Testbeam results of the first real time embedded tracking system with artificial retina

A. Abba[1,#], F. Caponio[1,#], M. Citterio[1], S. Coelli[1], J. Fu[1,2], A. Merli[1,2], M. Monti[1], N. Neri[1], M. Petruzzo[1,2]

for the INFN-RETINA collaboration

[1] INFN- Sezione di Milano, [2] Università di Milano

now at Nuclear Instruments

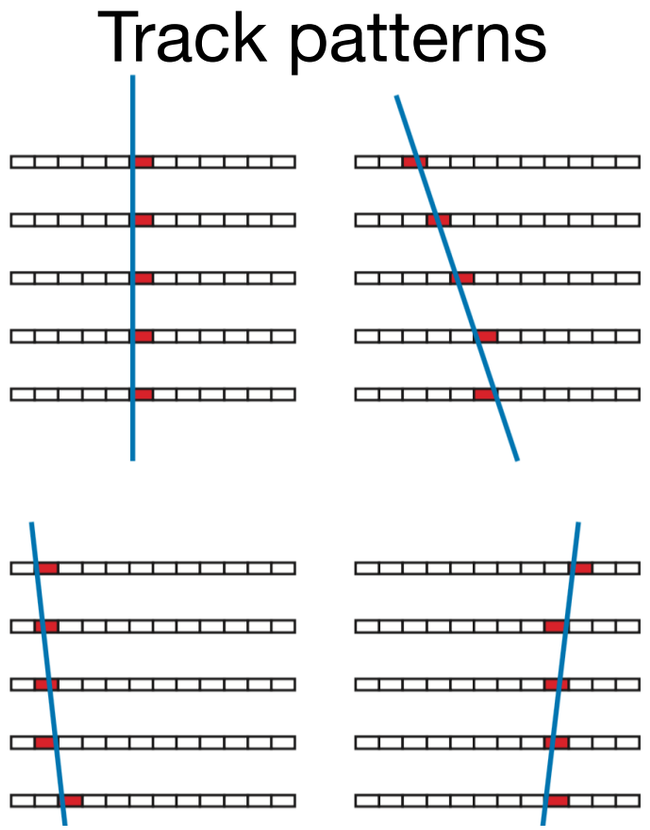
VCI 2016
Vienna, Austria
15 -19 February 2016

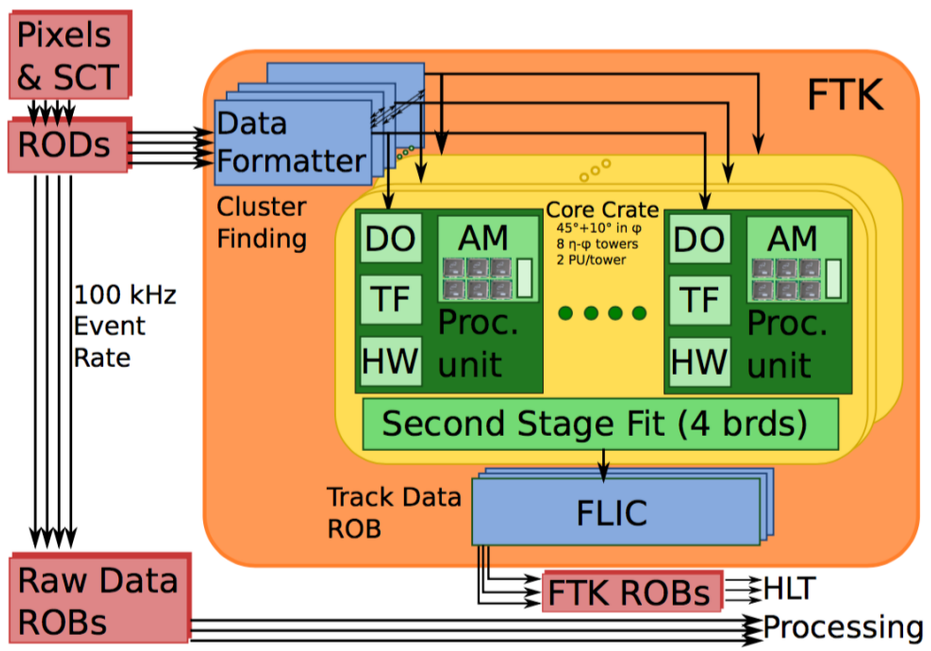
Outline

- Real time tracking
- Artificial retina algorithm and its implementation in hardware
- Detector prototype with embedded tracking capabilities
- Testbeam results
- Perspectives
- Summary

Existing fast track finders

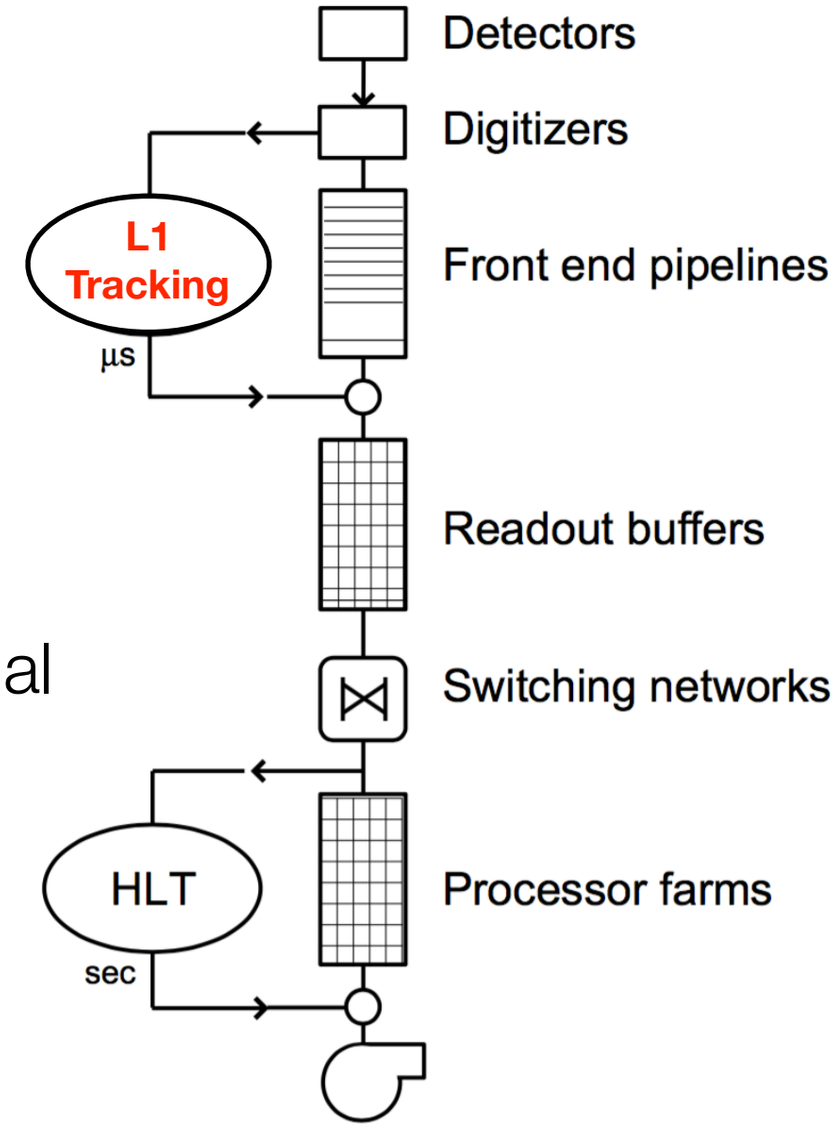
- Track pattern recognition without combinatorics
  - parallel matching of hits to pre-calculated track patterns, track parameters from linearised fit
  - use custom ASICs: Associative Memory (AM), based on content-addressable memory (CAM)
- First use in CDF experiment: SVT, latency 10μs and input rate 30 kHz
- FTK device in ATLAS use similar concept. Latency ~50μs and input rate 100 kHz





Real time tracking for HL-LHC

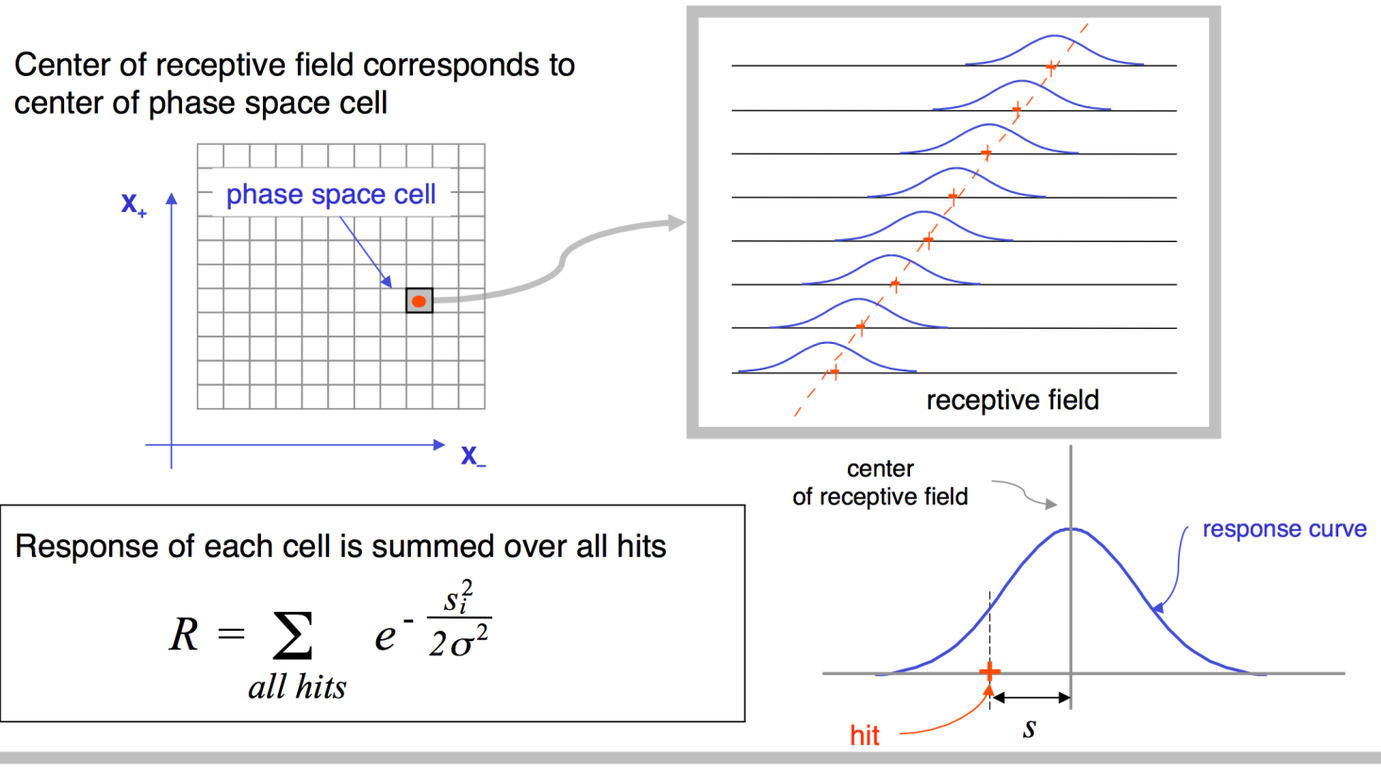
- Full exploitation of high luminosity LHC (HL-LHC) requires new detectors and trigger systems
- L1 trigger decisions based on tracking information are crucial:
  - reduce data rate to a sustainable level
  - maintain good efficiency and purity for signal events
- Real time tracking is extremely challenging at LHC: 40MHz throughput, large flow of data Tbit/s, short latency ≲1μs
- Necessary to find innovative solutions



Artificial retina algorithm

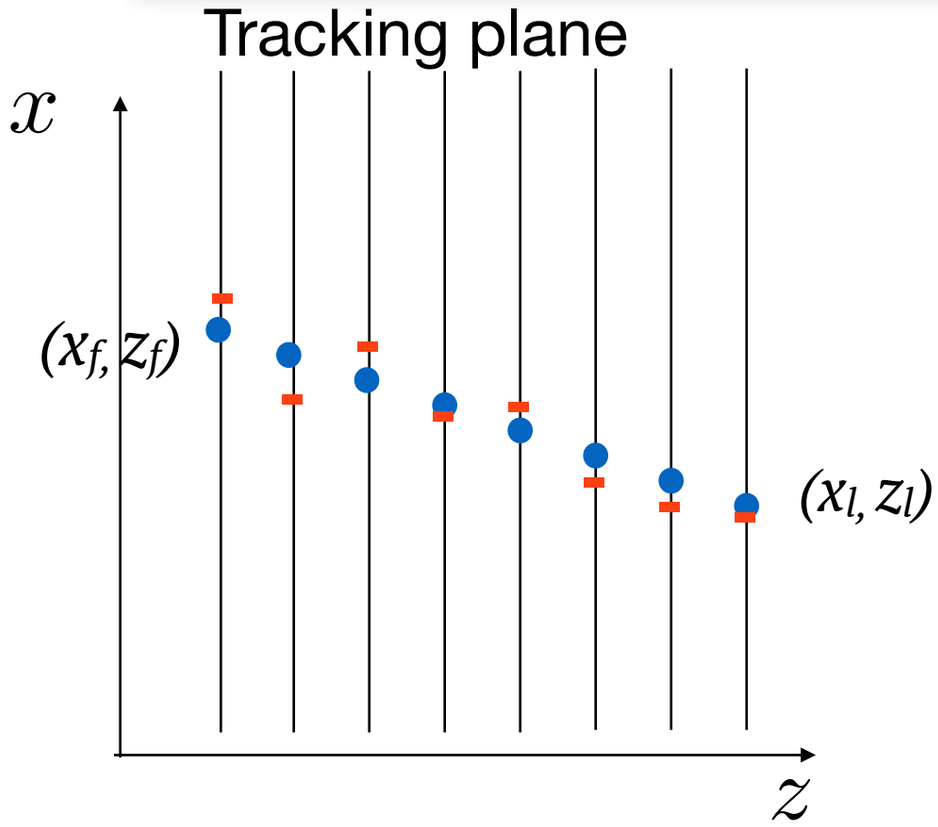
- Basic algorithm for fast track finding

L. Ristori, "An artificial retina for real-time track finding" [NIM A453 (2000) 425-429]



- Inspired by mechanism of visual receptive fields
  - massive parallelisation and analog response of track receptors (*R*)
  - pattern recognition and track fit by interpolation of *R* values

Track identified by retina algorithm



S_i distance between cluster and track receptor

cluster of hits

track receptors

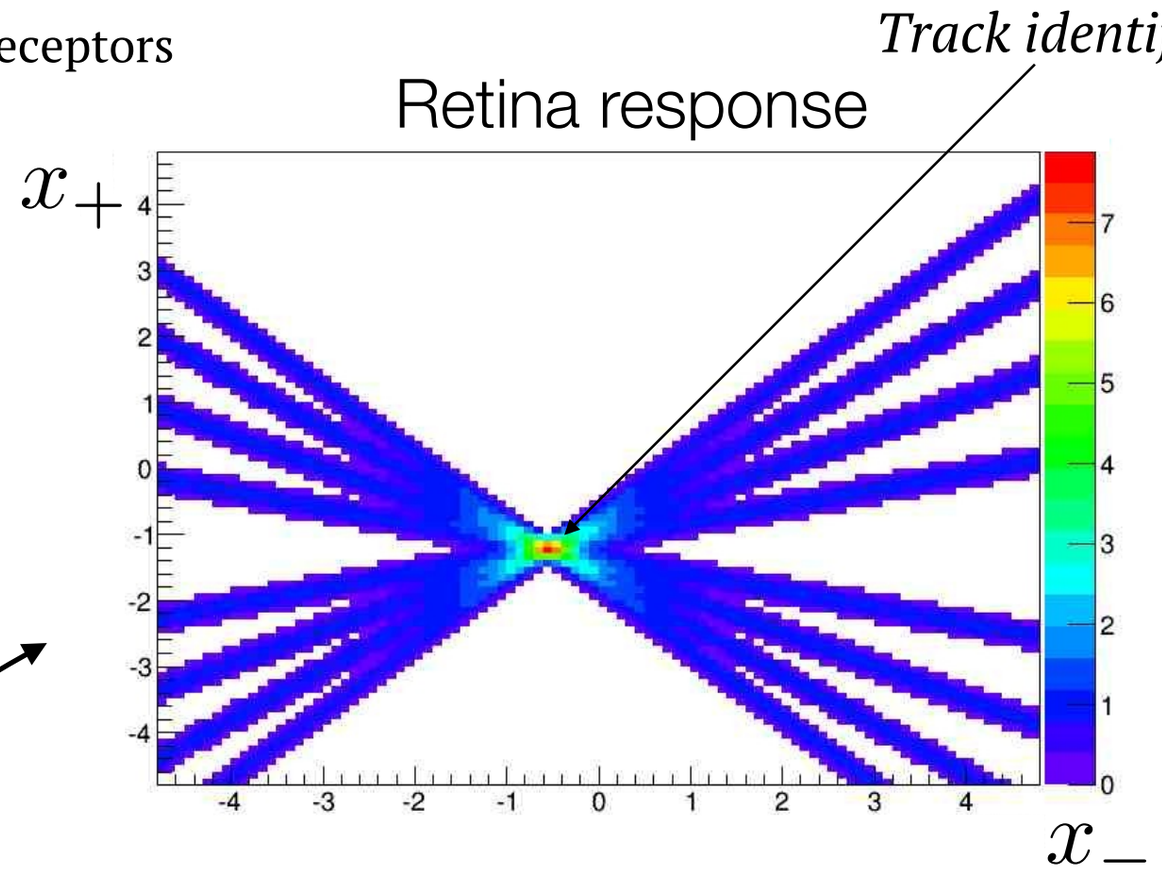
2D track: $x(z) = x_+ + x_- \dfrac{z - z_+}{z_-}$

Track parameters $x_\pm = \dfrac{x_l \pm x_f}{2}$

Excitation of the cellular units

$$R = \sum_i \exp\left(-\frac{s_i^2}{2\sigma^2}\right) \quad \text{if} \quad s_i < 2\sigma$$

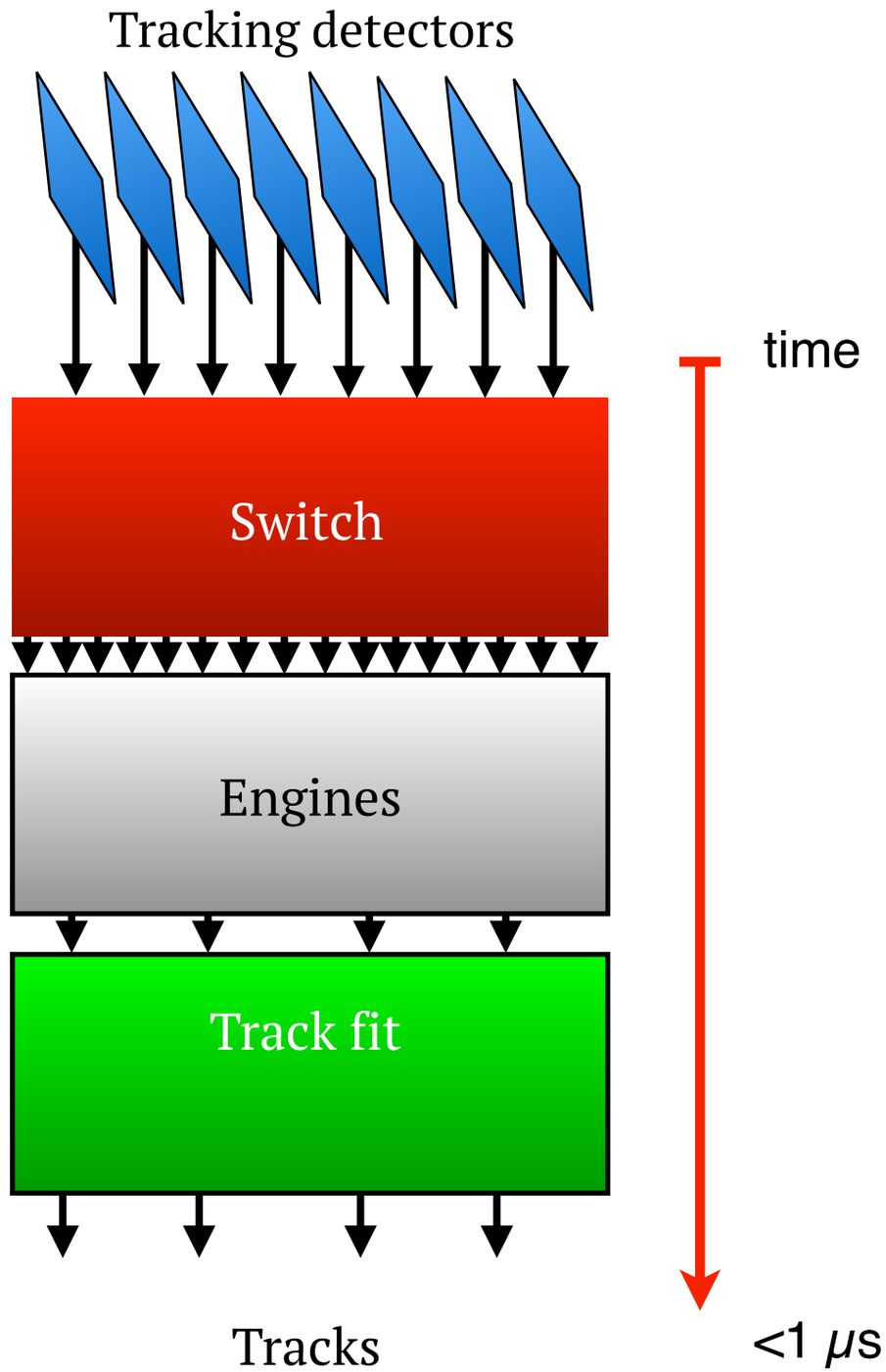
$$R = 0 \quad \text{if} \quad s_i > 2\sigma$$



- Track parameters obtained by interpolation of R values around maximum

Artificial retina architecture

- **Three main blocks:**
  - Switch: delivers in parallel the hits from the detectors to only appropriate cellular units
  - Engine: block of cellular units for parallel calculation of the weights
  - Track fit: interpolation of adjacent cell weights for track parameter determination
- **Main differences with AM approach:**
  - only relevant data reach the processing units (engines). Data processing starts already in the switch while data is transmitted
  - retina algorithm provides analog response contrarily to AM “yes/no” pattern matching



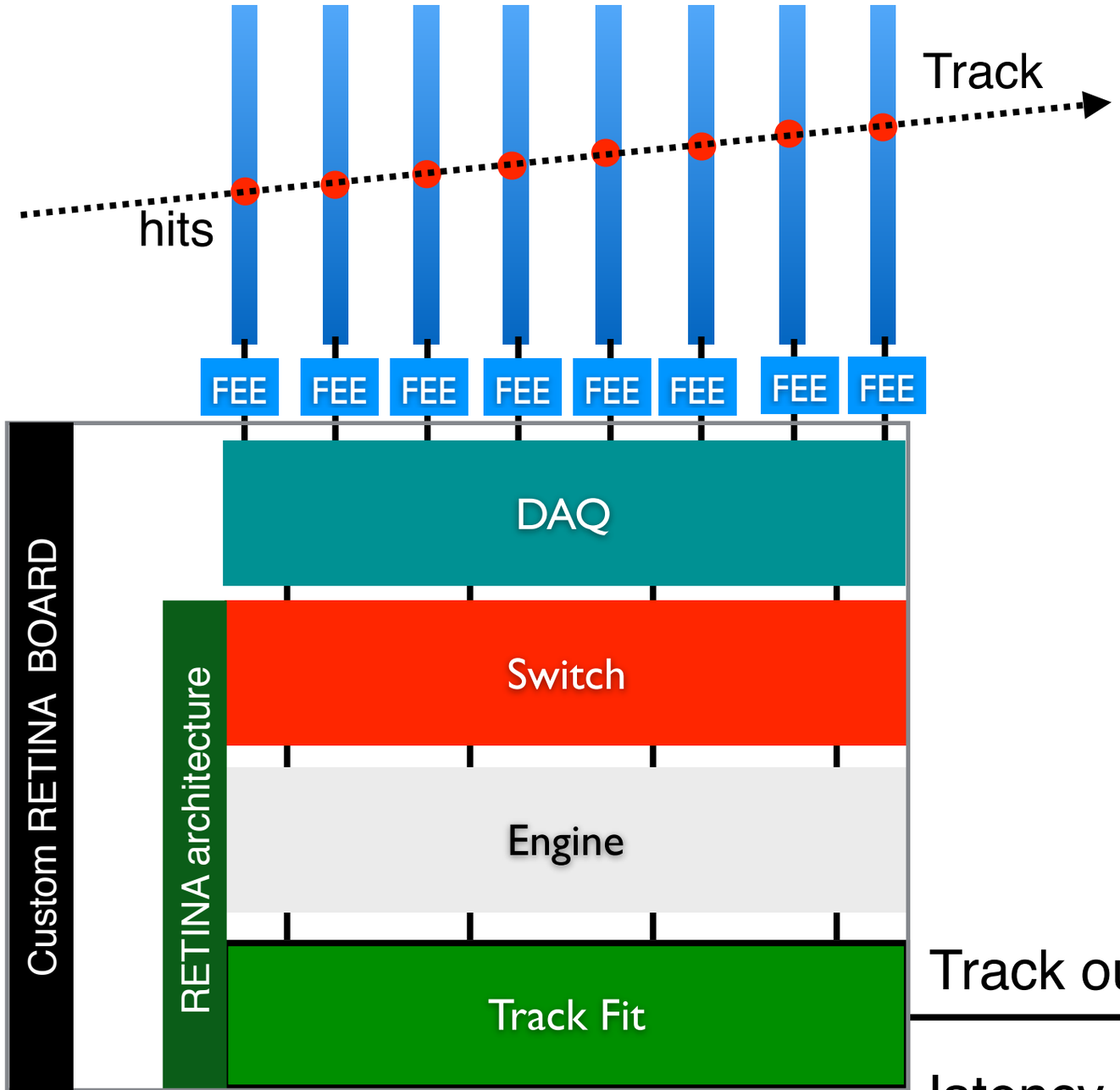
Retina INFN project

- INFN-Retina R&D project started in 2015. Milano and Pisa groups involved
- Develop hardware prototype of a real time tracking device for intensive tracking applications (1-100 Giga tracks/sec), *e.g.* HL-LHC experiments
- Main deliverables:

  - Real time tracking detector prototype for test beam (main subject of this talk)
  - Fast track finding system compatible with large DAQ framework for test with simulated data at 40 MHz event rate (next step)

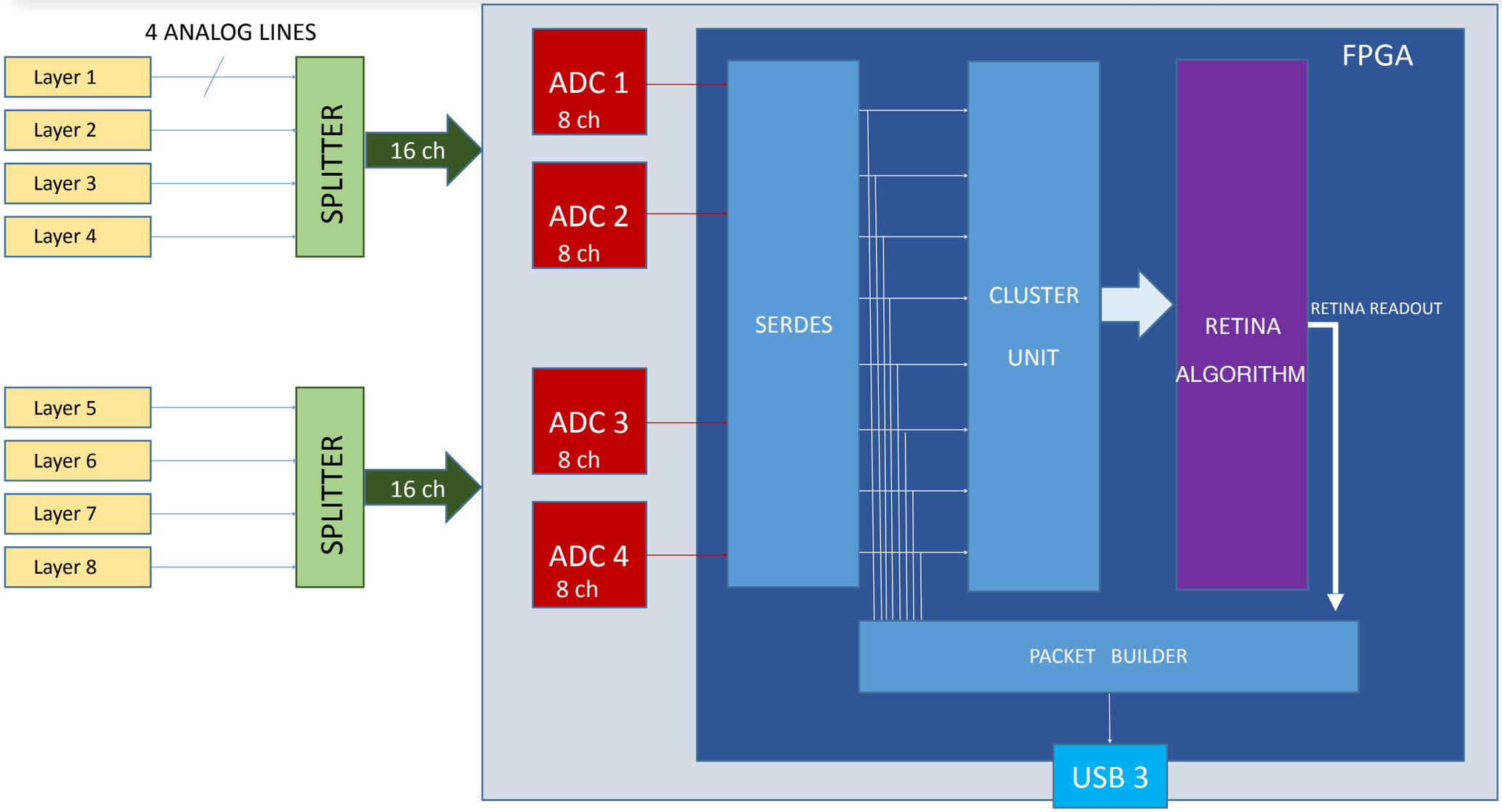
Detector prototype with embedded tracking capabilities

Real time tracking prototype



- Practical demonstrator: 8-layer tracking prototype. Single-sided silicon strip detectors, 183 µm pitch
- Custom DAQ board: Retina architecture implemented in last generation FPGA
- Test full tracking system chain using 180 GeV/c proton beam at CERN SPS
- Device response reproduced with high level and low level simulations and studied using data

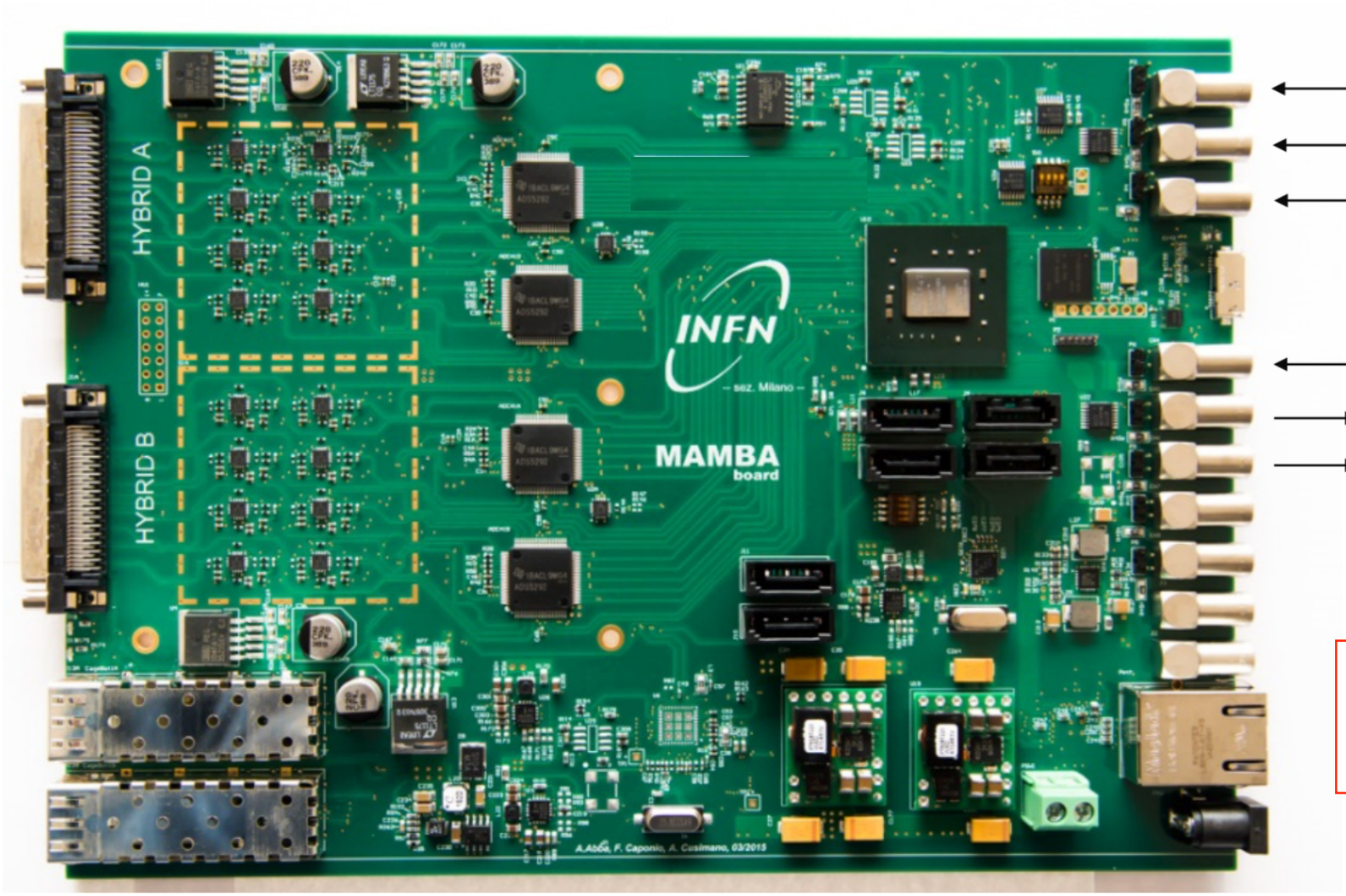
Data acquisition system



- 4 Beetle chips for each detector
- Readout rate 300 kHz (1x mode). [Max rate 1.1 MHz (4x mode)]
- Digitalisation with multichannel 12-bit ADC and zero suppression (threshold comparator)
- Data output to disk using fast USB3 port

Mamba board

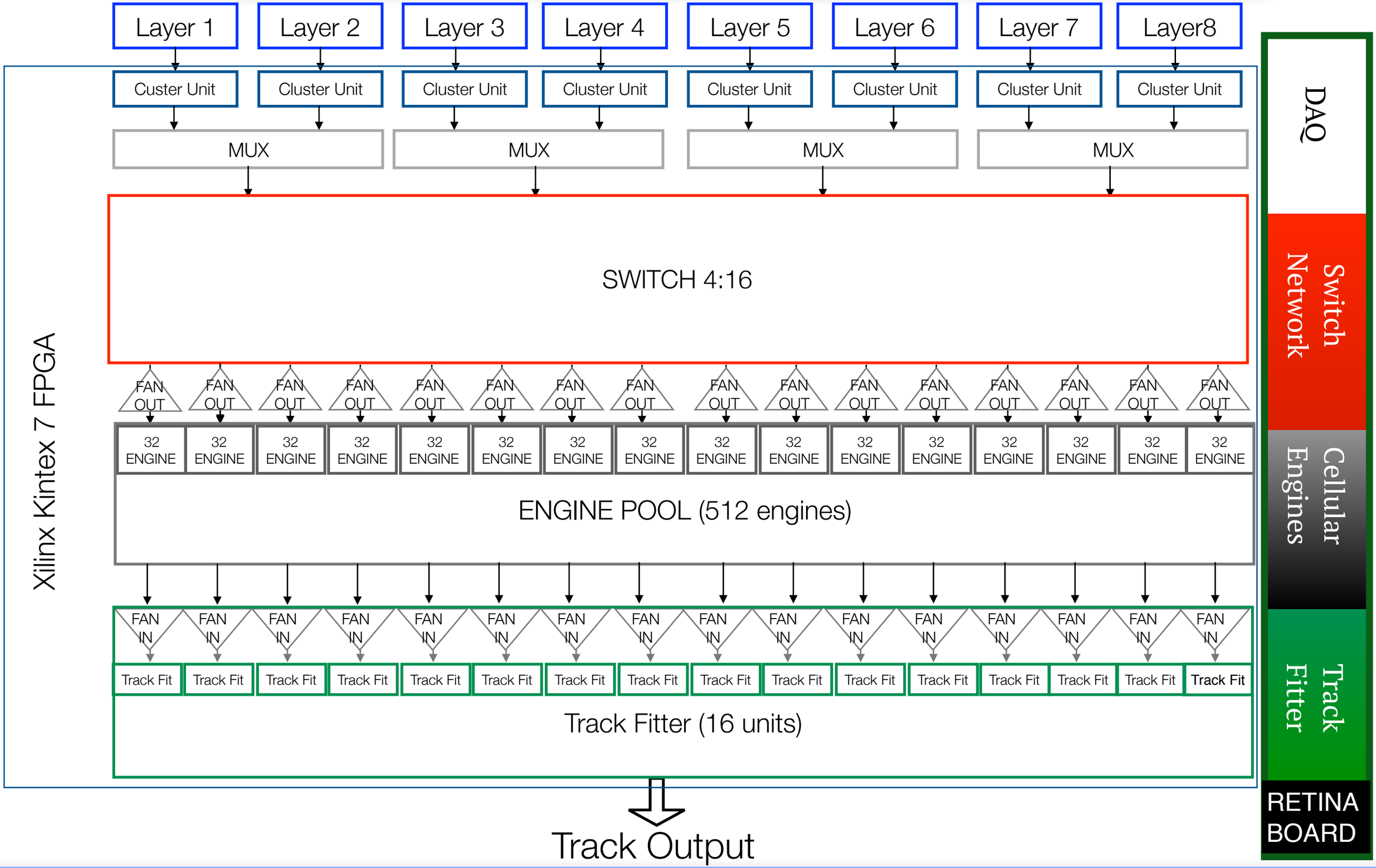
- Readout 8 detectors in 1x mode (300 kHz) or 2 detectors in 4x mode (1.1 MHz)



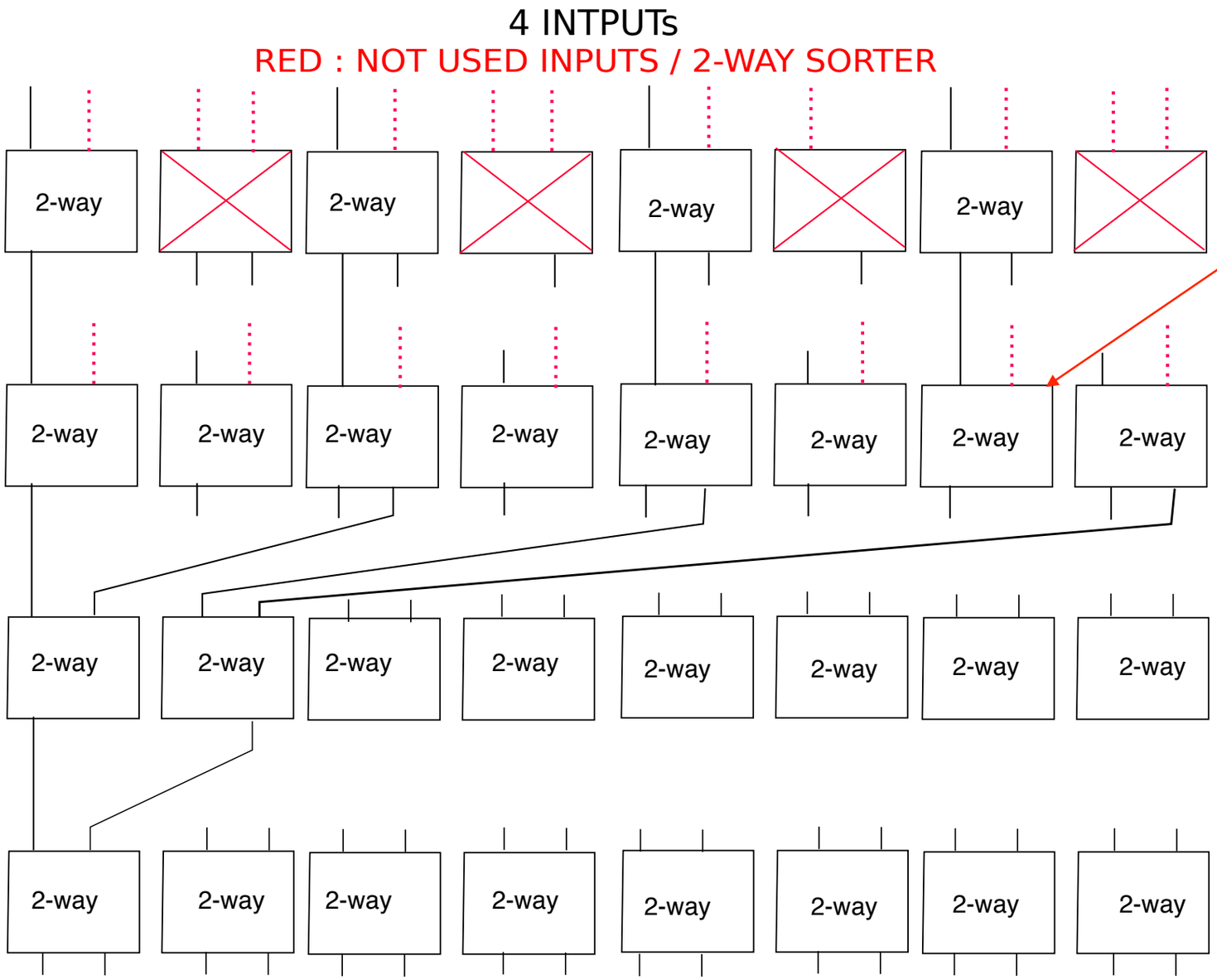
← SCINTILLATOR 1 / COINCIDENCE
← SCINTILLATOR 2
← TELESCOPE CLOCK
← TELESCOPE RESET
→ ACCEPTED TRIGGER
→ BUSY

Mamba board
by Nuclear Instruments

Artificial retina architecture

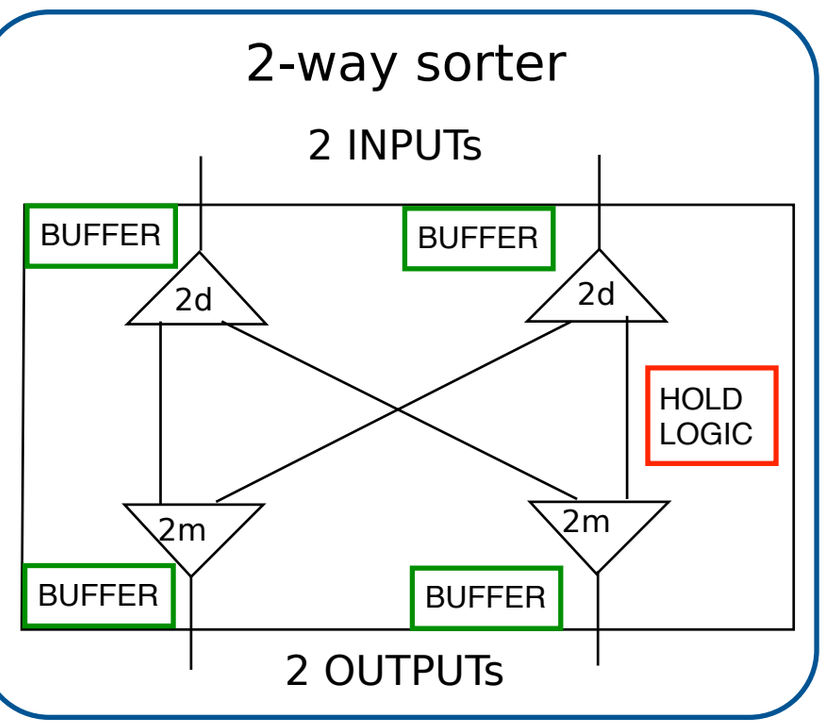


Switch network: a 4x16 way dispatcher

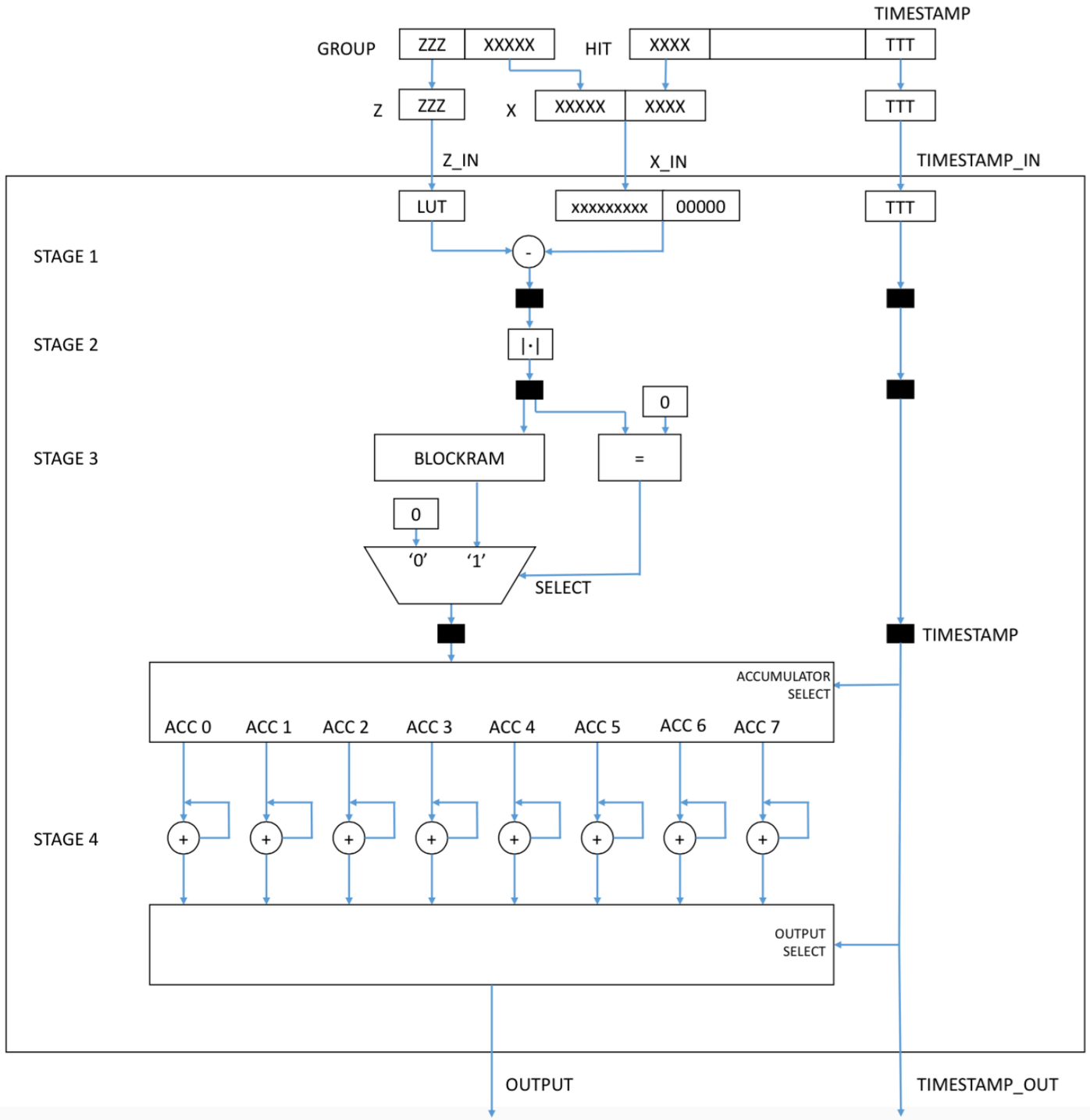


32 engines each output = 32x16=512 engines

- Each box is a programmable two way sorter
- Each input can be delivered to the left, to the right or both output ports according to LUT information
- The 2-way sorter acts also as a memory buffer in case of traffic jam. Input stream can be held



Cellular engine



- Clocked pipeline divided in 4 stages:

  1) s_i = hit-receptor distance

  2) $|s_i|$ absolute value

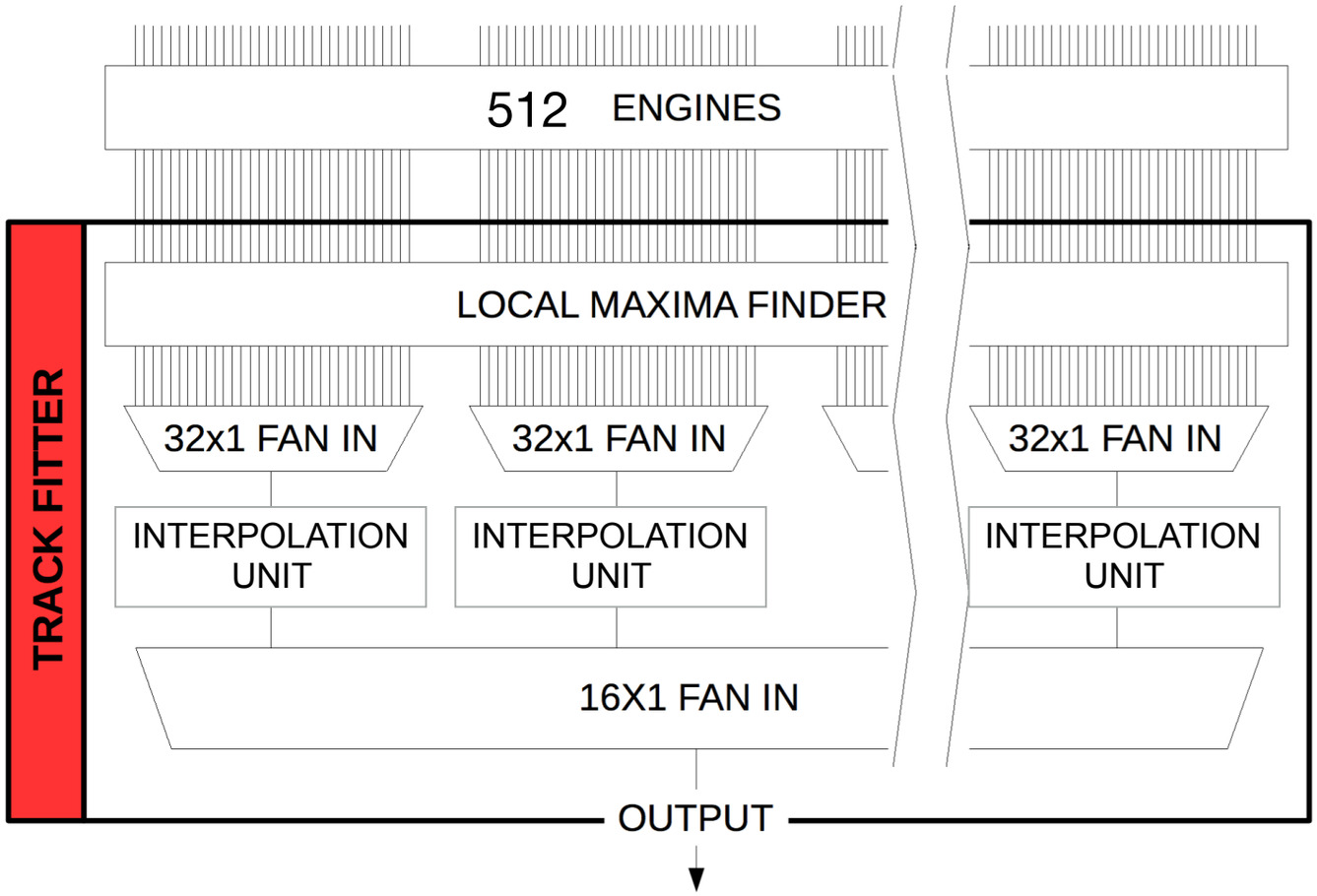
  3) $\exp\left(-s_i^2/2\sigma^2\right)$ tabulated in 10 bit (in) x 16 bit (out) LUT

  4) $R = \sum_i R_i$ sum of weights

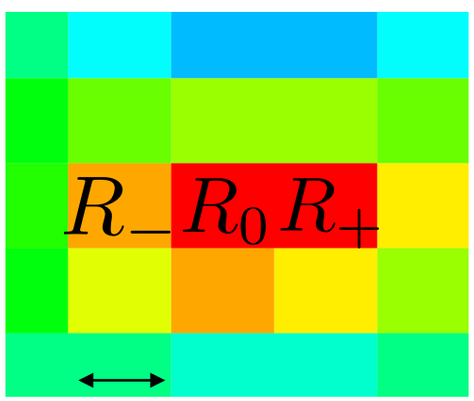
- 8 accumulators for hits arriving at different times (useful at hight event rates)

- Engines output values after EndEvent signal has arrived to accumulator

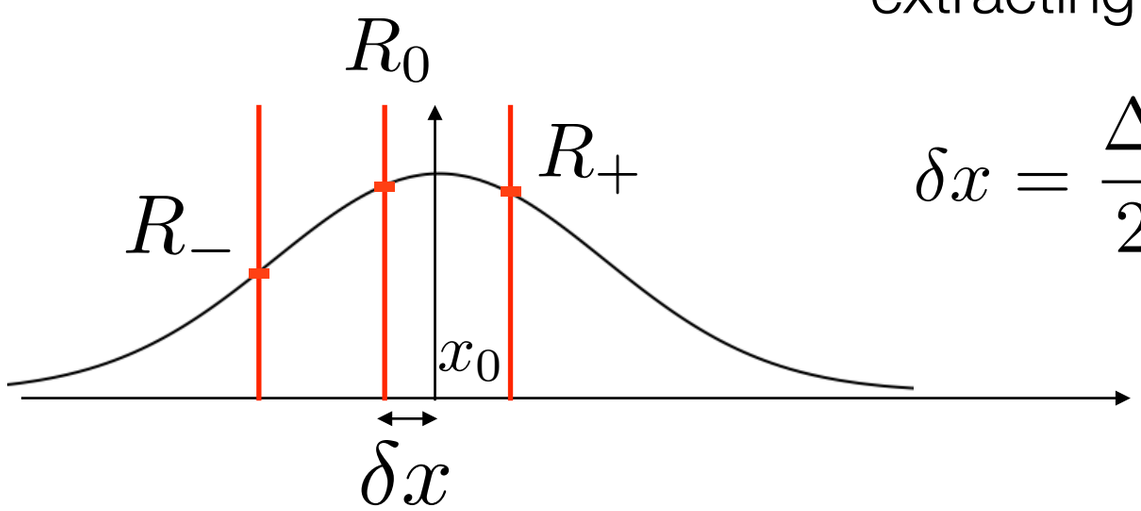
Track Fitter



- Identify in parallel local maximum weight above threshold and send relevant data to interpolation unit
- Data bandwidth now reduced by at least a factor 8
- Fan In: 32 engines connected to 1 interpolation unit
- Parabolic interpolation for extracting track parameters



Δ = grid step

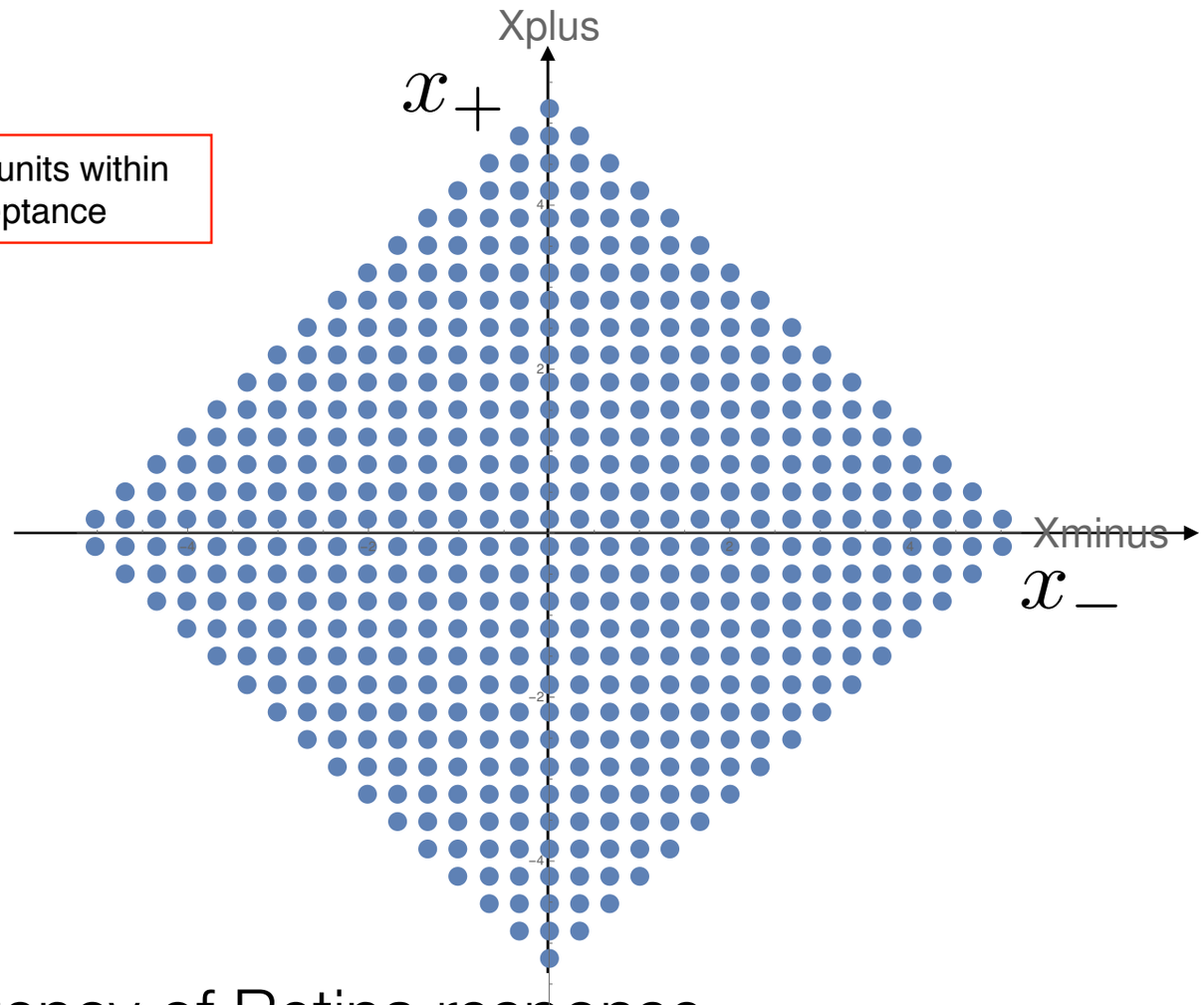


$$\delta x = \frac{\Delta}{2} \frac{R_+ - R_-}{2R_0 - R_- - R_+}$$

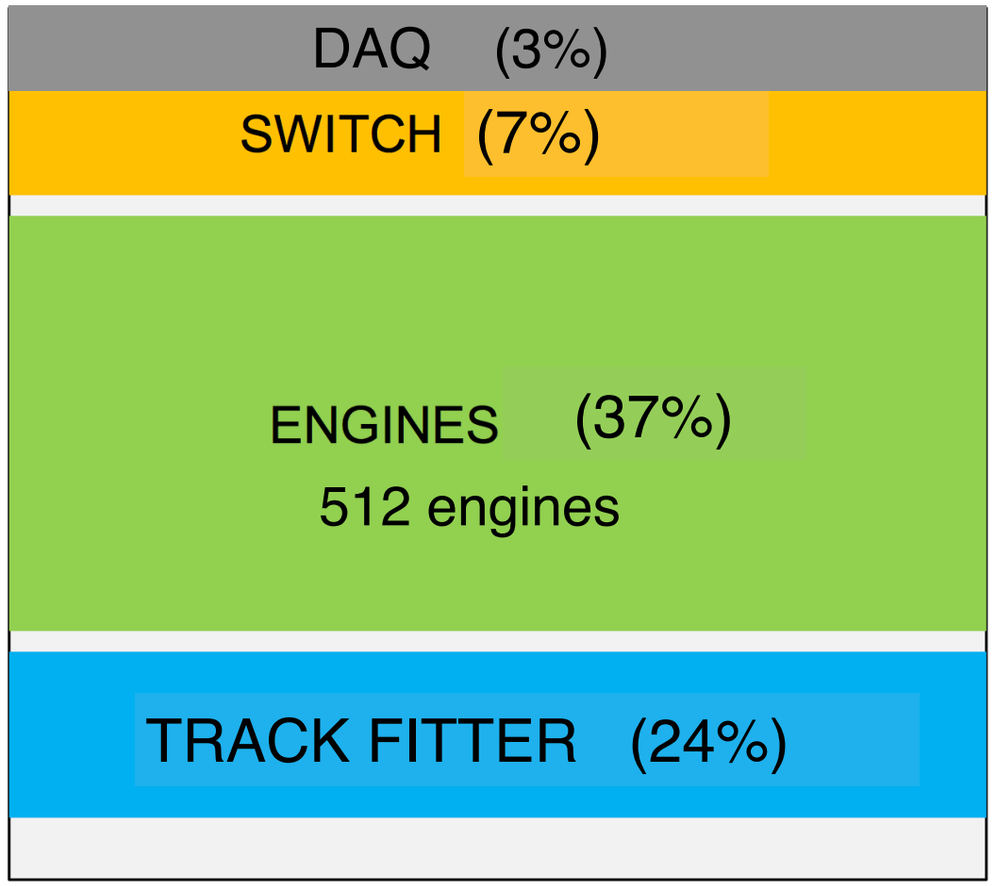
Resources and latency

- Here we present a solution based on Xilinx Kintex7 XC7K410T FPGAs

grid of cellular units within telescope acceptance



FPGA occupation



29% BACKUP

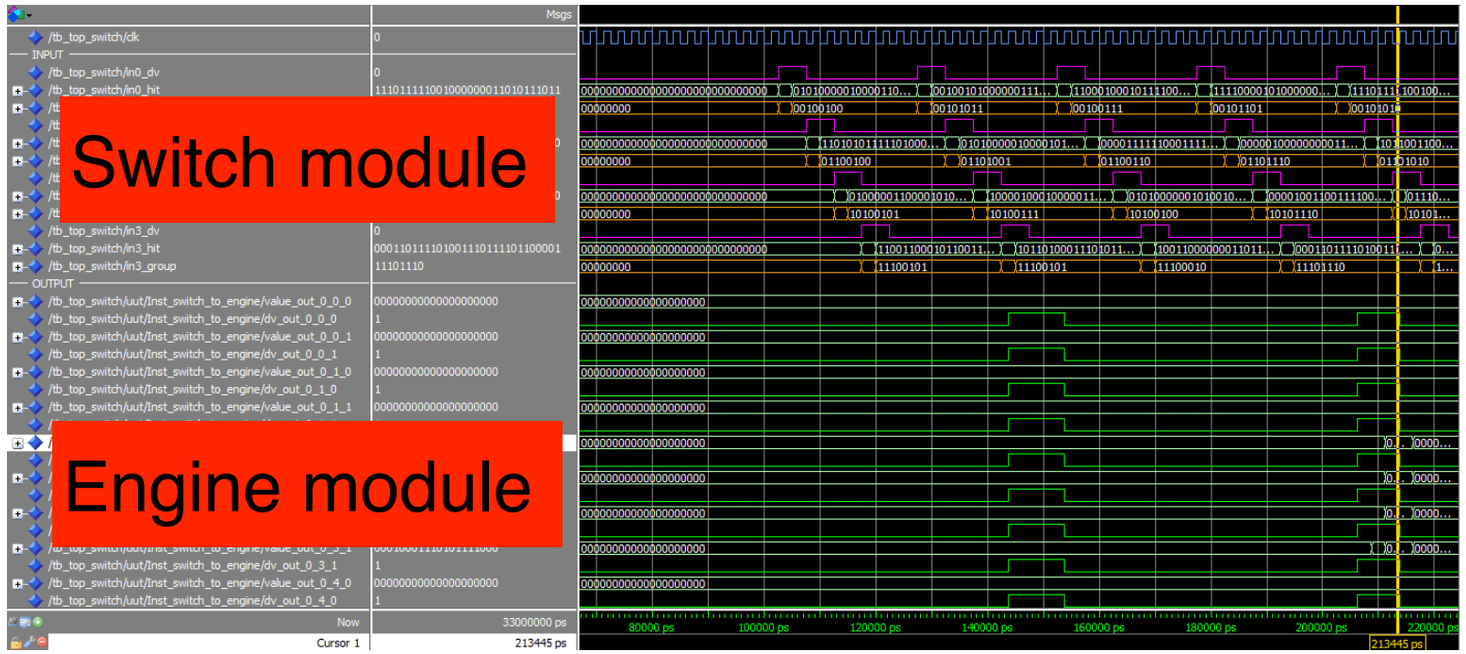
Latency of Retina response

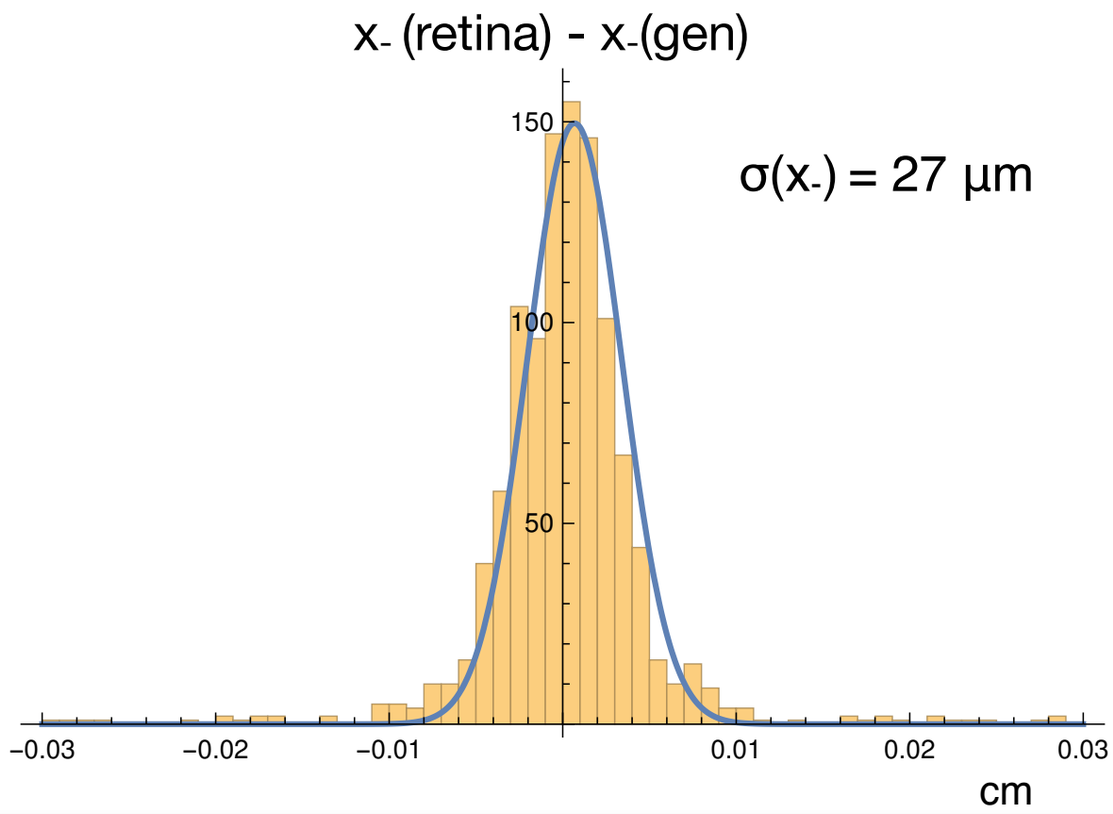
TrackFitter latency can be reduced to 10 t.u. using more logic resources

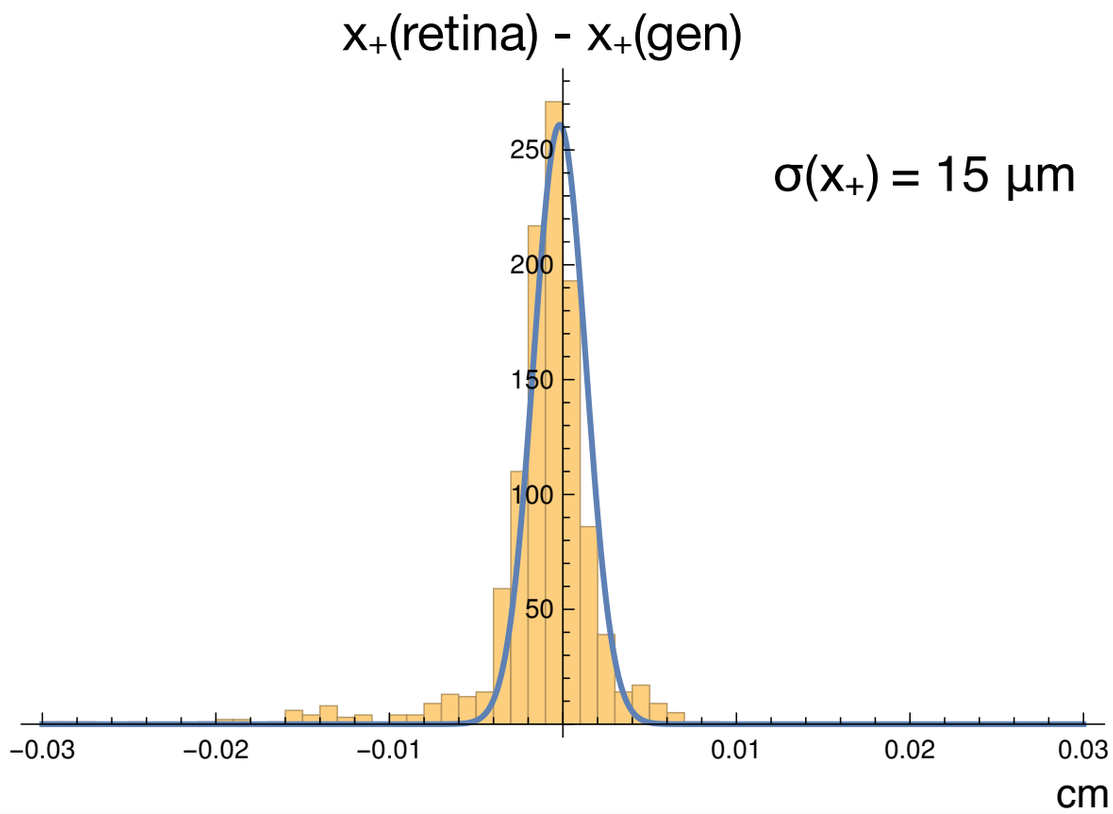
1 t.u. at 200 MHz clock = 5 ns

Simulation results

- Response simulated with ModelSim using single-track events
- Residual distribution of x_-, x_+ track parameters: retina - generated tracks





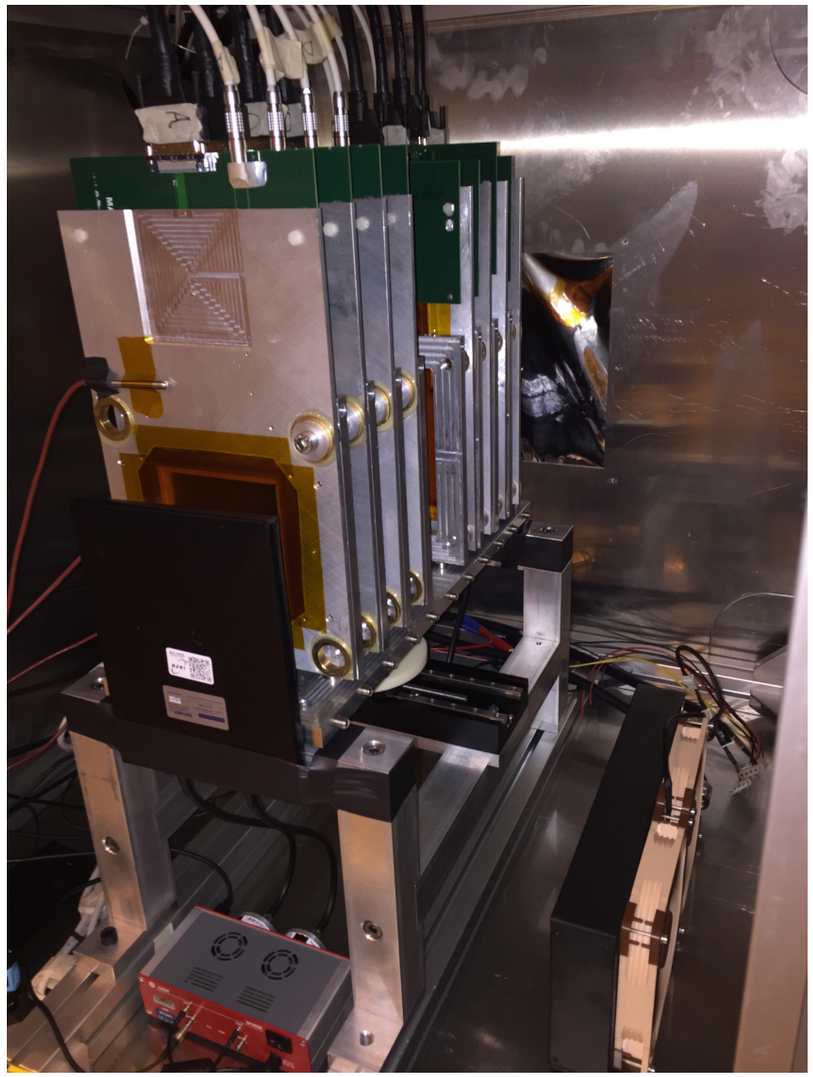


Testbeam results

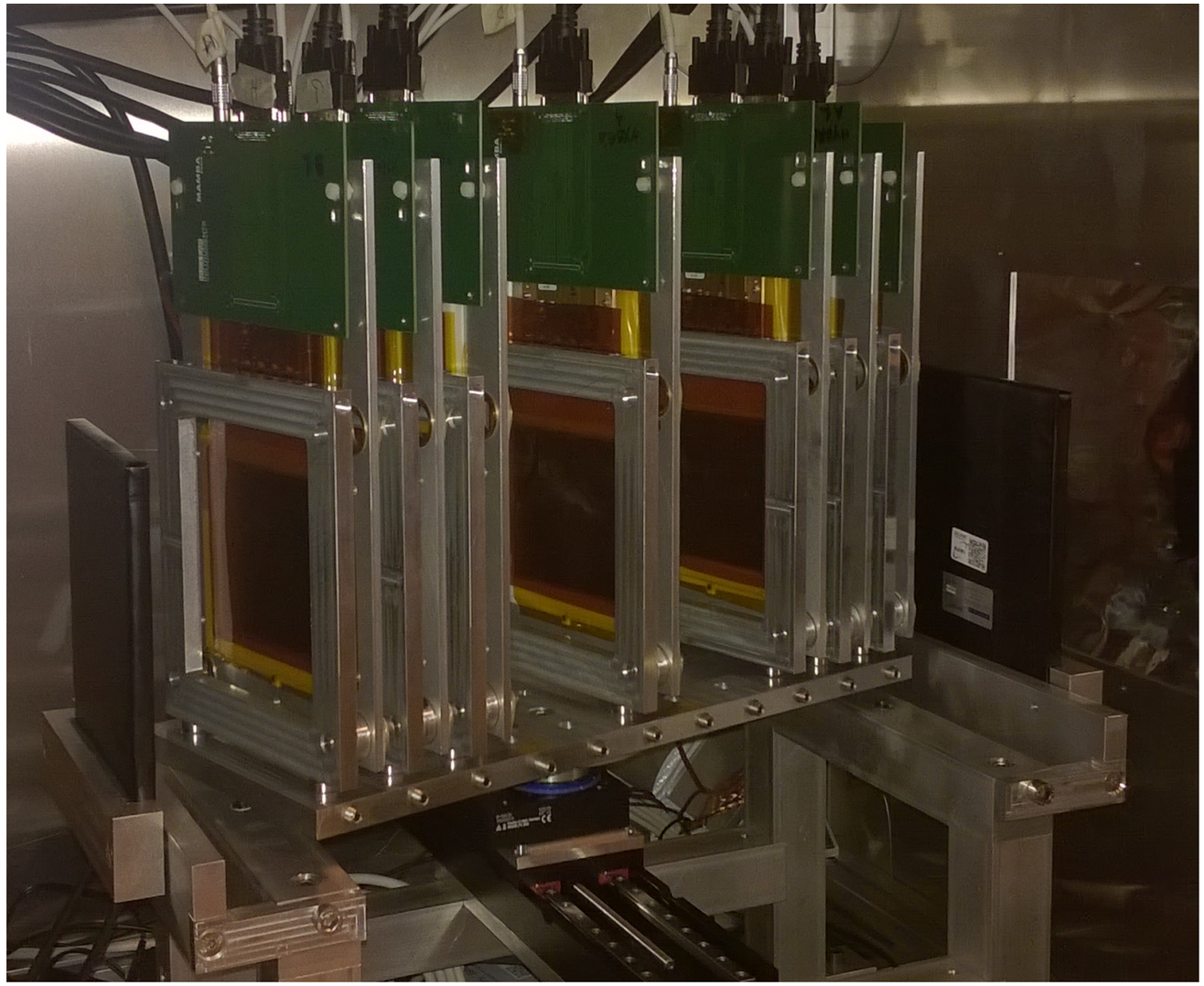
Telescope on beam at SPS

- Telescope tested on 180 GeV/c proton beam
- Rotation angle wrt beam axis: 0, 2, 4, 8, 16, 20 degree

Telescope aligned with beam axis

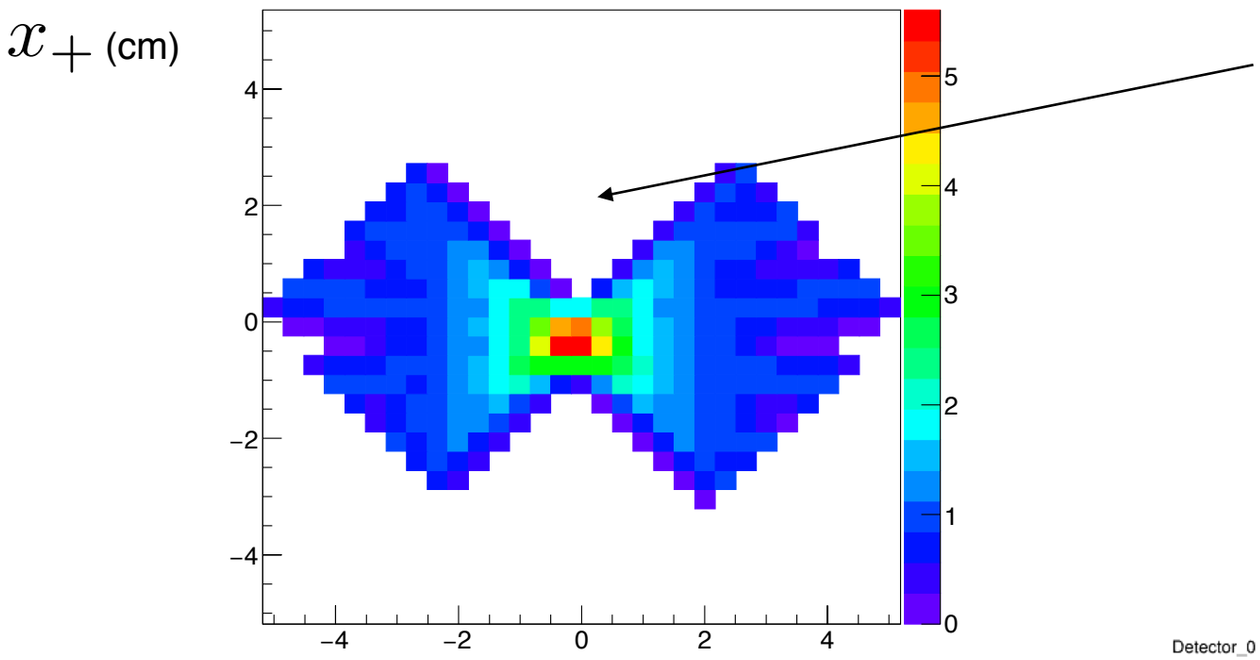


Telescope rotated wrt beam axis



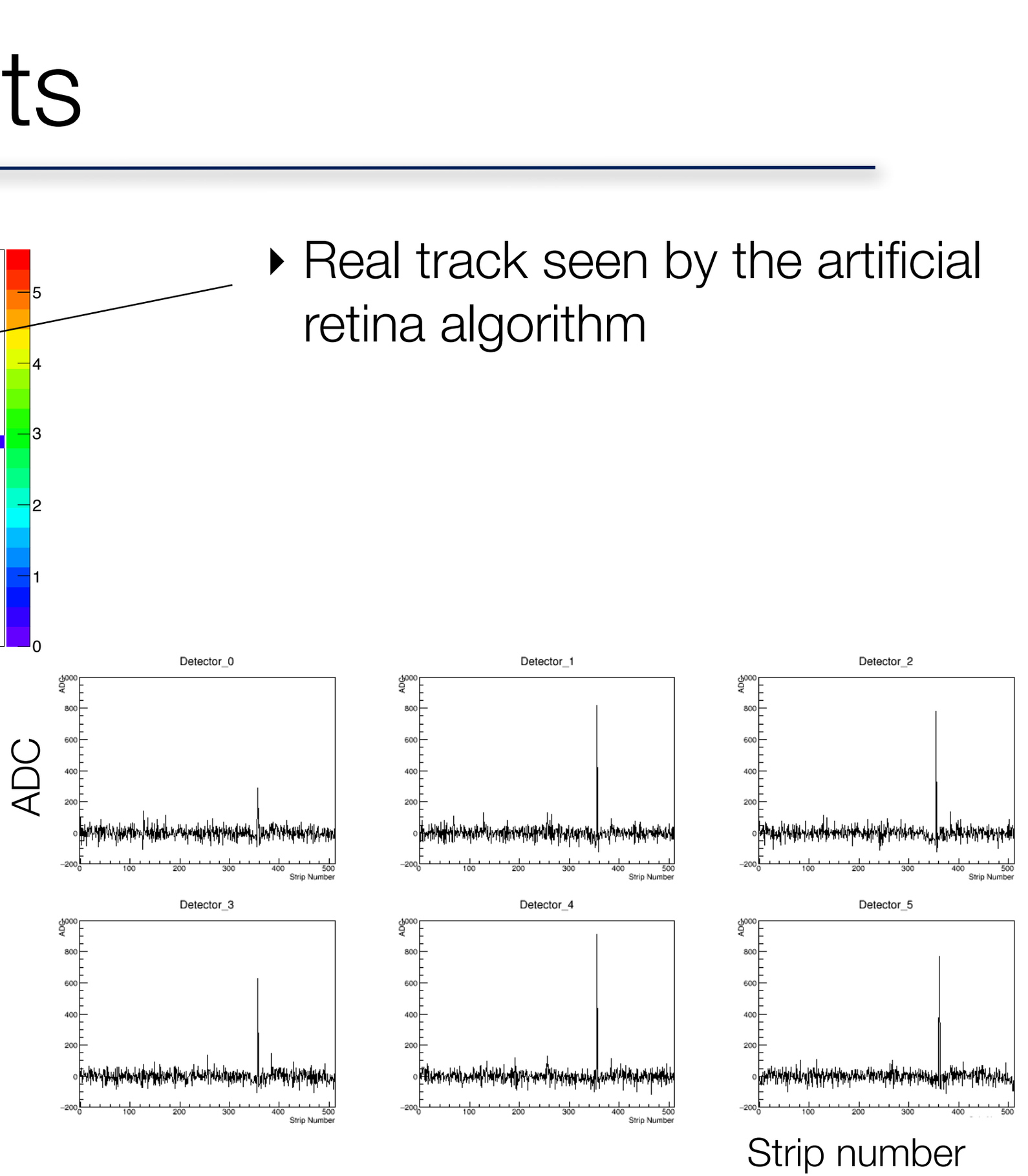
Data results

Retina response



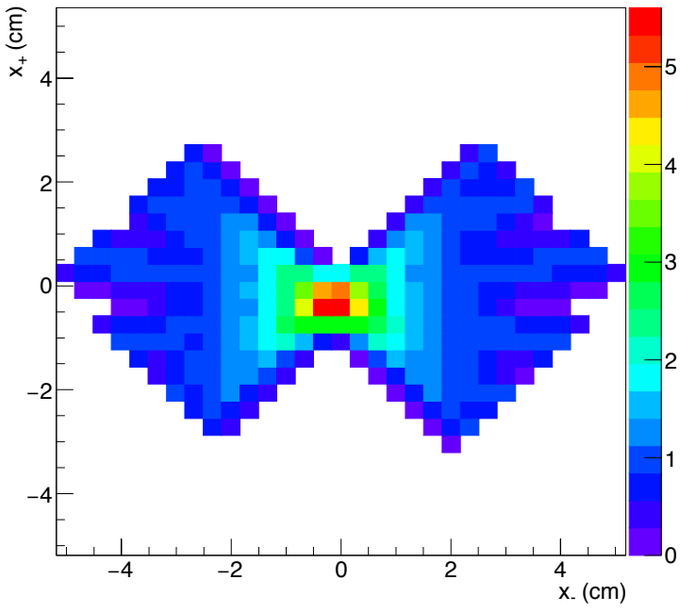
- Real track seen by the artificial retina algorithm

- Associated track hits on telescope detectors (used 6 layers in this test)

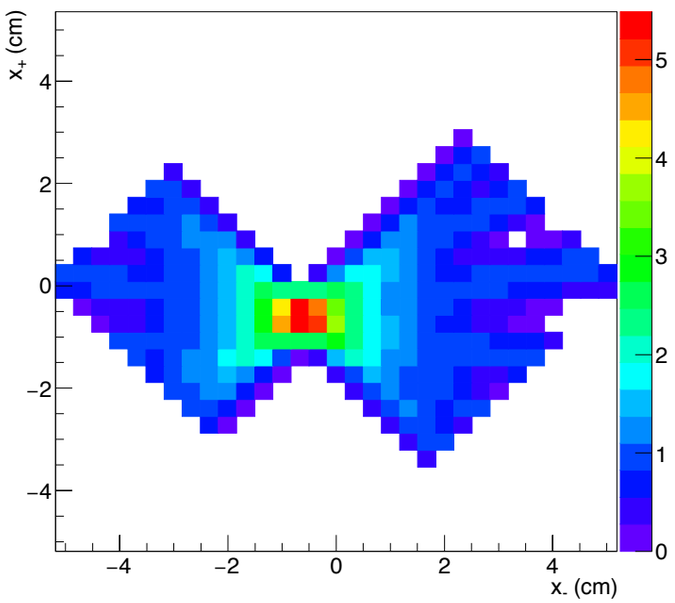


Retina response vs track angle

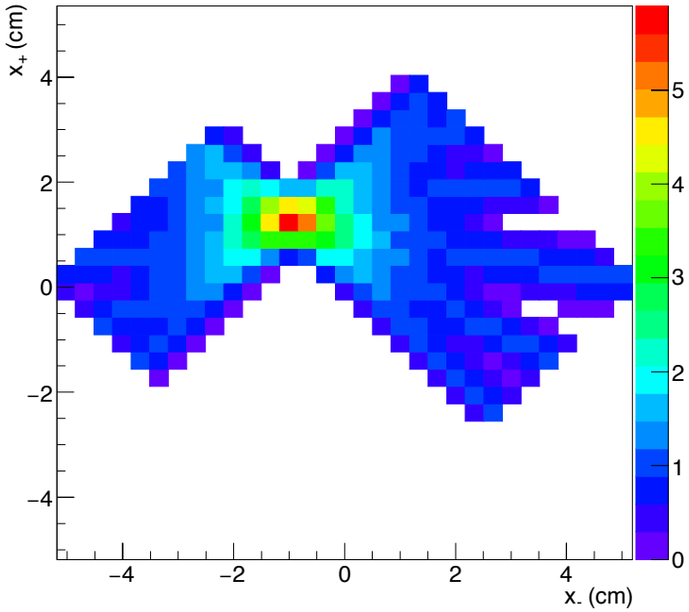
Track angle: 0 degree



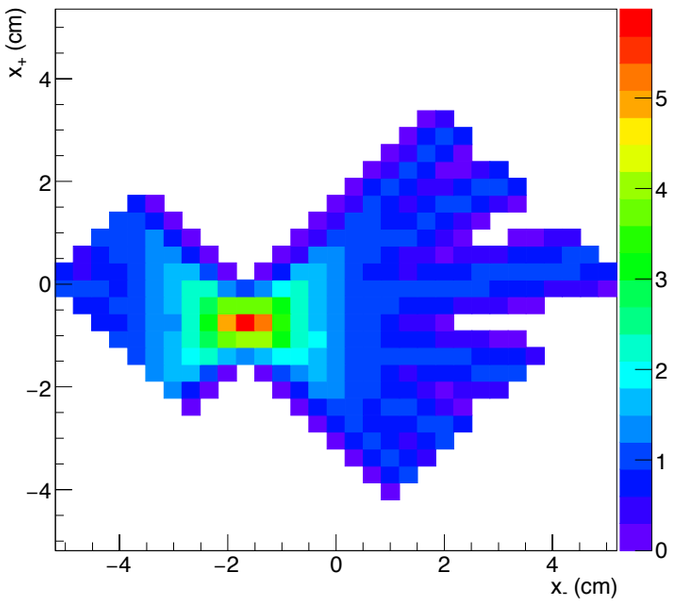
Track angle: 2 degree



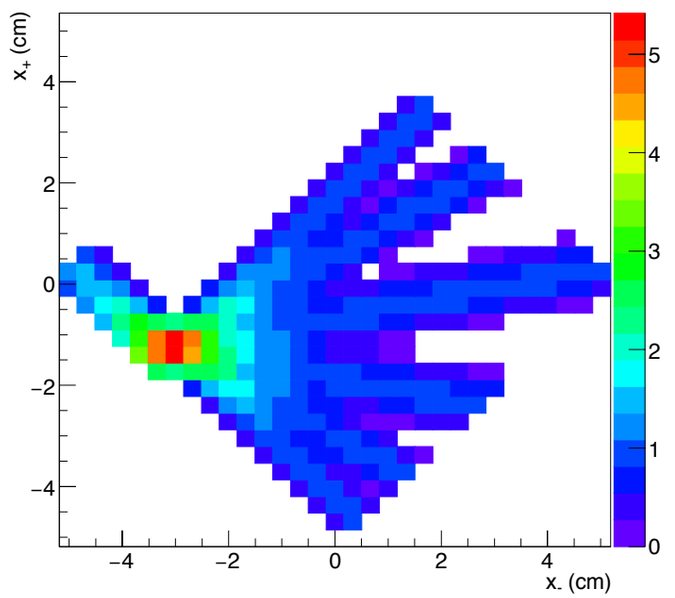
Track angle: 4 degree



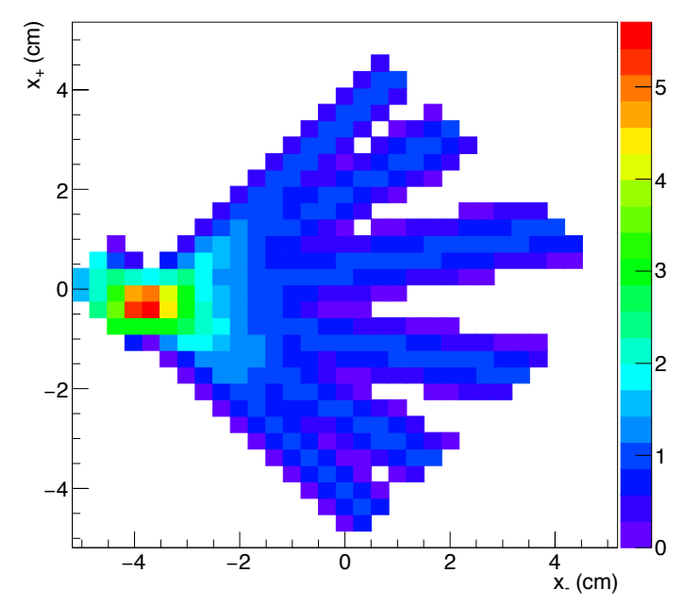
Track angle: 8 degree



Track angle: 16 degree

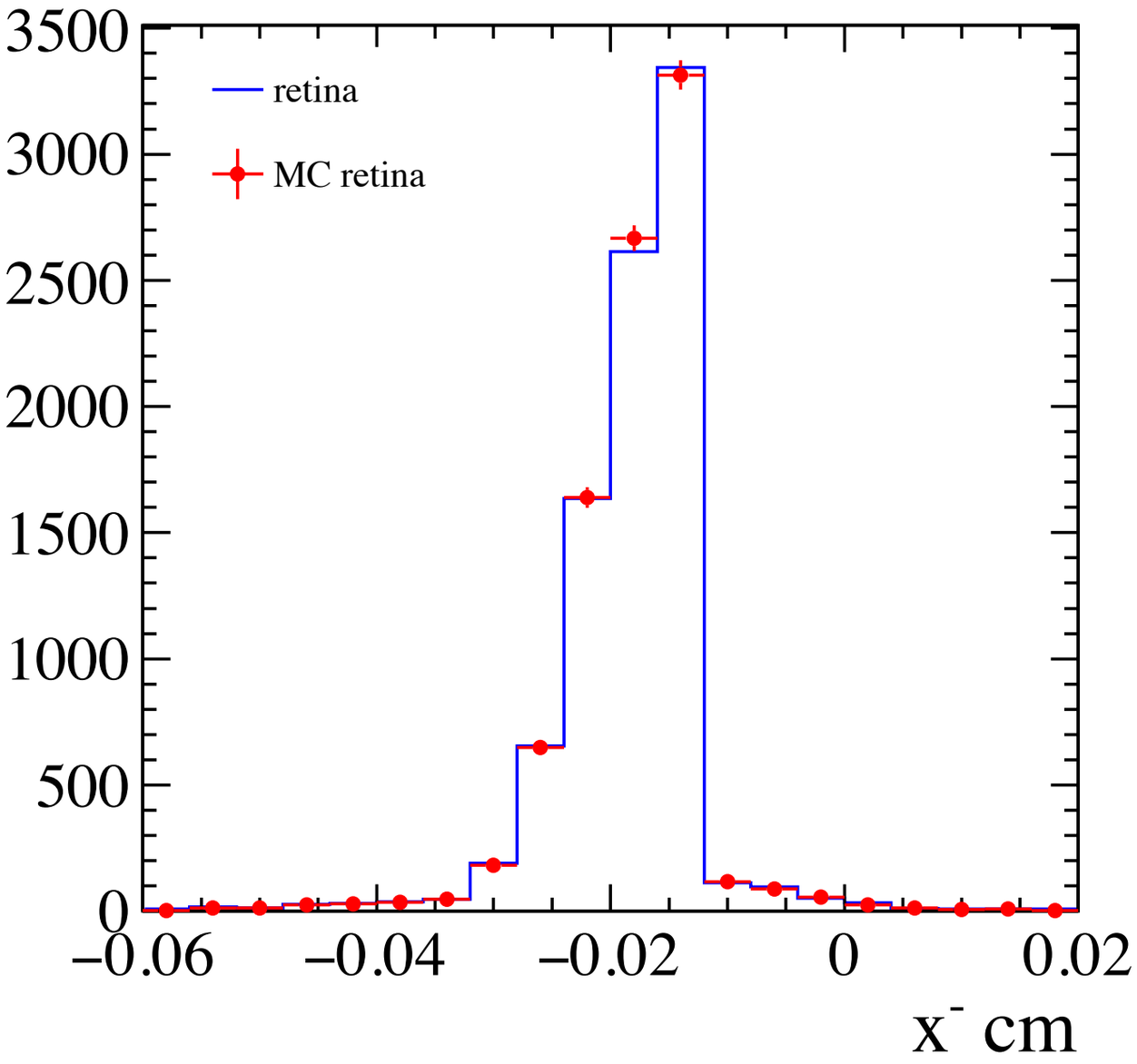


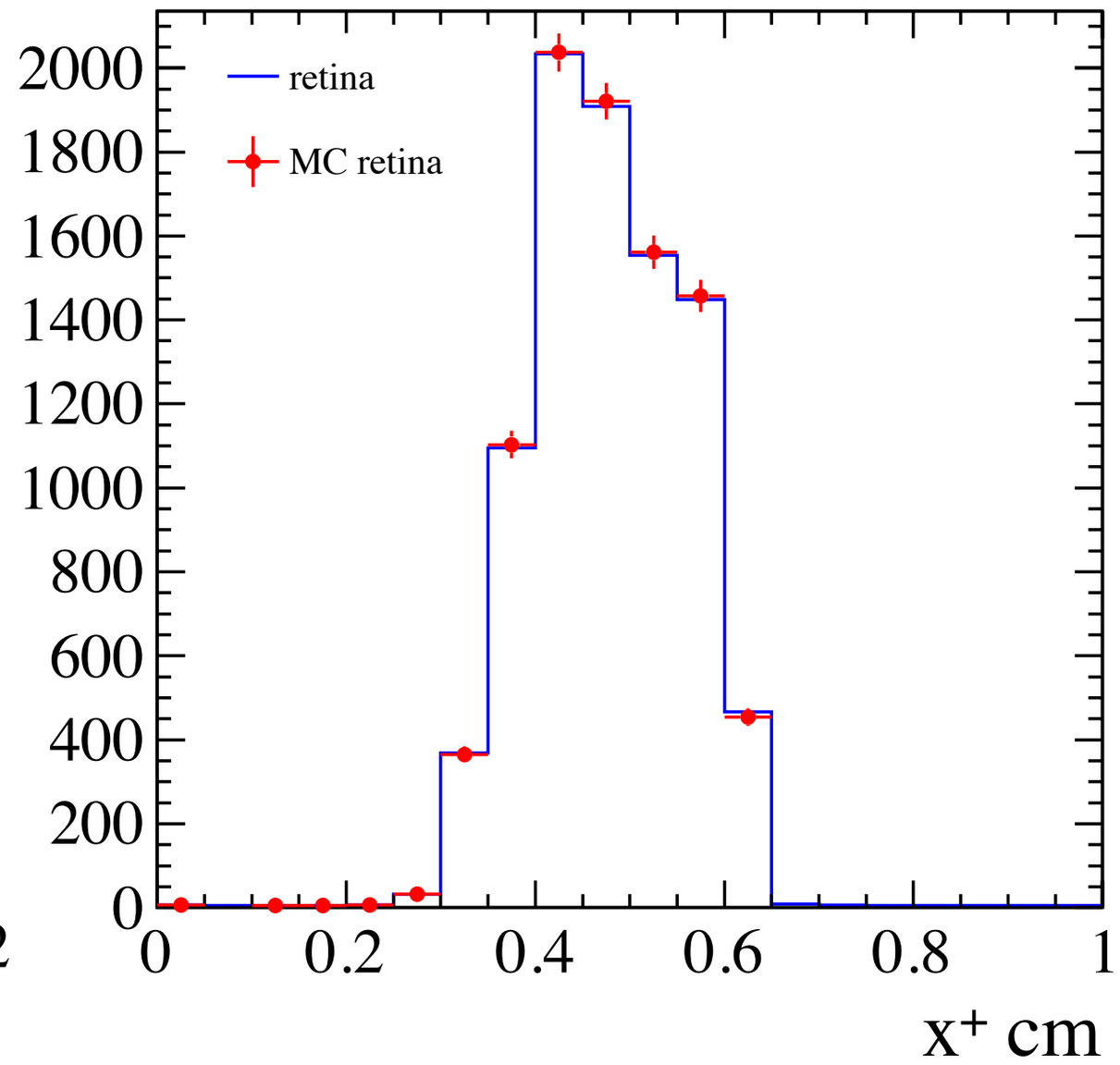
Track angle: 20 degree



Data/simulation comparison

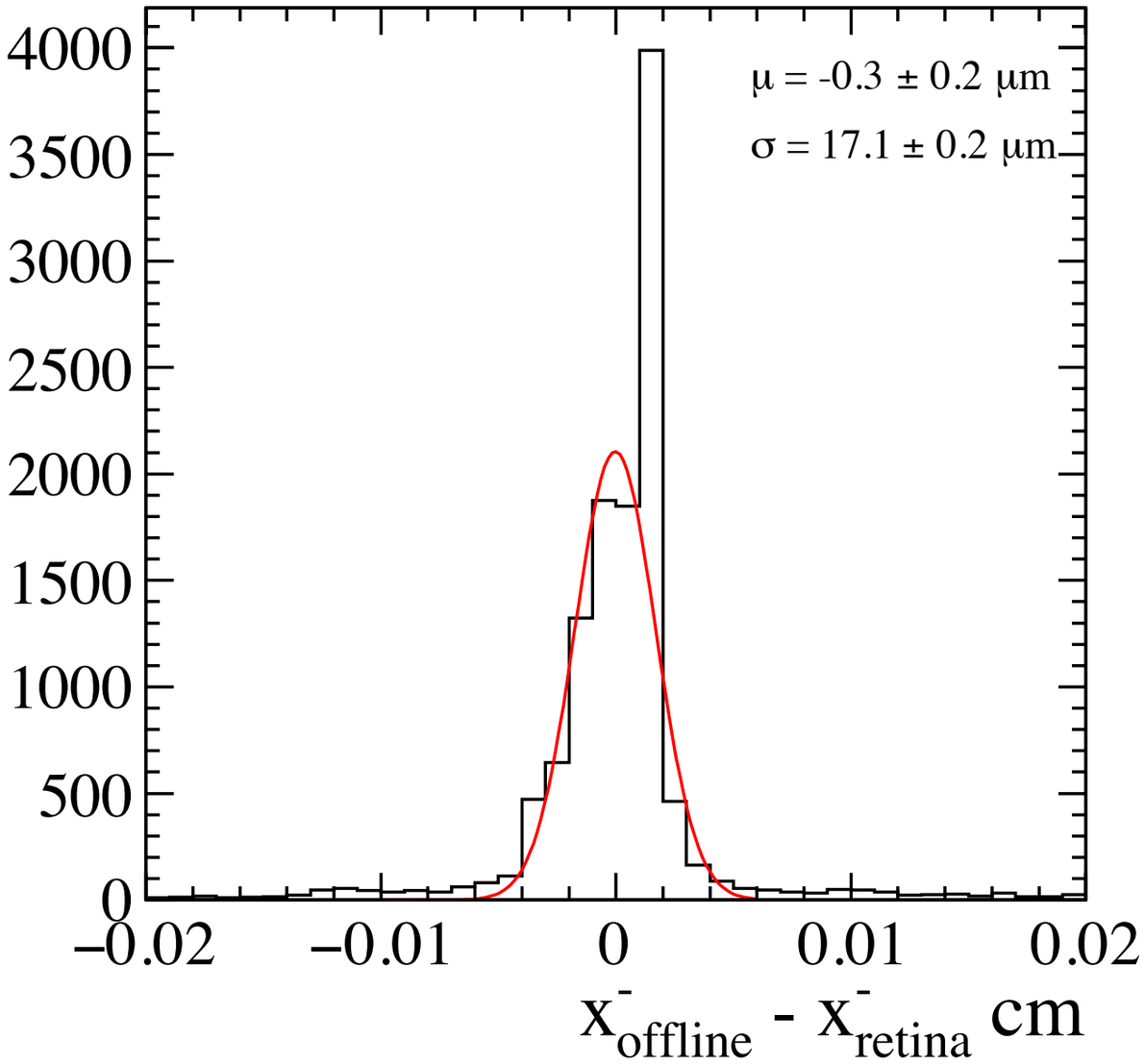
- Track parameter distribution determined by artificial retina algorithm
  - testbeam data processed by mamba board (retina) and verified using the artificial retina simulated response (MC retina)

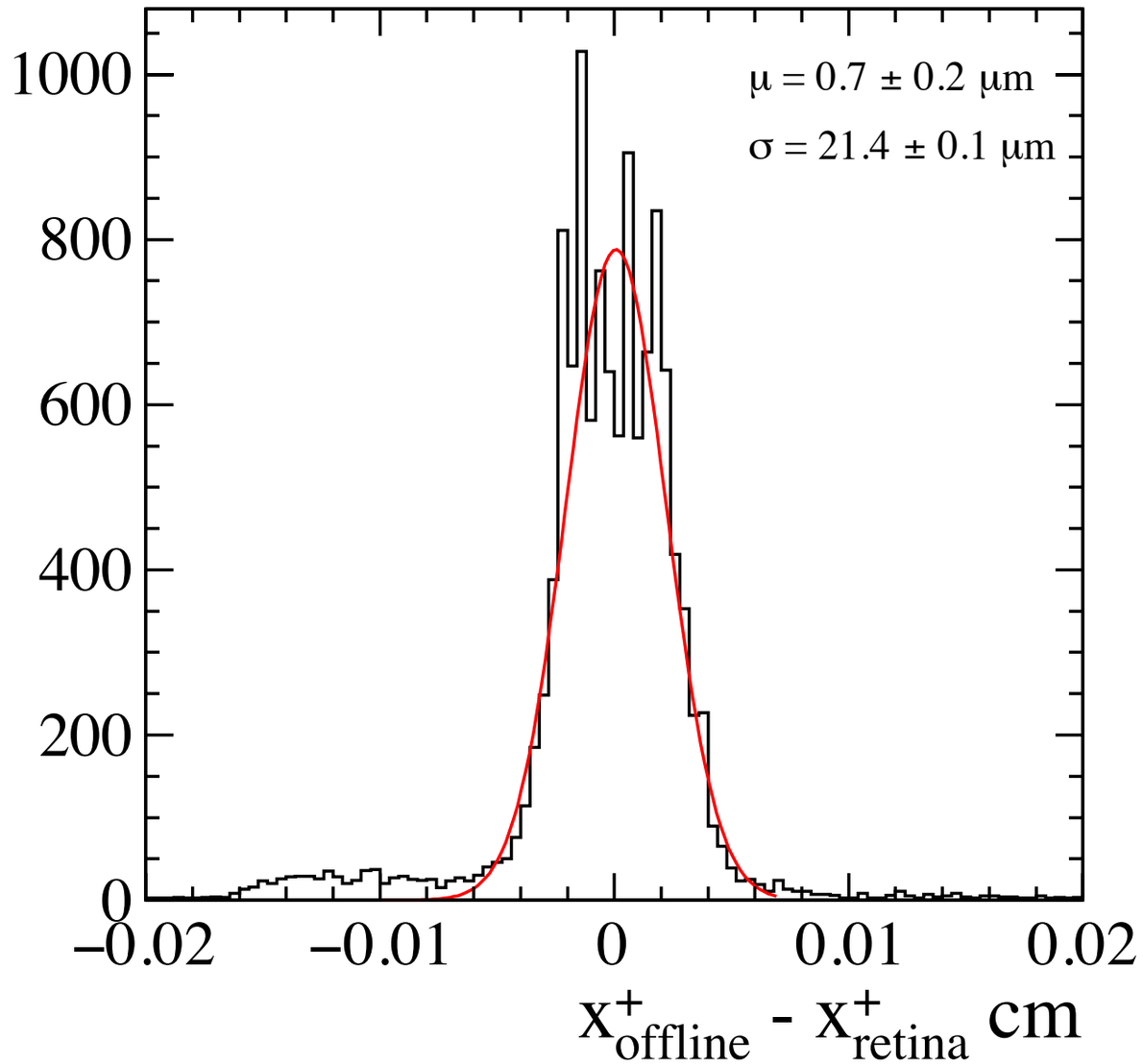




Track residuals: offline - retina

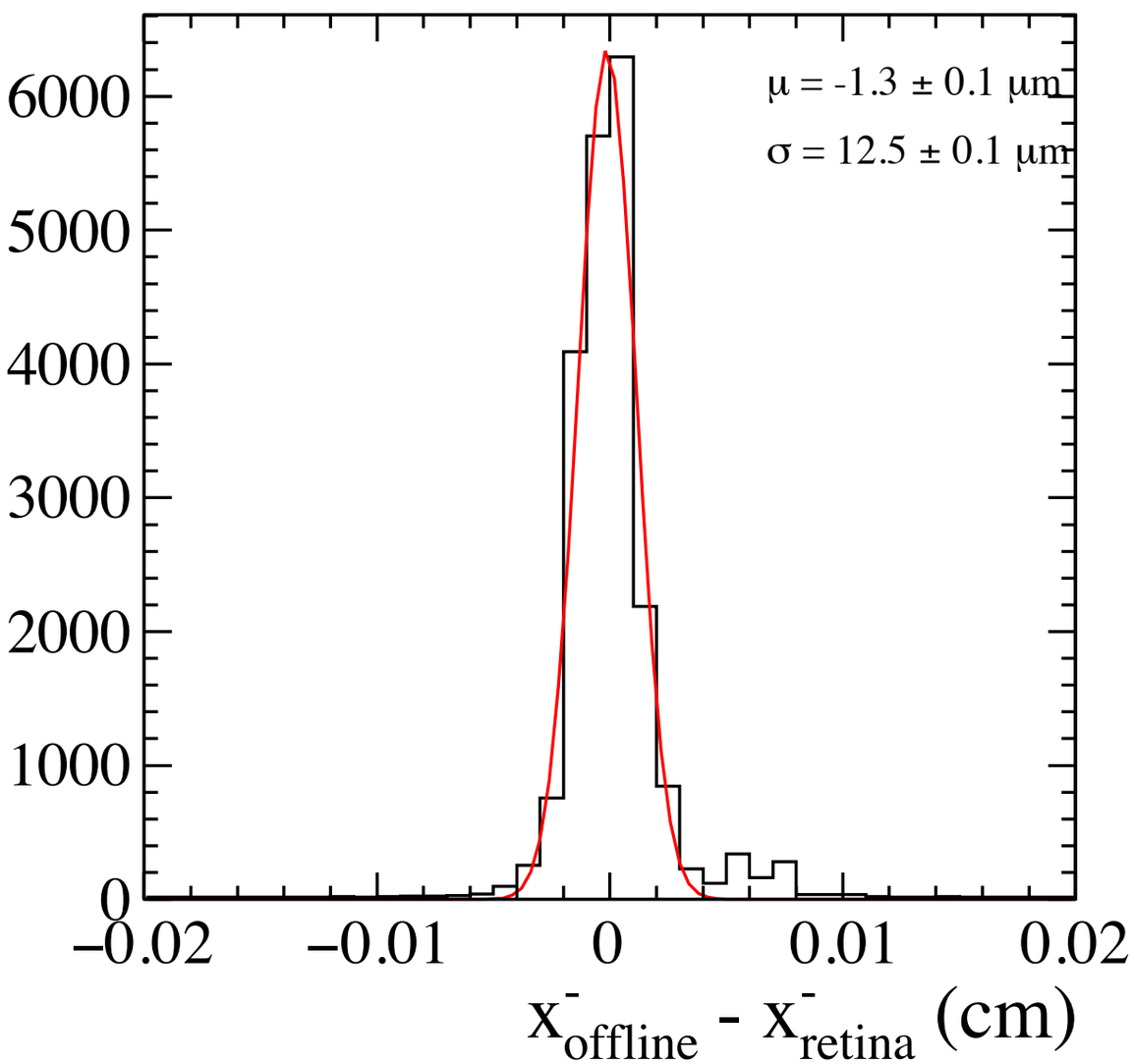
- It works! Offline-Retina track parameter residual are peaked at zero

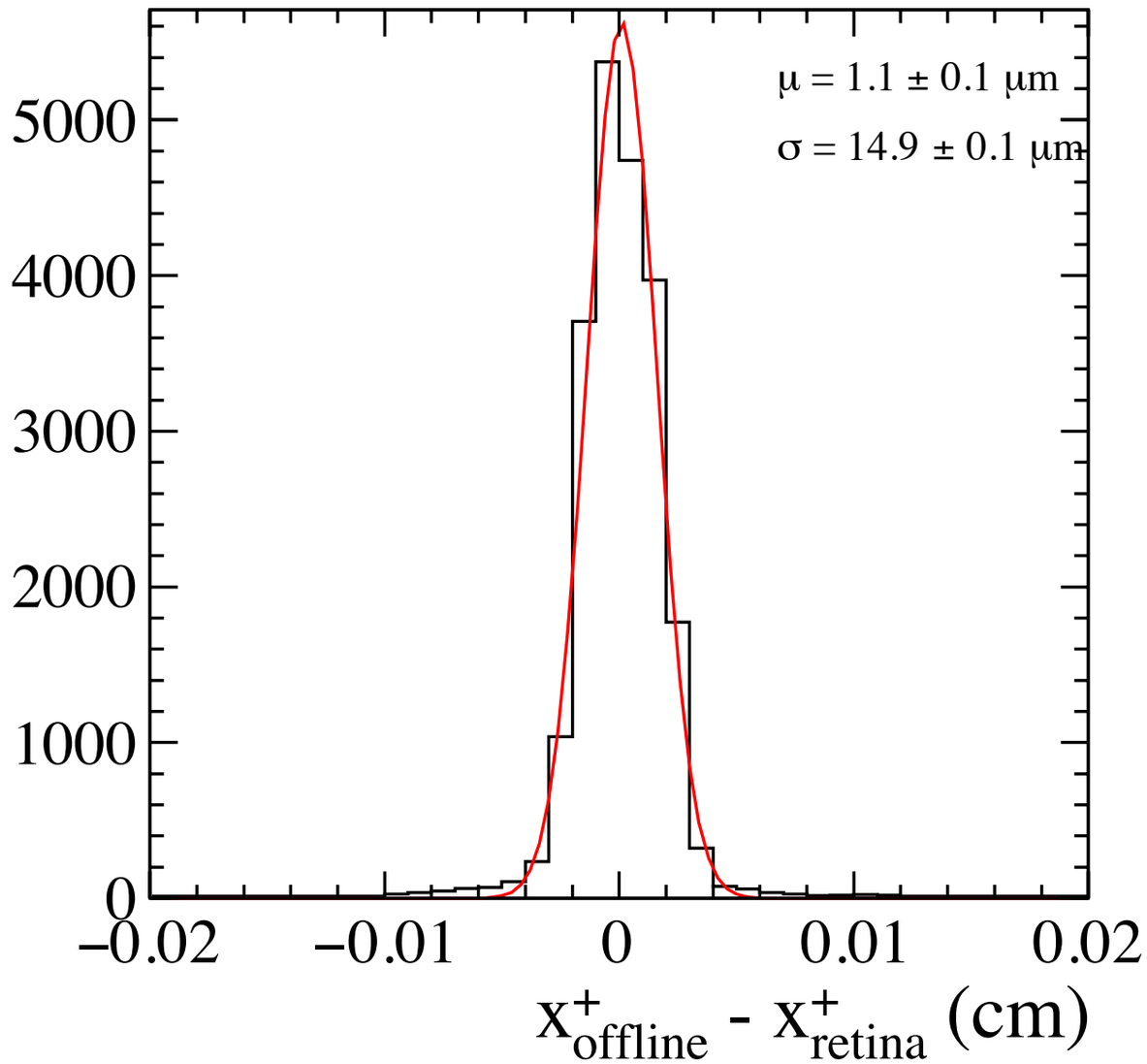




Track residuals: offline - retina

- Detector alignment constants can be fed into the firmware at run time. Sizeable improvement in residuals



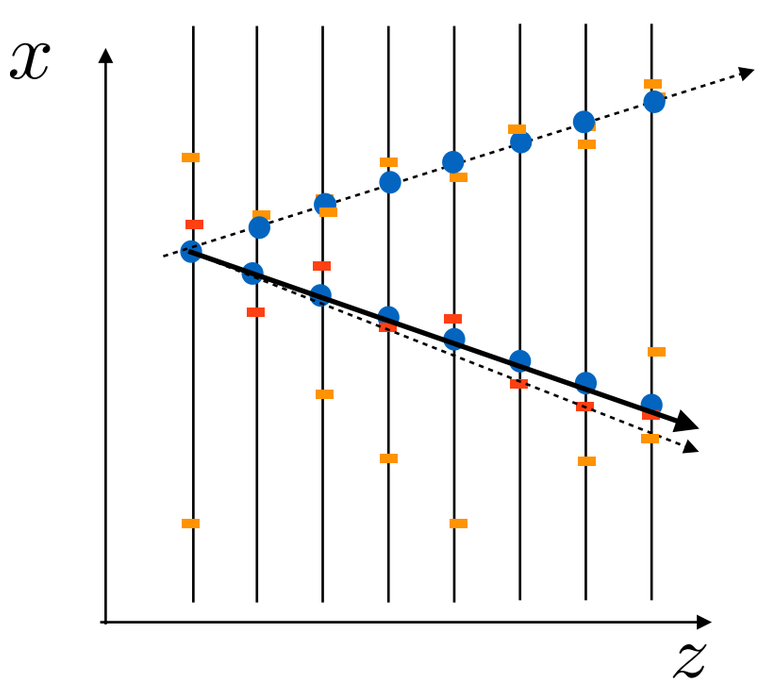


Perspectives: 4D fast track finding

- R&D on ultrafast silicon pixel detectors aims to achieve 10-20 ps time resolution **JINST 9 (2014) C02001**
- Hit time information can be used to further suppress noise hits

$$W_{ij} = \sum_k \exp\left(-\frac{s_{ijk}^2}{2\sigma^2}\right) \exp\left(-\frac{t_{ijk}^2}{2\sigma_t^2}\right) \qquad t_{ijk} = (t_{k,meas} - t_{ijk,exp})$$

Retina with spatial information

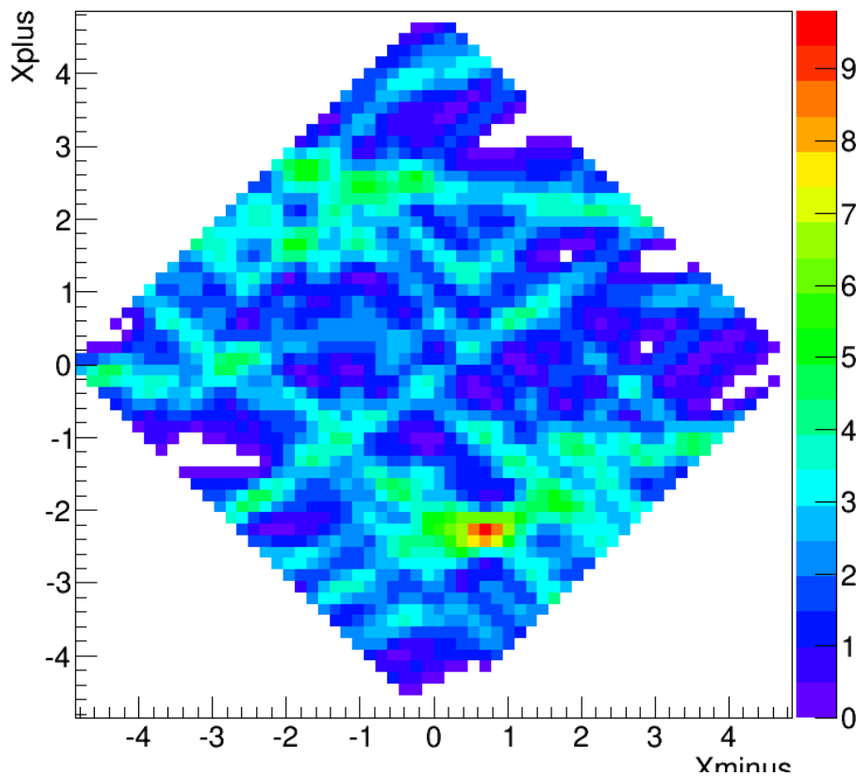


noise hit out of time

fake track

real track

No time information



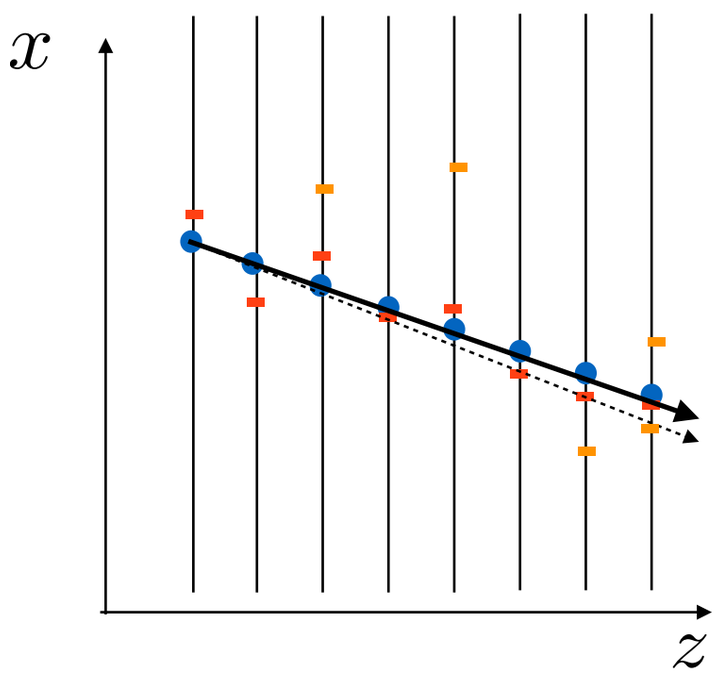
Using precise time information of the hit

- R&D on ultrafast silicon pixel detectors aims to achieve 10-20 ps time resolution **JINST 9 (2014) C02001**
- Hit time information can be used to further suppress noise hits

$$W_{ij} = \sum_k \exp\left(-\frac{s_{ijk}^2}{2\sigma^2}\right) \exp\left(-\frac{t_{ijk}^2}{2\sigma_t^2}\right)$$

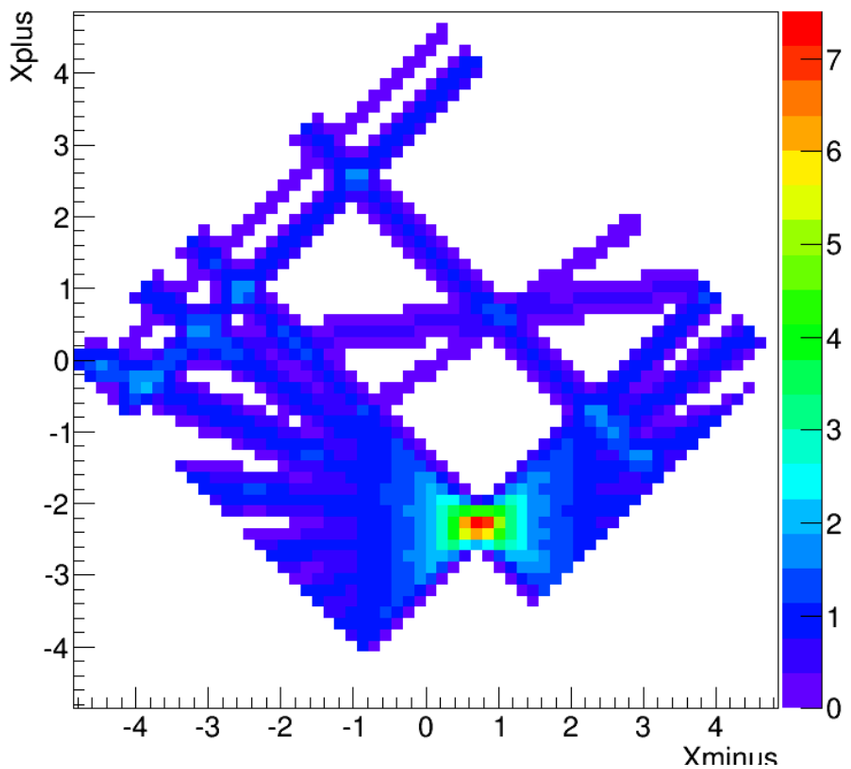
4D fast track finding system
arXiv:1512.09008

Retina with spatial information and time information



noise hit out of time
fake track
real track

time resolution 100 ps



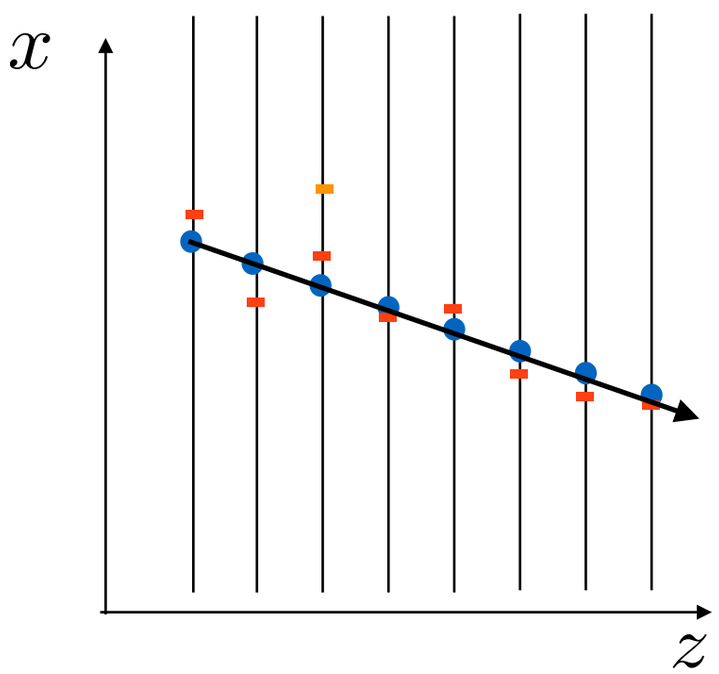
Using precise time information of the hit

- R&D on ultrafast silicon pixel detectors aims to achieve 10-20 ps time resolution **JINST 9 (2014) C02001**
- Hit time information can be used to further suppress noise hits

$$W_{ij} = \sum_k \exp\left(-\frac{s_{ijk}^2}{2\sigma^2}\right) \exp\left(-\frac{t_{ijk}^2}{2\sigma_t^2}\right)$$

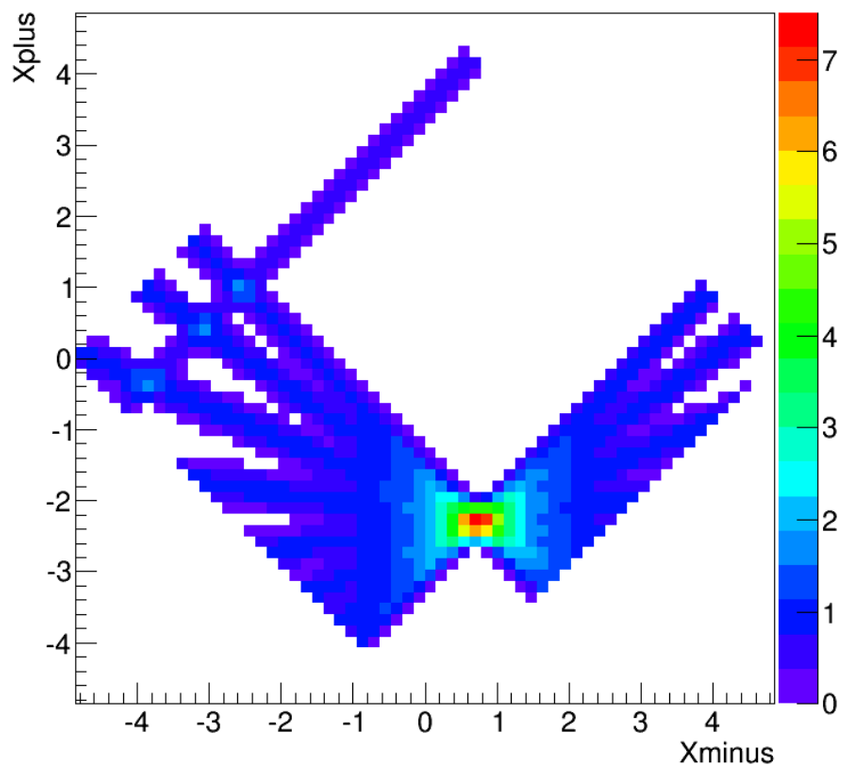
Determine time of the track
arXiv:1512.09008

Retina with spatial information and time information



noise hit out of time
fake track
real track

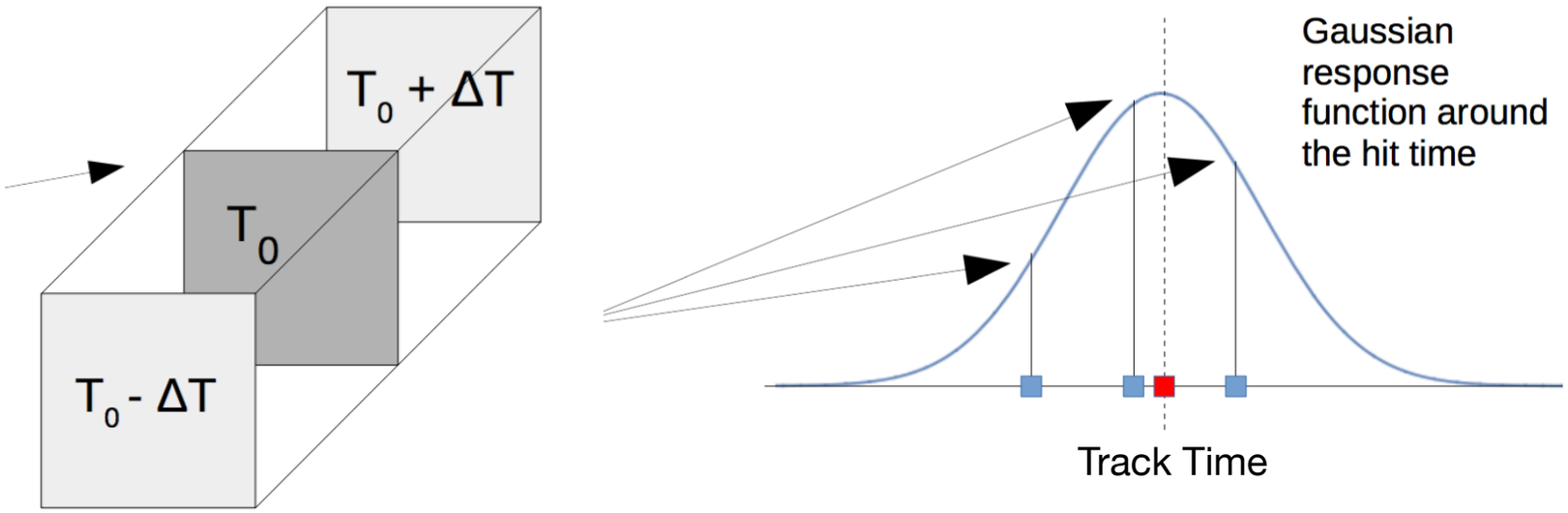
time resolution 10 ps



Evaluation of the time of the track

- Time of the track determined by interpolating retina response at 3 different pre-computed times: T_0-ΔT, T_0, T_0+ΔT.
  T_0= nominal bunch crossing time, ΔT= tuned for optimal response

$W_{ij}(T_i)$



- Determination of the time of the track with few ps precision is possible
- Resolution scales as $\frac{\sigma_t}{\sqrt{N_{hit}}}$ where σ_t is the hit time resolution

Summary

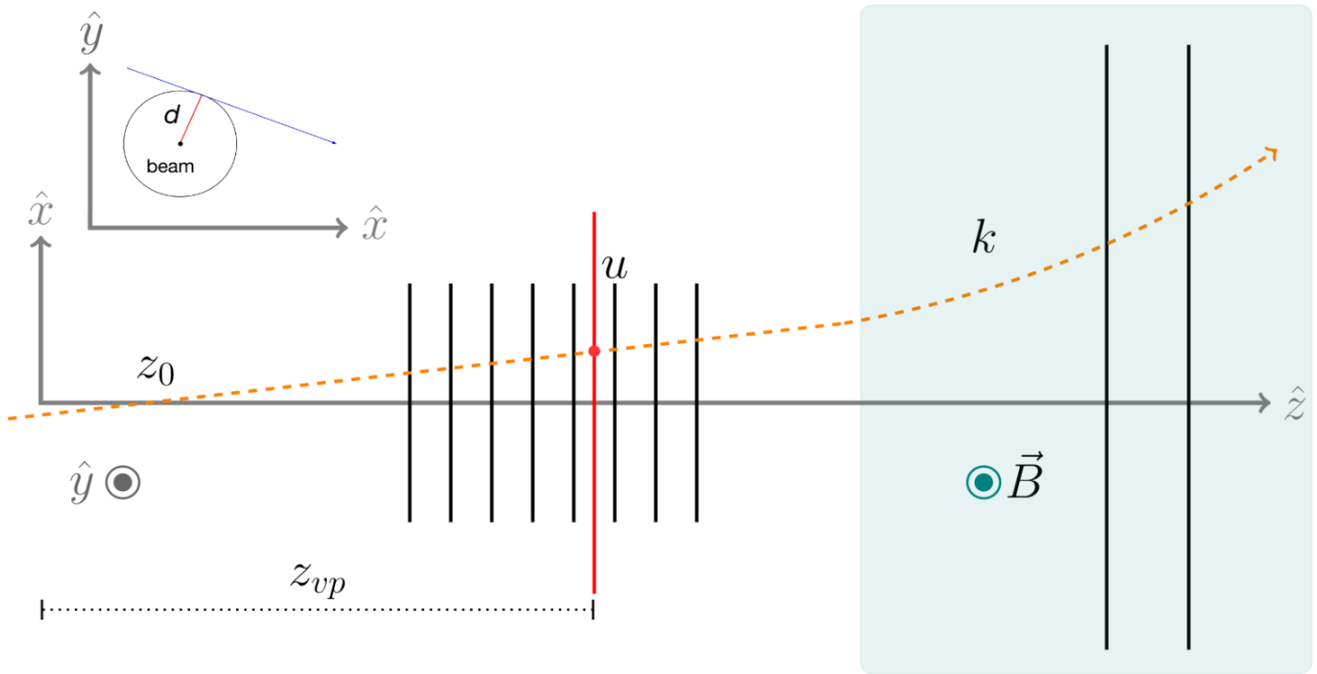
- First realtime tracking system based on artificial retina algorithm tested successfully on beam at the CERN SPS (180 GeV/c protons, track rate about 300 kHz)
- Online retina track parameters in good agreement with offline results and with simulated retina response
- 4D fast track finding system using precise space and time information of the hit. Possibility for fast timing detectors
- Next steps:
  - build a system compatible with large DAQ framework for test with simulated data at 40 MHz and hundreds of tracks per event

Backup slides

Feasibility study for LHC experiments

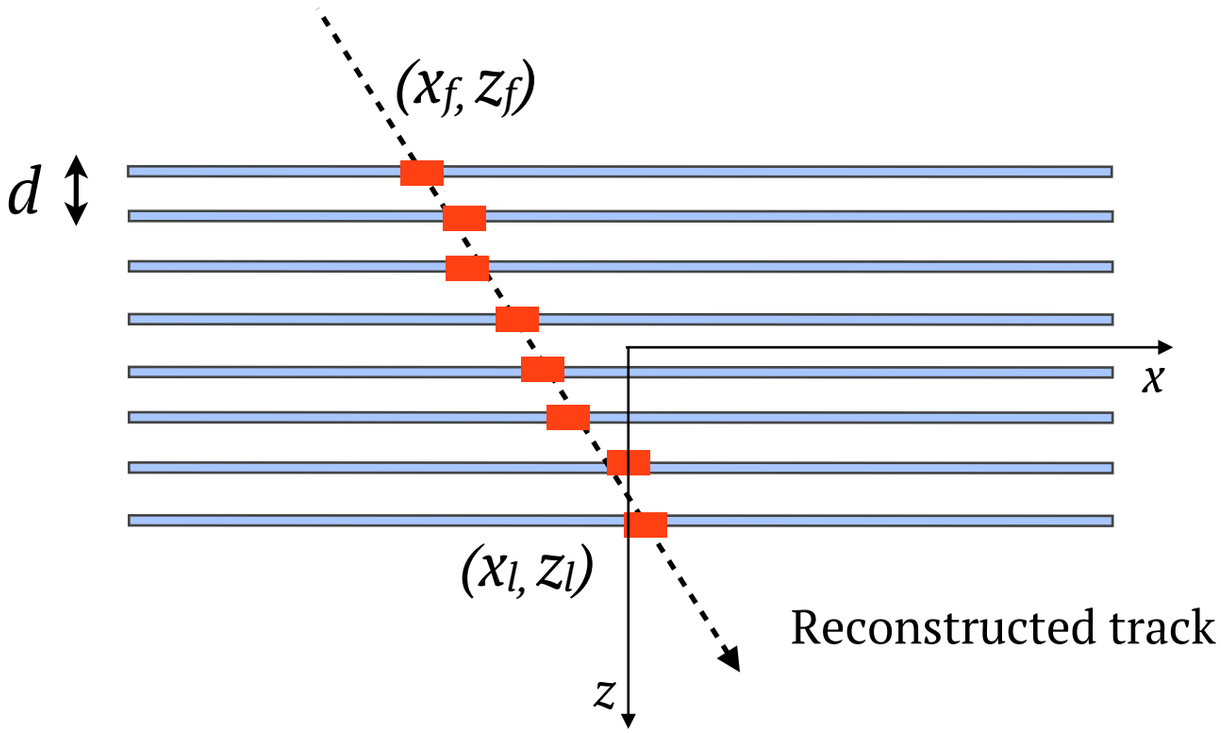
- The Retina architecture is modular, parallel processing units are scalable. Using adequate FPGA resources can cope with high particle rates and large detectors, e.g. 40 MHz event rate and 300 tracks/event of LHC.
- Delivers 3D tracks with offline-like quality at 40 MHz with <1μs latency
- Case study for the LHCb upgrade simulated and documented here: LHCb-PUB-2014-026, JINST 9 C09001 (2014). Affordable resources and cost (50,000 cells ~ 50 FPGA)

Application in forward spectrometer experiment



- 50 mrad acceptance
  - *O*(100) particles/event
- 8 pixel layers
- 2 silicon strip layers
- ~0.05 T magnetic field
- Pileup: ~8 *pp* events

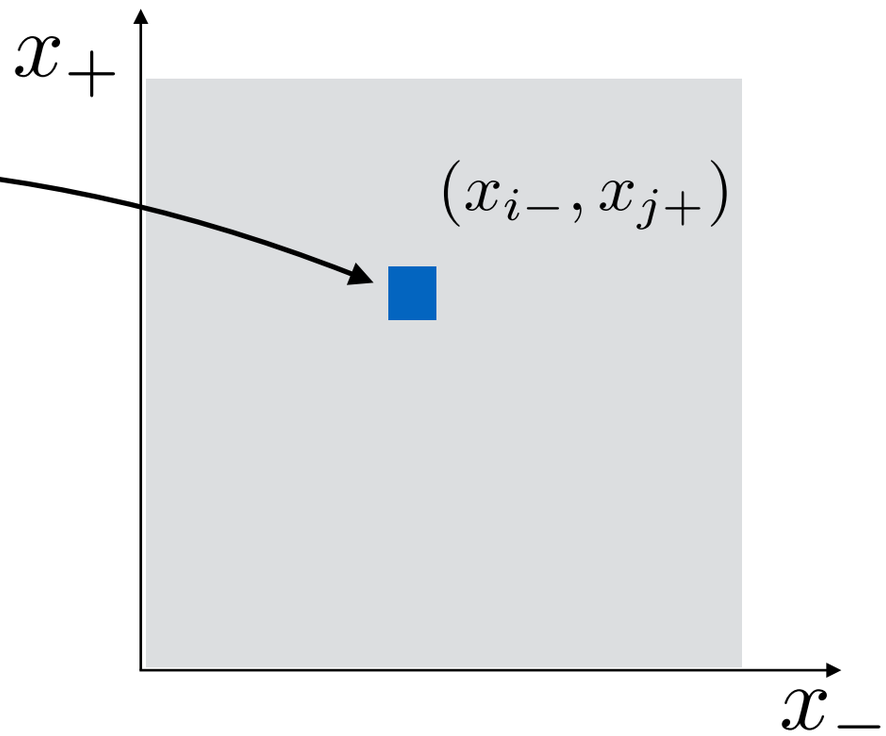
Track parameters for prototype



Track parameters

$$x_\pm = \frac{x_l \pm x_f}{2}$$

Grid of track parameters

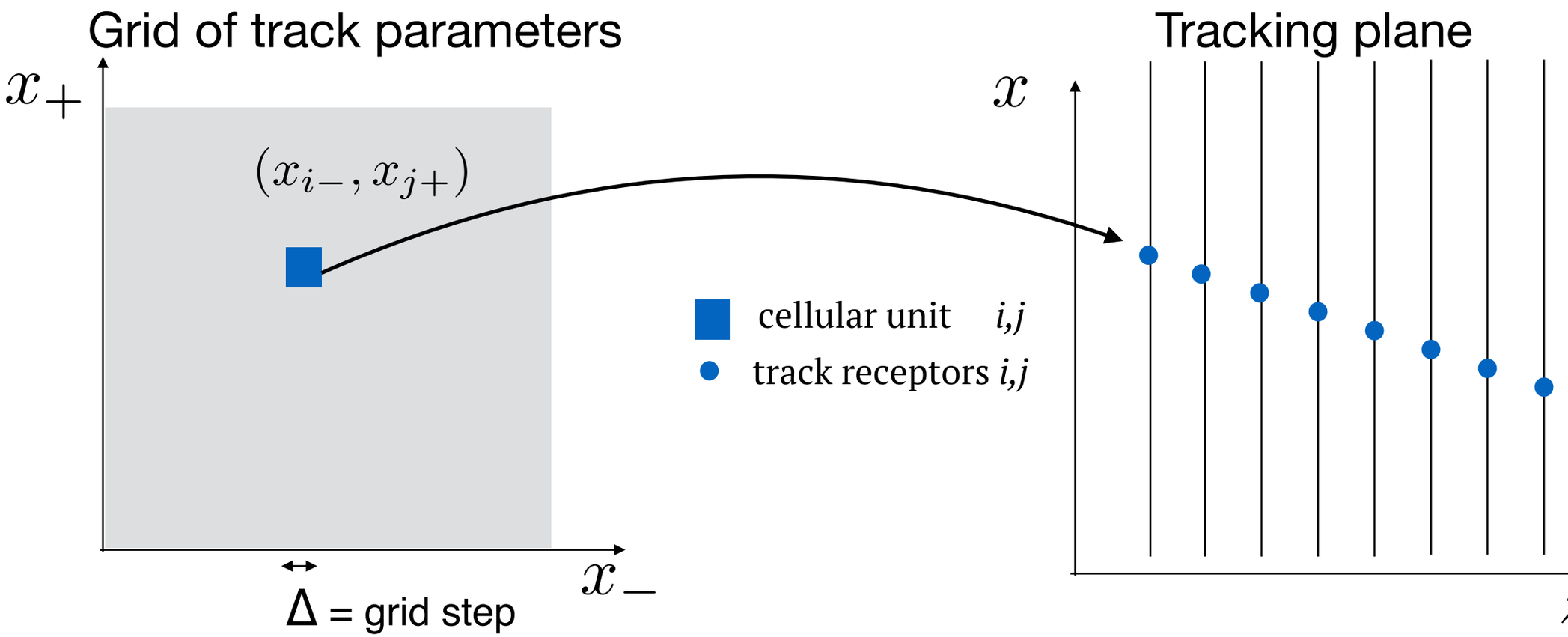


A 2-dim track is described as

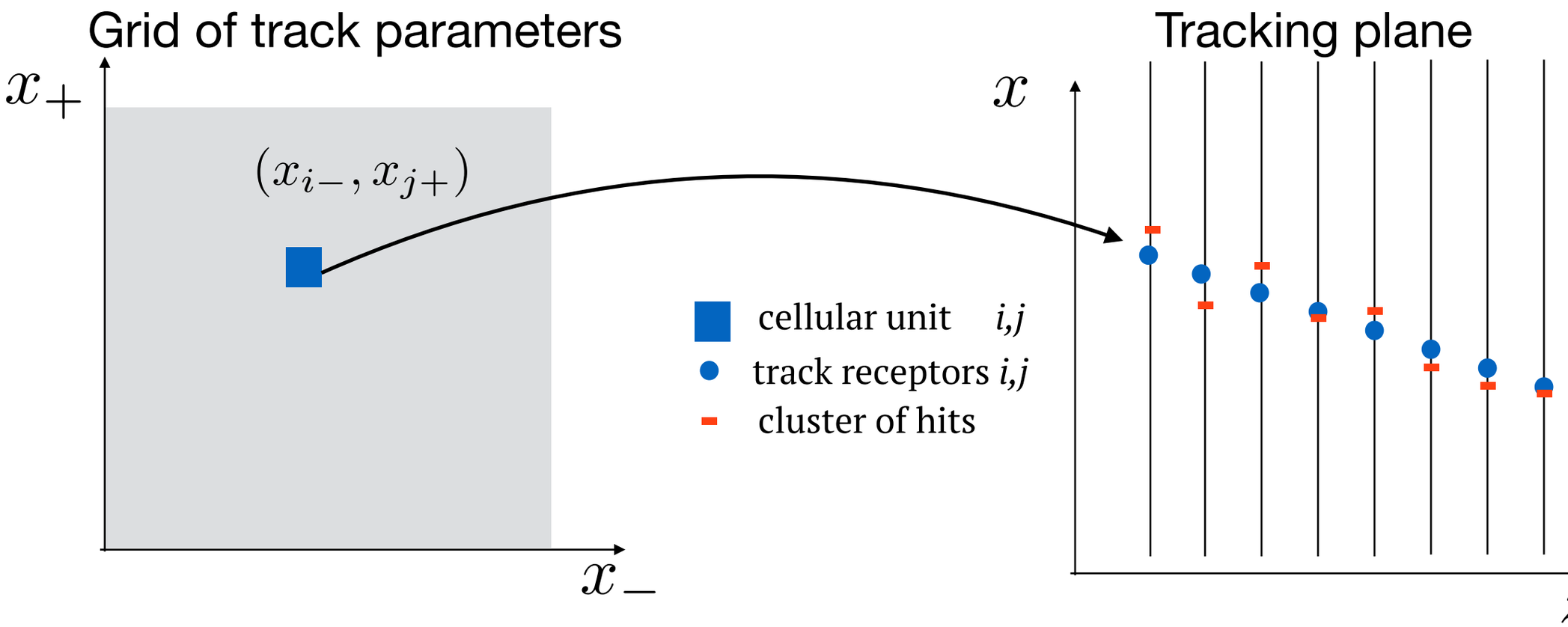
$$x(z) = x_+ + x_- \frac{z - z_+}{z_-}$$

$z_\pm = \frac{z_l \pm z_f}{2}$ (constant terms)

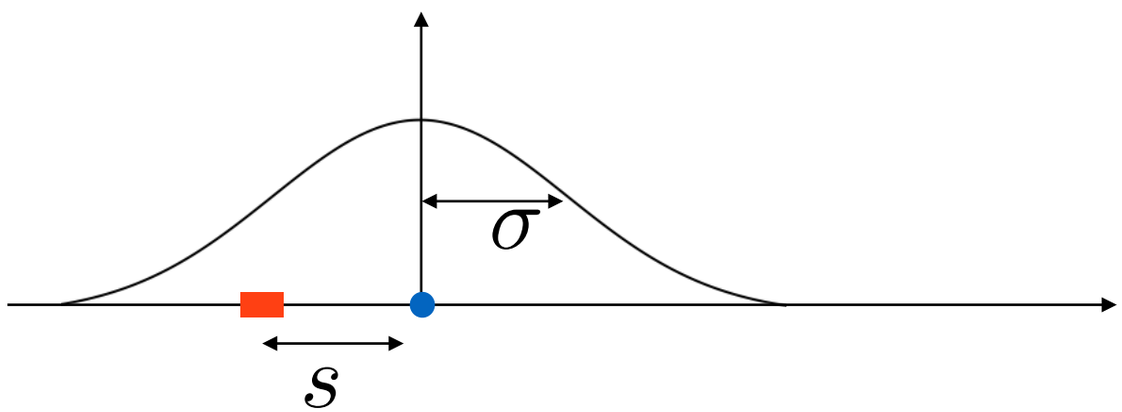
Track receptors



Track receptors



Receptor response

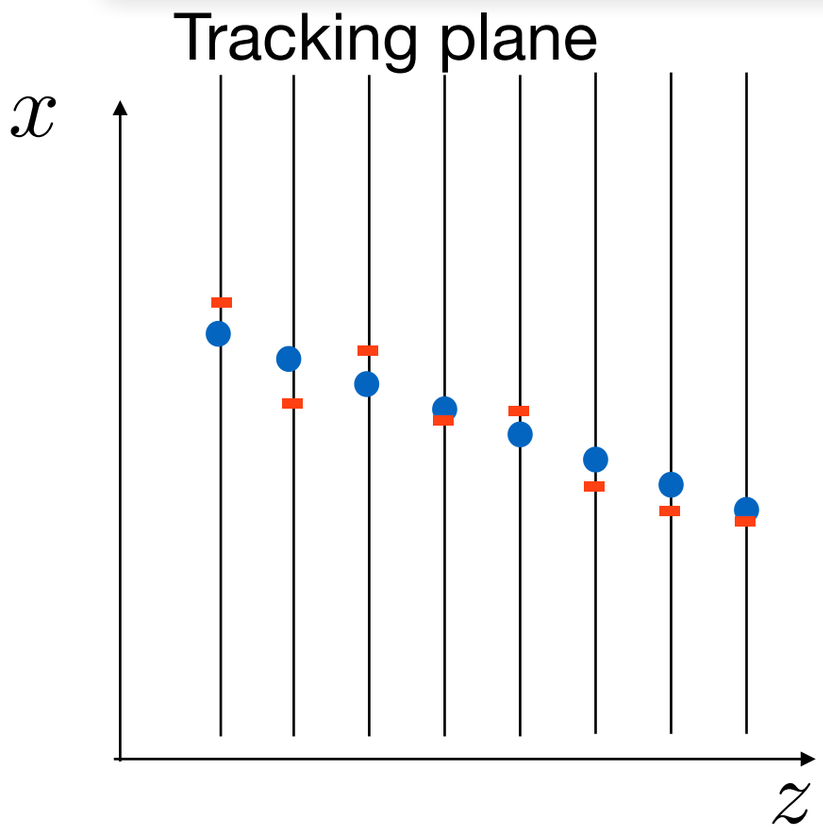


σ = width of the receptor response field.

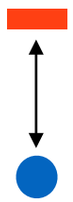
$\sigma \simeq \Delta$ grid step

It is much larger than the obtainable resolution on track parameters and has to be tuned.

Retina algorithm



s_{ijk} distance between cluster in layer *k* and track receptor *i,j*



$$s_{ijk} = \left| x_k - x_{j+} - x_{i-} \frac{z_k - z_+}{z_-} \right|$$

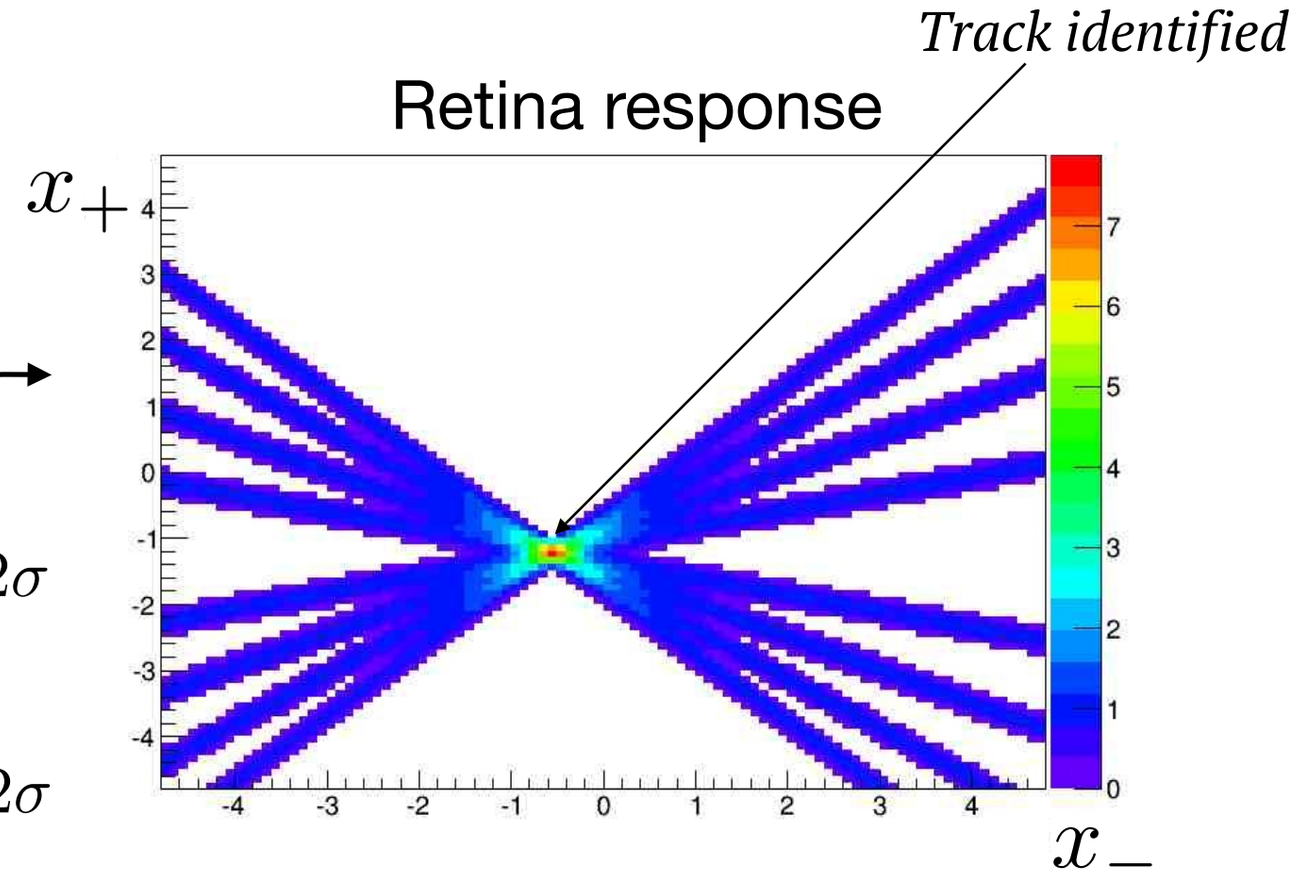
cluster of hits

track receptors *i,j*

Excitation of the cellular unit i,j

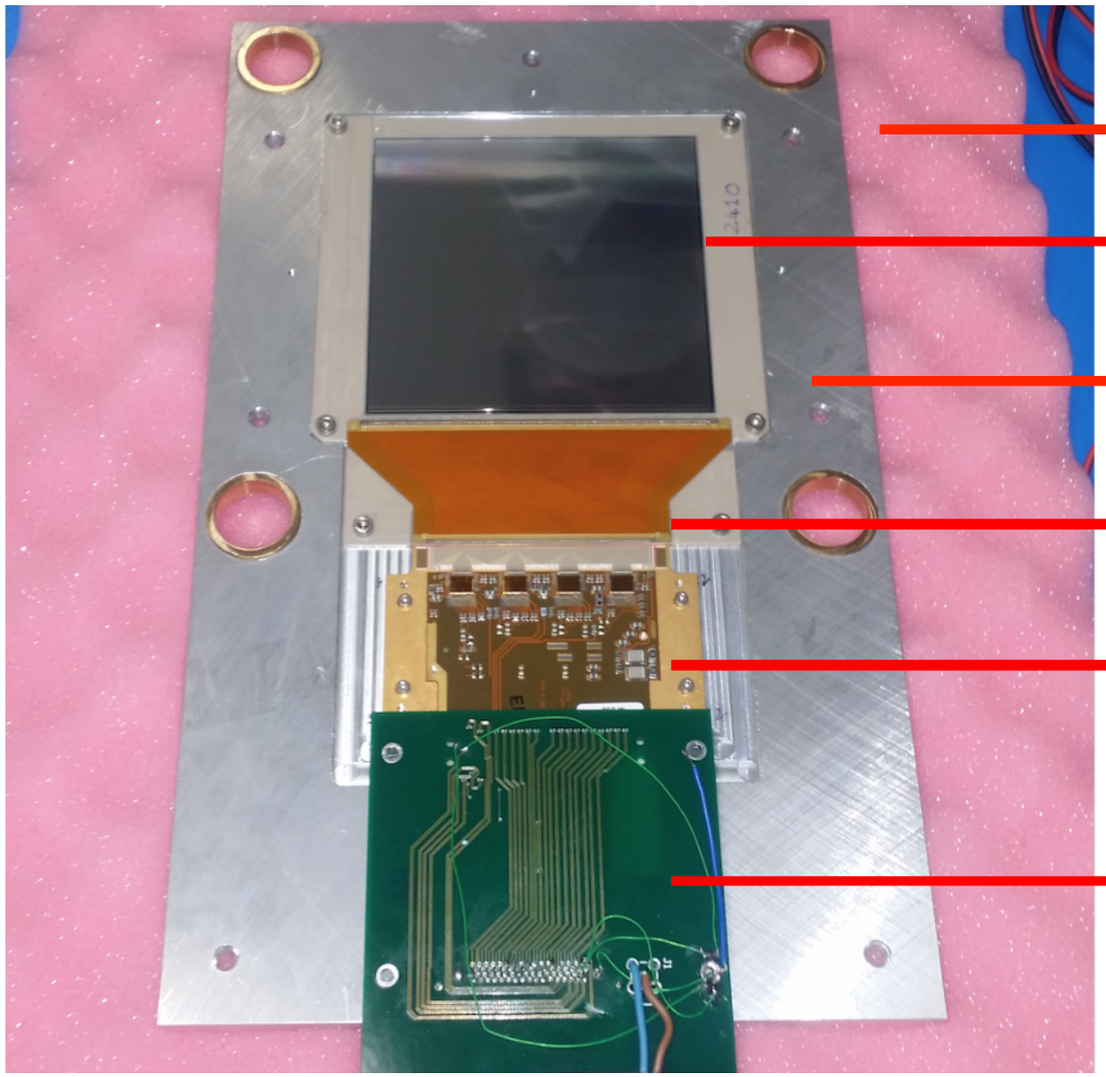
$$W_{ij} = \sum_k \exp\left(-\frac{s_{ijk}^2}{2\sigma^2}\right) \quad \text{if } s_{ijk} < 2\sigma$$

$$W_{ij} = 0 \quad \text{if } s_{ijk} > 2\sigma$$



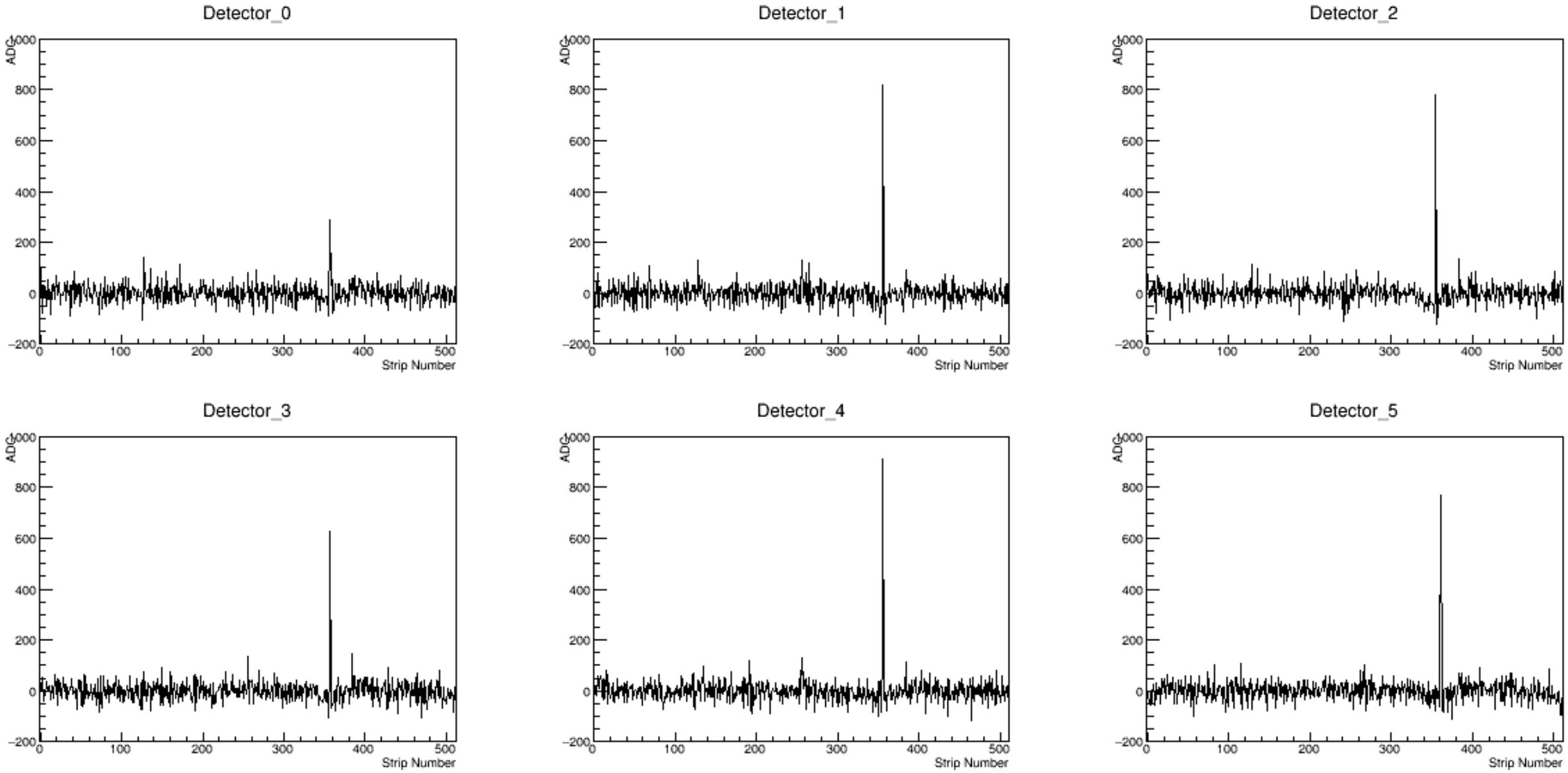
Telescope module

- Single-sided silicon sensors:
  - OB2 STM *p*-in-*n* sensor, 10 cmx10 cm active area
  - 512 strips, 183 μm pitch, 500 μm thickness

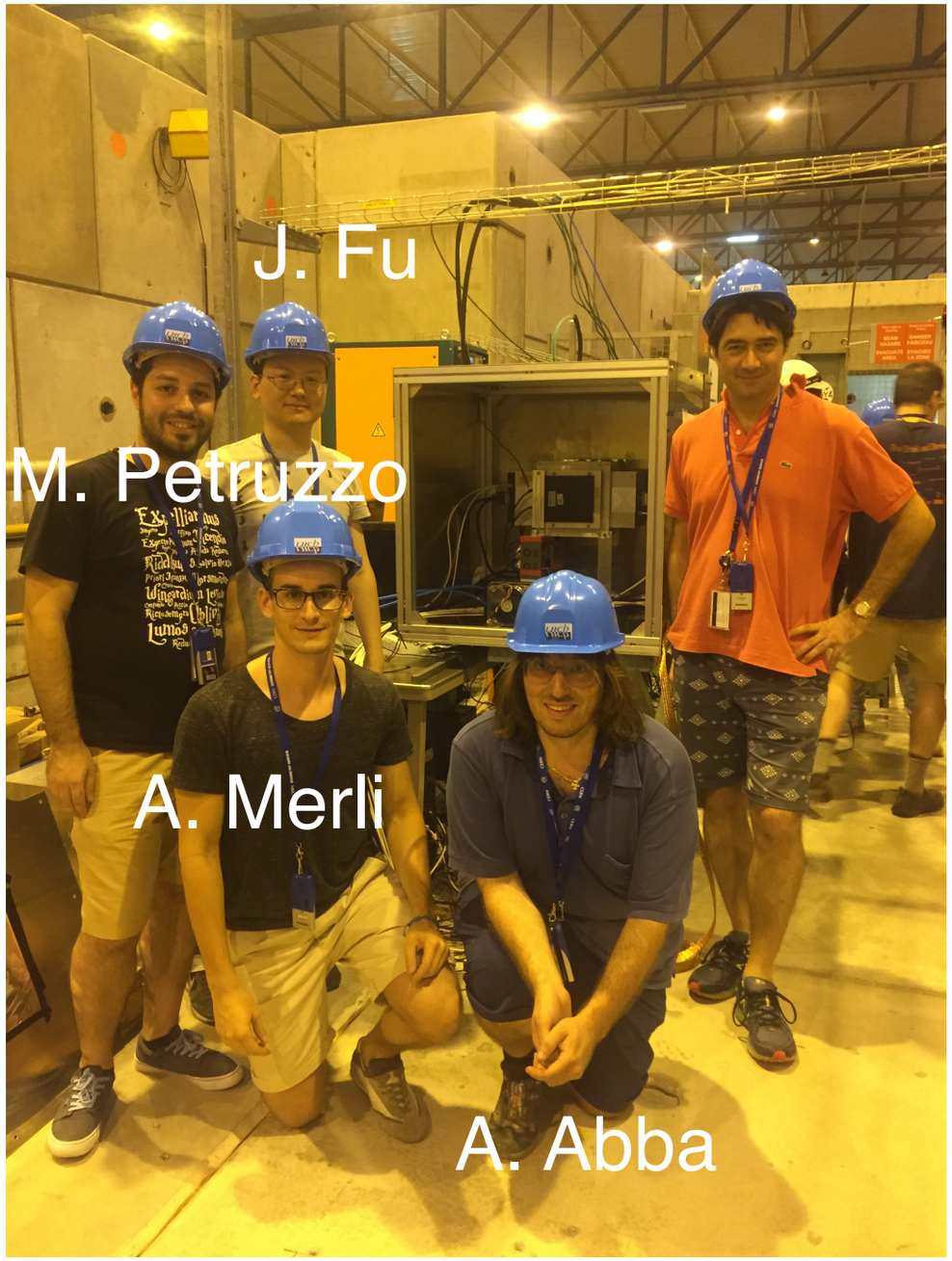


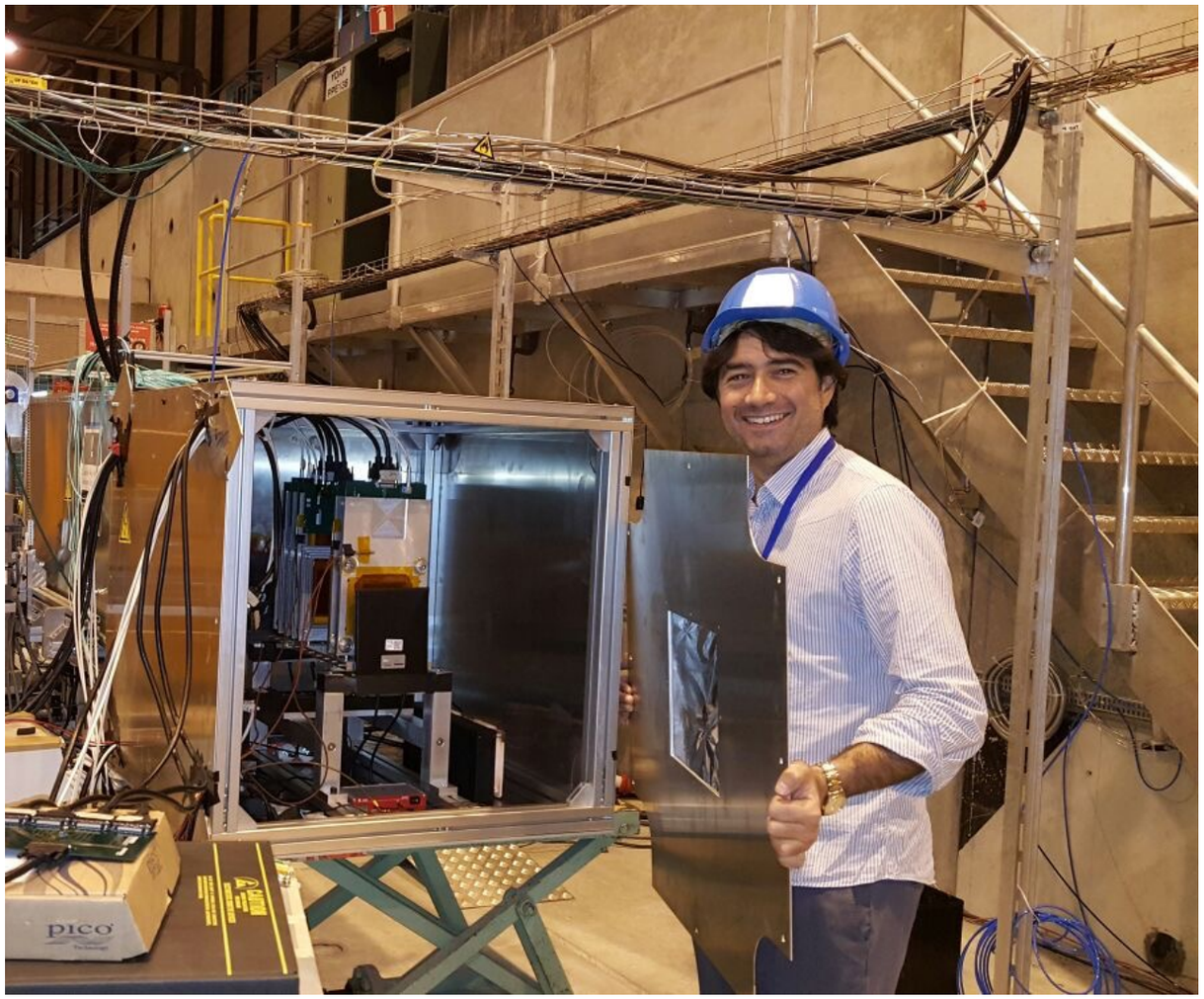
Event display

- Event display for 1 track event: ADC vs strip number

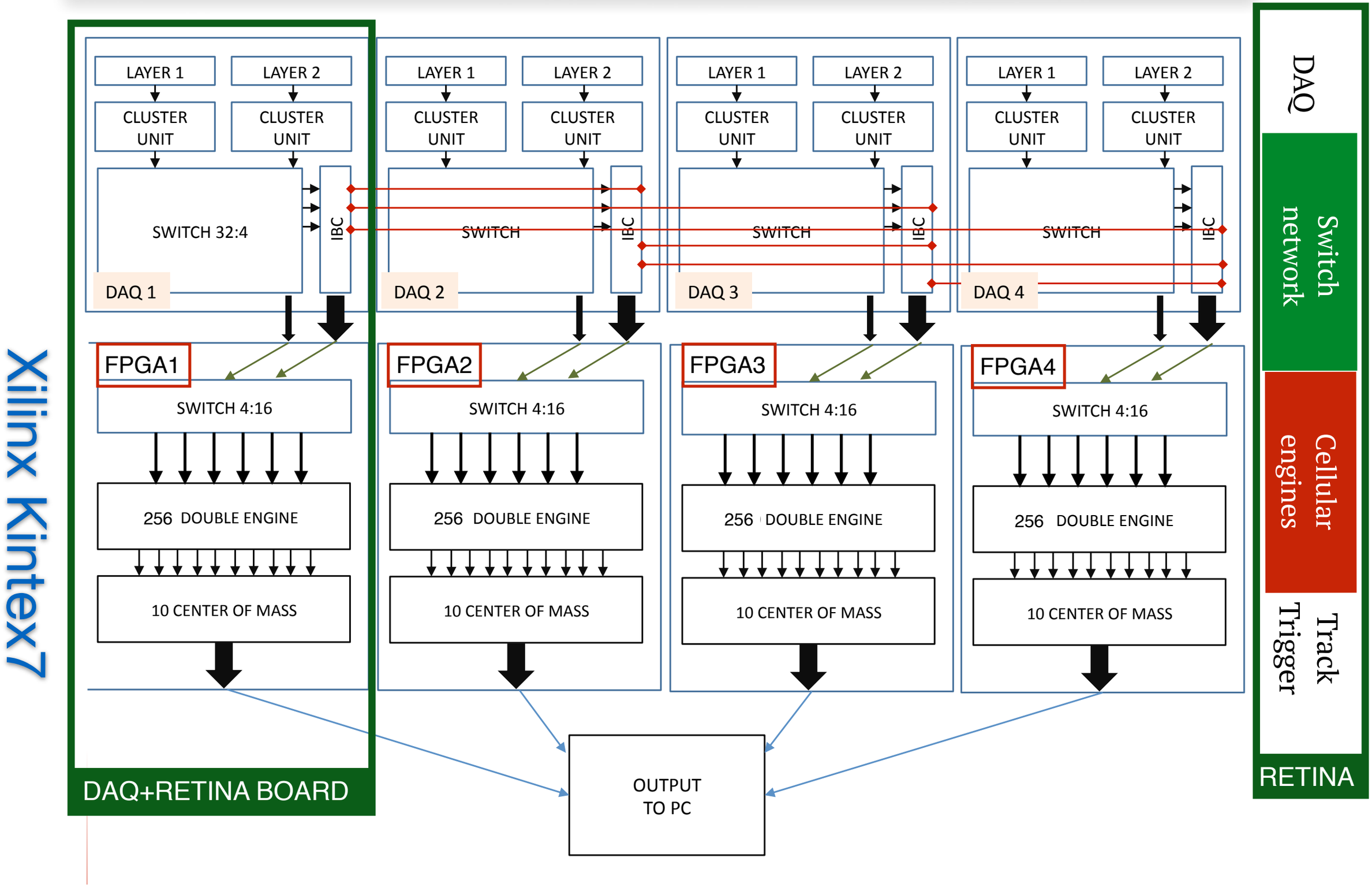


Testbeam crew



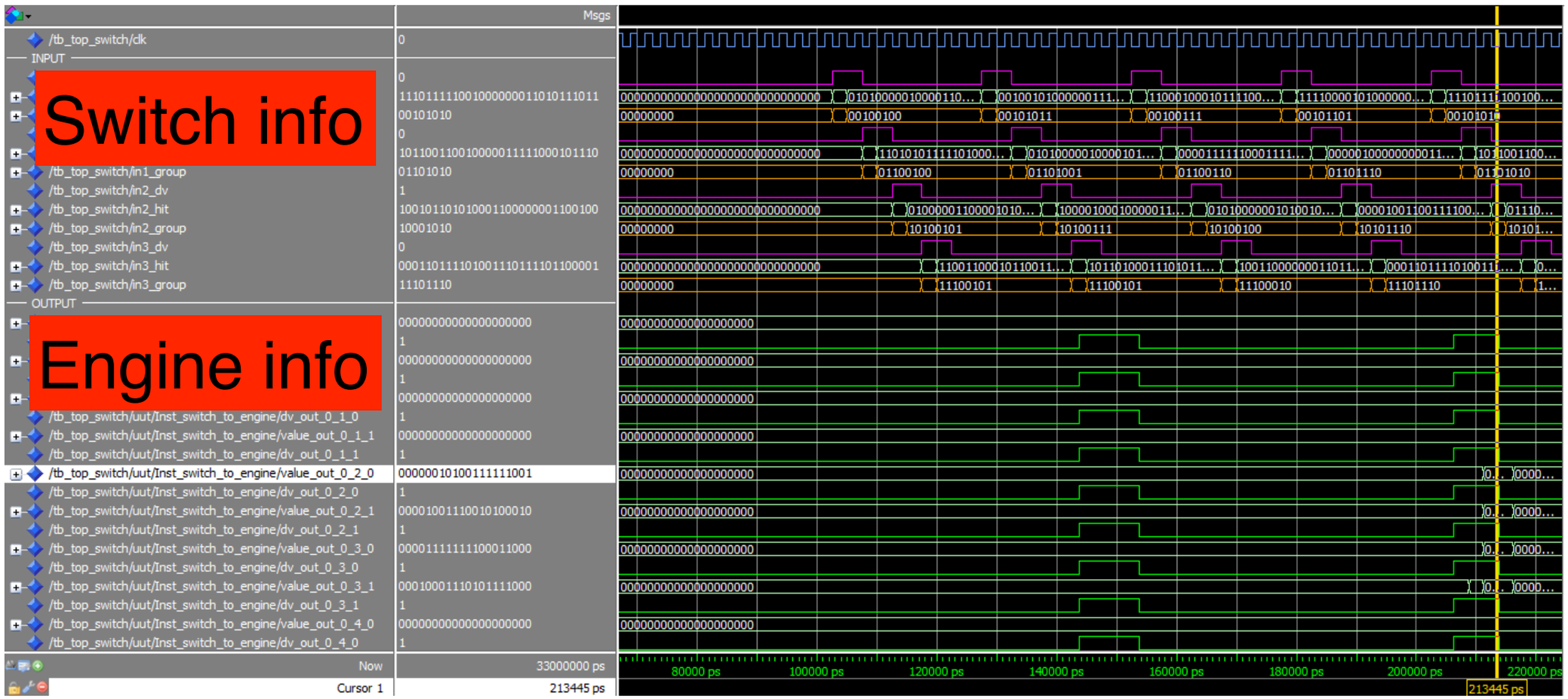


Retina architecture v2



Retina architecture simulation

- ModelSim results on Xilinx Kintex7 FPGA
- Switch+Engine simulated successfully up to 40 MHz input track rate



Latency of Retina response

SWITCH 14 t.u. | ENGINES 12 t.u. | TRACK FITTER 30 t.u.

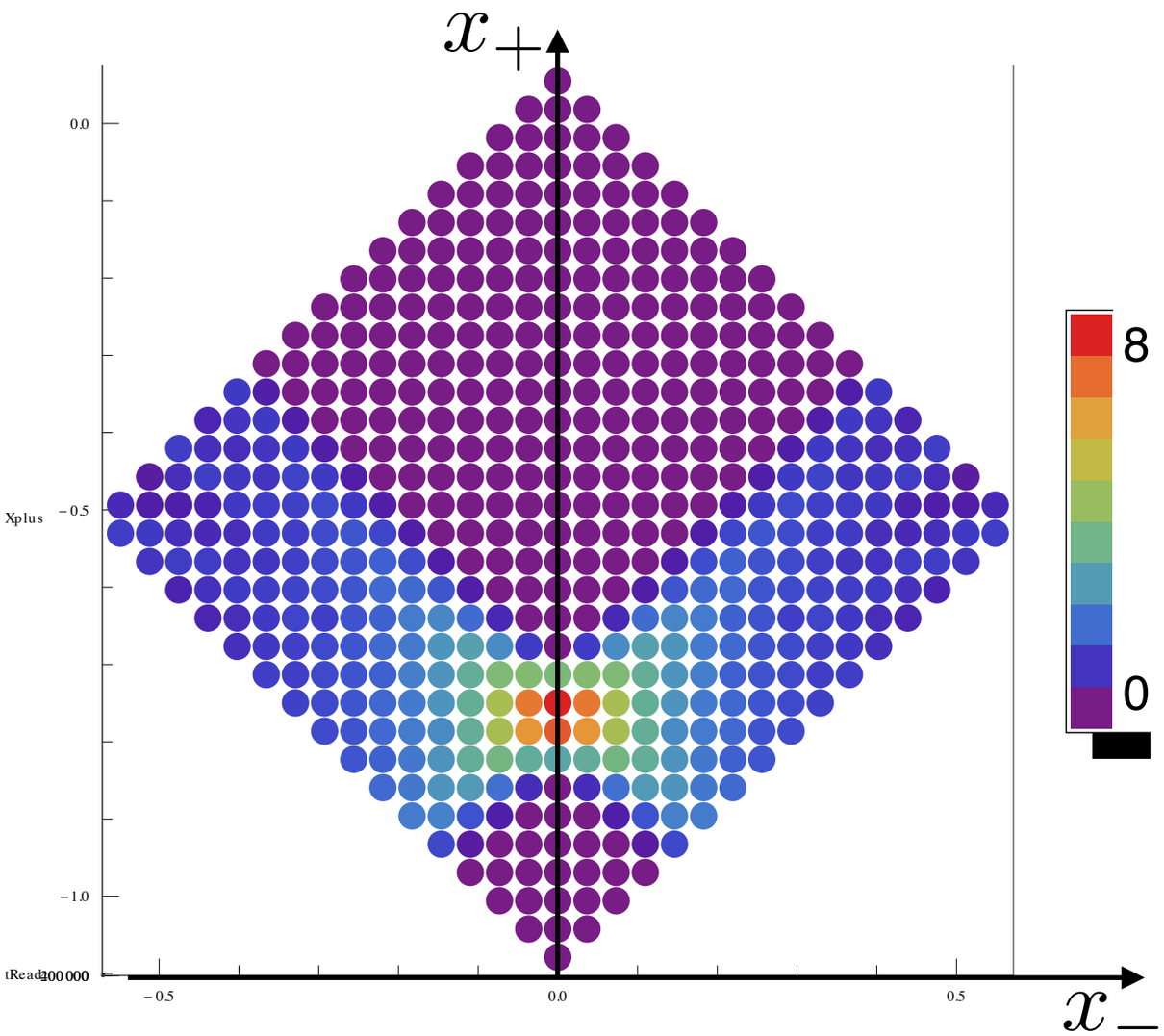
Clock frequency 400 MHz, i.e. 1 t.u. = 2.5 ns

<100 t.u.

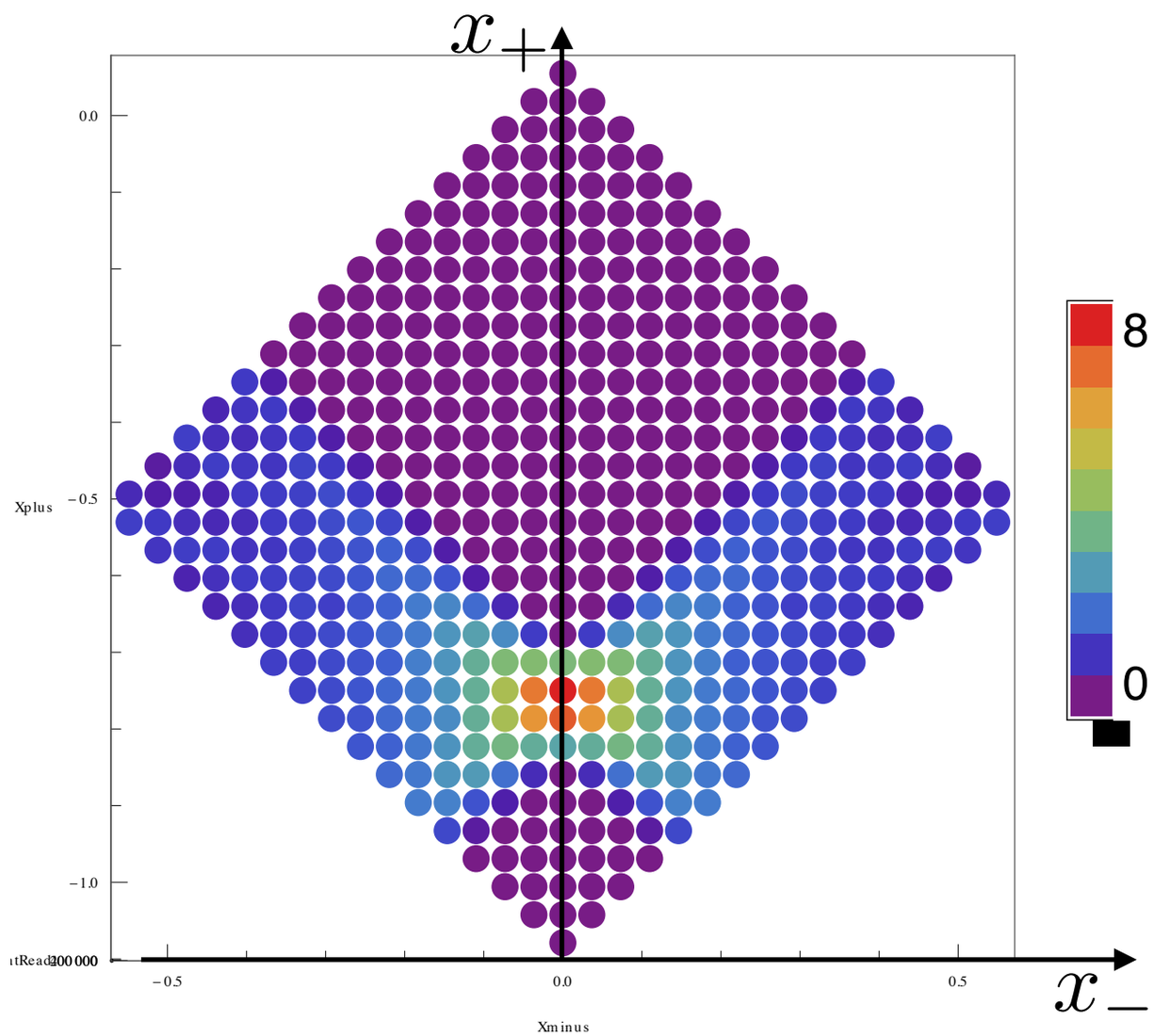
Retina response to tracks

The retina algorithm implemented in hardware is working properly

Weight distribution from high level C++ simulation for retina cellular units



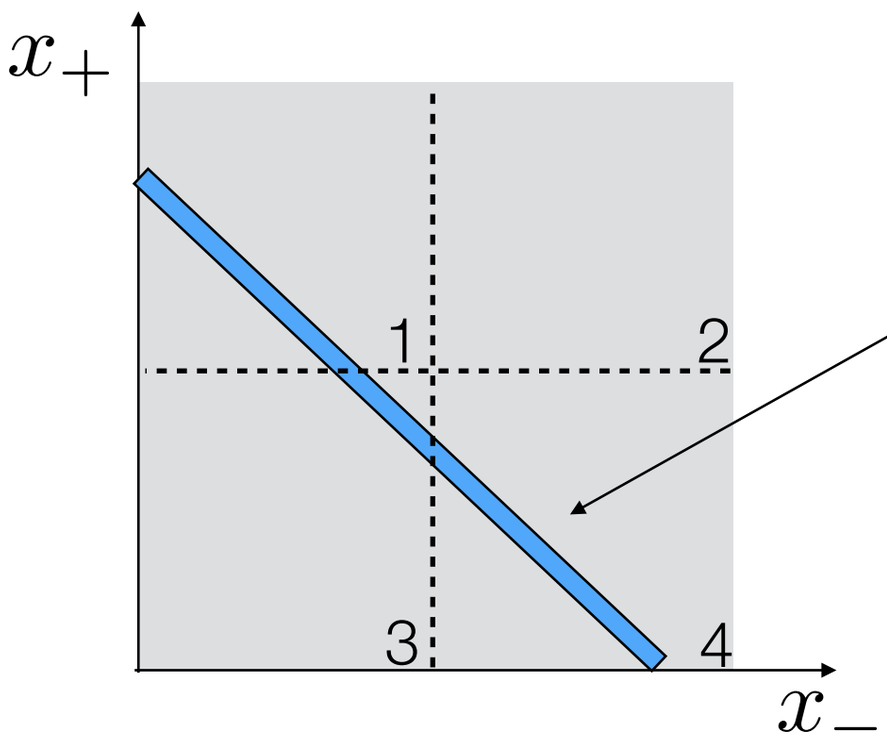
Weight distribution from ModelSim (Xilinx Kintex7 FPGA)



Delivering the data

- Engines receives data, through the switch, from all the tracking detectors
- Divide the grid in 4 regions corresponding to the number of available FPGAs (4 Xilinx Kintex7) for the processing engines.

A cluster seen by the retina



A cluster *(x,z)* corresponds to a line in the grid of parameters (x_+,x_-)

$$x_+ = -x_- \frac{z - z_+}{z_-} + x$$

- Engines with non negligible weights belong to different regions of the grid

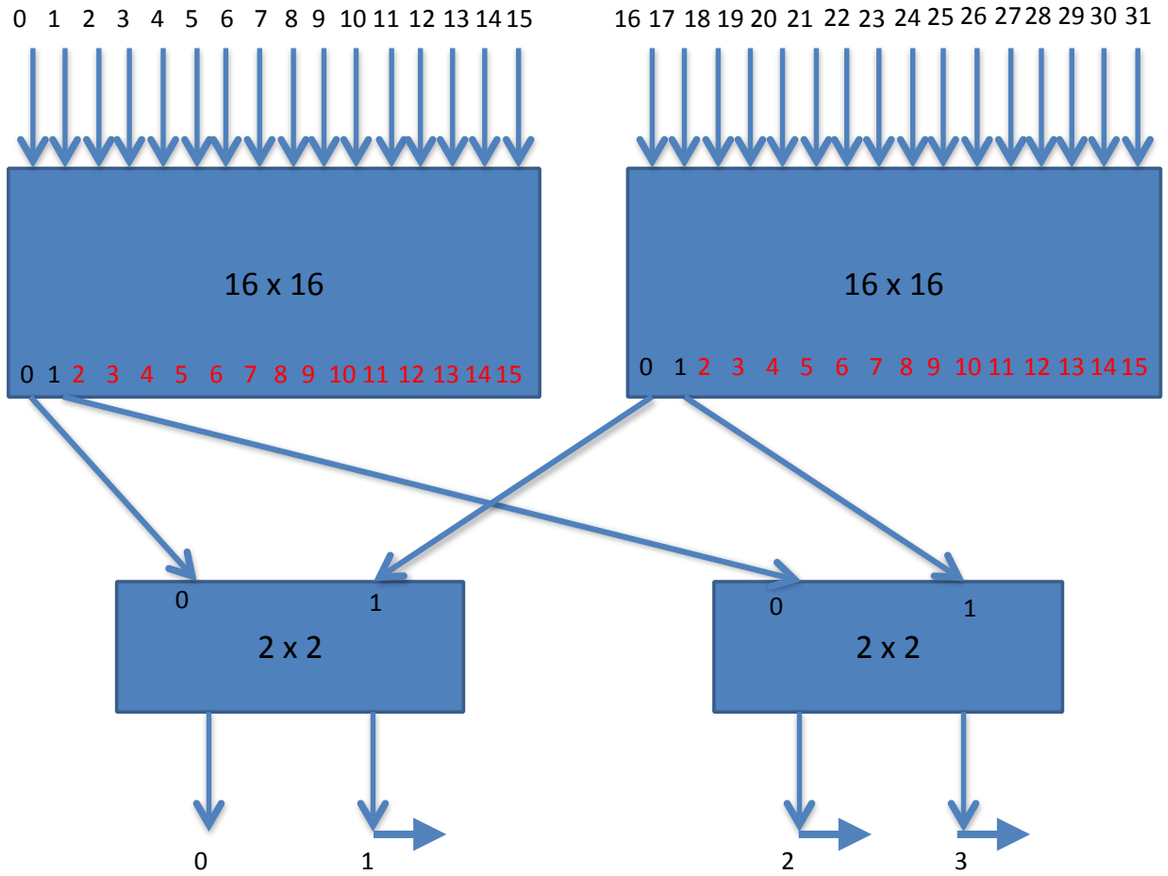
$$|x - x_+ - x_- \frac{z - z_+}{z_-}| < 2\sigma$$

- Deliver the data to the engines (in different FPGAs) using a *full mesh switch*
- *z* determines the slope and *x* the intercept with x_+ axis of a cluster in the (x_+, x_-) plane
- Data path is determined by the cluster coordinates *(x,z)* using 8 bit information: 5 bit for *x* and 3 bit for *z*

Switch modules

1st switch 32x4

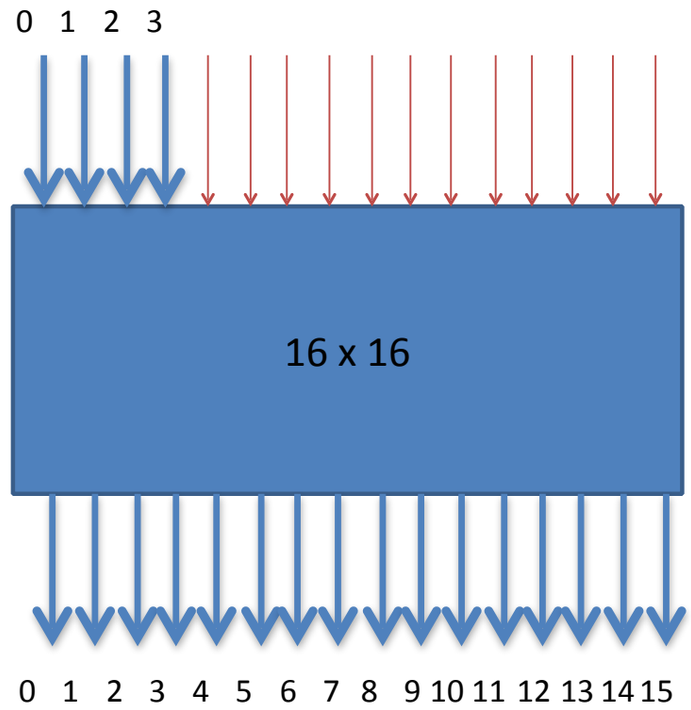
16 analog inputs from each sensor.



4 output ports: 1 output to 2nd switch level and 3 output to the other DAQ boards

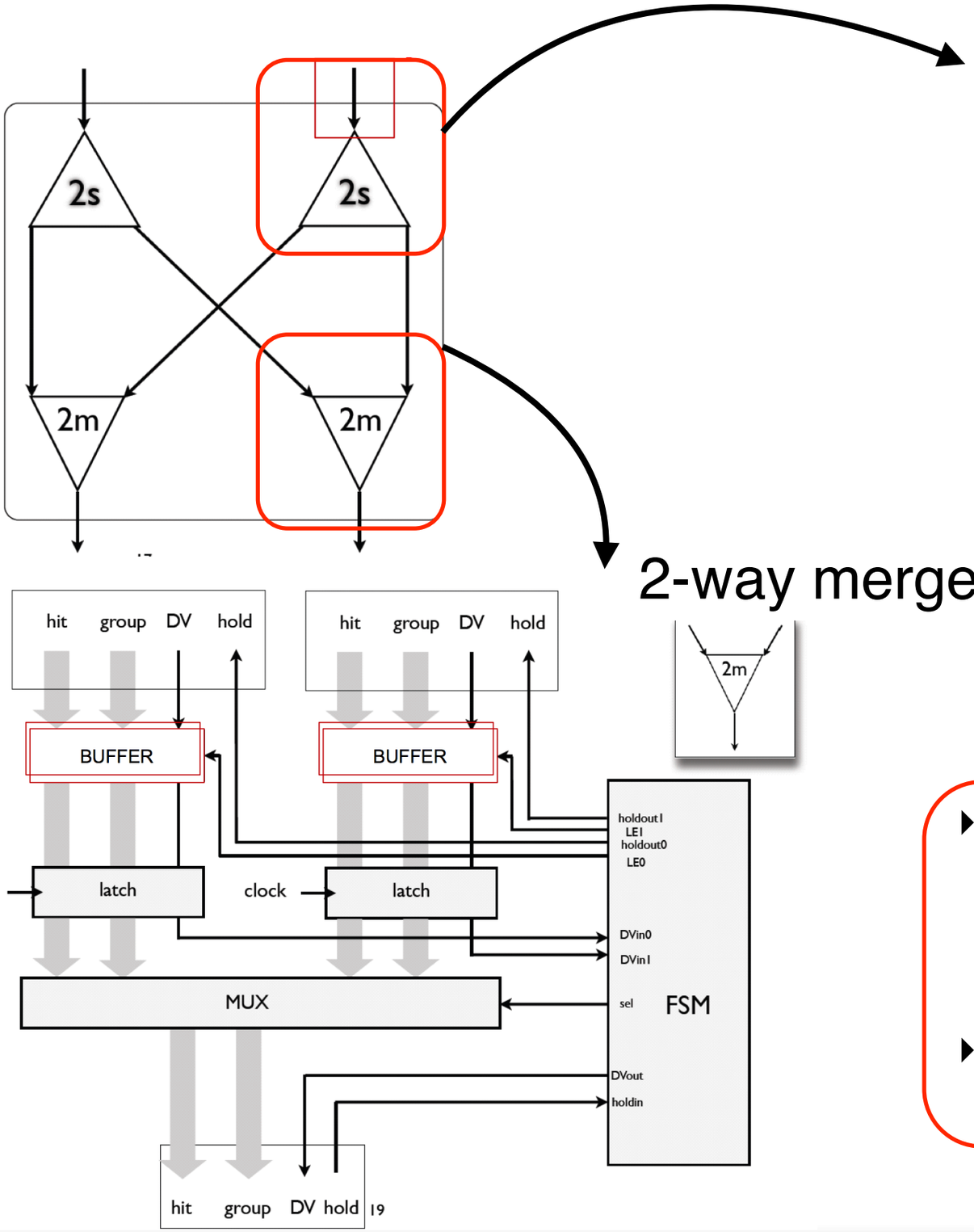
2nd switch 4x16

4 input ports: 1 from each DAQ board



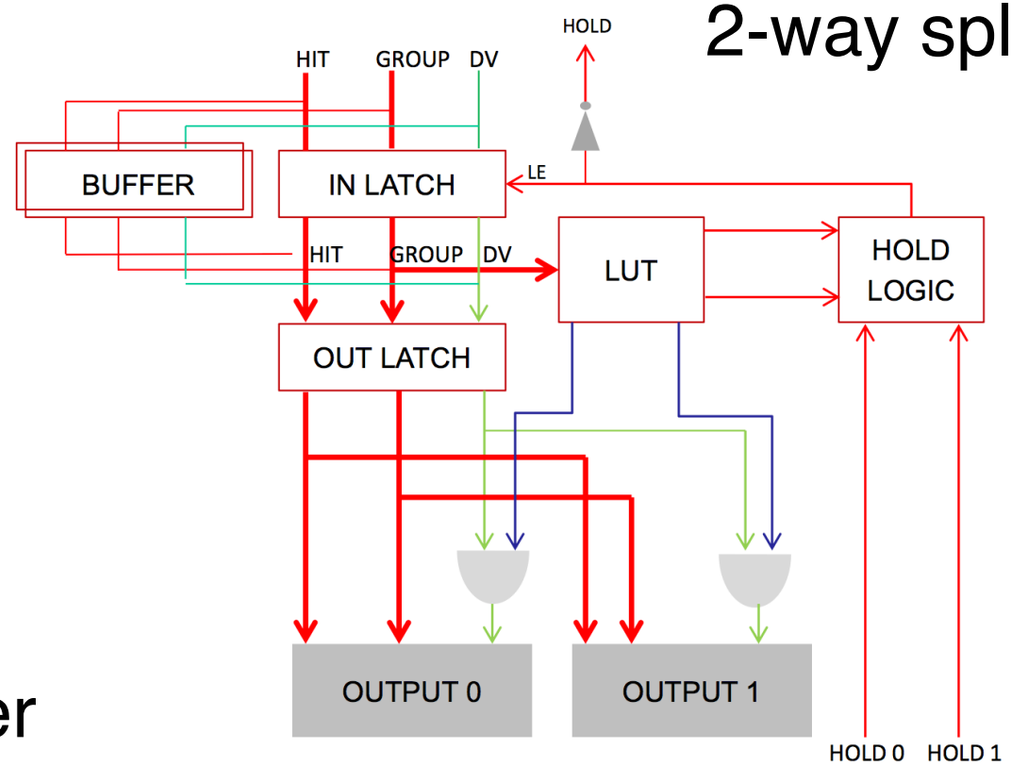
16 output ports, each connected to 16 engines in parallel

2-way sorter



2-way merger

2-way splitter



- The 2-way sorter acts also as a memory buffer in case of traffic jam. Input stream can be held
- Switch functionalities validated with VHDL simulation