A novel 4D fast track finding system using precise space and time information of the hit

paolo.gandini@cern.ch

Marco Petruzzo, **Paolo Gandini**, Nicola Neri,
Mauro Citterio, Stefano Riboldi

Abstract

A novel fast track-finding system capable of reconstructing 4D particle trajectories in real time using precise space and time information of the hits. This fast track-finding device is based on a massively parallel algorithm implemented in commercial FPGA using a pipelined architecture.
The working prototype represents the proof of principle of a FPGA tracker for LHCb Upgrade-II

The High Lumi Upgrade: Phase II

- x10 nominal LHC luminosity
- High pileup: 50 – 100 interactions x crossing
- LHCb VELO pixel detector as case study
- Simple case: 12 layer forward telescope
- Luminosity: 1 x 10^{34} $cm^{-2}s^{-1}$
- Pileup ~ 40 and ~1200 tracks/event

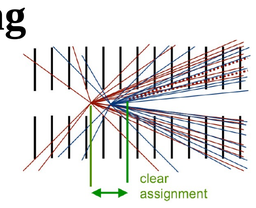
Sensor area = 6x6cm²
pixel size = 55x55µm²
thickness = 200 µm
time res σ_t=30 ps

The Timespot project

- Rad-hard pixel detector with precise 4D tracking
- R&D started and funded by INFN
- 3D pixels: goal 30ps time and 40µm resolution
- Goal is to produce a fully working prototype

The role of timing

Timing in PV-association
- Assuming already good track quality
- Hit time resolution needed: ~200ps
- NOT enough to separate hits in the pattern reco



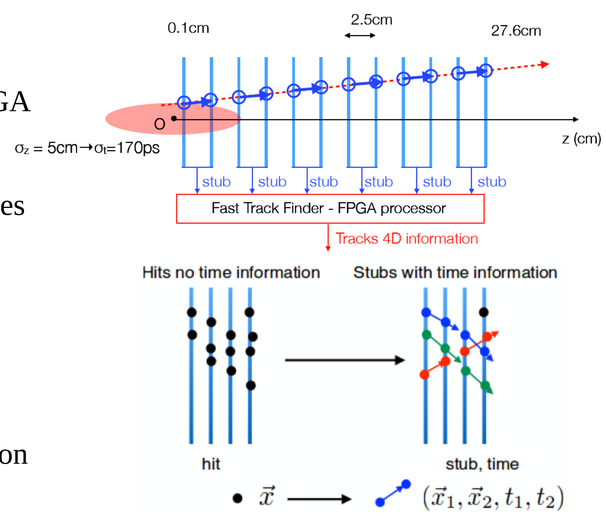
Timing in tracking
- Use only time compatible hits in the pattern reco
- Stricter hit time resolution requirement: ~30ps
- Ghost rate reduction

4D stub tracking algorithm

Idea
- Reconstruct stubs at early stage
- Reduce data flow and simplify track reco
- Provide stubs in input to Fast Track Finder FPGA

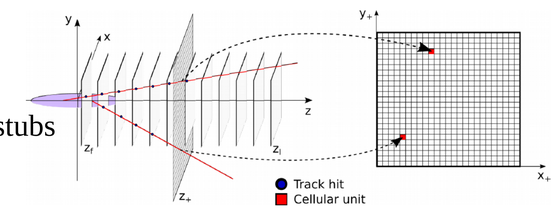
"Stub" approach
- Stub definition: doublet of hits in adjacent planes
- A stub provides a track hint
- Stubs are searched only in compatible regions
- Vetoes to reduce combinatorics

Stubs + Timing
- Fake stubs can survive the geometrical cuts
- Precise timing allows a combinatoric suppression
- Simplify pattern recognition and track finding



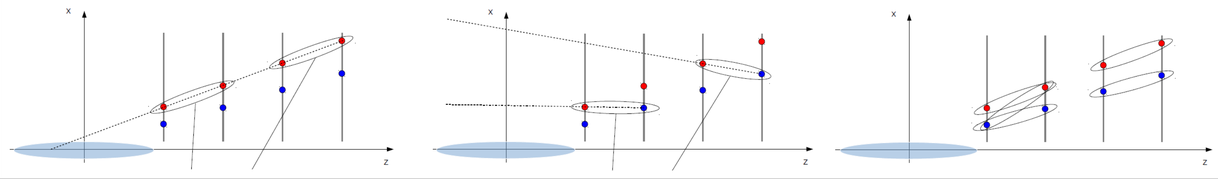
Algorithm details

Basics:
- Stubs identified and filtered with geom+time cuts
- Stubs projected to a reference 2D plane
- Tracks identified from groups of similar projection stubs
- 4+1 parameters: for now use only 2 for pattern reco



- A grid of engines is distributed in a reference plane
- An engine is a module that processes stubs and identifies tracks from groups of similar stubs

- Filtering reduces combinations to be processed
- Require pointing to interaction region
- Time of flight compatible with speed of light



Software simulation

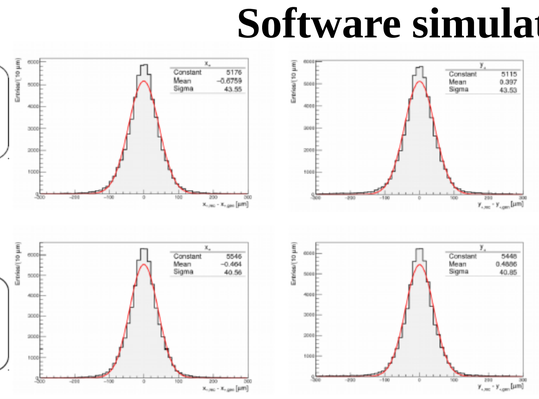
Standalone software simulation
Response simulated with and without timing
- Gaussian luminous region σ(z)=5cm, σ(t)=167ps
- **90000 engines**
- Uniform distribution in the (x+,y+) space
- Hardware implementation uses (r,φ)

Track parameters resolution improves with *t*
The reconstruction efficiency is stable
The track purity improves

$\sigma_{x\text{-}/y\text{-}}$ = 95.0µm
$\sigma_{x+/y+}$ = 43.5µm

Time

$\sigma_{x\text{-}/y\text{-}}$ = 70.1µm
$\sigma_{x+/y+}$ = 40.6µm
σ_t = 14.3ps

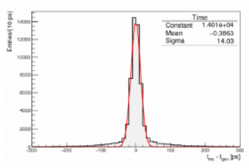


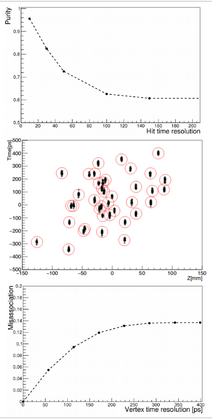
First row: resolution on track parameters (x+,y+) **without** using the time information of the stubs
Second row: resolution on track parameters (x+,y+, t) **using the time information** of the stubs

Efficiency = 99%
Purity = 64%

Time

Efficiency= 99%
Purity = 85%





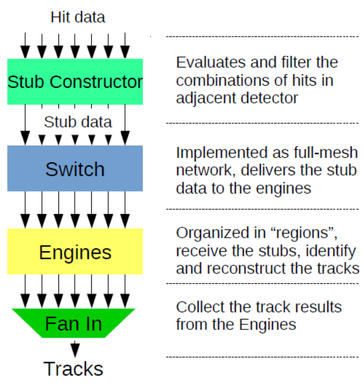
The track purity improves with time resolution

The track timing allows to "clusterize" the tracks in a 2D (z,t) space

The PV misassociation reduces with improved timeresolution

Hardware architecture

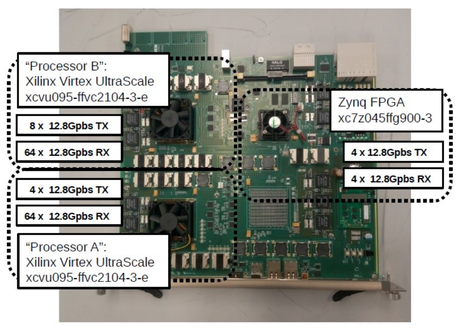
The system receives the hits
Output: tracks (for further processing)
Key points:
- Highly parallelized architecture
- Pipelined architecture
- Hold logic implementation for data flow management for minimum latency and reduced data serialization
- Low latency: <1µs total



- Each plane doublet → one stub-maker
- Stub constructor implemented in dedicated FPGA or on front-end electronics (FPGA as well)
- Stubs are filtered before sending to switch
- Modular system parallel processing

Hardware prototype

- FPGA implementation on a custom board
- Two Xilinx Virtex Ultrascale FPGAs
- High-speed optical transceivers ~1.6 Tbps input
- One Xilinx Zynq FPGA



- Working prototype in Milan
- Bit Error Ratio tests successful, GTY/GTH links speed 12.8 Gbps
- 32b/40b data communication between FPGAs

Results

- Switch and engines fully implemented in FPGA

- On a single FPGA: Switch with 64 input, 1024 engines and Fan-In
- This covers 50% of chip resources
- Current board has 2 FPGA Virtex UltraScale
- We can use simulation to estimate system needs
- 64*1024 engines to cover the whole parameter region (2x)
- In this scenario: 1board → 1/16 of the detector

- Stub-maker in development: optimisation of detector coverage
- Currently each sensor dividend in (16 radial,128 angular) sectors
- Number of stub-makers = 16*128 = 2048
- Each stub-maker processes about 10 combinations

- Switch+Engines tested to cope with 40 MHz
- Input rate to the switch evaluated with simulation to be 117.2 MHz

- **Using a system clock of 400 MHz the track finder proved to process events at rate larger than 40 MHz**

Summary

- The first real-time tracking system based on FPGA was implemented and tested (3D)
- 4D extension ongoing: aim to include time and stub concept in a working prototype
- Stub approach developed: stub constructor hardware implementation ongoing
- FPGA implementation and optimisation on commercial hardware with promising results

References

- PIXEL2016: Talk - 4D fast tracking for experiments at HL-LHC
- ANNECY Workshop on LHCb Upgrade II: Talk on Time gains
- N. Neri et al., JINST 11 (2016) no.11, C11040
- N. Neri et al., POS(TIPP2014)19

TWEPP2018: Antwerp, 17th - 21st September 2018