A coprocessor for the Fast Tracker Simulation

C. Gentsos[*,1,2,3], S. Citraro[2,3], G. Volpi[2,3], P. Giannetti[3], S. Nikolaidis[1], K. Kordas[1], F. Crescioli[4], S. Gkaitatzis[1,2]

[1]Aristotle University of Thessaloniki, [2]University of Pisa, [3]INFN-Pisa, [4]LPNHE CNRS

[*]Presenter email: cgentsos@physics.auth.gr

Introduction

The Fast Tracker (FTK)[1], an ATLAS upgrade, executes real time tracking for online event selection exploiting maximum parallelism and pipelining. The ATLAS detector records the hit positions of charged particles traversing its volume. These hits constitute an image (an event), and the job of FTK is to analyse the image and reconstruct the tracks of these particles in real time. Track reconstruction is executed in two steps: the Associative Memory (AM)[2][3] first implements a pattern matching (PM) algorithm by recognizing track candidates at low resolution. Then the Track Fitter (TF), implemented in FPGAs (Field Programmable Gate Arrays), performs Track Fitting in full resolution.

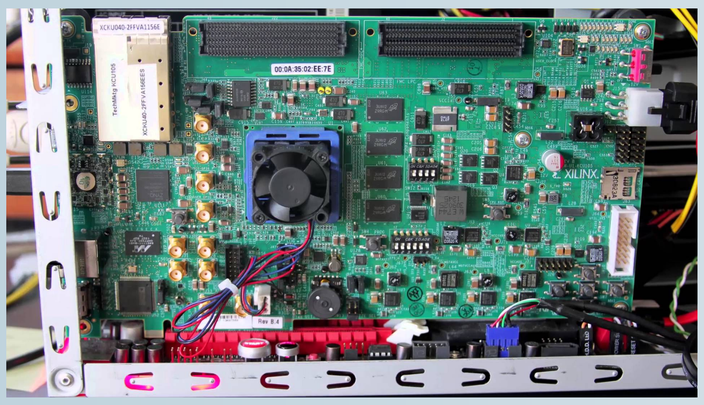
The use of such parallelized architectures for real time event selection opens a new huge computing problem related to the analysis of the acquired samples. For each kind of implemented trigger, millions of events have to be simulated to determine with a small statistical error the efficiency and the bias of that trigger. The AM chip emulation is a particularly complicated and time-consuming task.

The use of a hardware co-processor in place of its simulation can solve the problem. We report about a new compact embedded system for tracking simulation. It is a miniaturization of the complex FTK processing unit, suited for other High Performance Computing applications, as well.

Development Board

Xilinx KCU105 evaluation board

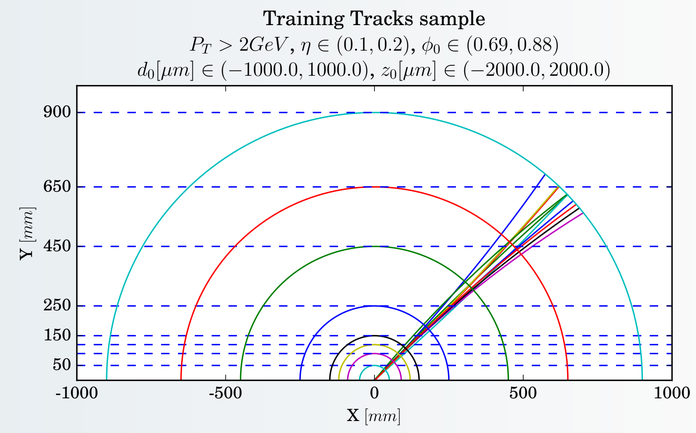
- Kintex-7 Ultrascale FPGA
- 2 FMC connectors for the PM mezzanine
- Ethernet and PCI interfaces for the PC



Xilinx FPGA development board

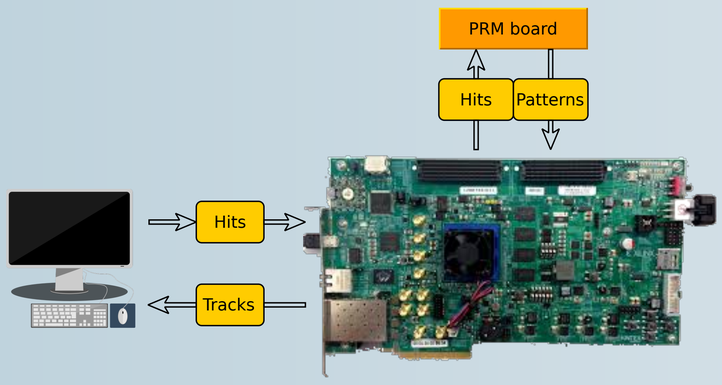
Toy detector model

- Stand-alone Python application
- Implements a "toy" detector model
  - Emulates barrel section of a detector
  - 3 inner layers resemble pixel detectors
  - 5 inner layers resemble strip detectors
- Helical tracks, described by 5 parameters
  - A linear fit[5] is used to compute parameters from full resolution local coordinates
- The application produces input vectors and hardware testing configuration
  - Tracks, layer intersection coordinates (hits)
  - Pattern bank for pattern matching
  - Linear fit constants for track fitting
- Produces plots from the hardware results



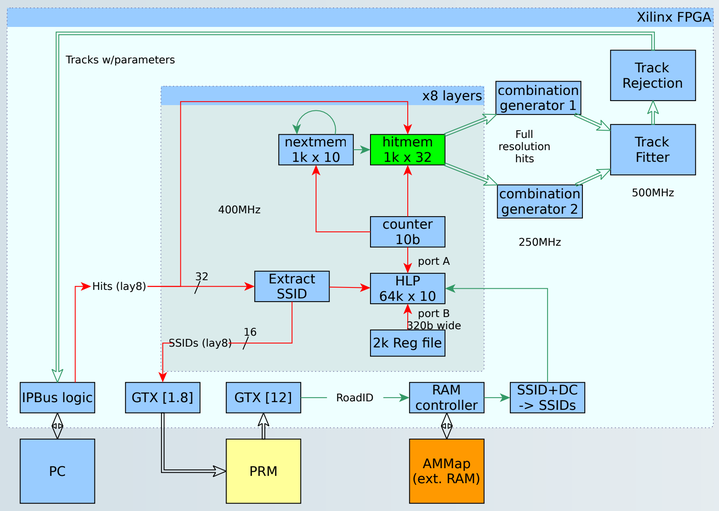
Cross-sectional view of cylindrical "Toy" detector layers, 10 random tracks confined in a very narrow region are shown (see plot title for exact ranges). The region shown is used for the tracks in the "vs Occupancy" plots, shown below.

Data flow



1. The PC sends hits to the FPGA board
2. The FPGA computes SuperStrips (SS) used for PM and stores the corresponding hits
3. SS data are sent to the PM mezzanine
4. Pattern matching results (roads) get read back from the PM, into the FPGA
5. A RAM module used as a look-up-table decodes each road to a list of SS
6. Hits associated with the SS list are retrieved
7. From the hits of each road, all possible track-forming combinations are generated
8. A linear fit is computed for each combination, using the appropriate fit constants
9. Combinations get approved or rejected according to the χ^2 value of their fit
10. Good combinations (tracks) are sent back to the PC

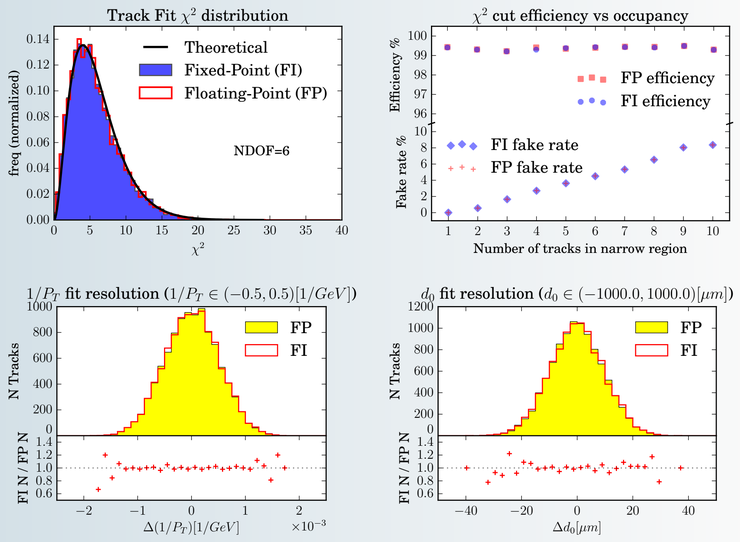
Firmware description and features



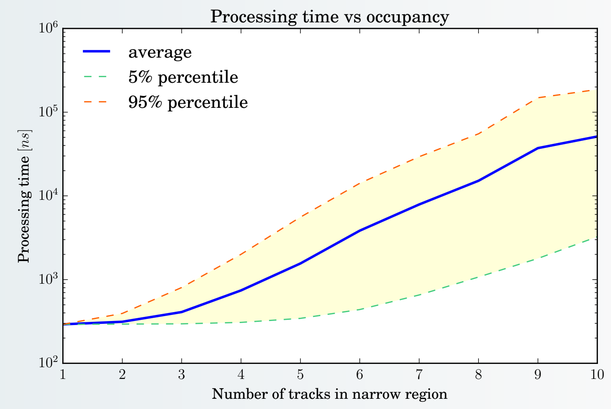
Simplified FPGA firmware block diagram

- IPBus handles Ethernet communications
- Data Organizer: Central component[4]
  - Up to 8 output ports in parallel
  - Can ignore up to the last 3 bits of SS
  - Fully supports patterns with missing layers
- Combination Generator
  - Handles one combination / clock cycle
  - 2 units feed 1 Track Fitter
- Track Fitter
  - Handles one combination / clock cycle
  - 4 units can fit to give 2GFits/s performance

Fixed-point HW fit resolution



Processing performance



Processing time vs occupancy, expressed by the # of tracks/event in a narrow region (see tracks plot, above); to generate each data point many events are processed, until the target of 10000 tracks per point is reached

References

[1] ATLAS Collaboration, "Fast tracker (ftk) technical design report," CERN, Geneva, Tech. Rep. CERN-LHCC-2013-007. ATLAS-TDR-021, 2013. [Online]. Available: {http://cdsweb.cern.ch/record/1552953}.

[2] A. Annovi, A. Baschirotto, M. M. Beretta, *et al.*, "A xor-based associative memory block in 28 nm cmos for interdisciplinary applications," in *2015 IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, 2015, pp. 392–395. DOI: 10.1109/ICECS.2015.7440331.

[3] A. Annovi, G. Broccolo, A. Ciocci, *et al.*, "Associative memory for l1 track triggering in lhc environment," *Nuclear Science, IEEE Transactions on*, vol. 60, no. 5, pp. 3627–3632, 2013, ISSN: 0018-9499. DOI: 10.1109/TNS.2013.2281268.

[4] C. Gentsos, F. Crescioli, P. Giannetti, D. Magalotti, and S. Nikolaidis, "Future evolution of the fast tracker (ftk) processing unit," in *Proceedings, 3rd International Conference on Technology and Instrumentation in Particle Physics (TIPP 2014)*, 2014.

[5] S. Amerio, A. Annovi, M. Bettini, *et al.*, "The gigafitter: a next generation track fitter to enhance online tracking performances at cdf," in *Nuclear Science Symposium Conference Record (NSS/MIC), 2009 IEEE*, R. C. Lanza, Ed., IEEE, New York, USA: IEEE, 2009, pp. 1143–1146, ISBN: 9781424439614. DOI: 10.1109/NSSMIC.2009.5402410. [Online]. Available: http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=5384532.

Acknowledgments

This work receives support from Istituto Nazionale di Fisica Nucleare; Grant-in-Aid for Scientific Research from the Japan Society for the Promotion of Science and MEXT, Japan; and the European community FP7 People grants ARTLHCFE 254410 FP7-PEOPLE-2009-IOF and FTK 324318 FP7-PEOPLE-2012-IAPP.