Introduction to L1 track-finder options for the CMS detector upgrade

Ian Tomalin (RAL)

Introduction

- WP4 particle physics work focussed on development of L1 track-trigger for CMS upgrade.

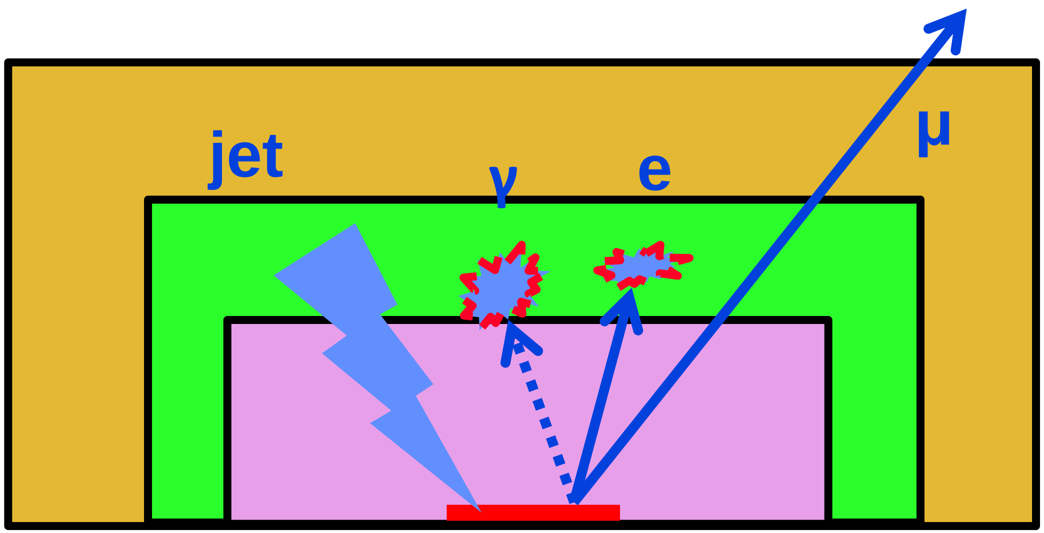
- From 2025+, HL-LHC produces an event containing ~140 pp collisions every 25ns.

- CMS L1 trigger must select interesting events within ~12 μs (max. time data can be buffered in front-end electronics), whilst rejecting ~99% of boring events.

- To achieve this, L1 trigger will use not only data from CMS calorimeter & muon chambers (traditional), but also reconstruct (high Pt) charged particle tracks in the tracker detector.
  - Tricky! --- LHC will produce ~5000 charged particles every 25 ns! And tracks must be found in 100 Tb/s of raw data.

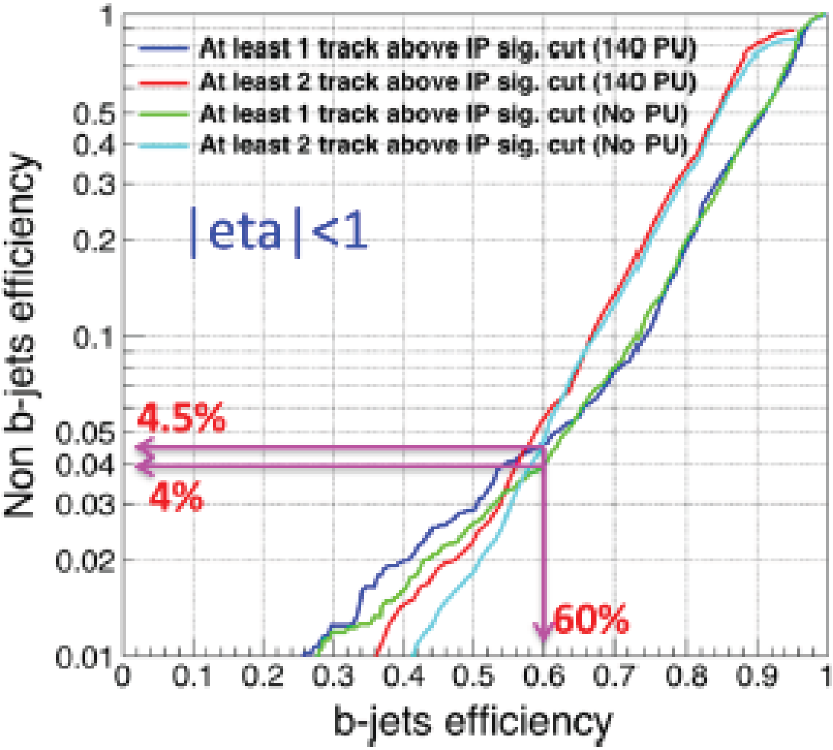
Justification

- If we keep current trigger thresholds (energy cuts etc.) & don't use L1 track reconstruction, the CMS L1 trigger rate would be 1500 kHz. Our high-level trigger (= PC farm) could not cope with this!
- Adding tracks allows L1 trigger to be more selective: rate = ~260 kHz.
- Why?
  - Muons: tracks improve Pt resolution
  - Electrons: tracks distinguish them from photons
  - Jets + EtMiss: tracks check they come from main pp collision vertex, not from boring pileup vertices.

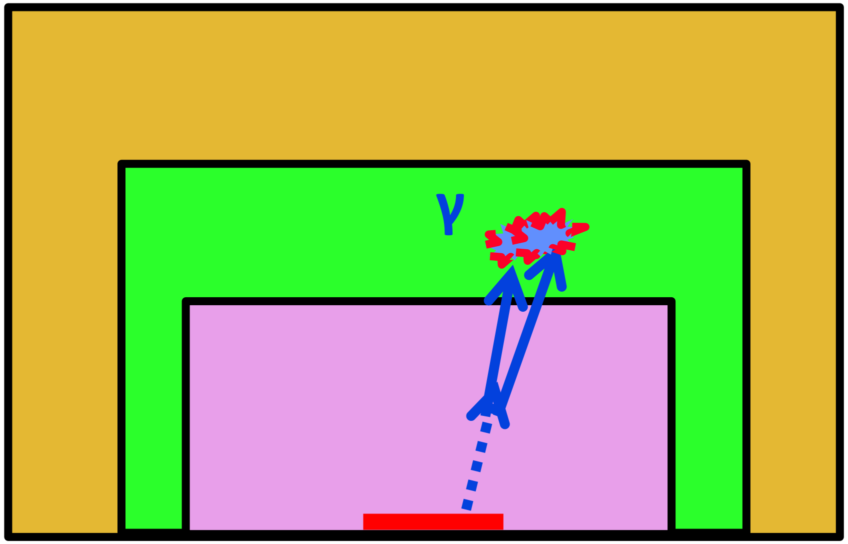


Justification – what about using the pixel detector?

- INFIERI involved in more speculative proposal to include inner pixel tracker in track reconstruction for L1 trigger. Why?
- Higgs boson + lots of exotic physics decay to b hadrons. These fly mm before decaying. Pixel tracker only 4 cm from beamline, so good enough vertex resolution to see this! Main tracker is 25 cm from beamline, so can't!



- Also, pixel tracker can spot converted photons pretending to be electrons from pp collision!



CMS L1 Track Options

- CMS is doing R&D into ideas for reconstructing tracks within ~5μs.
  INFIERI people involved in majority of them.

- Two basic proposals, each with two variants:

  1) Do tracking using only **FPGA**s:
     - Running "Tracklet" *or* "TMT Hough transform" tracking algorithms.

  2) Do rough tracking with custom **Associative Memory** chips
     & clean up these tracks with FPGAs.
     - AM06 *or* VIPRAM AM chips.

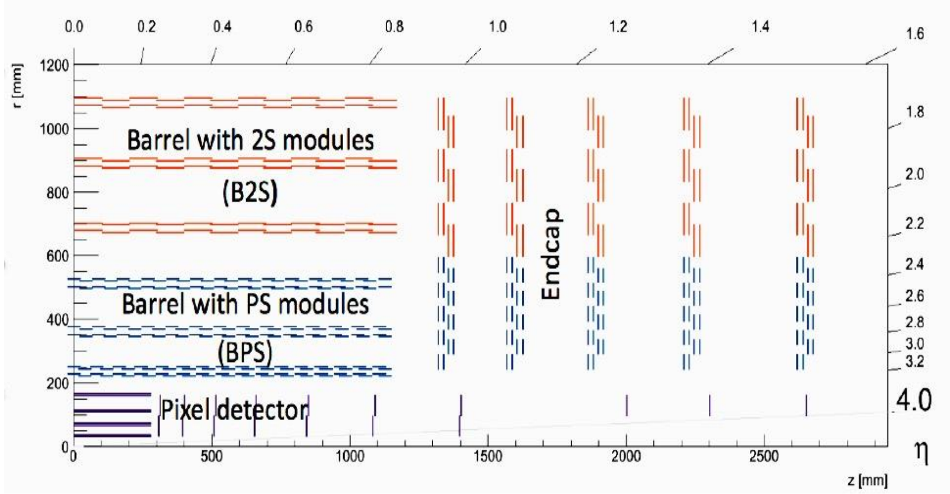
  \+ Tentative idea to use inner pixel tracker in L1 trigger.

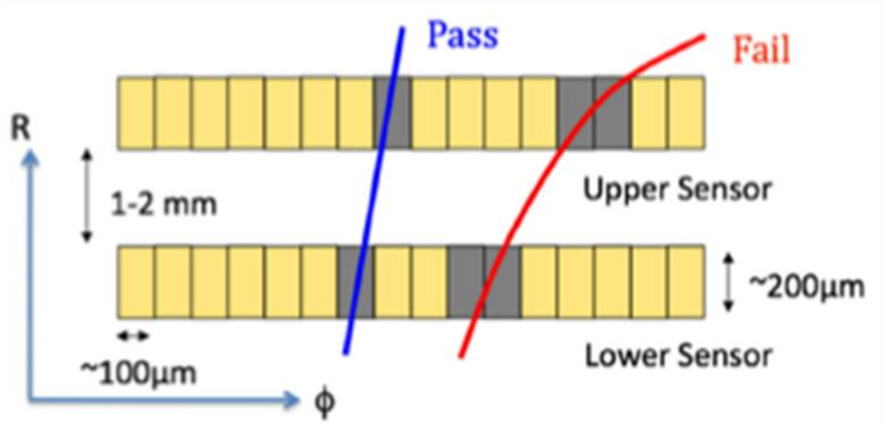
- CMS will narrow down the choices for Technical Design Report in Spring 2017, & therefore wants hardware demonstrators for all proposals by Nov. 2016.
  - These must reconstruct tracks in an angular slice of the CMS tracker using raw data from simulated LHC events as input.

The 2025 CMS tracker detector

The CMS tracker detector will be replaced in 2025 & its design aims to facilitate track reconstruction for L1 trigger:

- 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.

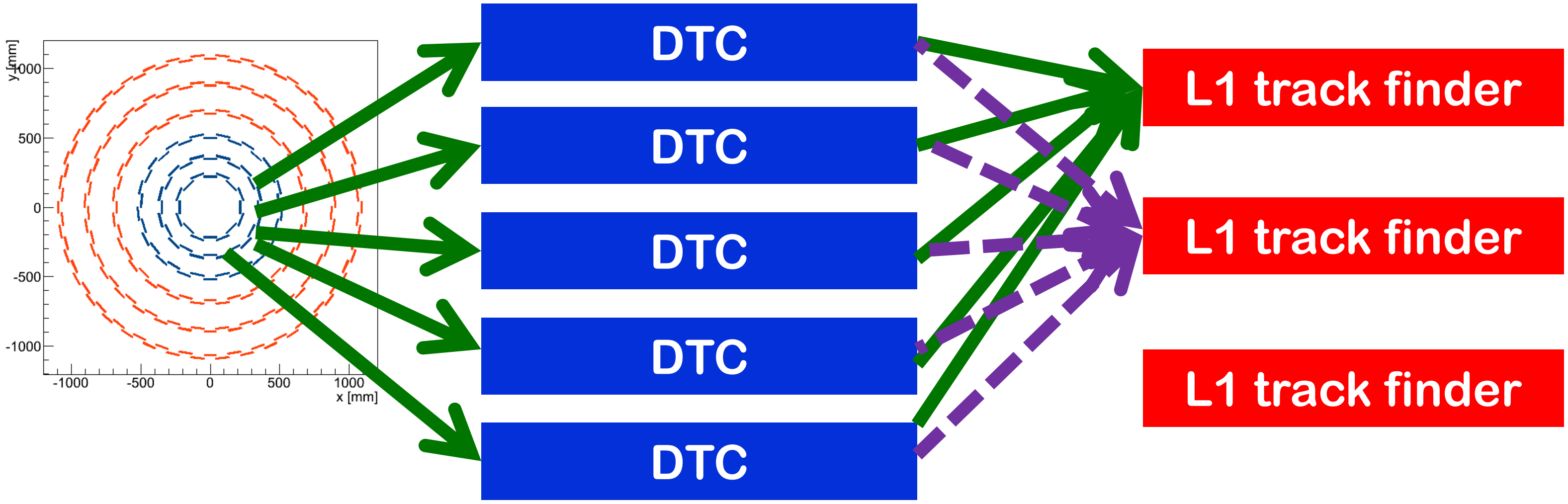




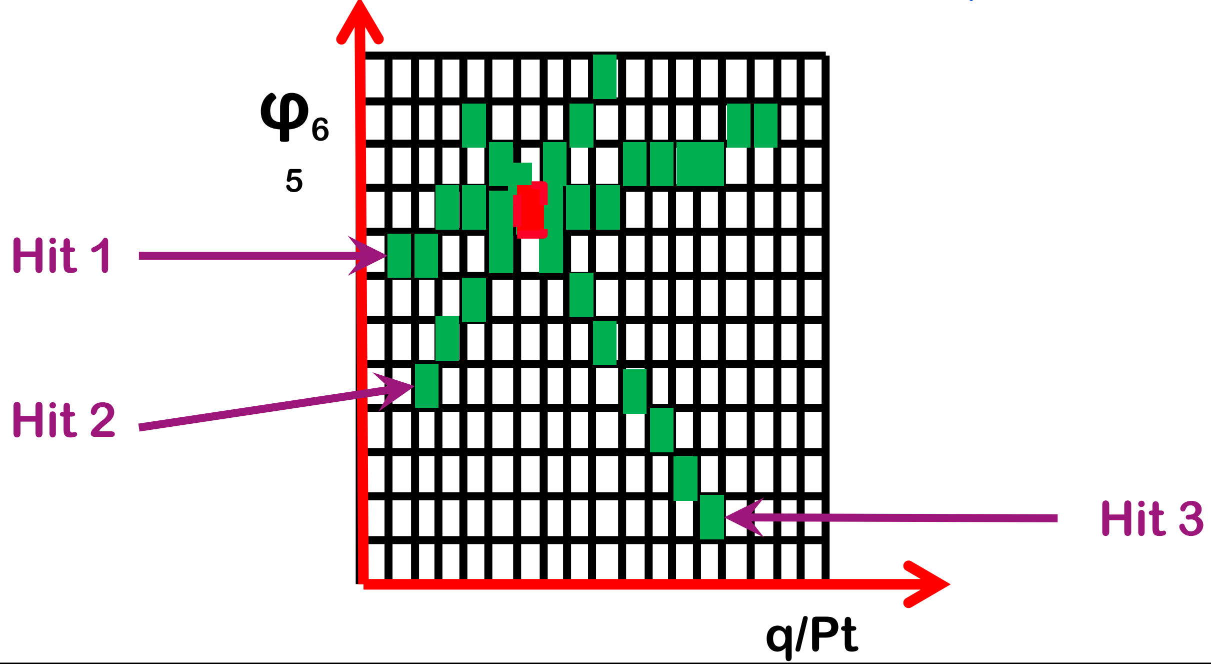
- On-detector electronics transmits only stubs consistent with Pt > 2 GeV to off-detector electronics, reducing by factor ~30 the number of stubs that L1 track-finding electronics must handle.

FPGA solutions
- Time-multiplexed Hough transform

- *Details in talks of Luigi Calligaris & Davide Cieri tomorrow.*
- ~300 "DTC" boards each read a narrow angular region of tracker.
- On 1st event, they all send their data to 1st "L1 track finder" board,
  2nd event ➔ 2nd "L1 track finder board" etc.,
  up to 36th event → 36th board; then 37th event → 1st board again.
- Each L1 track-finder board sees only one event in 36, so has 36 LHC pp bunch crossings to find tracks.
- This is most extreme use of time-multiplexing of any L1 track solution. – Each L1 track finder board responsible for 1/9 of tracker solid angle.



- Track-finding done using 'Hough-transform' -- simple track-finding algorithm, suitable for implementation in an FPGA.
- In r-ϕ plane, tracks from LHC beamline form circles described by two parameters (q/Pt, ϕ_{65}).
- Create 2D histogram of (q/Pt, ϕ_{65}). For every stub, put an entry in any bin of the histogram, if a track of that (q/Pt, ϕ_{65}) would pass through the stub.
  - Each stub gives a curve indicating which track (q/Pt, ϕ_{65}) values it is consistent with. Where several curves intercept, we have found a track!



FPGA solutions
- Time-multiplexed Hough transform

- 2016 L1 track-finder demonstrator uses Virtex7 FPGA mounted on MP7 μTCA cards. The MP7 is existing CMS card with ~1 Tb/s I/O.

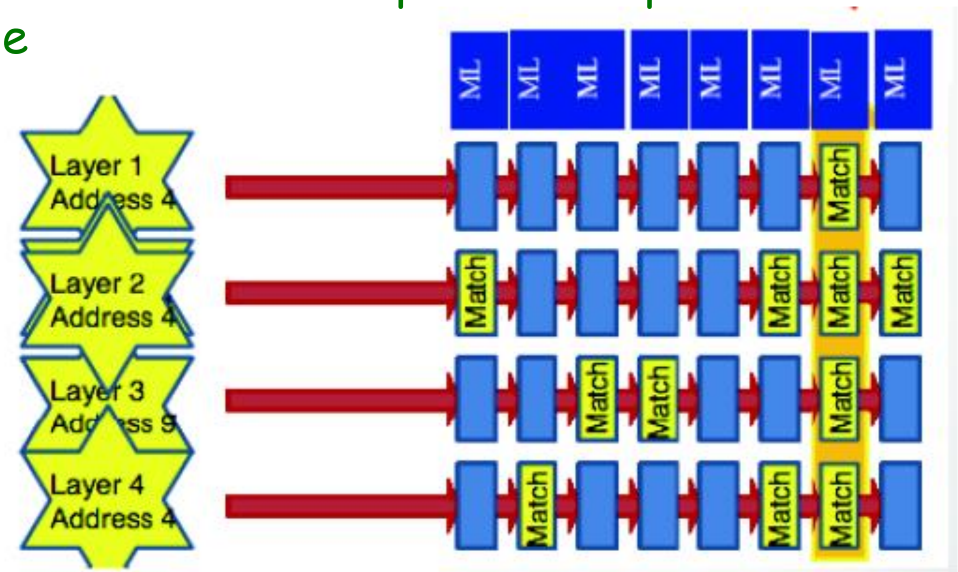
- One MP7 will be data source.
  One MP7 will organise data for input to the Hough transform.
  One MP7 will do HT.
  One or two others will refine tracks further.

ADVANTAGES OF FPGA SOLUTION

- No custom chips required.
- Flexible – algorithm can be changed at any time, due to good ideas, or unexpected physics or detector behaviour.

Associative Memory Solutions

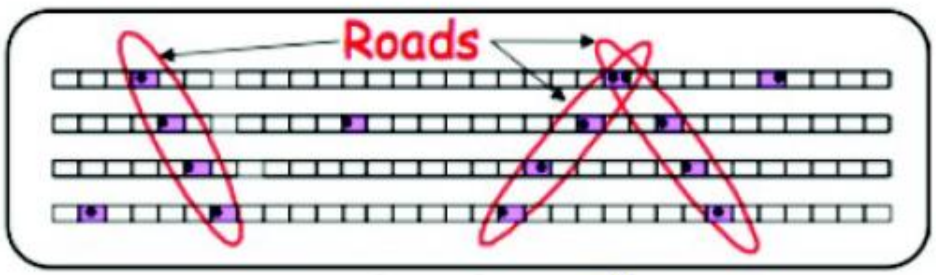
- *Details in Giacomo Fedi's talk. tomorrow*
- For each tracker hit, custom AM chips check in parallel which possible 'roads' it is compatible with, (where each road represents a possible trajectory of a charged particle through CMS tracker).



- If hits in in several tracker layers are compatible with the same `pattern', we have a track candidate!

ADVANTAGE:

- Rough tracks found very quickly.
- Subsequent FPGA only needs to tidy up.



Associative Memory Solutions

- INFN solution: Mount 16 AM06 chips, with 100k patterns each, on Pattern Recognition Mezzanine (PRM). Each PRM can find tracks in entire tower (= 1/48 solid angle of tracker).

- But one PRM could not process successive events. So need ~20x 'time-multiplexing'.
  - While one PRM is busy, another looks at next event.
  - Achieved by dedicating full ATCA crate to one tower, containing ~20 PRM mounted on Pulsar2b carrier boards.

- 2016 demonstator will use two ATCA crates, one for data source & one to find tracks in a tower.

Subsequent track refinement

- Both AM chip & TMT Hough transform FPGA solution produce rough track candidates (with lots of `fakes')
- Both therefore clean these tracks in an FPGA, & both use a similar algorithm!
  - Essentially put a helix through a pair of hits on the track candidate & check if the other hits on the track are compatible with this helix.
  - This is very similar to the FPGA `tracklet' track-finding solution! (Except they don't do rough track-finding first).

- After finding tracks, all groups fit them to obtain good estimate of the helix parameters.
  - Various fitting algorithms being explored (χ^2 Newtonian iteration, Kalman filter, Principal Components Analysis).
  - Could potentially be shared between groups.

My introduction is complete!

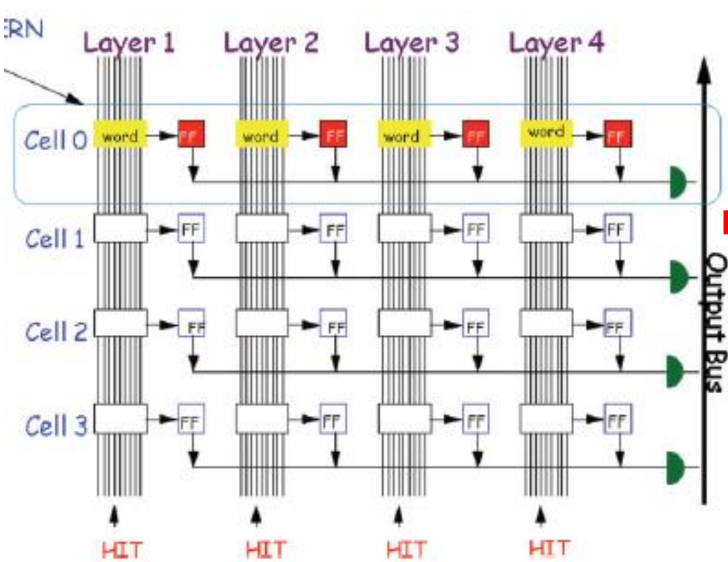
For detailed progress reports,
listen to
Giacomo, Luigi, Davide & Aurore's talks
Friday morning!

Backup

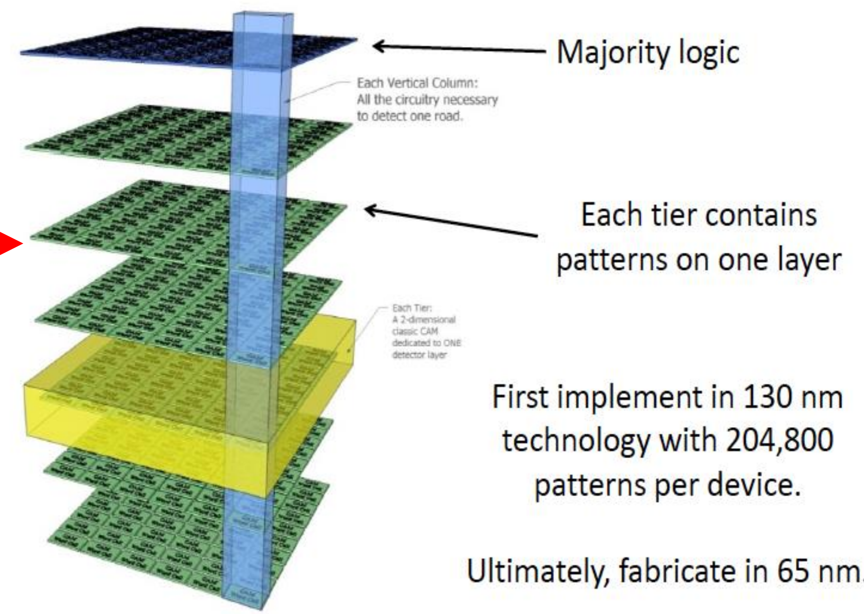
Associative Memory Solutions – FNAL AM chip

❖ 3D chip design (VIPRAM) simplifies data routing in chip.

2D INFN AM chip



3D FNAL VIPRAM chip



❖ Mount one prototype VIPRAM on FNAL PRM
(only 16k patterns, so unclear how will do demonstration?).

❖ This sits on Pulsar2 like INFN solution, so everything else in common.
(hardware, firmware, software, demonstator)!