BTeV Trigger and Data Acquisition Electronics

Joel Butler

Talk on Behalf of the BTeV Collaboration

10th Workshop on Electronics for LHC and Future Experiments

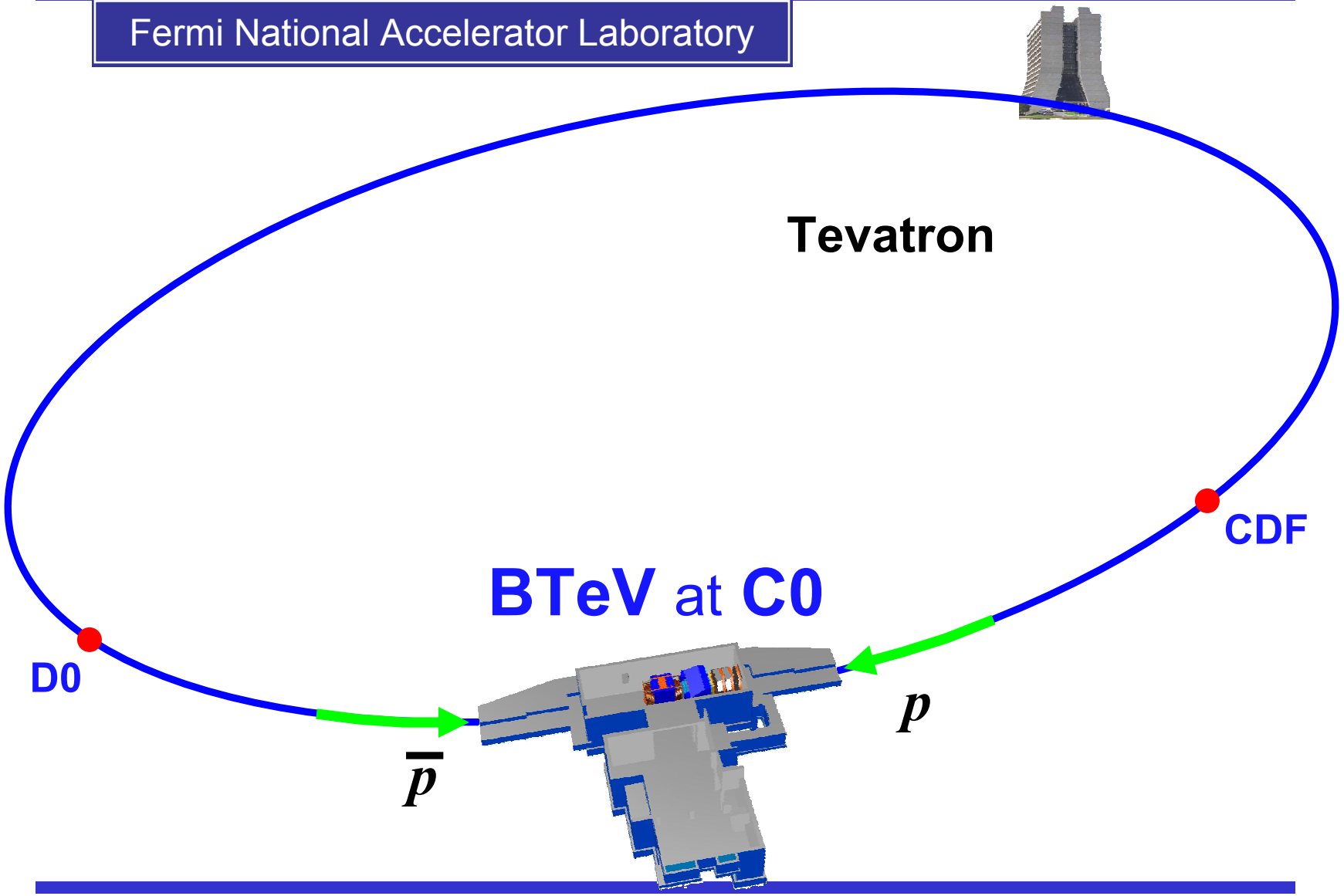
Sept. 13 –17, 2004

Outline

- Introduction and overview of BTeV
  - Physics Motivation
  - Expected Running Conditions
- The BTeV Spectrometer, especially the pixel detector
- Front-end electronics (very brief)
- BTeV trigger and data acquisition system (DAQ)
  - First Level Trigger
  - Level 2/3 farm
  - DAQ
  - Supervisory, Monitoring, Fault Tolerance and Fault Remediation
  - Power and Cooling Considerations
- Conclusion

BTeV - a hadron collider B-physics experiment

Fermi National Accelerator Laboratory



Why BTeV?

- **The Standard Model of Particle Physics fails to explain the amount of matter in the Universe**
  - **If matter and antimatter did not behave slightly differently, all the matter and antimatter in the early universe would have annihilated into pure energy and there would be no baryonic matter (protons, neutrons, nuclei)**
  - **The Standard Model of Particle Physics shows matter-antimatter asymmetry in K meson and B meson decays, but it predicts a universe that has about 1/10,000 of the density of baryonic matter we actually have**
  - **Looking among the B decays for new sources of matter-antimatter asymmetry is the goal of the next round of B experiments, including BTeV**
- **At $\mathcal{L} = 2 \times 10^{32}\ cm^{-2}\ s^{-1}$, ~ 4×10^{11} b hadrons (including B_s & Λ_b) will be produced per year at the Tevatron … vs. 2×10^{8} b hadrons (no B_s or Λ_b) in e^+e^- at the Υ(4s) with $\mathcal{L} = 10^{34}$.**
- **However, to take full advantage of this supply of B's, a dedicated, optimized experiment is required.**

The Tevatron Operating Parameters for BTeV

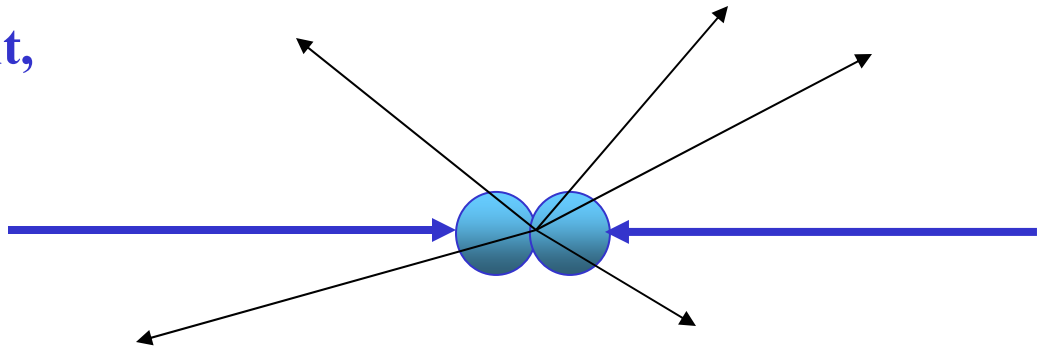
The requirements on the electronics are dominated by the operating conditions of the Tevatron and the physics goals of BTeV.

Luminosity	$2x10^{32}$
# of B - anti B pairs/10^7 s	$2x10^{11}$
# interaction/s	$15x10^6$
# of B events/background event	1/500 (1/500,000 interesting B decay)
Bunch spacing	396 ns (originally 132 ns)
Luminous region length	$\sigma_z = 30$ cm
Luminous region radius	$\sigma_x \sim \sigma_y \sim 30$ μm
#interactions/beam crossing	<6.0>

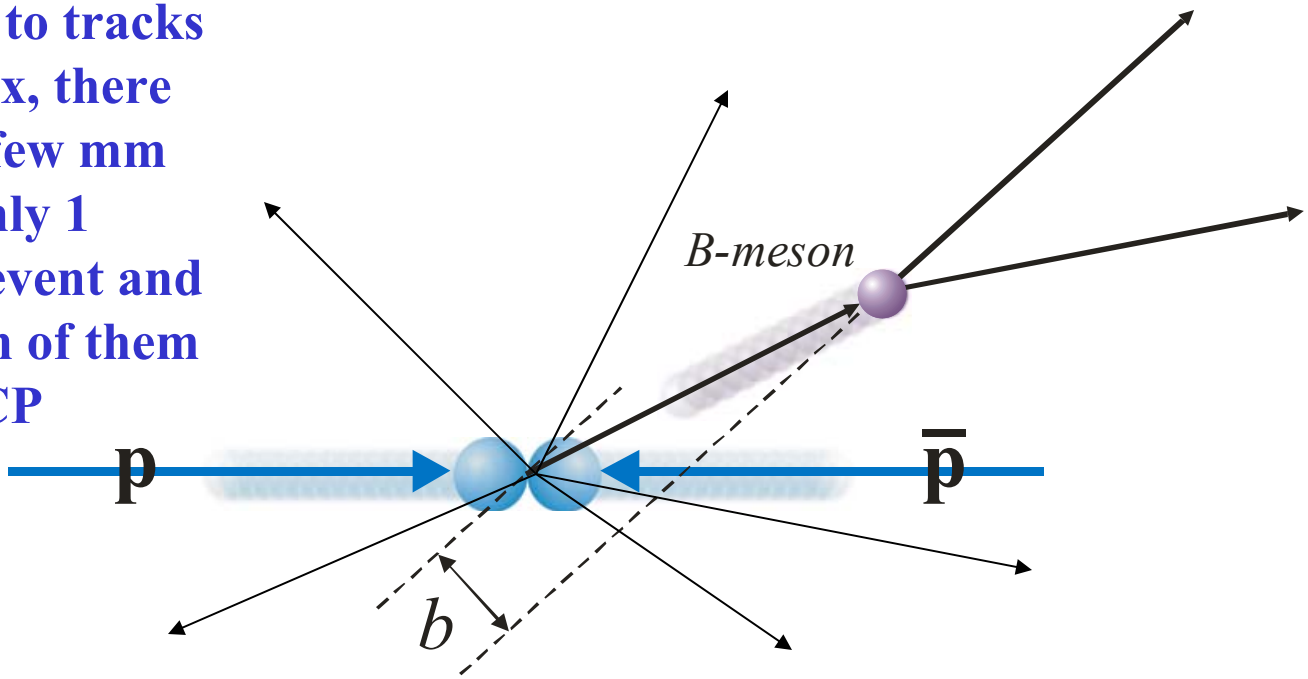
BTeV was originally designed for 132 ns crossing interval, the original Run 2 specification. Since the design was quite advanced when it was decided to stay at 396ns, BTeV is still capable of operation at 132 ns.

Basis of the Trigger

Minimum Bias event, all tracks from Interaction Vertex



B event, in addition to tracks at Interaction Vertex, there is a B that travels ~few mm and then decays. Only 1 Event in 500 is a B event and only a small fraction of them are interesting for CP violation



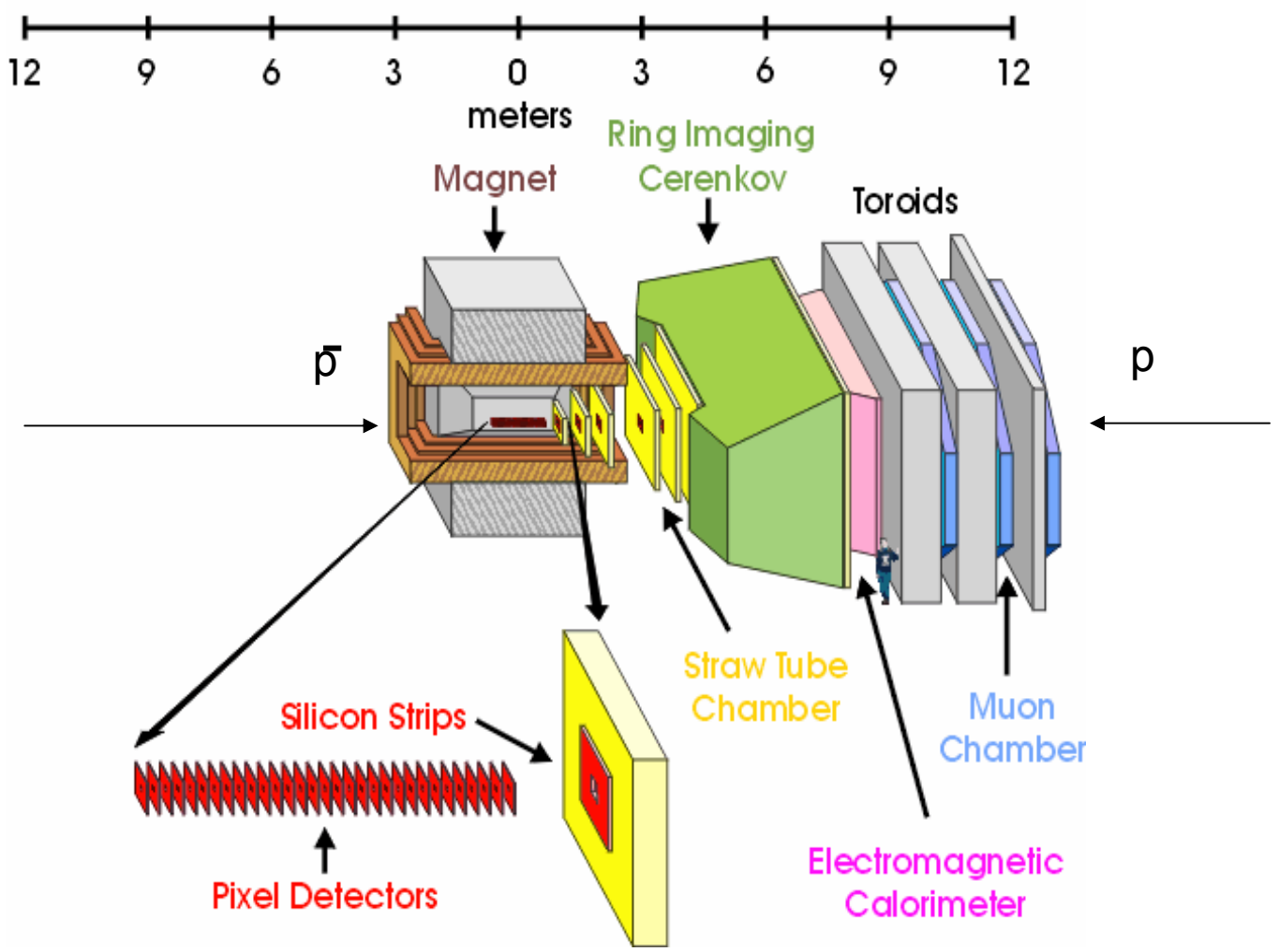
Challenge for Electronics, Trigger, and DAQ - I

- **We want to select events (actually crossings) with evidence for a "downstream decay", a.k.a "detached vertex."**
  - **THIS REQUIRES SOPHISTICATED TRACK AND VERTEX RECONSTRUCTION AT THE LOWEST LEVEL OF THE TRIGGER**
- To carry out this computing problem, we must
  - **Provide the cleanest, easiest-to-reconstruct input from the vertex tracking system – hence the silicon Pixel Detector**
  - **We have to expect that the computations will take a highly variable amount of CPU and real time, due to different numbers of interactions in the crossing, particles in each interaction, and variable multiplicity (2 to >10) in the B decay.**
    - **We have to employ a massively parallel pipelined archtecture which produces an answer every 396 (132) ns**
    - **We allow a high and variable latency and abandon any attempt to preserve time ordering. Event fragments and partial results are tied together through time stamps indicating the "beam crossing."**
    - **We control the amount of data that has to be moved and stored by sparsifying the raw data just at the front ends and buffering it for the maximum latency expected for the trigger calculations**
    - **We have a timeout, to recover from hardware or software glitches, that is an order of magnitude longer than the average computation time.**

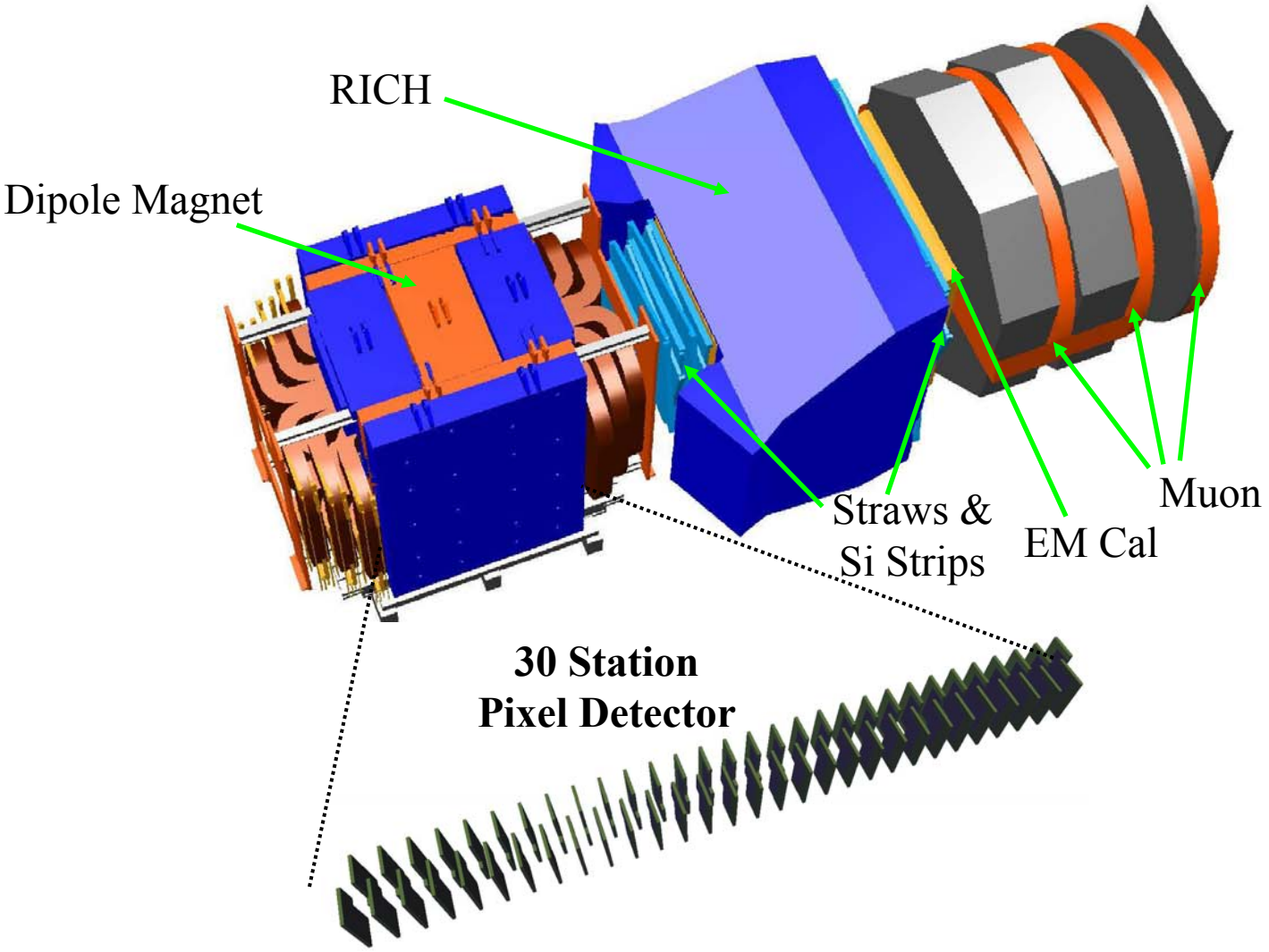
Challenge for Electronics, Trigger, and DAQ - II

- Monitoring, Fault Tolerance and Fault mitigation are crucial to having a successful system
  - They must be part of the design, including
    - **Software;**
    - **Dedicated hardware; and**
    - **A committed share of the CPU and storage budgets on every processing element of the system**
- For modern, massively parallel computation systems made out of high speed commercial processors, providing adequate power and cooling are major challenges
- Simulation is a key to developing a successful design and implementation, including
  - GEANT detector simulation data that can be used to populate input buffers to exercise hardware
  - Mathematical modeling
  - Behavioral Modeling

BTeV Forward Dipole Spectrometer



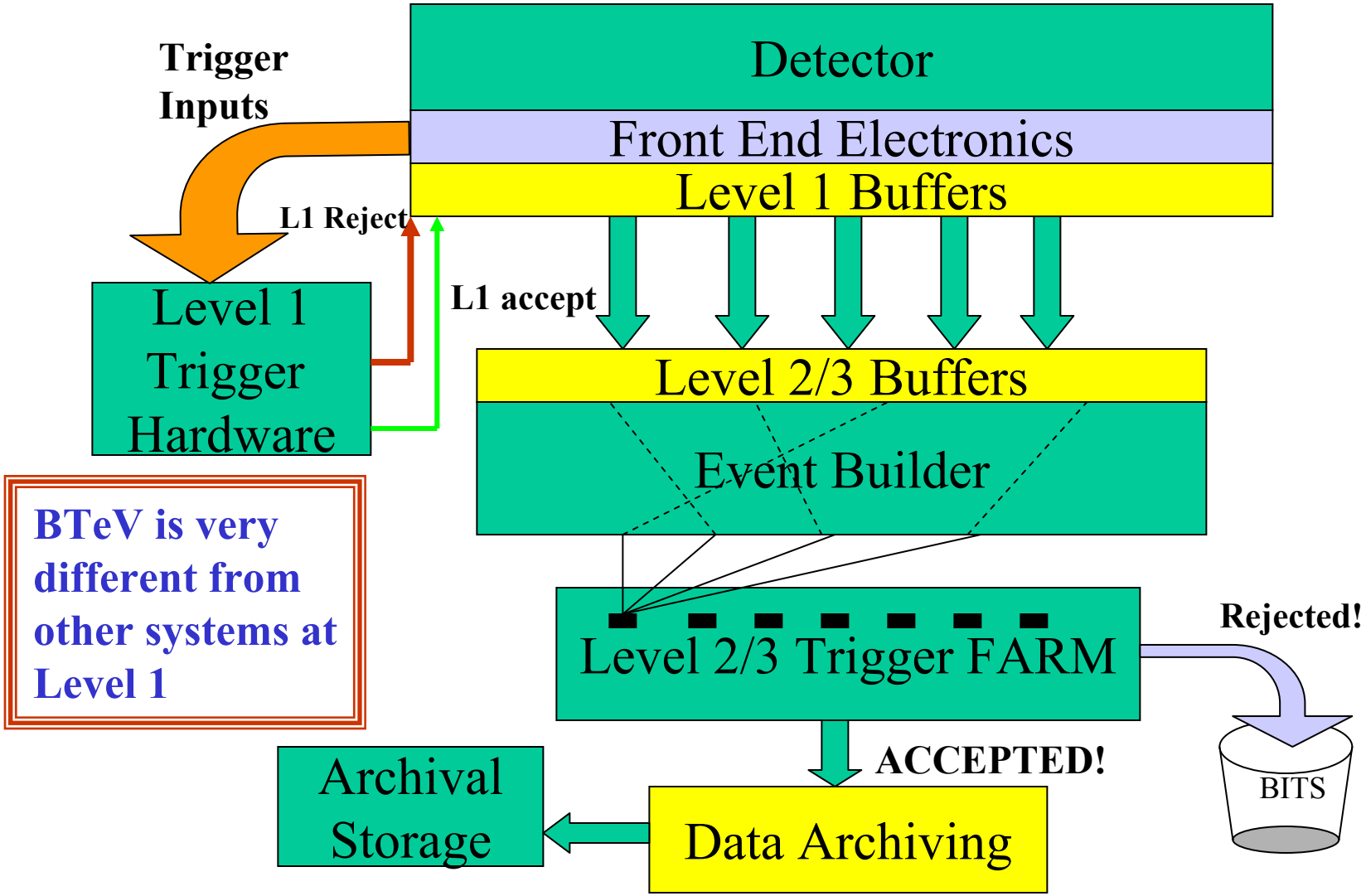
BTeV detector



Key Design Features of BTeV

- A precision vertex detector of **planar pixel arrays on the IR**
- A **dipole located ON the IR enables the pixel detector to make a standalone momentum measurements for the trigger**
- A **vertex trigger at Level I** which makes BTeV especially efficient for B decays. The tracking system design has to be tied closely to the trigger design to achieve this.
- Strong particle identification based on a **Ring Imaging Cerenkov counter.** Many states emerge from background only if this capability exists. It enables use of charged kaon tagging.
- **A lead tungstate electromagnetic calorimeter for photon and π^0 reconstruction.**
- A very **high capacity data acquisition system** which frees us from making excessively restrictive choices at the trigger level

General Structure of a Data Acquisition System



Requirements for BTeV Trigger and DAQ

- **The challenge for the BTeV trigger and data acquisition system is to reconstruct particle tracks and interaction vertices for EVERY interaction that occurs in the BTeV detector, and to select interactions with *B* decays.**
- **The trigger performs this task using 3 stages, referred to as Levels 1, 2 and 3:**
  "L1" – looks at every interaction, and rejects at least 98% of background
  "L2" – uses L1 results & performs more refined analyses for data selection
  "L3" – performs a complete analysis using all of the data for an interaction

Reject > 99.8% of background. Keep > 50% of *B* events.

- **The DAQ saves all of the detector data in memory for as long as is necessary to analyze each interaction (~ 1 millisecond on average for L1), and moves data to L2/3 processing units and archival storage for selected interactions.**

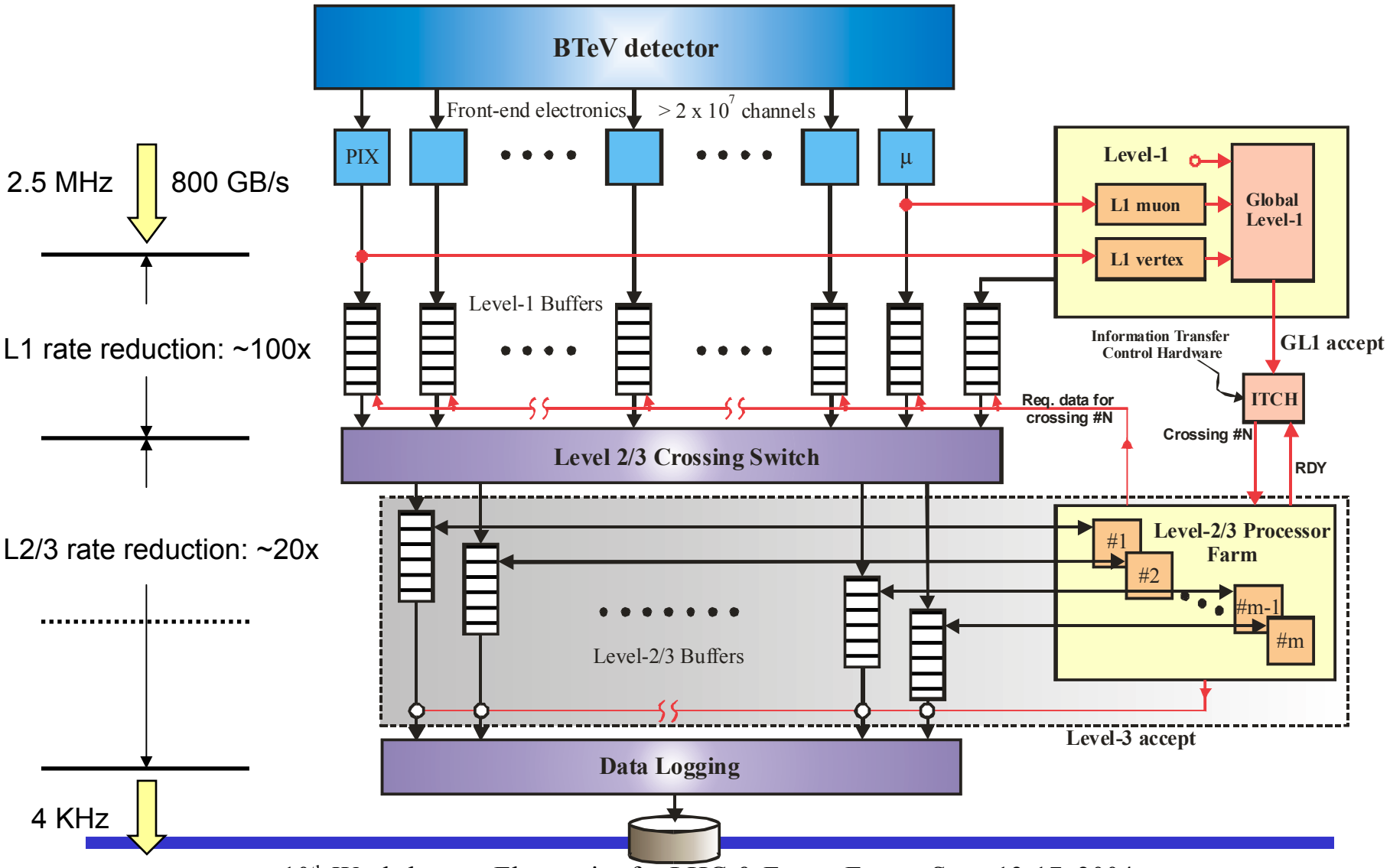
- **The key ingredients that make it possible to meet this challenge:**
  - **BTeV pixel detector with its exceptional pattern recognition capabilities**
  - **Rapid development in technology and lower costs for – FPGAs, DSPs, microprocessor CPUs, memory**

Key Issues in Implementation of Vertex Trigger

- We will be doing event reconstruction at the lowest level of the trigger. The highly variable complexity of different crossings means that there will be a wide spread in the times it talks to process them.
- To have any chance of succeeding, must give the trigger system the best possible inputs: high efficiency, low noise, simple pattern recognition.
- To have an efficient system, we must avoid idle time. This means pipelining all calculations and to have efficient pipelining:
  - **No fixed latency**
  - **No requirement of time ordering**
- In turn, this implies
  - **Massive amounts of buffering, and**
  - **On the fly sparsification in the front ends at the beam crossing rate**

BTeV sparsifies all data at the front ends and stores them in massive buffers (Tbyte) while waiting for the trigger decision. Event fragments are gathered using time stamps.

BTeV trigger overview



Pixel Vertex Detector

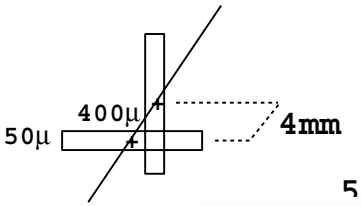
Reasons for Pixel Detector:

- Superior signal to noise
- Excellent spatial resolution -- 5-10 microns depending on angle, etc
- Very low occupancy
- Very fast
- Radiation hard

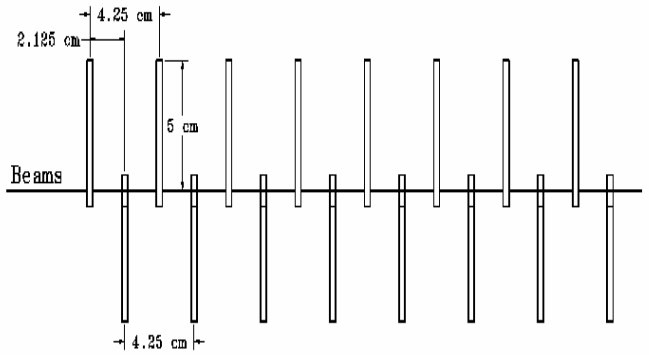
Special features:

- It is used directly in the Level 1 trigger
- Pulse height is measured on every channel with a 3 bit FADC
- It is inside a dipole and gives a crude standalone momentum

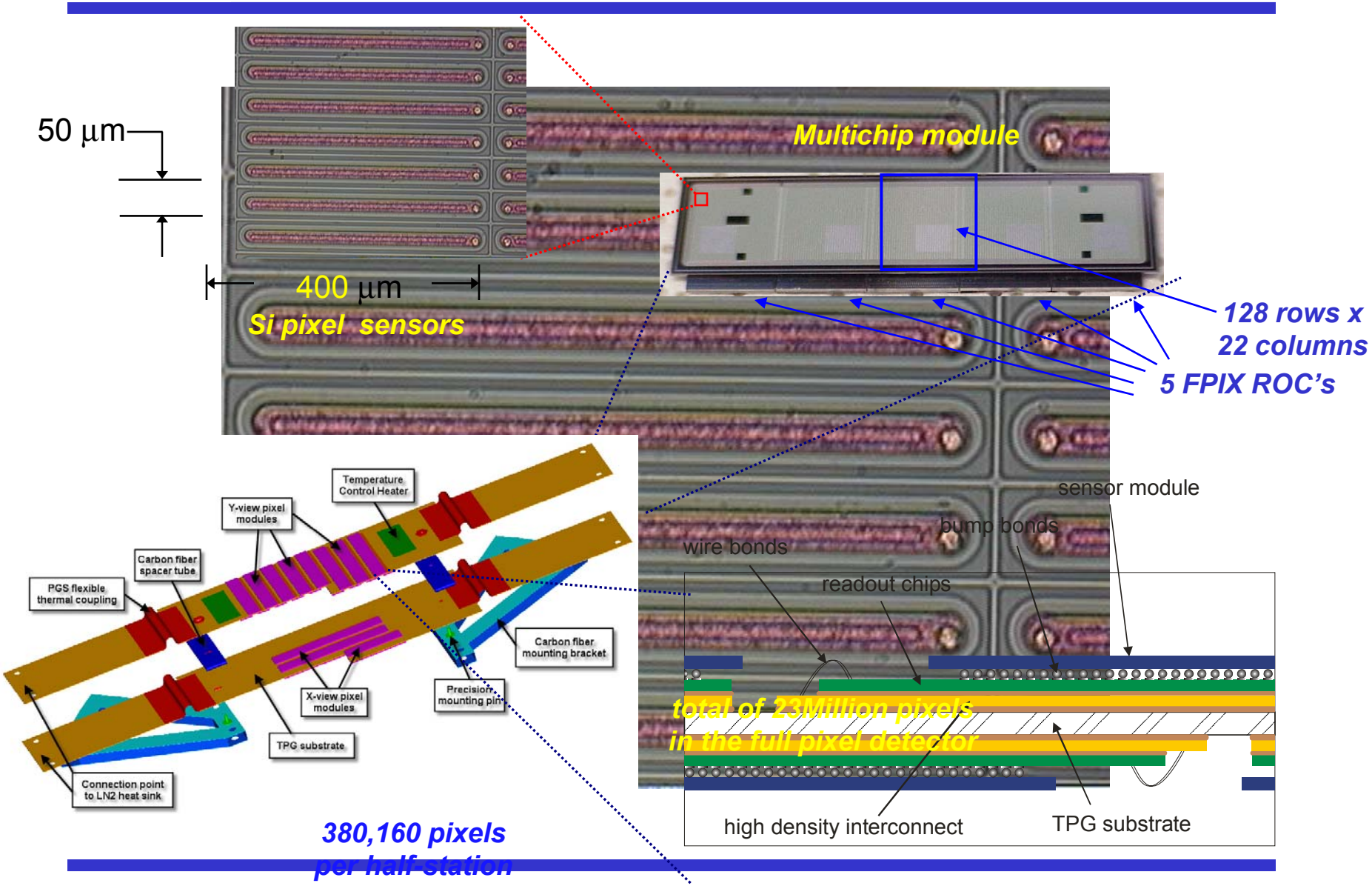
The BTeV Baseline Pixel Detector



Pixel Orier



Si Pixel Detector Sensor and Readout Chip



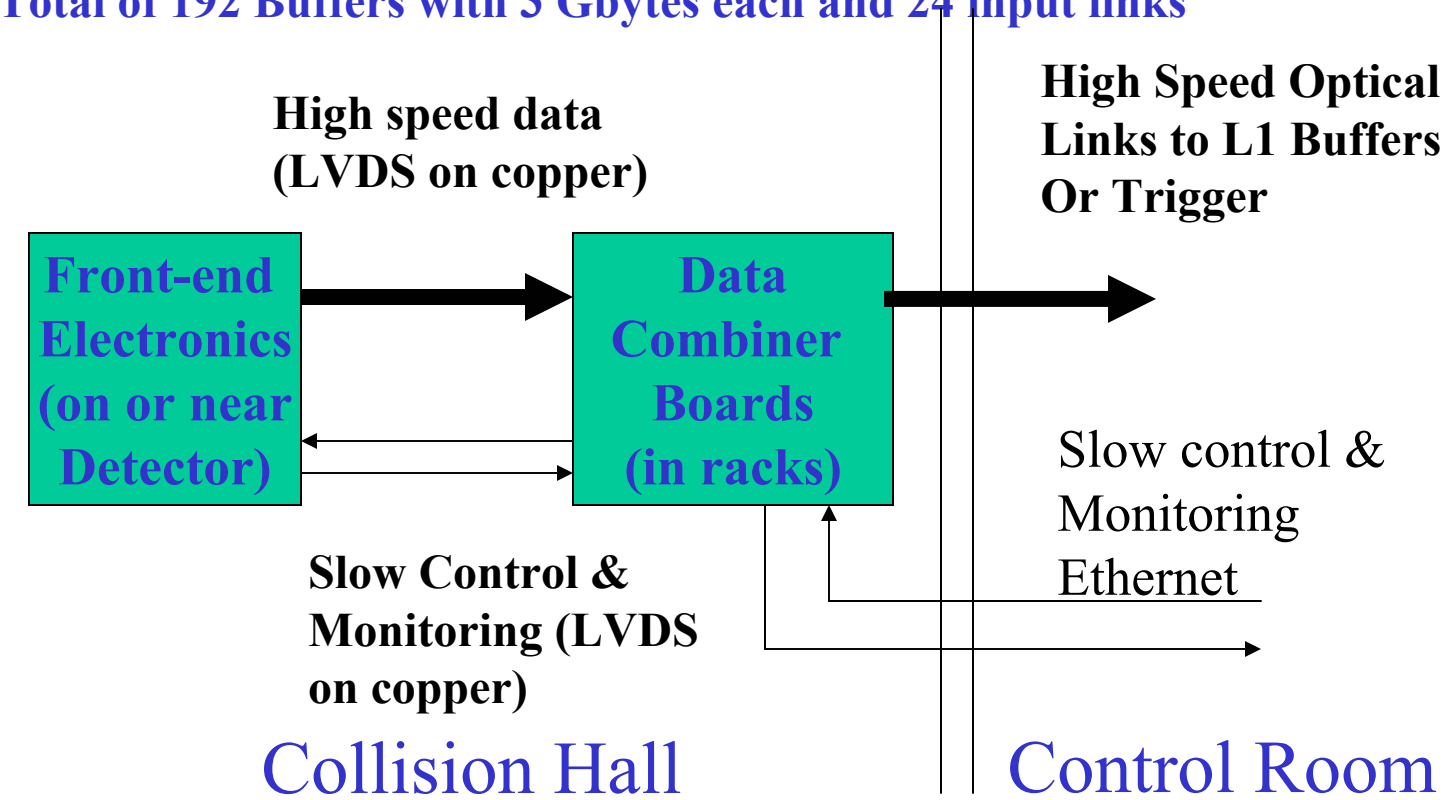
Other Custom Chips for Front Ends

- Pixel (23M channels)
  - **FPIX2* (0.25μ CMOS): pixel readout Chip, 22x128pixels, 8100 chips**
- Forward Silicon Microstrip Detector (117,600 channels)
  - **FSSR* IC s (0.25μ CMOS): 1008 chips**
    - **FSSR uses same readout architecture as FPIX2**
- Straw Detector (26,784 straws x ~2)
  - **ASDQ Chips(8 channel) (MAXIM SHPi analog bipolar): 6696 chips, 2232 boards each with 3 ASDQs**
  - **TDC Chips*(24 channels) (0.25μ CMOS): 2232 chips, 1116 cards each with 2 chips**
- Muon Detector (36,864 proportional tubes)
  - **ASDQ Chips(8 channel): 4608 chips**
- RICH (Gas Rich: 144, 256 channels; Liquid RICH, 5048 PMTs)
  - **RICH Hybrid*(128 channels): 1196+80**
  - **RICH MUX*(128 channels): 332 + 24**
- EMCAL (10100 channels)
  - **QIE9*(1 channel)(AMS 0.8μ BiCMOS): 10100 chips, 316 cards of 32 channel**

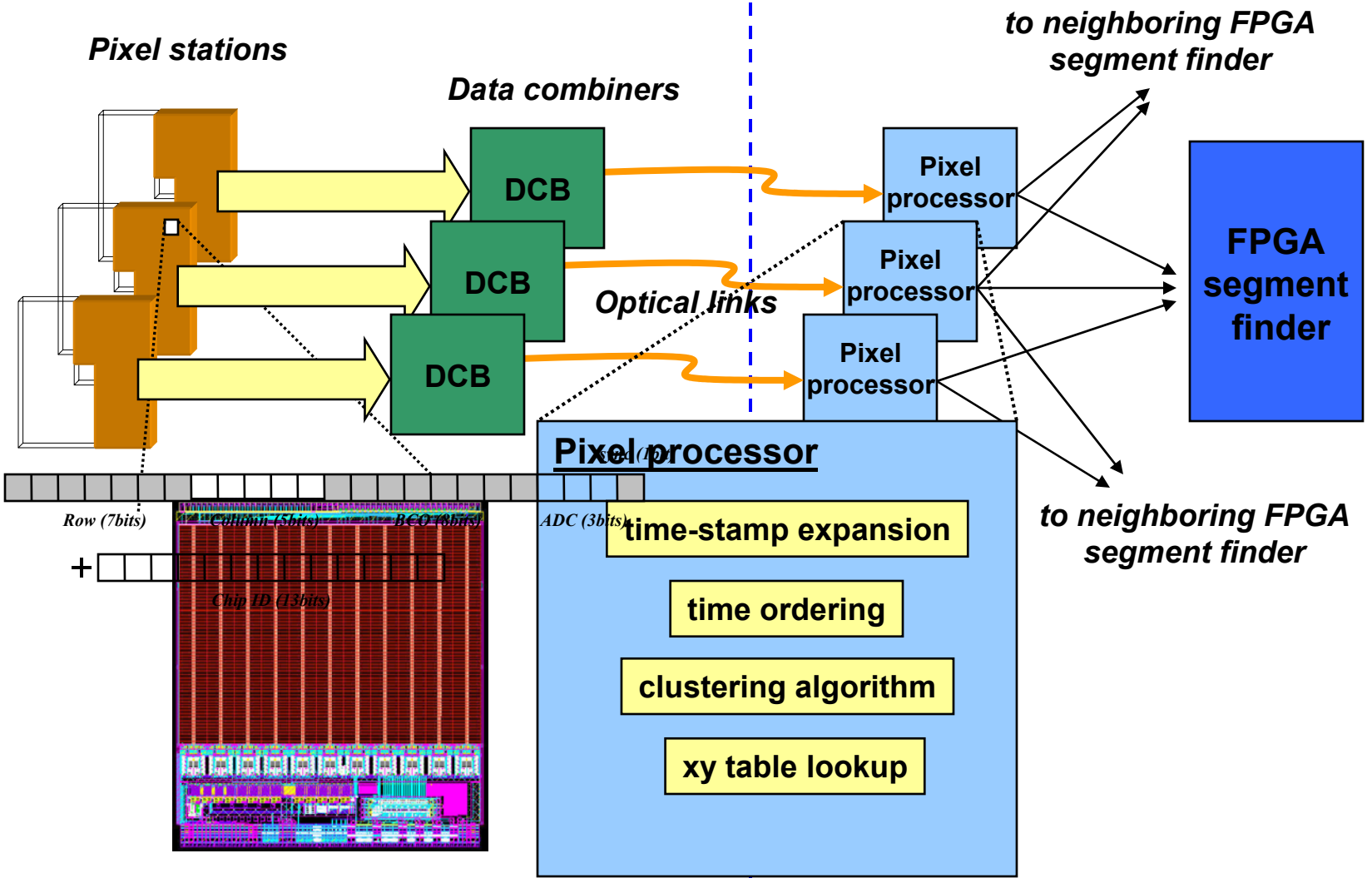
***= new chip**

Key DAQ Custom Boards

- Data Combiner Board (DCB), 24 input channels (2.5 Gb/s) multiplexes data onto serial fiber links to send to control room
  - **Total of 364**
- Level 1 Buffer (L1B)
  - **Total of 192 Buffers with 3 Gbytes each and 24 input links**



Pixel data readout

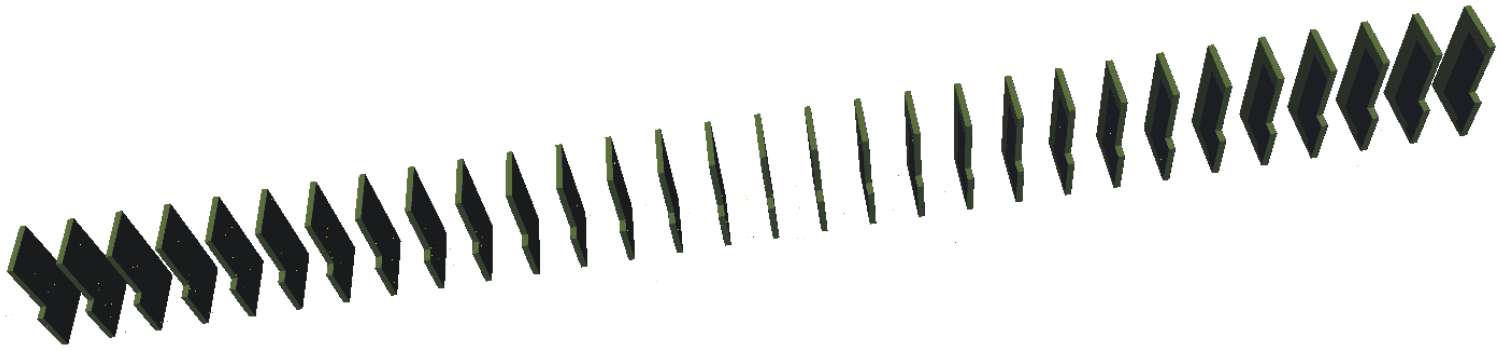


FPIX2 Read-out chip

L1 vertex trigger algorithm

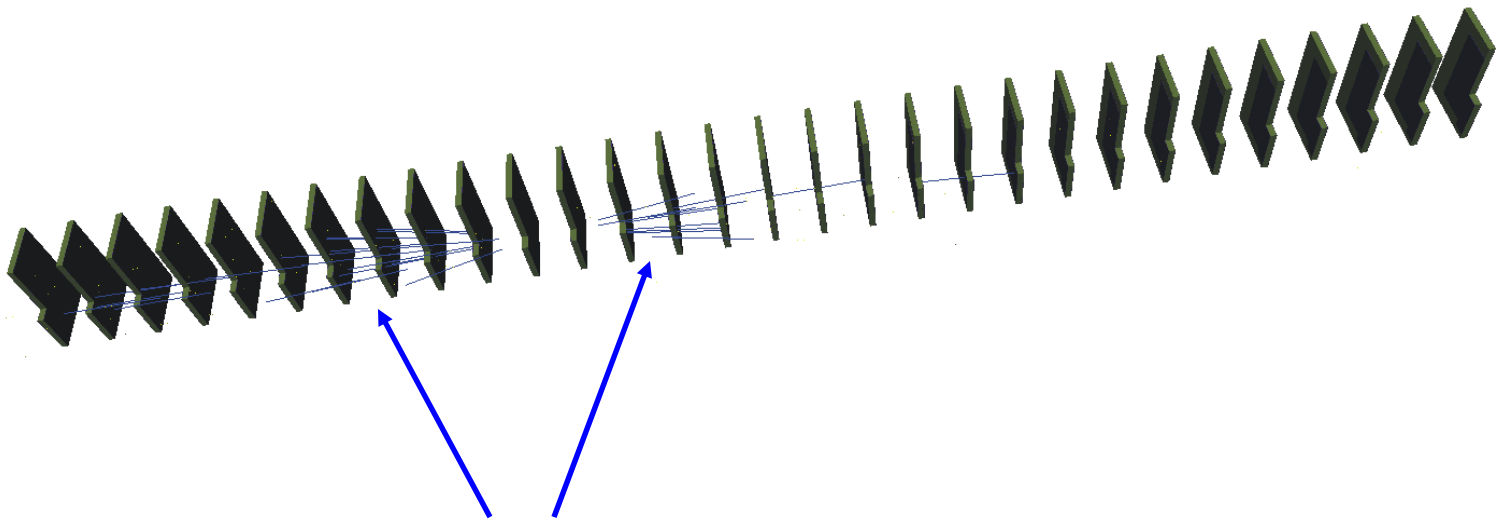
Two stage trigger algorithm:

1. ***Segment finding***
2. ***Track/vertex finding***



1) <u>Segment finding stage:</u>
Use pixel hits from 3 neighboring stations to find the beginning and ending segments of tracks. These segments are referred to as triplets

Segment finding: inner triplets

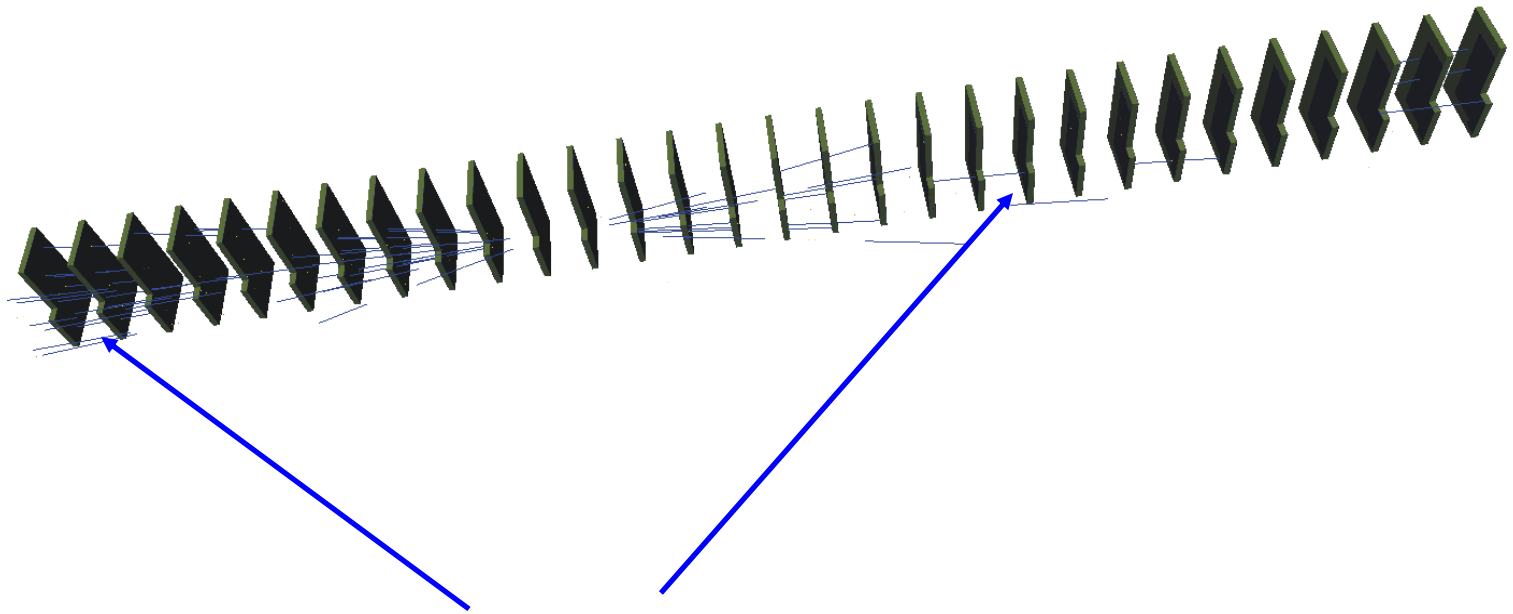


1a) <u>Segment finding stage: phase 1</u>

Start with inner triplets close to the interaction region.

An inner triplet represents the start of a track.

Segment finding: outer triplets



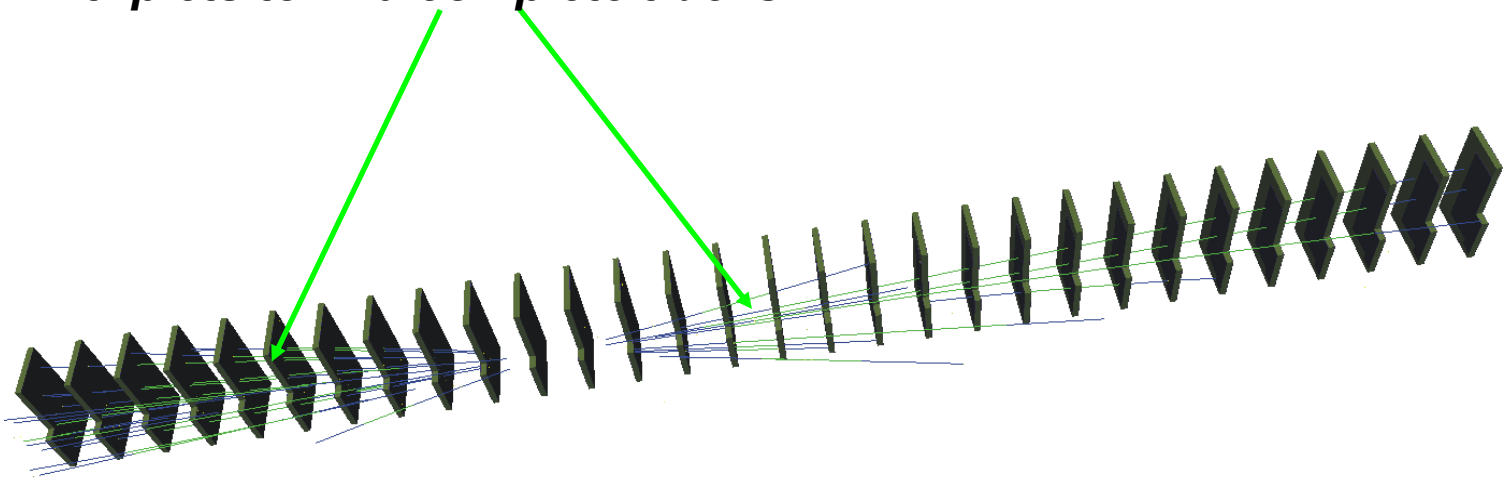
1b) <u>Segment finding stage: phase 2</u>

Next, find the outer triplets close to the boundaries of the pixel detector volume. An outer triplet represents the end of a track.

Track/vertex finding

2a) <u>Track finding phase:</u>

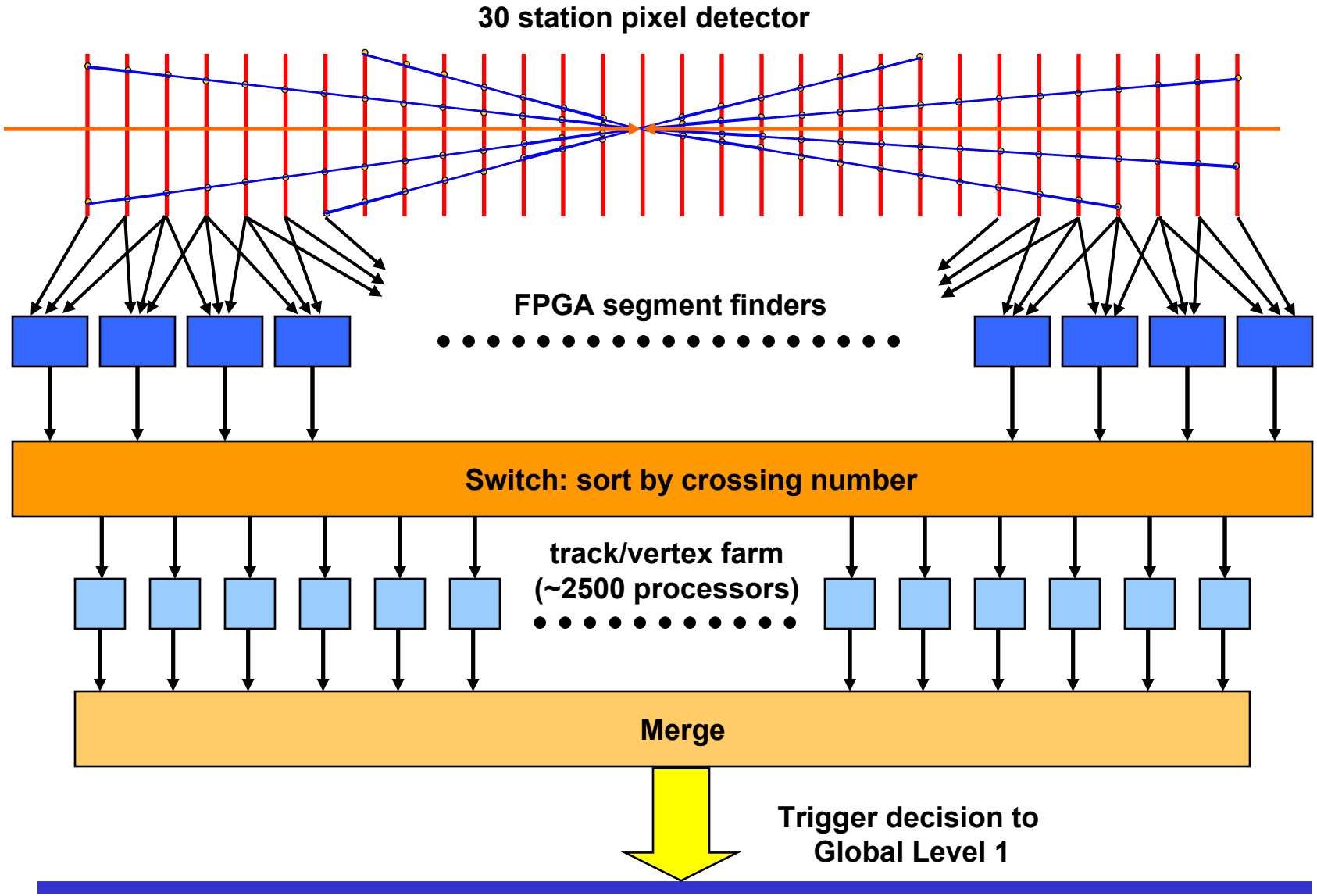
Finally, match the inner triplets with the outer triplets to find complete tracks.



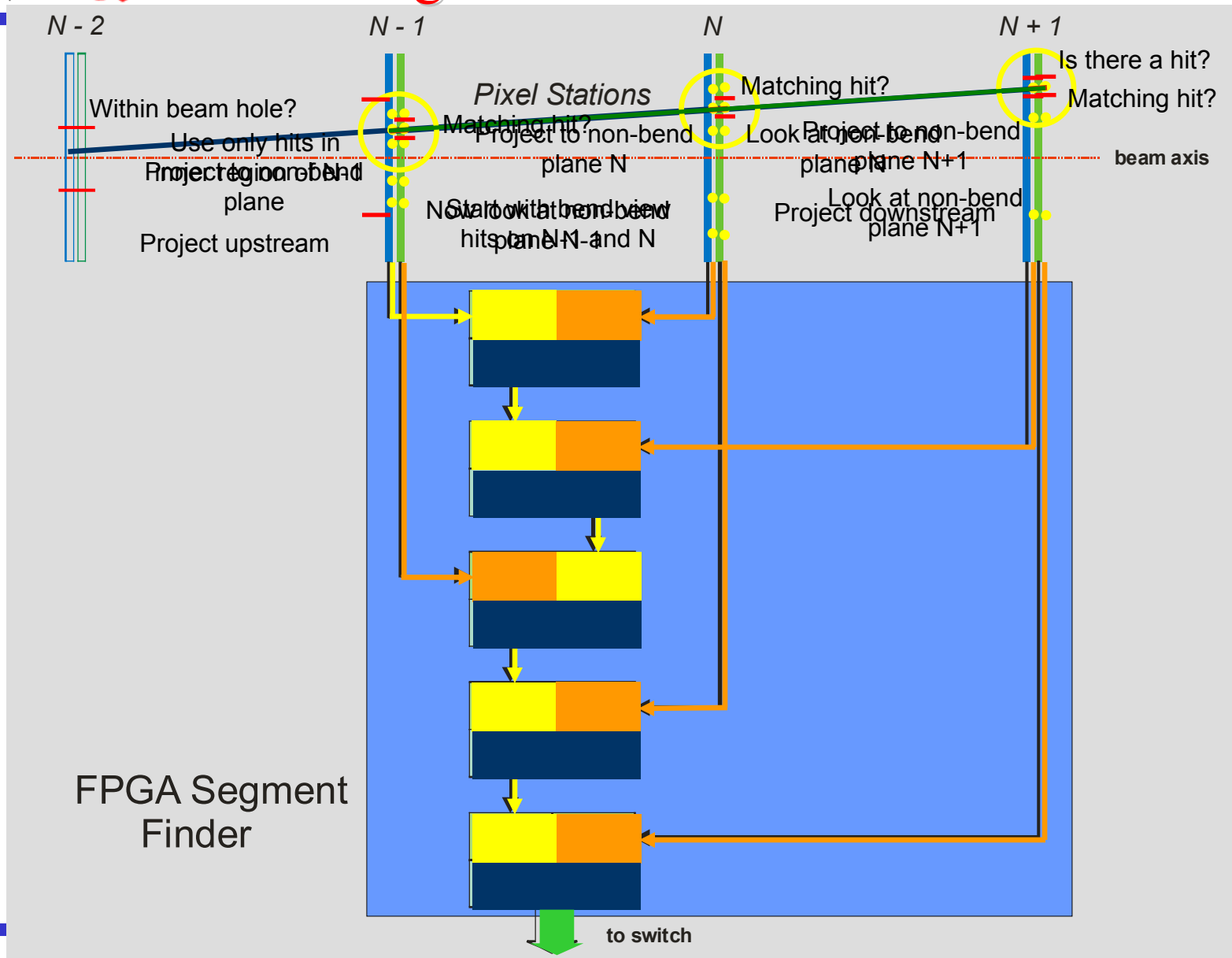
2b) <u>Vertex finding phase:</u>

- ***Use reconstructed tracks to locate interaction vertices***
- ***Search for tracks detached from interaction vertices***

Level 1 Vertex Trigger

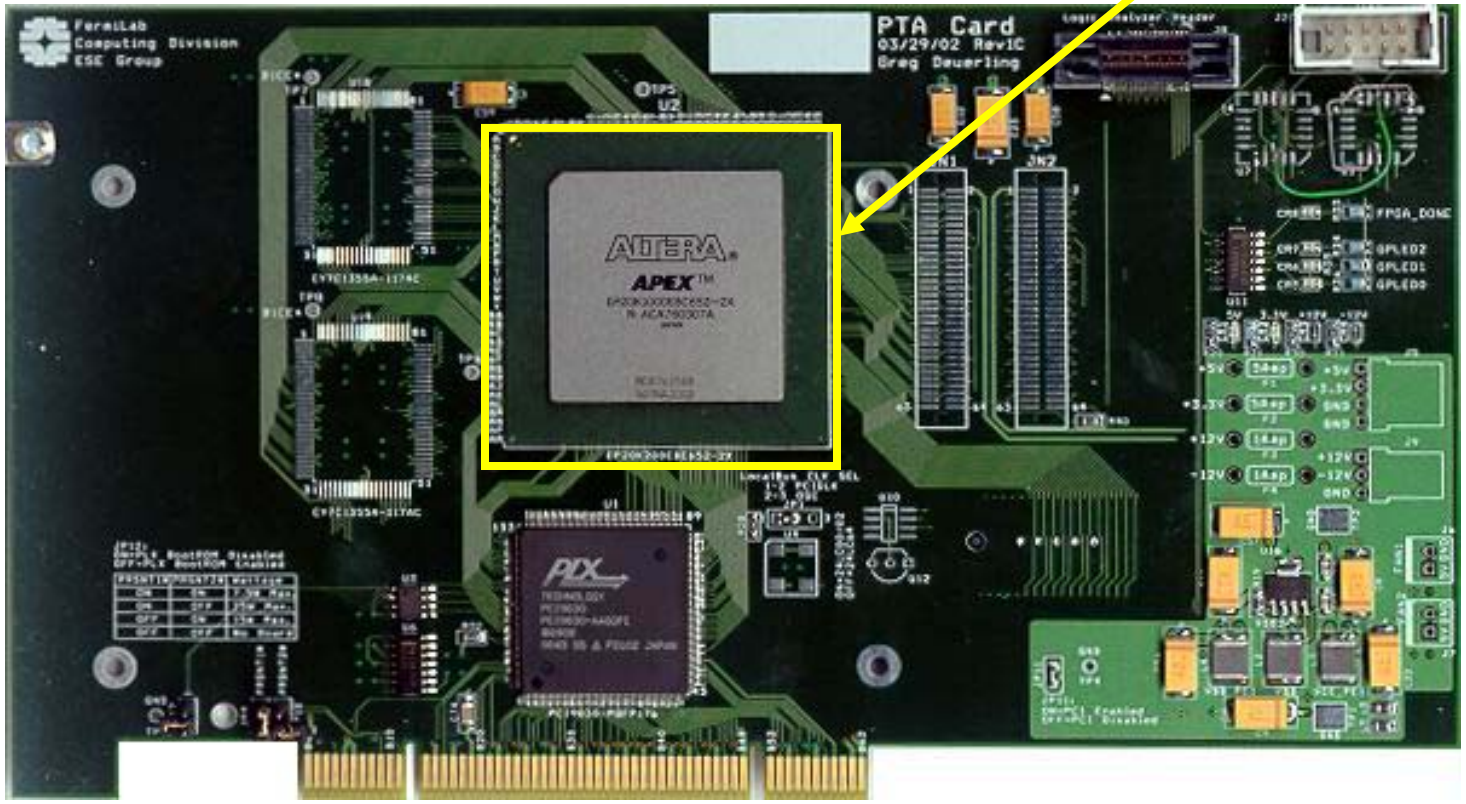


L1 segment finder hardware



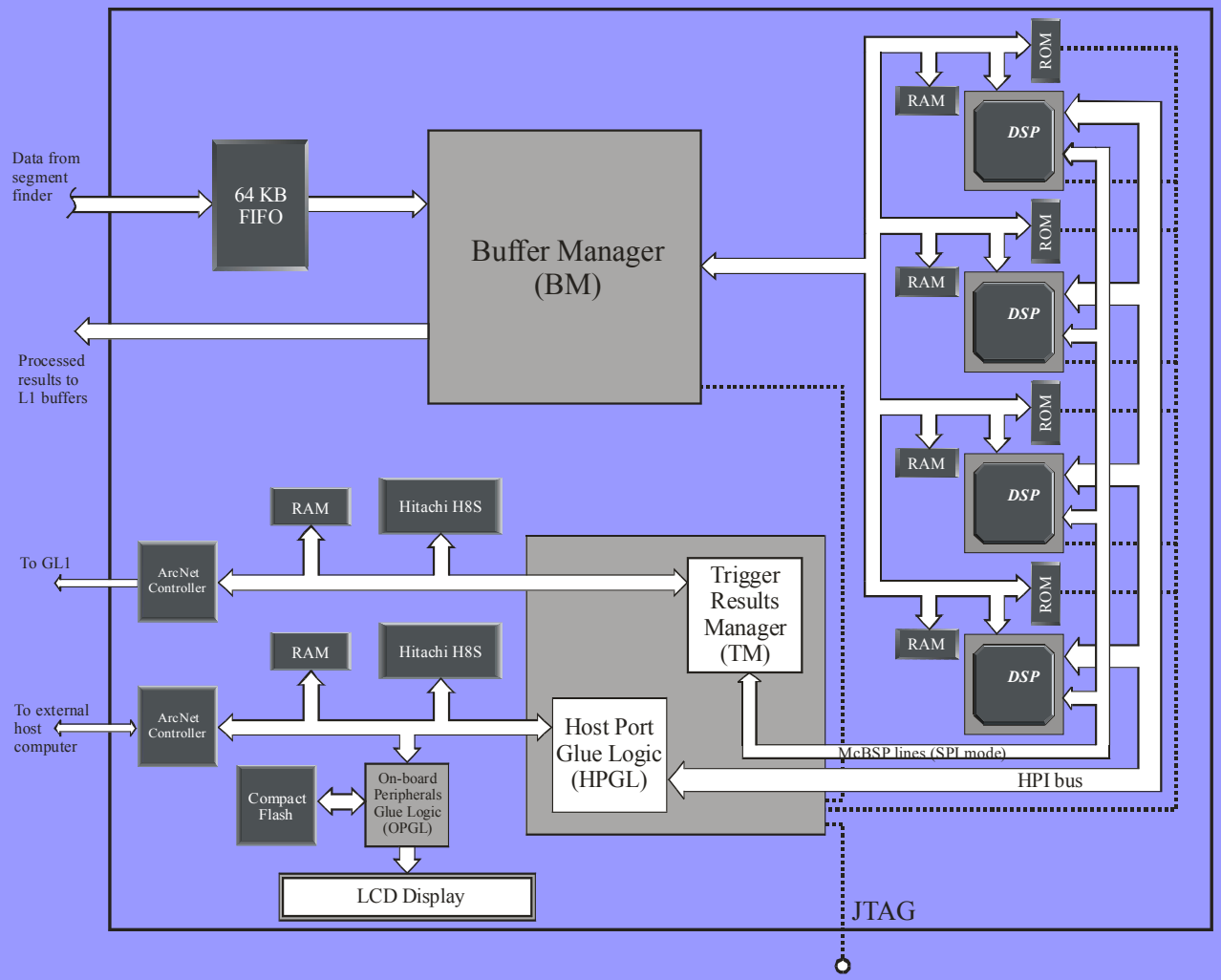
L1 segment finder on PTA card

Uses Altera APEX EPC20K1000 instead of EP20K200 on regular PTA



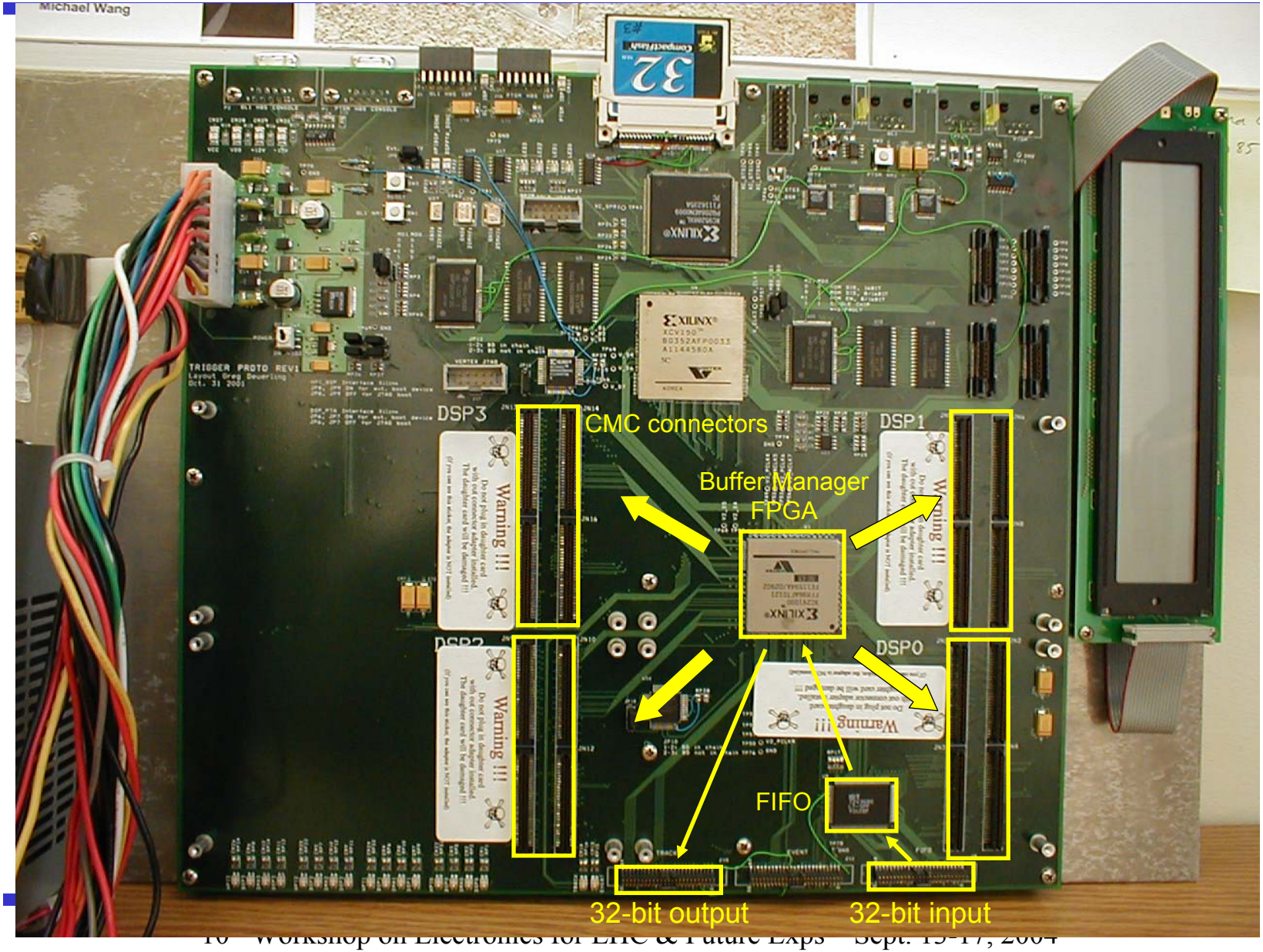
Modified version of PCI Test Adapter card developed at Fermilab for testing hardware implementation of 3-station segment finder (a.k.a. “Super PTA”)

L1 track/vertex farm hardware



Block diagram of pre-prototype L1 track/vertex farm hardware

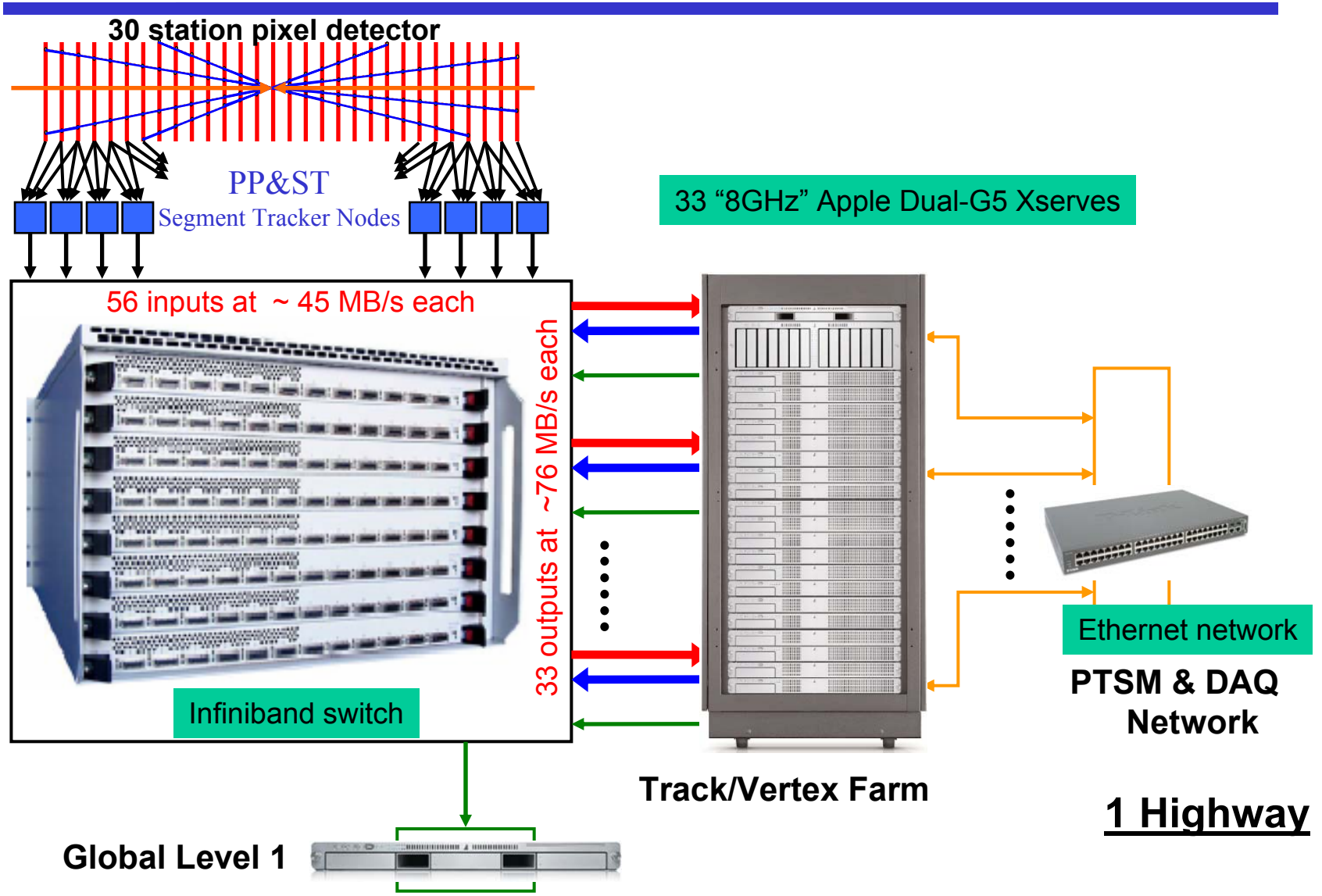
L1 trigger pre-prototype board



Highways

- Original BTeV system was criticized by because it required development of a high-speed **custom** switch, because of the need to handle data every 132 ns (original crossing interval) and not due to total throughput. Easy to underestimate the complexity and risk of a home-grown switch and associated software.
- **By dividing system into N~6-8 "highways", each system must deal with intervals of only Nx396 (132) ns and then switches based on commercial networking gear will work!!!**
- All Data paths must be available to each highway, but now they can be lower speed links.There are eight times more slow links but the cost is about the same.
- We first implemented highways for event building into Level 2/3 and went to commercial network gear there
- We have now demonstrated that we can use a commercial switch within Level 1 to sort the "track segments" according to time stamp
- We now have the system divided into highways through all the various trigger levels
- Large amounts of home grown hardware and software ELIMINATED

L1 Trigger Architecture (1 Highway)



Estimate of Computing Power Required for L1

Conservative estimate of computing power required for L1 Farm

- Assume an "8.0 GHz" G5 Xserve (compared to 2.0 GHz available today):
  - L1 track/vertex code (C code without any hardware enhancements like hash-sorter or FPGA segment-matcher) takes 379 µs/crossing on a 2.0 GHz Apple G5 for minimum bias events with <6> interactions/crossing
  - Include additional capacity:
    - L1 code: 50%, RTES: 10%, spare capacity: 40%
    - 379 µs + 76 µs + 303 µs = 758 µs
  - 758 µs ÷ 4 = 190 µs on a "8.0 GHz G5"
  - Include an additional 10% processing for L1-buffer operation
    - 190 µs + 19 µs = 209 µs

Industry news (rumor) suggests the availability of a 3GHz dual-core G5 next year. This is essentially a 6GHz G5.

- Number of "8.0 GHz" G5's needed for L1 track/vertex farm:
  - 209 µs/396 ns = 528 G5's for all 8 highways, 66 G5's per highway
  - 33 Dual G5's per highway

L1 trigger efficiencies

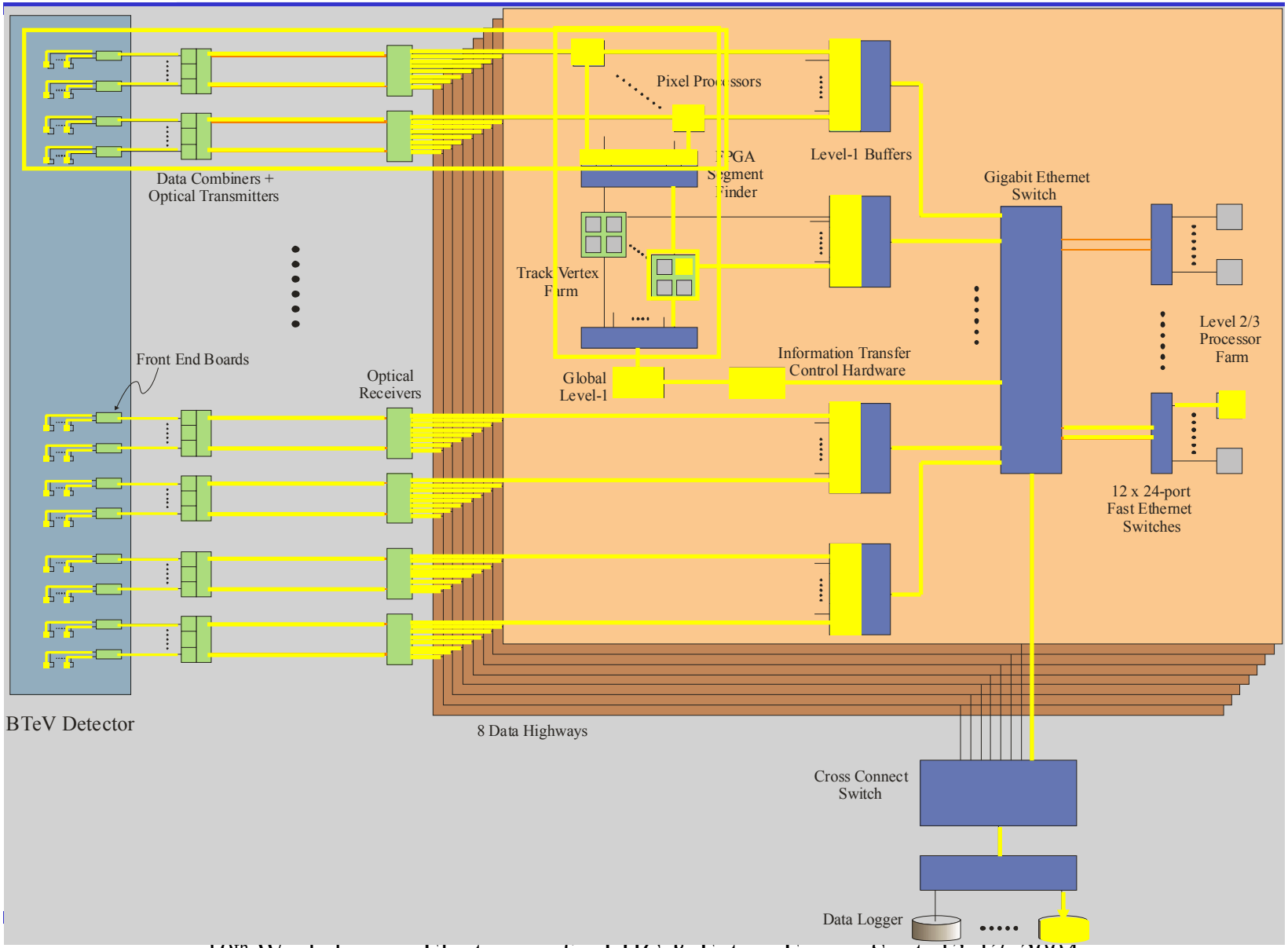
Process	Efficiency
Minimum bias	**1%**
$B_s \rightarrow D^+_s K^-$	80%
$B^0 \rightarrow J/\psi K_s$	65%
$B^- \rightarrow K_s \pi^-$	45%
$B^- \rightarrow \phi K_s$	74%
$B^0 \rightarrow$ 2-body modes $(\pi^+\pi^-, K^+\pi^-, K^+K^-)$	80%

L1 vertex trigger efficiencies

Global Level 1

- The actual Level 1 trigger is more complicated
  - The main physics Level 1 Vertex Trigger
  - Several other Vertex Triggers with various cuts relaxed to monitor how they turn on (generated in the same calculation as the main Vertex Trigger)
  - A standalone DIMUON trigger provides a continuous check on the Level 1 Vertex Trigger. This comes from a separate MUON trigger based on the same hardware but different inputs and computer programs.
  - Zero bias (crossing clock only) and minimum bias triggers
  - Calibration triggers
- Global Level 1 is a computer farm that receives "data packets" from the various sources and buffers them until it has them all. It then analyzes them against multiple lists of trigger criteria. After applying pre-scales, if there is a trigger, it sends a Level 1Accept and adds the crossing number to a list of crossings that satisfy "Level 1".
- When a L2/L3 processor asks for an crossing satisfying a given type of trigger, GL1 sends the next crossing number.
- GL1 maintains several lists to provide "partitioning", useful for commissioning and debugging.

BTeV Trigger Architecture



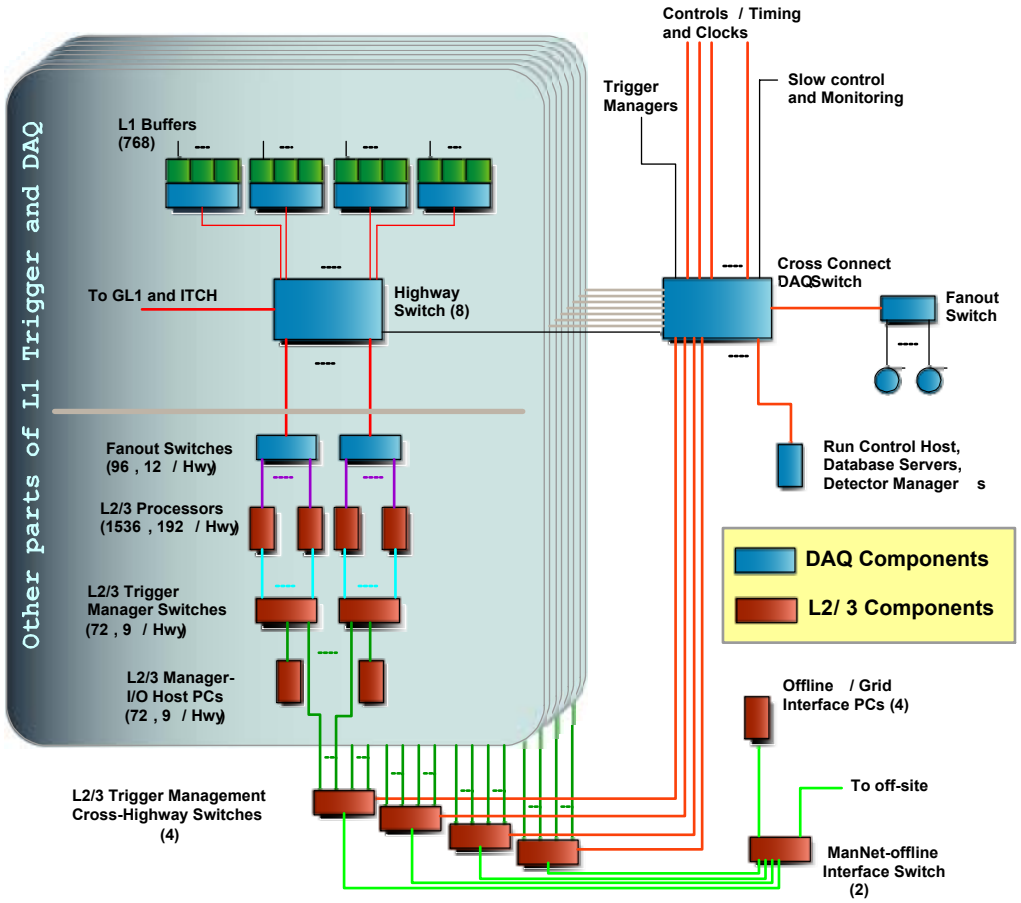
BTeV L2/3 Trigger Overview

- L2 and L3 triggers are implemented on the same PC Farm
- L2 Uses tracking information - detached vertices
- L3 does full reconstruction and writes DSTs - similar to traditional offline
- No data transfer between L2 and L3. **All crossing data sent in response to L2 request.**
- L3 processing occurs after L2 accept on same PC.

Trigger Level	Trigger Parameters			
	Input Rate	Output Rate	Reduction Factor	Processing Time
Level 2 refined tracking, vertex cut	50 KHz 250KB/event 12.5 GB/s	5 KHz 250KB/event 1250 MB/s	10	5 ms
Level 3 uses full event data	5 KHz 250KB/event 1250 MB/s	2.5 KHz 80KB/s 200MB/s	2	134 ms

BTeV L2/3 Trigger Overview

Highway-Network View



L2/3 View of Baseline Design

- L2/3 Workers consist of 1536 "12 GHz" CPUs in 768 dual-CPU 1U rack-mount PCs
- L2/3 Trigger includes Trigger Manager-I/O Host PCs, for database caches, worker management, server, monitoring, and event pool cache
- Contains separate Management network for fault tolerance and concurrent offline processing capability

BTeV L2/3 HW Costing Procedure Overview

- L2/3 Major Hardware

Description	Quantity
Farm Worker PCs	1536 CPUs
Manager-I/O Host PCs	54 PCs

- L2/3 Farm Workers are a major part of the cost
  - **L2 code exists and meets time constraints. Still improving.**
  - **No full L3 code**
    - **However, "tracking core" takes only 1/3 of allowed budget. Beyond that, we are really making a head start on physics analysis and offline**
    - **large 50% contingency**
  - **Assume CPU speed doubling every 2.5 years**

BTeV L2/3 HW Capacity Estimate

- L2 Trigger is benchmarked:
  - **Requirements satisfied by already existing CPUs**
  - **CPU Usage: 60-70% (L2/3); 10% (DAQ); 10% (RTES and other monitoring)**

CPU	Time/event (ms) - 2 Int/crossing		Time/event (ms) - 6 Int/crossing	
	Min. Bias	bb-bar	Min. Bias	bb-bar
P4 Xeon 2.4 GHz	2.3		3.2	
AMD Athlon 1.2 GHz	3.0	3.0	4.3	4.5

Trigger Level	**Trigger Parameters**				
	Input Rate	**Output Rate**	**Reduction Factor**	**Processing Time**	**Min. # CPUs (% Usage)**
Level 2	50 KHz 250 KB/event 12.5 GB/s	5 KHz 250 KB/event 1.25 GB/s	10	5 ms	250 (100%) 416 (60%)
Level 3	5 KHz 250 KB/event 1.25 GB/s	2.5 KHz 80 KB/event 200 MB/s	2	134 ms	672 (100%) 1120 (60%)

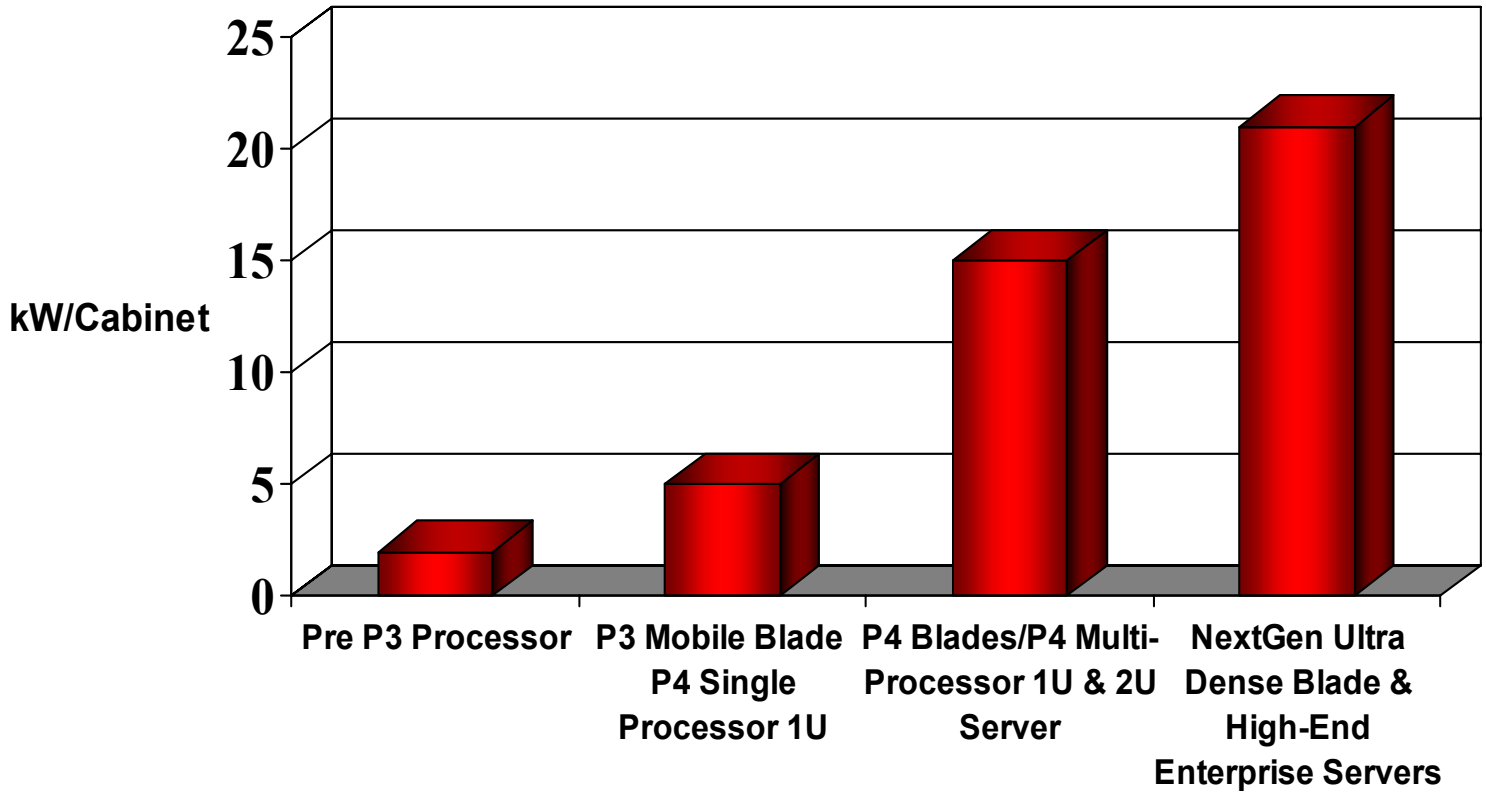
Other Trigger/ DAQ Issues

- There is substantial disk buffering capability in this system. There is of order 300 Tbytes of local disk and a "backing" store of about 0.5- 1.0 Pbyte. Given the high efficiency of the main trigger for B physics, we can have several "opportunities" :
  - We can buffer events at the beginning of stores if we get behind and then perform the L3 trigger when the luminosity declines
  - We can use idle cycles to do other chores like reanalysis if we have idle resources towards the end of stores, during low luminosity stores, or during off-time.

The full architecture of Level 2/3 actually makes it a very flexible computing resource.

Electrical Load Growth Projection

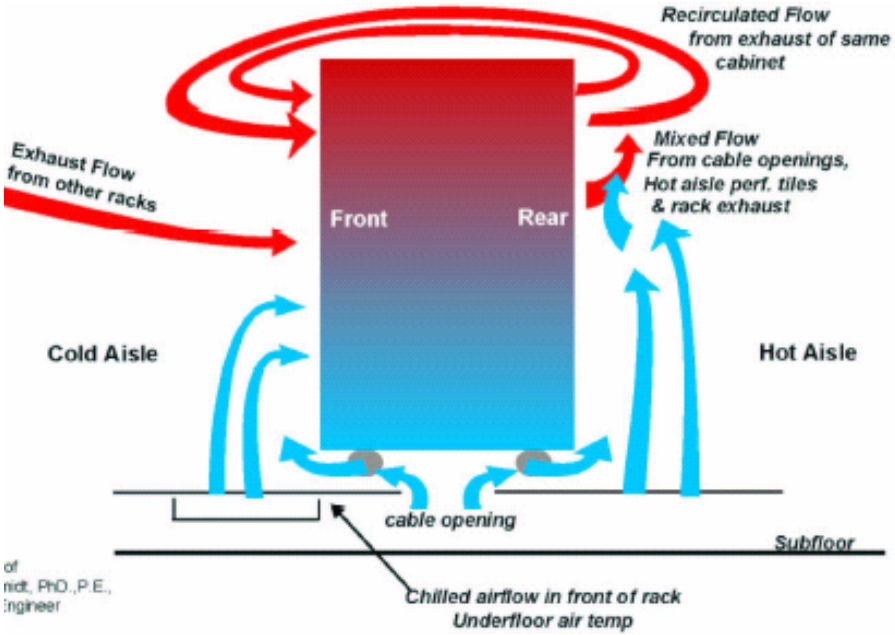
Cabinet Load Growth by Processor Type/Density in fully "optimized" 42U Enclosures



From presentation By ECHOBAY

Impact of High Density Processing on Data Centers

- Efficiency and reliability of traditional Data Center declines and "De-population" of room occurs when input power exceeds values >**3.5kW** per enclosure
- Data Centers with raised floor (Plenum) cooling systems cannot support input power values of >**7kW** per enclosure without overheating problems
- New generation processors in high density configurations can require as much as **25kW** input power per enclosure

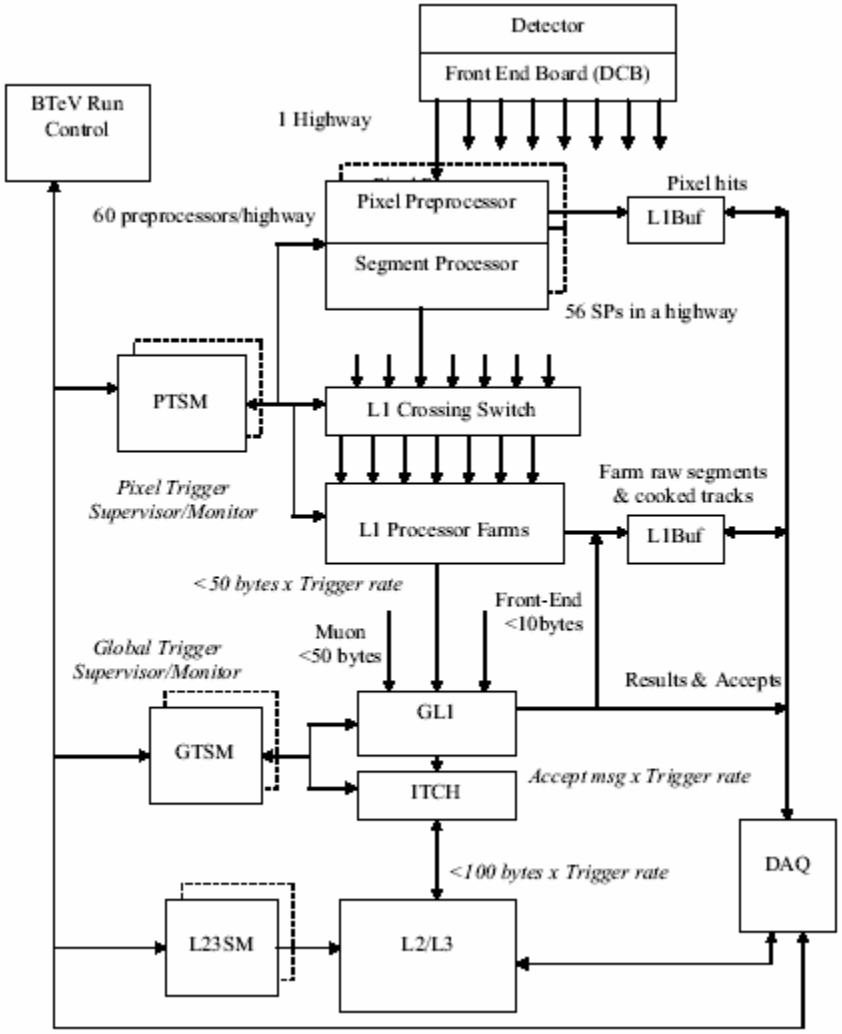


We are paying a lot of attention to this, having been warned by FNAL Computing Division staff .

We have over 600KW concentrated in a relatively small area

From presentation By ECHOBAY

Supervisory, Monitoring, Fault Detection \Functions



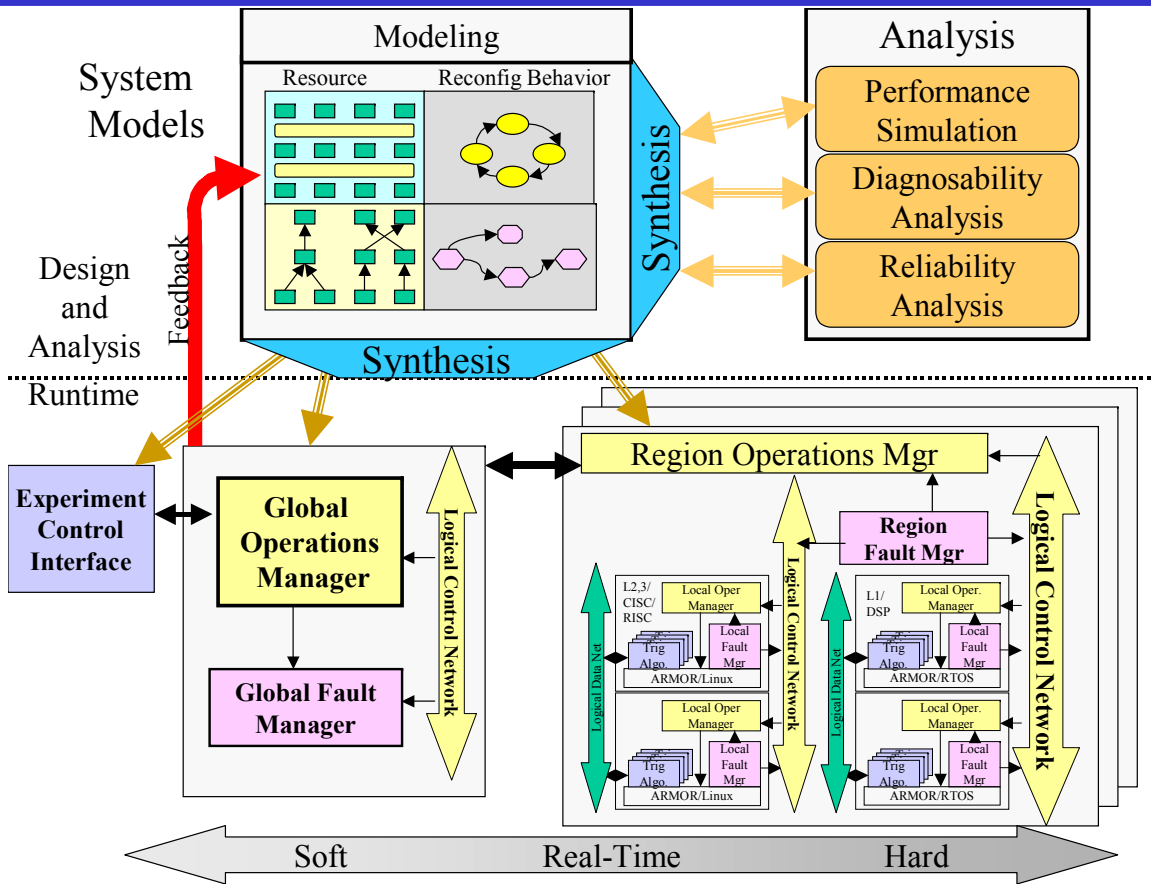
Monitoring: We have supervisor nodes for each part of the Level 1 Trigger – Vertex, Muon, and Global L1. We have supervisor nodes for the DAQ, a complete slow Control system, and many distributed fault monitoring nodes. At least 10% of the system resources are dedicated to fault monitoring, fault recovery and mitigation.

Main Requirements

- The systems must be **dynamically reconfigurable**, to allow a maximum amount of performance to be delivered from the available, and potentially changing resources.
- The systems must also be **highly available**, since the environments produce the data streams continuously over a long period of time.
- To achieve the **high availability**, the systems must be
  - **fault tolerant**,
  - **self-aware**, and
  - **fault adaptive**.
- Faults must be corrected in the shortest possible time, and corrected **semi-autonomously** (i.e. with as little human intervention as possible). Hence **distributed** and **hierarchical monitoring** and **control** are vital.
- The system must have a excellent life-cycle Maintainability and evolvability to deal with new trigger algorithms, new hardware and new versions of the Operating System
- **We may want to use the reconfigurability to arrange to have the system do other tasks, such as offline analysis, when it had excess resources, such as at the end of stores when luminosity is low.**

The proposed solution is a distributed, hierarchical Fault management system.

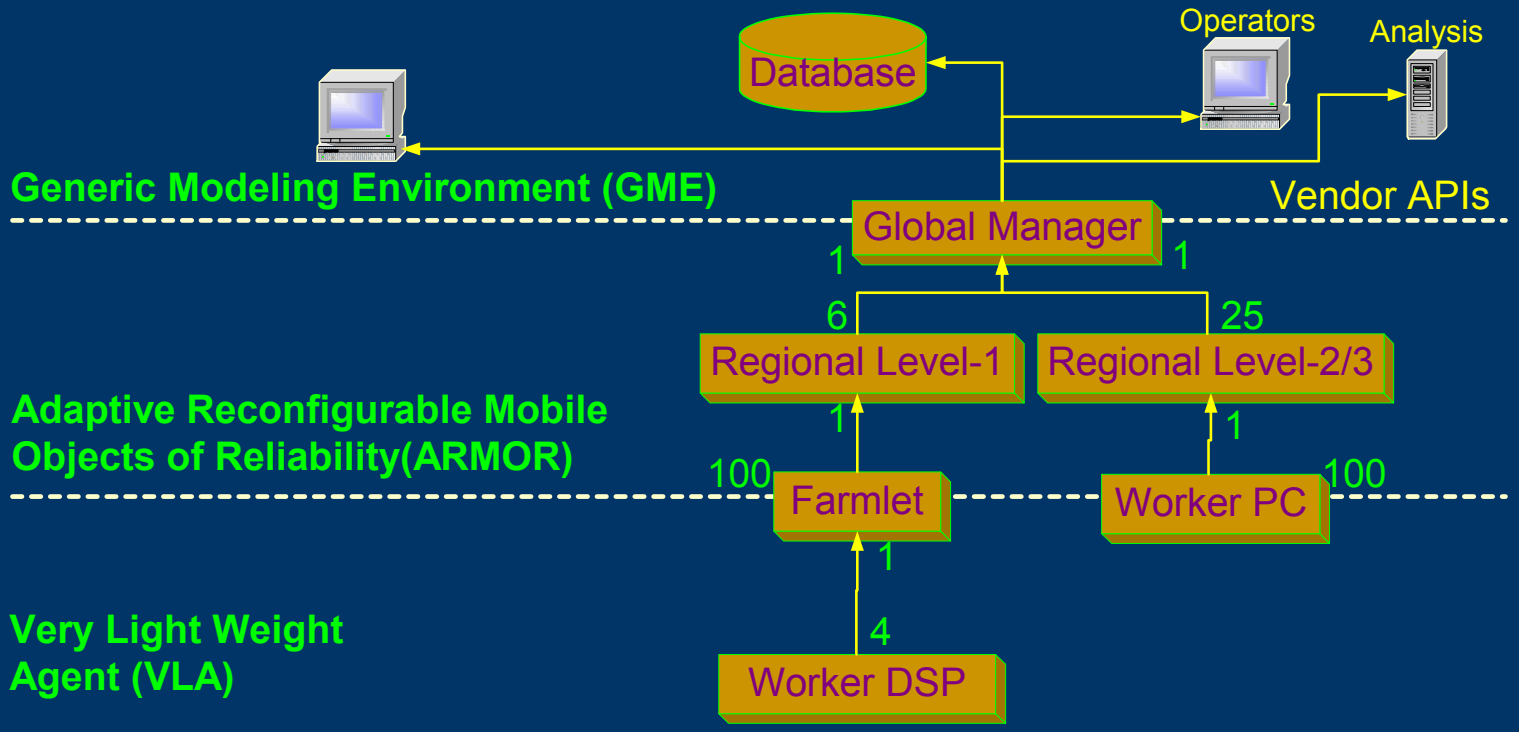
The Proposed Solution



Bi-Level System Design and Run Time Framework – System Models define behavior, function, performance, and fault interactions.. Analysis tools evaluate predicted performance to guide designers prior to implementation. Synthesis tools generate system configurations directly from the models. A fault-detecting, failure mitigating runtime environment executes these configurations in real-time, high. Local, regional, and global aspects are indicated. On-Line cooperation between runtime and modeling/synthesis environment permit global system reconfiguration in extreme-failure conditions.

Real Time Embedded Systems (RTES)

- RTES: NSF ITR (Information Technology Research) funded project
- Collaboration of computer scientists, physicists & engineers from: Univ. of Illinois, Pittsburgh, Syracuse, Vanderbilt & Fermilab
- Working to address problem of reliability in large-scale clusters with real time constraints
- BTeV trigger provides concrete problem for RTES on which to conduct their research and apply their solutions

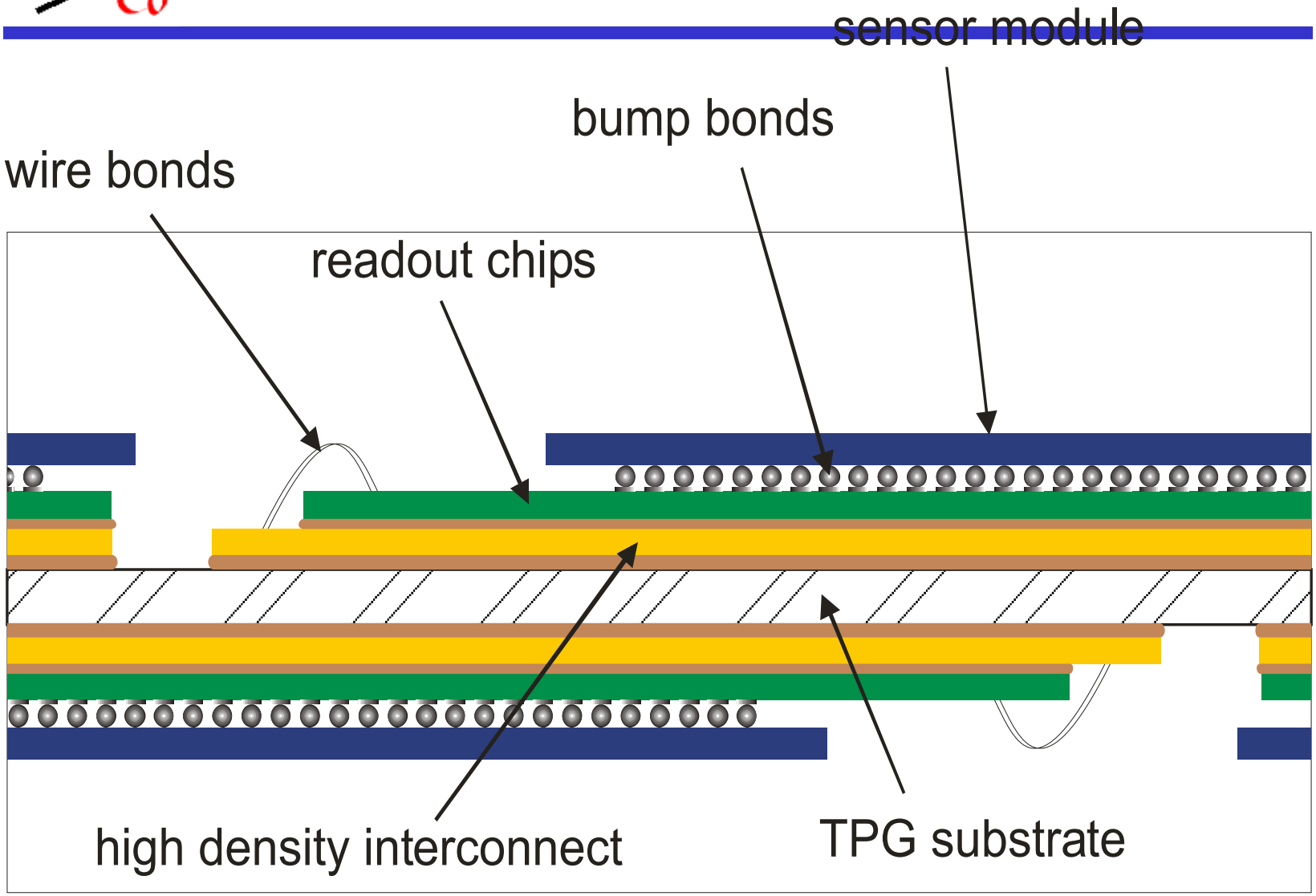


Conclusion

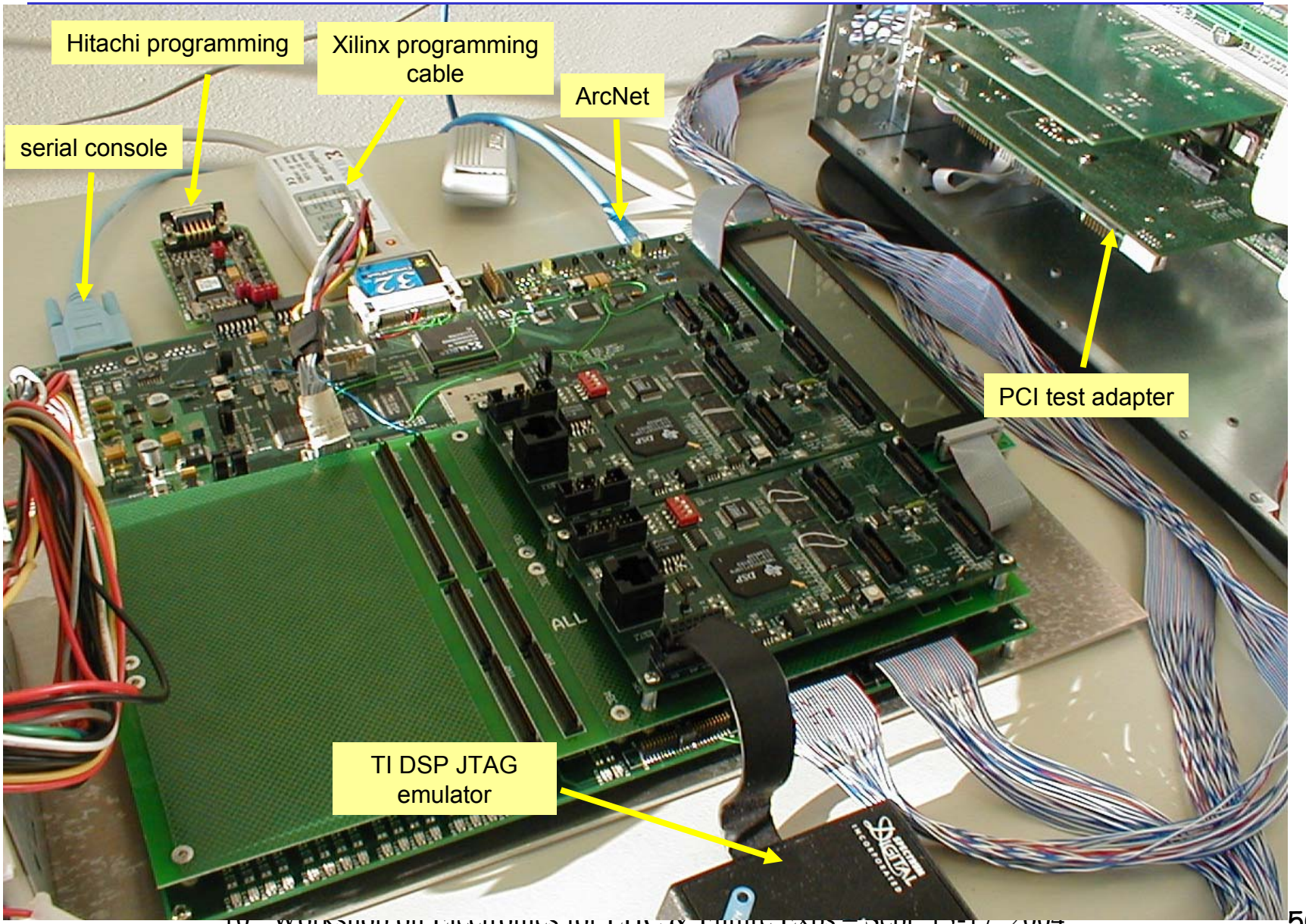
- **Triggers look more like normal "computer farms" and can be based on COTS components. The rapid decrease in the cost of CPUs, memory, network equipment and disk all supports this trend. The "highway" approach helps make this possible.**
- **Custom chips are still required for the front end. If on-the-fly sparsification can be implemented there is no need for on-board (short) pipelines and short, low latency Level 1 triggers. These constrain the experiment in nasty ways.**
- **With the large number of components the system must be made robust, I.e. fault tolerant. Significant resources must be provided to monitor the health of the system, record status, and provide fault mitigation and remediation. In BTeV, we expect to commit 10-15% of our hardware and maybe more of our effort on this.**
- **Infrastructure, especially power and cooling, is a big challenge**
- **Because all systems have buffers that can receive fake data, these systems can be completely debugged before beam.**

Constructing this system and making it work is a big challenge, but also an exciting one. The reward will be excellent science!

End


sensor module
bump bonds
wire bonds
readout chips
high density interconnect
TPG substrate

L1 trigger pre-prototype test stand



L1 pre-prototype with DSP mezzanine cards