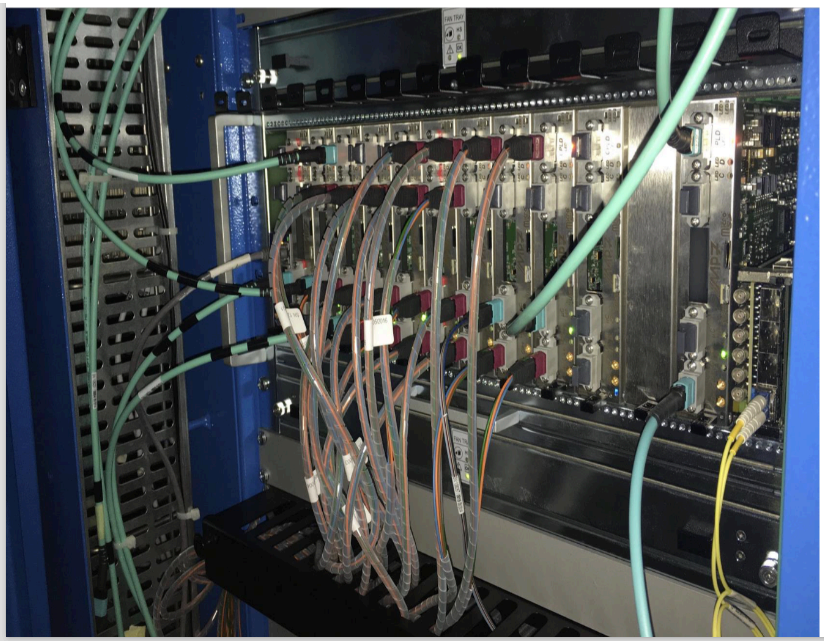
Time-Multiplexed Track-Trigger Overview

Ian Tomalin (RAL)



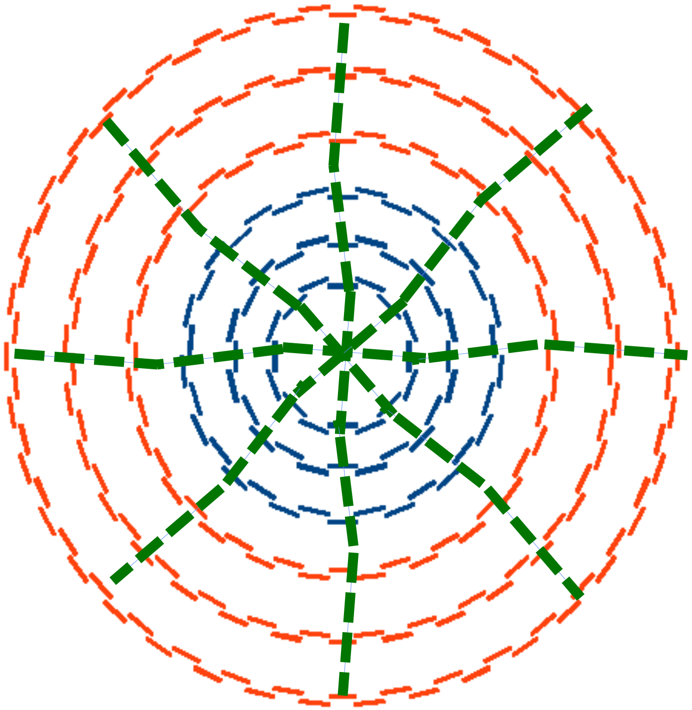
Introduction

- I will present an overview of our track-finding proposal.
  - Technical details to follow in later talks.
- Our design exploits time-multiplexing & FPGA technology.
- We have successfully demonstrated in hardware, the most ambitious of the design variants we proposed at previous meetings.
  - This design meets CMS requirements with today's technology.
- We have seen rapid emergence of new ideas during this project & are confident even better solutions will be found in coming years.
  - Must profit from expertise of all groups to arrive at an ultimate solution for CMS & to build final system.

Accommodating constraints from CMS tracker cabling scheme

- The tracker will be ~divided into φ octants, known as "detector octants", each read out by a separate group of (~32) DTC boards.

- The DTCs calculate the global coordinates of each "stub".

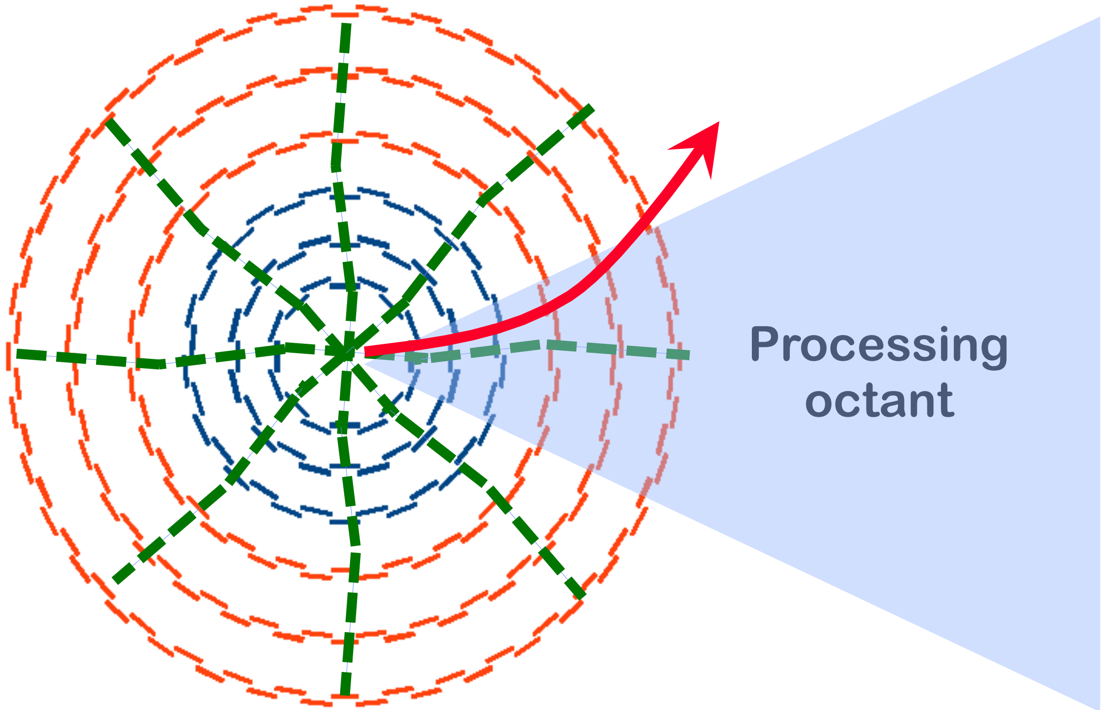


Detector octants shown by green lines.

Accommodating constraints from CMS tracker cabling scheme

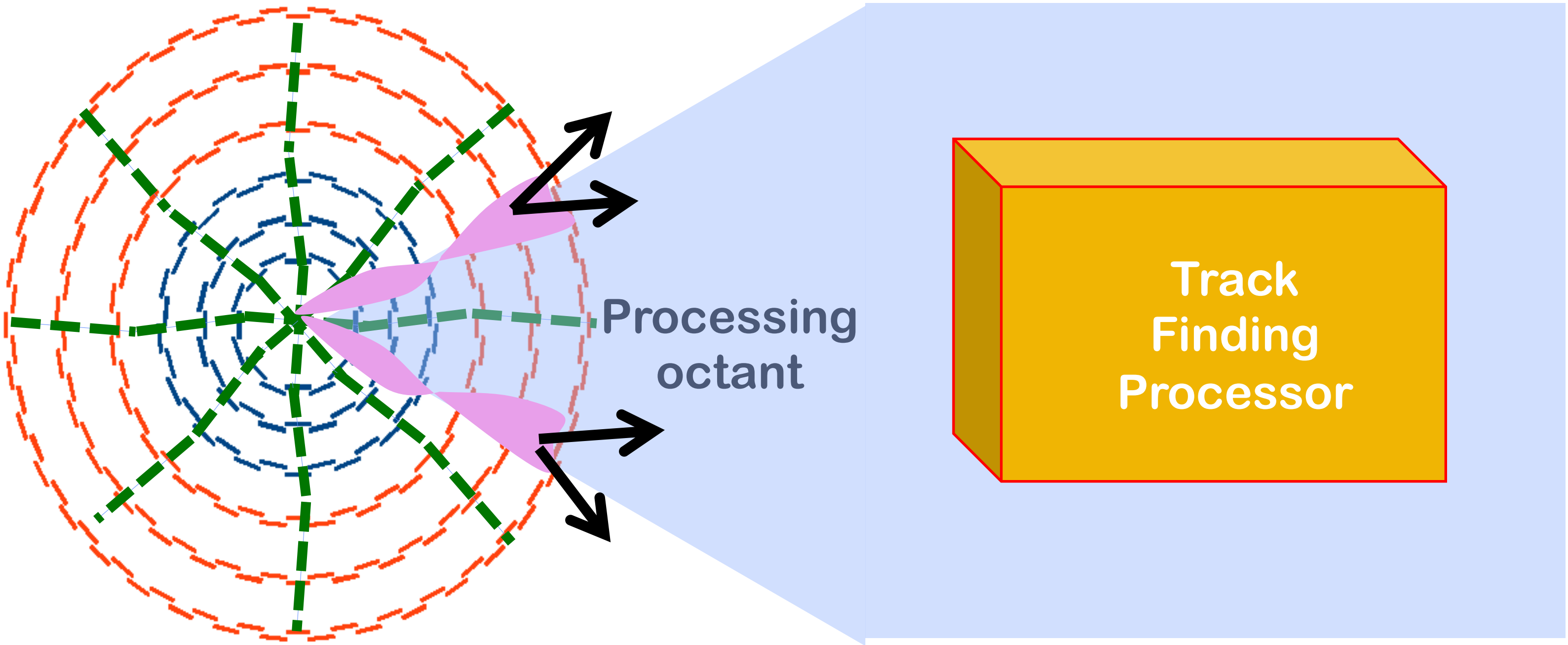
- Our "Track-Finding Processor" (TFP) is responsible for reconstructing all the tracks in one φ octant, known as a "processing octant".

- We rotate the "processing octant" by 1/2 octant w.r.t the "detector octant".
  - To reconstruct particles within its processing octant, a TFP never needs stubs from > 2 detector octants, despite track curvature.



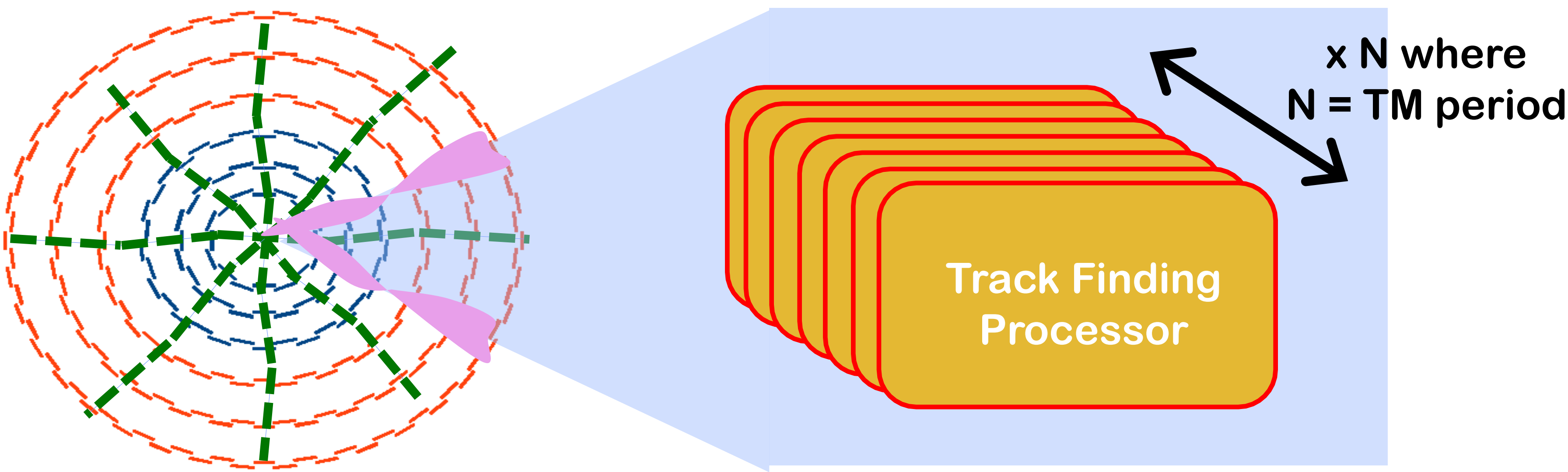
Allowing for track curvature

- The DTC duplicates stubs in overlap region near processing octant boundaries (pink), & sends them to the two neighbouring processing octants.
  - No sideways communication is needed between Track Finding Processors from neighbouring processing octants.
  - Makes it natural to demonstrate system by building a TFP.



Time Multiplexing

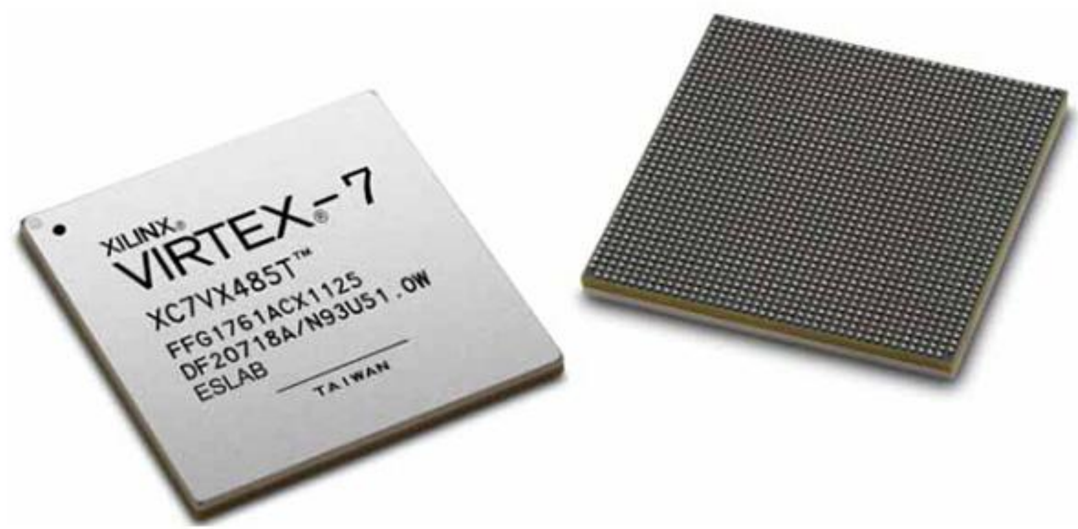
- In each processing octant we have N identical TFP's, where each TFP processes only 1 event in N. This is *time multiplexing*.
  - The DTC sends each event in turn to a different TFP.
  - With current electronics/links, we choose N = 36.
- Nice features of time multiplexing:
  - Already successfully used in current CMS trigger.
  - By building just one TFP, we can demonstrate that final system will work.
  - In final system, a spare TFP cabled to DTCs allows recovery in case one TFP fails, or parasitic testing of new tracking algo during LHC running.



Track-Finding Processor (TFP)

- We implement the TFP using FPGA's.

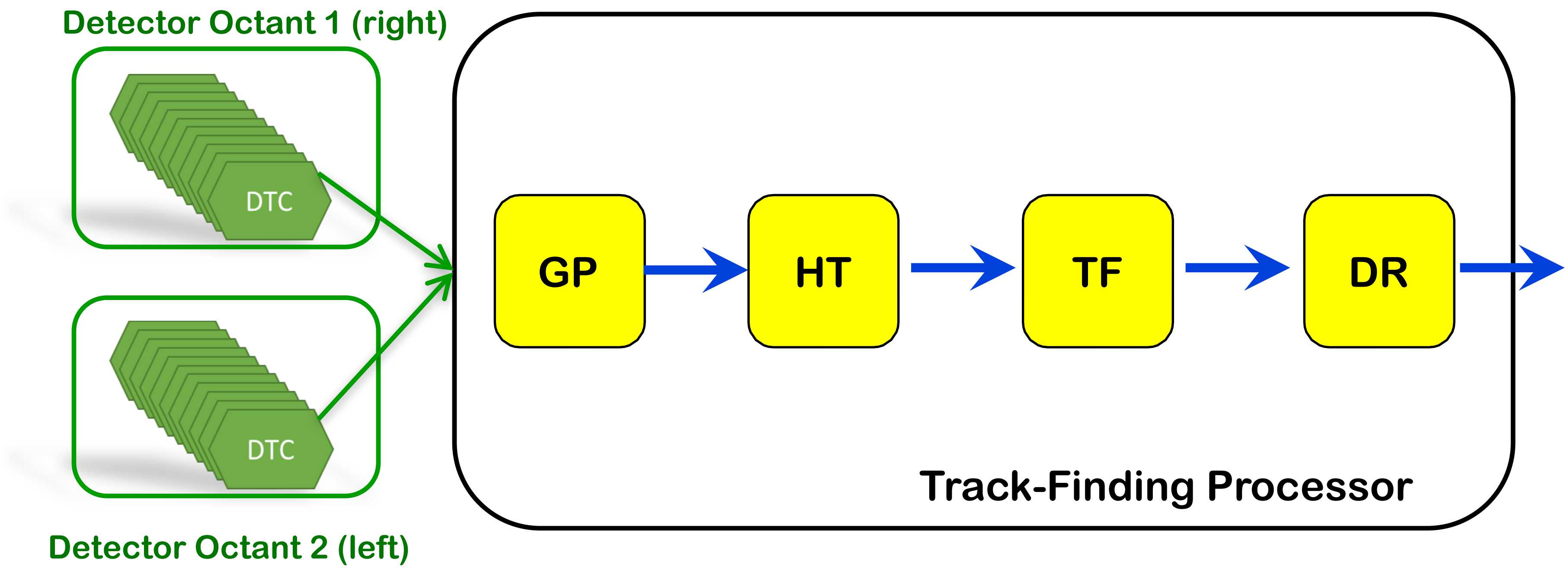
- Advantages:
  - Off-the-shelf component.
  - Flexibility to modify tracking algorithm based on LHC conditions or new ideas.



Track-Finding Processor (TFP)

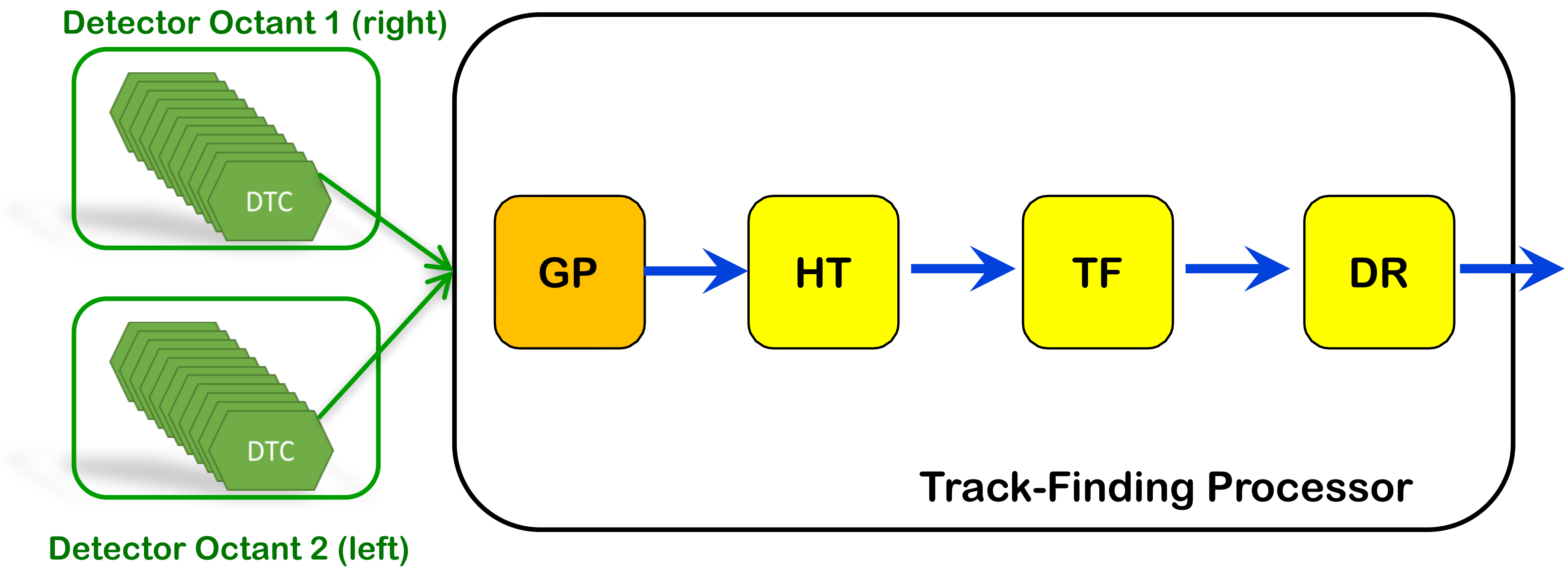
The TFP is divided into several logical blocks:

- Geometric Processor (GP): subdivides the octant into sectors.
- Hough transform (HT): does simple tracking in the r-φ plane.
- Track filter/fitter (TF): cleans tracks & fits their helix parameters.
- Duplicate removal (DR): for when we reconstruct the same particle twice.



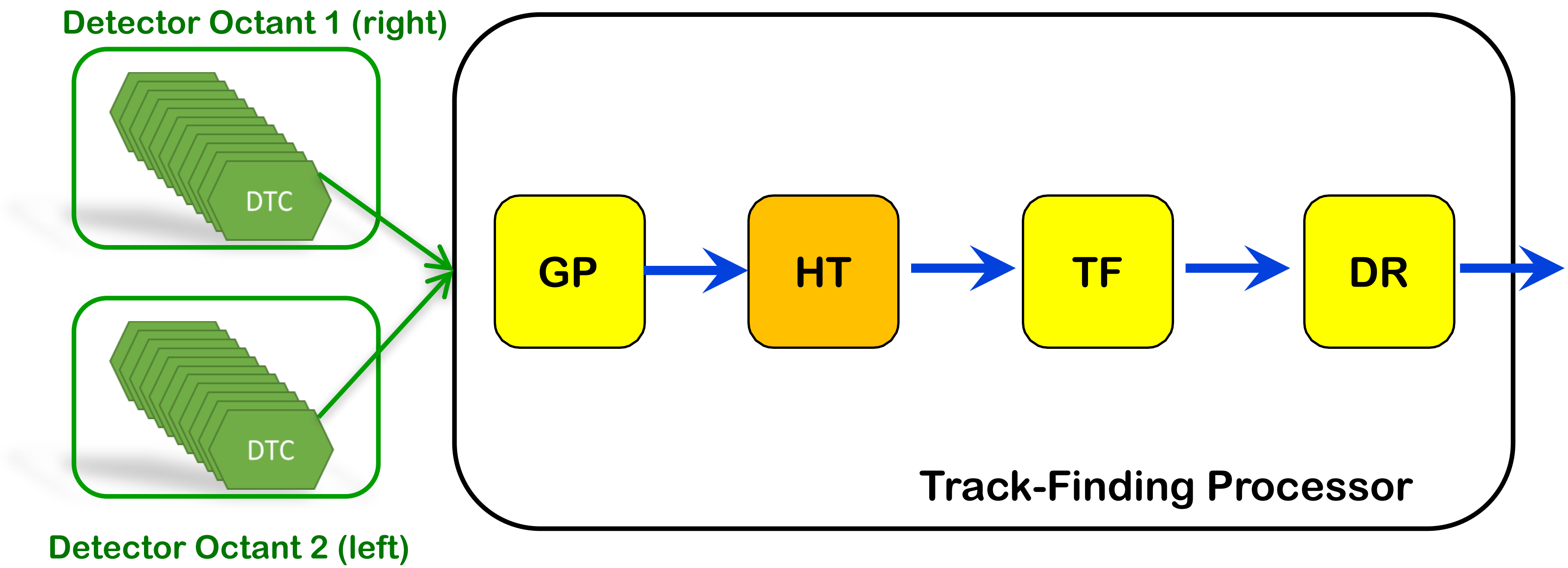
Geometric Processor (GP)

- Subdivides each octant into 2 x 18 sectors in $\varphi \times \eta$.
- Assigns stubs to one or more sectors
  & transmits stubs from each sector along dedicated link to next stage.
- Duplicates stubs if consistent with more than one sector,
  due to track curvature or beam-spot length.
- Formats the stub data for easier use by next stages.



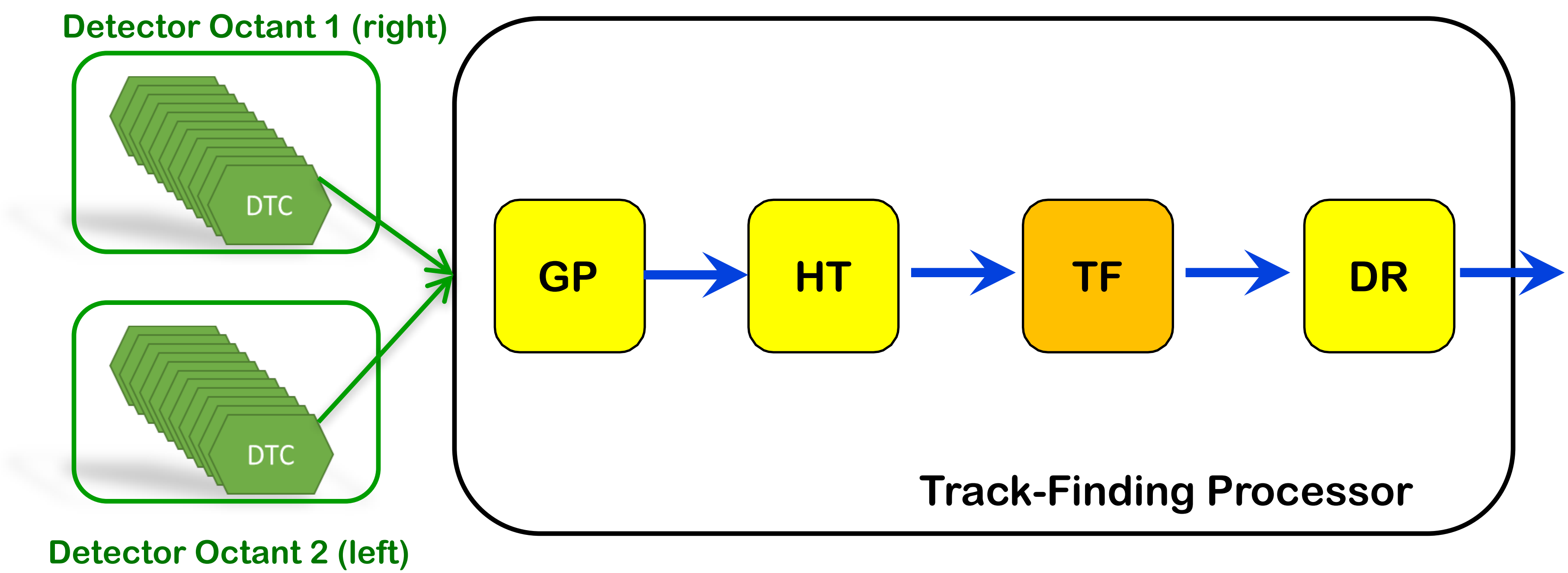
Hough Transform (HT)

- One HT is used within each sector, to do fast track-finding in the r-φ plane.
- The algorithm's simplicity makes it good choice for an FPGA.
- Our sectors are narrow in rapidity, so tracks found by HT are also ~ consistent with straight line in r-z plane.
- The HT hugely reduces the data rate (finding ~1.1 tracks/event/sector in "ttbar+200 pile-up").
  - Permits downstream algorithms be more sophisticated.



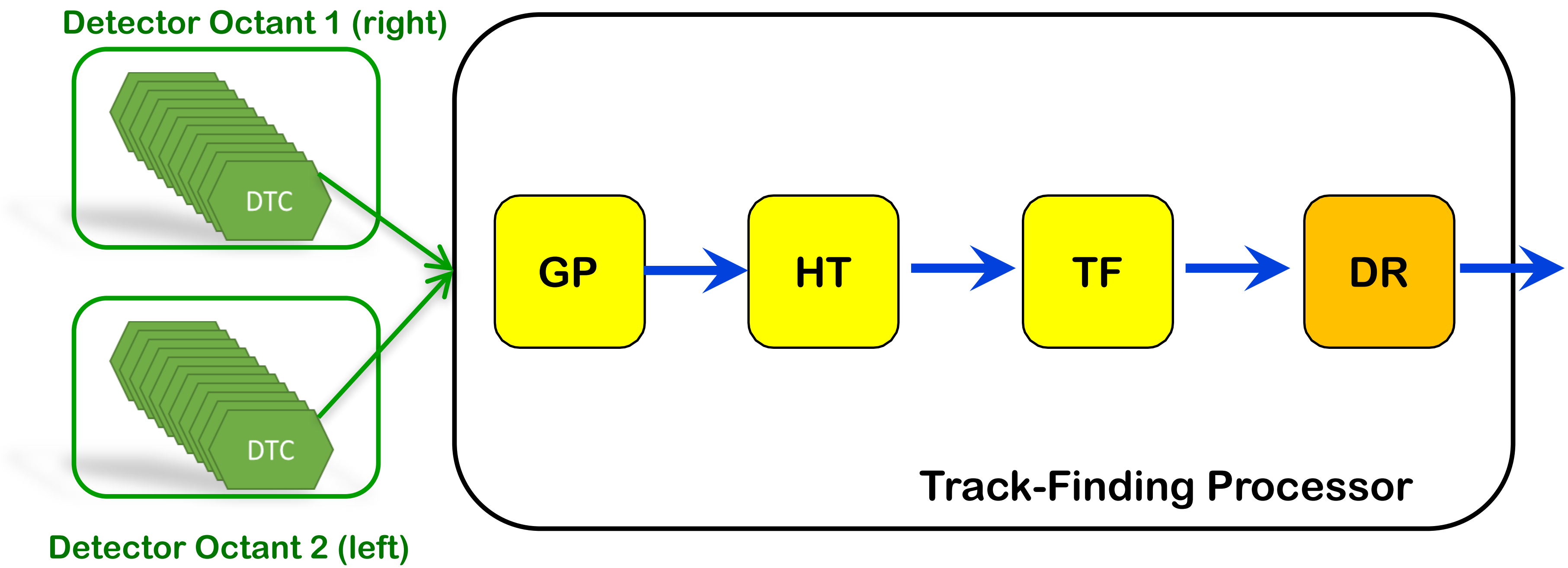
Track Filter/Fitter (TF)

- We fit track helix parameters using a Kalman Filter (KF).
- KF also rejects incorrect stubs (based on residuals) & fake tracks.
- CMS already uses KF for offline track reconstruction.
  - Huge reduction in data rate provided by the HT lets us to use this sophisticated fitting algorithm.



Duplicate Track Removal (DR)

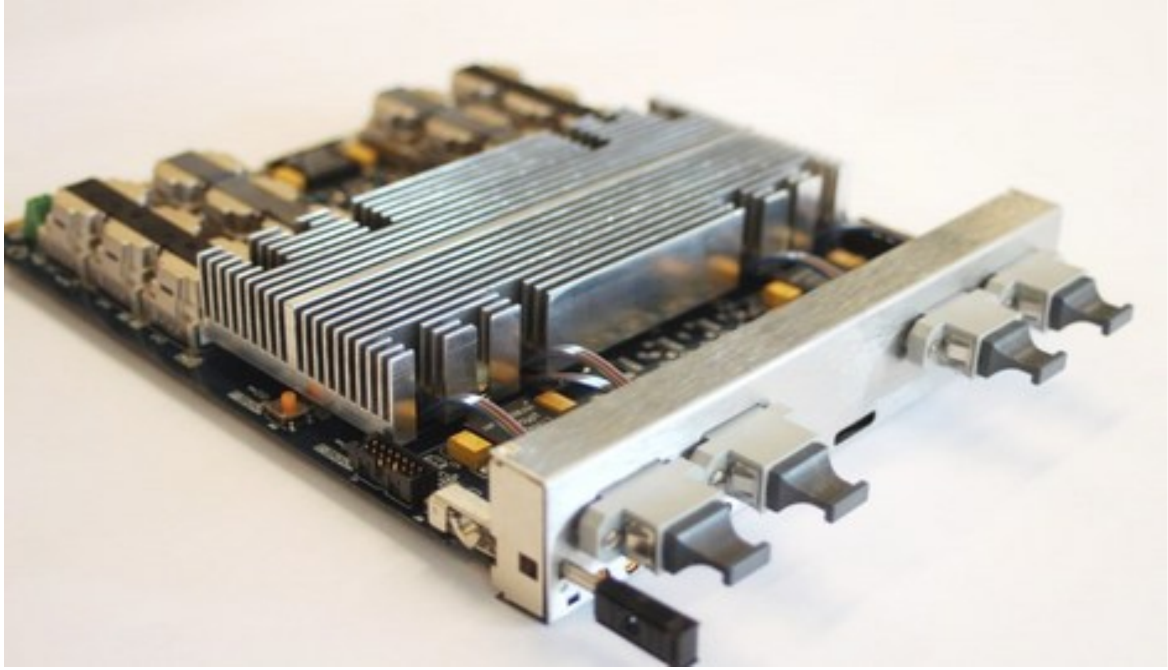
- Kills extra tracks if we accidentally reconstruct same particle multiple times.
- We can tell if a track is a duplicate simply by looking at that *individual* track.
  - Much simpler than conventional DR algorithms, which identify duplicates by comparing pairs of tracks!
  - Simplification made possible by an understanding of how duplicate tracks form in a HT.



2016 Hardware Demonstrator

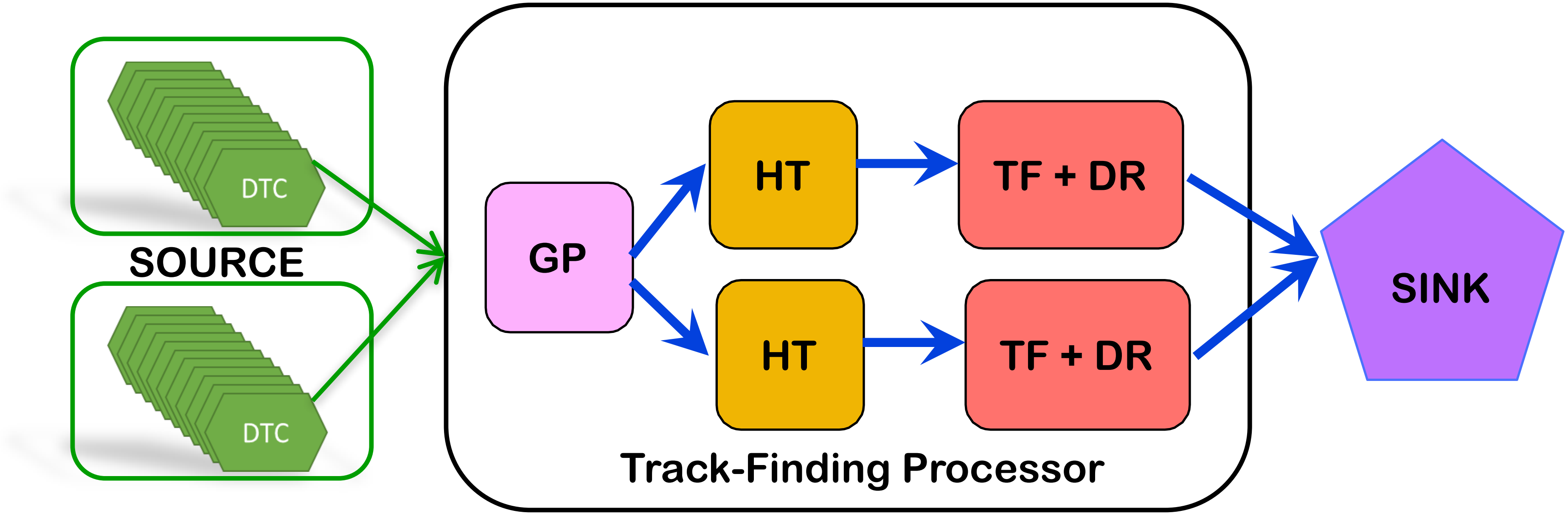
- Our 2016 hardware demonstrator corresponds to one track-finding processor.
  - Reconstructs tracks in an entire φ octant (= 1/8 of the tracker solid angle)
  - Reconstruct them for one LHC event in 36 (= time multiplexing factor).

- Implemented on a number of MP7's, where MP7 is generic μTCA card widely used in CMS, equipped with a Virtex7 FPGA with ~1 Tb/s I/O capacity.



2016 Hardware Demonstrator

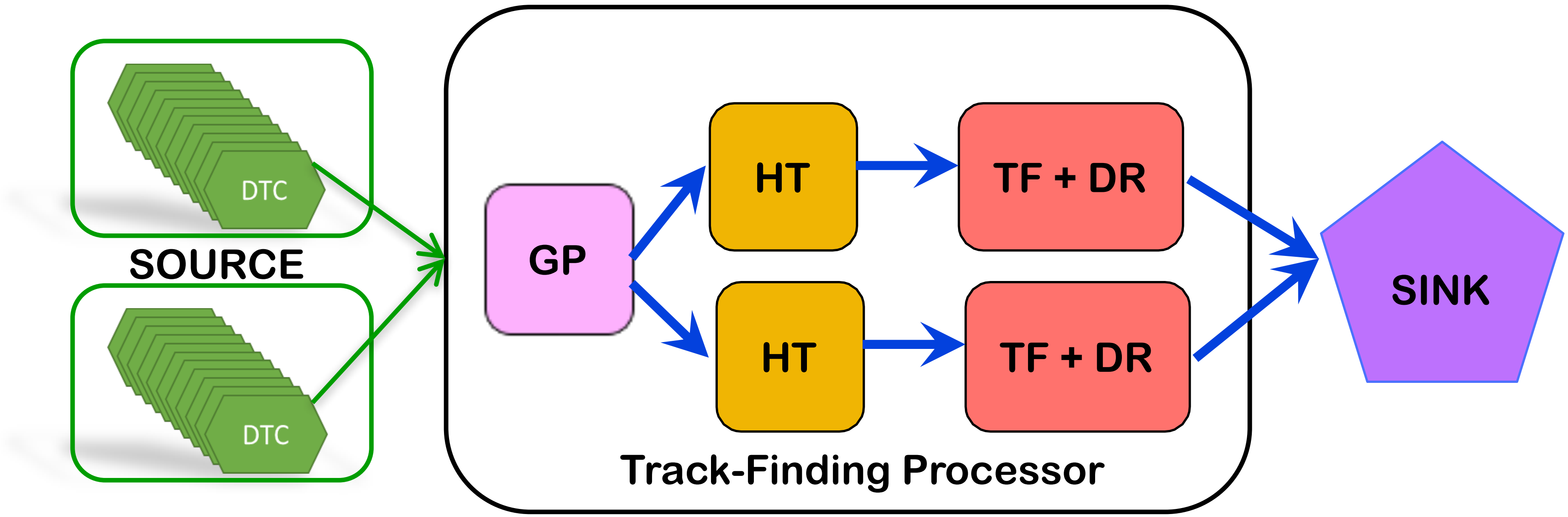
- 1 MP7 implements the Geometric Processor (GP).
- 2 MP7 implement the Hough transforms (HT) for all the sectors in an octant.
- 2 MP7 implement the Kalman filter track fitter (TF).
- The Duplicate Removal algorithm (DR) is so simple that it fits in the same MP7's as the TF.



2016 Hardware Demonstrator

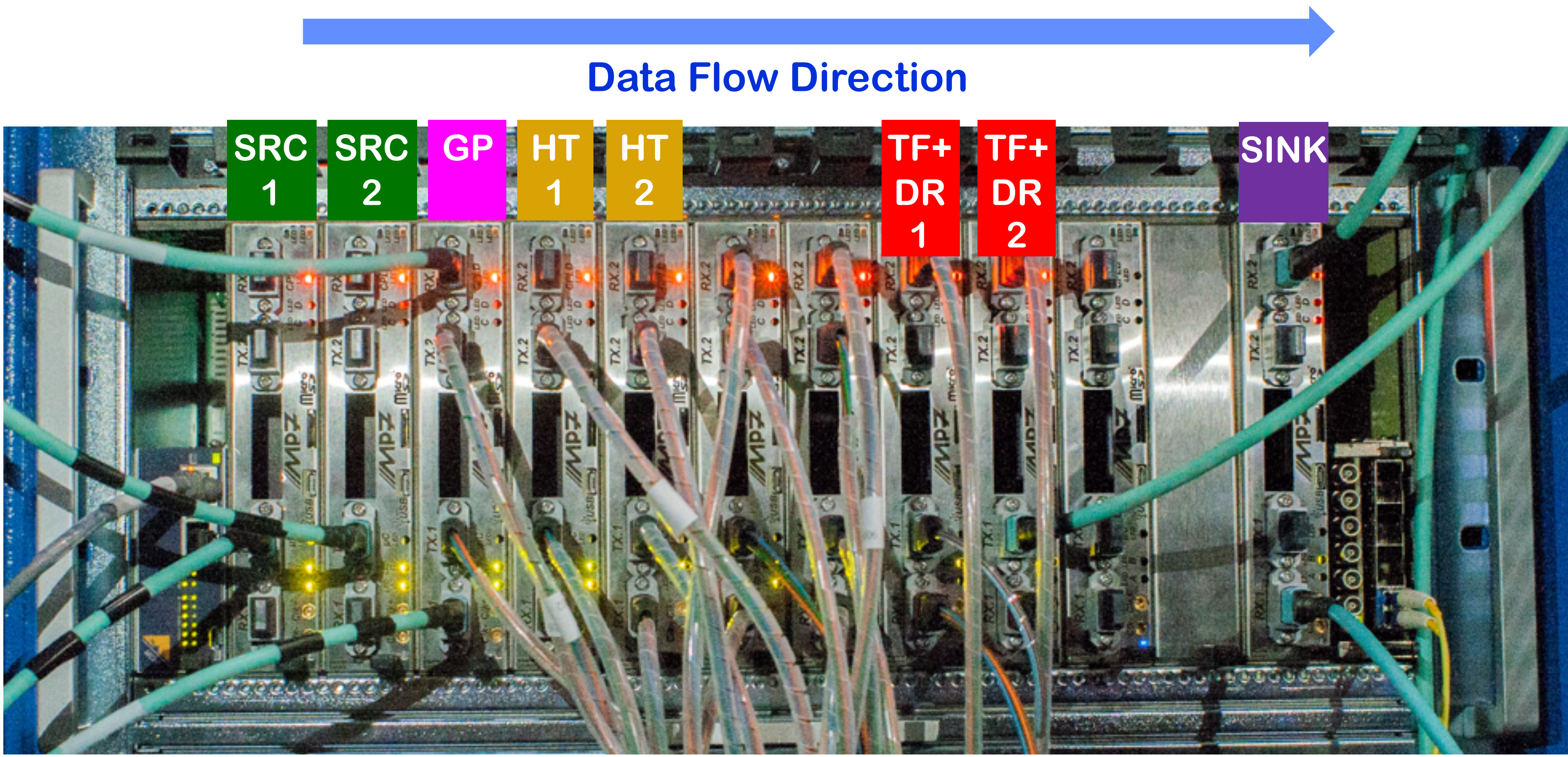
- 2 MP7 are used as a SOURCE, each transmitting the stub data produced by all DTC within one detector octant, for one event in 36.
- The SOURCE can store stub data from 30 MC events (loaded via IPbus).

- 1 MP7 is used as a SINK to capture tracks produced by the TFP, which can then be read out via IPbus.



2016 Demonstrator Hardware in Situ

- Demonstrator in Tracker Integration Facility (TIF) in B186, CERN.
- 11 MP7 installed in Schroff μTCA crate, connected together by optical fibres.



Hardware Demonstrator Flexibility

- Our demonstrator is very flexible!

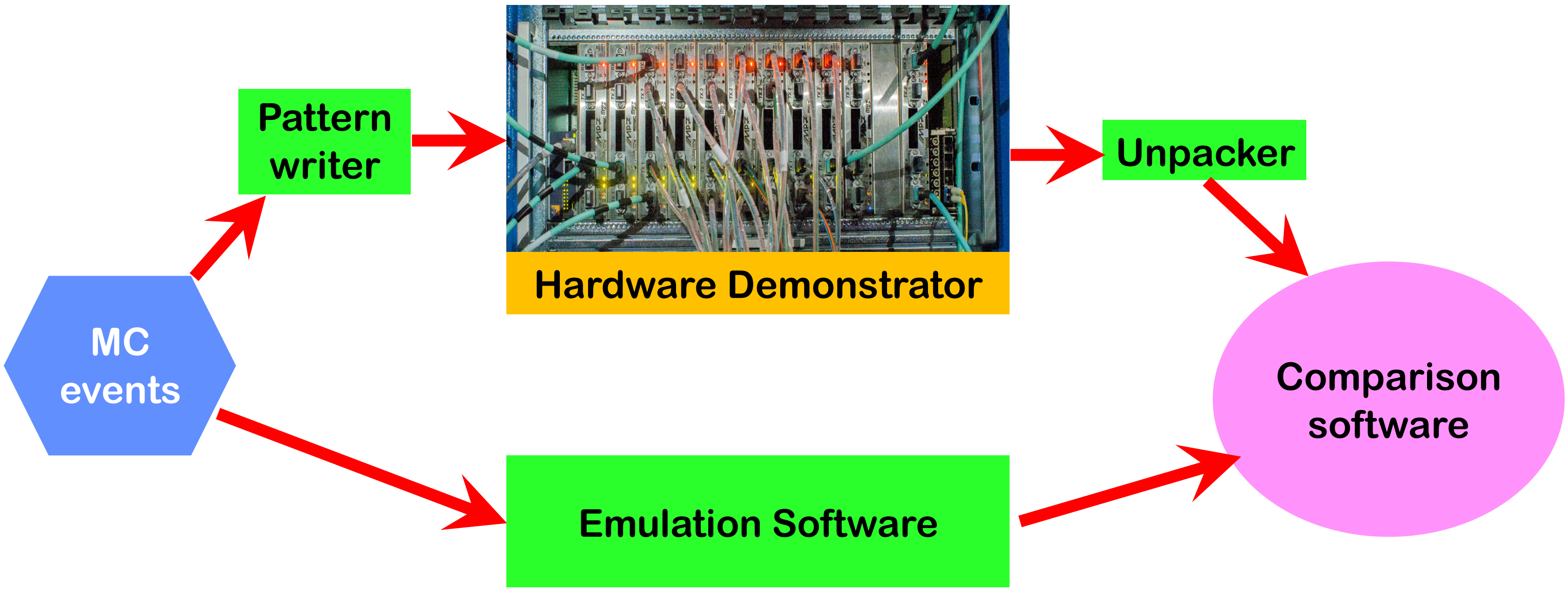
- In past ~2 years, we have used it to explore wide range of architectures:
  - 3 HT firmware implementations.
  - 2 duplicate track removal algorithms.
  - An "R-Z Seed Filter" to clean the tracks from the HT, requiring them to be consistent with a straight line in r-z.
  - A "Linear Regression" track fitter as alternative to Kalman filter.

- The solution we present today "GP+HT+KF+DR" is our current favourite.
  - Corresponds to successful completion of the most ambitious of the 3 alternative designs we described at the Oct. meeting.

- For future, have several interesting ideas to improve HT & KF algorithms. And demonstrator allows wider design variants to be explored.

Hardware Demonstrator Validation

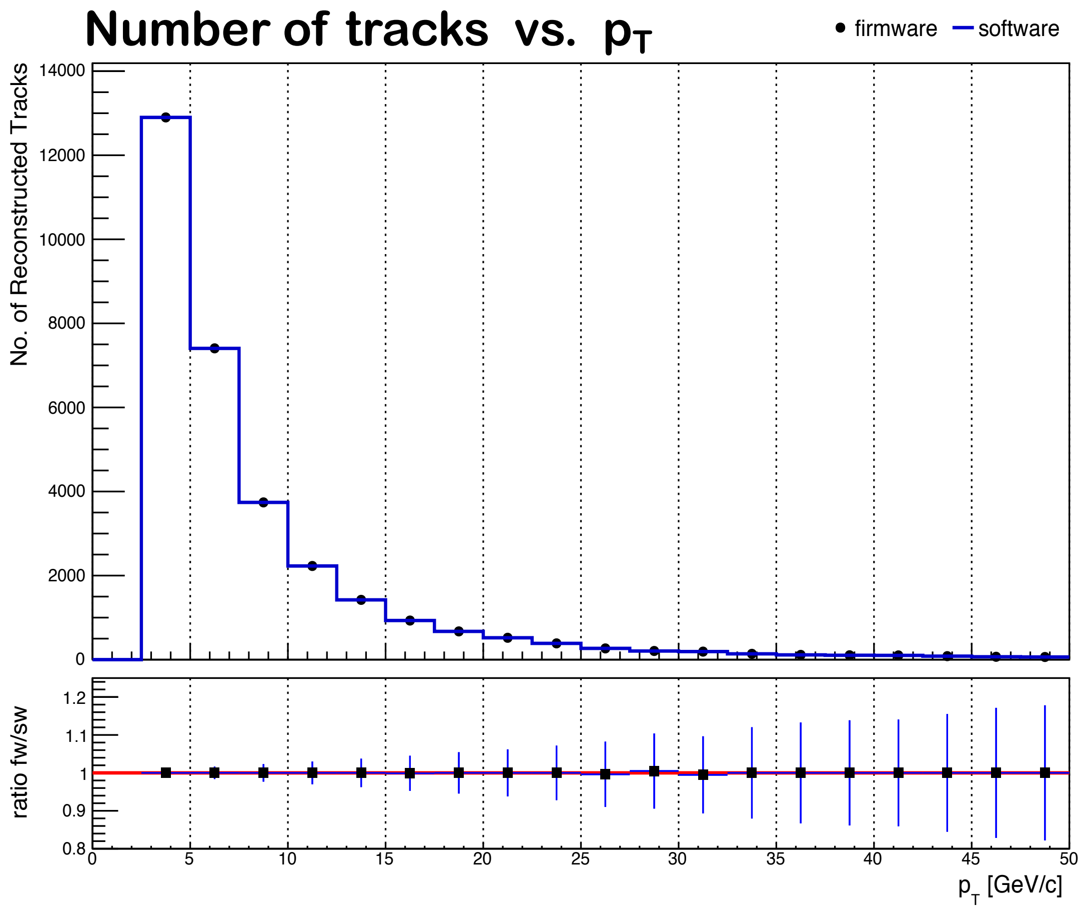
- Inject stubs from MC events (e.g. "ttbar + 200 pileup") both into hardware demonstrator (via IPbus) & into C++ emulation software.
- Read out tracks found by hardware & compare them for consistency with those predicted by software.
- We loop over all 8 octants, so as to obtain results for entire tracker.
- Can do this either for the entire demonstrator chain, or for individual elements of the chain, such as the Hough transform.



Hardware Demonstrator Validation

- Example: comparison of tracks found at end of track reconstruction chain by duplicate track removal algorithm: hardware (points) vs. software (histogram).

- Such studies are essential to uncover issues, such as truncation effects due to excess data rate.



Tracking efficiency definition used in talks

- When presenting results for our full demonstrator chain, either in simulation or hardware, we use the officially agreed definition of tracking efficiency.
  - This defines a particle to be successfully reconstructed if:
    1) it shares stubs in ≥ 4 tracker layers with a reconstructed track.
    2) & every single stub on the track was produced by this one particle.

  This allows comparison with AM/Tracklet results.

- When presenting results for parts of our demonstrator chain, (e.g. if we only run the GP+HT), we remove the requirement (2).
  - This is because tracks will obviously contain some incorrect stubs if only part of the chain has been run.

Conclusions

- Track-finding implemented in generic FPGA boards using time-multiplexing.
  - Off-the-shelf chips.
  - Give flexibility to modify algorithm.
  - Can demonstrate design by building single TM slice.

- Our hardware demonstrator reconstructs tracks in 1/8 of the tracker solid angle, from stub data produced at 200 pile-up.
  - You will see its tracking performance & latency meet L1 tracking needs of CMS.
  - This is achieved with today's technology! Future technology should reduce cost/improve performance.

- Our design could form basis of L1 tracking system for CMS, but we are confident even better solutions can be found in coming years.
  - Expertise from all groups needed to achieve this.

Remaining TMTT presentations in this session

Three talks describing the logical blocks within our tracking chain & results obtained from them in our demonstrator:

1) The Geometric Processor + Hough Transform
2) Track Fitting: Kalman Filter
3) Duplicate Track Removal.

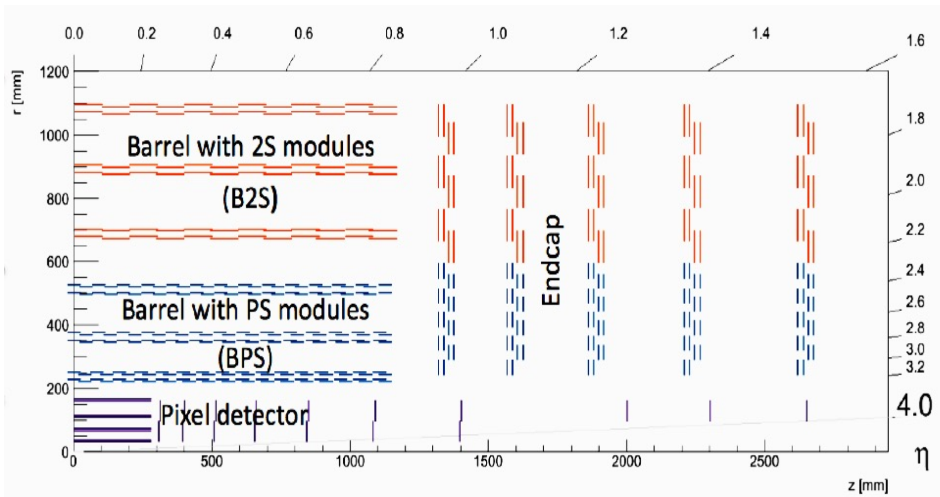
Then:

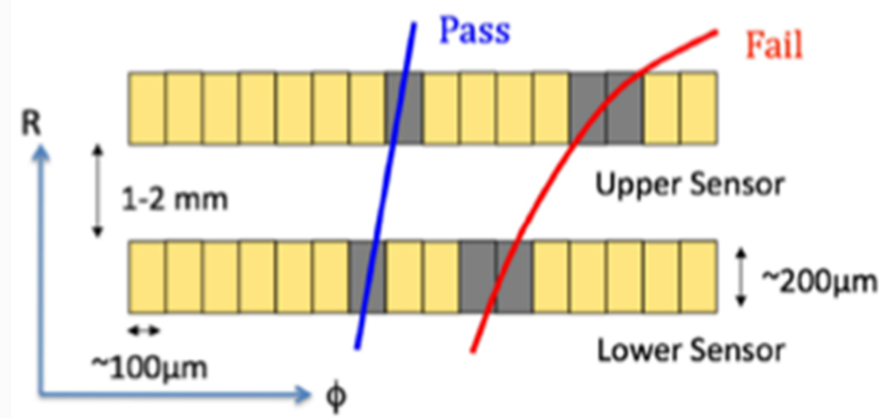
4) Tracking performance studies with simulation.
5) Demonstrator results from full tracking chain.
6) Scaling up from our demonstrator to a design for 2026 running + Summary.

Back up

The upgraded CMS Tracking Detector

- Each tracker module consists of 2 closely spaced silicon sensors.
- A charged particle produces a pair of hits (known as a `stub') in these two sensors.
- Assuming the particle originates from the LHC beamline, the relative position of the two hits determines the track Pt.





- On-detector electronics transmits only stubs consistent with Pt > 2 to 3 GeV to off-detector electronics, reducing by factor ~30 the number of stubs that L1 track-finding electronics must handle.