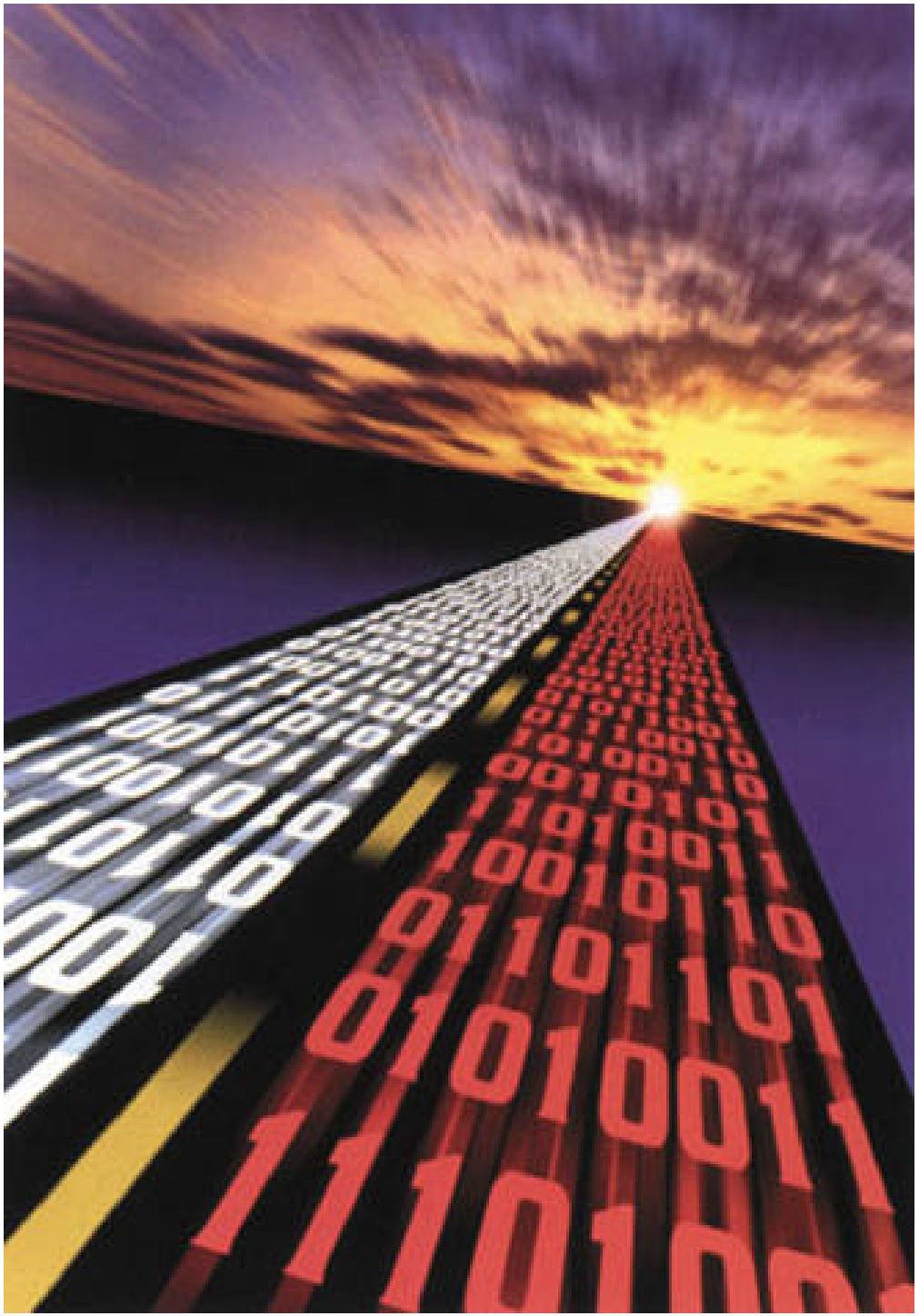
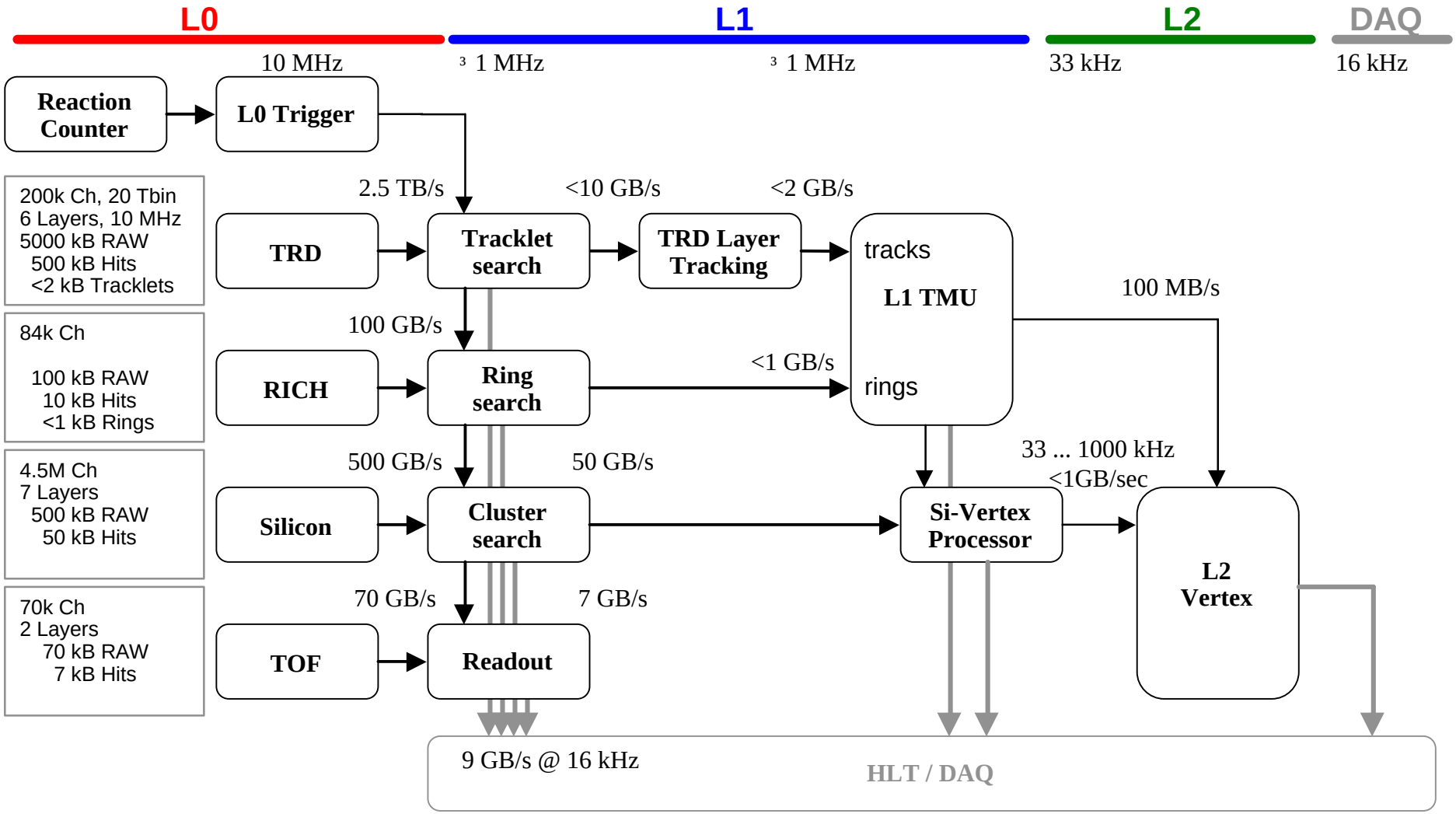
GSI
CBM

Volker Lindenstruth
Kirchhoff Institute for Physics
Chair of Computer Science
University Heidelberg, Germany
Phone: +49 6221 54 4303
Fax: +49 6221 54 4345
Email: ti@kip.uni-heidelberg.de
WWW: www.ti.uni-hd.de



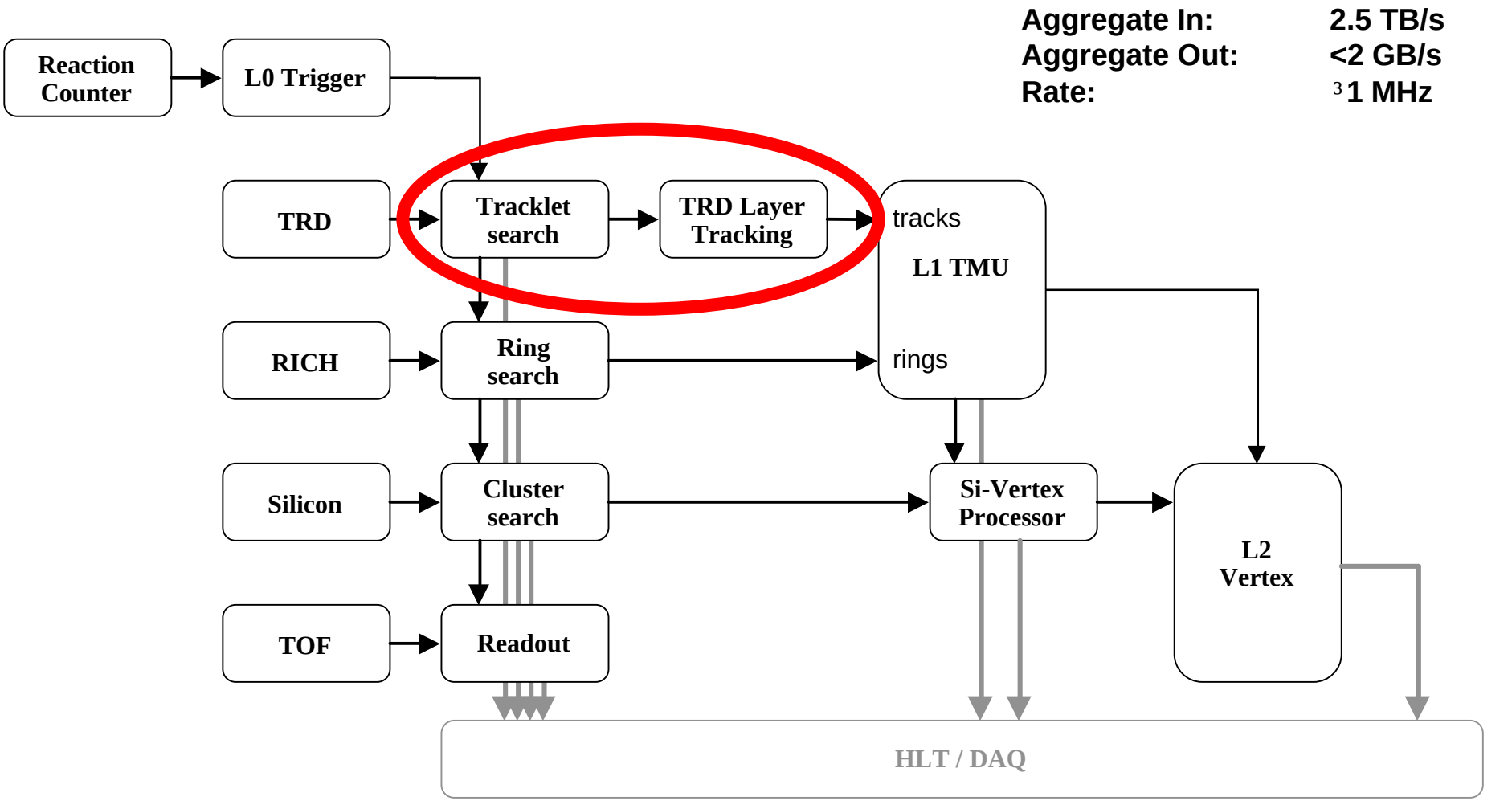
Overview


L0
L1
L2
DAQ
10 MHz
3 1 MHz
3 1 MHz
33 kHz
16 kHz
Reaction Counter
L0 Trigger
200k Ch, 20 Tbin
6 Layers, 10 MHz
5000 kB RAW
500 kB Hits
<2 kB Tracklets
2.5 TB/s
<10 GB/s
<2 GB/s
TRD
Tracklet search
TRD Layer Tracking
tracks
L1 TMU
rings
100 MB/s
84k Ch
100 kB RAW
10 kB Hits
<1 kB Rings
100 GB/s
RICH
Ring search
<1 GB/s
4.5M Ch
7 Layers
500 kB RAW
50 kB Hits
500 GB/s
50 GB/s
33 ... 1000 kHz
<1GB/sec
Silicon
Cluster search
Si-Vertex Processor
L2 Vertex
70k Ch
2 Layers
70 kB RAW
7 kB Hits
70 GB/s
7 GB/s
TOF
Readout
9 GB/s @ 16 kHz
HLT / DAQ

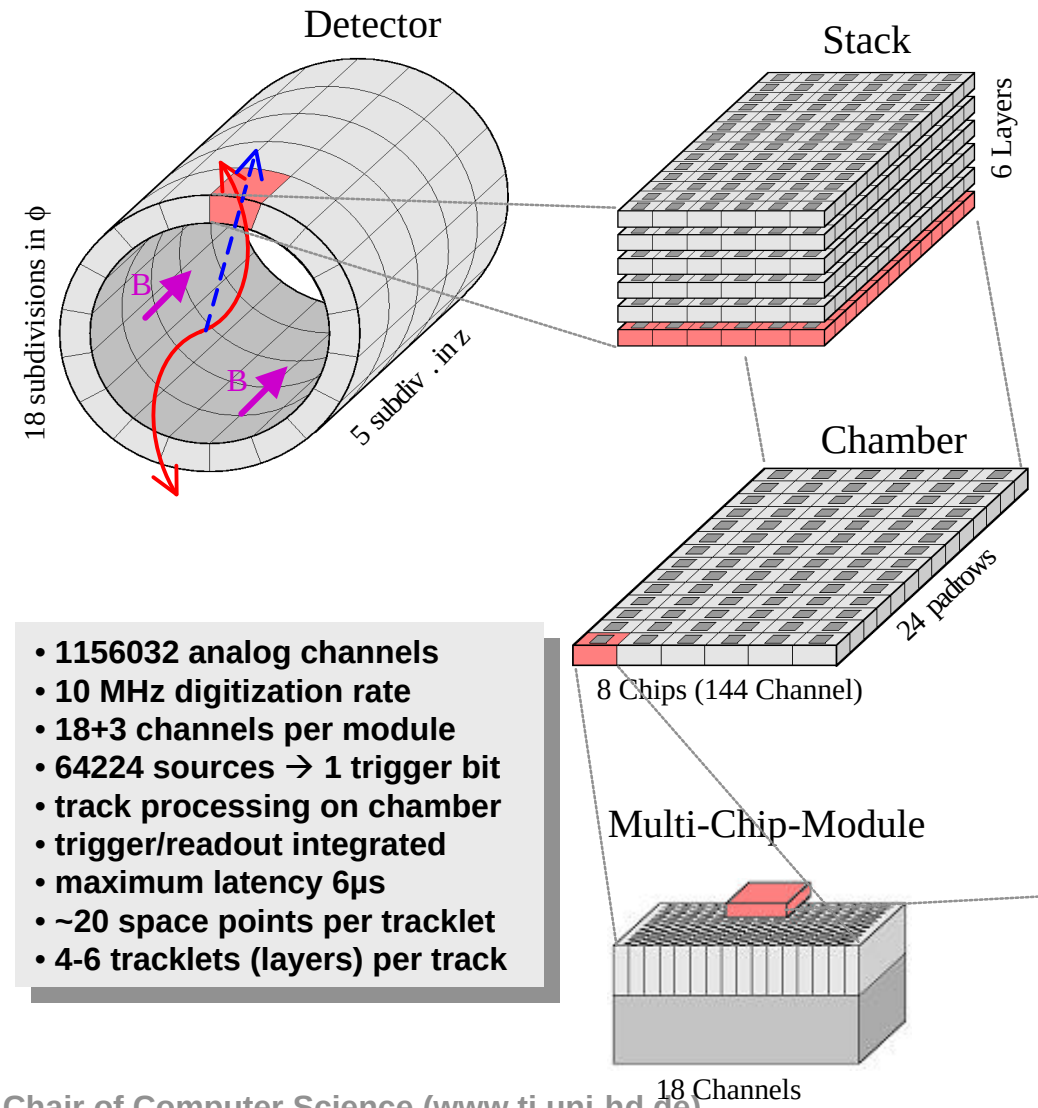
Overall Observations and Axioms

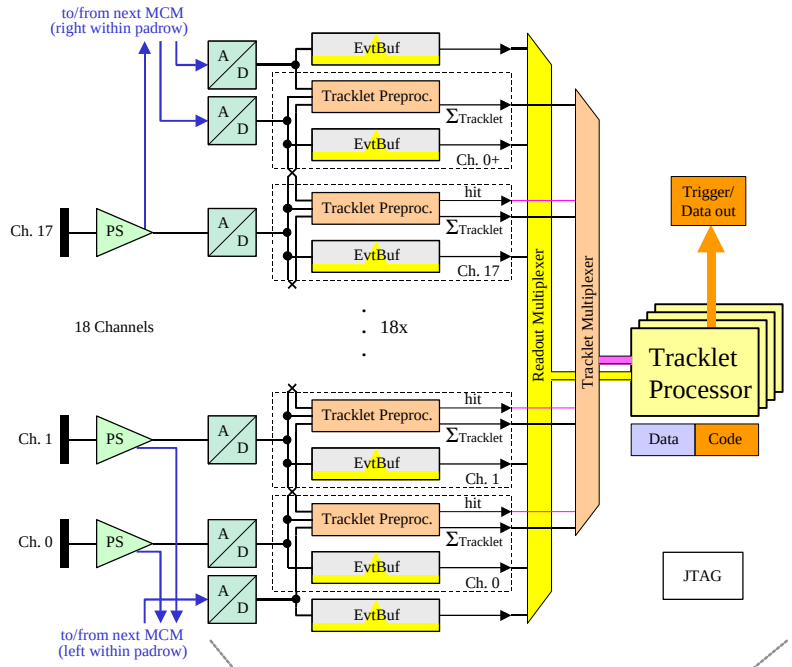
- **No physics or detector requirement for maximum trigger latency**
- **Can afford to implmement large derandomizing Buffers at L0**
- **Move Data only if necessary**
  - **(keep RAW data in FE until L2 accept)**
- **Process local data locally**
  - **ship only results (tracklets, rings, (displaced) vertices)**

TRD Trigger Architecture

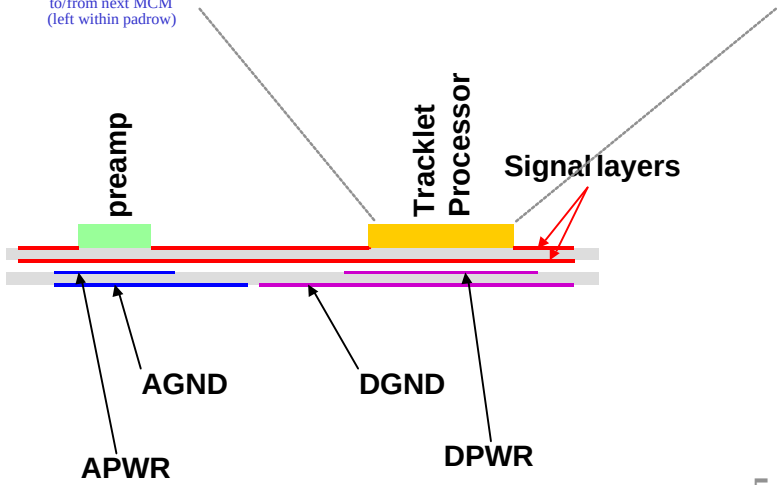


ALICE TRD Overview

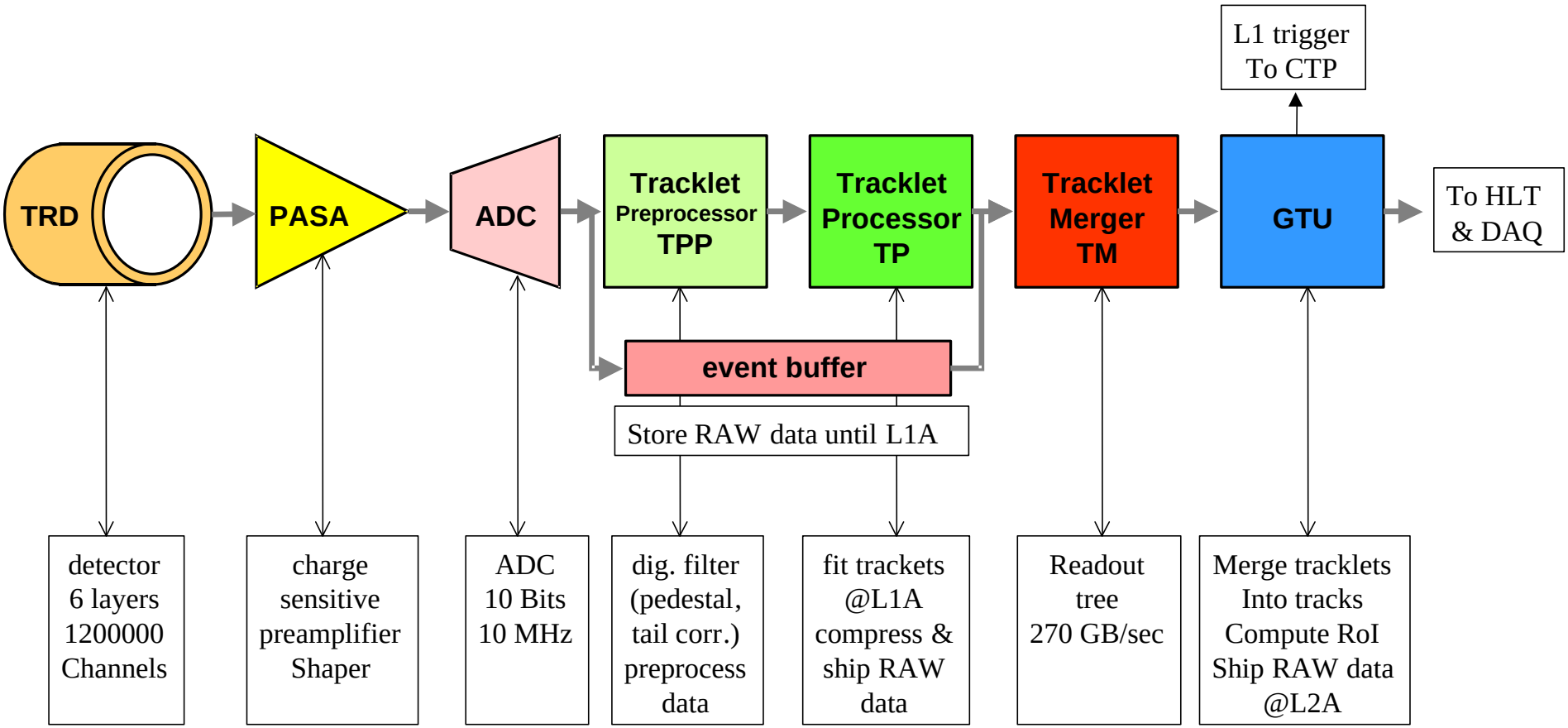




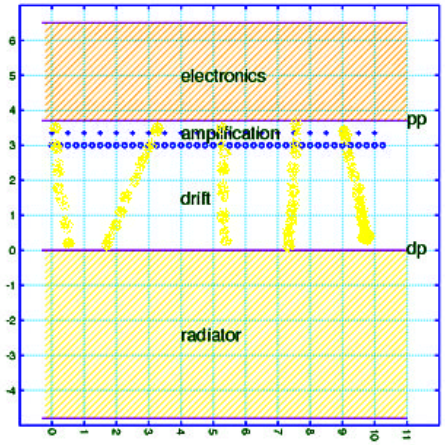
- **1156032 analog channels**
- **10 MHz digitization rate**
- **18+3 channels per module**
- **64224 sources → 1 trigger bit**
- **track processing on chamber**
- **trigger/readout integrated**
- **maximum latency 6µs**
- **~20 space points per tracklet**
- **4-6 tracklets (layers) per track**

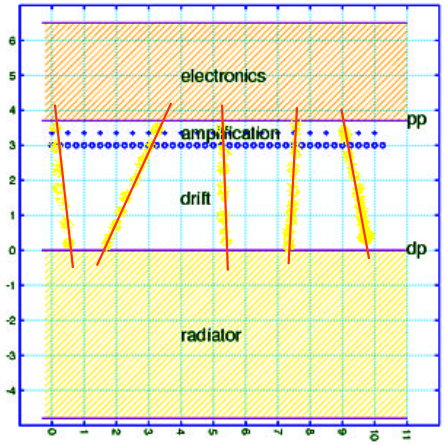


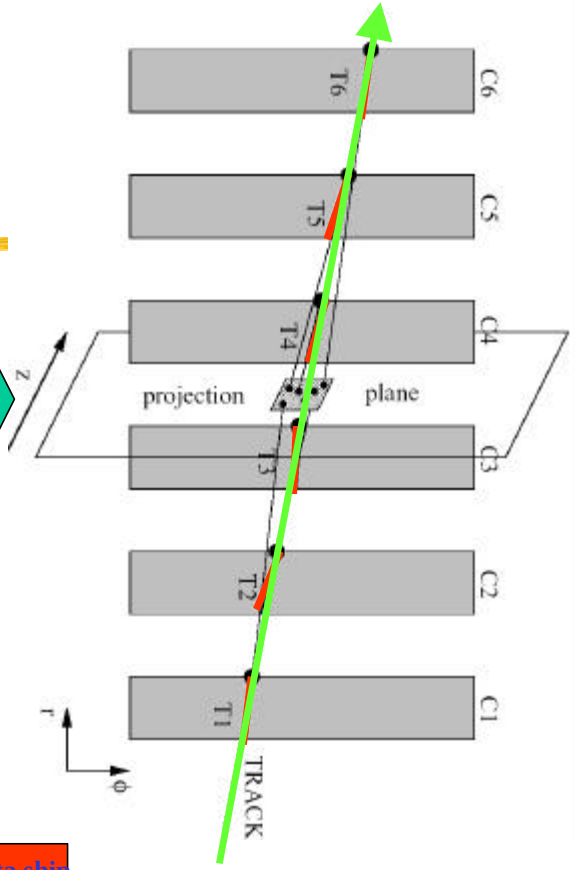
TRD Electronics Chain

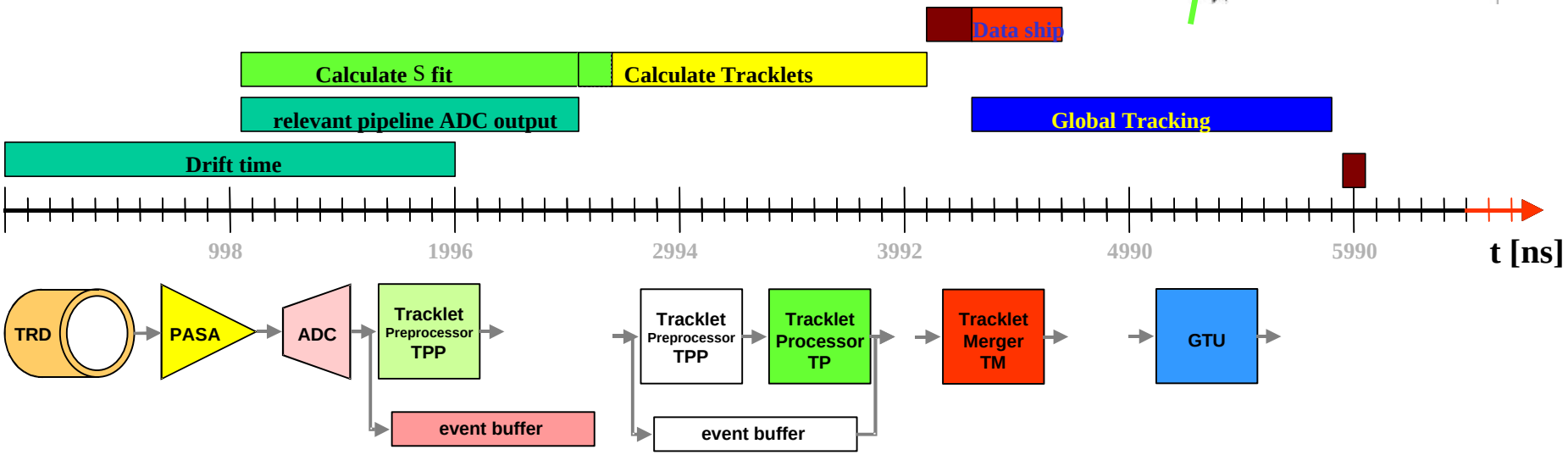


TRD Trigger Timing

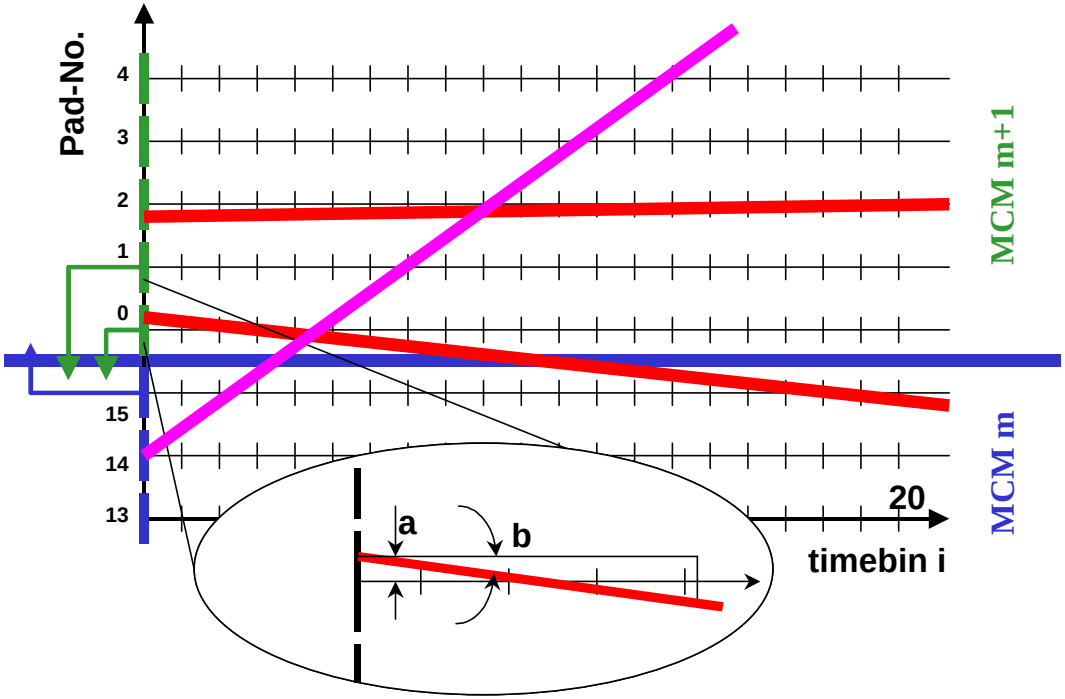








Tracklet Fit (Tracklet Preprocessor)



During Drift Time:

N = hit count $\quad y_i$ = position

$\sum x_i$ = time bin sum $\quad \sum y_i$ = position sum

$\sum x_i y_i$ = time bin*position sum

$\sum y_i^2$ = position² sum

After Drift Time:

a = intercept

$$a = \frac{\sum x_i^2 \sum y_i - \sum x_i \sum x_i y_i}{N \sum x_i^2 - \left(\sum x_i\right)^2}$$

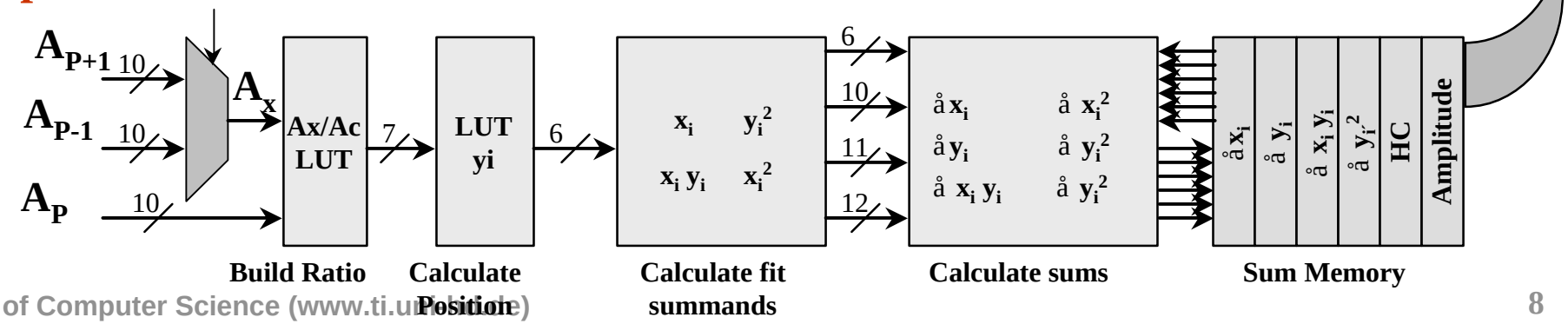
b = slope

$$b = \frac{N \sum x_i y_i - \sum x_i \sum y_i}{N \sum x_i^2 - \left(\sum x_i\right)^2}$$

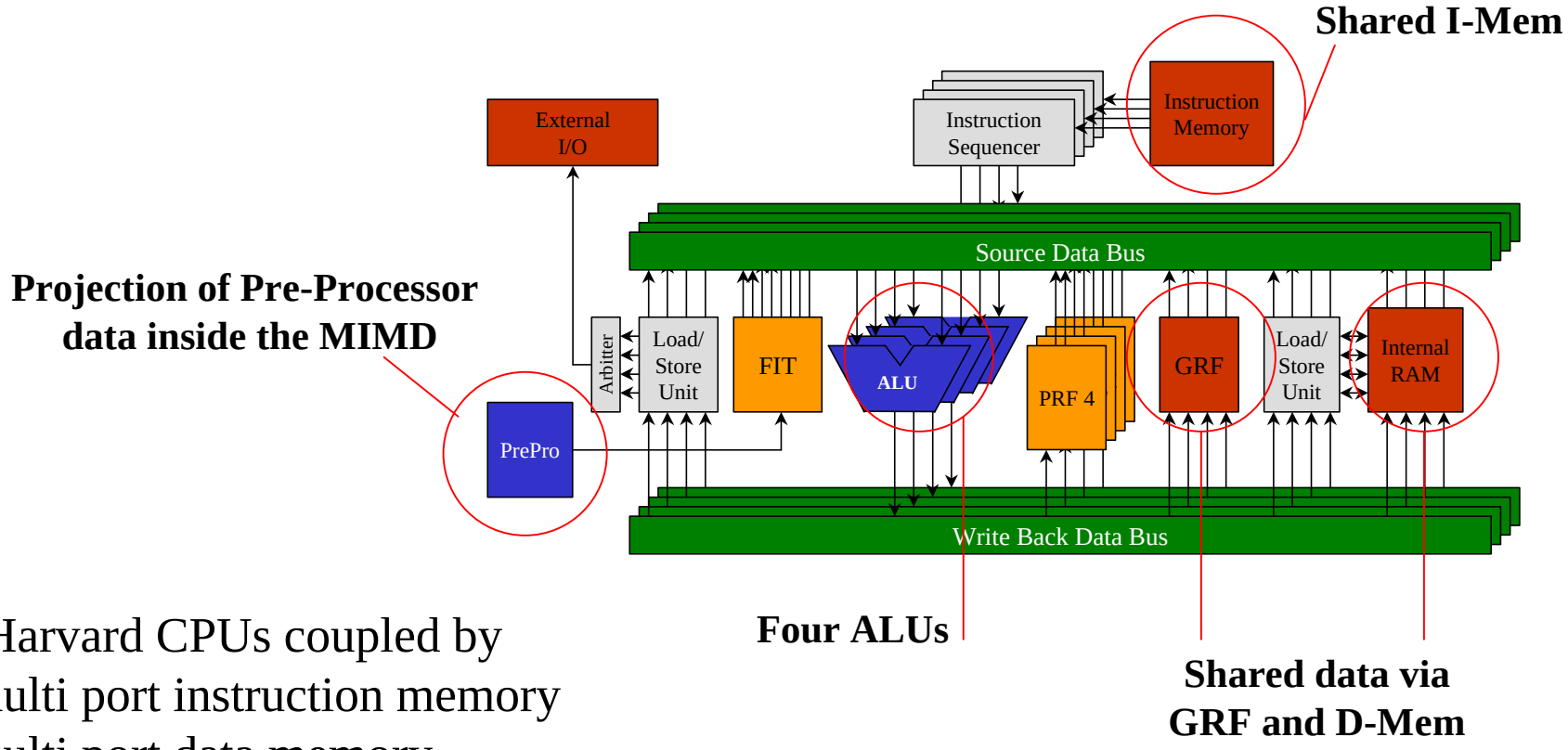
χ^2 = track quality

merge track segments in pad row

Pipelined Calculation :

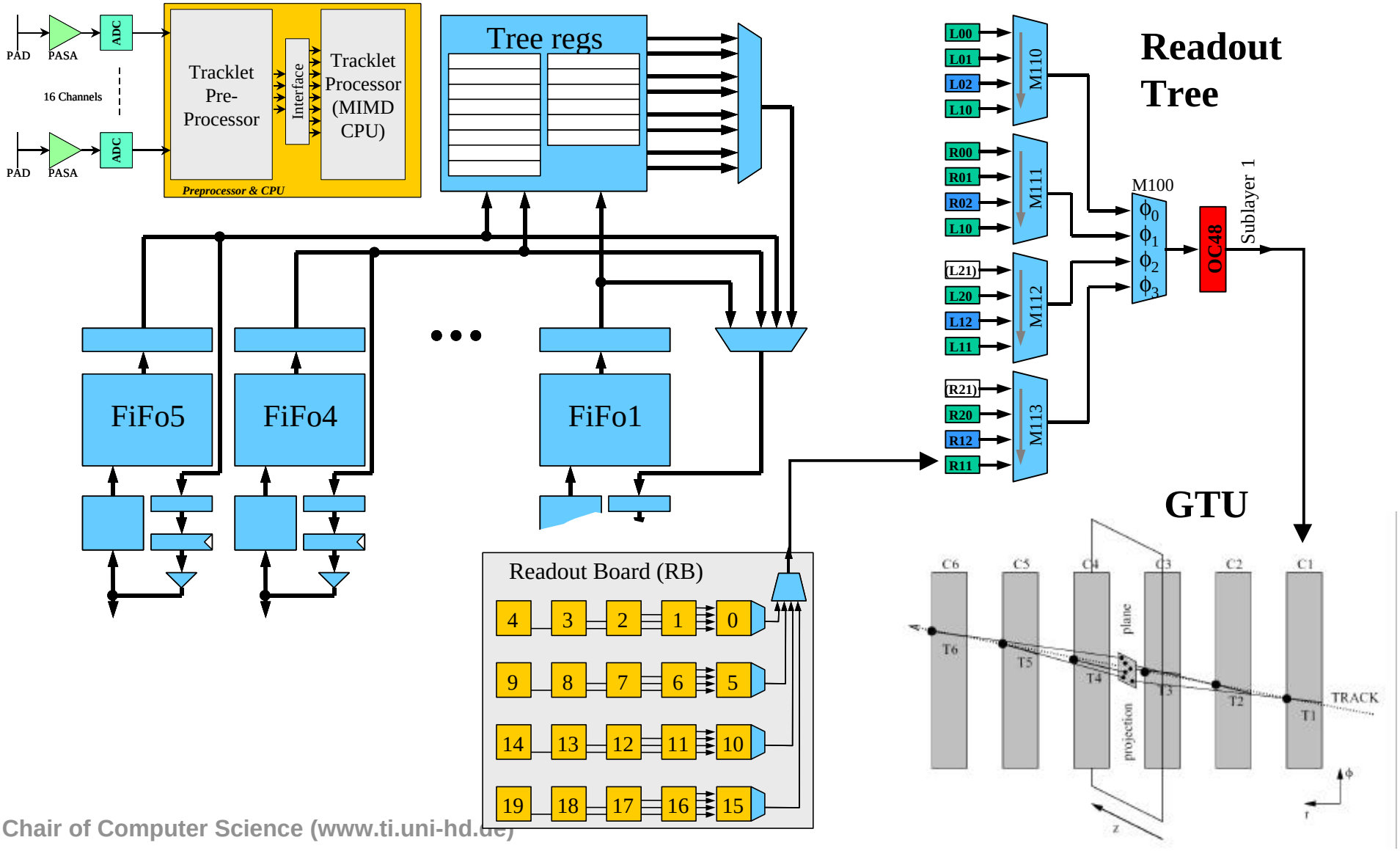


Tracklet (MIMD) Processor



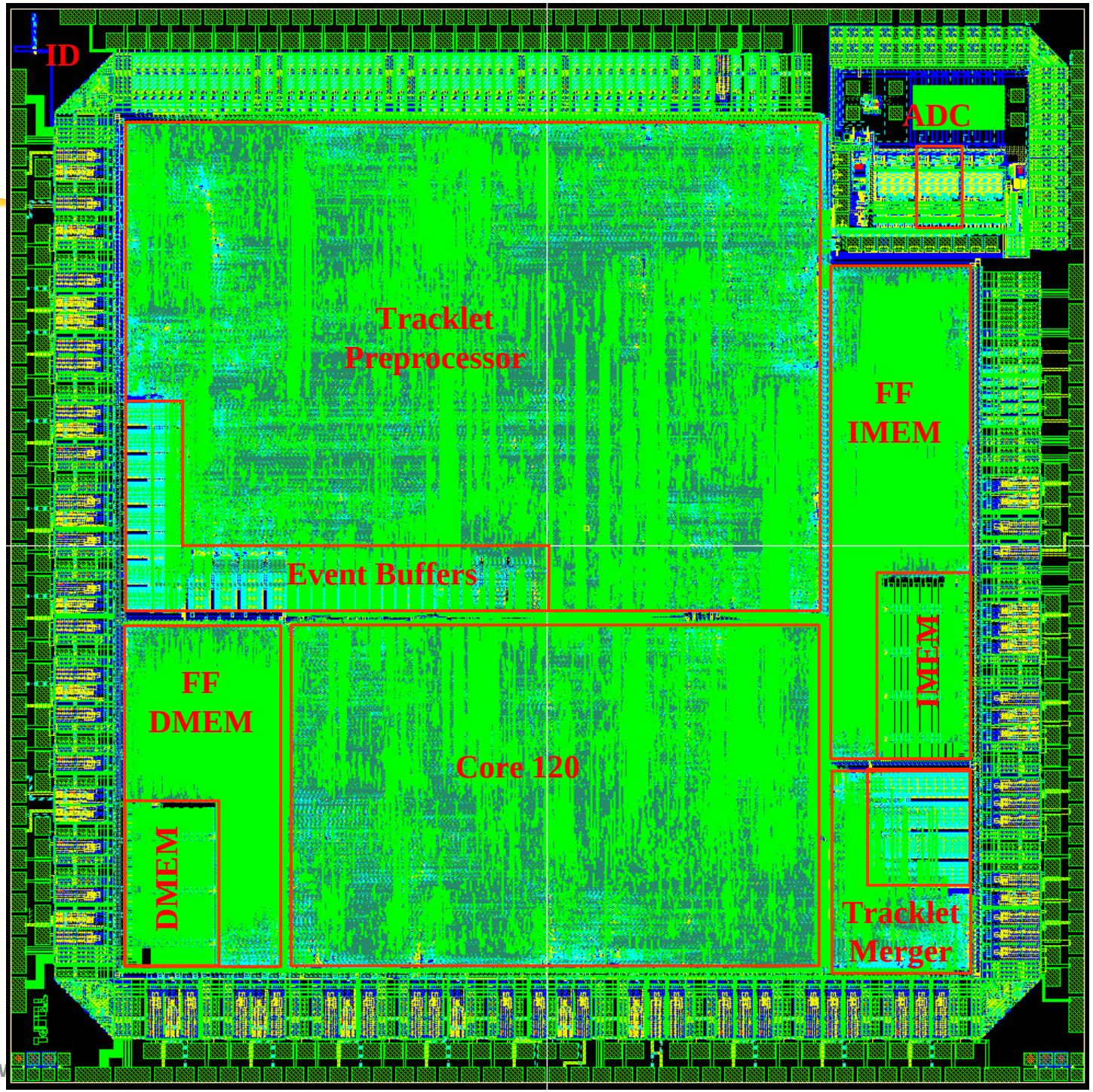
- Four Harvard CPUs coupled by
  - multi port instruction memory
  - multi port data memory
  - Global Register File (GRF)
- Register based interface to Pre-Processor
- Two stage pipeline (fetch/decode, execute/write back)
- **Architecture can be adopted to any general purpose processor**

Tracklet→Track→Trigger (Tracklet Merger)

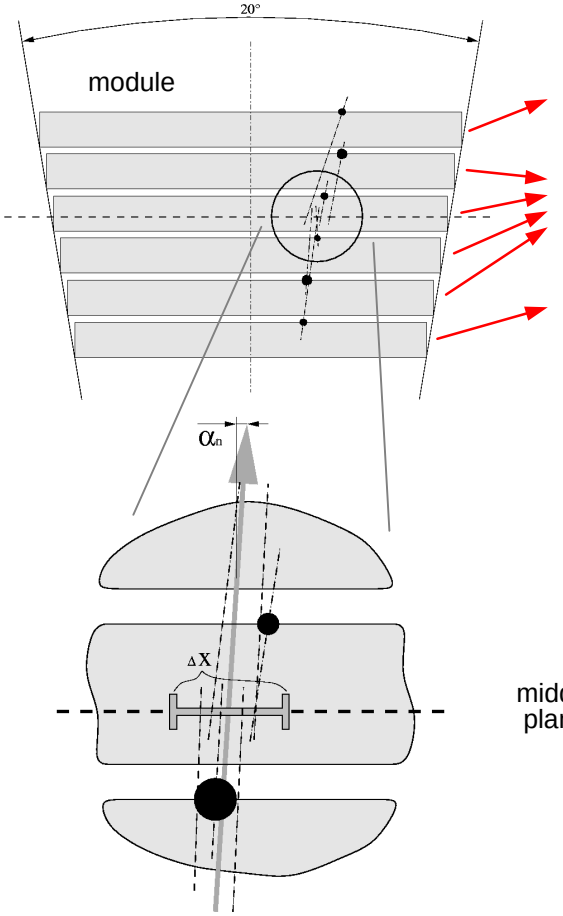


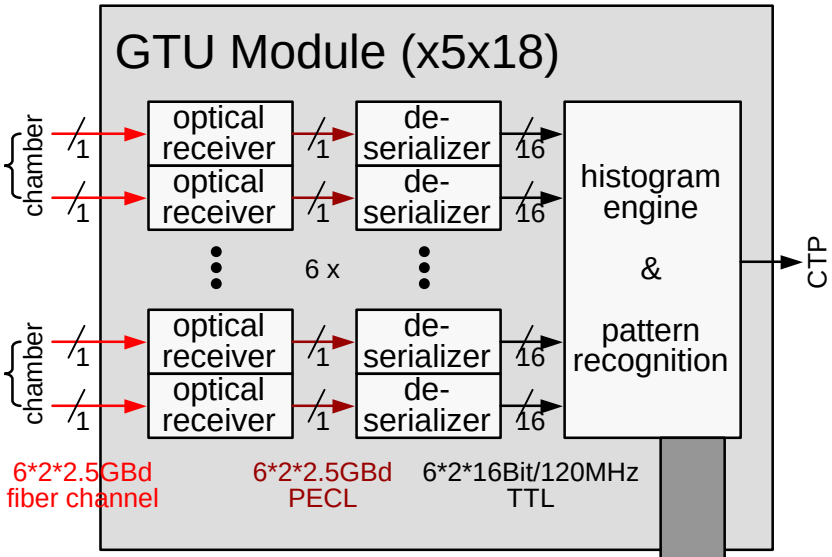
TRAP1 Die

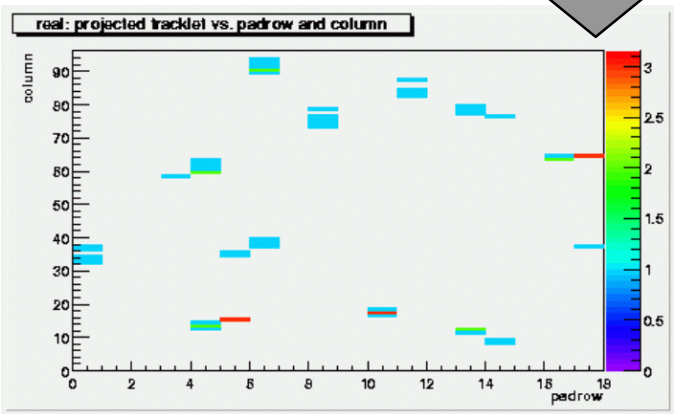
- 4.5 M transistors
- 180 nm DSM process
- all clocks gated
- 21 digital filters (Xtalk + 1/t)
- 18 full channels
- multi-event buffers
- 3x 10Bit, 10MHz ADC
- ADC 0.1 mm²
- PLL clock recovery
- 4+1 2.5 Gbit Links
- BIST, JTAG
- 4 RISC processors
- redundant, fault tolerant nonfig. network
- full custom QPM
- tape-out May 12
- 5000x5000 µm die
- 1.8 V Core
- TOX = 4nm



ALICE TRD Global Tracking Unit



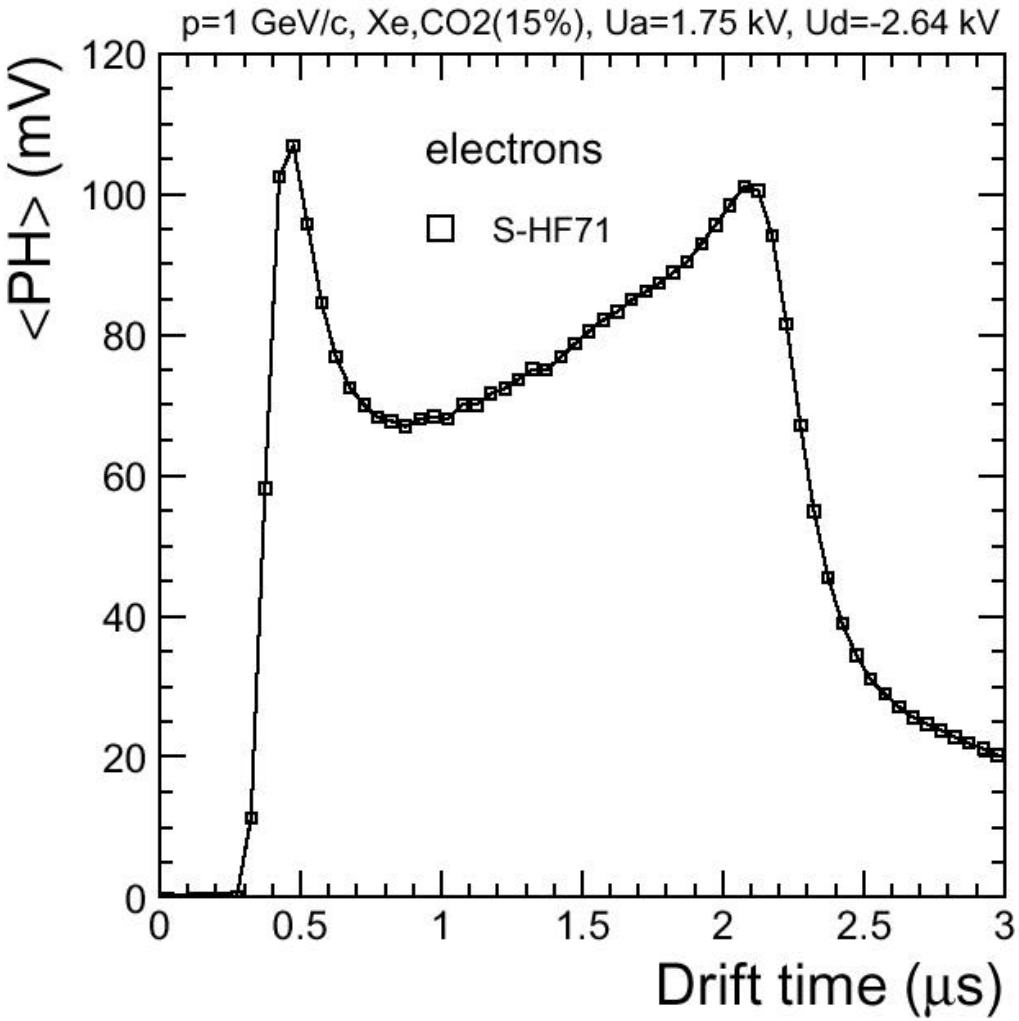




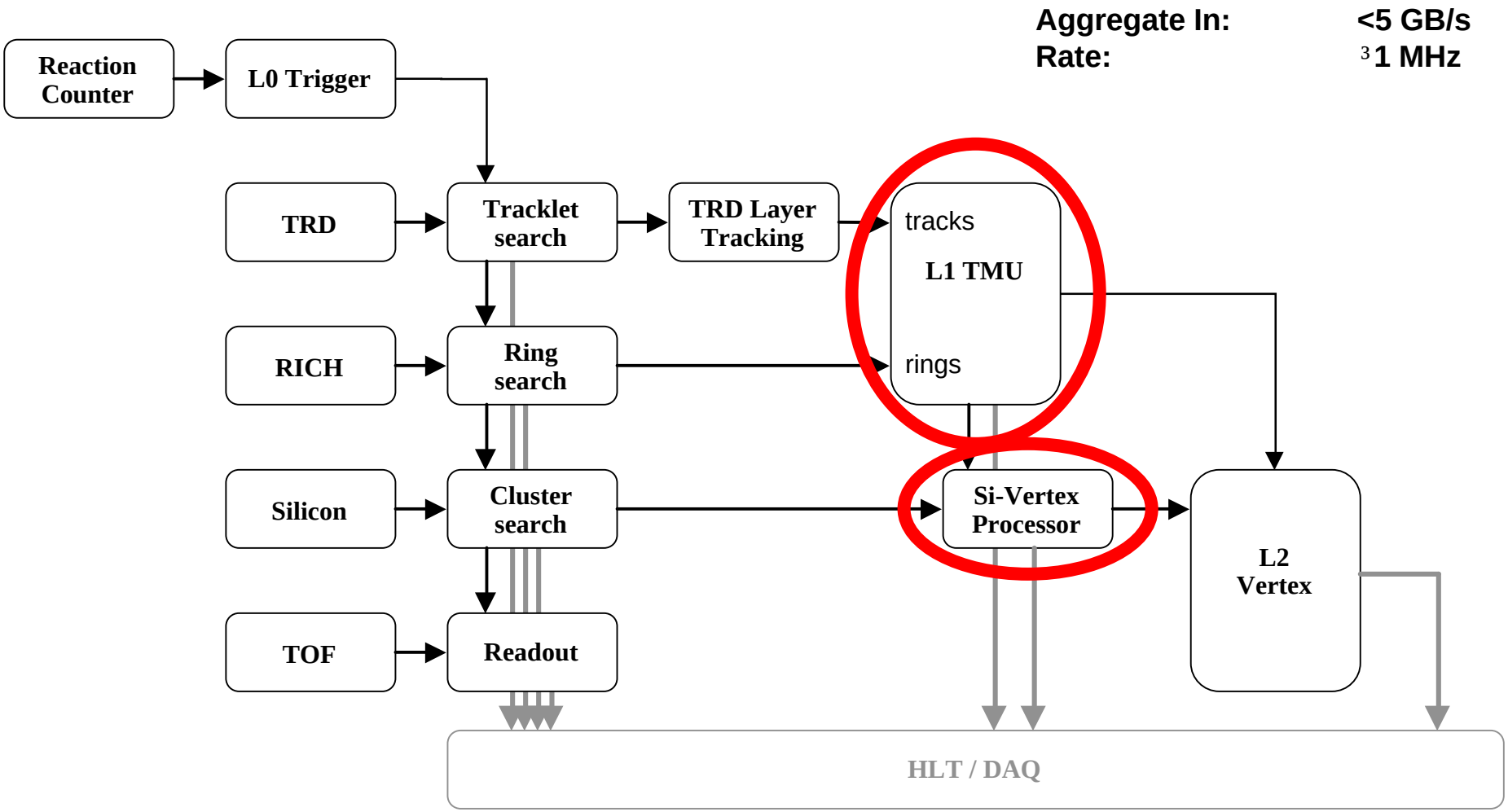
- global tracking is performed module wise (540 modules).
- input of one GTU are 6*2 2.5GBd optical fiber channel links which are converted into 6*2 16Bit/120Mhz data streams.
- readout order of individual chambers are optimized for histogramming and pattern recognition in parallel.
- histogramming engine fills a data field with tracklet candidates regarding their position.
- looking for patterns with
  - a cluster of at least 4 candidates
  - adequate similar angle
  - high momentum
  - adequate quality
- quality and position of found hit is sent to central trigger processor
- GTU operates also for raw data readout and compression

CBM TRD Outlook

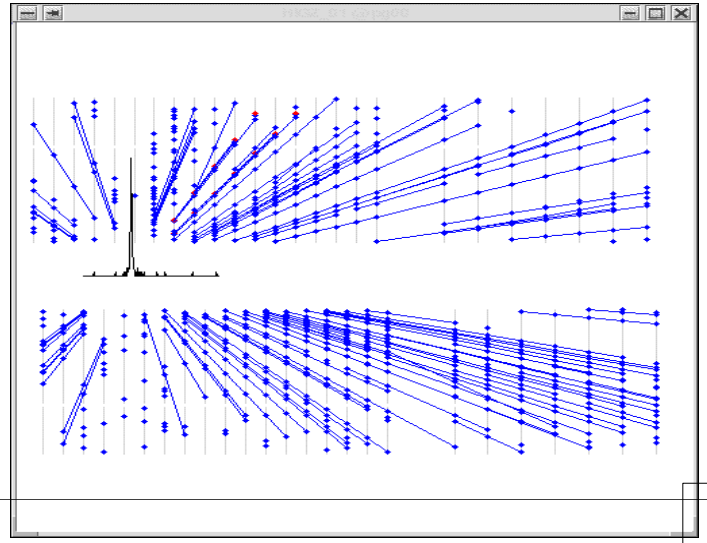
- Detector Occuppancy critical
- Interaction rate > drift time
- L0 provides more time-stamp kind of information
- Multi-Event pile-up has to be detected and sorted out electronically
- Large numbers of Channels and large event buffers can be well integrated
- Digitize as early as possible
- Use digital filters
- Interference of digital circuitry into analog front-end even more critical



L1/L2 Si-Vertex Trigger Processor



LHCb Vertex Position



2D tracking

Primary vertex

Secondary tracks

3D tracking

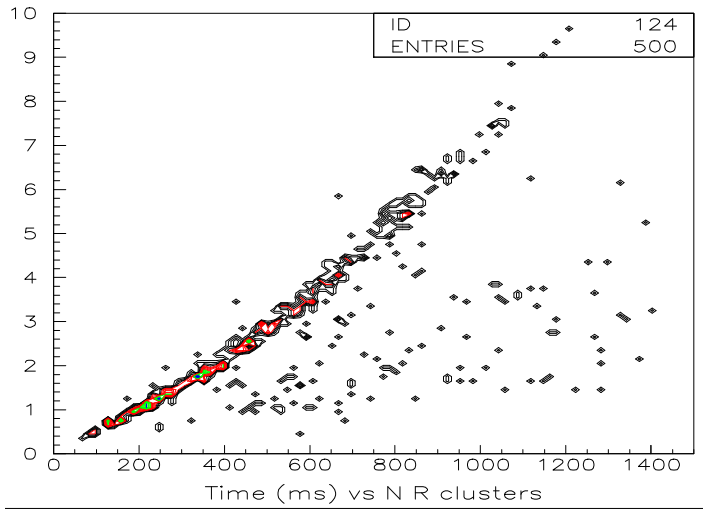
Secondary vertices

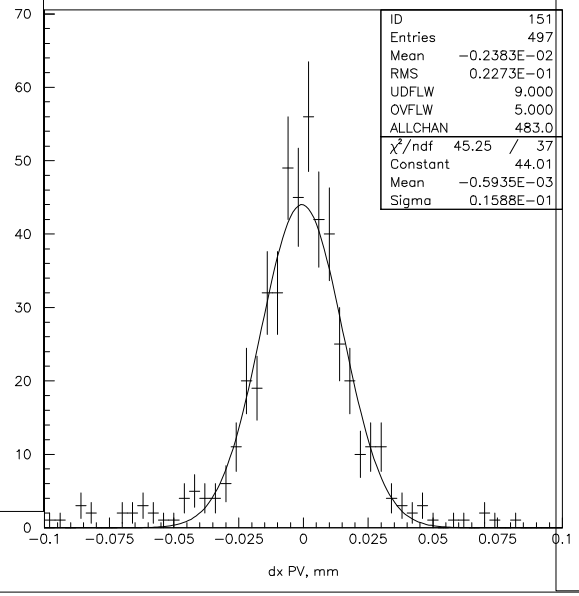
- **VELO STATISTICS (2D)** : 380 events

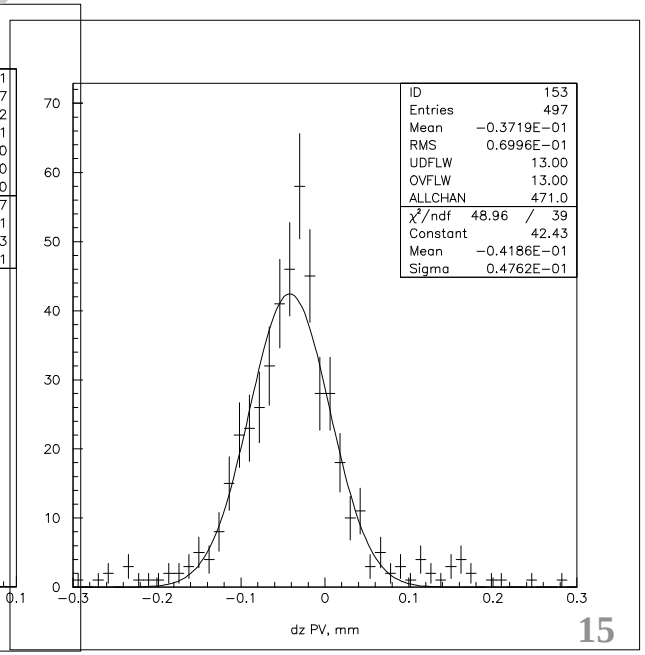
- RefB efficiency (%|Nmc) : 0.9801 | 853
- Refprim efficiency (%|Nmc) : 0.9741 | 5179
- Refset efficiency (%|Nmc) : 0.9598 | 20797
- Allset efficiency (%|Nmc) : 0.9161 | 26565
- Extra efficiency (%|Nmc) : 0.7581 | 5768
- Clone probability (%|Rec) : 0.0288 | 700
- Ghost probability (%|Rec) : 0.0205 | 498
- MC tracks/event found : 64

PRIMARY VERTEX RESOLUTION:

S (x,y) = 16 mm, S z = 47 mm

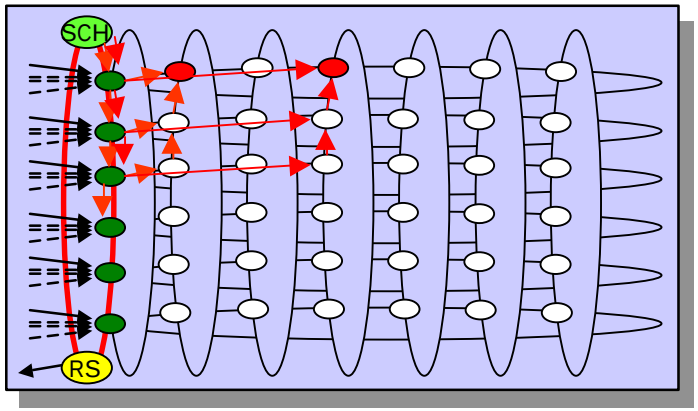


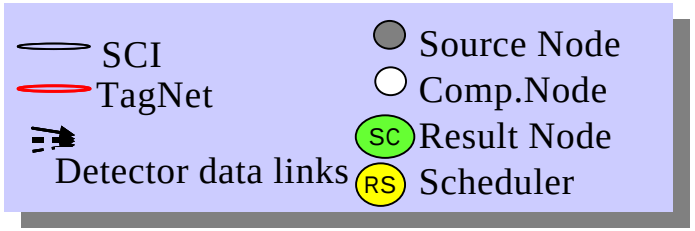




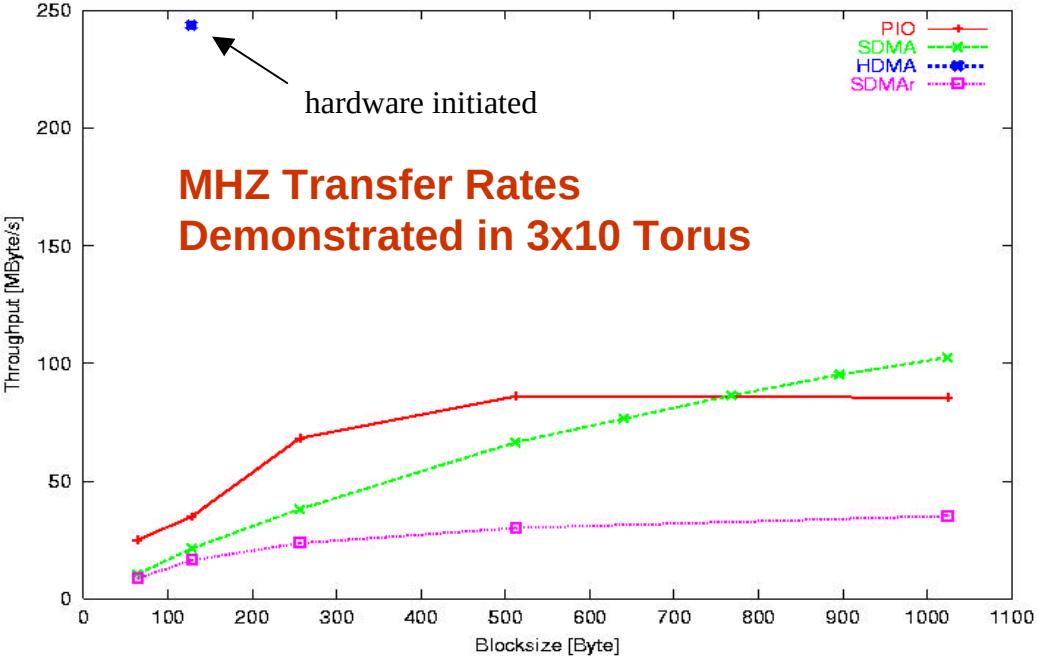
Scheduled Subevent Transfer @ LHCb

Packets of 128 Bytes are assembled in PCI Cards and distributed in PC cluster, implementing event building and load balanced data distribution at rates exceeding 1.9 MHz.
Hardware initiated DMA through Scalable Coherent Interface
Final System will implement about 300 PCs. Maximum Latency 2ms.



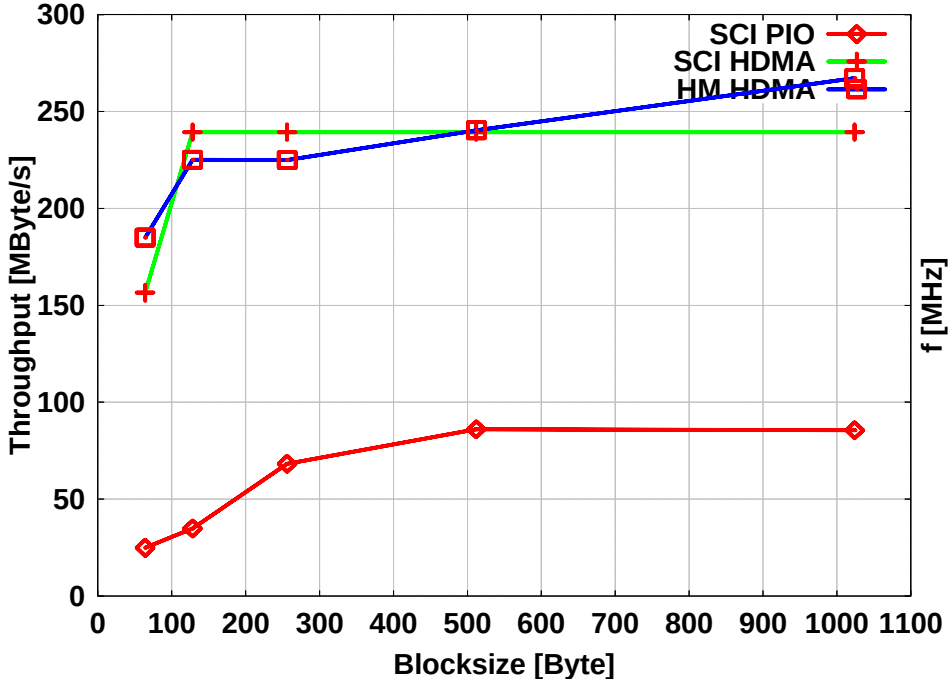


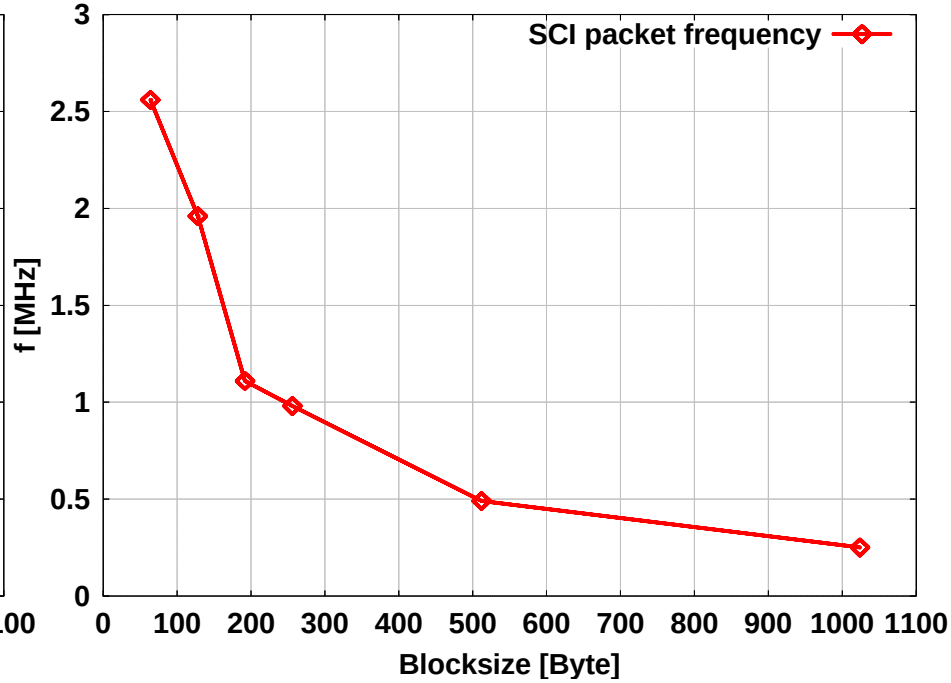
128 byte packets @ a rate of 1.96 MHz



http://www.ti.uni-hd.de/LHCb

MHz Event Rate in PC Clusters





64B @ PCI 32/33, 2xORCA interleaved

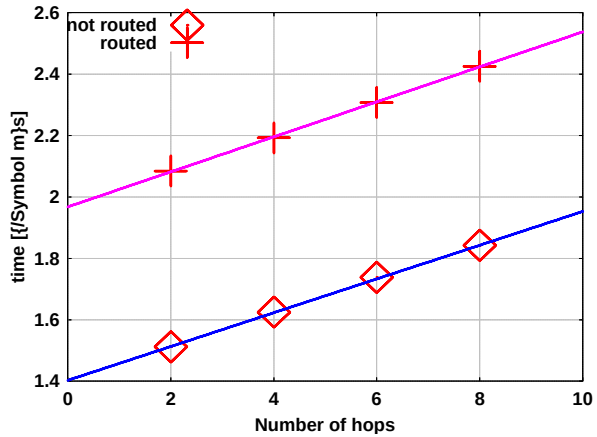
Address (1 cc)	Wait (**4 cc**)	Data (16 cc)	Idle (1 cc)

f_{max} = 1.50 MHz

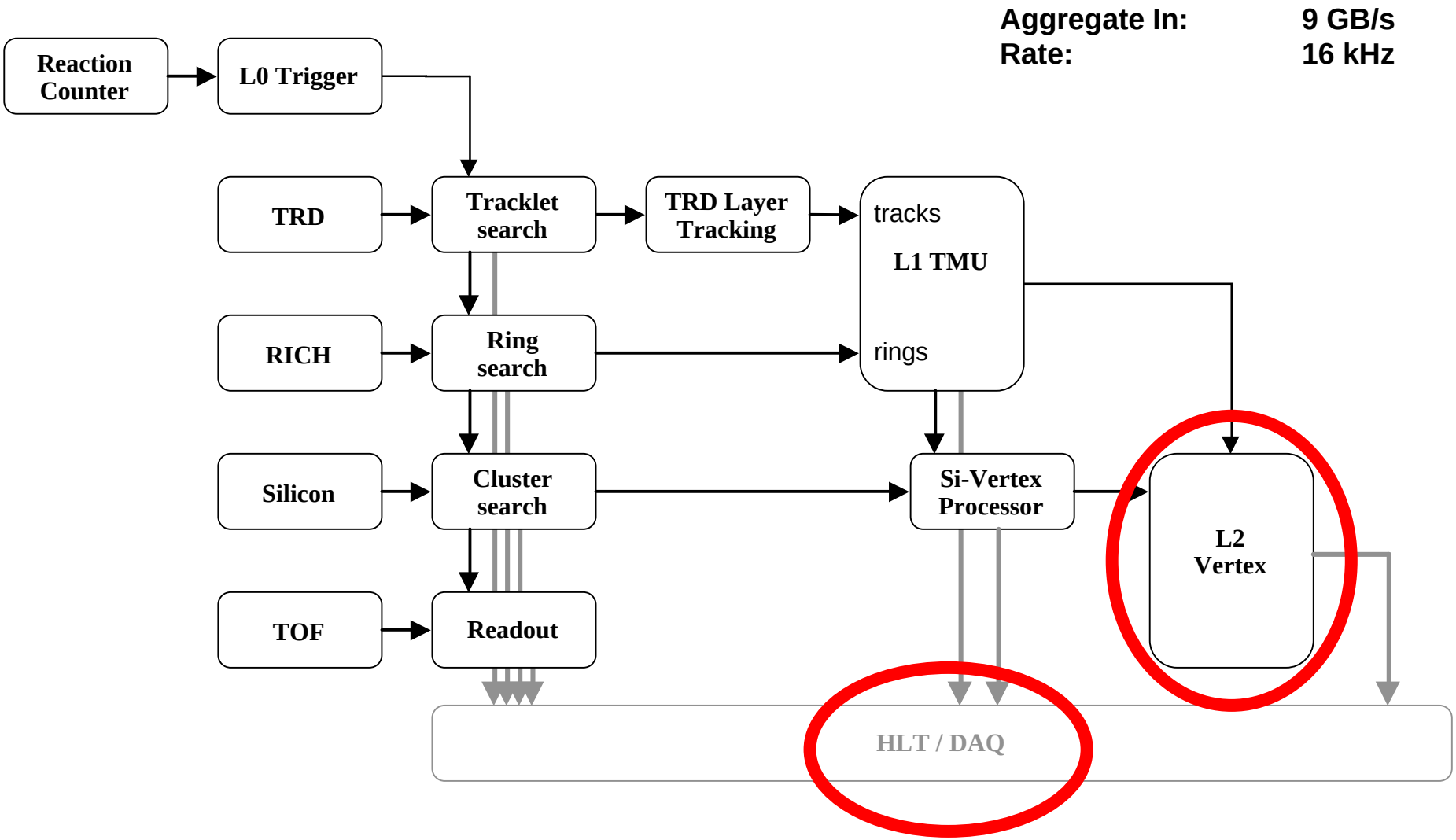
128B @ PCI 64/66, 1xALTERA

Address (1 cc)	Wait (**7 cc**)	Data (16 cc)	Idle (10 cc)

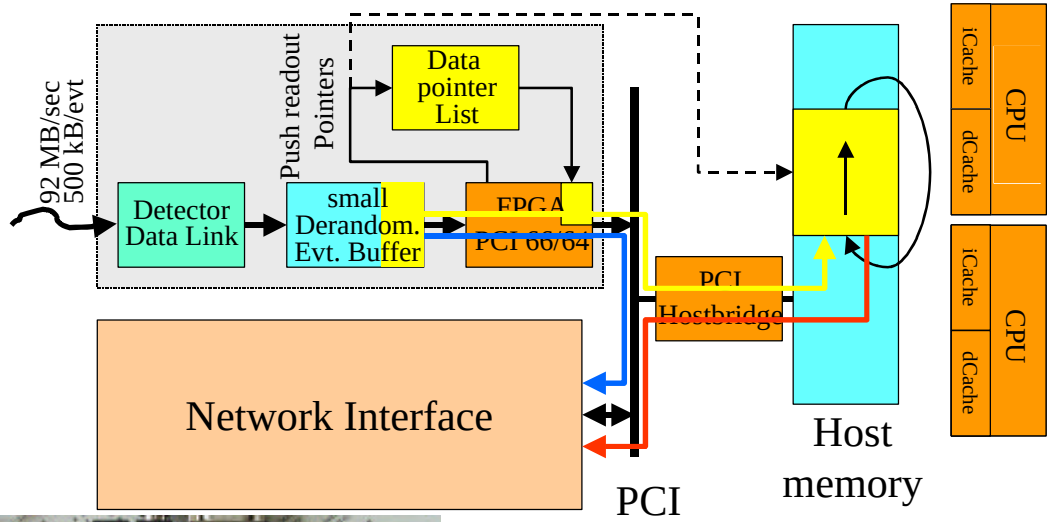
f_{max} = 1.94 MHz



Trigger Processor Farms

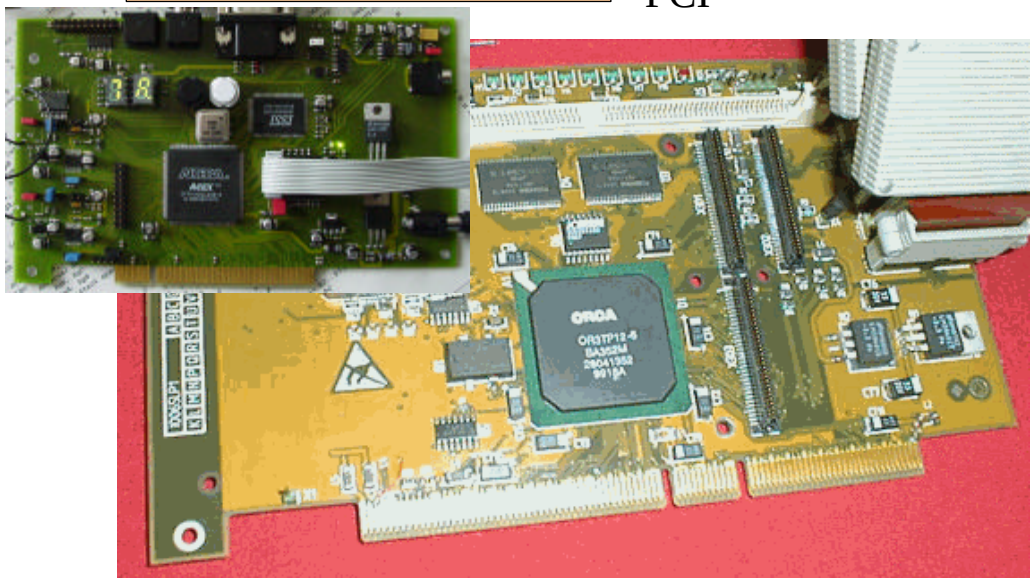


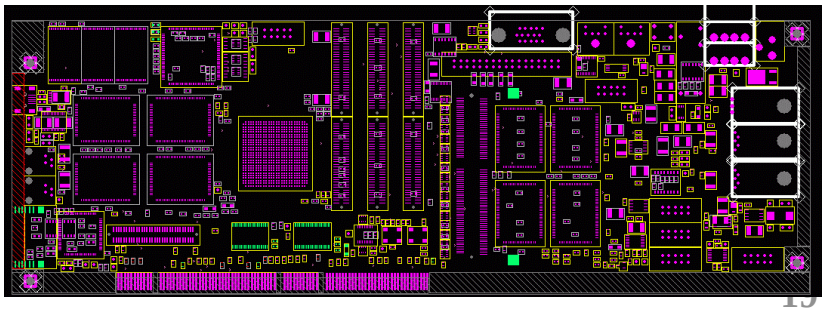
Receiving Detector Data into Cluster



- Probably cheapest memory available is in commodity computers
- Use that memory as event- and latency buffer
- Perform first level processing (e.g. cluster finder) where data exists
- Avoid unnecessary copying of data
- Use programmable FPGA to off-load host processors (FPGA Coprocessor)

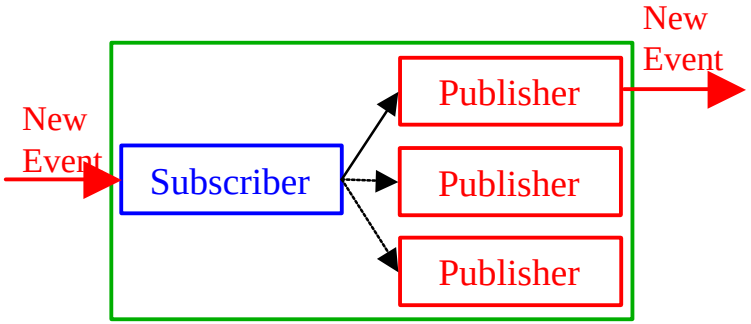
- Make PCI(-X) baseline bus enabling use of all commodity computers
- Use CMC form factor for optical detector data link circuitry on mezzanine





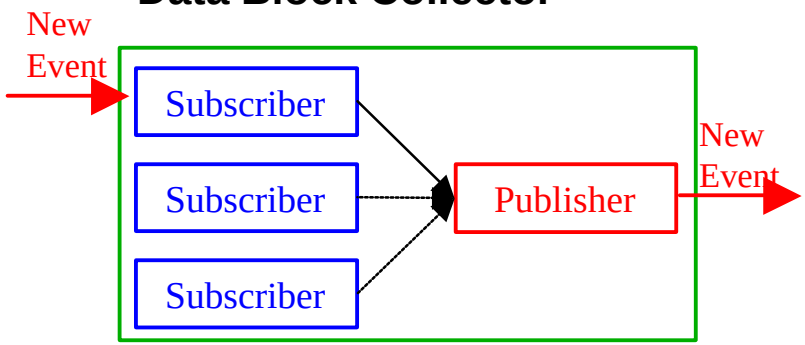
Modular Cluster Communication Package

Data Block Distributor

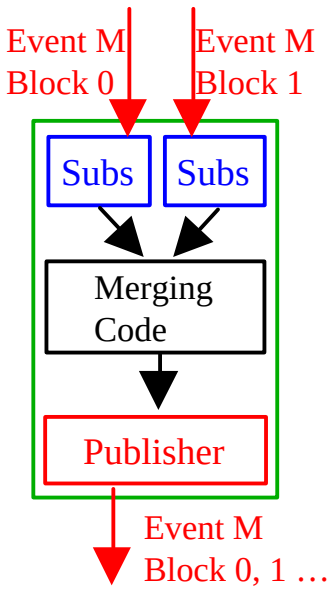


Load balancing
Fan-out

Data Block Collector



Data Block Merger



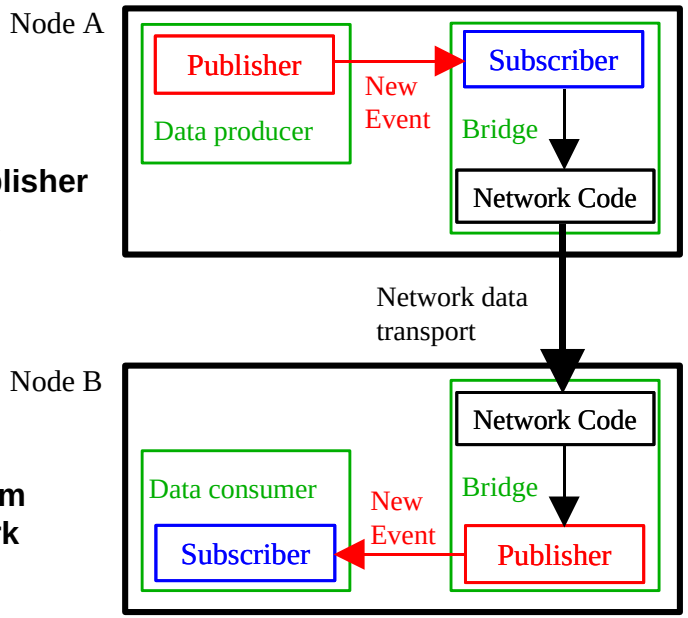
Bridging between Nodes

SubscriberBridgeHead

- **Subscribes to upstream**
- **Gets input data from publisher**
- **Sends data over network**

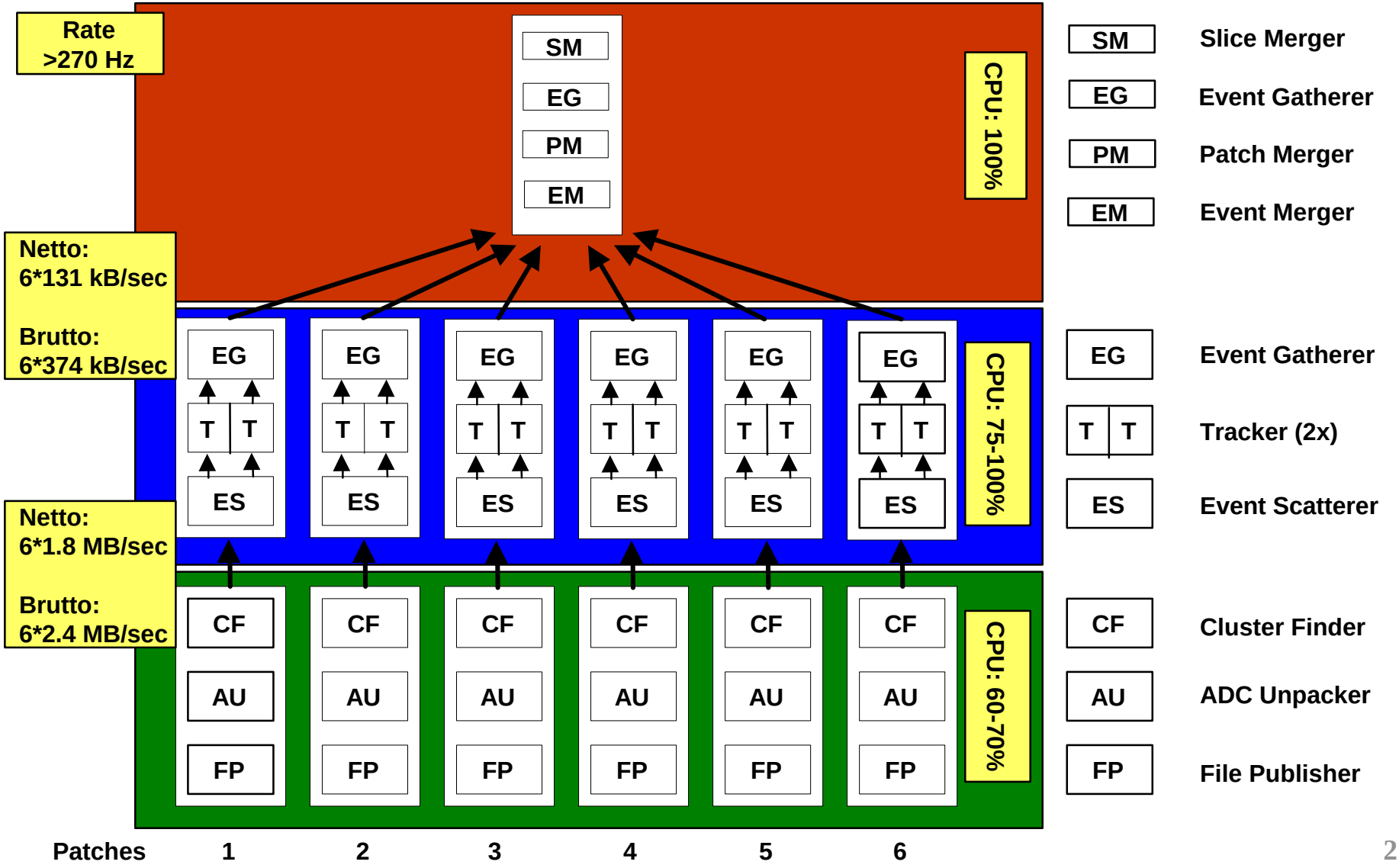
PublisherBridgeHead

- **Publishes to downstream**
- **Reads data from network**

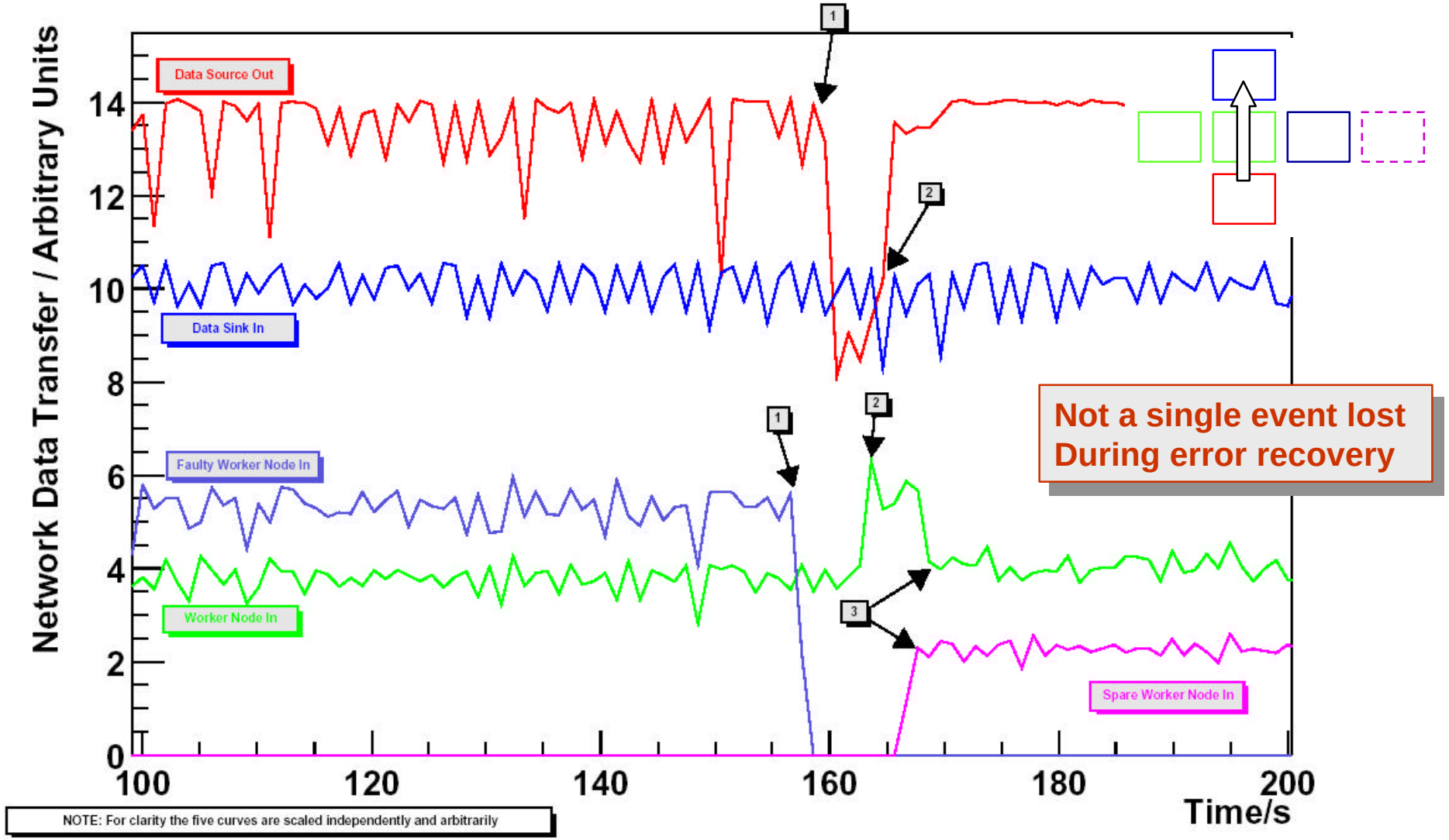


System benchmark

- **pp pileup removal at 270 Hz** (TCP/IP, fast ethernet)

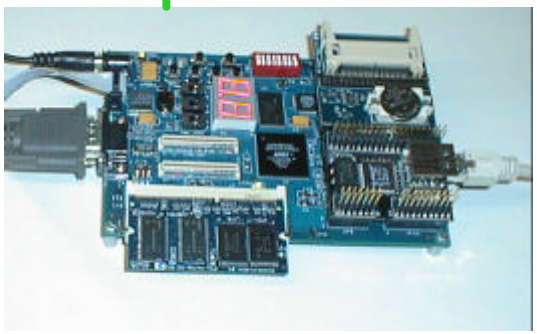


Publisher / Subscriber Fault Tolerance



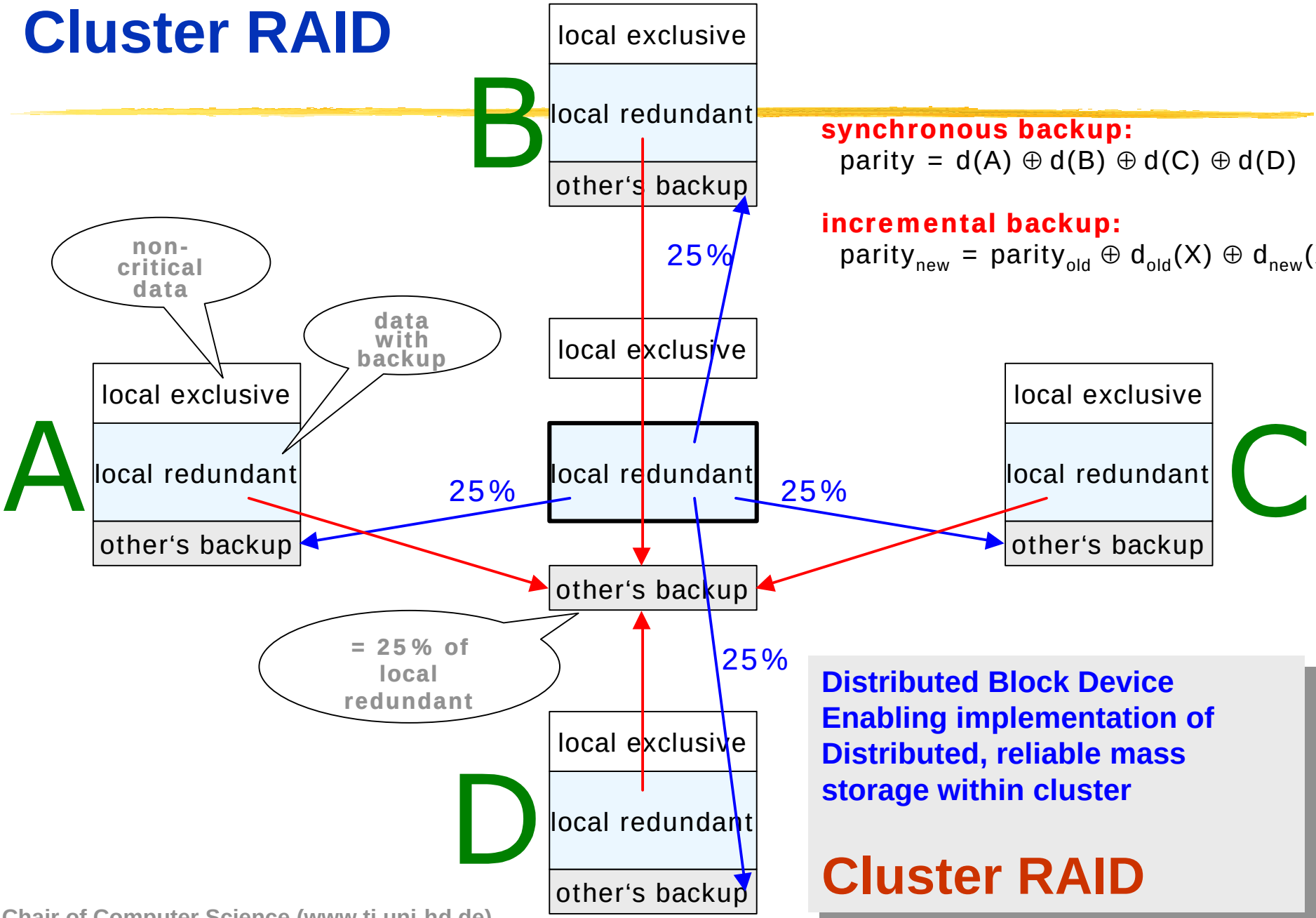
Not a single event lost During error recovery

Fault Tolerance in large Clusters



-PCI Linux PC in every PC
-BIOS access
-Hardware controlling
-POST codes
-Floppy emulation
-Remote access to a broken PC
-Detecting hard- & software errors
-Starting self-acting error correction
-Web interface for Monitoring & interacting
- Part of DataGRID Project

Cluster RAID



synchronous backup:

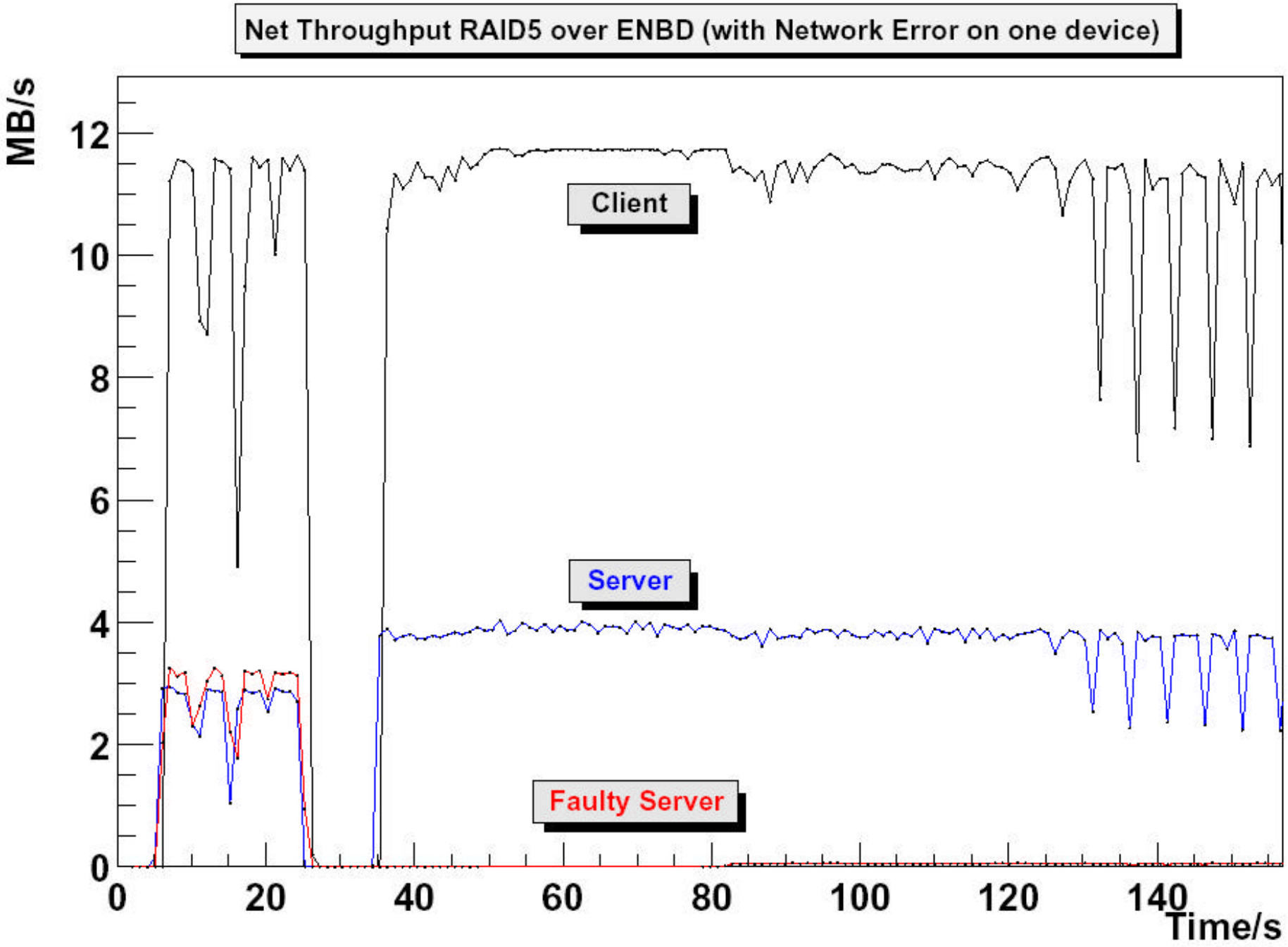
$$\text{parity} = d(A) \oplus d(B) \oplus d(C) \oplus d(D)$$

incremental backup:

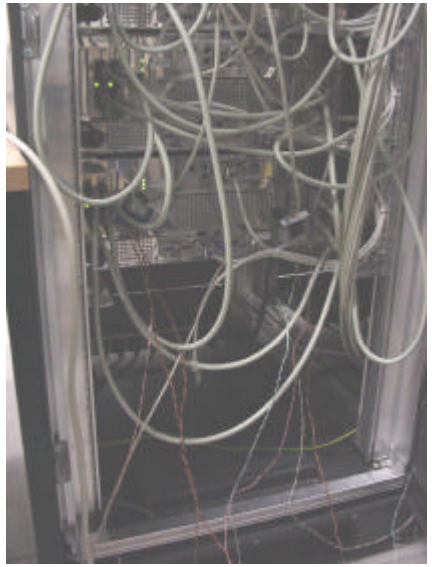
$$\text{parity}_{new} = \text{parity}_{old} \oplus d_{old}(X) \oplus d_{new}(X)$$

Distributed Block Device Enabling implementation of Distributed, reliable mass storage within cluster

Cluster RAID


Net Throughput RAID5 over ENBD (with Network Error on one device)
MB/s
12
10
8
6
4
2
0
0
20
40
60
80
100
120
140
Time/s
Client
Server
Faulty Server

Heidelberg HLT cluster



Summary - Conclusions

- PC Clusters are a powerful tool up to 1... 2 MHz Subevent Rates and 2...4 GB/sec transfer rates
  - Data easily transferred into L1 / L2 Caches
  - Scalability due to Moores law
- At higher (data or transaction) rates Systolic FPGA (Co)Processors
- ASICs for large channel counts
- Complex low-power readout and (pre)trigger circuitry possible and likely to benefit much from Moores Law
- High integration density and high-speed processes allow implementation of very inexpensive and low-power ADCs
- Co-existence of ADC and clocked circuitry demonstrated to work well
- Challenging GSI CBM Program does not seem to present any unsurmountable problems w.r.t. trigger and readout