Input Mezzanine and Data Formatter for the Fast Tracker at ATLAS

Takashi Mitani

for FTK_IM & Data Formatter team

14-15/May/2015

Aristotle University's Research Dissemination Center

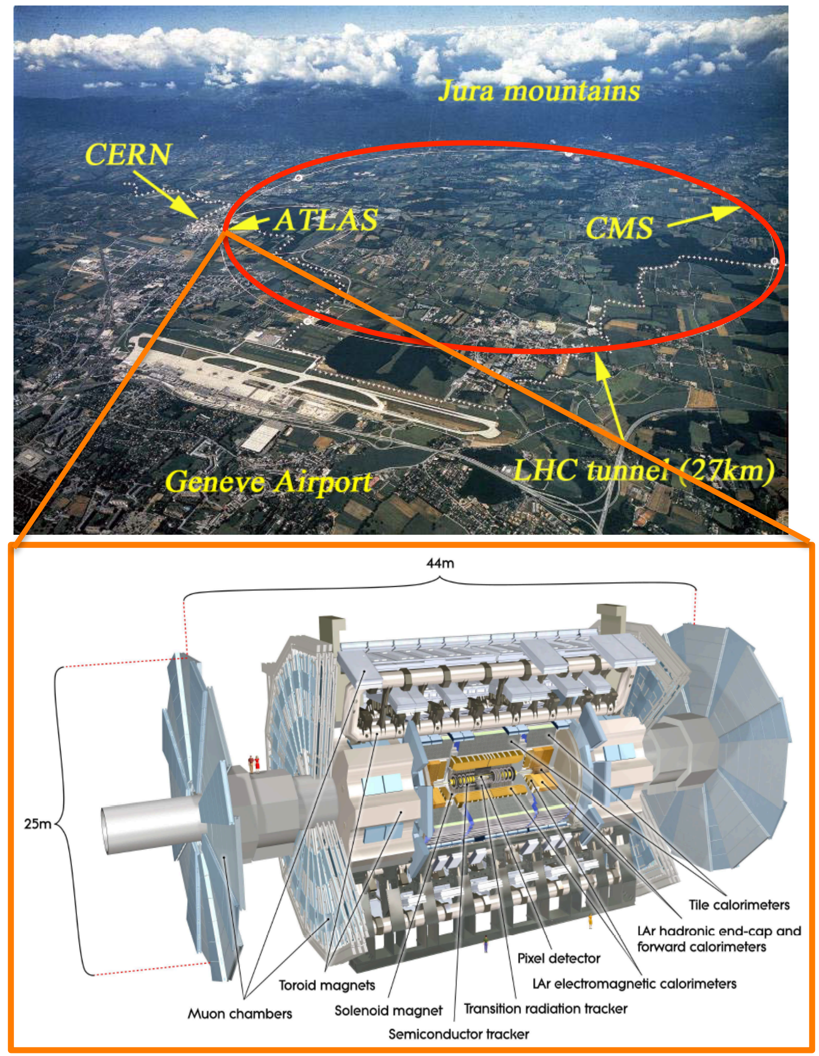
Thessaloniki, Greece

ATLAS Experiment at LHC

- ATLAS is a general purpose detector
  - Muon spectrometers
  - EM/Hadronic calorimeters
  - Trackers
    - ➢ 2D Pixel detector (**IBL, Pixel**)
    - ➢ 1D micro strip detector (**SCT**)

Very successful run during 2010-2012 (Run 1)

… Discover Higgs with a mass of ~ 125 GeV/c^2

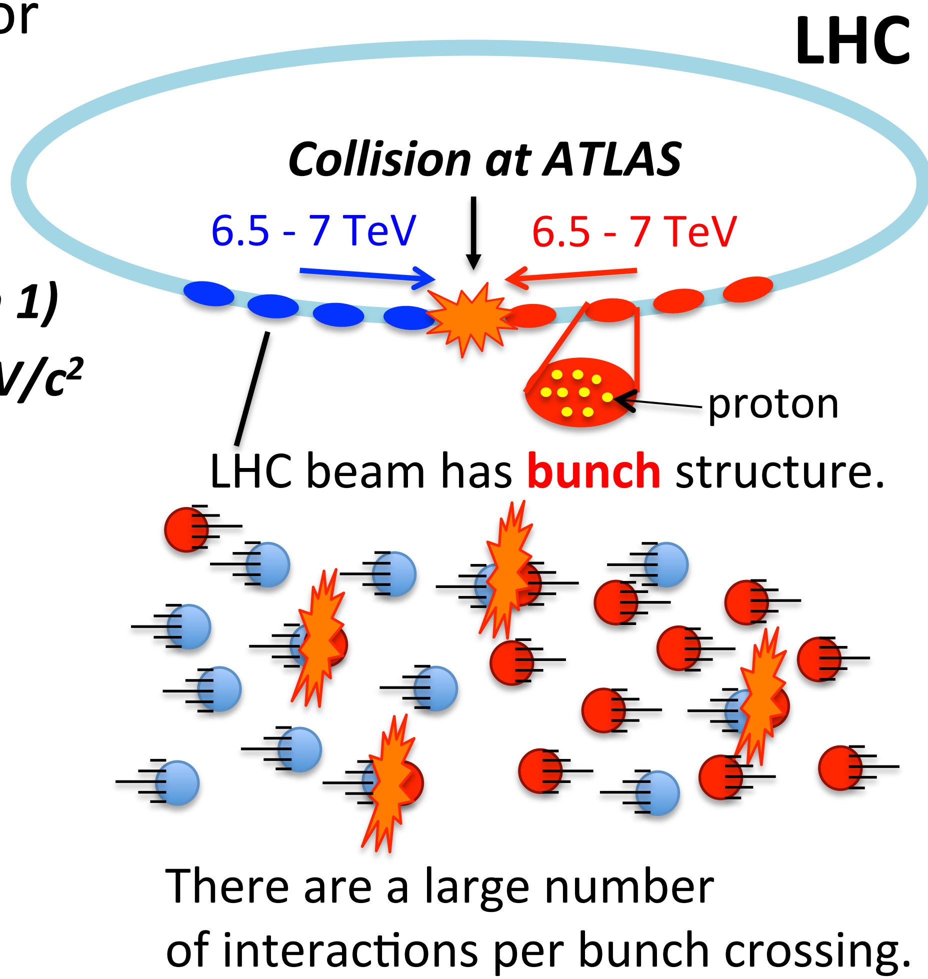


ATLAS Experiment at LHC

- ATLAS is a general purpose detector
  - Muon spectrometers
  - EM/Hadronic calorimeters
  - Trackers

Very successful run during 2010-2012 (Run 1)

… Discover Higgs with a mass of ~ 125 GeV/c^2



ATLAS Experiment at LHC

- ATLAS is a general purpose detector
  - Muon spectrometers
  - EM/Hadronic calorimeters
  - Trackers

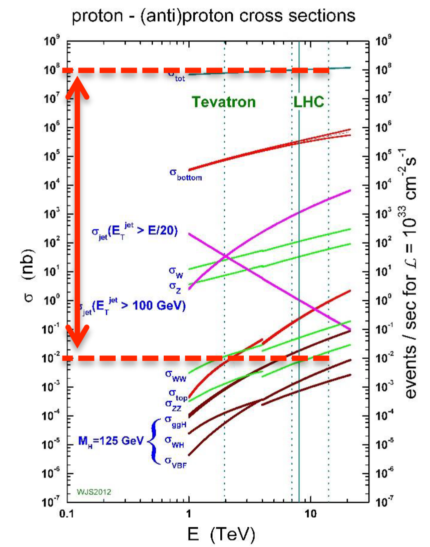
Very successful run during 2010-2012 (Run 1)

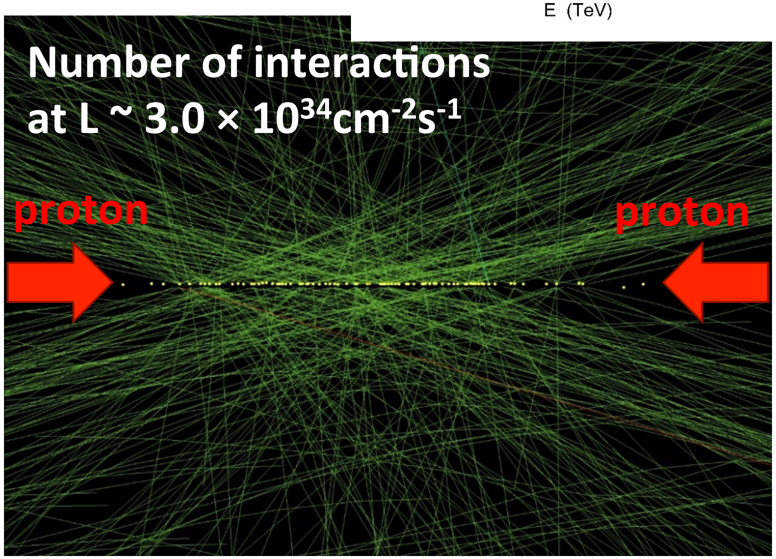
… Discover Higgs with a mass of ~ 125 GeV/c²

- LHC will be upgraded in Run 2 & 3 (2015-).
  - Energy: 8 → **13 -14 TeV**
  - Luminosity: 0.8×10^{34} → **1 - 3×10^{34} $cm^{-2}s^{-1}$**

… Increase cross section for all processes, and number of interactions per bunch crossing (pile-up).

~10^{10} difference
Signal vs Background

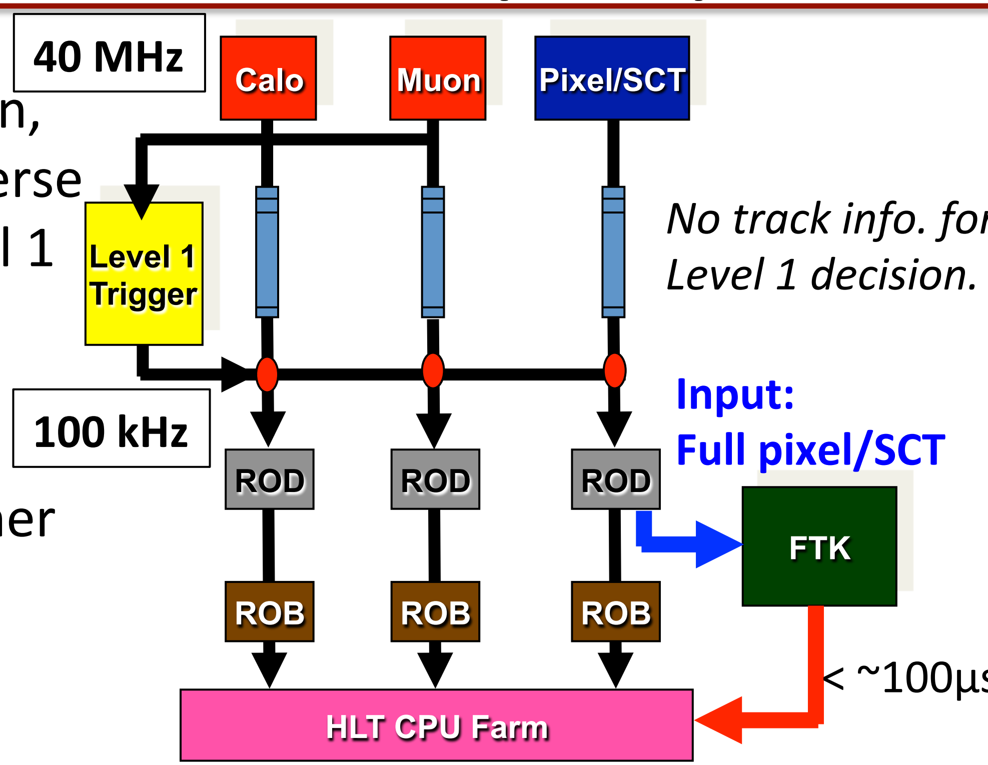




It will become much more challenging to select interesting events in real time.

What is the Fast Tracker (FTK) ?

- Using full pixel and SCT information, FTK reconstructs all tracks (transverse momentum, p_T > 1 GeV) with Level 1 accepted events (~100kHz).
- FTK provides track information to high level trigger (HLT) for further event selection within ~100μs.



Output: High quality tracks with p_T > 1 GeV

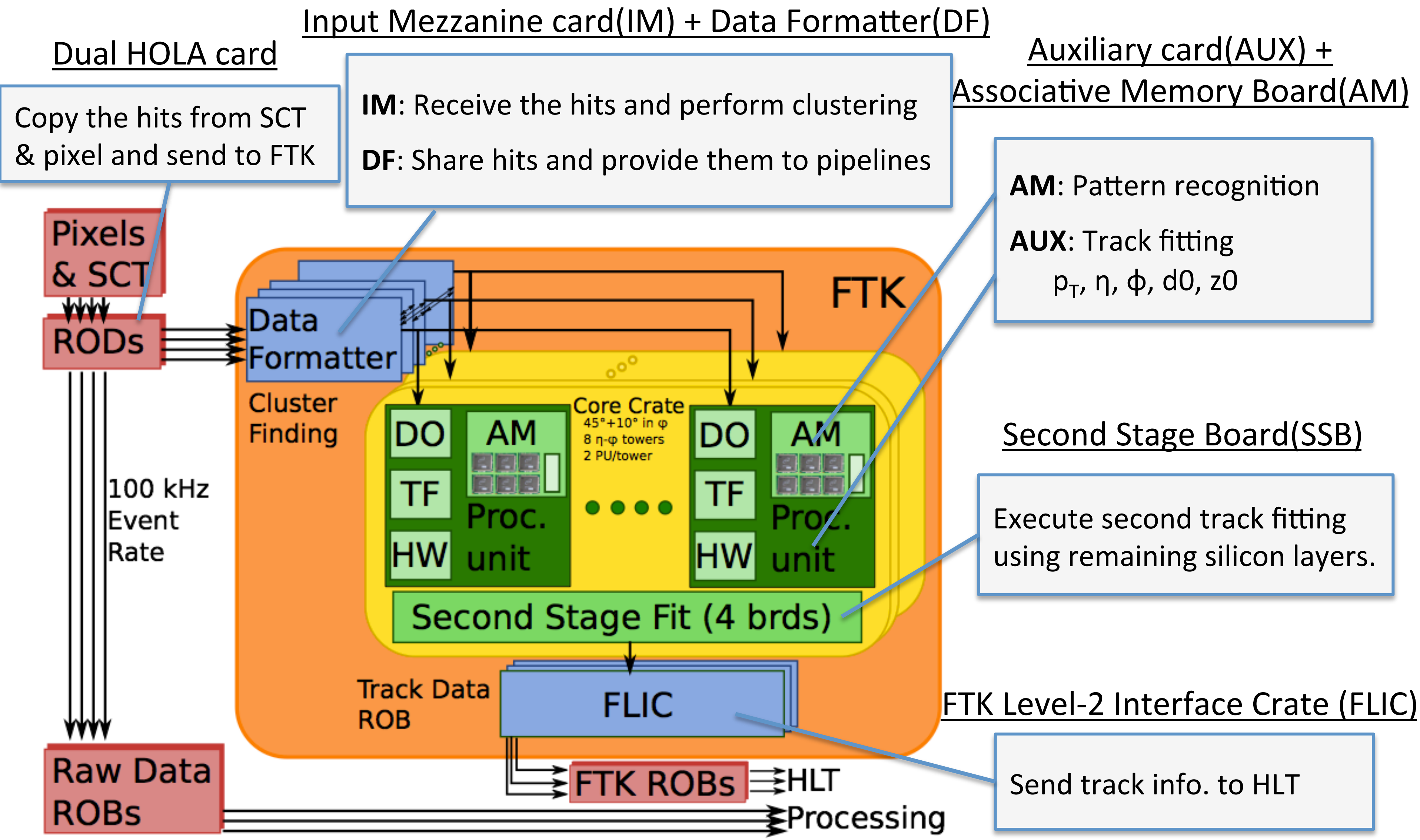
Benefit of FTK

Save the HLT CPU time and allow to execute more sophisticated algorithms at HLT.

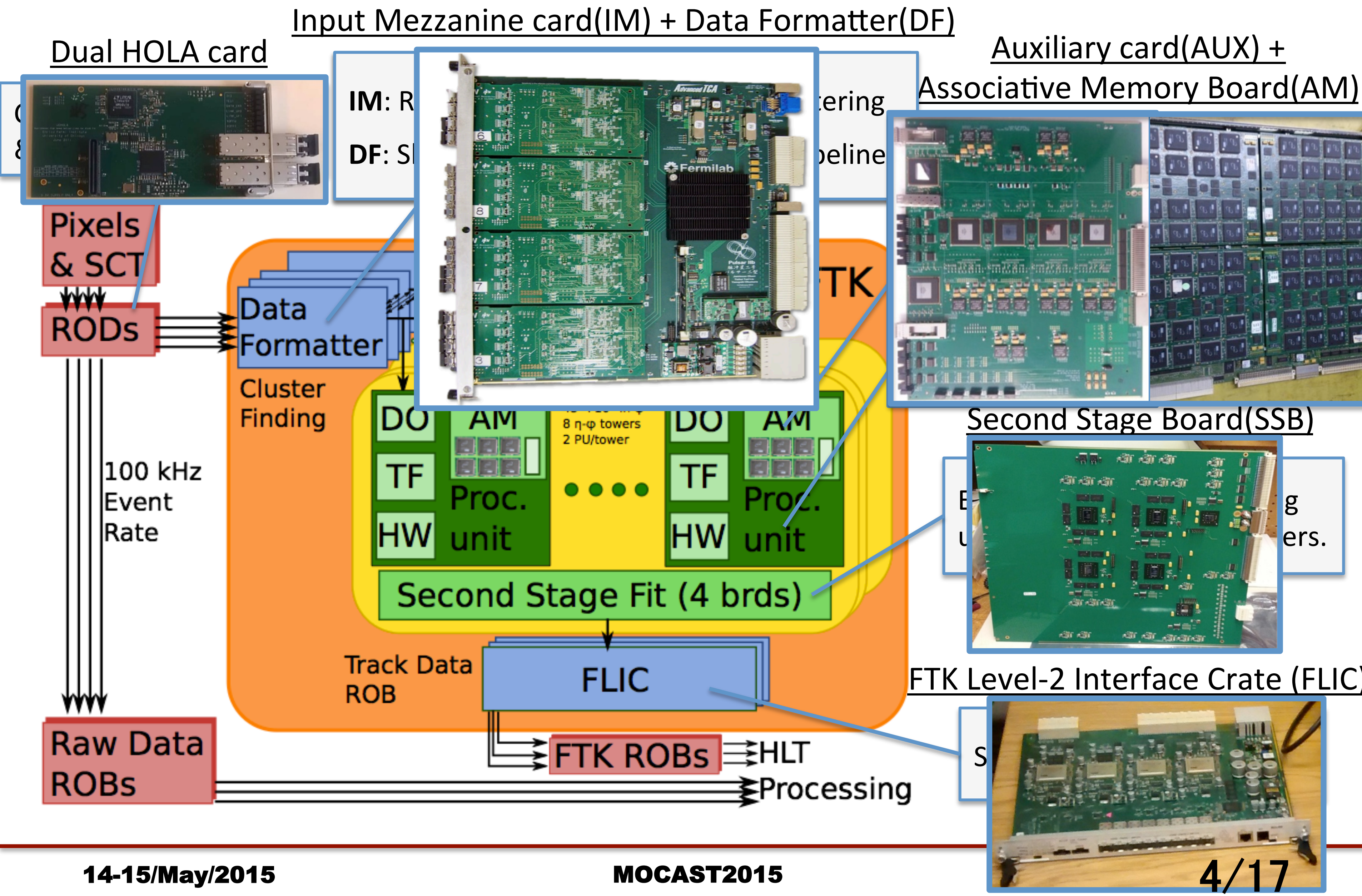
✓ Apply the correction for high pile-up condition with vertex reconstruction.

✓ Improve τ, b-jet identification, missing energy resolution, etc…

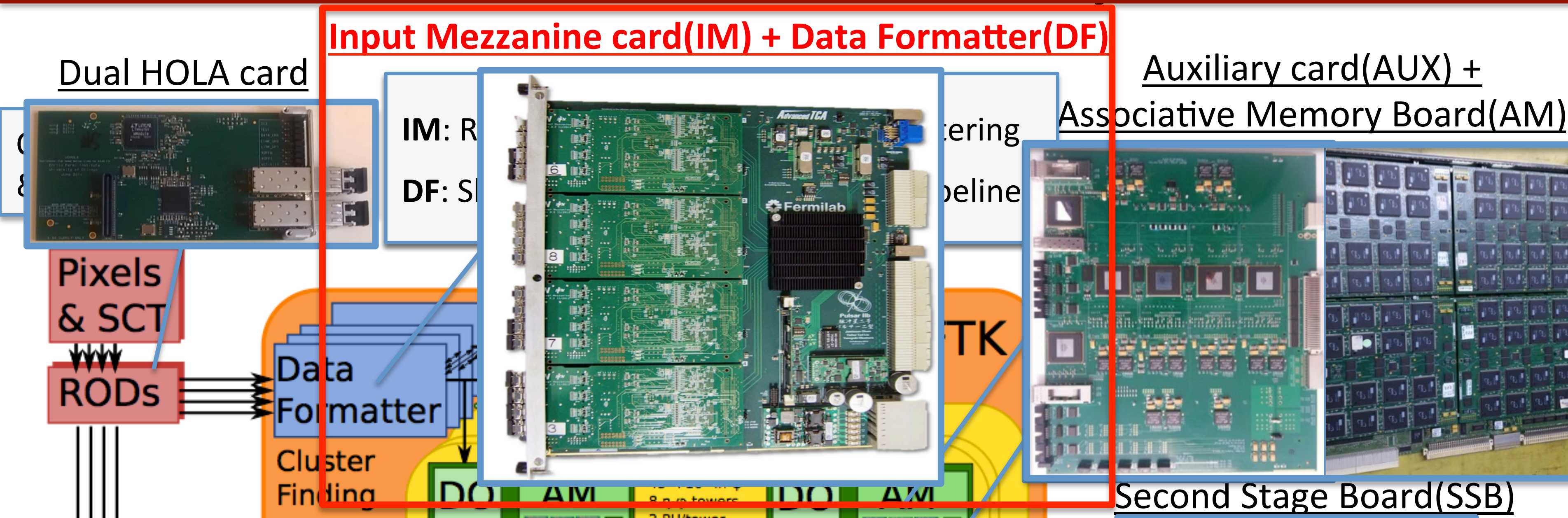
Architecture of FTK System



Architecture of FTK System



Architecture of FTK System

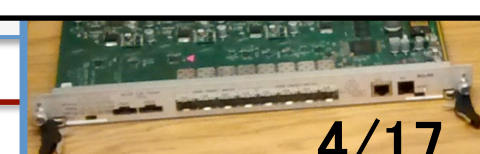


In this presentation, focus on input interface of the system.

- **FTK Input Mezzanine card (FTK_IM)**
- **Data Formatter board (DF)**

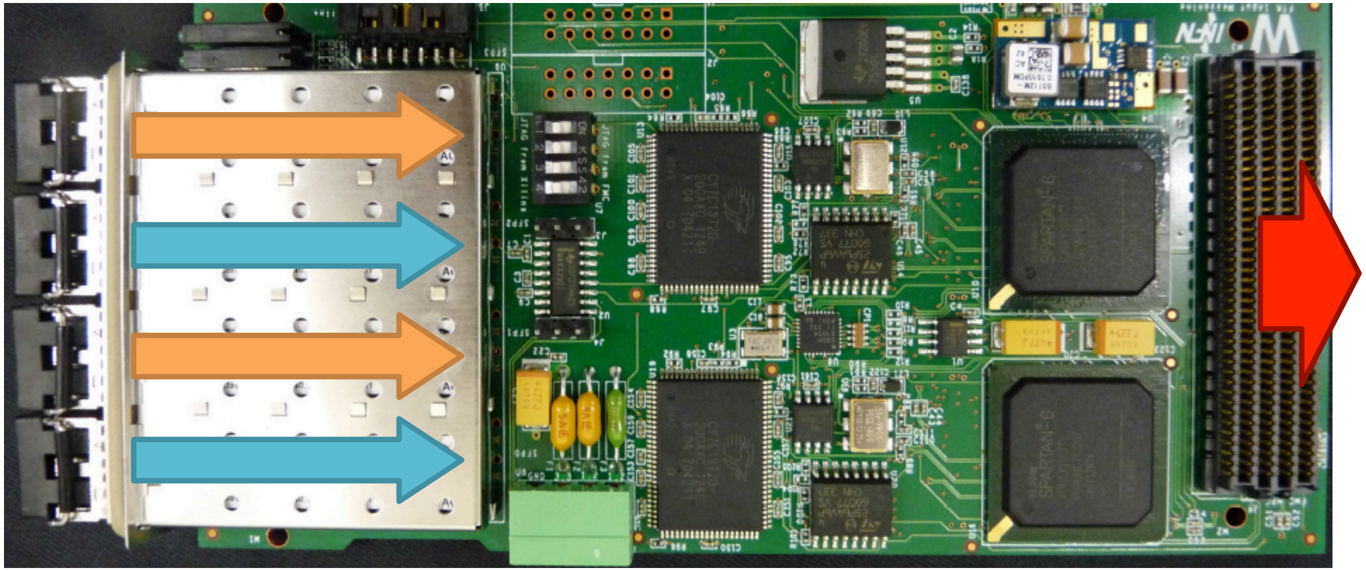
FYI: There are 2 other FTK presentations in poster session from ATLAS.

- Fast Tracker(FTK): A Hardware Track Finder for the ATLAS Trigger, T. Iizawa
- The Serial Link Processor for the Fast Tracker (FTK) processor at ATLAS, N.V. Biesuz



Requirements of FTK_IM

Input from
SCT and Pixel

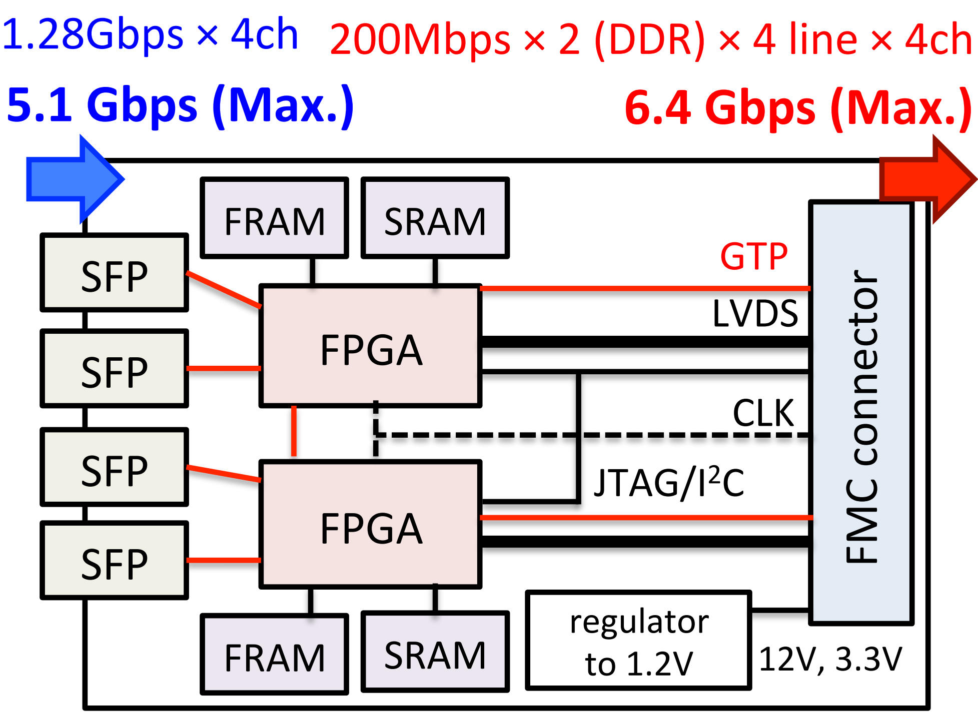


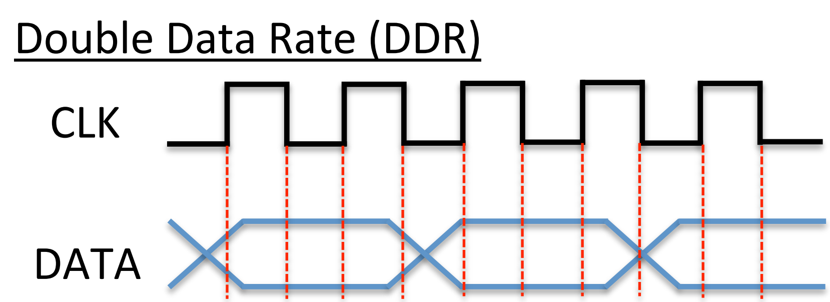
Output
to DF

- Receive hit information from SCT and Pixel Read Out Drivers (ROD) at Level 1 accepted rate (100kHz).
- Perform hit clustering to reduce data size.
- Send the clustered hit data to Data Formatter.

FTK_IM Board Design

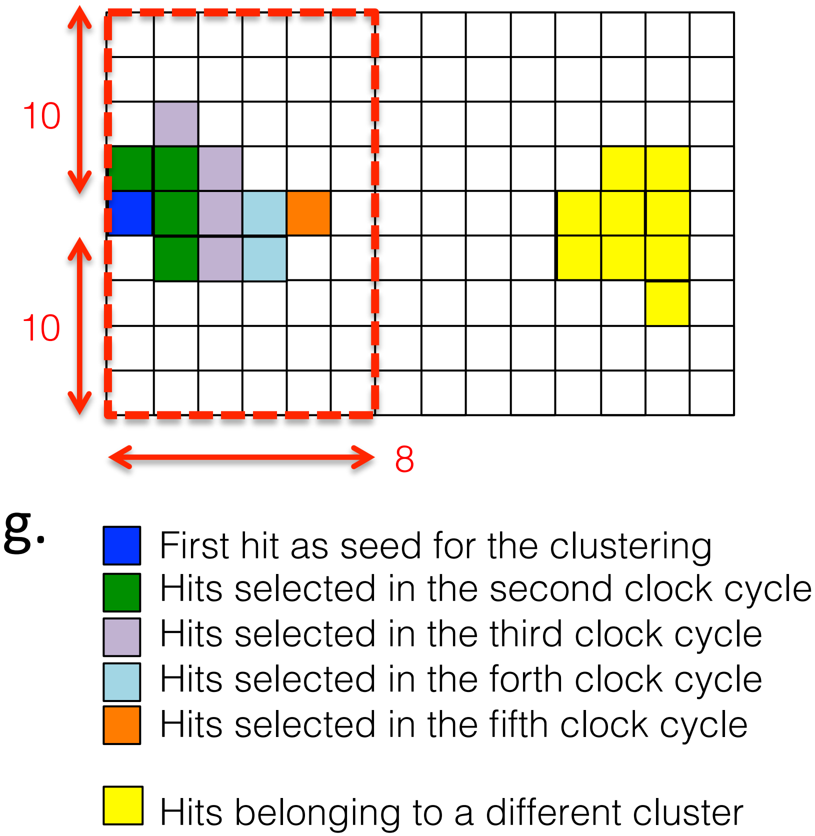
- Receive 4 optical fibers with 4 SFP transceivers (**1.28Gbps**).
- Perform clustering by 2 FPGAs
  - Spartan-6 for pixel Artix-7 for IBL. Sufficient logic cells for clustering.
  - 4 GTP lines/FPGA (**3.2Gbps**)
  - 16 LVDS lines/FPGA
- Use 12 LVDS lines/FPGA of 200 MHz DDR for sending data to DF via FMC connector.
  - 8 LVDS (data) + 4 LVDS (control bit)





Clustering Algorithm

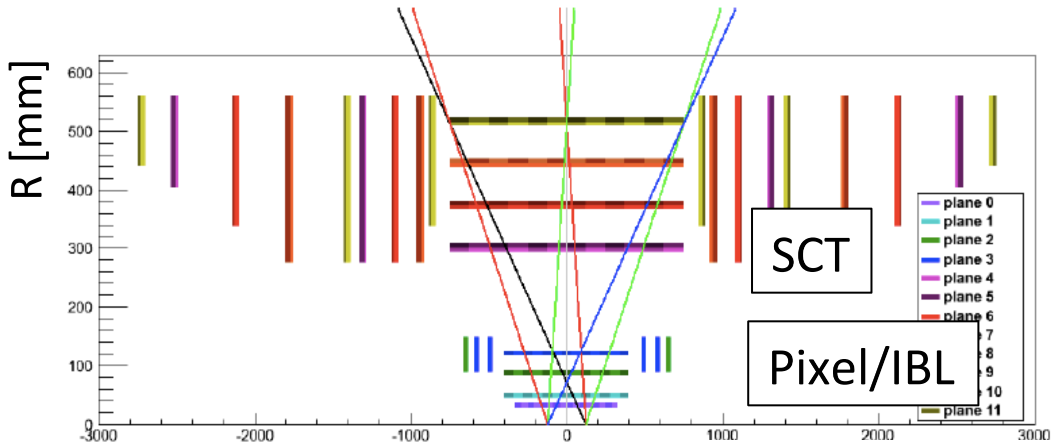
- Map pixel hits in 2D structure of cells on the FPGA.
- Sliding window technique:
  1. Place a clustering window (21×8) around the first hit (reference hit)
  2. Load all hits in the window.
  3. Select all hits neighboring to the reference/selected hits for clustering.
  4. Read selected hits as a cluster.
  5. Iterate 1 - 4 processes.

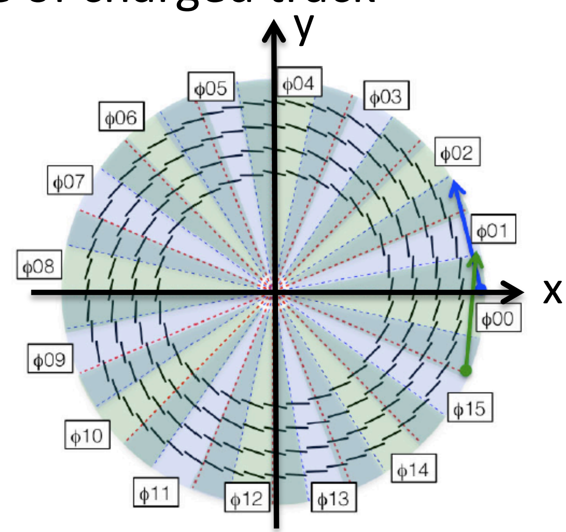


- The algorithm implementation is fully parallelized and flexible. Achieves the required speed by using 30% of the device resources.

Requirements of Data Formatter

- Pattern recognition and track fitting is performed by 64 η-ϕ overlapping towers as parallel processing unit.
  - Finite size of beam luminous region/ Curvature of charged track



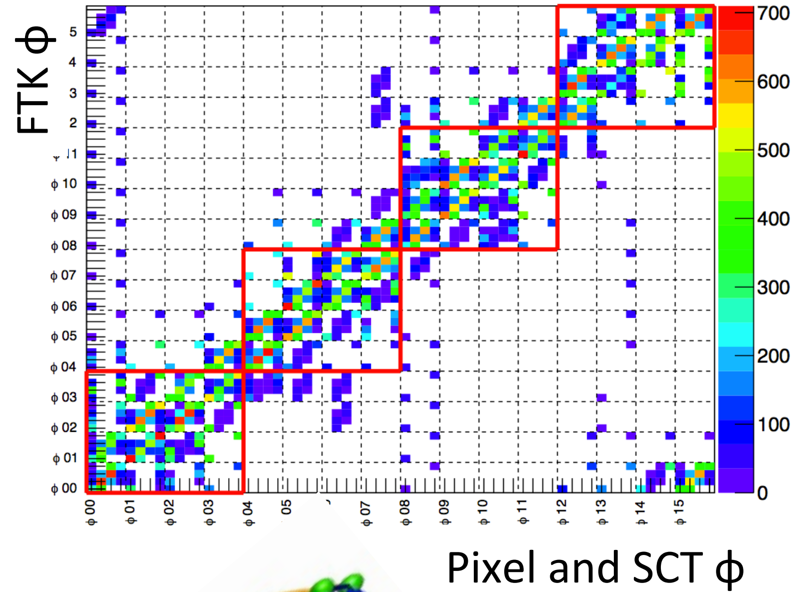


Requirements

- Remap input data into 64 towers.
- Share data considering overlap region.
- Handle a large amount of data (~400 RODs' data).

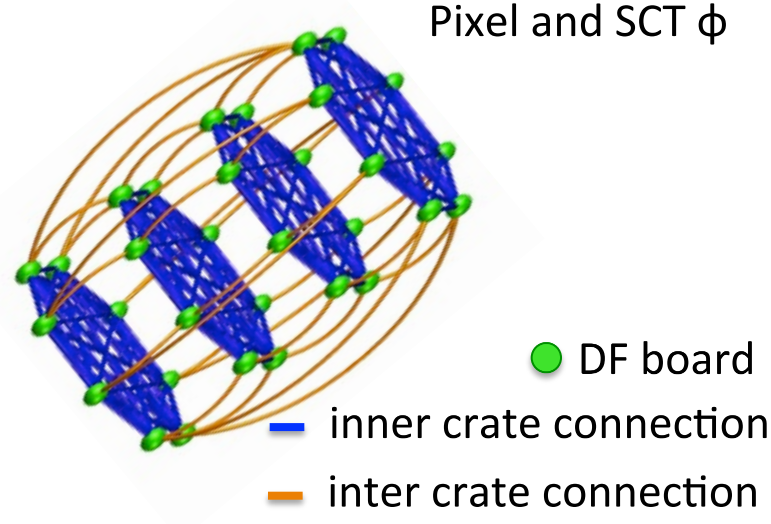
System Design for Data Formatter

- Remap pixel and SCT detector η - ϕ information to FTK logical η - ϕ information.
- Huge number of hits have to be shared among the processors to avoid inefficiency at boundary region.



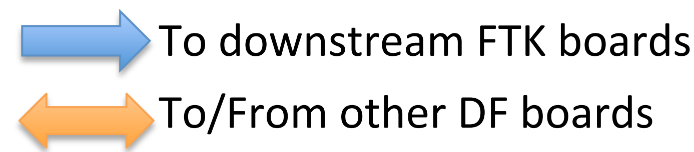
ATCA system with full mesh backplane **is a natural solution.**

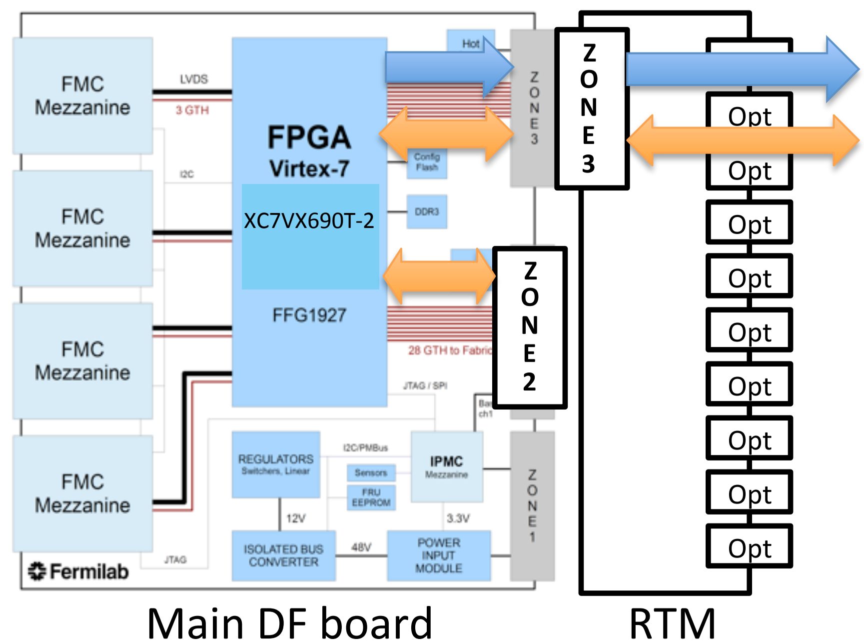
- ➢ Use 4 ATCA shelves.
- ➢ Assigned 8 boards to each shelf.
  - ✓ Full mesh communication with backplane.
  - ✓ Inter crate connection with optic fibers



Data Formatter Board Design

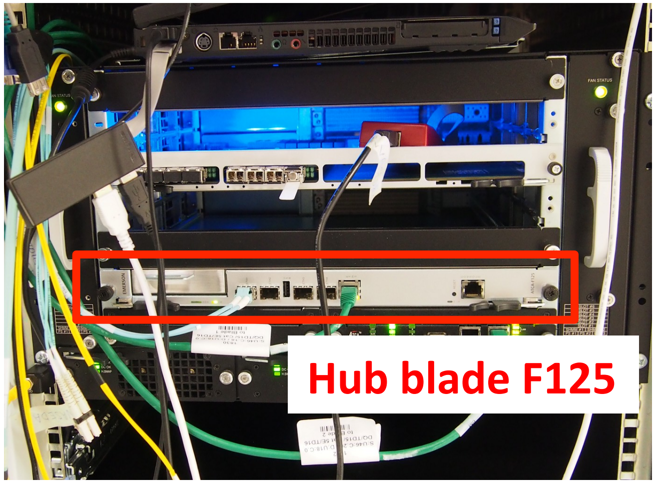
- Receive 16 inputs from 4 FTK_IMs with LVDS lines of 200 MHz DDR via FMC connector.
- Share hit information with other DF boards and send downstream FTK tracking boards.
  - Main engine of DF is one **Virtex-7 FPGA**.
  - GTH high speed lines (**10 Gbps**)
- Rear Transition Module (RTM) for optical fiber I/Os.
  - Required speed: **6.4 Gbps**
- Zone2 for ATCA backplane interconnection.
  - Required speed: **6.25 Gbps**

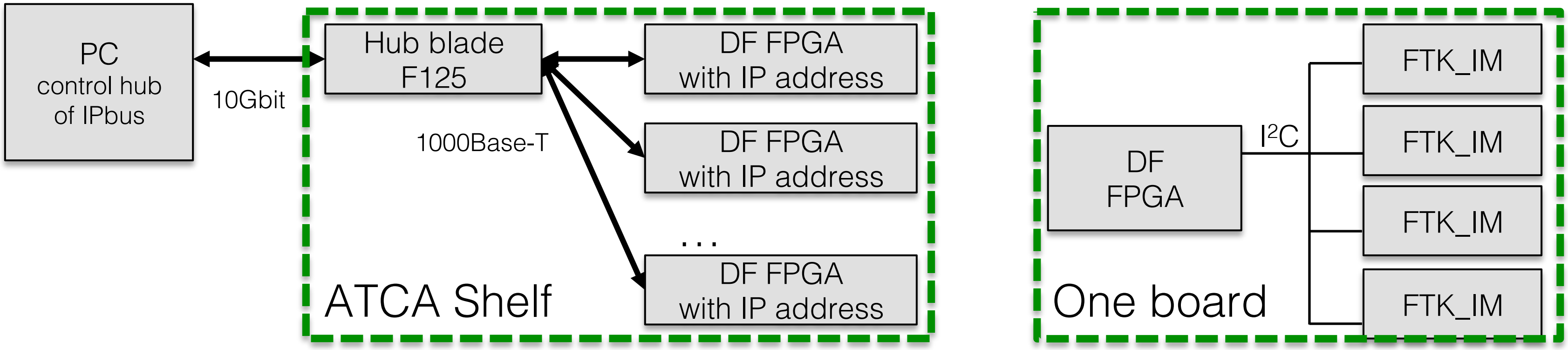




Slow Control & Monitoring

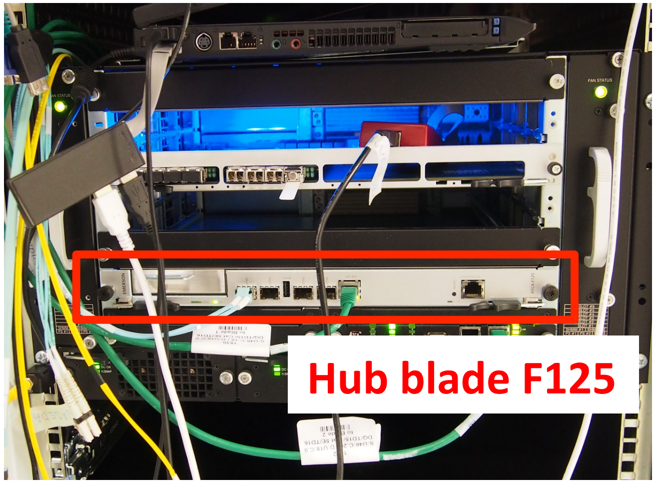
- DF boards are accessed via Ethernet links.
  - Each DF FPGA has IP address
    - ✓Control over Ethernet packet.
    - ✓IPbus is used for this access.
  - Each DF FPGA drives the I²C bus on the board to access FTK_IM.





Slow Control & Monitoring

- DF boards are accessed via Ethernet links.
  - Each DF FPGA has IP address
    - ✓Control over Ethernet packet.
    - ✓IPbus is used for this access.
  - Each DF FPGA drives the I^2C bus on the board to access FTK_IM.



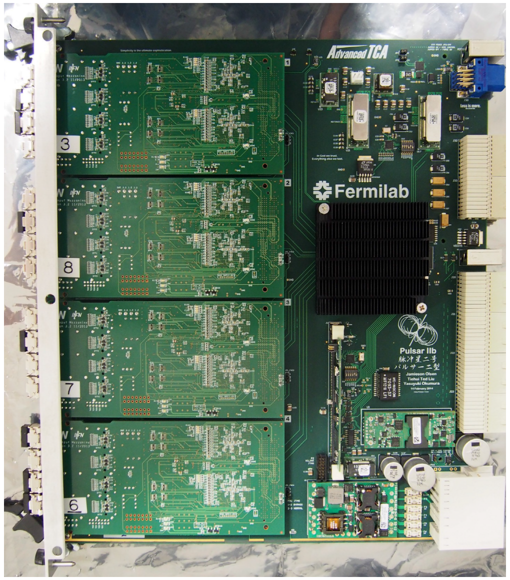
Using IPbus + I^2C connection, we can control and monitor DF and FTK_IM functionalities.

✓ Download Look Up Tables (LUTs) for each board configuration.

✓ Read spybuffer in DF & FTK_IM, etc…

Test Status of FTK_IM and DF at Lab

- **FTK_IM - DF communication**
  - BER < 10^{-15} (ATLAS requirement) in LVDS lines.
  - Clustering algorithms work well.
  - Dataflow can run with full configuration at maximum input rate (~400 words/event at 100kHz).
- **DF-DF data sharing**
  - BER < 10^{-15} in both lines to RTM and ATCA backplane.
  - Established data sharing with few channels.
  - Improvement for many channels is ongoing.
- **FTK_IM - DF - downstream board communication**
  - BER < 10^{-15} in GTH + optical fiber line.
  - With full configuration, data flow can run up to ~10 kHz.
  - Improvement for higher rate is ongoing.



Full configuration (4 FTK_IMs + DF)

Test Status of FTK_IM and DF at USA15

- FTK_IM and DF system are installed in the ATLAS underground counting room (CERN USA15).
- We prepared all optical fibers for FTK system.
- ***Confirmed that FTK_IM and DF can receive inputs from SCT and Pixel RODs.***
- Currently FTK_IM and DF are being tested using cosmic rays and first low luminous beam collision data.

Mass Production & Quality Control

- 80 FTK_IM boards with Spartan-6 FPGAs were produced by a company in Japan.
  - Requirement: 64 + 16 spares.

- A quick check was done and the mass production found to be in good shape.
  - Checked part's positions.
  - Continuity check between parts.
- Quality control tests are ongoing.
  - Check I/O functionalities.
  - Check SRAM and FRAM connectivity and functionality.
  - Do burn-in test with an extended period of time.

After completing quality control tests,
FTK_IMs will be ready to be delivered to CERN and installed.

Summary

- FTK provides full detector track information for events accepted by Level 1 trigger.
  - Allow to execute more sophisticated selection algorithms at HLT.
- FTK_IM and DF are input interface of the FTK system.
  - Receive hits from SCT and Pixel/IBL ROD at Level 1 accepted rate.
  - Perform clustering to reduce data size.
  - Distribute hits to proper downstream FTK boards.
- FTK_IM and DF boards have been tested extensively.
- Mass production has started.
  - 80 FTK_IMs of Spartan-6 are under quality control.

FTK_IM and DF installation will start this summer.

FTK_IM and DF team members

- T. Mitani: Waseda University
- A. Annovi: INFN Sezione di Pisa
- M. Berreta: INFN Laboratori Nazionali di Frascati
- R. Brown: Stanford University
- M. Gatta: INFN Laboratori Nazionali di Frascati
- S. Gkaitatzis: Aristotle University of Thessaloniki
- T. Iizawa: Waseda University
- N. Ilic: Stanford University
- N. Kimura: Aristotle University of Thessaloniki
- K. Kordas: Aristotle University of Thessaloniki
- Y. Okumura: Chicago University
- C. Petridou: Aristotle University of Thessaloniki
- C.-L. Sotiropoulou: Universita'di Pisa and INFN Sezione di Pisa
- L. Tompkins: Stanford University
- K. Yorita: Waseda University

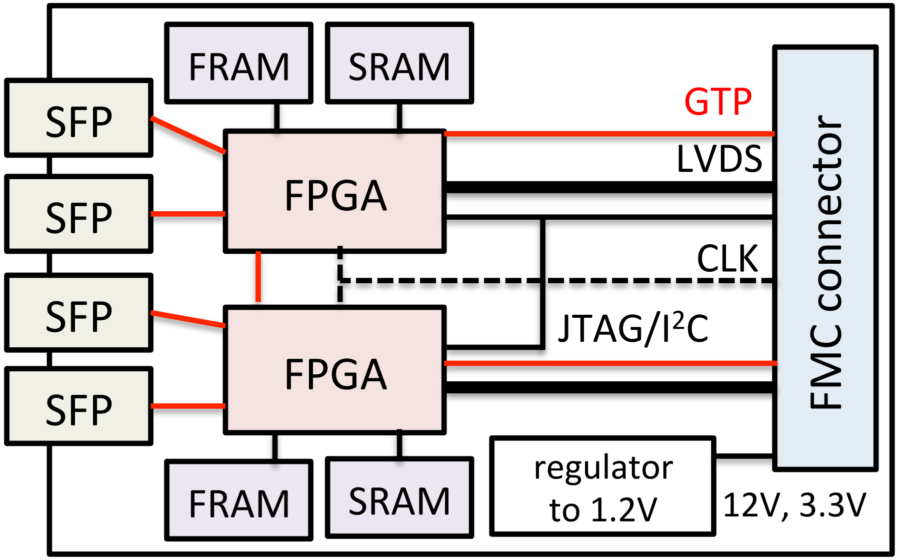

Why Add a Hardware Tracker?
Thank you for your attention

Backup

FTK_IM Board Design

- Communicate with DF via FMC connector.
  - Distribute electric power.

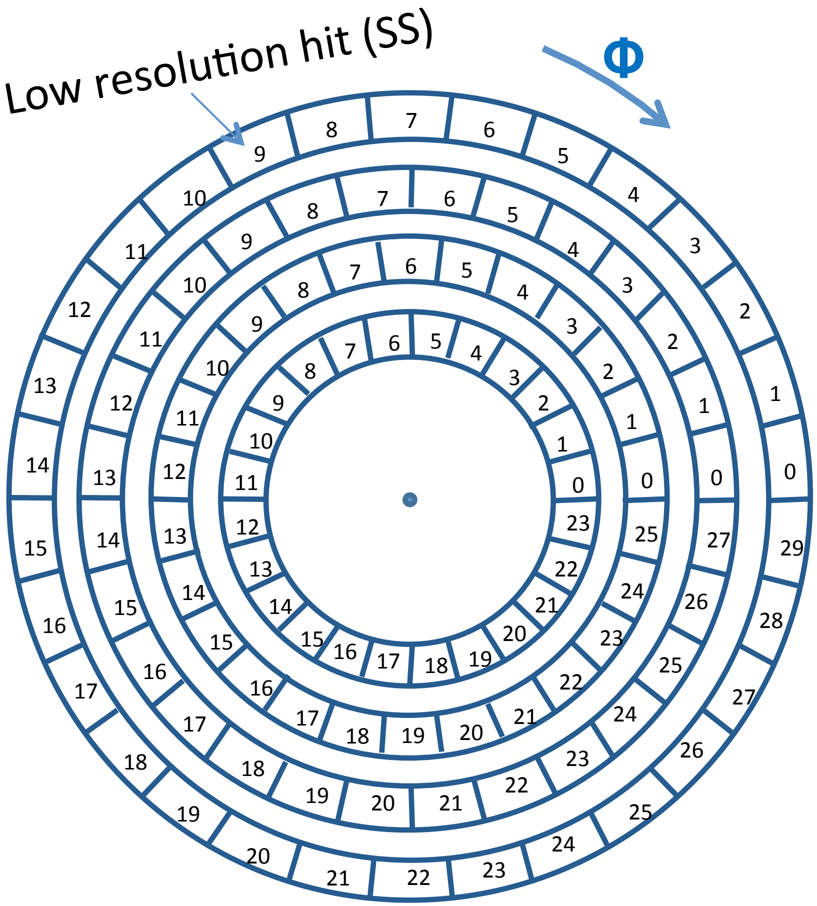
    12 → 1.2V (10A)
    3.3 → 1.2V (5A)
    } for FPGA
  - CLK sharing (200MHz).
  - 4 LVDS lines/FPGA for hold and reset signal from DF.
  - JTAG lines to download FW.
  - I^2C lines for slow control and monitoring functionality.
  - Additional 2GTP lines/FPGA for high speed.

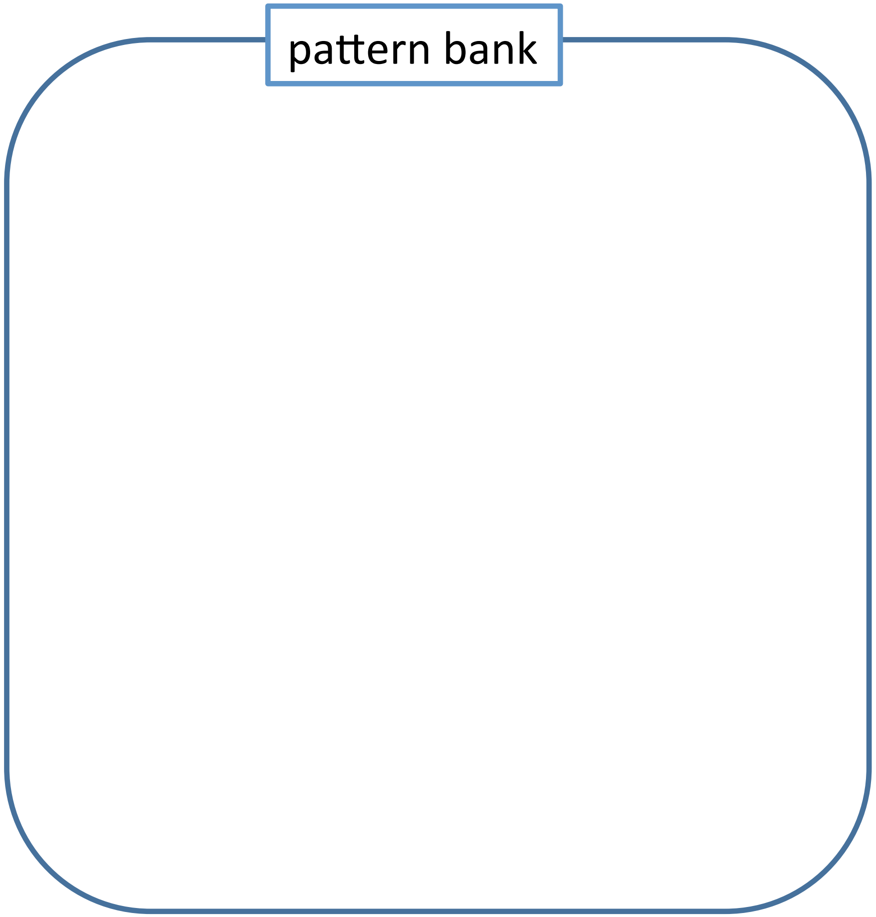


- FRAM & SRAM support IM functionality: e.g.) Store pseudo input data, etc.

FTK principle : Pattern Recognition

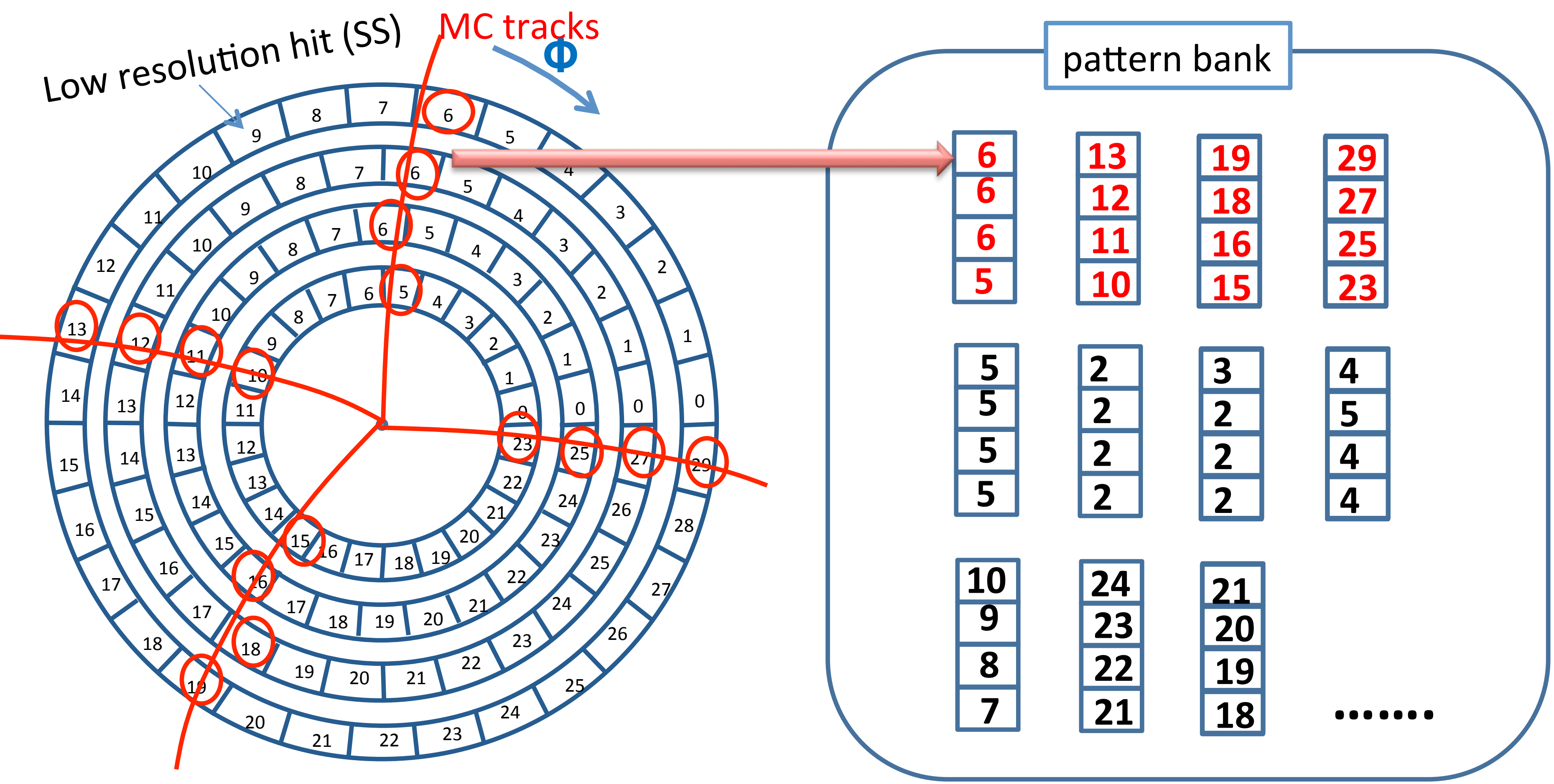
Find low resolution track called as "pattern".





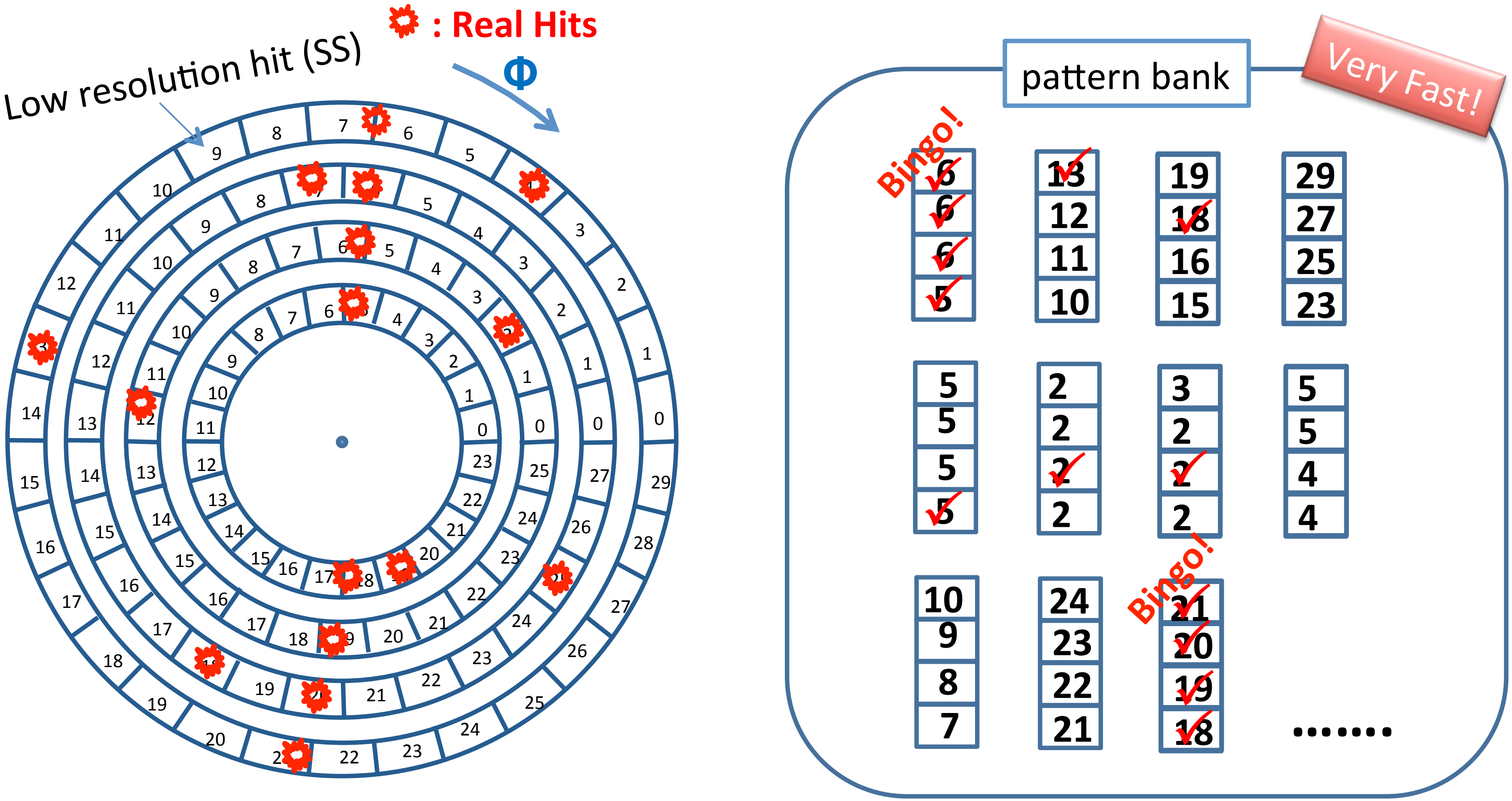
FTK principle : Pattern Recognition

1. Generate possible all patterns by **MC simulation**.
   ~300 M events of single muon event are used for pattern generation.



FTK principle : Pattern Recognition

2. Find the pattern in real data using **pattern recognition.**
   Real data's hits are sent to pattern bank sequentially, and patterns are recognized like a bingo game. All patterns are founded when all hits are arrived.



FTK principle : Track Fitting

Estimate the track parameters using full resolution hit info.

1. Pre-Calculate the Constants for the 5 parameter's **linear approximation** as a function of hit coordinate using MC simulation.
2. Estimate the track parameters using **linear approximation** equation with pre-calculated constant using real data's **full resolutions** hits coordinate.

$$\tilde{p_i} = \sum_{l=1}^{N} C_{il} x_l + q_i$$

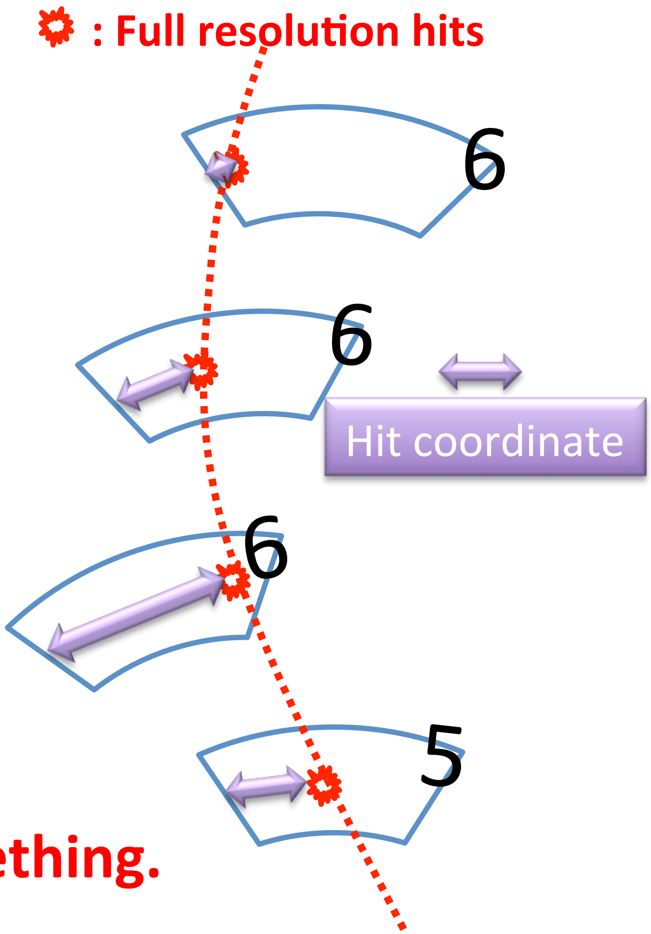
Parameter s

$\tilde{p_i}$: Track Parameters (i=0~4)

x_i : Hit Coordinate

$\vec{C_i}\ q_i$: Constant

5 parameters
d0, z0, eta, phi, p_T



FTK can estimate track information very fast without any looped process to minimize something.