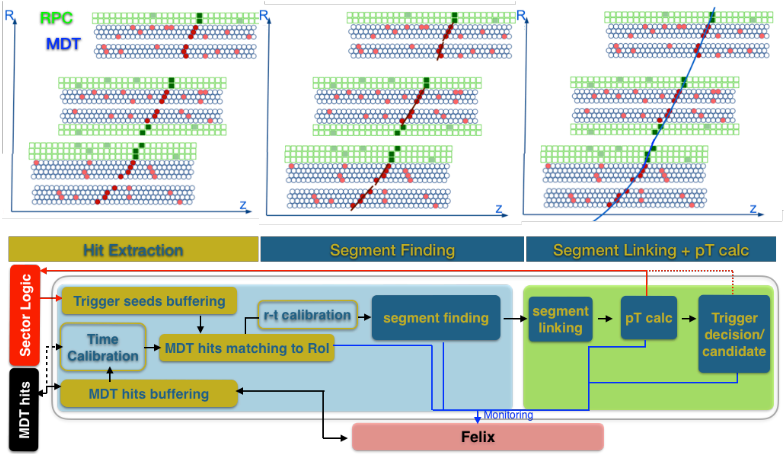




[1] MDTs are much slower (max drift time $\simeq 700\,\mathrm{ns}$) than RPC/TGC so the MDT hits of one muon track span over several BCs

[2] RPC/TGC on-chamber electronics will be replaced to cope with the Phase-II trigger rate and latency (data buffering and trigger logic moves off-detector)

MDT Trigger Processor (TP) - Functional Diagram

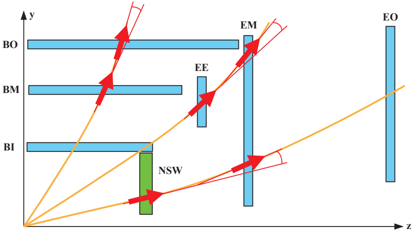


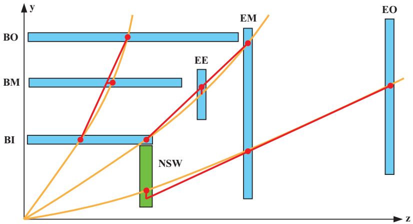
In addition to the trigger decision the MDT TP is also used for

- Readout of the MDT hits (hits are buffered and sent on L0 accept)
- MDT electronics configuration and monitoring (slow control)

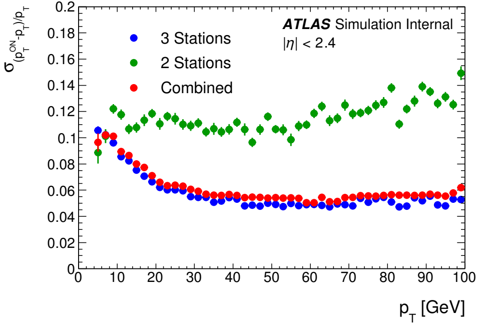
MDT Trigger - p_T Determination

- Exploiting the precise tracking with the MDTs the muon p_T can be measured with an accuracy close to the one of the offline MS reconstruction
- 3-station: Measuring the sagitta using three segment positions (applicable to 73% of the muons)
- 2-station: Measuring the segment deflection angle (applicable to 94% of the muons)
- **Need accuracy in both angle and position of the segment for a precise p_T determination**





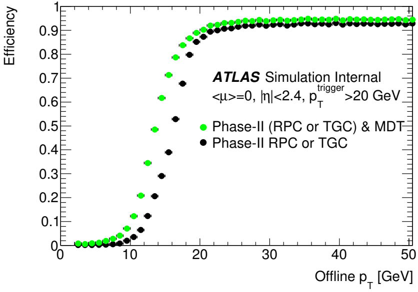
MDT Trigger - p_T Determination : Performance

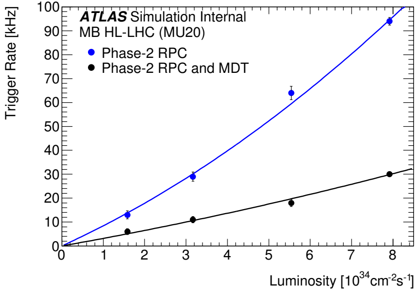


- Combine both methods to maximise the acceptance of the MDT trigger
- Sagitta (3-station) superior to the deflection angle (2-station)
  $\rightarrow$ Use 3-station when applicable and have 2-station as a backup
- Combining the two methods we can achieve p_T resolution $\sim 5\%$
- Require segment reconstruction with $\sigma_r \leq 1\,\mathrm{mm}$, $\sigma_\theta \leq 1\,\mathrm{mrad}$ to reach this accuracy

MDT Trigger Efficiency & Rates

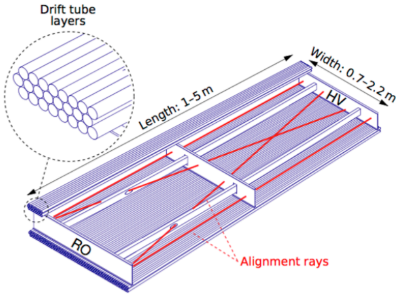
- Efficiency plateau ($\sim$93%) defined by the RPC/TGC trigger
- The precise p_T determination using the MDT enhances the suppression of the muon candidates with p_T below the nominal threshold (sharper turn-on)
- Fake trigger rate (coming mostly from low quality RPC coincidences in the barrel) is suppressed with the inclusion of the MDT in the first level muon trigger
  - More than x3 rate reduction predicted for HL-LHC conditions

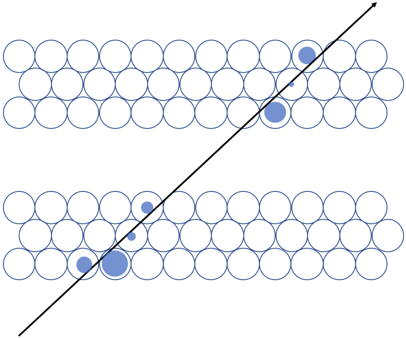




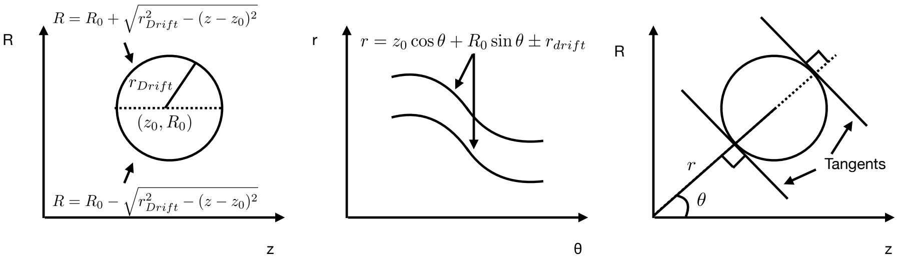
MDT Segment Reconstruction

- Each ATLAS MDT chamber consists of two multilayers (ML) of tubes (with 3 or 4 staggered layers each) separated by a spacer frame
- The MDT segment is defined as the common tangent line to the drift circles in the two MLs
- Offline reconstruction evaluates common tangent lines for each pair of circles to determine the best tangent line
- Segment reconstruction accuracy $\sigma_r \leq 100\,\mu\text{m}$ and $\sigma_\theta \leq 1\,\text{mrad}$



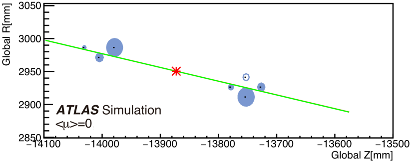


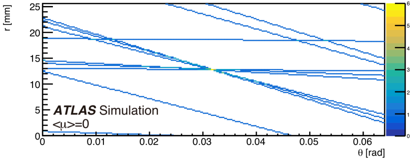
Legendre Transform Algorithm for Drift Circles



- Algorithm introduced in nima.2008.04.038, T. Alexopoulos et al.
- For each drift circle the Legendre transformations of its concave and convex parts are evaluated for various θ values (r, θ polar coordinates)
- Each drift circle in the cartesian space corresponds to 2 lines in the Legendre space
- The bin with the maximum number of entries (where the different lines cross) corresponds to the most popular tangent line

Legendre Transform Algorithm for Drift Circles

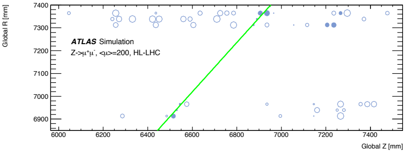


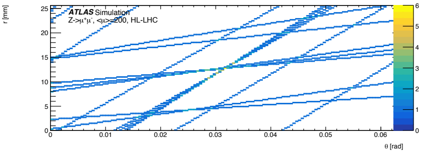


- Algorithm introduced in nima.2008.04.038, T. Alexopoulos et al.
- For each drift circle the Legendre transformations of its concave and convex parts are evaluated for various θ values (r, θ polar coordinates)
- Each drift circle in the cartesian space corresponds to 2 lines in the Legendre space
- The bin with the maximum number of entries (where the different lines cross) corresponds to the most popular tangent line

Legendre Transform Algorithm for the MDT trigger

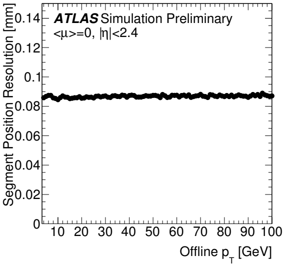
- Pattern recognition nature of the algorithm makes it suitable to be implemented in an FPGA for fast segment finding (trigger level)
- Can parallelise drift circle processing such that the overall latency does not scale with the number of hits (assuming $\lesssim 100$ matched hits per RoI)
- Can be adjusted to reconstruct multiple segments using one Legendre map
- Cannot process all the hits of a single chamber on the trigger level if we want to fit within our latency budget $\rightarrow$ Relying on RoI provided by the (faster) muon trigger chambers
- High occupancy expected at HL-LHC conditions $\rightarrow$ need to narrow down the RoI as much as possible to save processing time and resources

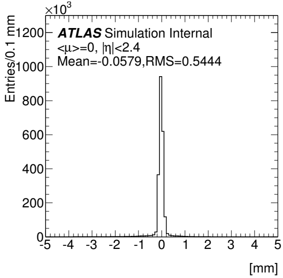


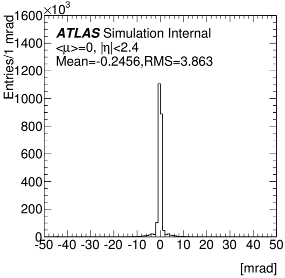


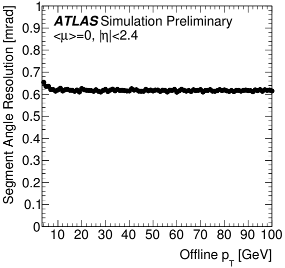
Legendre Transform - Performance

- Comparing the Legendre transform segment with the one reconstructed offline
- Decent resolution achieved in the segment reconstruction accuracy with the Legendre transform $\sigma_r = 90\,\mu\text{m}$ and $\sigma_\theta = 0.6\,\text{mrad}$ flat in p_{T} $[0, 100\,\text{GeV}]$
- Currently focussing on events in the tails trying to improve on mis-reconstructed segments (fine tuning of the algorithm)



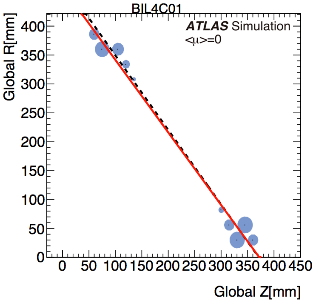


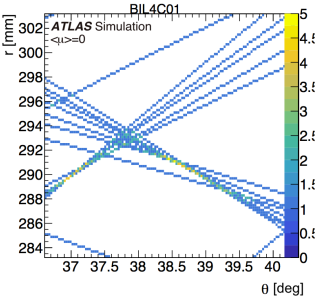




Legendre Transform - Points of optimisation I/II

- Legendre space resolution (bin size)
  - Tuning is essential since it affects the FPGA choice (resource usage)
  - Too fine bins result in making the Legendre lines overlap more difficult
  - The bin size limits the accuracy that can be achieved in the segment angle/position determination
- Accuracy that can be assumed in the drift radius (time calibration and r-t relation) of the circles impacts the bin size choice
  - Lower accuracy → larger bin size
- Track bending within a single chamber (low p_T particle in B-field) results in incompatible segments in the two MLs
  1. Run segment finding per ML separately and combine the two ML segments into one
  2. Identify two peaks from one Legendre histogram and combine the two (r, θ) pairs into one



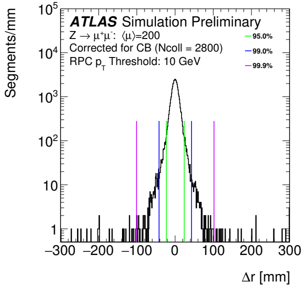


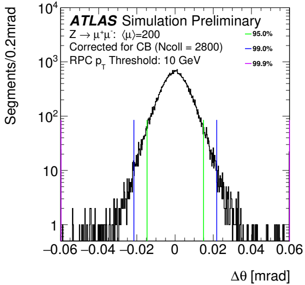
Legendre Transform - Points of optimisation II/II

- Legendre space limits (r,θ) are dictated by the RoI segment resolution*
- Need to ensure that the space around the RoI r, θ is large enough to enclose the MDT segment (max bin)
- Lower resolution RoI segments require larger span of the r, θ axes

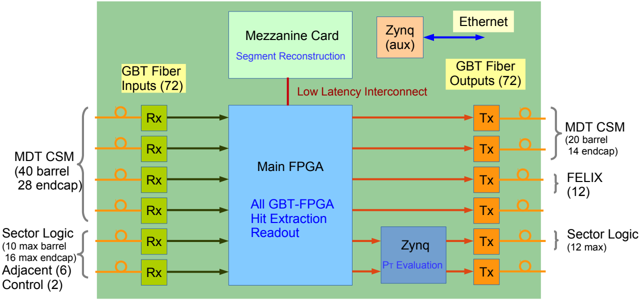
	$\Delta\theta$			Δr		
	95%	99%	99.9%	95%	99%	99.9%
BIL	0.015	0.021	0.0353	24.0	38.4	70.8
BIM	0.015	0.022	0.0343	30.0	48.0	75.6
BIR	0.025	0.047	>0.06	20.4	34.8	62.4
BIS	0.019	0.034	>0.06	28.8	54.0	127.2
BMF	0.035	0.05	>0.06	16.8	27.6	88.8
BML	0.025	0.035	0.0554	14.4	20.4	44.4
BMS	0.036	0.053	>0.06	16.8	26.4	88.8
BOF	0.02	0.043	>0.06	32.4	92.4	>300
BOG	0.025	0.059	>0.06	51.6	163.2	>300
BOL	0.012	0.022	>0.06	21.6	44.4	162.0
BOS	0.016	0.033	>0.06	31.2	79.2	258.0
EML	0.0065	>0.06	>0.06	24.0	70.8	274.8
EMS	0.006	0.056	>0.06	28.8	56.4	258.0
EOL	0.0067	>0.06	>0.06	56.4	>300	>300
EOS	0.0062	>0.06	>0.06	58.8	>300	>300

*The RoI vector is extrapolated to each MDT layer taking into account the local B-field



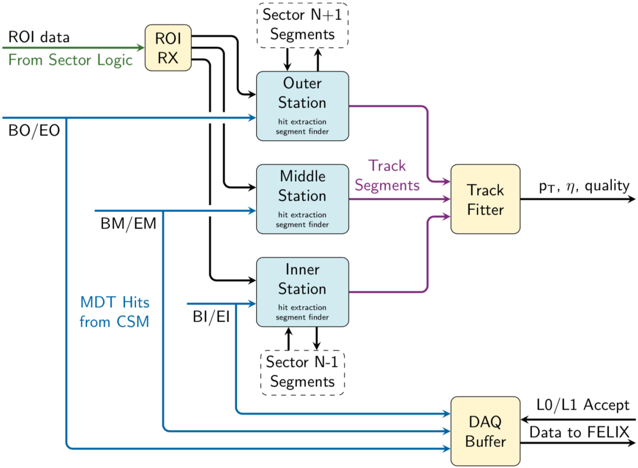


MDT Trigger Processor - Conceptual H/W design



- ATCA blade with GBT (or lpGBT) with $4.8 \mathrm{Gb/s}$ (or $9.6 \mathrm{Gb/s}$) links
- Following ATLAS MDT segmentation one MDT trigger processor handles 1/16 ϕ sectors
- Separate also in Barrel/Endcap regions and A/C side
  - 64 trigger processors in total for the full MDT system
  - Max 18 MDT chambers per trigger processor

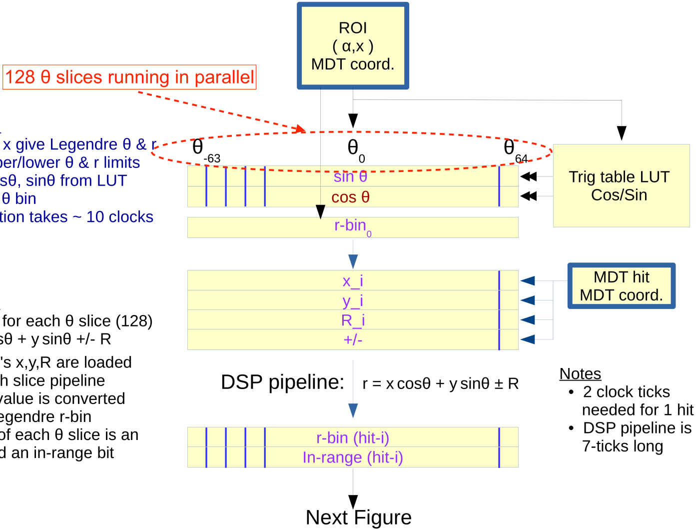
MDT Trigger Processor - Block diagram



- Reconstructed segments are shared between neighbouring sectors to handle sector overlap and barrel-endcap transition region

Legendre Segment Finder - FPGA Implementation I/II

Initialization
- ROI α & x give Legendre θ & r
- Find upper/lower θ & r limits
- Copy cosθ, sinθ from LUT to each θ bin
- Initialization takes ~ 10 clocks



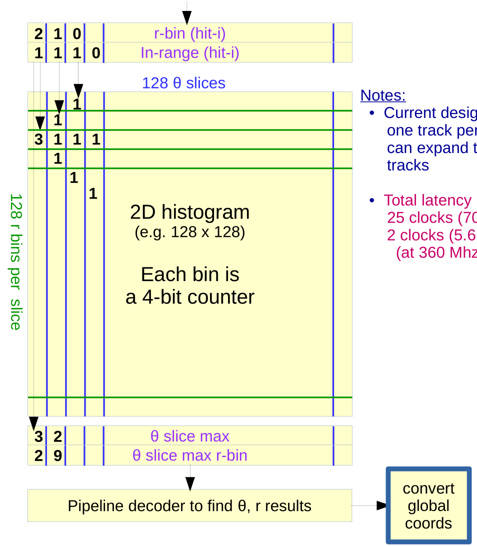
Calculation
- Parallel for each θ slice (128)
- r = x cosθ + y sinθ +/- R
- MDT hit's x,y,R are loaded into each slice pipeline
- Each r value is converted into a Legendre r-bin
- Output of each θ slice is an r-bin and an in-range bit

Notes
- 2 clock ticks needed for 1 hit
- DSP pipeline is 7-ticks long

Legendre Segment Finder - FPGA Implementation II/II

Histogram Building
- Each clock gives a new r-bin for each θ.
- Each θ-slice increments its counter-bin in parallel
- Each θ-slice caches its max-bin/value
- Ideally processes all hits in $7+2*N_{hits}$

Finalization
- Search θ-slice max values to find Legendre max r,θ.
- Decode and convert resulting (r,θ) to global coords
- Finalization takes ~ 10 clocks



Notes:
- Current design handles one track per ROI but can expand to multiple tracks
- Total latency is about: 25 clocks (70ns) plus 2 clocks (5.6ns)/hit (at 360 Mhz)

Resources & Latency

- Overall latency of the MDT trigger processor is $\sim 400\,\mathrm{ns}$ with the segment finding performed within $78\,\mathrm{ns}$ (MPV)

Contents	Latency [μs]
Earliest MDT hit signal arrival	0.609
Sector logic track candidate arrival	1.785
Latest MDT hit signal arrival	2.358
Decoding and domain crossing	2.458
Buffering	3.424
Hit extraction	3.432
Data transfer to the mezzanine cards	3.457
Segment reconstruction	3.510
Conversion to the detector coordinates	3.535
Data transfer to the main board	3.560
Transverse momentum evaluation and selection	3.635
Optical link to the Sector Logic (10 m)	3.810

- Preliminary FPGA resource estimate for the mezzanine board for 3 Legendre segment finder engines, handling inputs from 3 MDTs
- The mezzanine board will contain 3 such FPGAs that can operate in parallel to handle multiple simultaneous RoIs

Mezzanine FPGA (1 of 3)

	Number	**LUTs**	**Regs**	**BRAMs**
Legendre Engine	3	96,500	104,500	275
Total		289,000	313,500	825
XKCU115		663,360	1,326,700	4,320
%Use		44%	24%	19%

Summary

- The MDTs will be introduced in the first-level muon trigger of the ATLAS experiment for HL-LHC
- Exploiting the precise tracking of the MDTs the muon p_{T} measurement at the trigger level is refined $\rightarrow$ Sharper efficiency turn-on and significant rate reduction
- The Legendre transform algorithm is able to provide very accurate segment reconstruction using the MDT drift circles
  - Depending on the hardware resources and the implementation the accuracy in the MDT Trigger Processor can be comparable to the offline segment reconstruction
- The main advantage of the algorithm is that it can be easily parallelised $\rightarrow$ "fixed-latency" FPGA implementation independent of the number of MDT hits
- Processing time estimations show that it fits the available latency budget while, despite the significant resource allocation, it can be accommodated in modern large FPGAs

Thanks for your attention!