JUSTUS-LIEBIG-UNIVERSITÄT GIESSEN

ATCA-based Computation Platform for Data Acquisition and Triggering in Particle Physics Experiments

Under the collaboration of

JLU, IHEP, and KTH

Presented by

Ming Liu

Outline

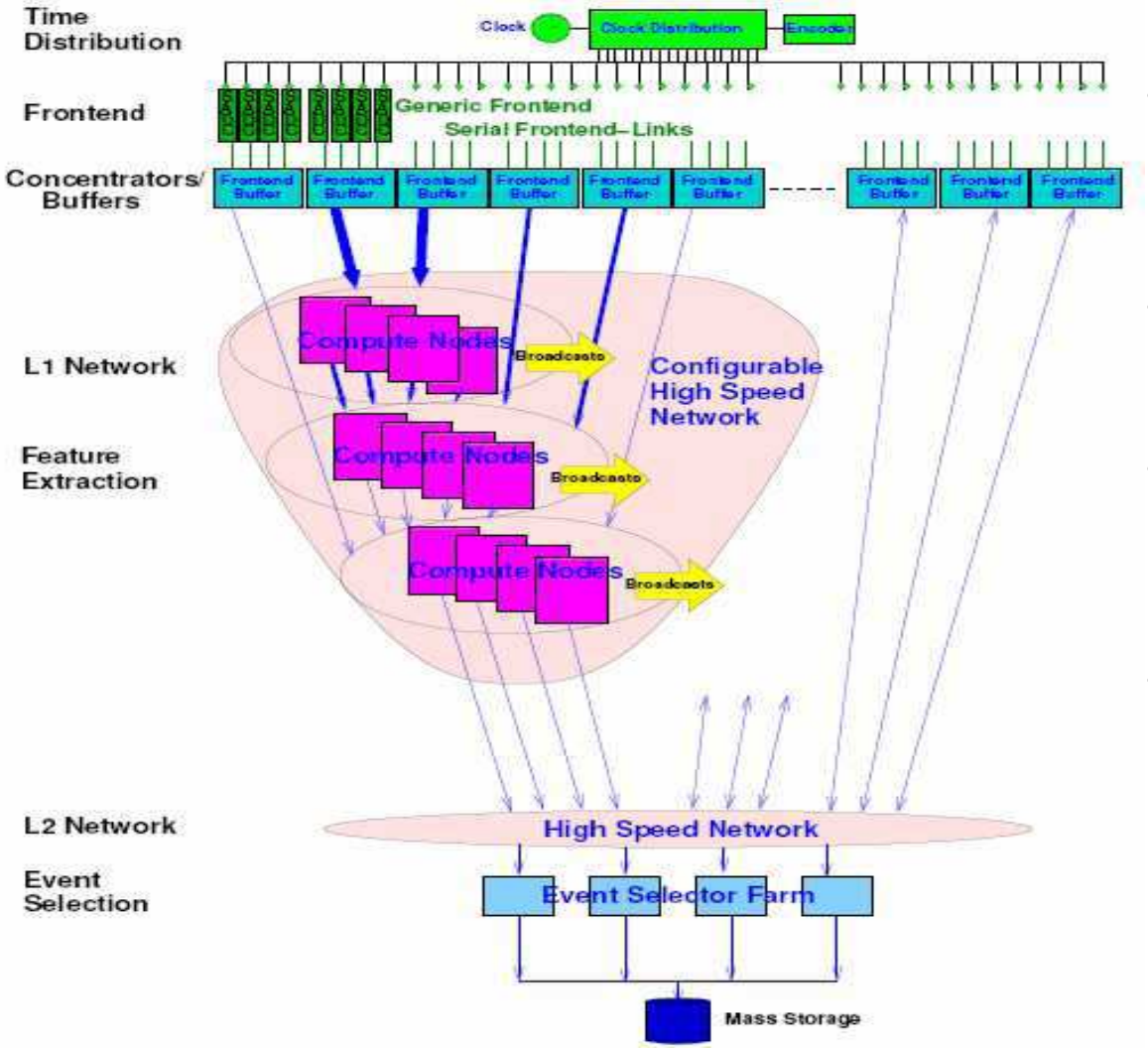
- Nuclear & particle physics background
- ATCA platform architecture
- Compute Node (CN) HW design
- HW/SW co-design on FPGAs
- Detector-specific algorithm development
- Current status and outlook

Physics Experiments

Modern nuclear & particle physics experiments feature

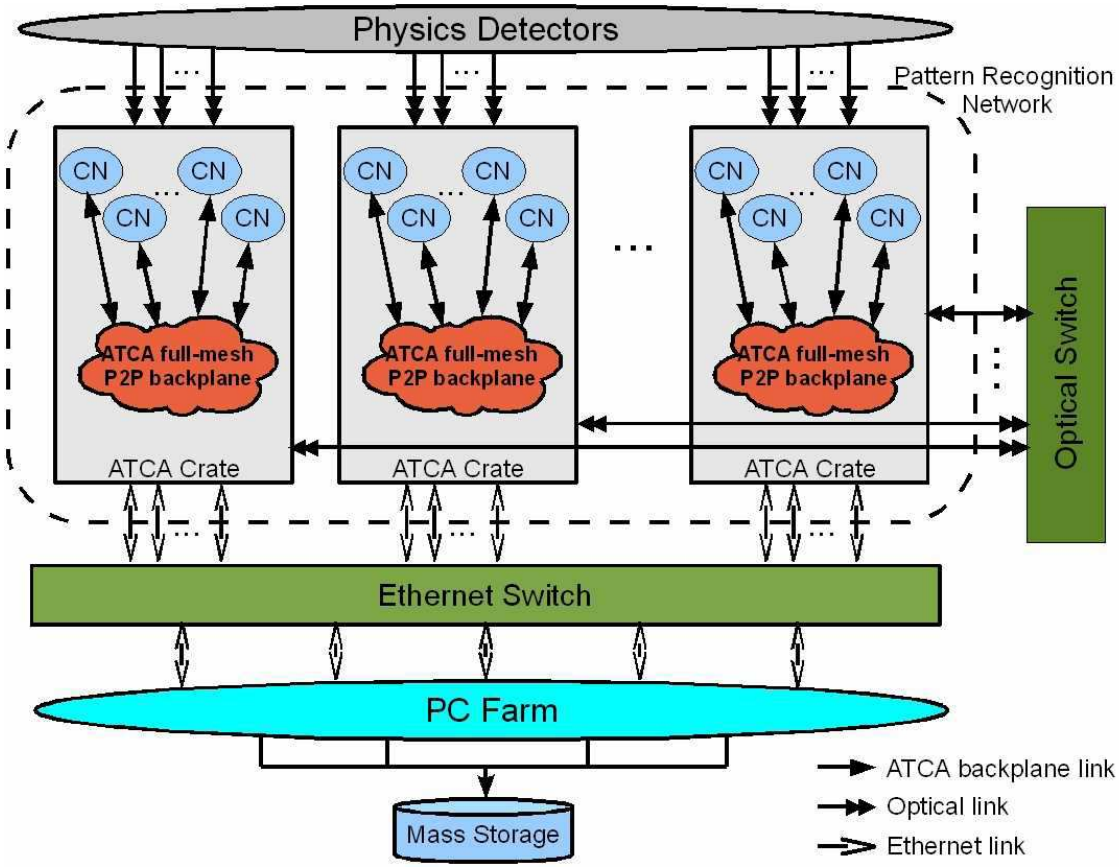
- High reaction rate (PANDA 10-20 MHz)
- Large channel counts from detectors (>10^5 channels)
- High data rate (PANDA 200 GBytes/s)
- Scalability requirements (new detectors, higher data rate, etc.)

DAQ and Trigger System



- Challenge: high reaction rate and high data rate (PANDA, reaction rate of 10-20 MHz, data rate up to 200 GB/s!!!)
- Not possible to entirely store all the data, due to the storage capacity limitation
- Only a rare proportion (e.g. $1/10^6$) is of interest for extensive offline analysis. The background can be discarded on the fly.
- Pattern recognition algorithms used to identify interesting data.
- Motivation: a powerful, scalable, and general-purpose platform for DAQ & Triggering computing

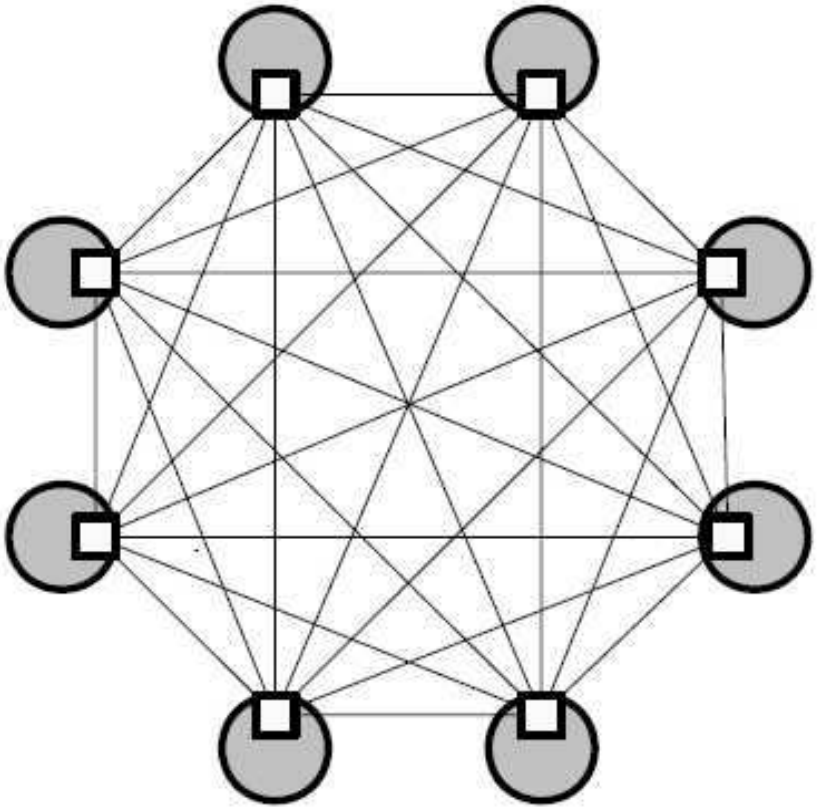
Computation Platform Architecture



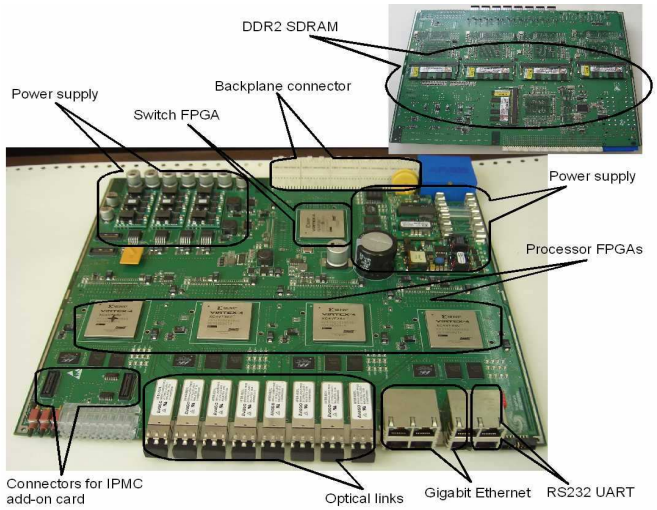
- Pattern recognition algorithms implemented
- Multiple CNs for algorithm partition and parallel/pipelined processing
- CNs internally interconnected by the full-mesh ATCA backplane
- External interconnections:
  - Optical links
  - Gigabit Ethernet

ATCA Full-mesh Backplane

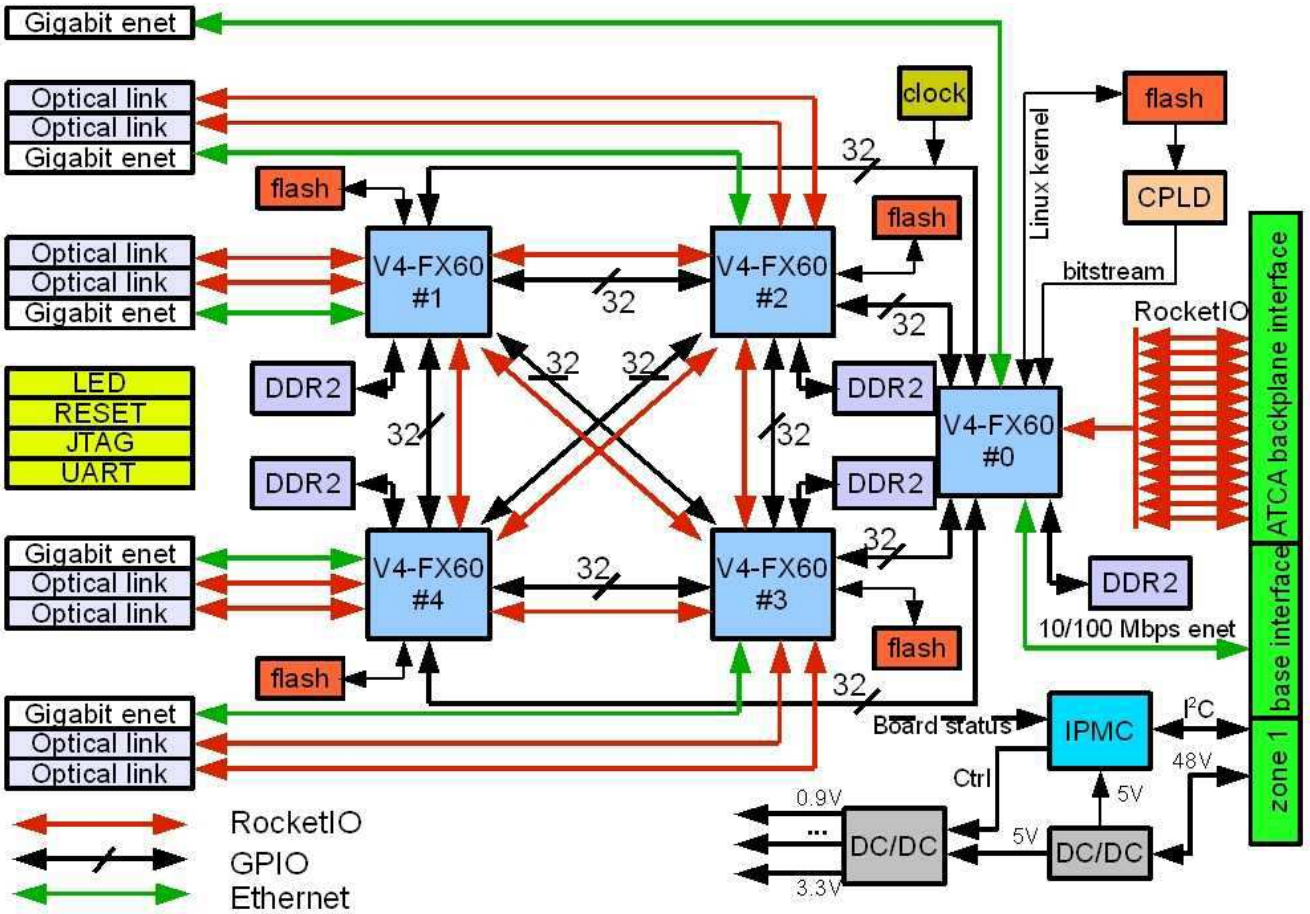
- Full-mesh backplane network
- High flexibility to correlate results from different algorithms
- High performance



Compute Node

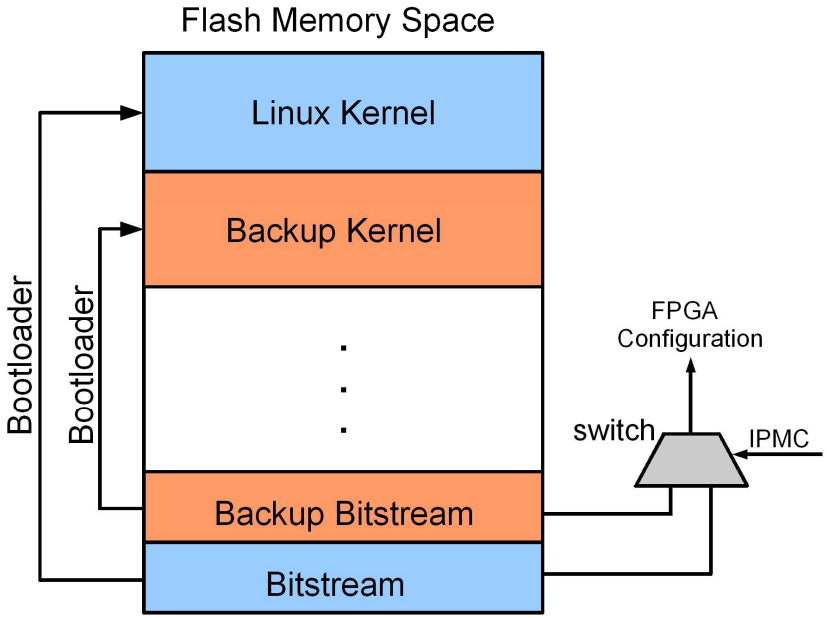


- Prototype board with 5 Xilinx Virtex-4 FX60 FPGAs
- 4 FPGAs as algo. Processors
- 1 FPGA as a switch
- 2 GB DDR2 per FPGA
- Full-mesh communication onboard
- IPMC, Flash, CPLD, ...
- External links
  - Optical links
  - Gigabit Ethernet



Remote Reconfigurability

- Remote reconfigurability is provided to solve the spatial constraint in experiments.
- Both the OS kernel and the FPGA bitstreams are stored in the NOR flash memories.
- With the support of the MTD driver, the bitstreams and the kernel can be overwritten and upgraded in Linux.
- Commands are issued remotely through network.
- Backup mechanism to guarantee the system alive.



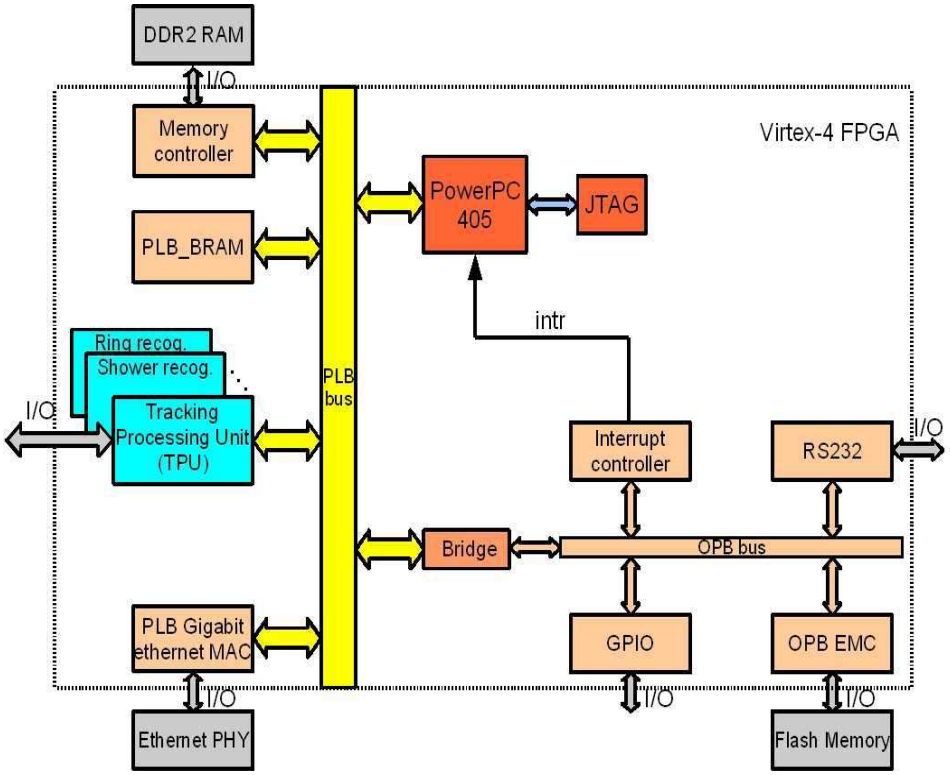
HW/SW Co-design on FPGAs

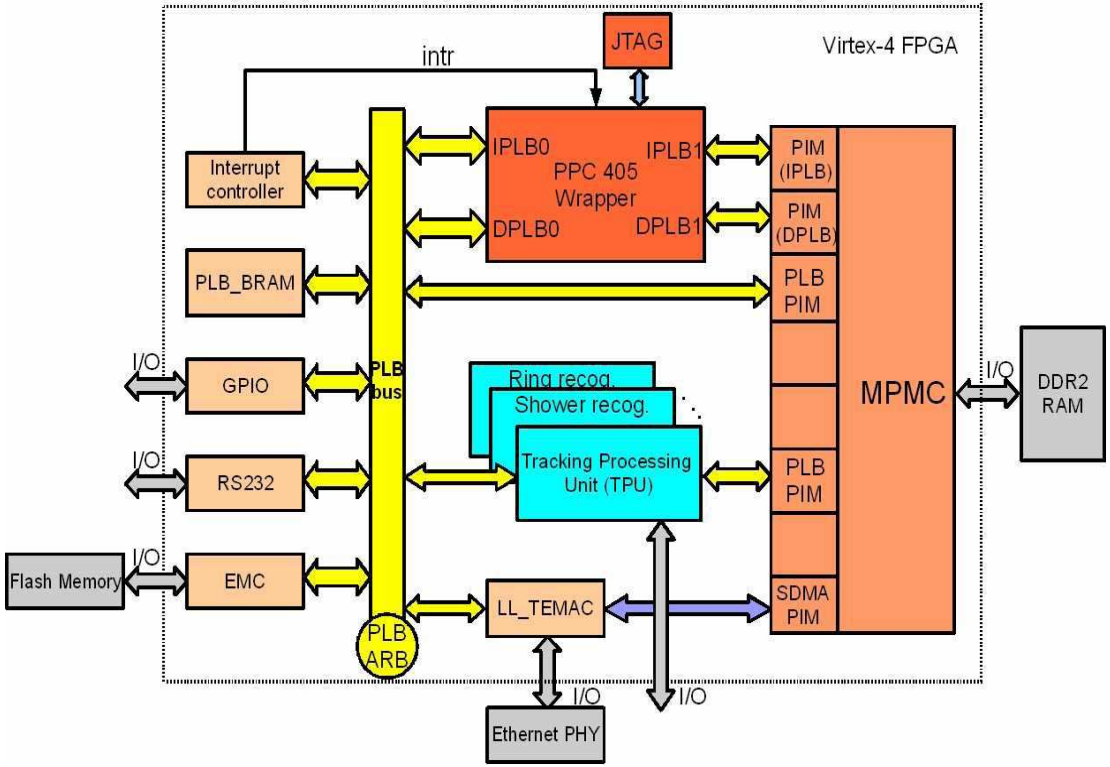
Aim: to ease and accelerate development on CNs for different experiments & algorithms

Partitioning strategy:

- Computation-intensive algorithms implemented in the FPGA fabric for high performance and real-time features (parallel & pipelined processing in HW)
- Slow controls in SW (OS + Applications):
  - To remotely upgrade the HW and SW designs
  - Network test and measurements
  - To display and adjust experimental parameters
  - ....
- Communication stack processing (TCP/IP) in Linux OS in SW

HW Design on FPGAs





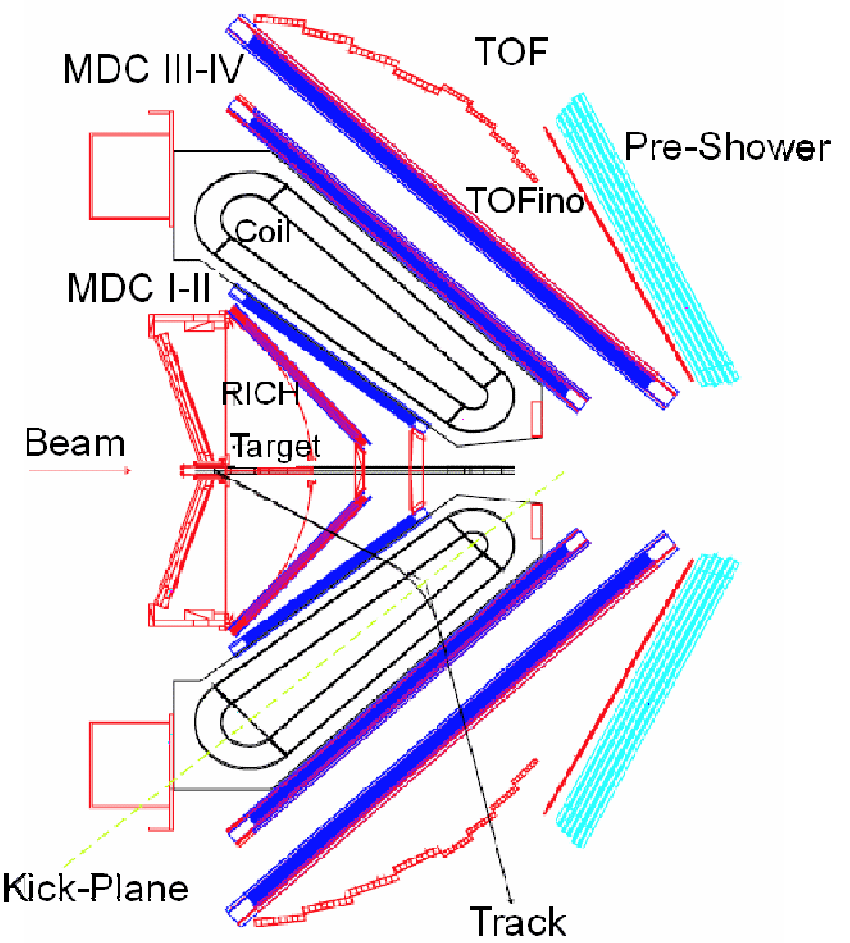
- A uniform system design for all applications (bus-based or MPMC-based)
- Customized processing modules for different algorithms
- Easy system integration with the guarantee of high performance

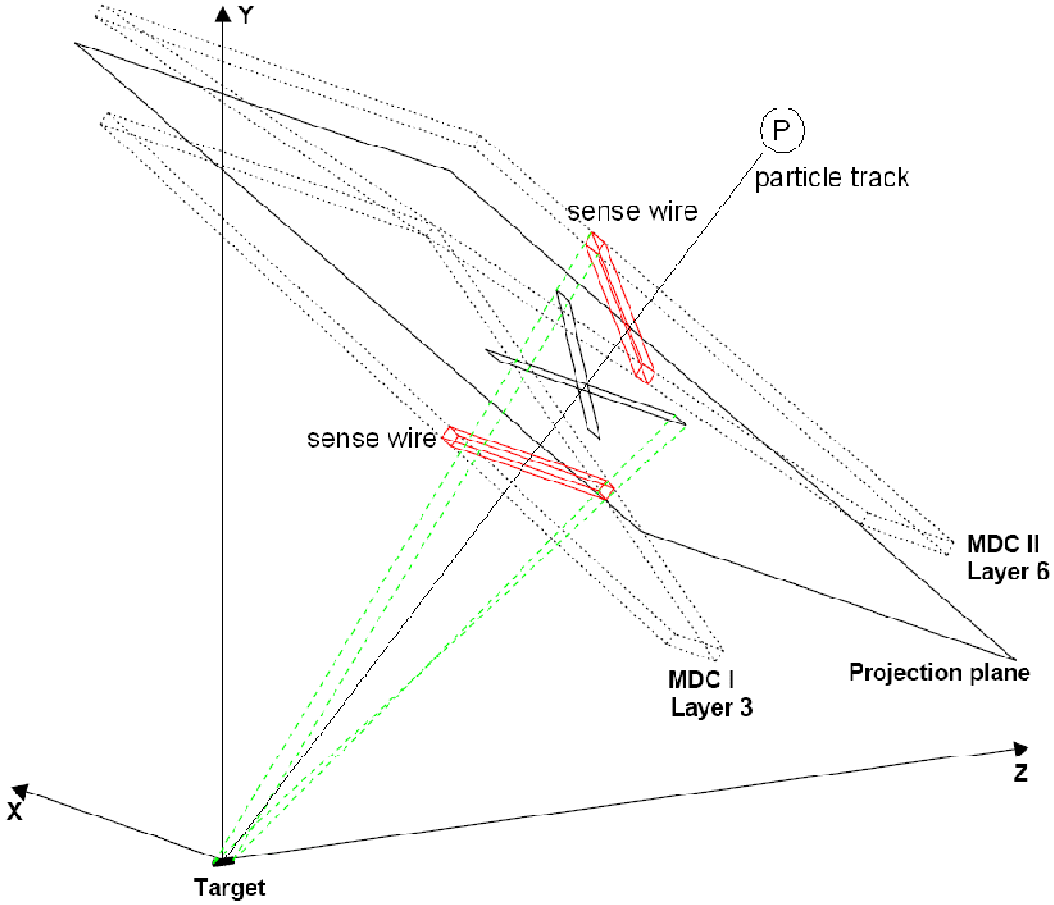
SW Design

- Open-source Linux on embedded PowerPCs
- Physicists favorite OS and easy to operate and program
- Device drivers:
  - For Ethernet, UART, Flash memory, etc.
  - For customized processing units
- Applications for slow controls:
  - High level scripts
  - C/C++ programs
  - Webpages on Apache server
  - Java program on VM
  - …
- Many tools provided, NFS, telnet, …
- Software cost: almost zero

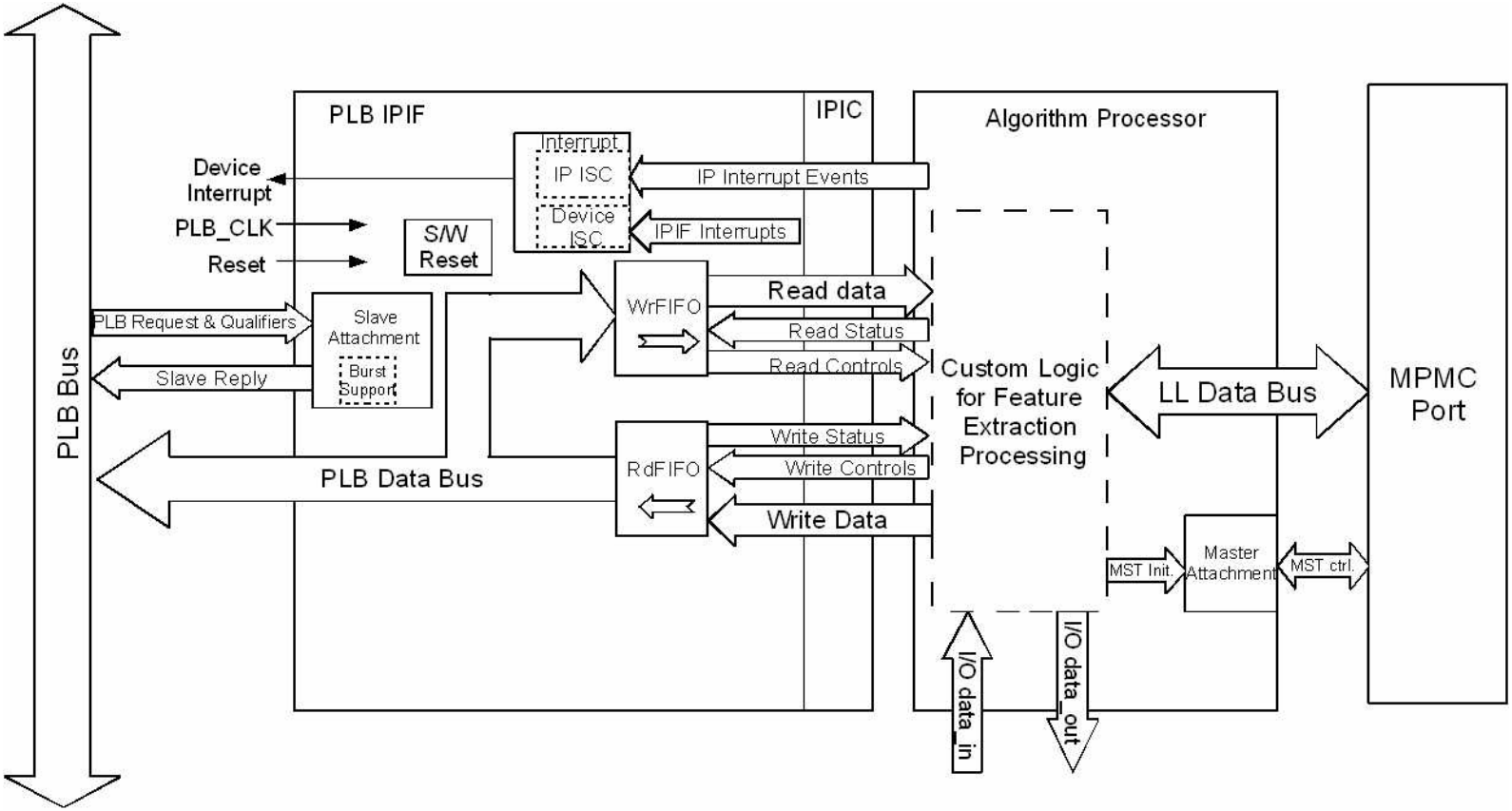
Algorithm Development

Example 1: HADES track reconstruction (inner)





HADES Track Reconstruction



- PLB slave interface (PLB IPIF) for system control
- LocalLink master interface for data movement from/to memory
- Algorithm processor (tracking processor)

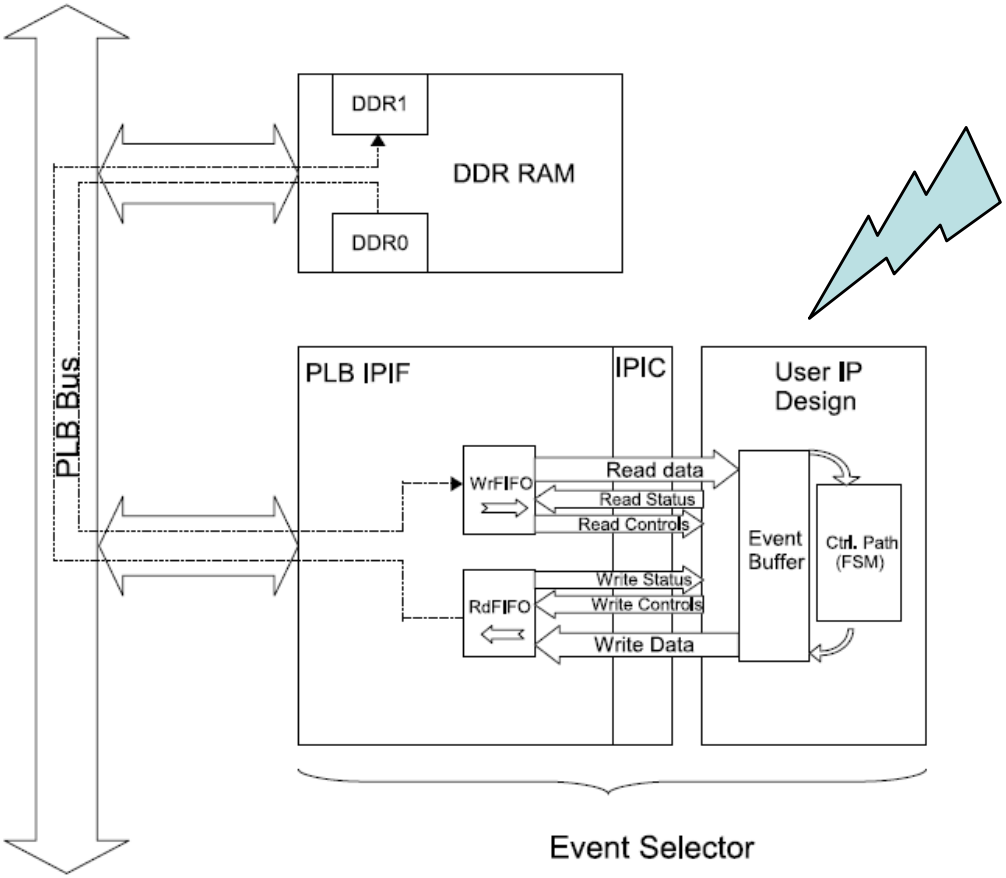
HADES Track Reconstruction

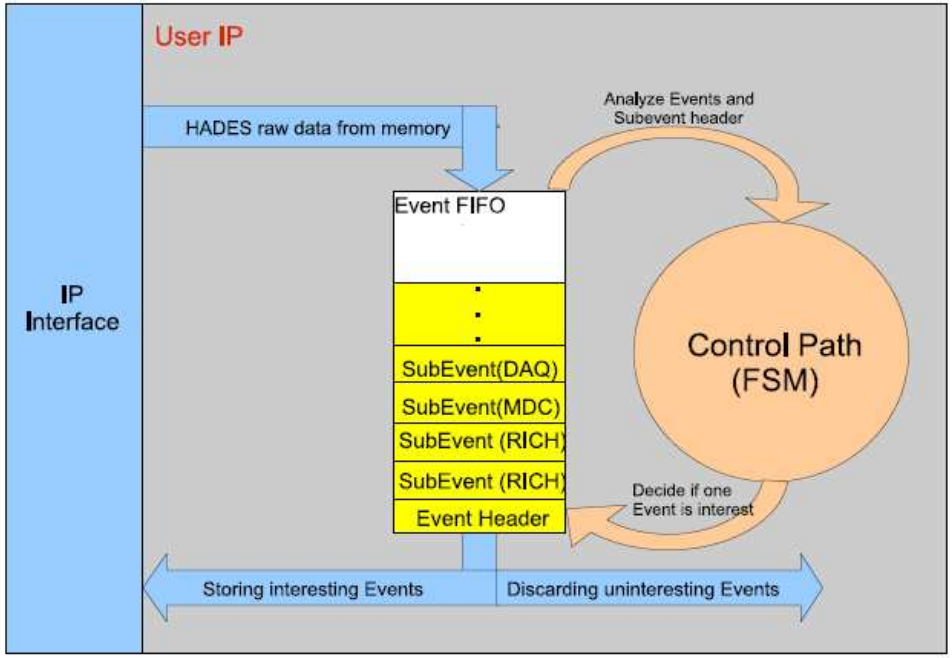
Experimental results:

- FPGA resource utilization of Virtex-4 FX60 (<1/5) – acceptable!
- Timing limitation: 125 MHz without optimization
  - Clock frequency fixed at 100 MHz, to match the PLB speed
- Processing capability measurements:
  - A C program running on the Xeon 2.4 GHz computer as the software reference
  - Measurement points on different wire multiplicities (10, 30, 50, 200, 400 fired wires out of 2110)
  - Speedup of 10.8 – 24.3 times per module compared to the software solution
  - Multiple cores integrated on one FPGA for parallel processing (even higher performance speedup of two orders of magnitude)

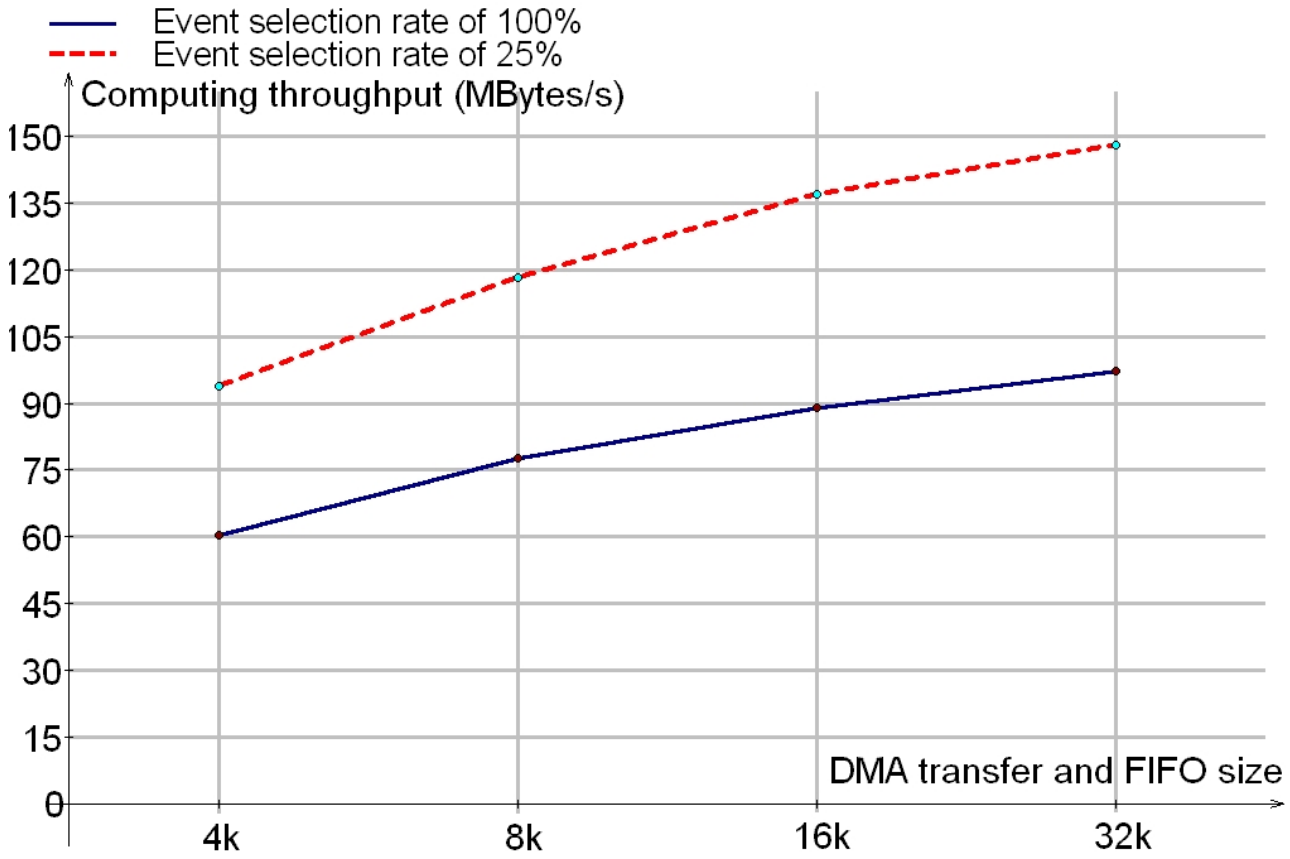
Algorithm Development

Example 2: A universal event selector





Event Selector



Measurement results:

- Processing capability of data flow
- Event selection rates of 100% & 25%
- Different FIFO sizes (DMA sizes)
- Processing throughput of ~150 & ~100 MB/s (could be higher)

Other Algorithms for HADES & PANDA

Except for the HADES MDC tracking, other algorithms are also being developed for HADES and PANDA:

- HADES ring recognition for RICH (Johannes Roskoss)
- HADES shower recognition for Electromagnetic Shower (Andreas Kopp)
- PANDA tracking for Straw Tube Tracker (David Muenchow)
- …

All algorithms are to be implemented on CNs for HW processing.

Current Status

- The first version CN PCB has been tested
  - Optical links (@ 2Gbps to TRB2, 0 bit error for 150-hour test)
  - Gigabit Ethernet (UDP/IP:~400 Mbps, TCP/IP:~300 Mbps)
  - JTAG chain
  - CPLD+Flash system start-up mechanism and remote reconfigurability
  - DDR2 SDRAM
  - Other peripherals
- Algorithms under development & implementation
- More than 10 papers & posters have been published based on CN & its development methodology, both in physics area, and in electronic & computer area.

Outlook

- The next version PCB will be produced soon.
- More than 3 boards for network investigation
- All algorithms to be implemented
- Network parallel/pipelined processing investigation with multiple CNs
- In the end of 2009, one running ATCA crate for HADES upgrade
- For WASA, BESIII, PANDA, SuperBelle … ?

Thanks for your attention!