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

Ming Liu¹,³, Johannes Lang¹, Shuo Yang¹, Tiago Perez¹, Wolfgang Kuehn¹, Hao Xu², Dapeng Jin², Qiang Wang², Lu Li², Zhen'An Liu², Zhonghai Lu³, and Axel Jantsch³

1 II. Experimental Physics Institute, Justus-Liebig-University in Giessen, Germany
2 Experimental Physics Center, Institute of High Energy Physics, China
3 Dept. of Electronic, Computer and Software Systems, Royal Institute of Technology, Stockholm, Sweden
Outline

- Particle physics background
- Related work
- Computation platform architecture
- FPGA node development
- Current status
- Conclusion and future work
Particle Physics Experiments

- Particle physics is a branch of physics that studies the elementary constituents of matter, and the interactions between them.
- Many elementary particles do not occur under normal circumstances in nature, but can be created and detected during energetic collisions of other particles.
- Particle physics experiments generate collisions between beam particles and target particles, and study produced particles with huge/complex detector systems.
- Examples:
  - HADES & PANDA @ GSI, Germany
  - LHC @ CERN, Switzerland & France
  - BES III @ IHEP, China
  - WASA @ FZ-Juelich, Germany
  - ......
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 reconfigurable and scalable computation platform for high data rate processing
Related Work

- Previously commercial bus systems, such as VMEbus, FASTbus, CAMAC, etc., were used for DAQ and triggering.
  - Time-multiplexing of the system bus exacerbates the data exchange efficiency and cannot meet high-performance requirements.
- The solution of existing reconfigurable computers sounds good, but not suitable for physics experiment applications:
  - Some are augmented computer clusters with FPGAs attached to the system bus as accelerators. (Bandwidth bottleneck between the microprocessor and the accelerator)
  - Some are standalone boards. (Not straightforward to scale the system to a large size, due to the lack of efficient inter-board connectivity)
  - Flexible and massive communication channels are required to interface with detectors and the PC farm.
  - All-board-switched or tree-like topology may result in communication penalty between algorithm steps. (P2P direct links are preferred.)
Computation Platform Architecture

- Pattern recognition algorithms implemented in the computation network.
- Multiple Compute Nodes (CN) for algorithm partition and parallel/pipelined processing.
- CN boards 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 on-board
- External links:
  - Optical links
  - Gigabit Ethernet
  - IPMC, Flash, CPLD, ...
Old bus-based architecture (PLB & OPB)
- CPU & Fast peripherals on PLB
- Slow peripherals on OPB
- Tracking Processing Unit (TPU) on PLB as a fast device

New LocalLink-based architecture
- Multi-Port Memory Controller (8 ports)
- Direct access to the memory from the device
- TPU interfaced to MPMC directly
PLB-based Algorithm Processor Design

- PLB master & slave interface (PLB IPIF)
  - Master/DMA transfer
  - Burst mode
  - Data FIFO
  - Interrupt
  - Algorithm processor
LocalLink-based Algorithm Processor Design

- PLB slave interface (PLB IPIF) for system control
- LocalLink master interface for data movement from/to memory
- Algorithm processor
Open-source embedded Linux on the embedded PowerPCs

Device drivers:
- For Ethernet, RS232, Flash memory, etc.
- For the customized processing modules

Applications for slow controls:
- High level scripts
- C/C++ programs
- Webpages on the Apache server
- Java programs on the VM

Software cost: zero budget!
Remote Reconfigurability

- Remote reconfigurability is desired due to the spatial constraint in experiments.
- Both the OS kernel and the FPGA bitstream are stored in the XOR flash memories.
- With the support of the MTD driver, the bitstream and the kernel can be overwritten and upgraded in Linux.
- Reboot the system and then the updated system will function.
- Backup mechanism to guarantee the system alive.
Current Status

- The first prototype PCB of the CN manufactured and under test
Current Status

- Optical links tested
  - Connect to the front-end Trigger and Readout Board version 2 (TRBv2) by optical links
  - Pseudo-random data transceiving
  - @ 2 Gbps with 8B/10B encoding
  - No bit error occurred for 150-hour test

- Gigabit Ethernet tested
  - Bulk data transfer measurement using “Netperf”
  - Features enabled: S/G DMA, checksum offloading, interrupt coalescing, jumbo frame of 8982, etc.
  - UDP/IP: ~400 Mbps     TCP/IP: ~300 Mbps
  - Bottleneck: 300 MHz CPU processing capability
Current Status

- Other peripherals tested or being tested
  - DDR2 SDRAM
  - JTAG chain
  - Flash and CPLD
  - ...
- Pattern recognition algorithms for the HADES upgrade are being or have been implemented and evaluated on the platform.
  - MDC track reconstruction [1]
  - Cherenkov ring recognition
  - Time-Of-Flight analysis
  - Event building and event selection
  - ...

Conclusion and Future Work

- An FPGA- and ATCA-based computation platform is being constructed for the DAQ and trigger system in modern particle physics experiments.

- The platform features high-performance, scalability, reconfigurability, and universal use for different application projects.

- A co-design methodology is proposed to develop applications.
  - HW: system design + customized processing modules
  - SW: Linux OS + device drivers + application programs

- Design flaws will be fixed in the later PCB versions.
- The network communication will be studied by multiple CN PCBs.
- All pattern recognition algorithms are to be implemented in the FPGA fabric.
Thanks for your attention!