Experience with Hybrid Intel® Xeon® FPGA System

Christian Färber
CERN Openlab Fellow
LHCb Online group

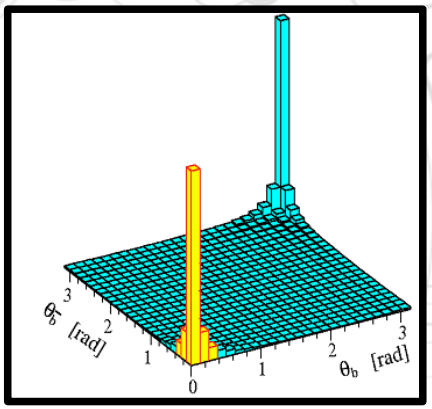
On behalf of the LHCb Online group and the HTC Collaboration

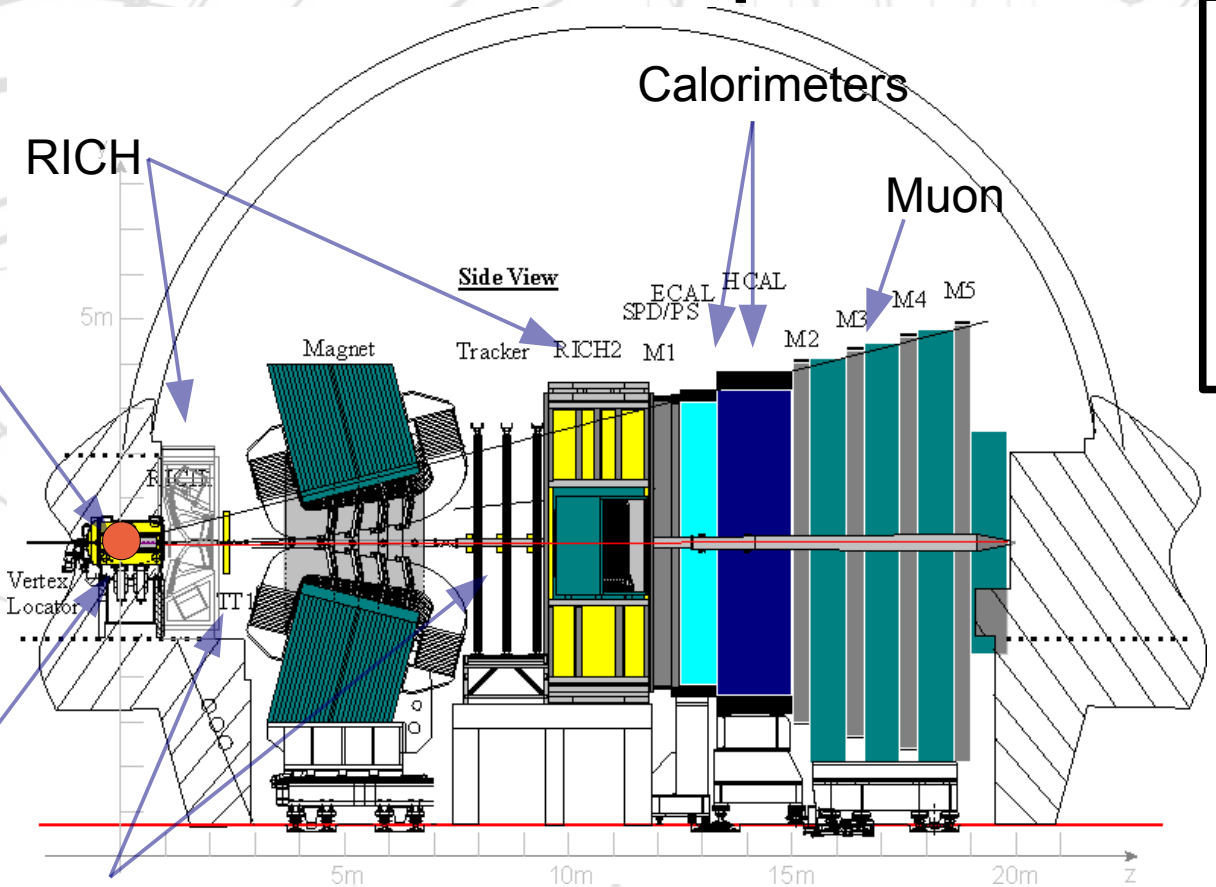
IML Machine Learning Working Group Meeting
23.11.2016

HTCC

- High Throughput Computing Collaboration
- Members from Intel and CERN LHCb/IT
- Test Intel technology for the usage in trigger and data acquisition (TDAQ) systems
- Projects
  - Intel® KNL computing accelerator
  - Intel® Omni-Path 100 Gbit/s network
  - Intel® Xeon/FPGA computing accelerator

Detector Example: LHCb



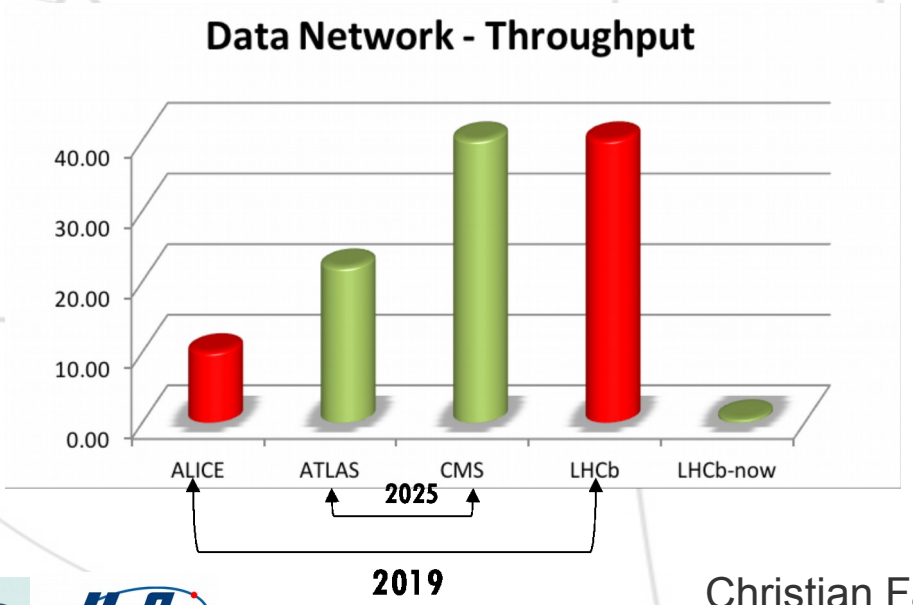


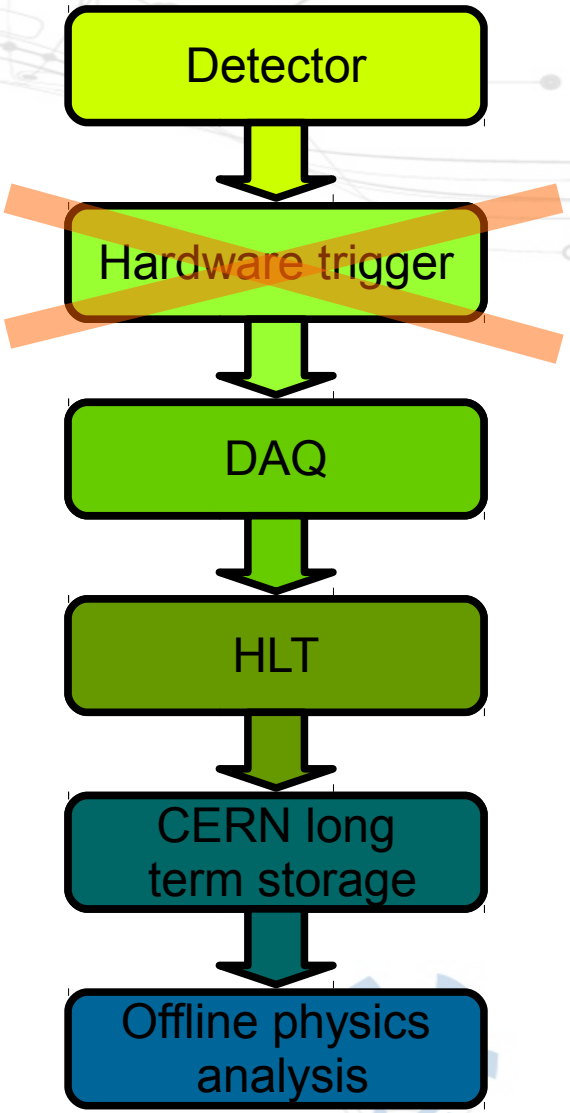
- Single-arm spectrometer designed to search new physics through measuring CP violation and rare decays of heavy flavour mesons.

- 40 MHz proton proton collisions
- Trigger with 1 MHz, upgrade to 40MHz
- Bandwidth after upgrade up to 40TBit/s

Future challenges

- Higher luminosity from LHC
- Upgraded sub-detector Front-Ends
- Removal of hardware trigger
- EFF has to handle:
  - Larger event size (50KB to 100KB)
  - Larger event rate (1MHz to 40MHz)



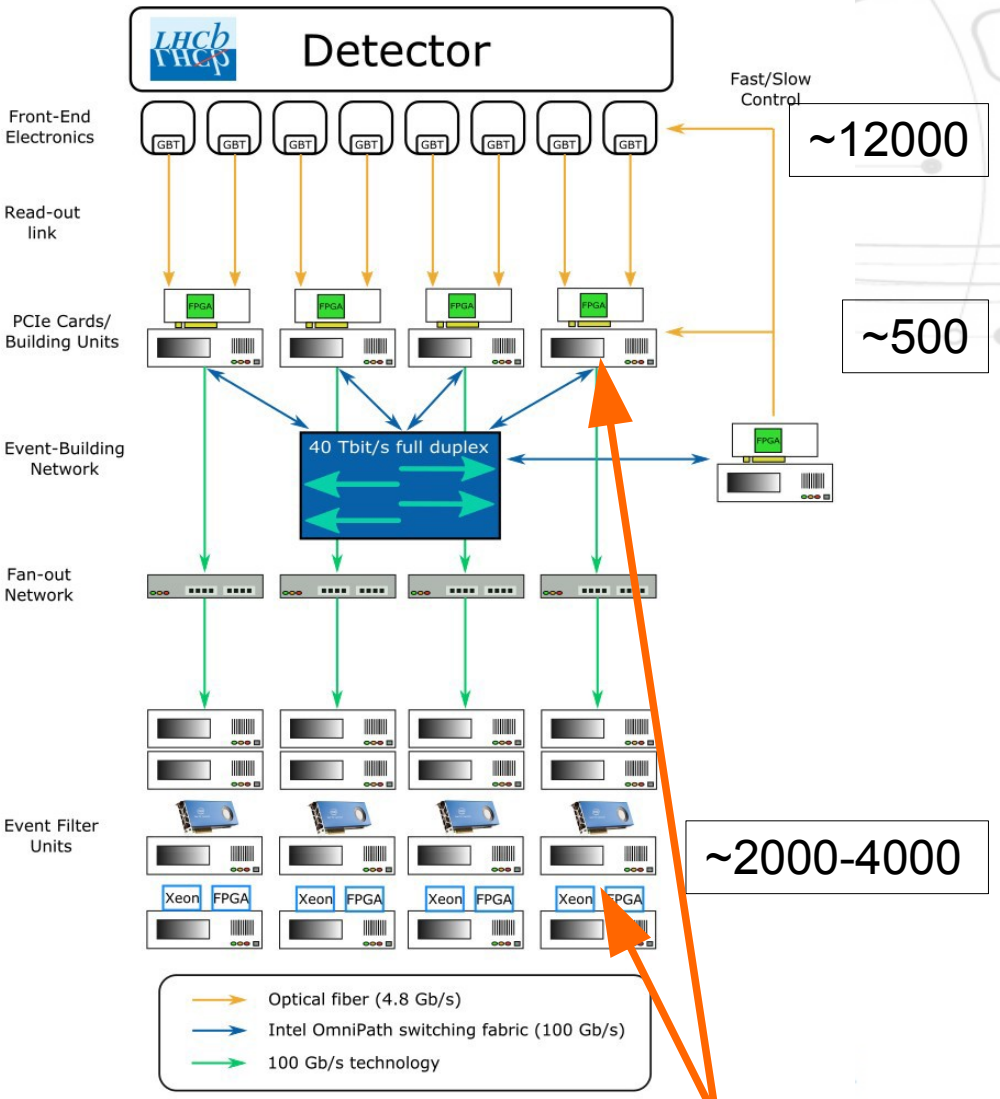


Upgrade Readout Schematic

- Raw data input ~ 40 Tbit/s
- EFF needs fast processing of trigger algorithms, different technologies are explored.

- Test FPGA compute accelerators for the usage in:
  - Event building
    - Decompressing and re-formatting packed binary data from detector
  - Event filtering
    - Tracking
    - Particle identification

- Compare with: GPUs, Intel® Xeon/Phi and other compute accelerators



Which technologies?

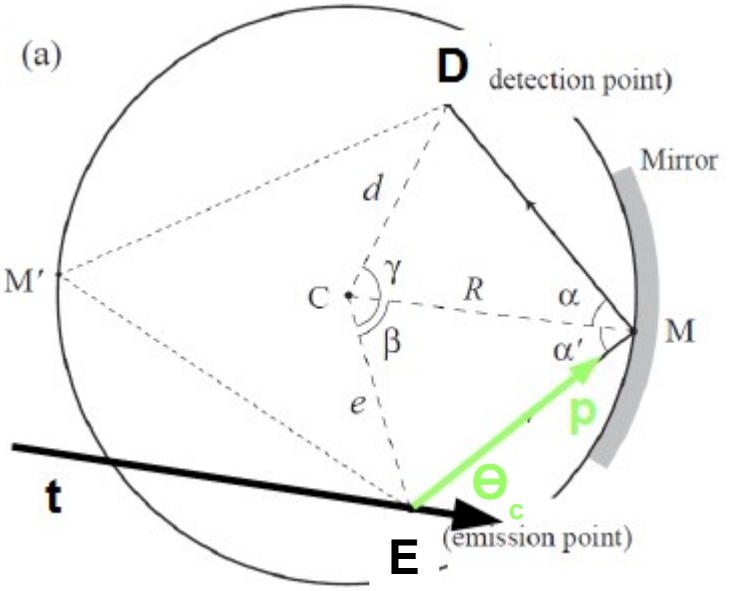
FPGAs as Compute accelerators

- Microsoft Catapult and Bing
  - Improve performance, reduce power consumption

- LHCb: Test for future usage in upgraded HLT farm:
  - Event building
  - Track fitting, pattern recognition, PID algorithms
- Current Test Devices in LHCb
  - Nallatech PCIe with OpenCL
  - Intel® Xeon/FPGA

Test case: RICH PID Algorithm

- Calculate Cherenkov angle Θ_c for each track **t** and detection point **D**
- RICH PID is not processed for every event, processing time too long!



Reference: LHCb Note LHCb-98-040

Calculations:

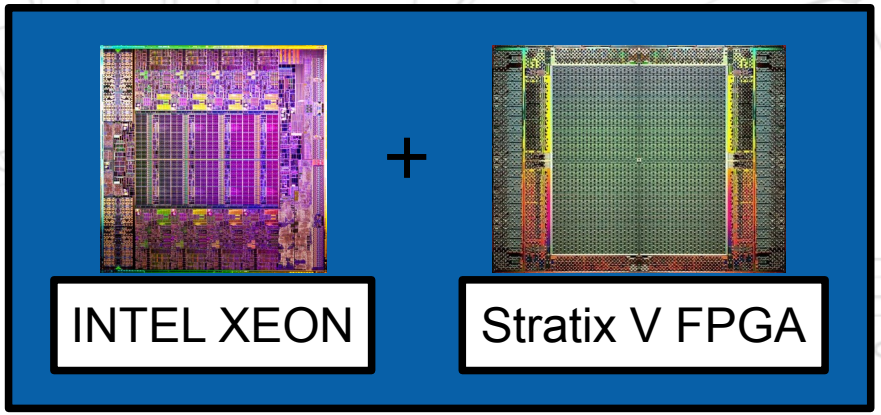
- solve quartic equation
- cube root
- complex square root
- rotation matrix
- scalar/cross products

Intel® Xeon/FPGA

- Two socket system:

  First: Intel® Xeon®
  E5-2680 v2

  Second: Altera Stratix V GX A7 FPGA
    - 234'720 ALMs, 940'000 Registers, 256 DSPs



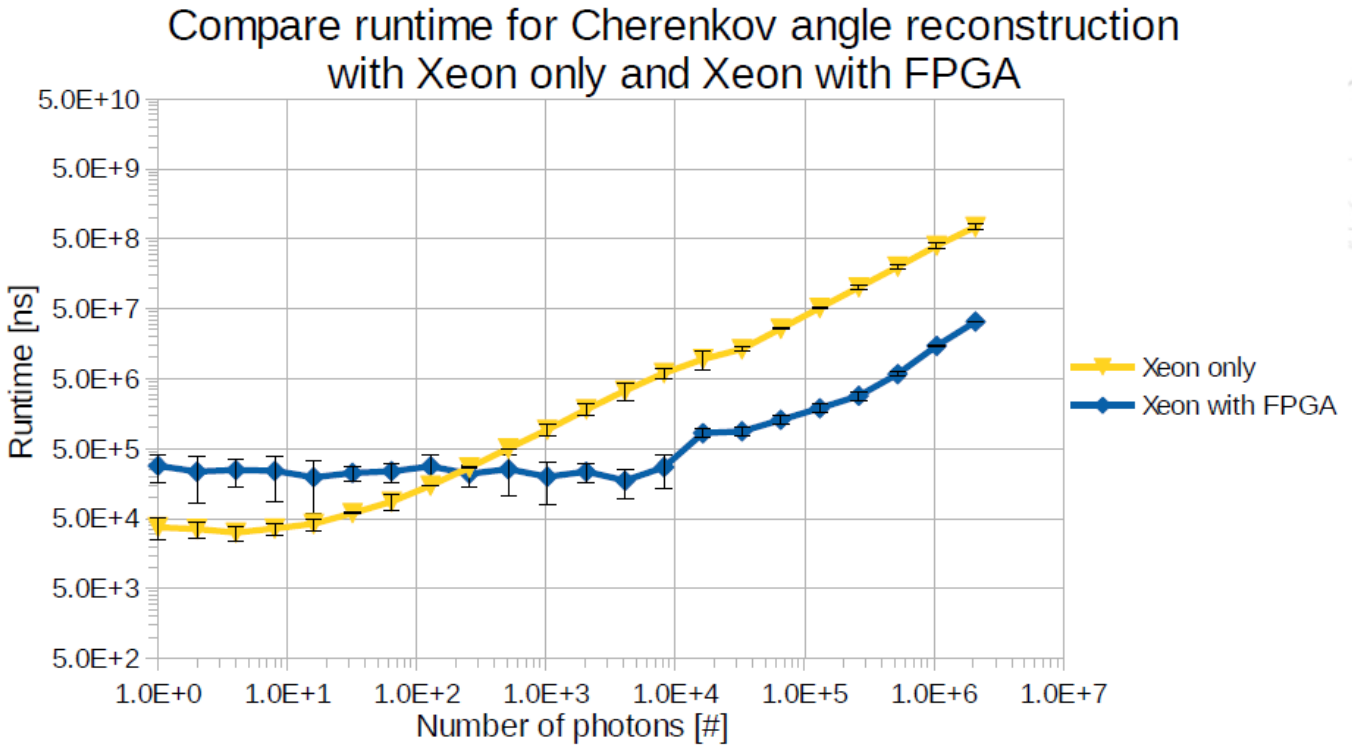
- Host Interface: high-bandwidth and low latency
- Memory: Cache-coherent access to main memory
- Programming model: Verilog now also OpenCL
- Power usage: Will be tested with next version

Implementation of Cherenkov Angle reconstruction

- 748 clock cycle long pipeline written in Verilog
  - Additional blocks developed: cube root, complex square root, rot. matrix, cross/scalar product,...
  - Lengthy task in Verilog with all test benches (implementation took 2.5 months)
- Pipeline running with 200MHz → 5ns per photon
- FPGA resources:

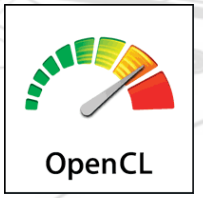
FPGA Resource Type	FPGA Resources used [%]	For Interface used [%]
ALMs	88	30
DSPs	67	0
Registers	48	5

Intel® Xeon/FPGA Results



- Acceleration of factor up to 35 with Intel® Xeon/FPGA
- Theoretical limit of photon pipeline: a factor 64 with respect to single Intel® Xeon® thread
- Bottleneck: Data transfer bandwidth to FPGA

Compare Verilog - OpenCL



- Development time
  - 2.5 months – 2 weeks — **Faster**
  - 3400 lines Verilog – 250 lines C — **Easier**
- Performance
  - CBRT : x35 – x30
  - RICH : x35 – x26

  Comparable performance
- FPGA resource usage

RICH Kernel | Verilog RTL | OpenCL

FPGA Resource Type	FPGA Resources used [%]	FPGA Resources used [%]
ALMs	88	63
DSPs	67	82
Registers	48	24

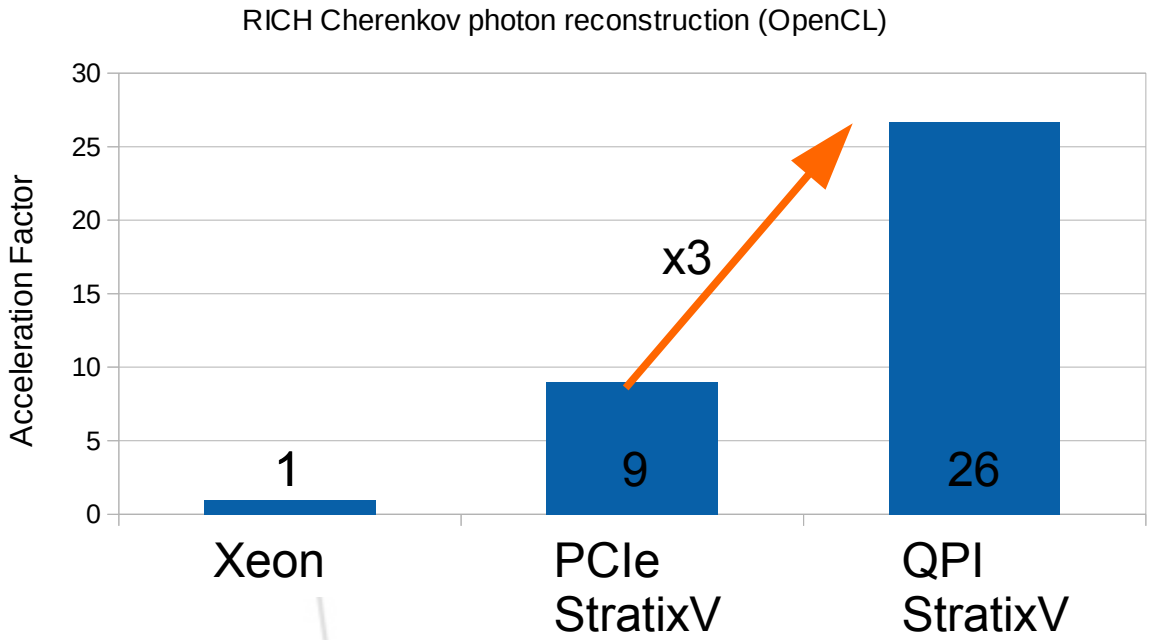
Similar resource usage

Compare PCIe – QPI Interconnect

- Nallatech 385 PCIe vs. Intel® Xeon/FPGA QPI
- Both Stratix V A7 with 256 DSPs
- Programming model: OpenCL
- Reconstruct 1'000'000 photons

RICH Kernel

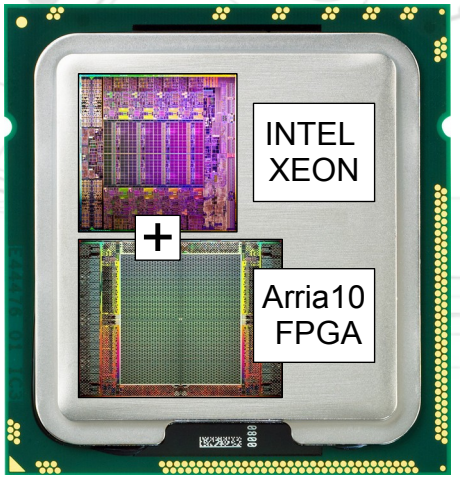
Compare Nallatech 385 and Intel Xeon/FPGA acceleration



Future Tests

- Implement additional LHCb HLT algorithms
  - Tracking, decompressing and re-formatting packed binary data from detector, …
  - Later also ML, Intel is very interested!
- Compare performance with first multichip Intel® Xeon/FPGA system with Arria 10 FPGA
  - Broadwell+Arria10 arrived in our lab now
- Measurements of Arria10 PCIe accelerators
- Compare Verilog vs. OpenCL
- Power measurements
  - Compare with GPUs, ...

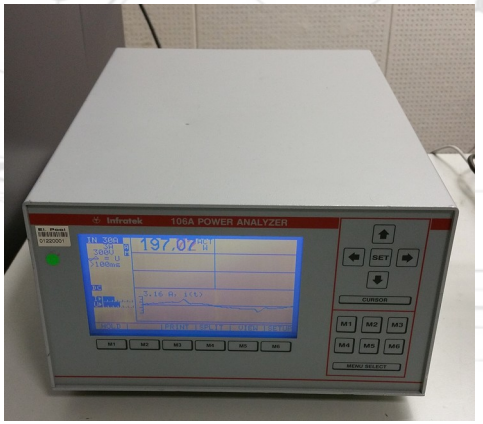
New Intel® Xeon/FPGA with Arria10 FPGA

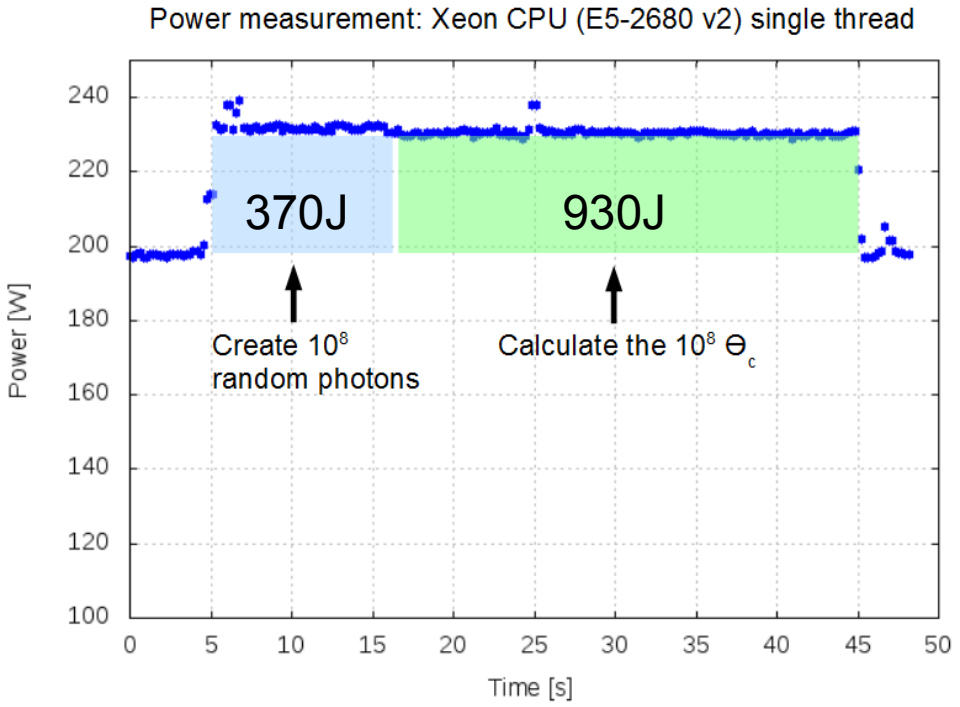


- Multichip package including:
  - Intel® Xeon® E5-2600 v4
  - Intel® Arria10 GX 1150 FPGA
    - 427'200 ALMs, 1'708'800 Registers, 1'518 DSPs
- Hardened floating point add/mult blocks!
- Host Interface: Bandwidth 5x higher than Stratix V version
- Memory: Cache-coherent access to main memory
- Programming model: Verilog soon also OpenCL

Power Measurements

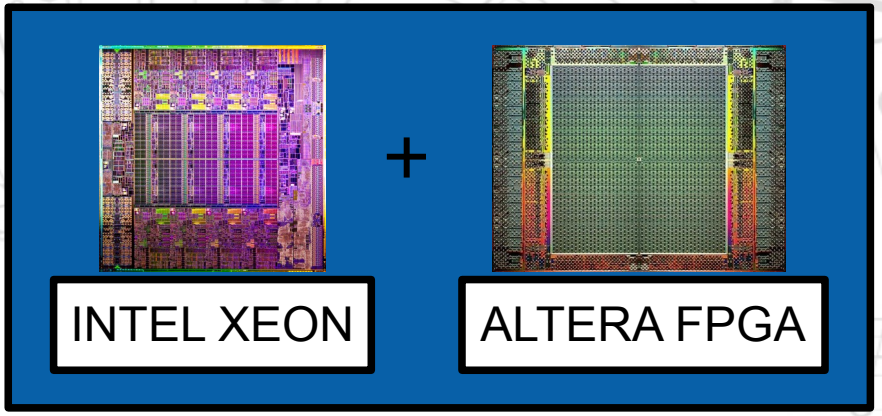
- Just started with power meter
- Run and analyse scripts are finished
- We have measurement permission for Intel® Xeon/FPGA Skylake + Arria10
  - We will get this 2017 – Q1
- Also Xeon CPUs, GPUs and other PCIe FPGA accelerators will follow
- Interesting metric is: photons/(s*J)





Summary

- Results are very encouraging to use FPGA acceleration in the HEP field
- Intel® Xeon/FPGA accelerator performs better than the Nallatech PCIe board using the same FPGA



- Programming model with OpenCL very attractive and mandatory for HEP field
- Also other experiments want to test the usage of the Intel® Xeon/FPGA with Arria10!
- High bandwidth interconnect and modern Arria10 FPGA lets expect high performance and performance per Joule for HEP algorithms! Don't forget Stratix10!

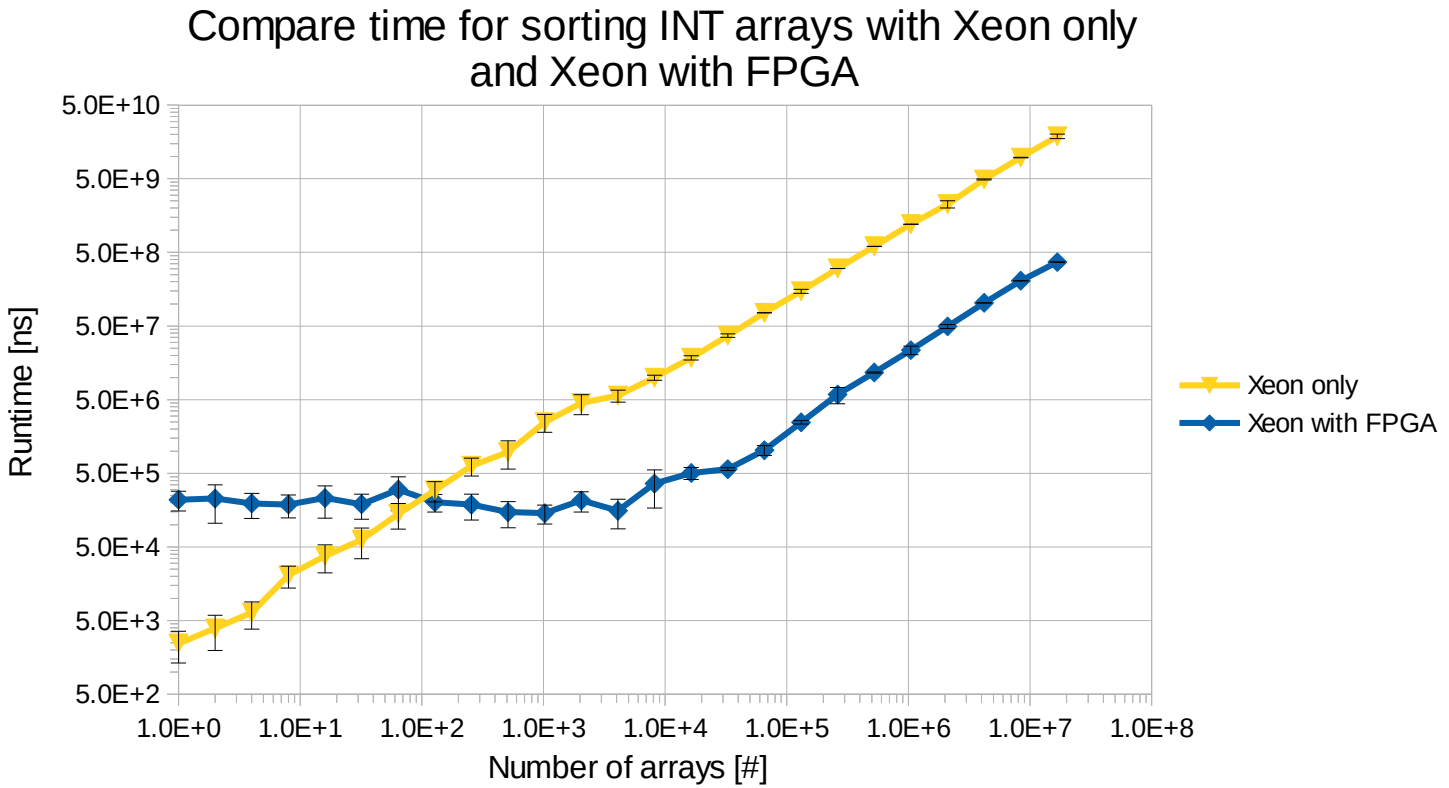
Thank you

Christian Färber,
IML Machine Learning Working Group Meeting – 23.11.2016

Backup

First results with Xeon/FPGA I

- Sorting of INT arrays with 32 elements
  - Implemented pipeline with 32 array stages
  - FPGA sort is x50 faster than single Xeon thread



First results with Xeon/FPGA II

- Mandelbrot with floating point precision
  - Implemented 22 fpMandel pipelines running at 200MHz, each handles 16 pixels in parallel (total: 352 pixels).
  - FPGA is x12 faster as Xeon running 20 threads in parallel.
  - Used 72/256 DSPs
  - Reuse of data on FPGA high