FPGA accelerated HPC for Experimental Physics

Ken Oyama

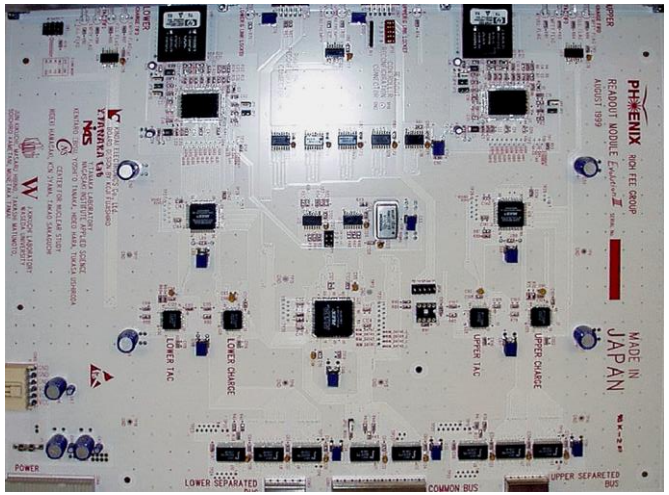
Nagasaki Institute of Applied Science

Jun. 6-7, 2019, HEART2019 @ Nagasaki

JP17H02903
JP16K13808
JP17K18783

Introduction

- I'm neither computer scientist nor FPGA engineer, but nuclear physicist
- In physics, especially nuclear and particle physics, we have been using reconfigurable system and even FPGA since decades



teledynelecroy.com

HELIOS (CERN NA34)

physicist's reconfigurable system in 1980th

- end 20th century, physicists started using FPGA for data transmission, trigger logic, controlling, etc.
- pioneering work in NIAS in PHENIX experiment at Brookhaven National Laboratory in US

but not much for computing

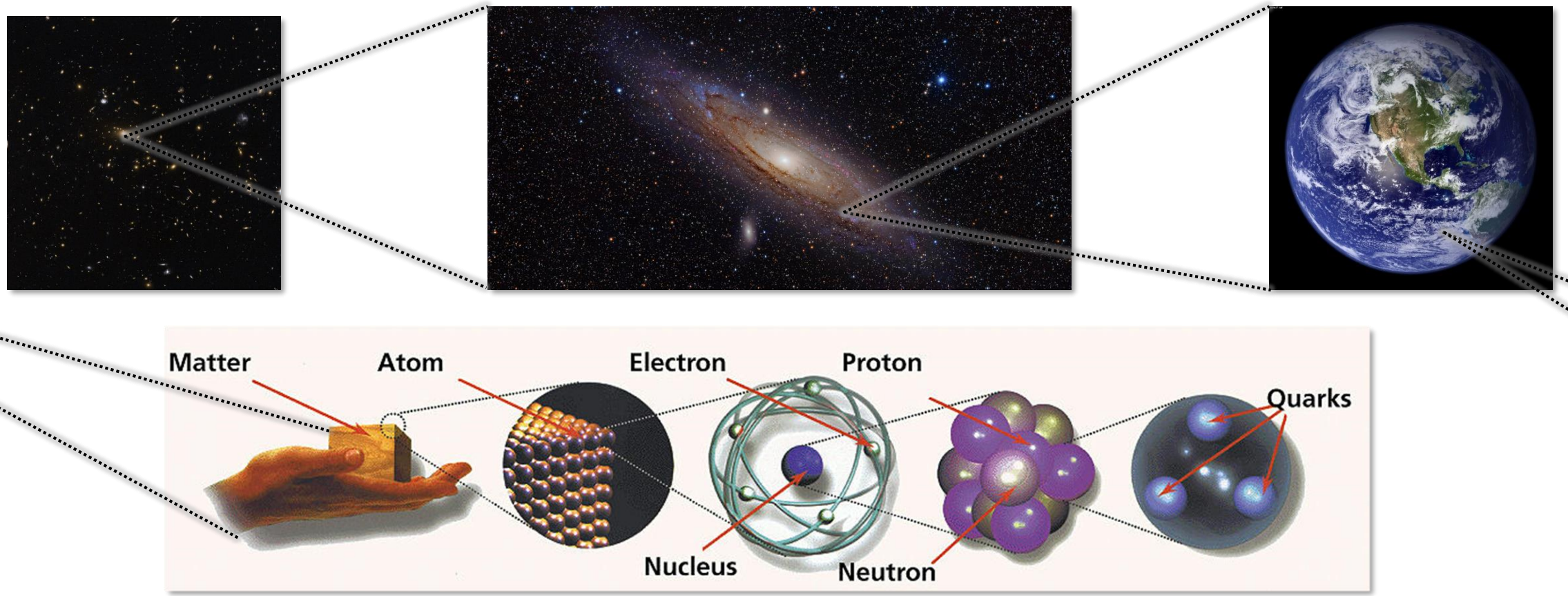
however, now, we (must) use FPGA also for computing and big data processing

Outline of this talk

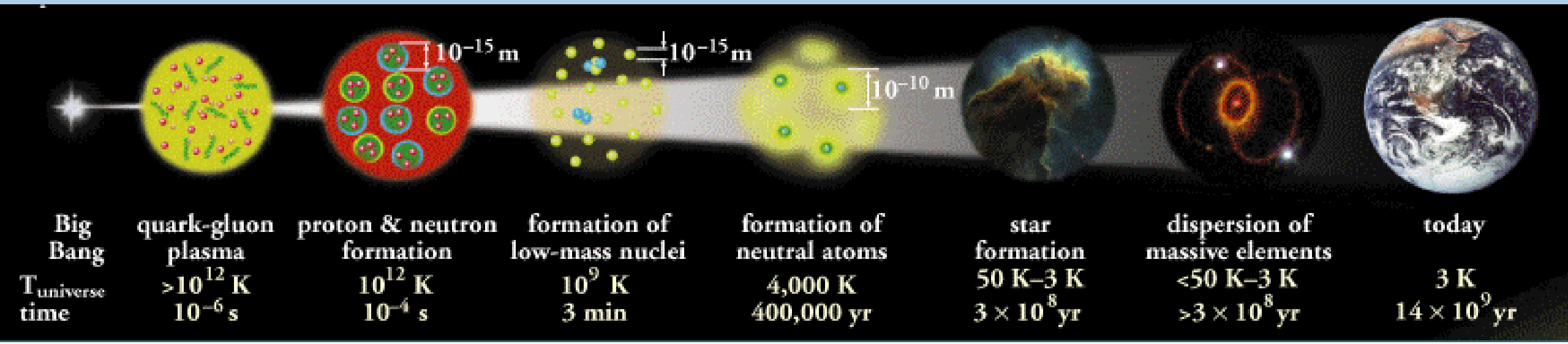
- Introducing modern high energy nuclear physics
  - our purposes (without Schröedinger or Einstein-like equations)
  - situations so far
- Advancing particle detectors and its electronics
  - we are in problematic phase (really)
    → let's say, challenging
  - from "triggered" data readout to "continuous readout & realtime processing" era
- Applying acceleration by FPGA to solve our problems
  - our detector project as one of the typical cases, but (probably) the most challenging case

Understanding nature is our final goal

- From very large structure to very small structure
- Describing all the forces
- Explain and predict all phenomena happening in the universe



Explaining from big bang till today



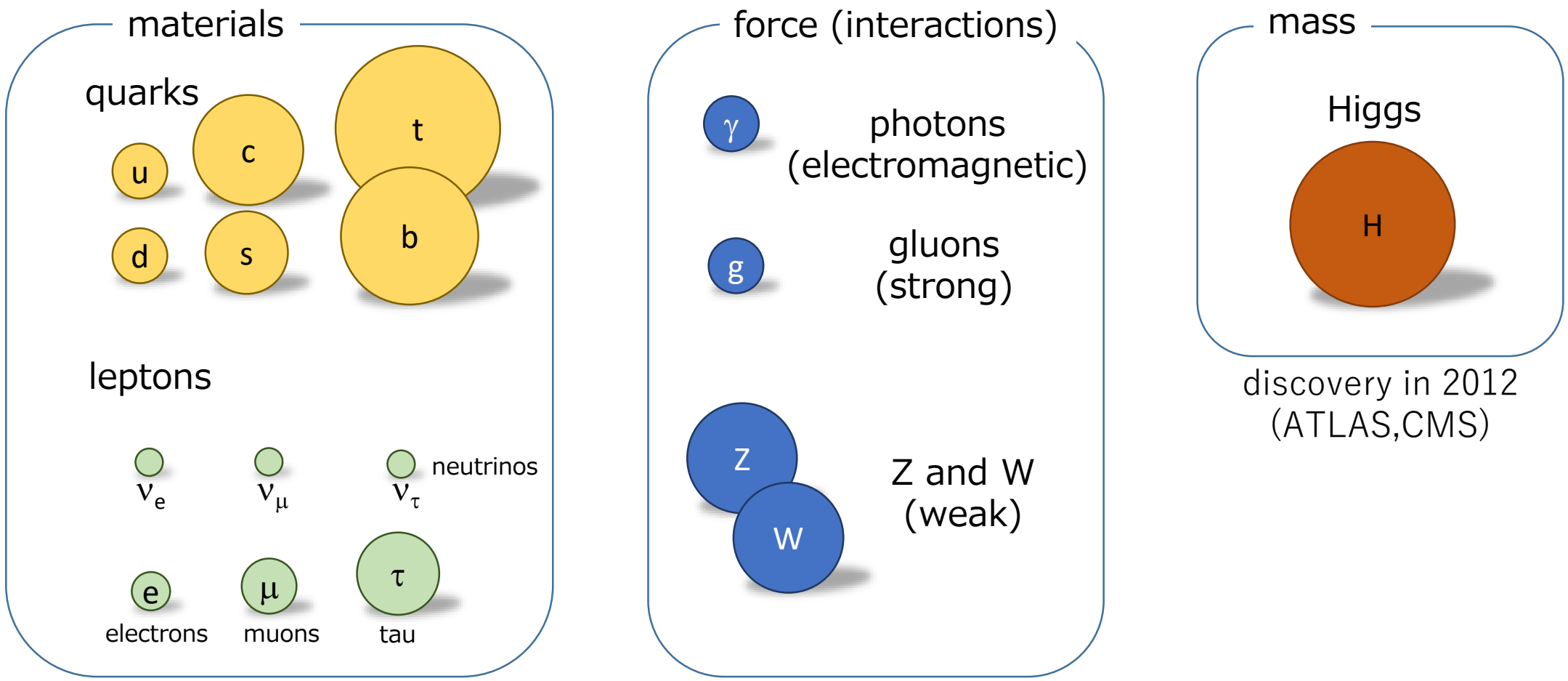
- Nowadays, we know (or think) big bang is the start of everything
- Necessary to understand all the stages universe may have passed
  - want to see closer to big-bang condition
    ⇓
  - higher energy density and temperature

⇒

Solution 1
observe far away (Hubble's law)
Solution 2
achieving high temperature condition in laboratory then observe it

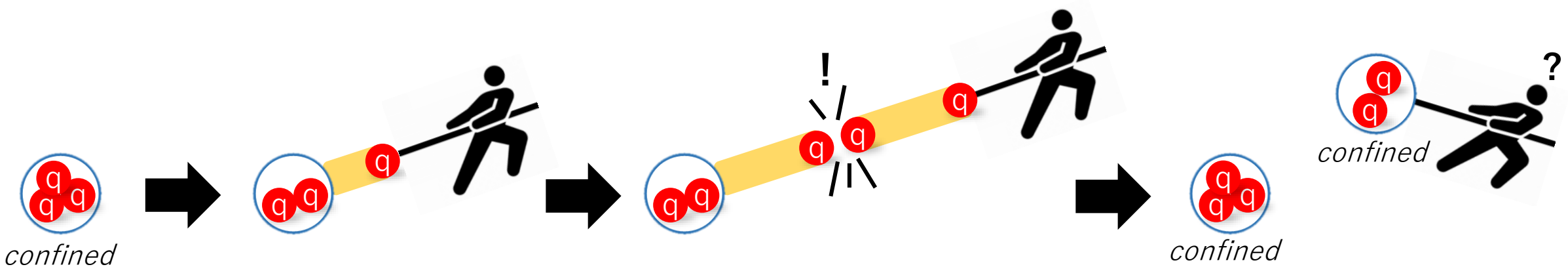
What actually was the early universe?

■ Before thinking that, we need to remember the elements of the universe



Quarks are frozen into nucleus

■ In the world (now), quarks are never observed alone but always confined into nucleus



- **this characteristic is called “confinement” and reason of that is not clear yet**
- **applied energy was converted to mass of new nucleus**

■ The situation is similar to **H_2O** atoms frozen (confined) into **ice**

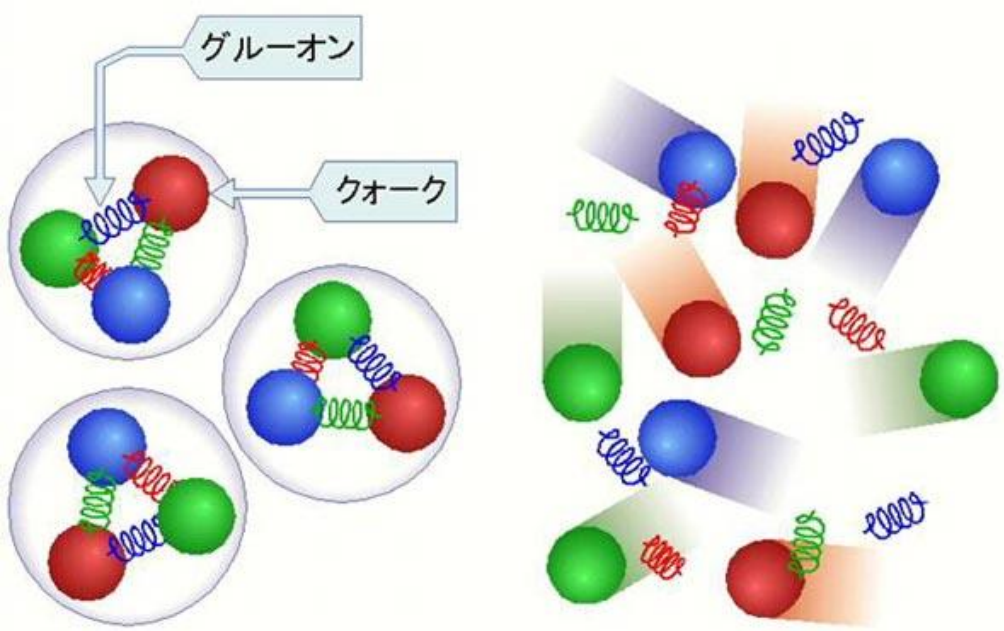
- heating it will produce freely moving H_2O atoms (water or vapor)
- similarly heating nucleus frees quarks?
  - what that is? water? or vapor?

What actually was the condition? (again)

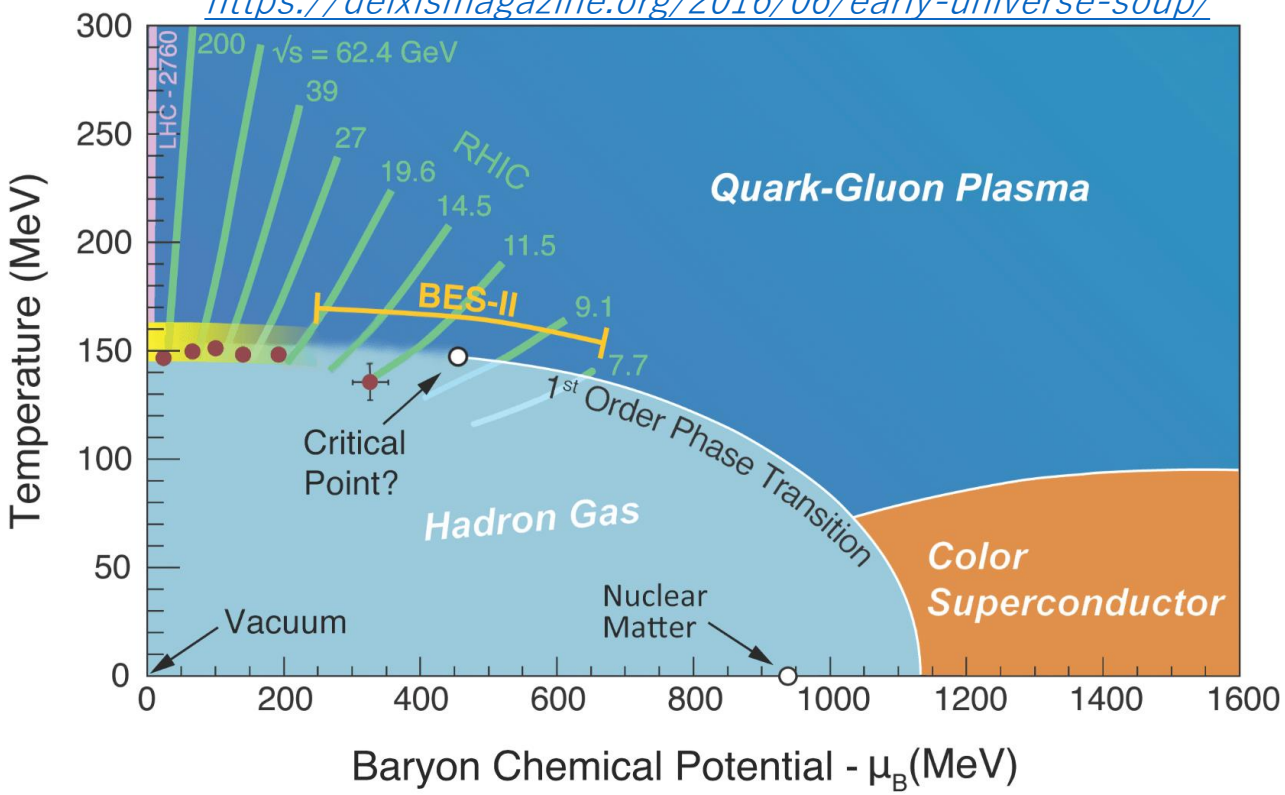
■ Answer: probably the universe was “melted” matter made of freely moving quarks

i.e. **quark gluon plasma (QGP)**

- analogous to electromagnetic plasma (freely moving nuclei and electrons)
- not governed by electromagnetic but by “strong” interactions
- **$T > 150$ MeV $\simeq 1.5 \times 10^{12}$ K**
  (laser nuclear fusion reactor $\leq 2 \times 10^{7}$ K)

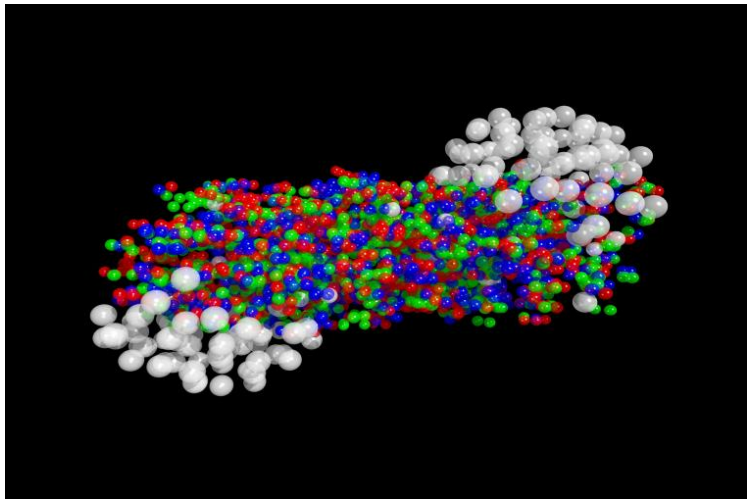


https://deixismagazine.org/2016/06/early-universe-soup/



How to reach that condition?

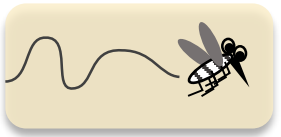
- **nucleus+nucleus (Pb+Pb) collision may concentrate 0.16 mJ into $(1\text{x}10^{-12}\ \text{cm})^3$ cube**
  - energy density: $1.6\text{x}10^{32}\ \text{J/cm}^3$
  - even Sun ($4\text{x}10^{26}$ W) takes 4 days to fill $(1\ \text{cm})^3$ cube with this energy density
  - lifetime = 10^{-23} s
  - immediately evaporate and emits “normal” particles



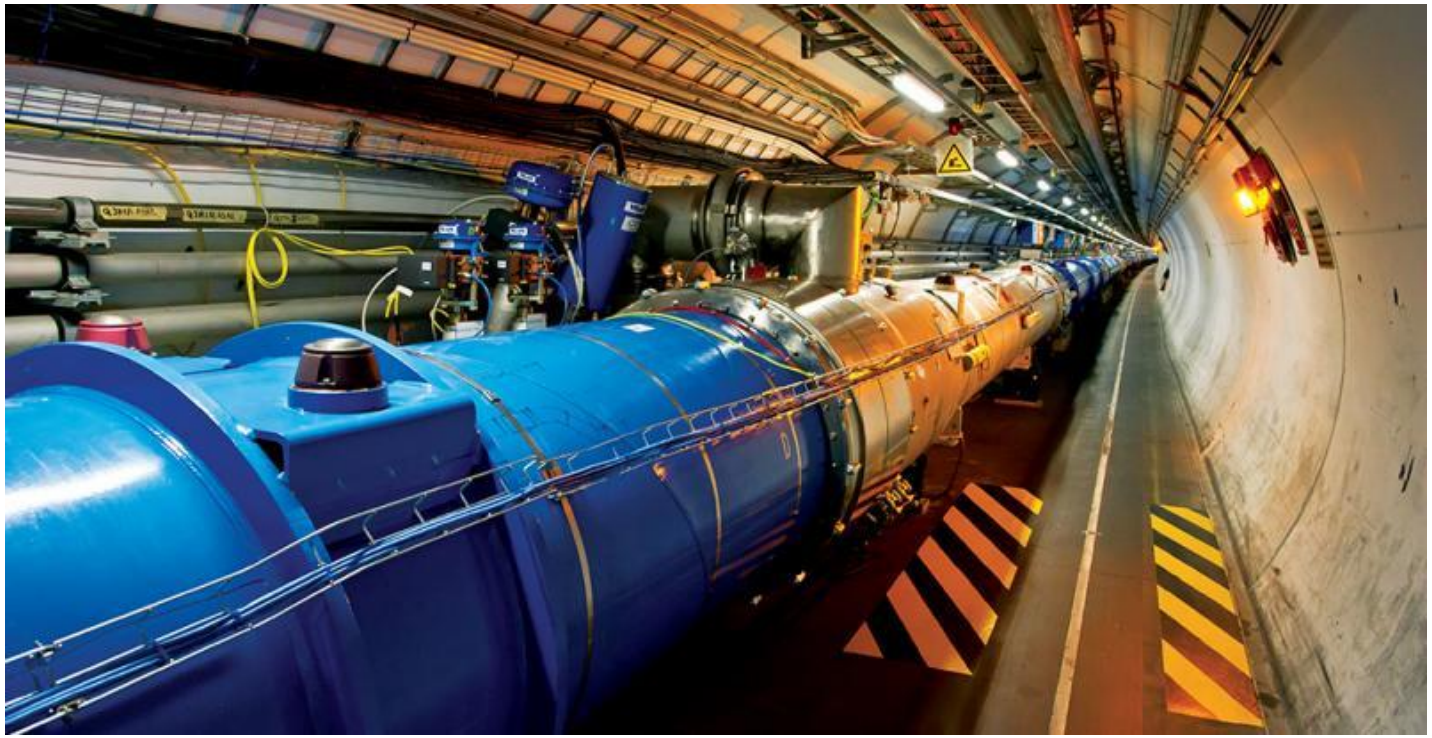
- Reconstruct the event by measuring all fragmentations (particles) and analyzing it
  - example: photon information tells achieved temperature
  - QGP at $5\text{x}10^{12}$ K is indeed produced
    (we confirmed already, and making a lot)

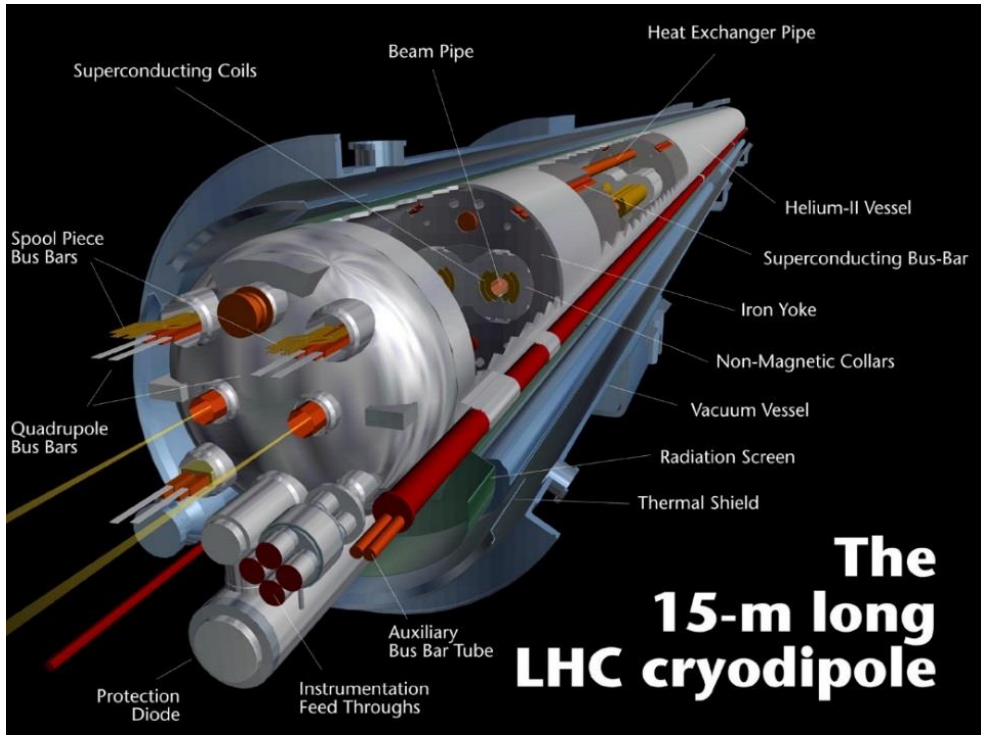
LHC(Large Hadron Collider)

- Accelerates protons up to 7 TeV, then collide → center of mass energy of 14 TeV
  - corresponds to kinetic energy of a 2 mg insect at 1 m/s
  - 99.999999% of speed of light = c – 3 m/s

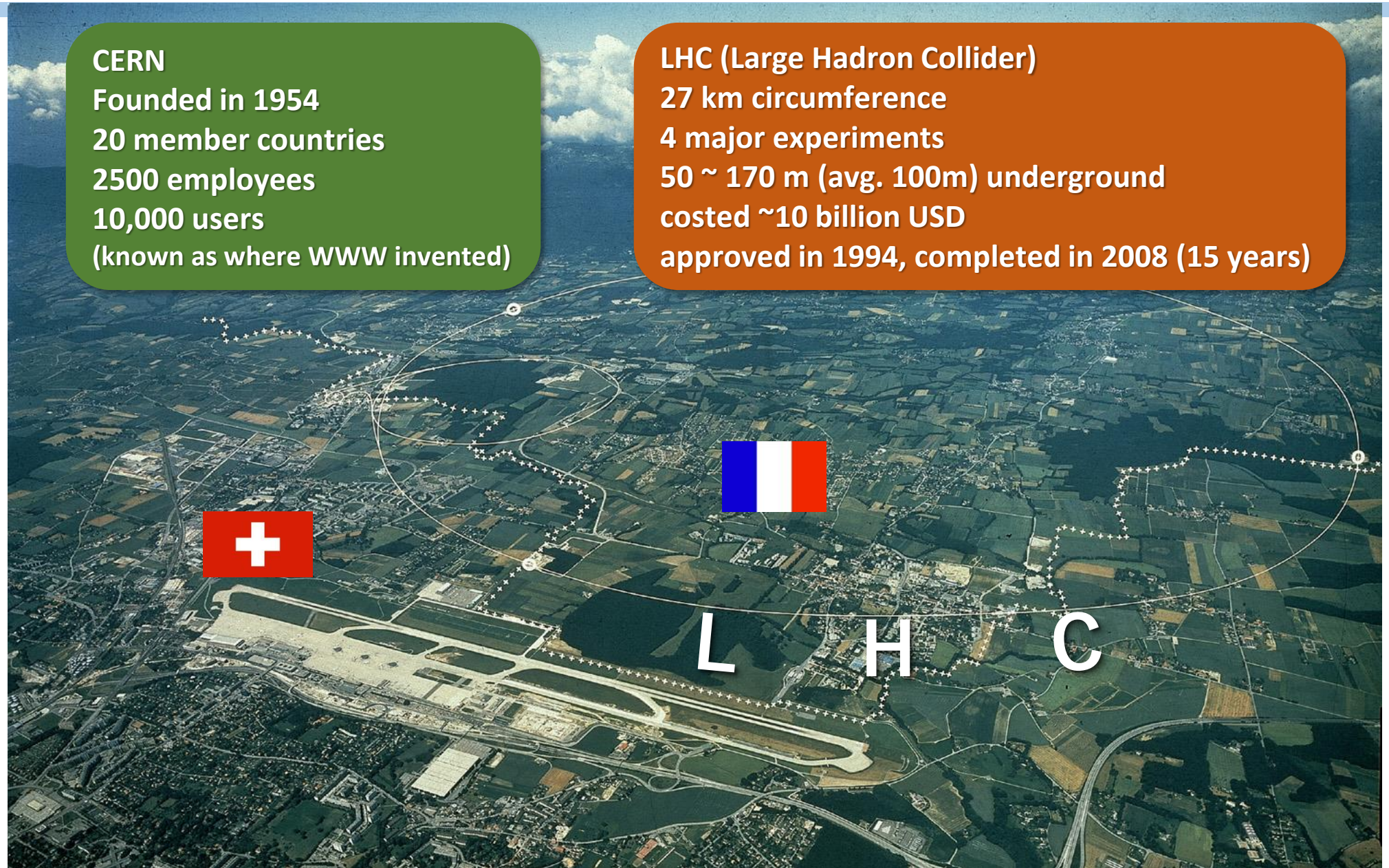


- 1232 superconducting magnet with liquid helium cooling system (1.9 K)

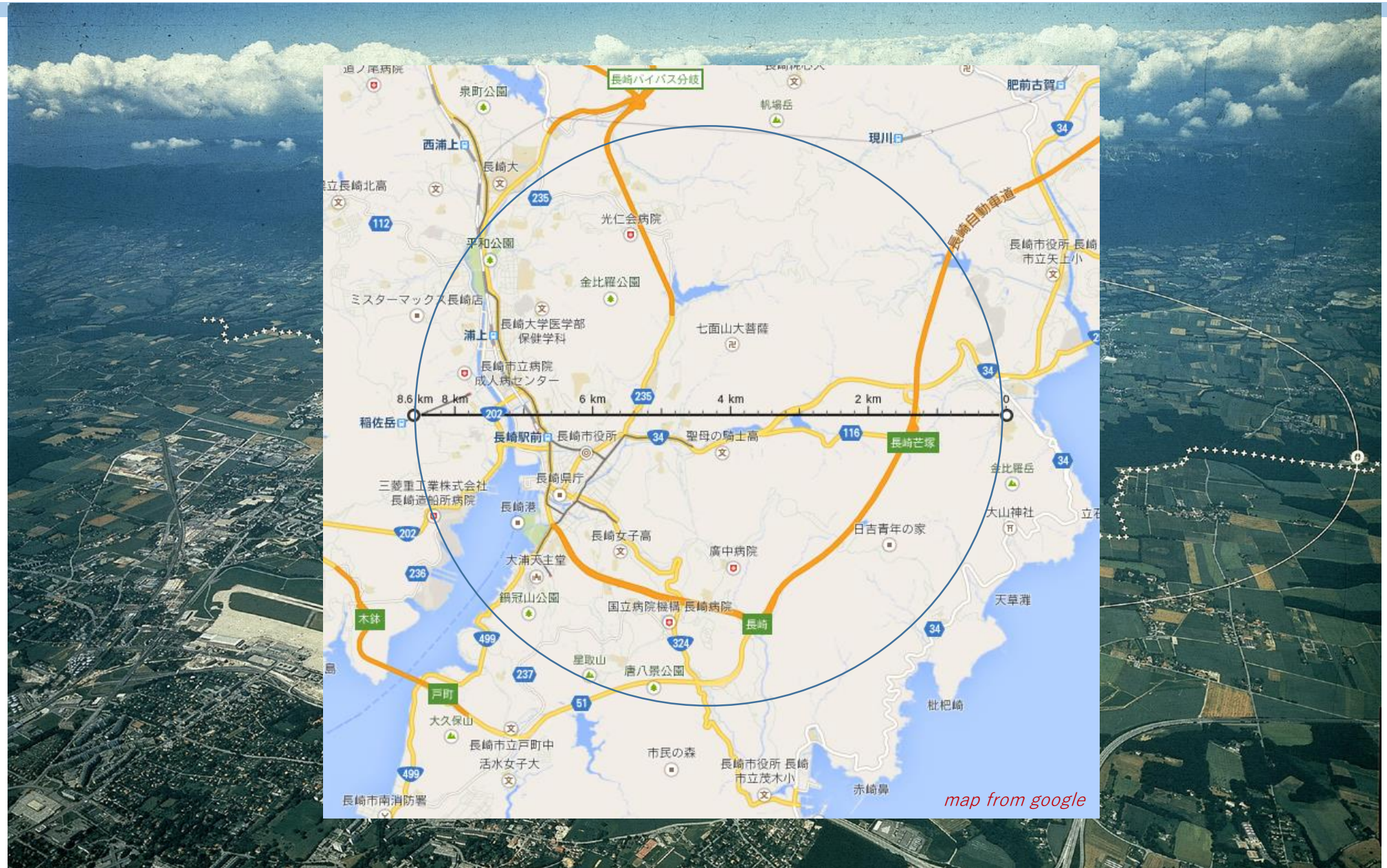




CERN accelerator complex



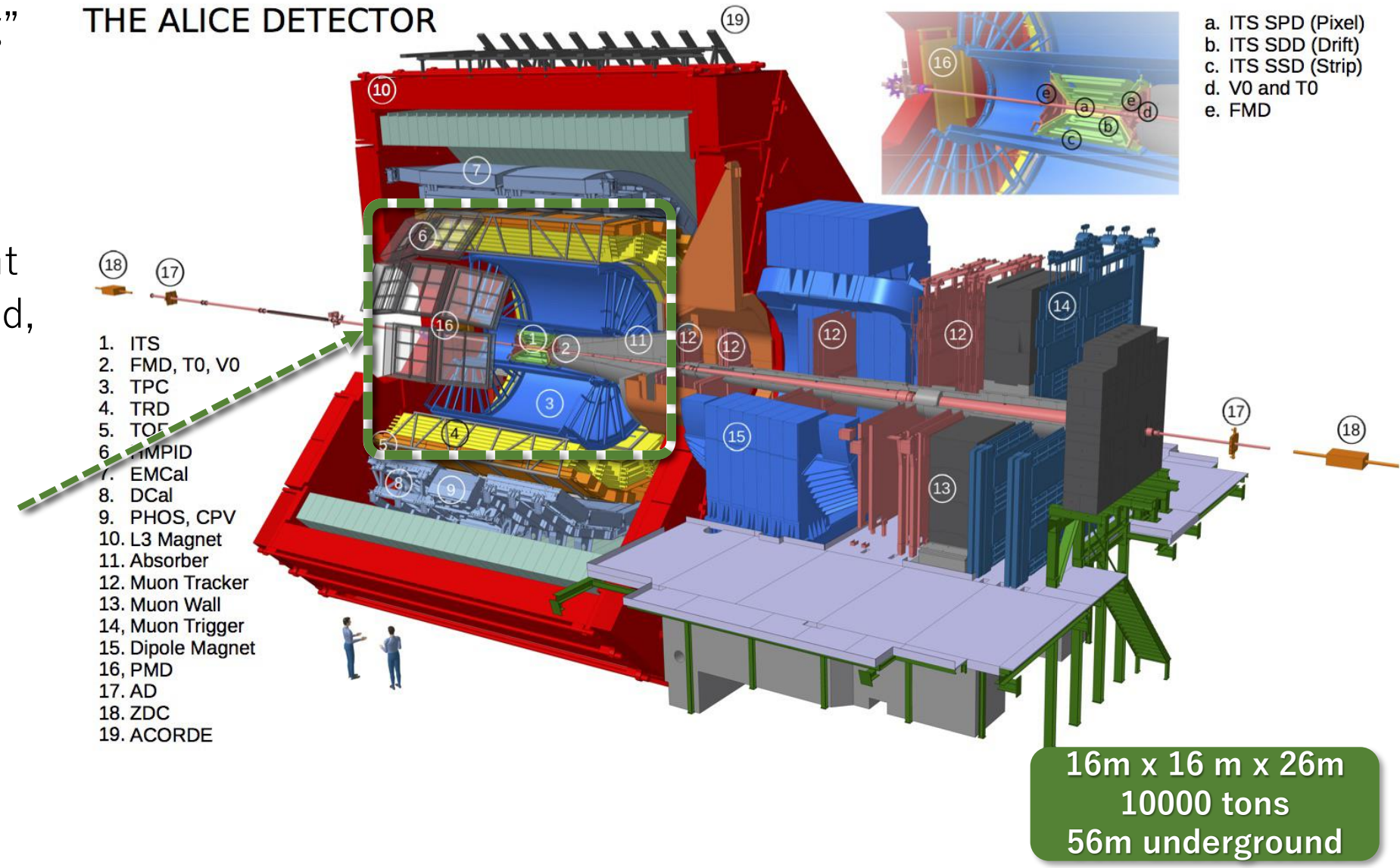
CERN accelerator complex



CERN accelerator complex

LHC ALICE Experiment

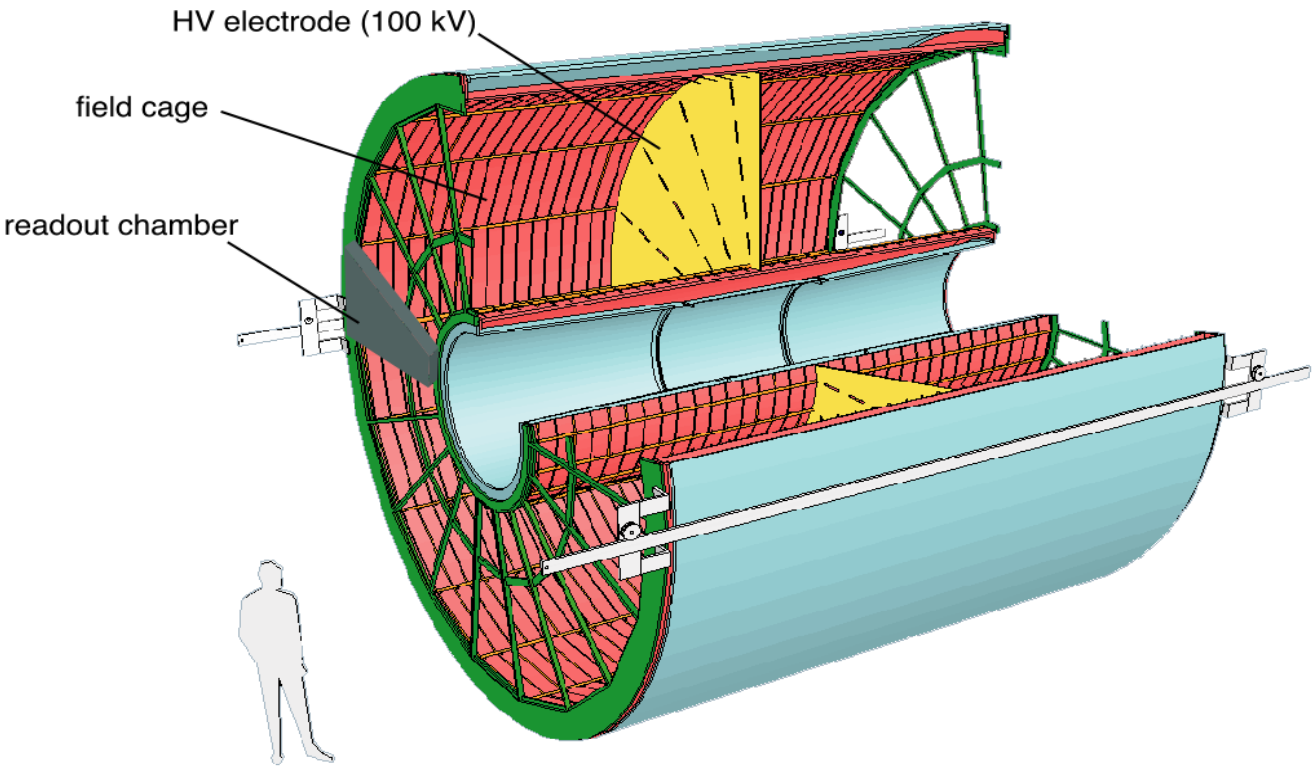
- We measure “everything” come from QGP to find its characteristics
- Different detector technologies for different type of particles (charged, not charged, fast/slow)
- Today, I pickup one of the detectors with “largest” data volume thus the most challenging detector

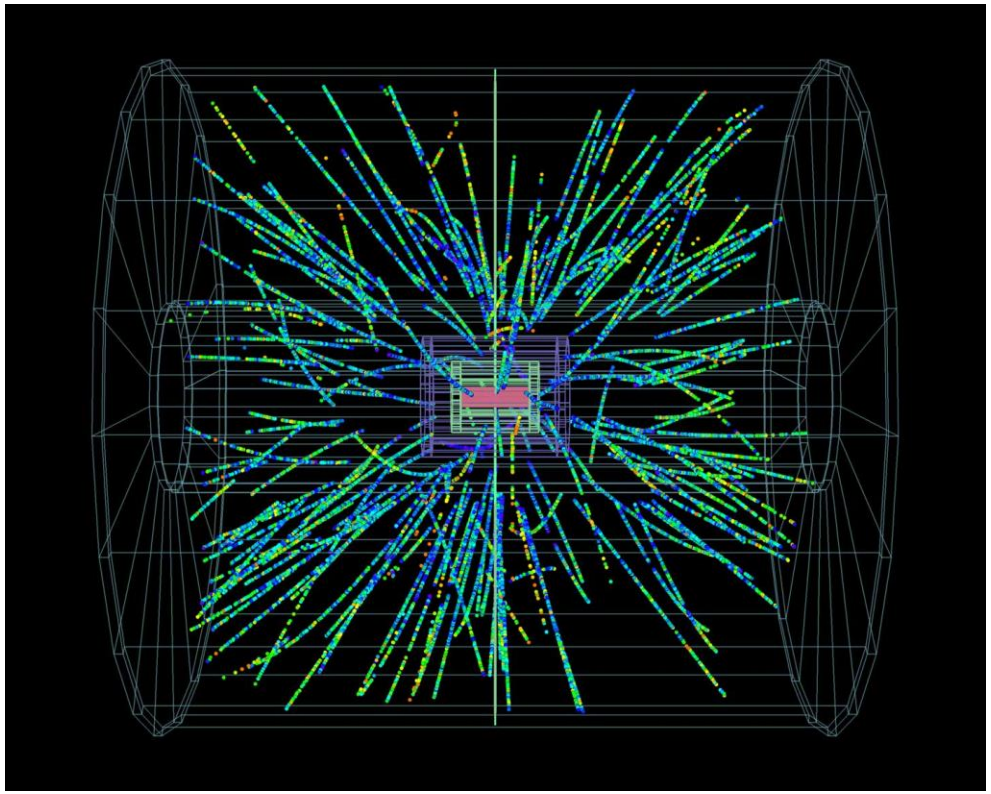


TPC of ALICE

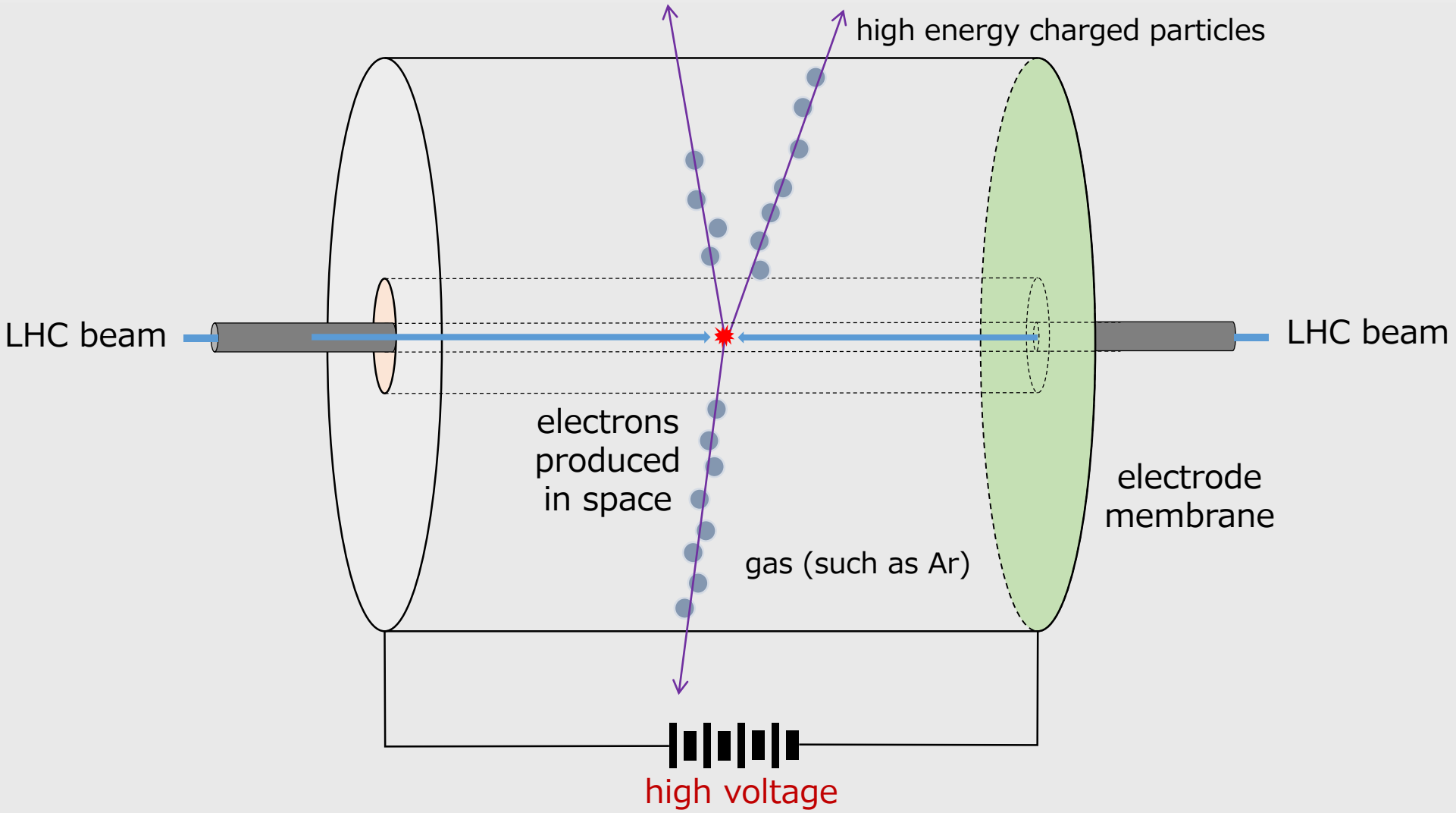
■ TPC: Time Projection Chamber … a tracking device

- 3D memory that memorize charged particle trajectories
- rare gas as the memory medium (data is stored in gas volume, and read like shift register)
- 600M memory pixels with frame rate of 2000 Hz ← too slow, we want 50,000 Hz now

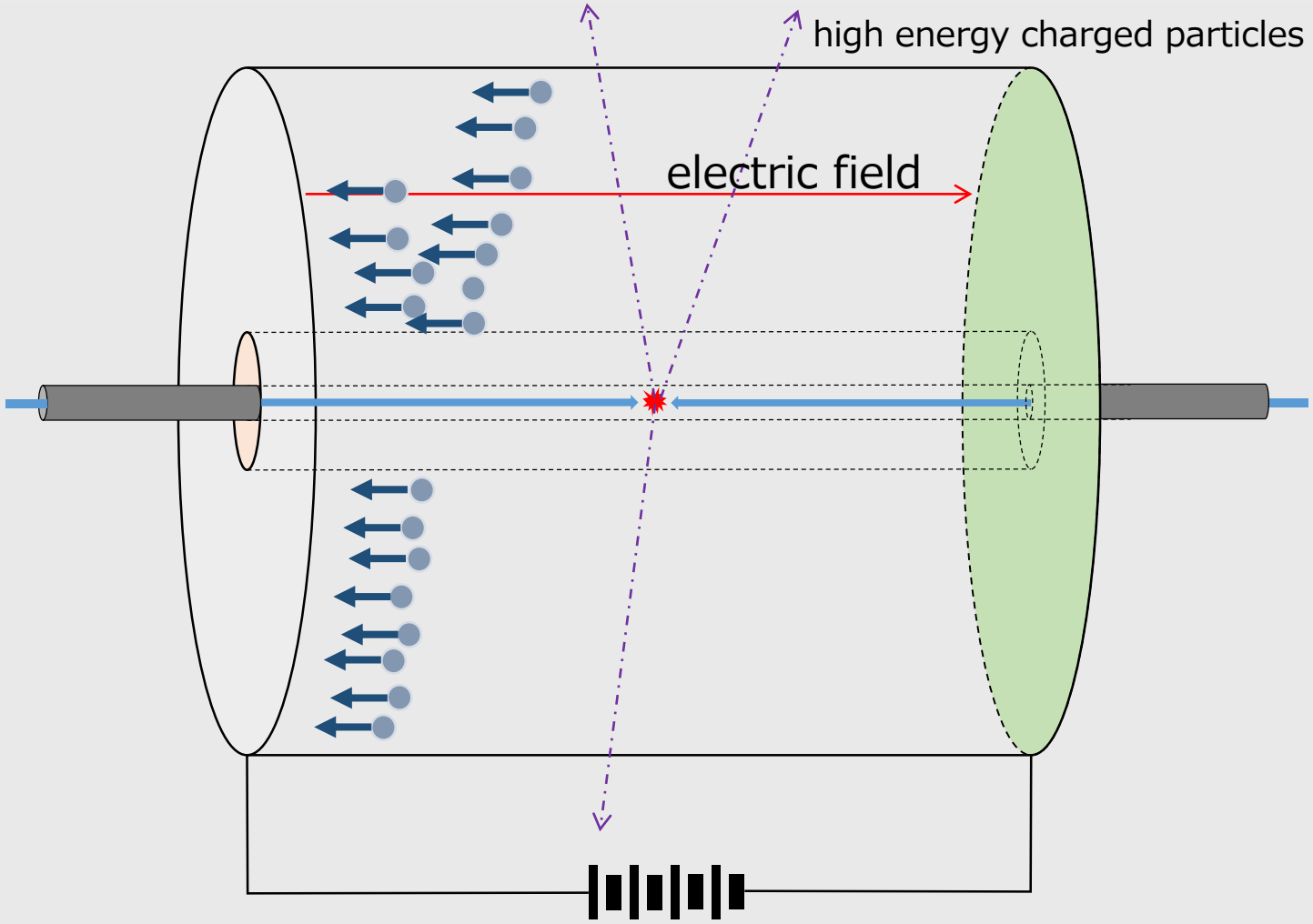




TPC: 3D snapshot memory

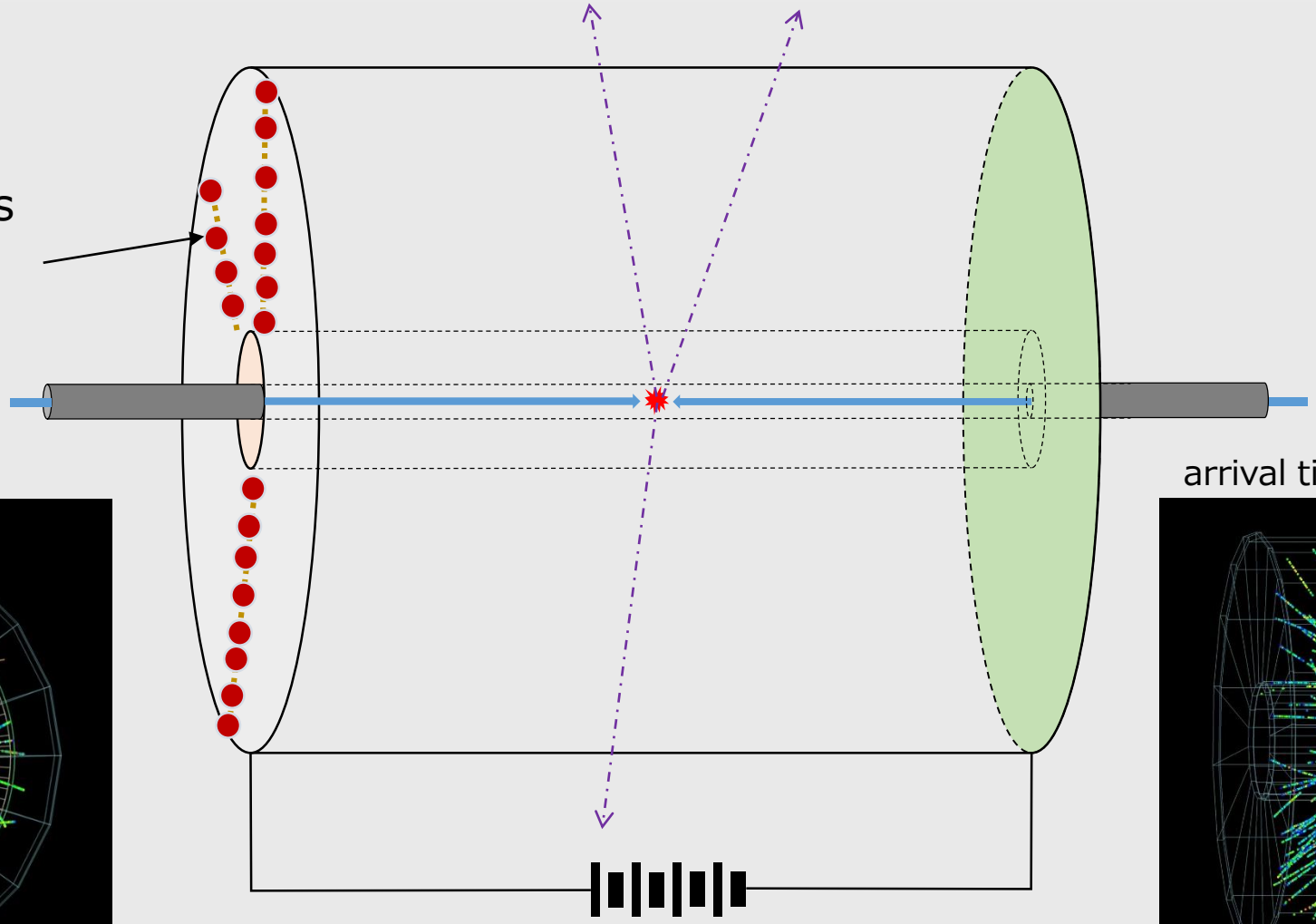


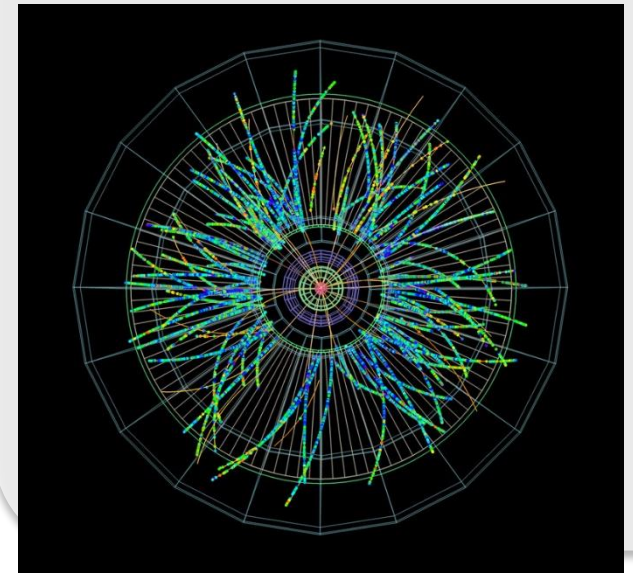
TPC: 3D memory readout (shift register)



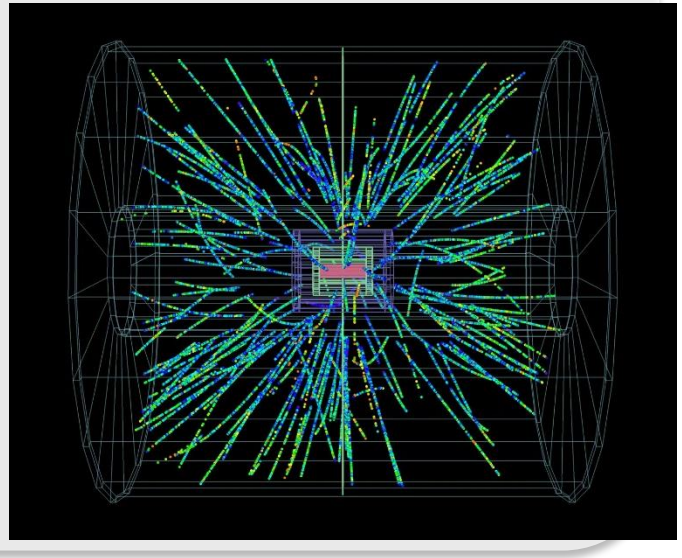
TPC: memory contents converted to signal

electron detection part with electronics (convert analog memory output to digital data)

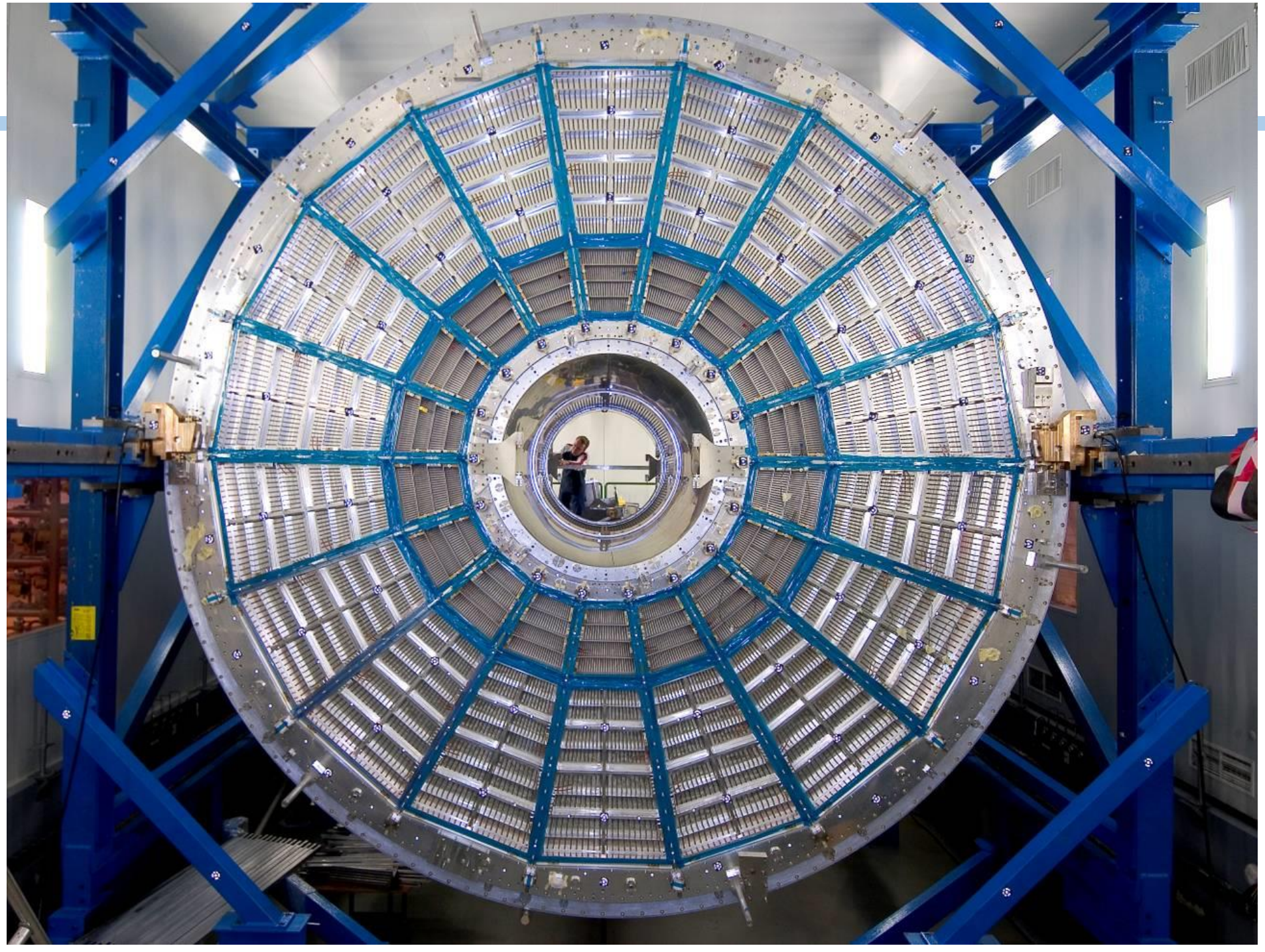




arrival time → horizontal position

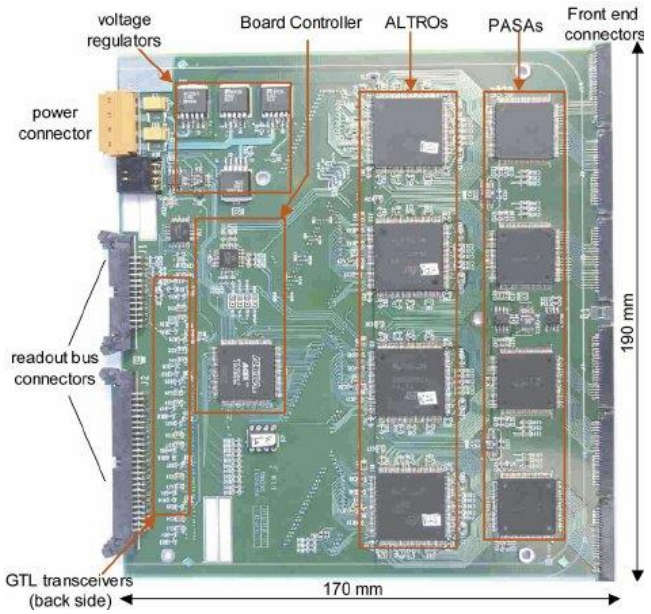


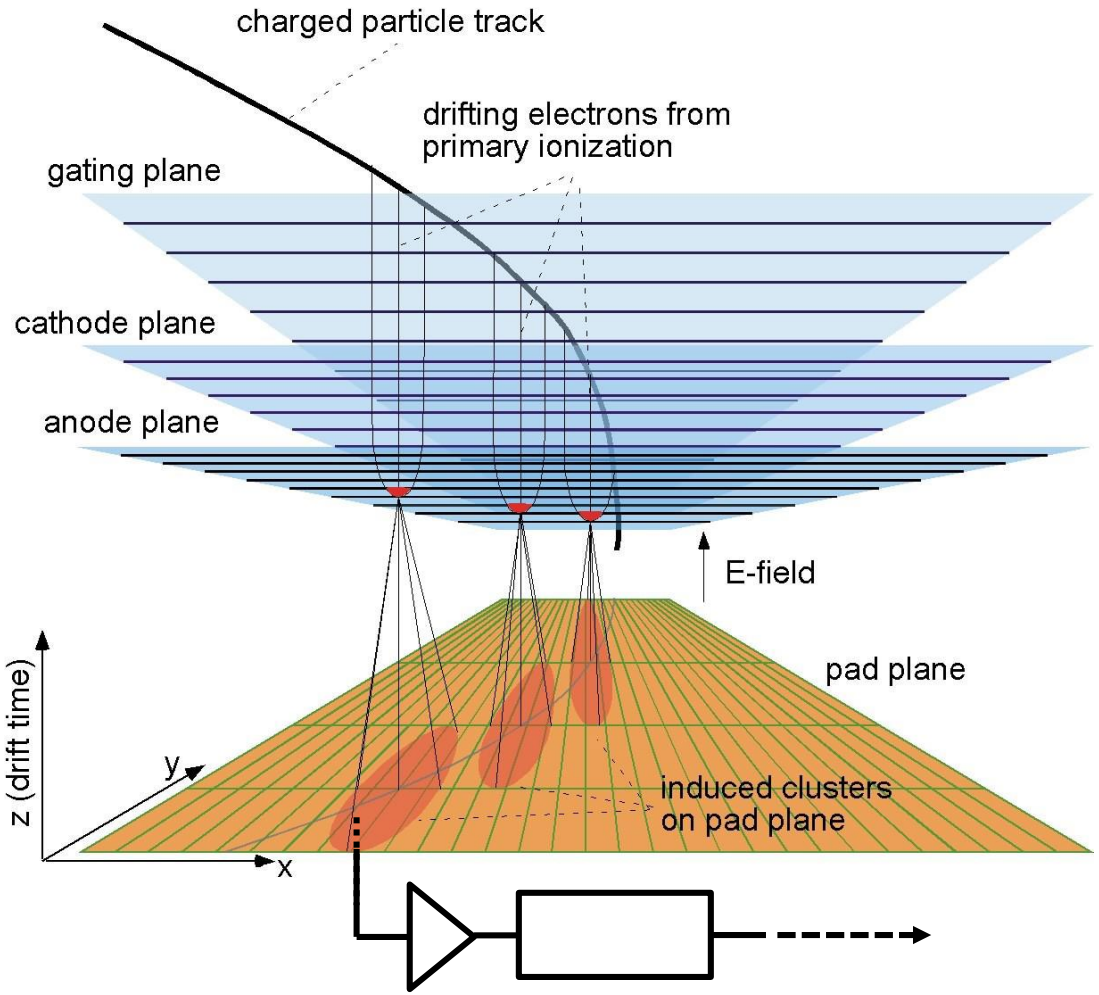
TPC



Conversion from electrons to digital data

- Electrons drift towards an “amplification region”
- High voltage wire (anode), quickly accelerate electrons, produce more ion - electron pairs (amplification)
- Electrodes (pad) below detect induced current
- We prepared 570k ADCs and DSP (ASIC)





digitization, DSP, then to storage

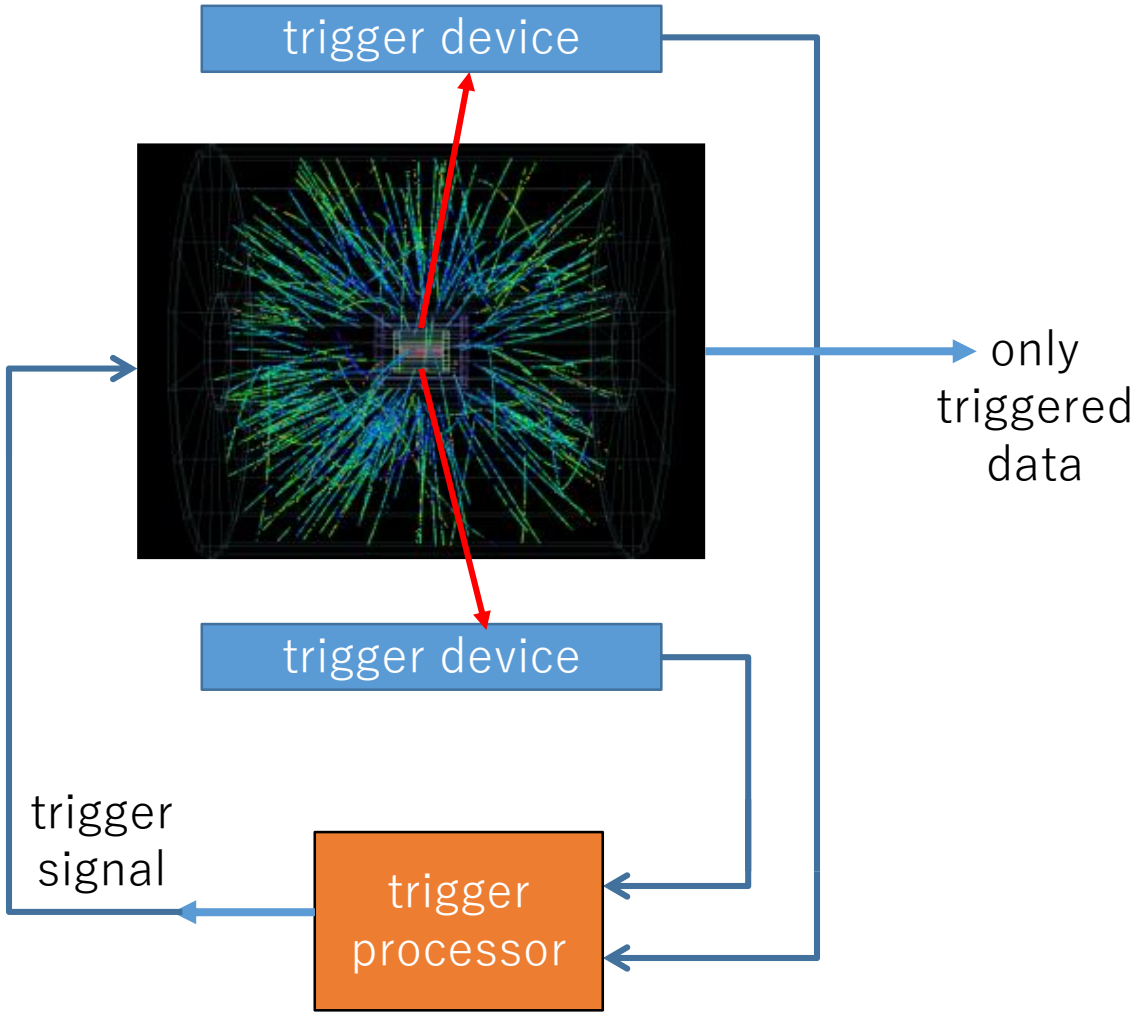
Readout performance of present system

- Reading 570k channels ADC in parallel
  - 10 MHz, 10 bits each
  - will create 57 Tbps (7 TB/s) data, continuously (required 10 years ago)
- On-detector data reduction and compression was mandatory; we implemented:
  - trigger
  - zero suppression
- entire readout process (from event time to completing data readout) takes too much time
  - electron signal drift ~100 μs
  - cleaning positive ions ~400 μs
  - limits frame rate at 2 kHz

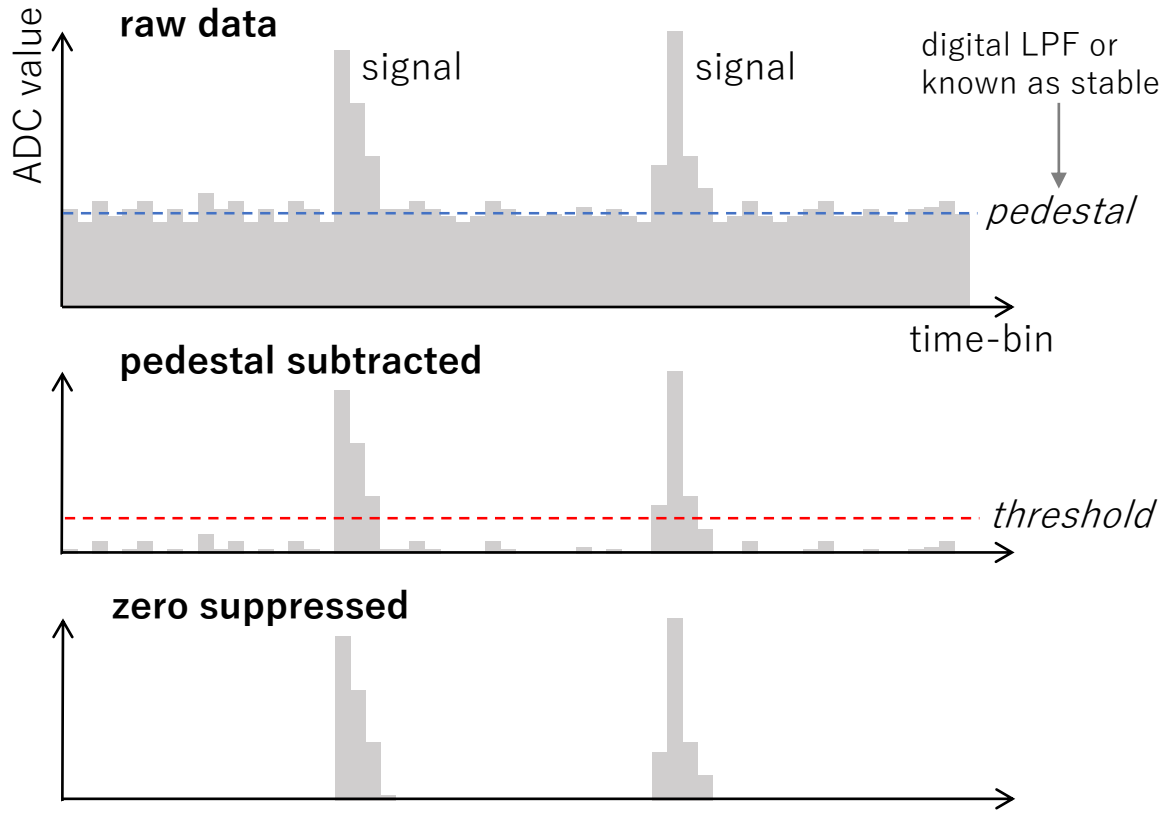
  → about 1 GB/s of average data rate (was realistic and doable 10 years ago)

Trigger and zero suppression

trigger suppresses dramatically data rate
if interesting event is occurring not very often



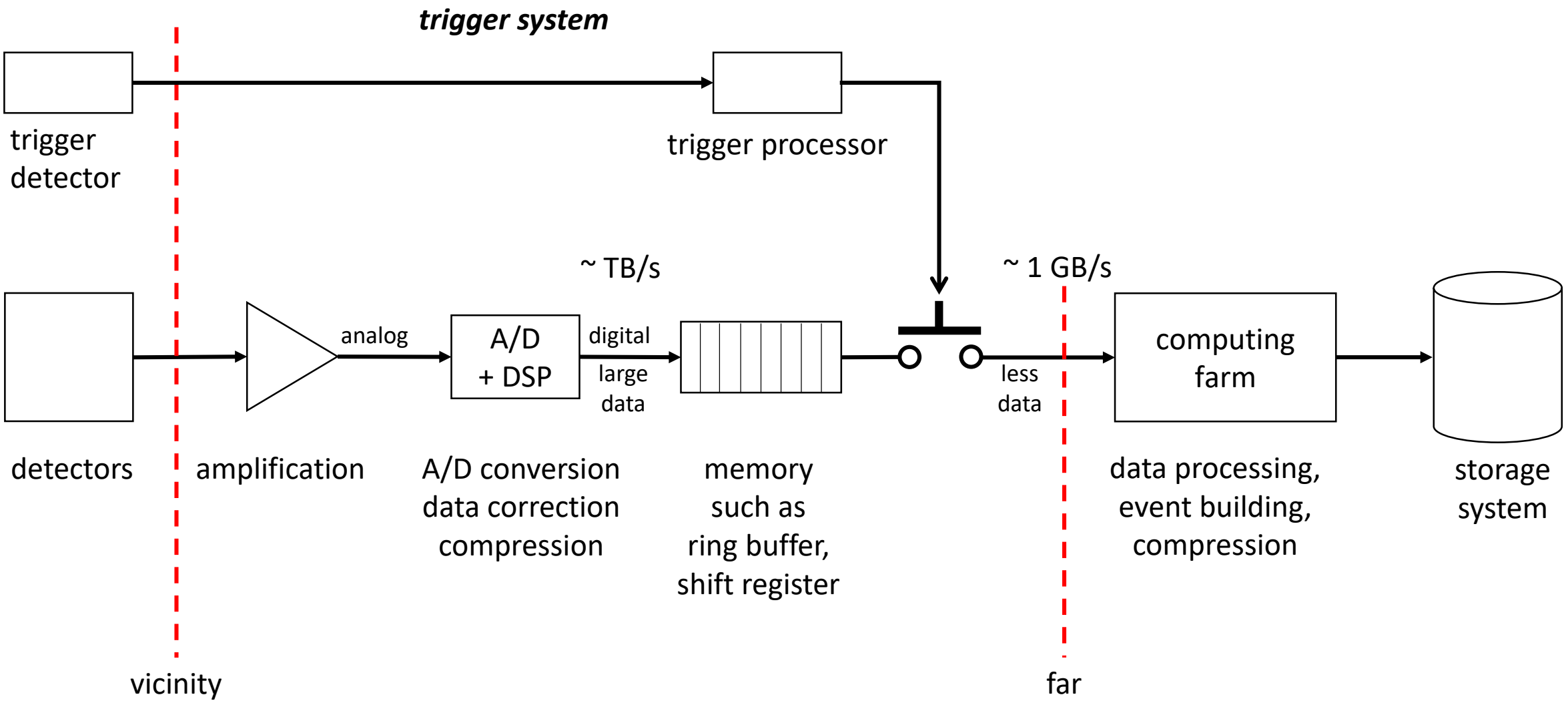
zero suppression is efficient if ADC has
signal only sometimes



`#zeros,t1,t2,t3,t4,#zeros,t1,t2,t3,t4,#zeros`

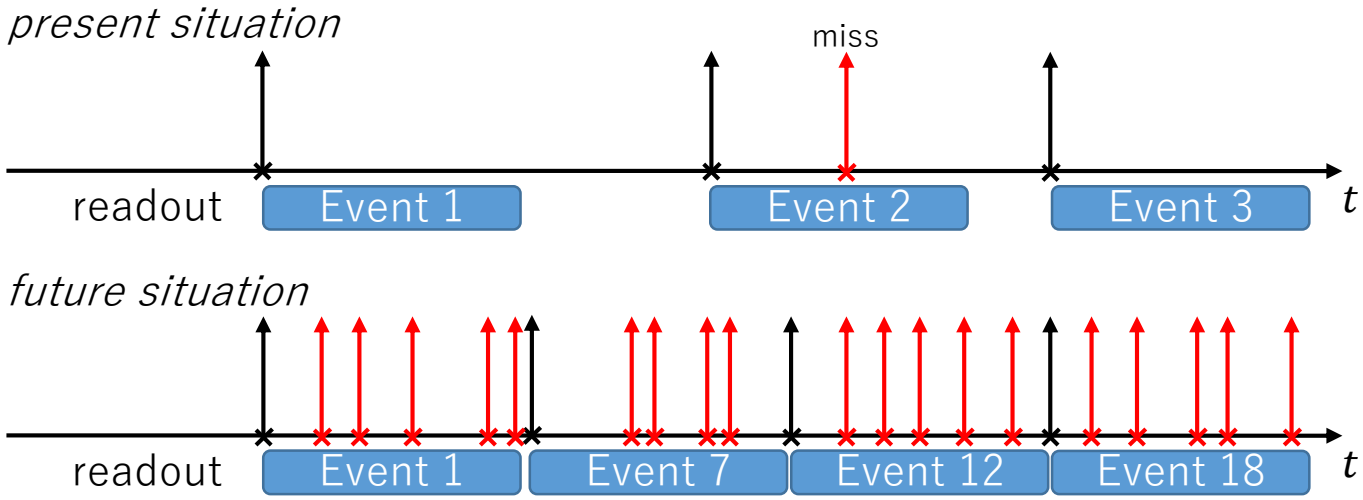
+ whatever better coding

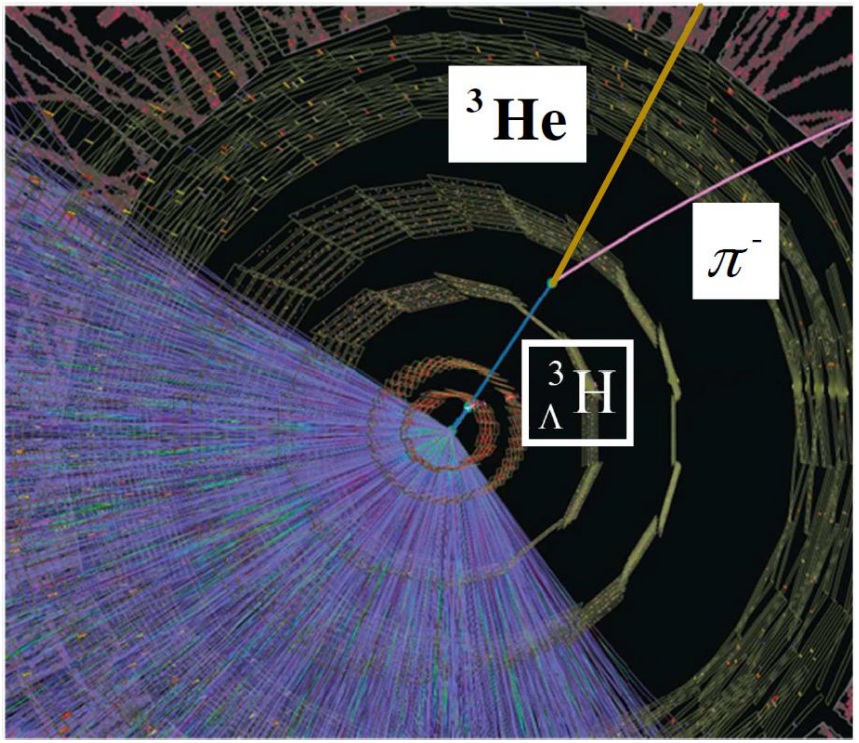
Detector signal processing in last 10 years



We reached limitations with LHC upgrade

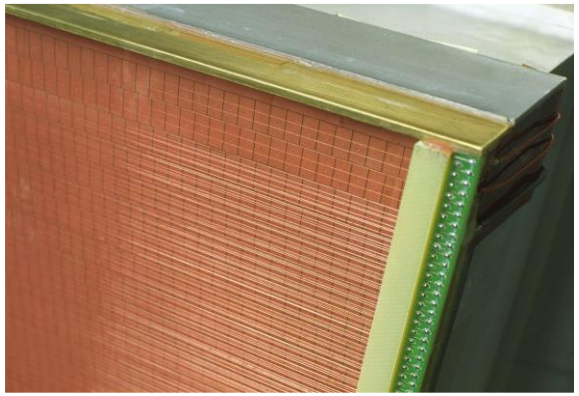
- Physics requires more data (100 times faster) ⋯ with LHC and detector upgrade
- Trigger & zero suppression do work any more because of too many particles
  - if you try to find “interesting event”, all events are interesting
- Event rate too high (50 kHz)
  - our wire-amplification detector would not survive (breaks)
  - we call this situation “99% trigger dead time”

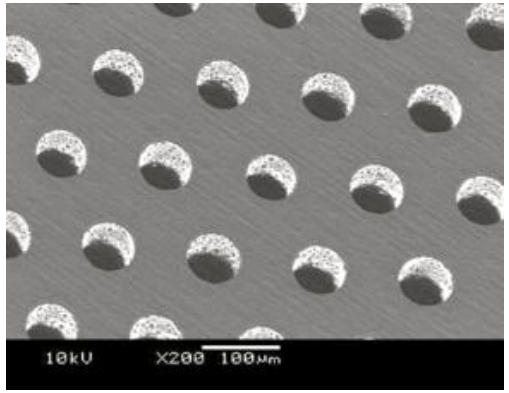


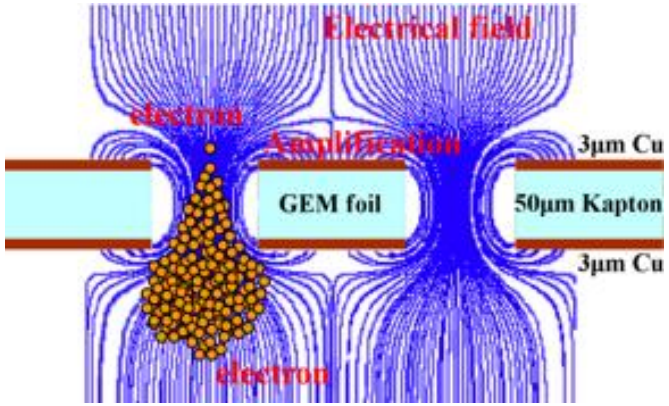


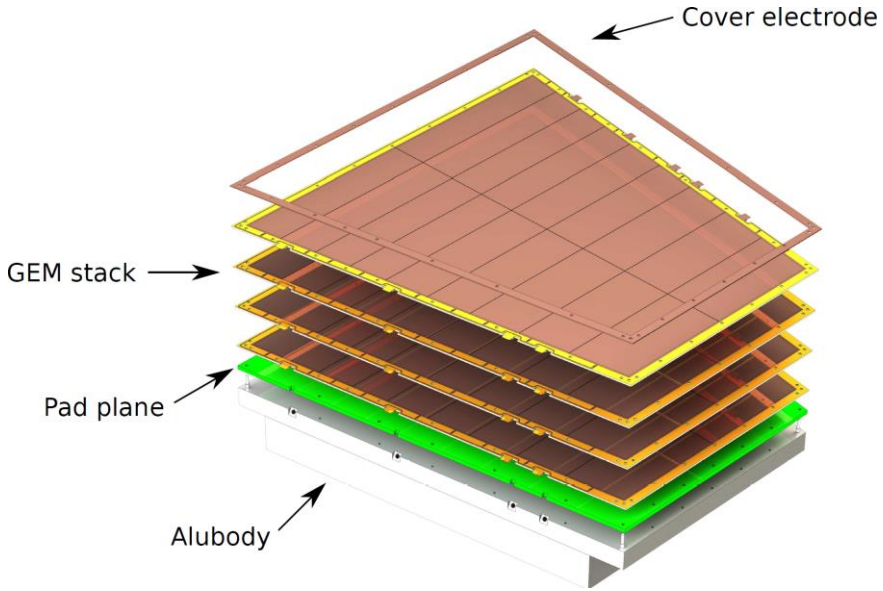
From MWPC to Micro Pattern Gas Detector

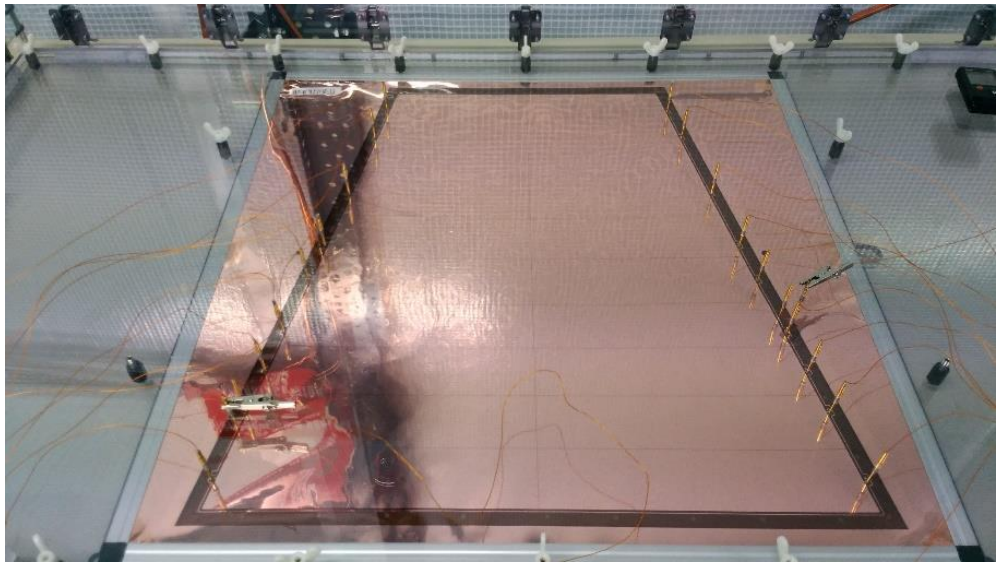
■ We need to refurbish our detector (especially electron amplification part; bottleneck) with new system: **GEM** (Gas Electron Multiplication) made of micro pattern technology









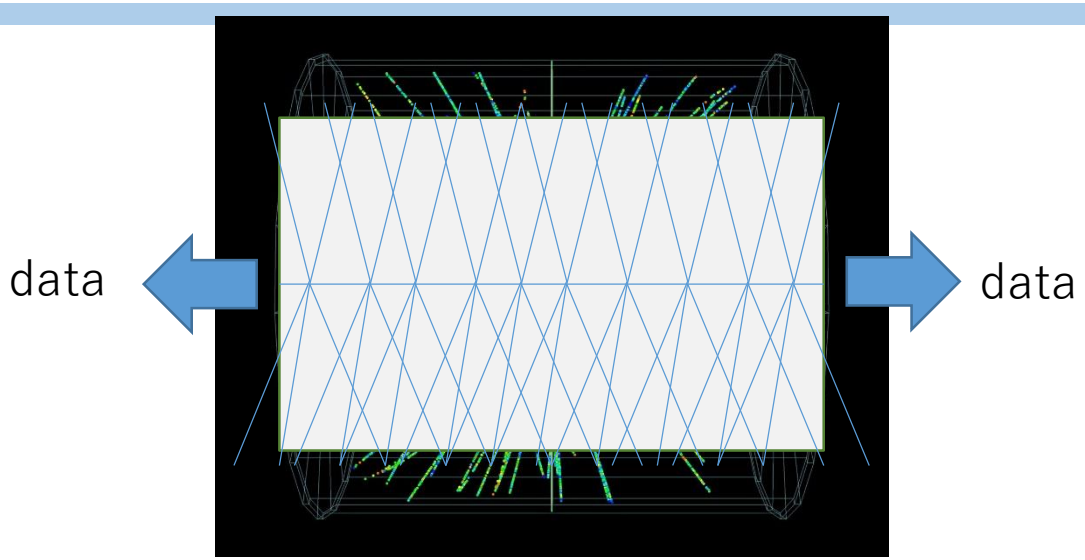


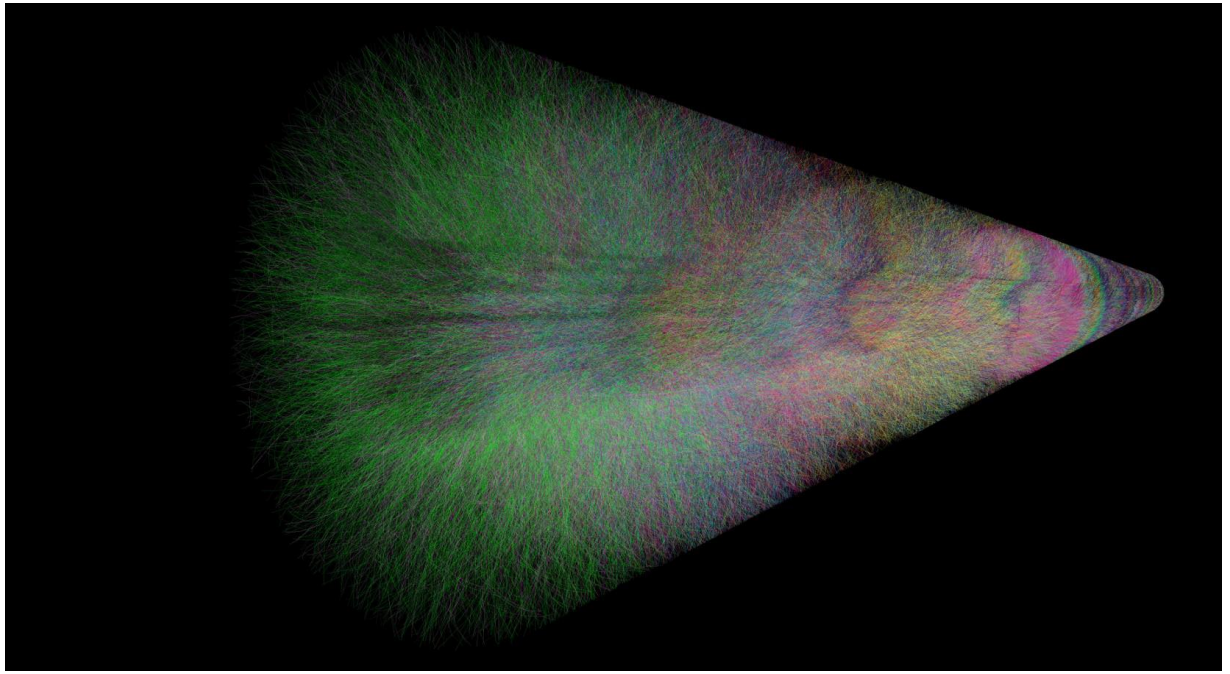
So, we got breakthrough but

- GEM removes major detector limitations
- But still electron drift takes time (100 μs)
  - average 5 events in one memory snapshot

community decision (new challenge)

- Read ALL ADC data continuously without trigger (3.5 TB/s)
- Accept overlapping multiple events in data
- Do much more efficient data reduction
  - a factor of 40 → 100 GB/s
  - traditional zero suppression (a factor of 3-4) not enough





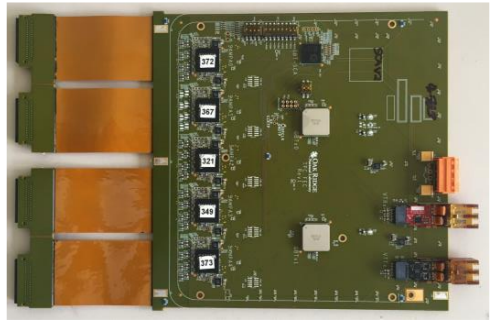
ALICE TPC Data Acquisition System

detector
GEM+Pad

analog
(560k ch.)
short wire

Front-End Electronics

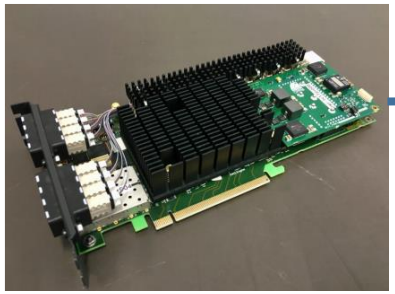
amp → A/D 5 MHz 10 bit → ~~DSP + Zero Suppression~~ → GBT rad-hard optical TX



optical fibers 4 Gbps x 7000 → 28 Tbps (3.5 TB/s), distance ~100 m

underground ↑

surface ↓



FLP(First Level Processor) node

FPGA Arria 10 (GTX1150, 10AX115S3F45E2SG)

x20
80 Gbps
(10 GB/s)
1600 ADCs

GBT x20 → sort → filtering DSPs → clustering → format

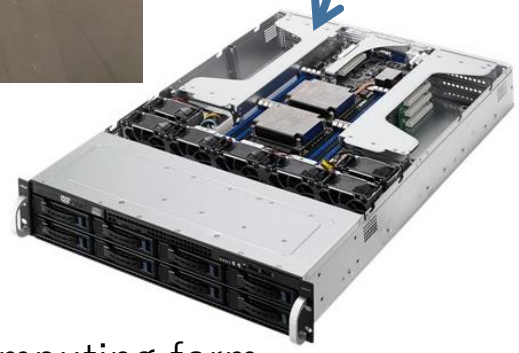
16 Gbps

post-clustering
formatting
loss-less compression
x86 CPU

PCI Express Gen3

network IF → to computing farm

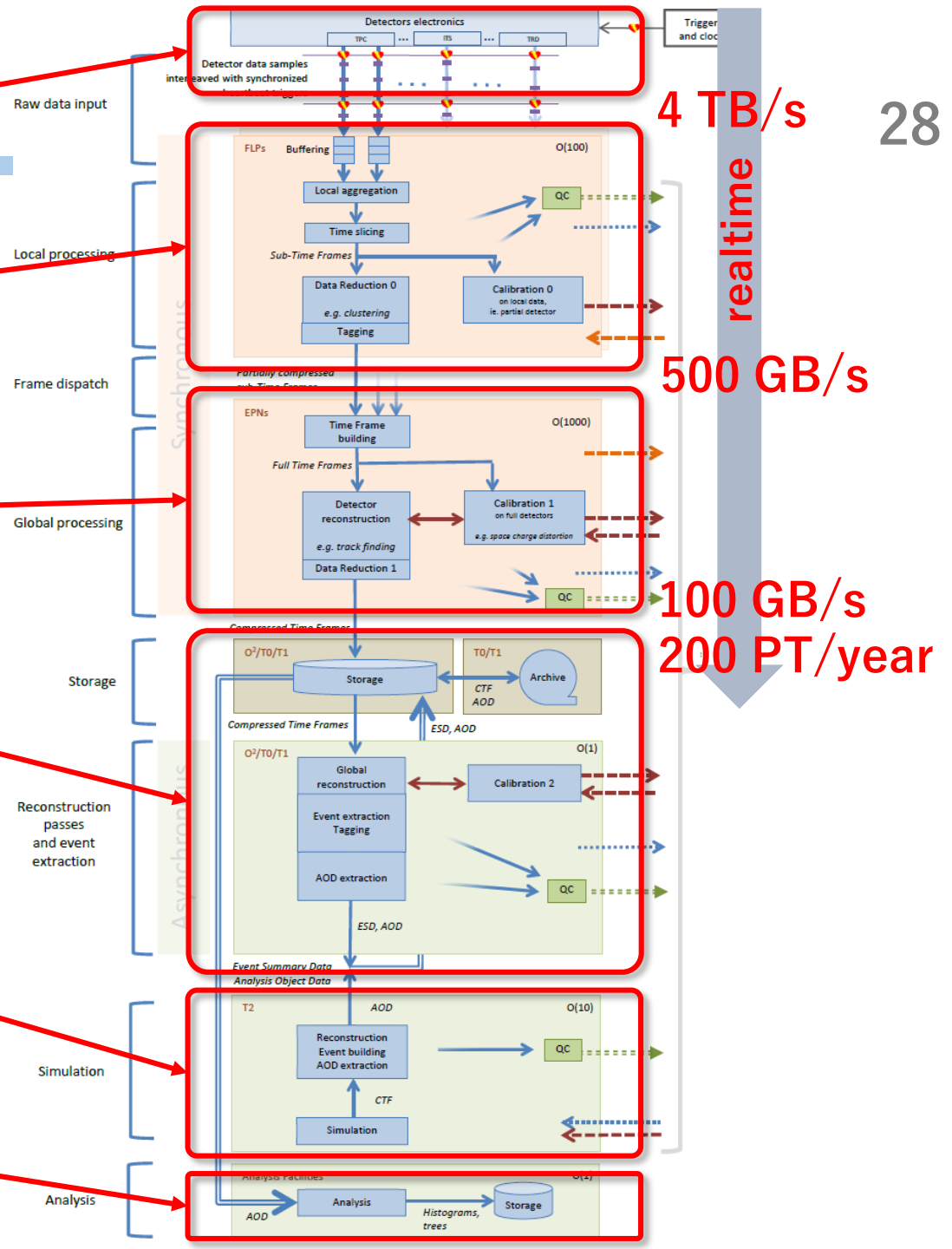
10 Gbps/node, 3.6 Tbps total



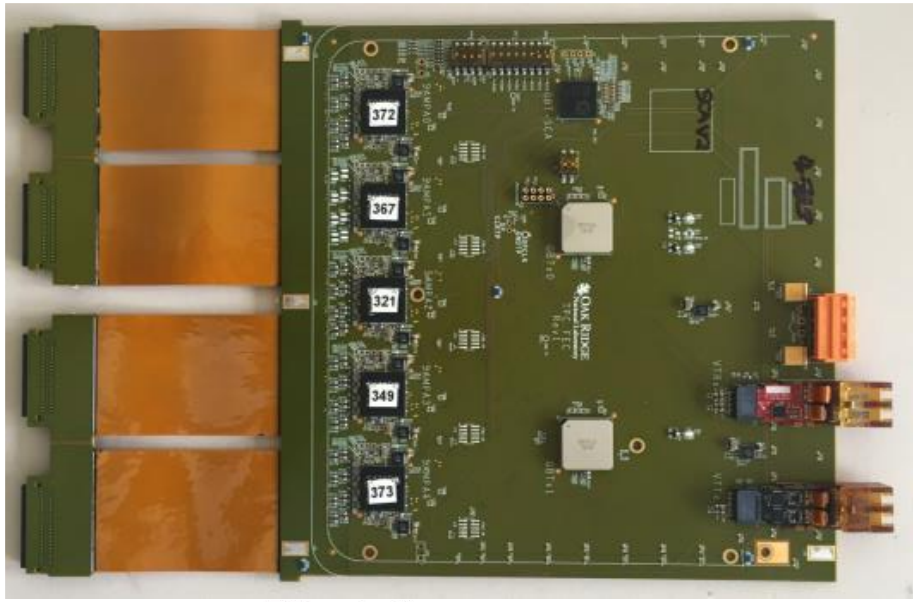
to other nodes
(total **360 nodes**)

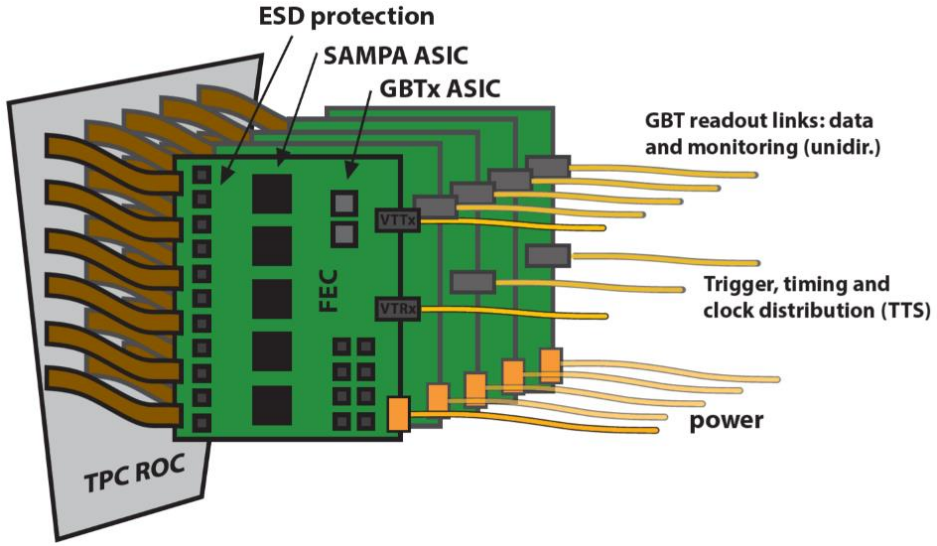
Our system overview

- Detectors
- FLP (First Level Processor) cluster
  - **Arria10 FPGA + CPU**
  - **~400 nodes**
- EPN (Event Processing Node) cluster
  - CPU+GPU
  - ~1000 nodes
- Storage at CERN and offline processing nodes
  - ~ 1 EB storage
  - ~1000 CPU nodes
- Simulation facilities not at CERN
  - ~10000 CPU(+GPU+FPGA?0 nodes
  - distributed over world (grid system)
- Physics analysis facility
  - ~1000 CPUs



TPC front-end readout





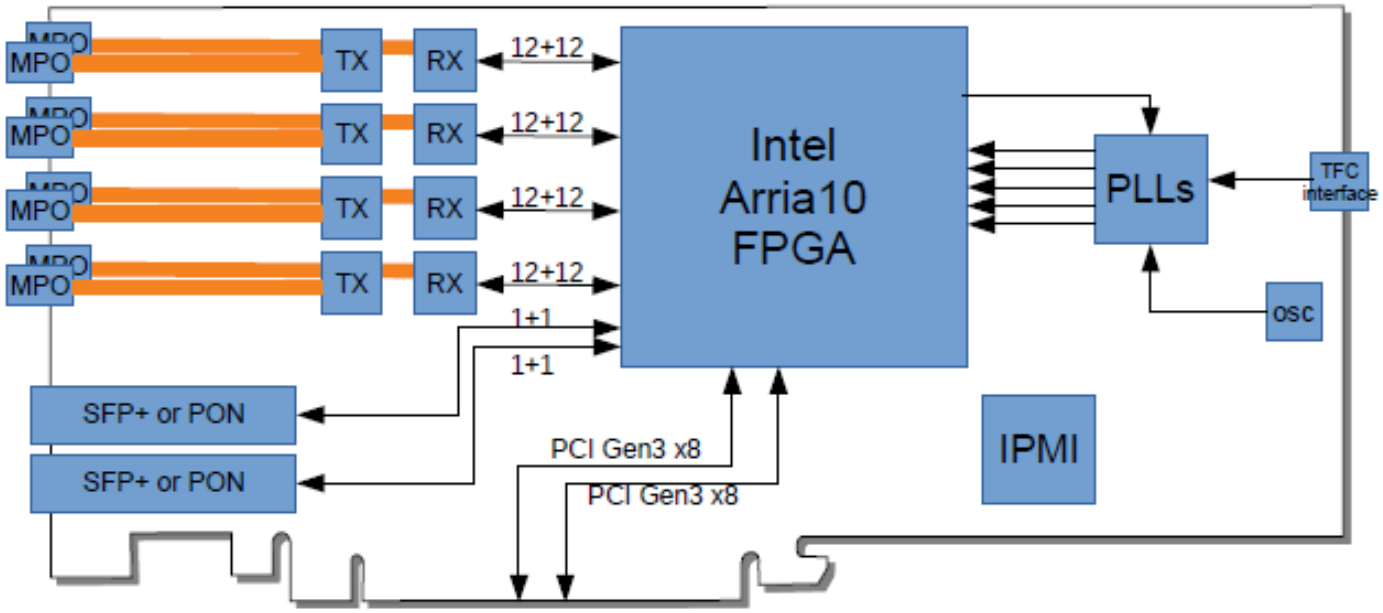
FEC (Front-end Card)

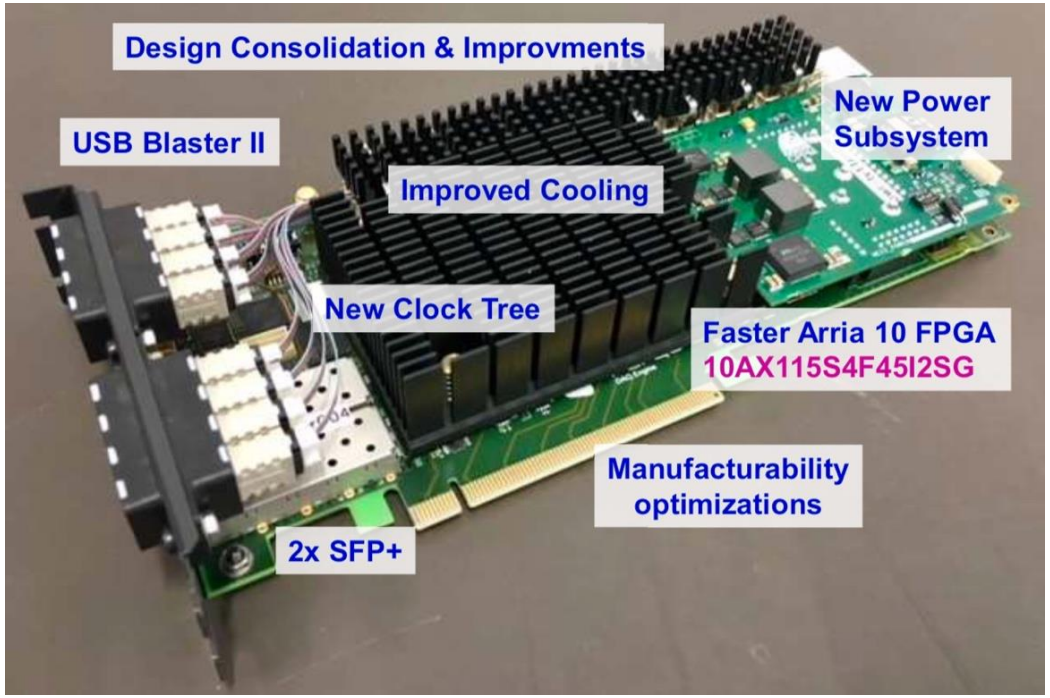
- 5 ASICs (SAMPA)
  - 32 charge sensitive shaping amplifier
  - 32 ADC at 10 bit 5 MHz
  - continuous readout
- 2 GBT optical links
  - 4 Gbps each
  - radiation hard
- design by ORNL, USA
- 3276 FECs to be installed

CRU with a FPGA

CRU (Common Readout Unit): commonly used in all ALICE detectors and other experiments

- Hardware design by Marseille group (LHCb + ALICE)
- 48 GBT duplex SERDES links → maximum 4.48 Gbps x 48 = 215 Gbps
- Intel/Altera Arria 10 FPGA → **>200 times acceleration(TPC CF) w.r.t. Intel Xeon CPU core**
- PCI Express 3 x16Lane (128 Gbps) → max tested throughput ~90 Gbps

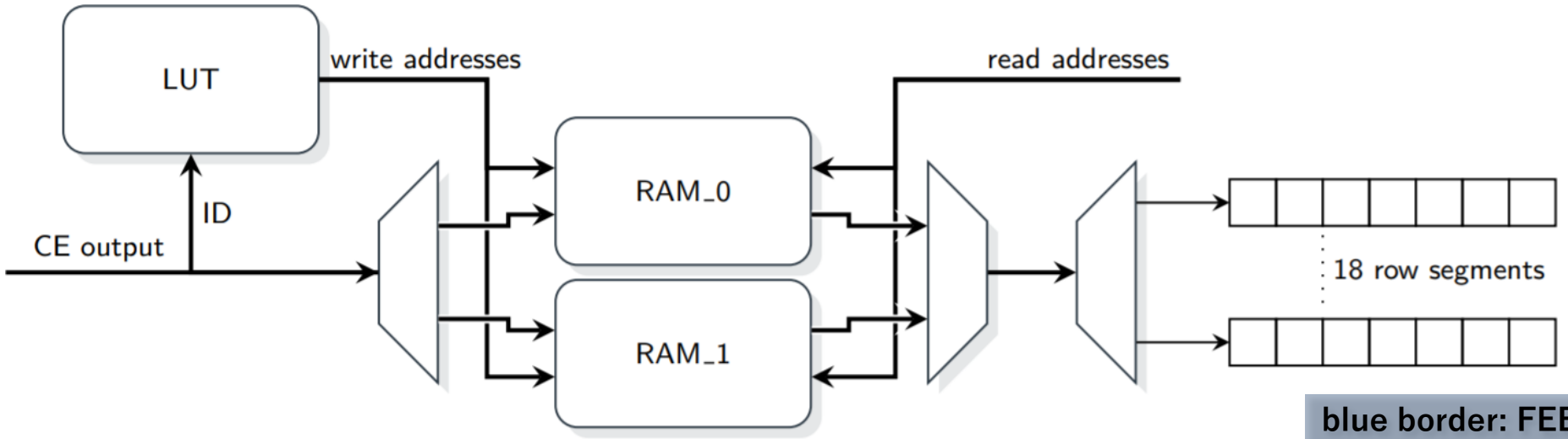


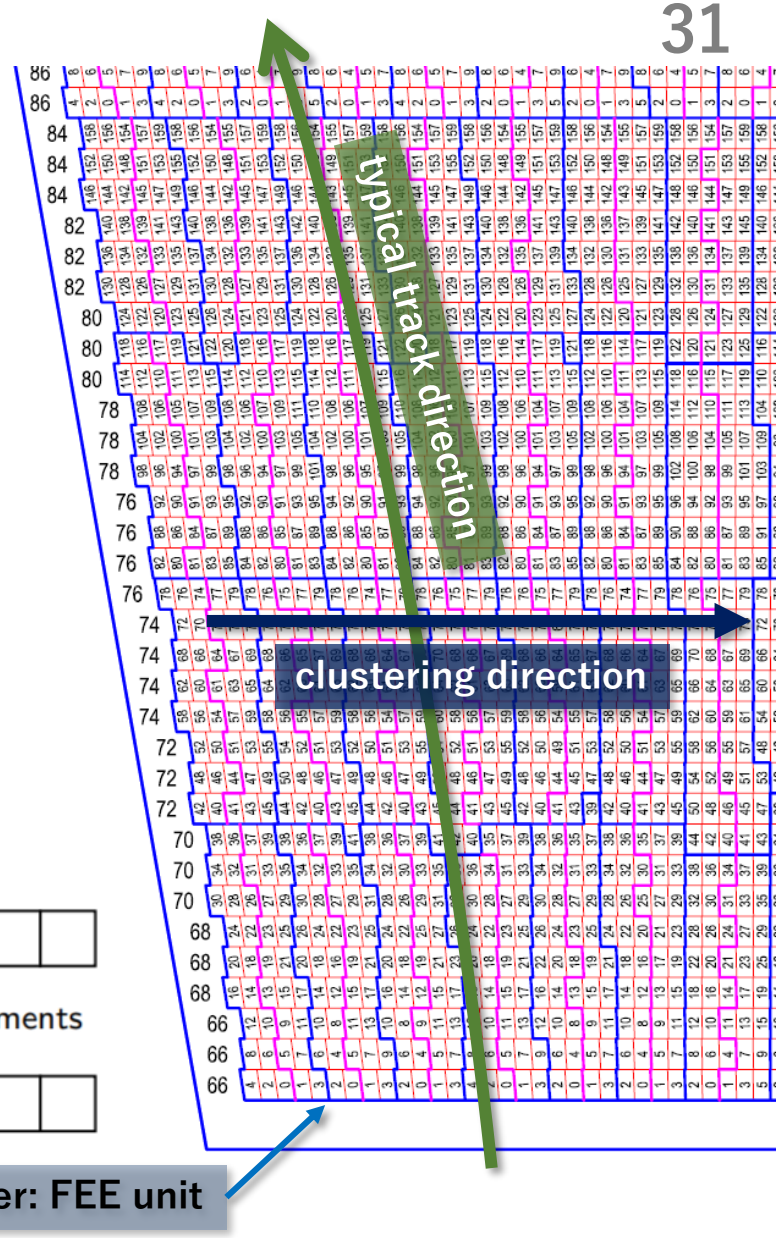


Drawing and photo by Kiss Tivadar

Sorting

- due to geometrical limitation, ADC data from one pad row (clustering unit) arrive to FPGA more less randomly
  - large routing matrix (1600-to-1600) inside FPGA
  - configurable (360 FPGAs with different mapping)
    - firmware compilation takes 24 hours with standard PC
  - using memory in FPGA (configurable LUT for writing address)





Sebastian Klewin, Dec. 2017

Common mode noise filtering

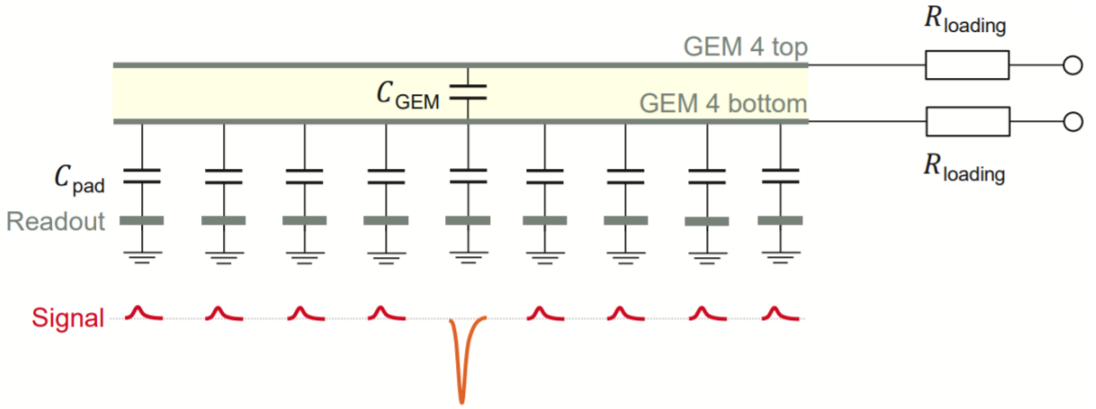
- GEM + pads = parallel plate capacitor→ capacitive coupling causes strong common mode crosstalk
- FPGA is powerful for removing this crosstalk
  - 1600 channel concurrent adaptive filter
  - calc. average of 1600 ADC values at every 200 ns

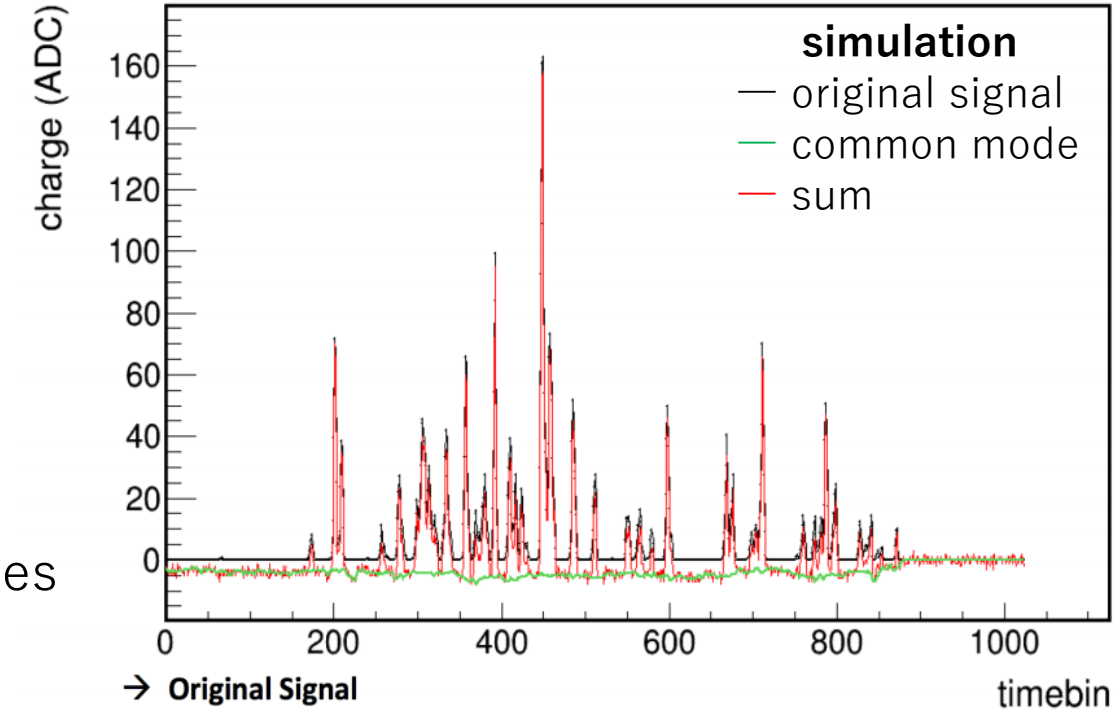
$$O_j = I_j - I_{CM} \quad , \qquad I_{CM} = \frac{\sum I_i}{N_{cont}}$$

  - only this operation even occupy one CPU core!
- Weak point: average value is biased by signal

Plan A: peak rejection by detecting rising & falling edges

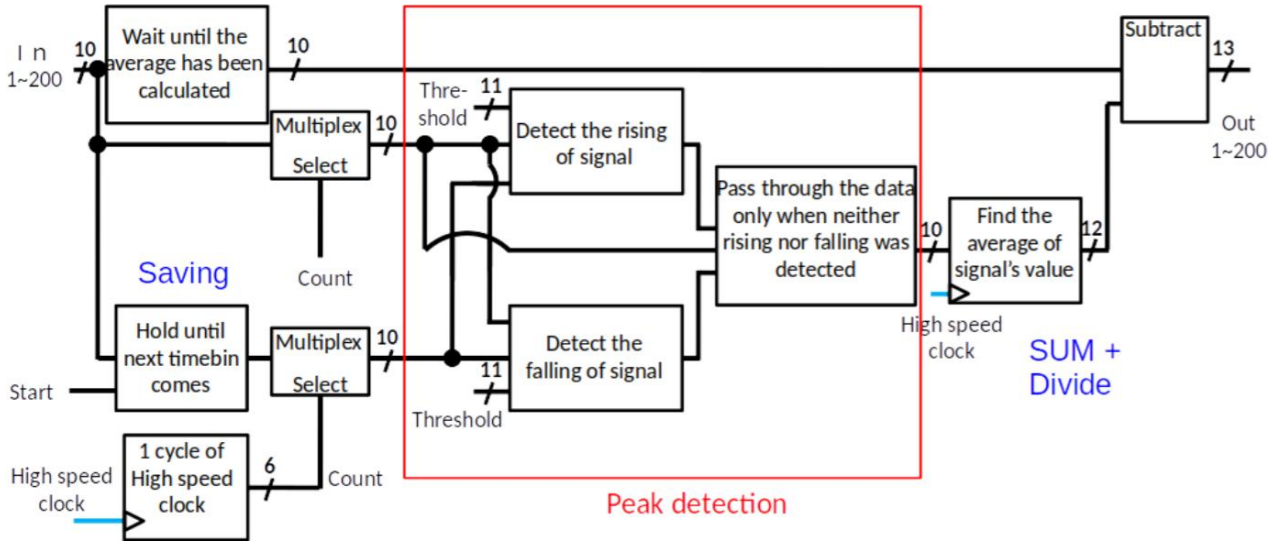
Plan B: calculate median value

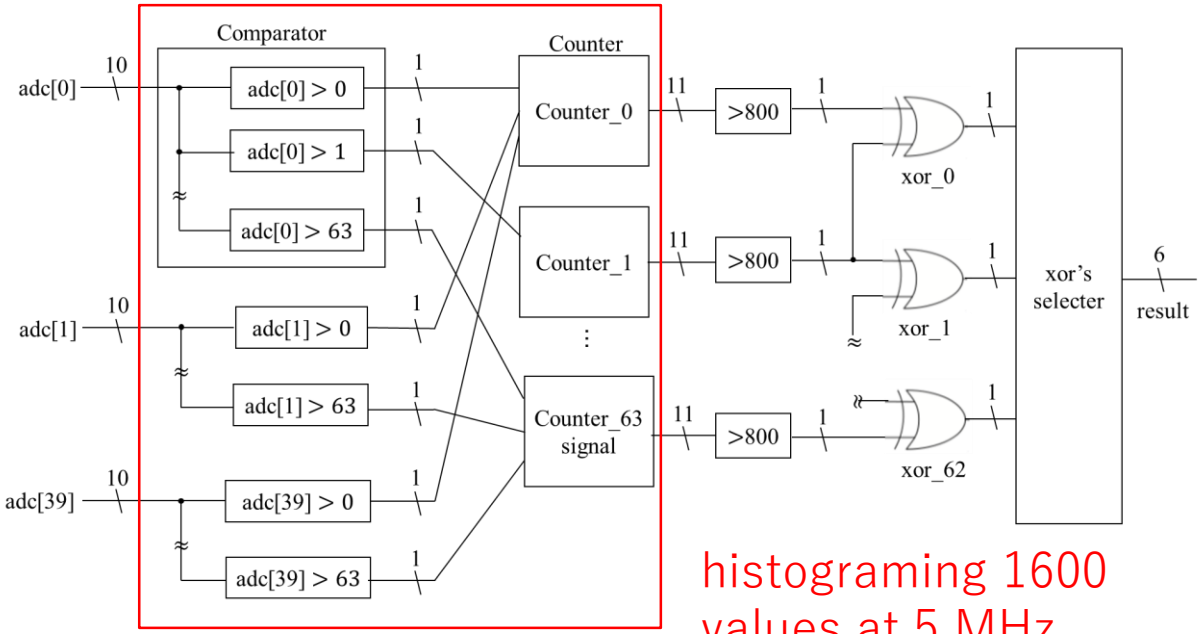




Common mode noise filtering (cont.)

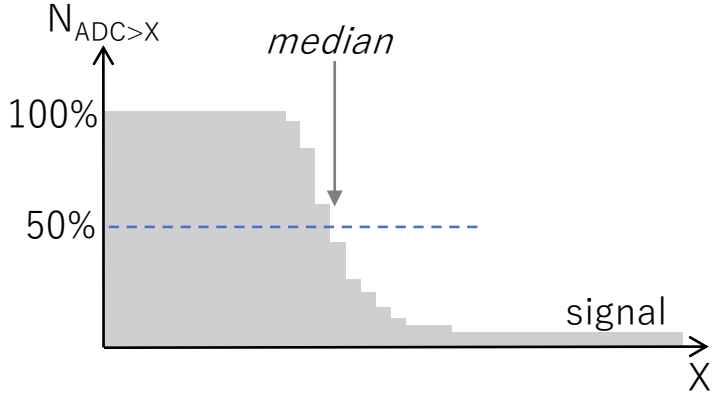
[peak rejection by Y. Takeuchi] v.s. *[median by Y. Matsuyama]*



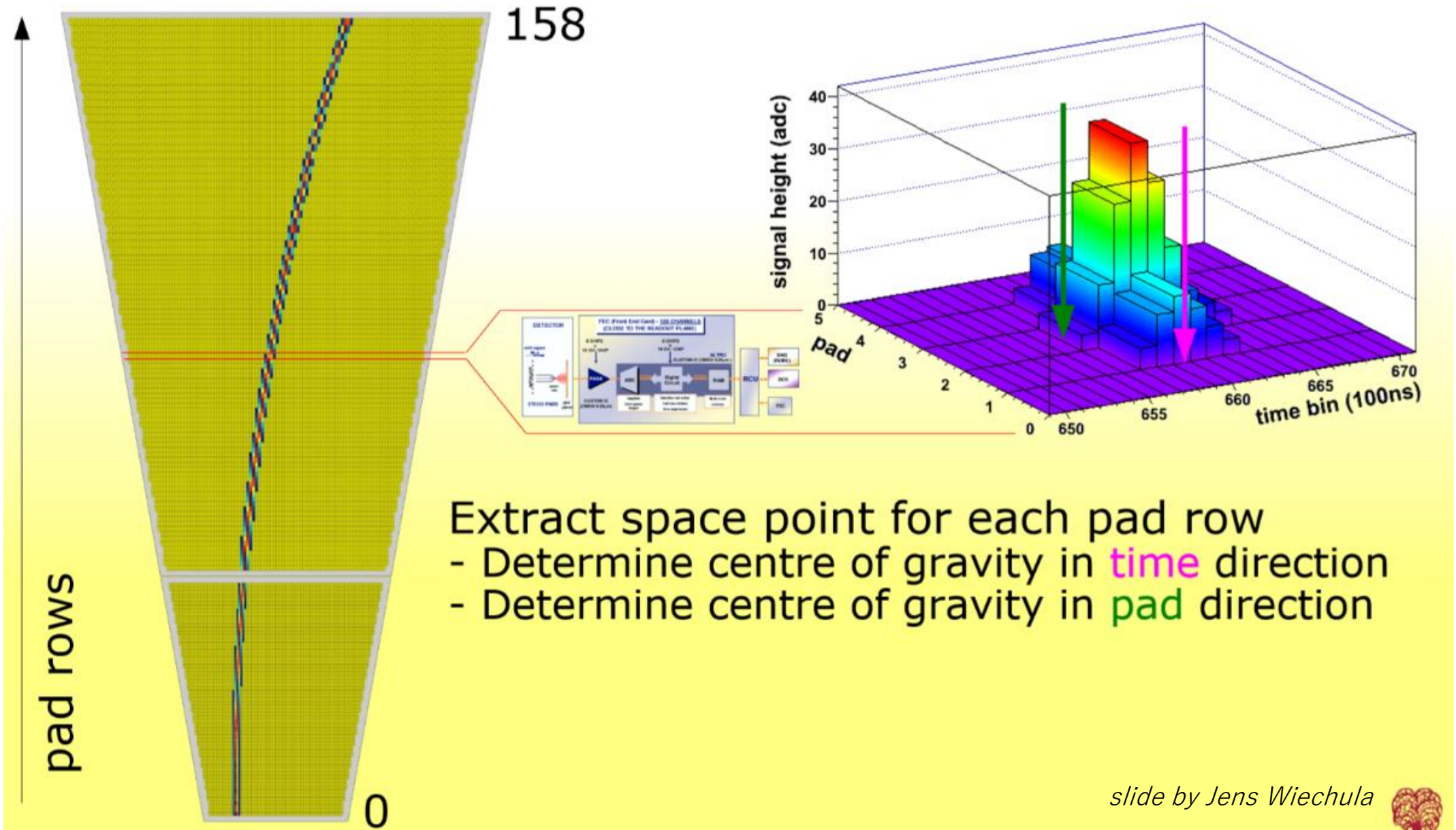


histograming 1600 values at 5 MHz

Type	plan A: peak detection	plan B: median
ALM use	2%	**15%**
bias	bad (1-2 ADC value)	good(~zero)
occupancy limit	ok up to 70-80%	up to 50%



Clustering instead of zero suppression

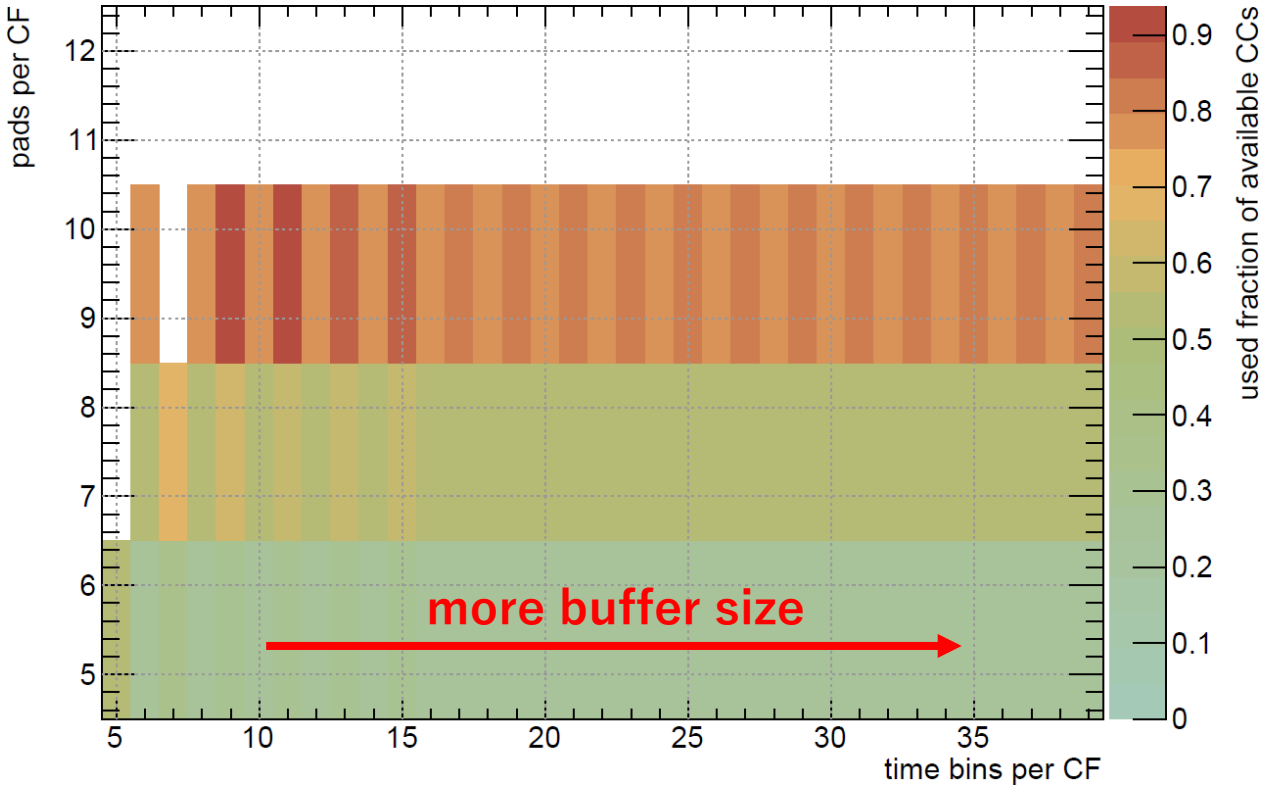


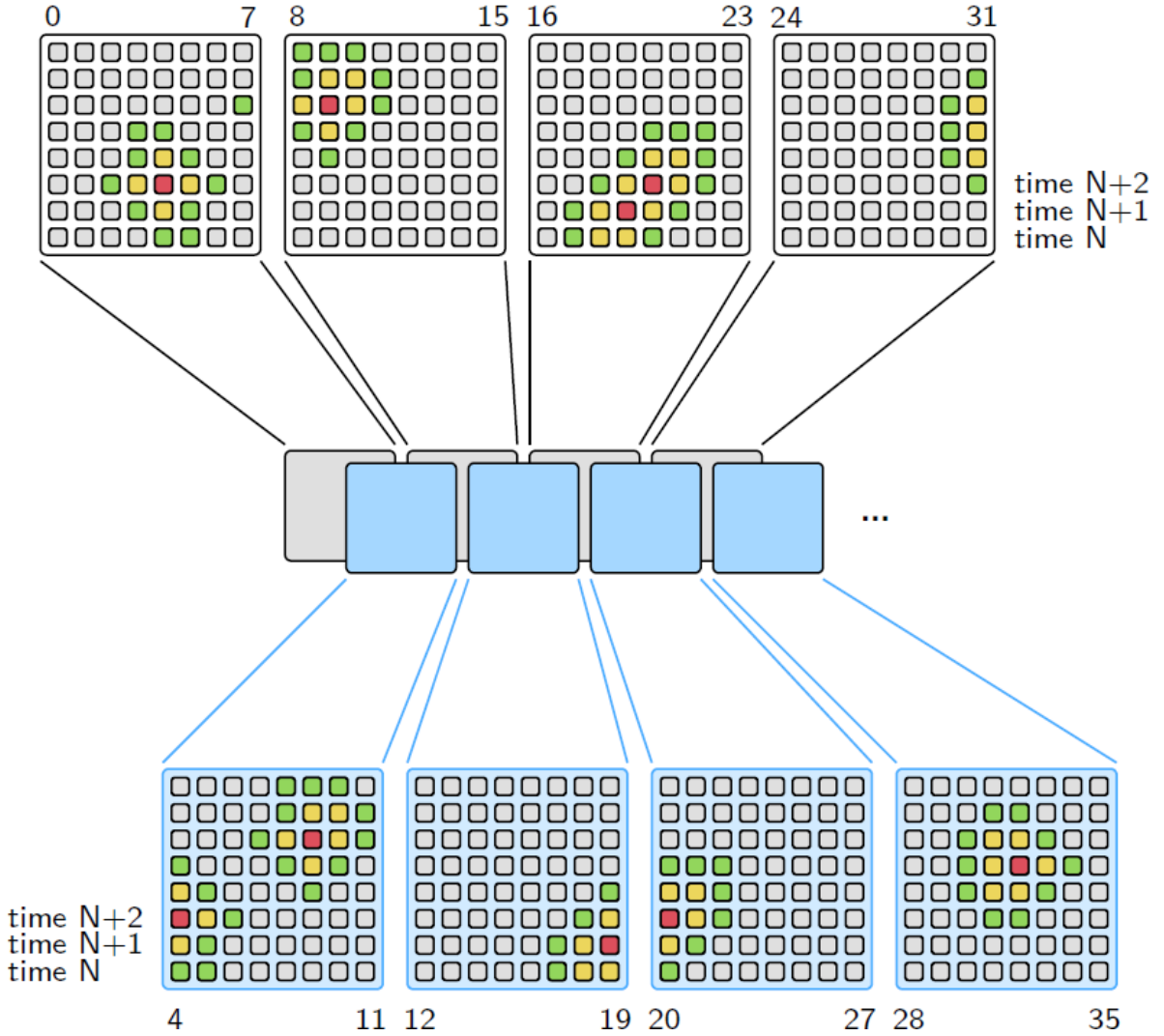
Extract space point for each pad row
- Determine centre of gravity in time direction
- Determine centre of gravity in pad direction

slide by Jens Wiechula

Cluster finding algorithm

- find local maxima in pad-timebin 2D space
- processor modules to scan rectangular regions
  - optimize size vs number of processors
  - available clock cycle is the boundary condition





drawings by Sebastian Klewin

Cluster parameter pre-calculation

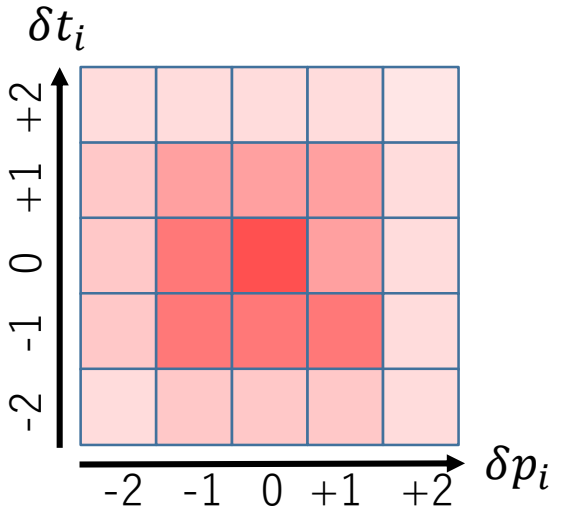
■ Calc. cluster characteristics for 5x5 pad-timebin region around the peak

local coordinate

- $q_{tot} = \sum q_i$
- $\mu_x = x + \frac{\sum q_i \, \delta x_i}{q_{tot}}$
- $\sigma_x^2 = \frac{\sum q_i \, \delta x_i^2}{q_{tot}} - (x - \mu_x)^2 = \frac{\sum q_i \, \delta x_i^2}{q_{tot}} - \left(\frac{\sum q_i \, \delta x_i}{q_{tot}}\right)^2$

$i = 1 \ldots 25$, x: pad, timebin index

product with $\delta x_i = \{-2, -1, 0, +1, +2\}$ =bit shift



■ Avoid division in FPGA, and put that work for CPU

FPGA friendly (sums + bit shifts)

- $q_{tot} = \sum q_i$
- $\hat{\mu}_p = \sum q_i \delta p_i$
- $\hat{\mu}_t = \sum q_i \delta t_i$
- $\hat{\sigma}_p = \sum q_i \delta p_i^2$
- $\hat{\sigma}_t = \sum q_i \delta t_i^2$

transfer data via PCI Express

250 bit → 160 bit packing

CPU friendly (division, square)

- $\mu_p = p + \hat{\mu}_p / q_{tot}$
- $\mu_t = t + \hat{\mu}_t / q_{tot}$
- $\sigma_p^2 = \hat{\sigma}_p / q_{tot} - \left(\hat{\mu}_p / q_{tot}\right)^2$
- $\sigma_t^2 = \hat{\sigma}_t / q_{tot} - (\hat{\mu}_t / q_{tot})^2$

S. Klewin's PhD thesis coming soon

FPGA resources to be used

■ Arria10 10AX115S3F45E2SG

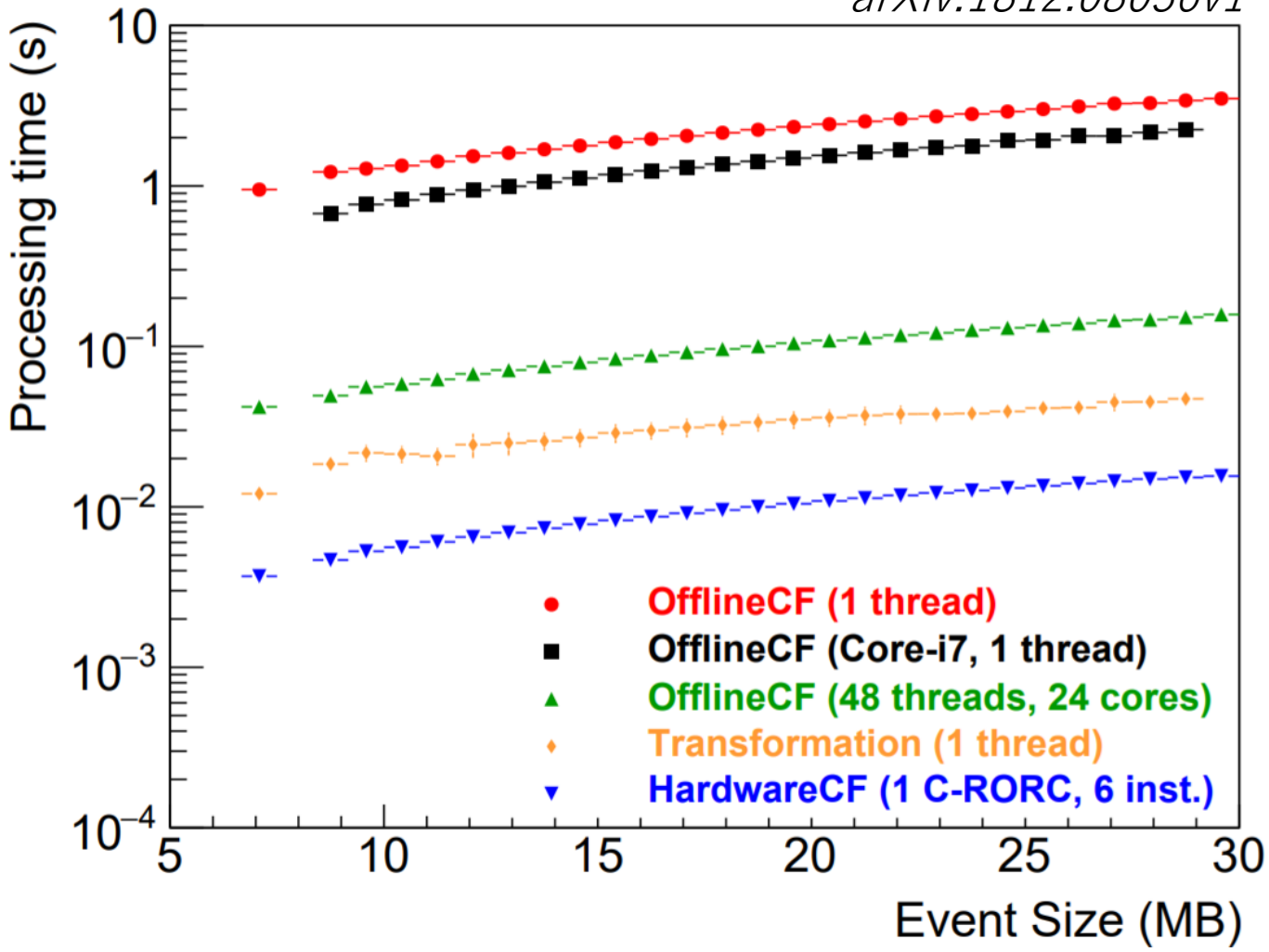
Module	ALMs	%	M20Ks	%	DSPs	%
peripheral logics	7144	1.7	116	4.3	54	3.6
GBT stream decoding	10264	2.4	0	0	0	0
sorting	43956	10.0	40	1.5	0	0
common mode filter	u.e.	2-15	0	0	0	0
clustering	167905	39.3	906	33.4	362	23.8
readout	3890	0.9	0	0	0	0
configuration	10024	2.3	0	0	0	0
total user logic	**243184**	**57**	**1062**	**39**	**416**	**27**
common logic	119762	28	1252	46	0	0
total	**362946**	**87-100**	**2314**	**85**	**416**	**27**

looking for rooms to reduce ALM usage (common mode, cluster finder)
by using DSP, better algorithm, etc

FPGA acceleration factor for TPC clustering

- Tested with Virtex-6 only for clustering yet
  - testing with full function with Arria10 will come soon
- Xeon E5-2697 and Core-i7 compared to hardware (C-RORC)
- > x10 improvement compared to 48 threads software processing
  - → a FPGA (Virtex-6) corresponds to 240 Xeon E5 cores
- Transformation: only coordinate transformation

arXiv:1812.08036v1



Conclusion and Outlook

- Breakthroughs on particle accelerator and detector are bringing high energy physics field to a new era
  - HPC with acceleration technology using FPGA is indispensable
- Modern FPGA is found to be quite powerful for HPC for physics
  - strong data reduction with massive parallelism
  - replaces two orders of magnitude larger number of CPU core
- In ALICE, we are also developing acceleration using GPU (tracking, Kalman filter algorithm, DNN for particle finding, etc)
  - FPGA to GPU efficient data connection (at above 100 Gbps b.w.) is one of the desired technologies
- In future, high energy physics field is proposing many larger projects and those may need even higher performance computing
  - where FPGA and GPU may become more and more important

Future projects