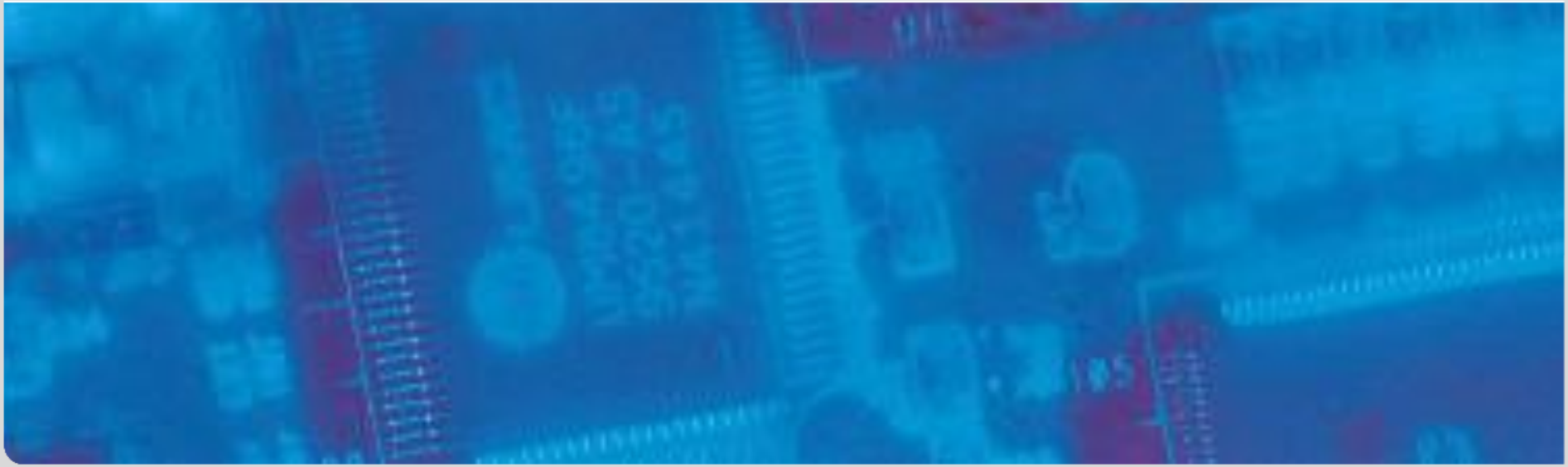
Status of the Neurotrigger

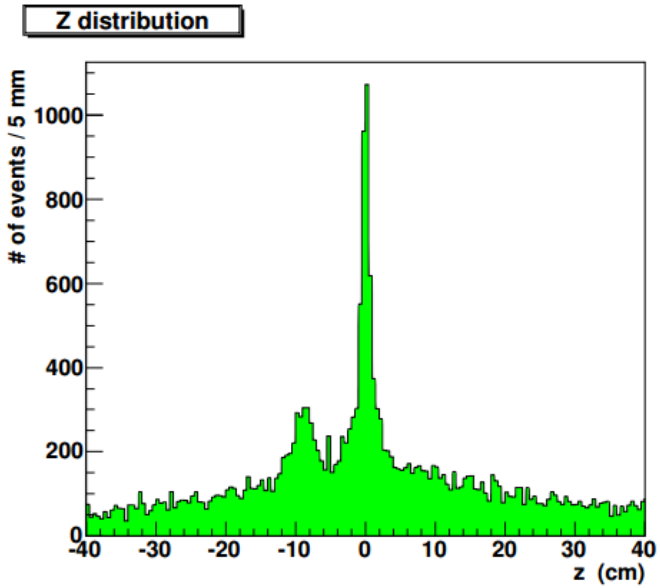
Steffen Bähr, Sara Neuhaus, Sebastian Skambraks, Christian Kiesling

Institute for Information Processing Technologies (ITIV)



Introduction

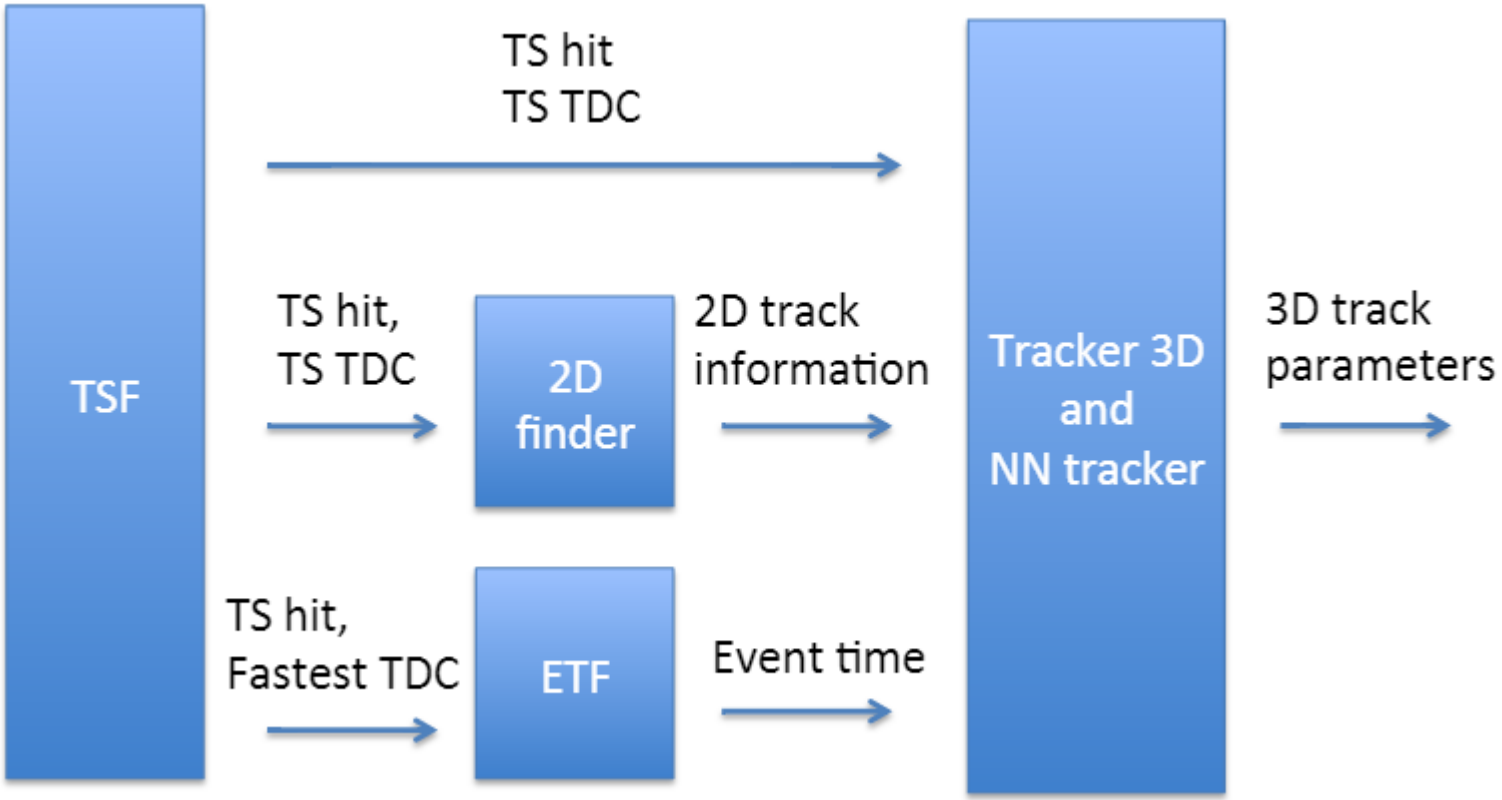
- The Neurotrigger
  - A Z-Vertex track trigger for Belle II deployed on FPGAs
  - Goal : Rejection of events from z != 0
- Requirements
  - Z-Vertex prediction < 2 cm
  - Stay within demanded latency
  - Manage resource consumption on FPGA
- Methods
  - CDC track segment data as input
  - Multi Layer Perceptron for prediction of z-Vertex



Offline Distribution of z in Belle

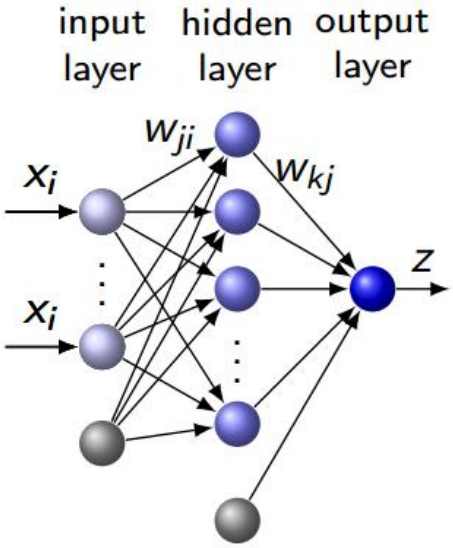
Data Flow

- Thank you Jae-Bak



MLP – Multi Layer Perceptron

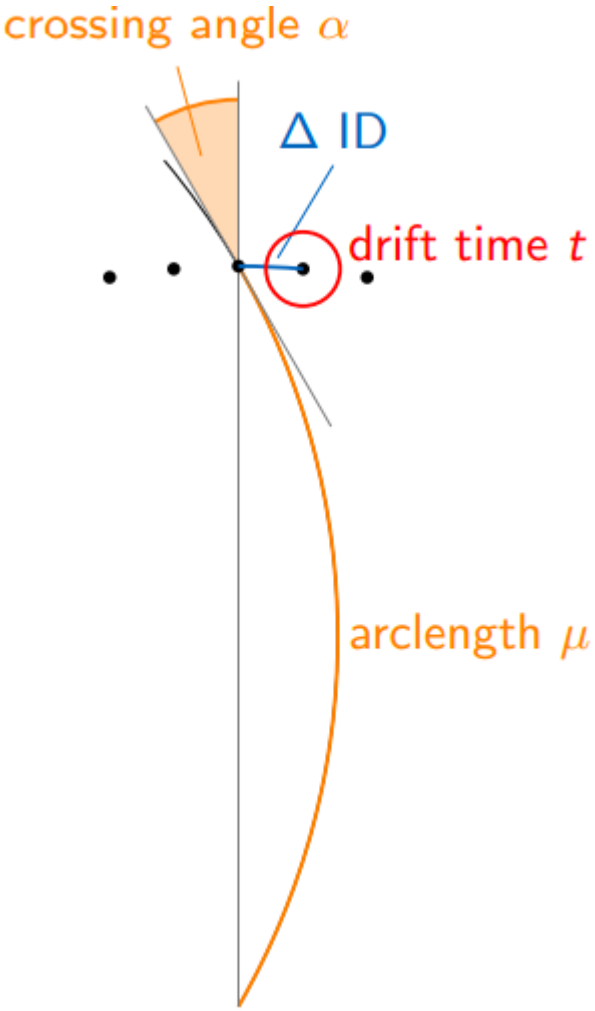
- General
  - Supervised machine learning
  - Approximation of real valued function
  - Deterministic runtime



Structure of general MLP

- Setup
  - Consists of several neurons grouped into layers
  - Neuron description : tanh($\sum_{k=0}^{n} x_k w_k + w_0$)
  - 3 layers : input, hidden and output
  - Output value interpreted as scaled z-vertex position
  - Trained with rprop algorithm

Inputs for the MLP



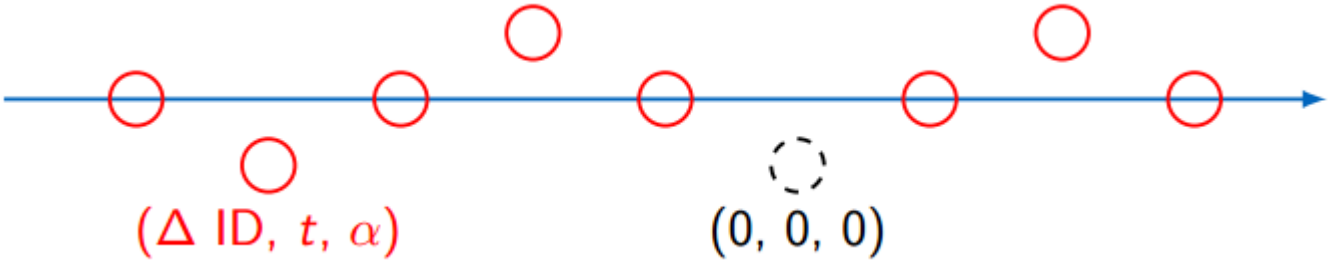
- For each SuperLayer 1 hit is chosen
- Each Hit has 3 Inputs in the MLP
  - Delta ID : Distance wire – track
  - T : Drift Time
  - Alpha : crossing angle
- Calculation of these inputs necessary before executing the MLP

Some Challenges

- How to handle missing hits from TSF ?
  - Not every SuperLayer may contain a hit
- How to translate real valued MLP and necessary preprocessing to FPGAs ?
  - Usage of fixpoint calculation on the FPGA
  - Prediction quality cannot be decreased too much
  - Resources and latency to be kept in mind

Handling of Missing Hits

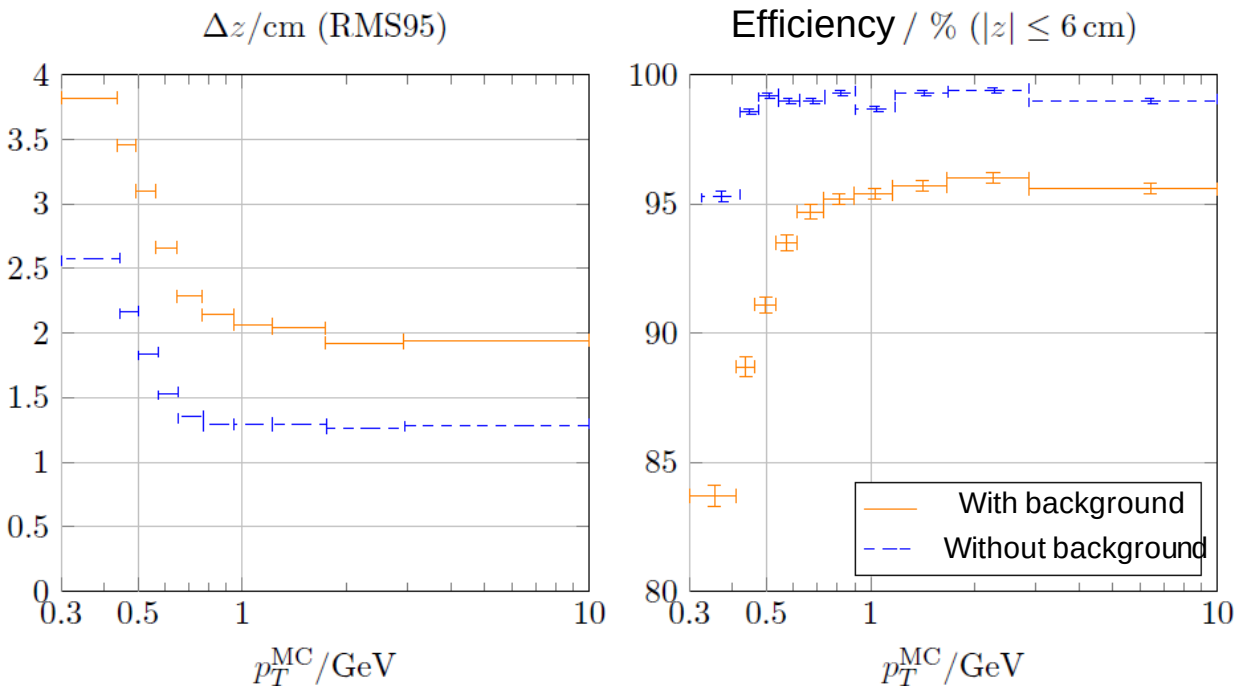
- MLP expects hit inputs for all SuperLayers
  - What to do in case a hit is missing for a SuperLayer ?



- Default would be to set inputs for the respective SuperLayer to 0
- Usage of a specialized MLP, trained for having 8 Inputs is better
  - 5 Networks used : 1 for all hits present, 4 each for one of the stereo SLs missing a hit

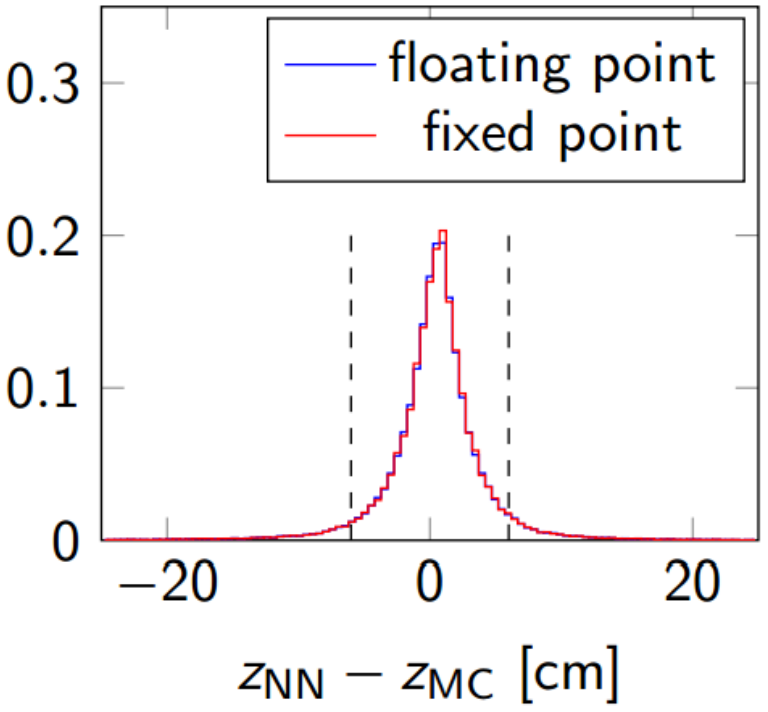
Minimal Setup Achievements

- Minimal setup : consists of 5 MLPs due to missing hits
- Resolution of 2 cm can be achieved for pt > 500 MeV
  - Worse efficiency with background but still sufficient



Fixpoint Calculation Analysis

- Fixpoint values to be used throughout the FPGA
  - Width has to be as narrow as possible to allow for good routing
  - Preprocessing and MLP

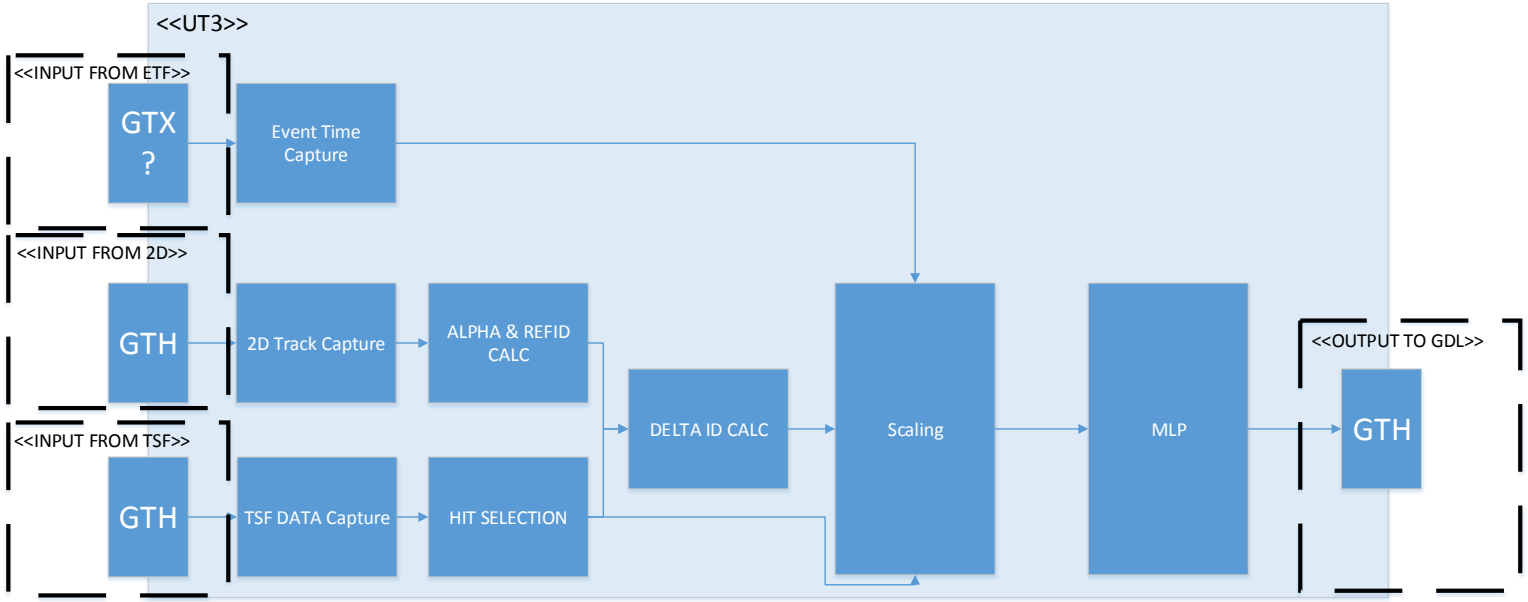


IP tracks with background, 5 MLPs for different missing stereo hits

	φ	ω	α	ID_{ref}	$\frac{N_{wires}}{2\pi}$	nodes	weights	tanh
fractional bits	12	14	12	8	8	12	10	10
maximum	π	0.015	$\frac{\pi}{2}$	288	61.1	1	$\lesssim 32$	4.85
total bits	15	9	14	18	14	13	16	10^4 bins

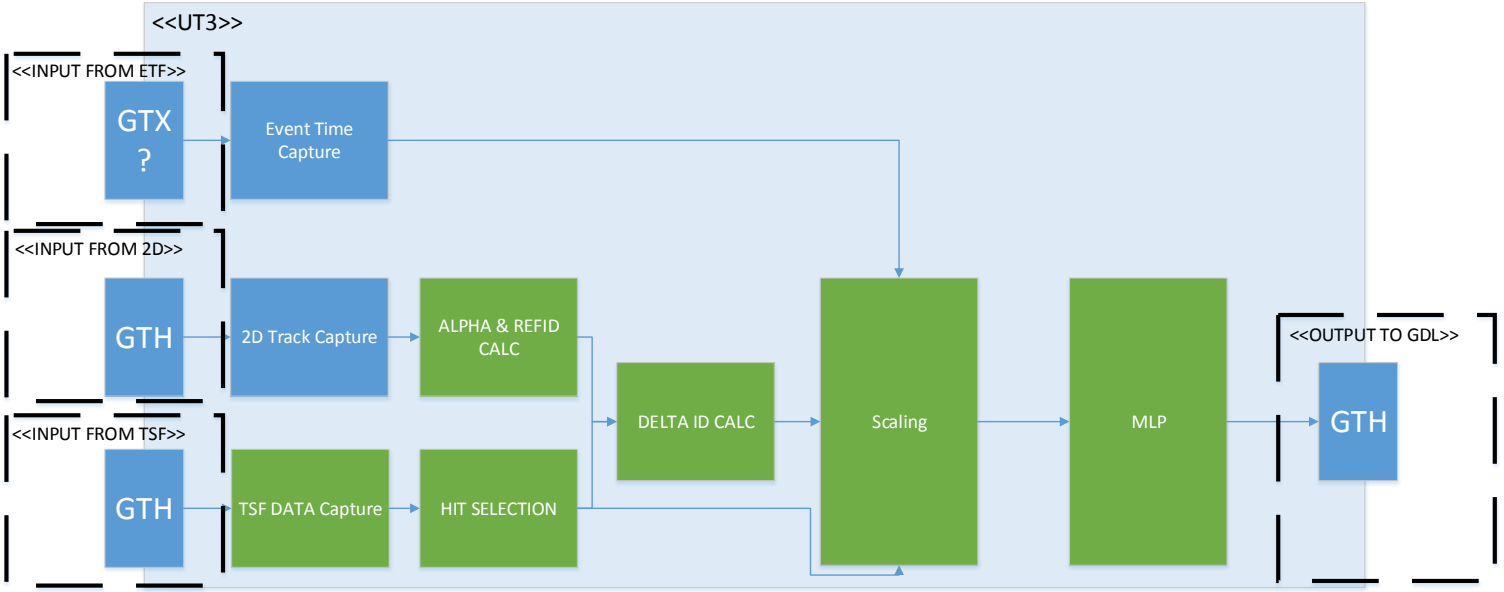
Architecture of Neurotrigger on the FPGA

- Pipelined architecture for processing of data
- Inputs to be calculated using TSF Hits and 2D Data
  - Scaling and Calculation Modules
  - Hit Selection, rule-based selection of one Hit per SuperLayer
- MLP calculates the prediction for Z-Vertex



Implementation Status

- All stages of the Neurotrigger processing are implemented
  - Simulation shows no deviation from SW precision
- 2D-Data and ETF Data taking not finished
  - Currently assuming fixed Input / fastest time (see Cosmic Ray test setup)



Implementation Characteristics

- Latency
  - MLP currently takes 10 Clock Cycles
  - Preprocessing depends :
    - TSF Hit Processing takes around 5 Clock Cycles
    - 2D Track Processing takes around 10 Clock Cycles
  - Total Latency right now at 20 Clock Cycles

- Resources (xc6vhx380)
  - DSPs pretty much used up completely : 93 % Usage
  - BRAM : 57 % Usage
  - SLICES : 48 % Usage

Implementation Tool Usage Considerations

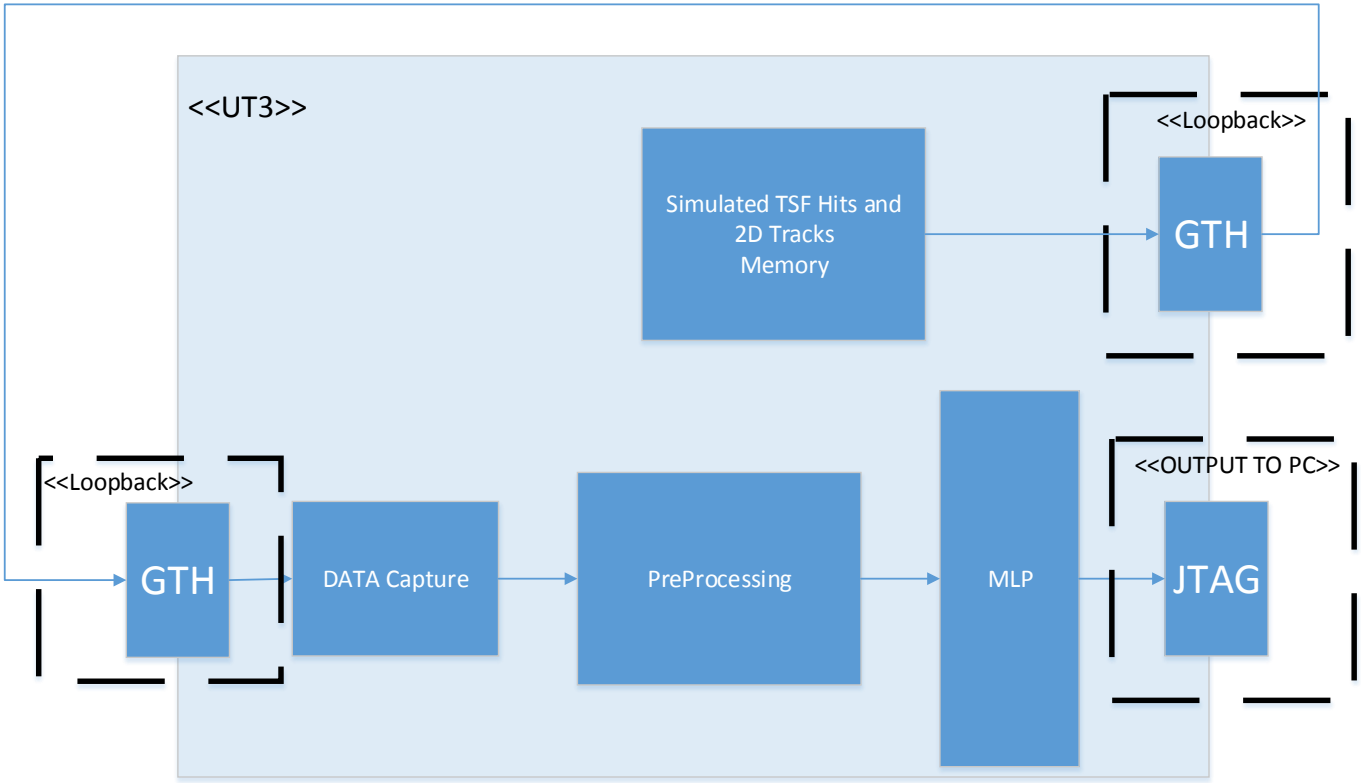
- The achievable clock frequencies heavily depend on the used tool

- Comparing implementation using Synopsys and Xilinx XST (default in ISE)
  - Synopsys Clock Frequency for MLP at : 205.4 MHz
  - ISE Clock Frequency for MLP at : 127.28 MHz

- Synopsys Tools seem to implement pipelined adder trees much better than XILINX's XST

Currently Tested Setup of the Neurotrigger

- So far all processing stages of Neurotrigger tested in simulation and by using a loopback test
  - No problems with size or latency encountered

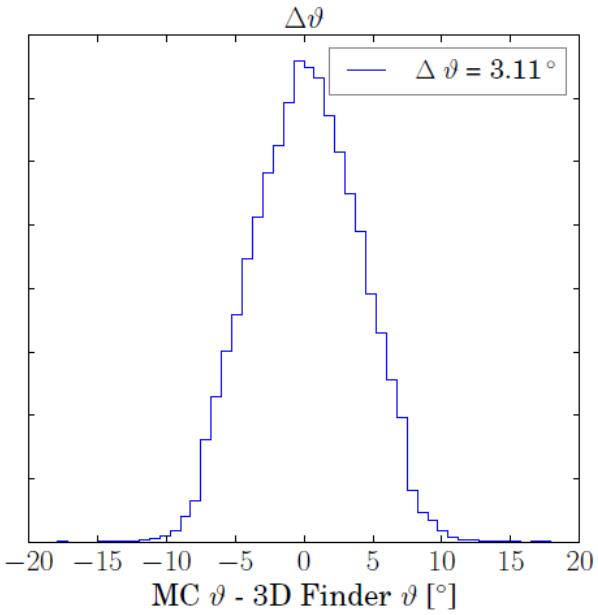


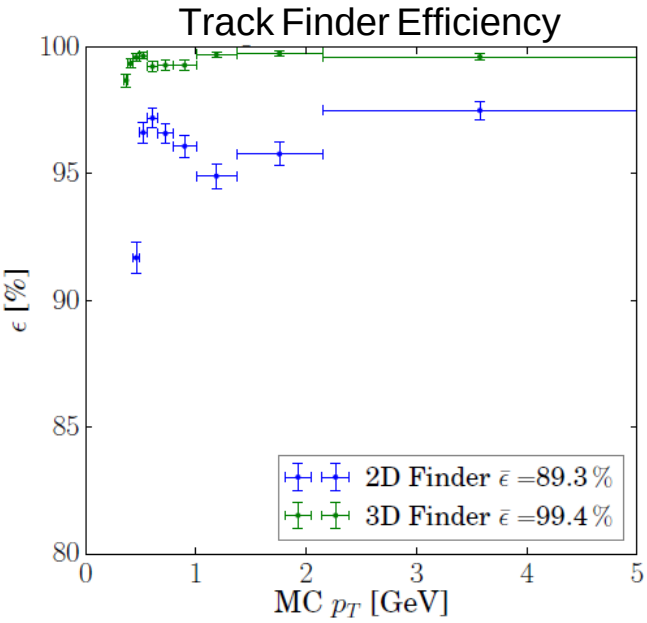
Cosmic Ray Test Setup

- Idea : Use data from test to confirm correct processing
- Assume 2D finder and event time are not present
  - Default tracks are used to generate Inputs for the MLP
  - Fastest drift time (no background) is used
- TSF Data from all SuperLayers necessary, since no 2D Data available

Outlook 3D-Finder

- Usage of 3D Track Finder as a preprocessing stage of the neural network
  - Additionally uses information from stereo layers
- Based on 3D Hough transformation
  - Estimating memory consumption only, implementation could be possible





Open Questions

- Testing State
  - Extend Loopback Test to actual data from TSF
  - Is data from CDC available ? Captured by Pocket DAQ ?
- Interfacing
- Monitoring of correct Neurotrigger processing
  - How to Monitoring the correct execution of the neural network ?
  - Save data via B2Link and check with SW afterwards ?
- How to implement 3D-Finder on the same board

Conclusion

- Using MLPs on FPGAs to estimate z-vertex
- All stages of the processing implemented and fixpoint calculation matches target
- Currently tested in loopback mode
- Tests with TSF data pending
- Upgrade possibilities using 3D-Finder

Thank You