FPGA-
Pre-/Co-processors

Outline

- Motivation
- Pre-Processors
- Co-Processors
- FPGAs in the ALICE TPC datachain
- Cluster-Finder Pre-Processor
- Summary

Motivation

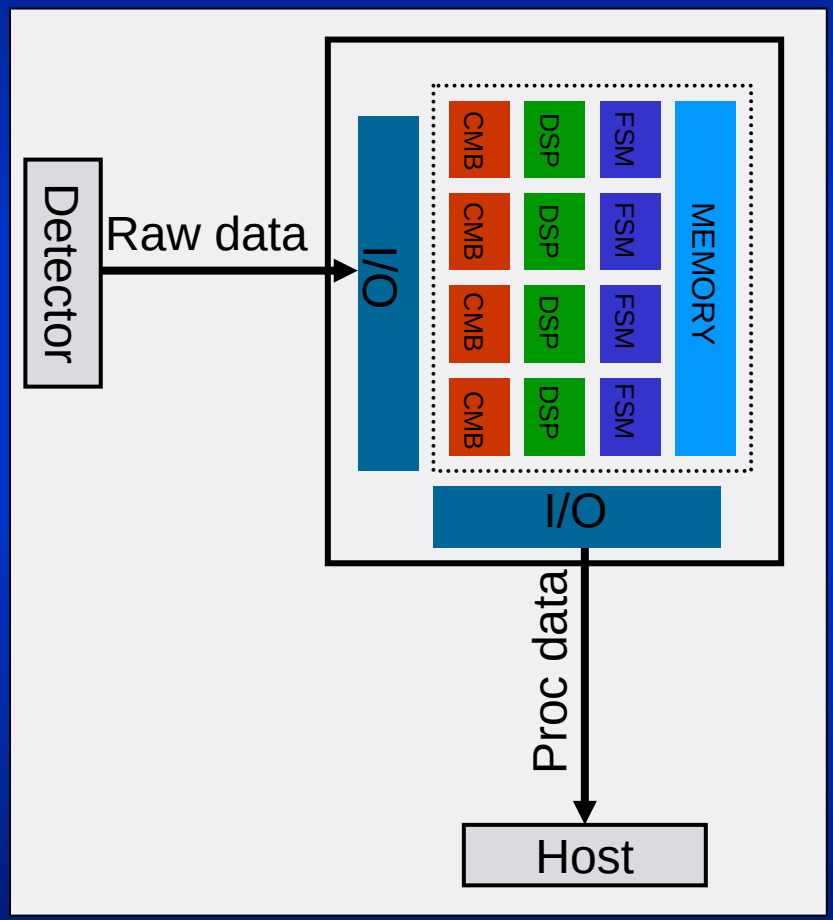
„Why FPGA pre-/coprocessors ?“

- Today‘s FGPAs support various I/O standards : LVTTL, LVDS, PCI, …
  => FPGAs are used to „inject“ raw detector data into the memory of the host

- Futhermore FPGAs provide a broad variety of other logic :
  - embedded DSP
  - fast SRAM
  - programmable PLLs
  - configurable LUT for sequential logic (FSM) and combinatorial logic (ARITHMETIC)
  - fast logic interconnections

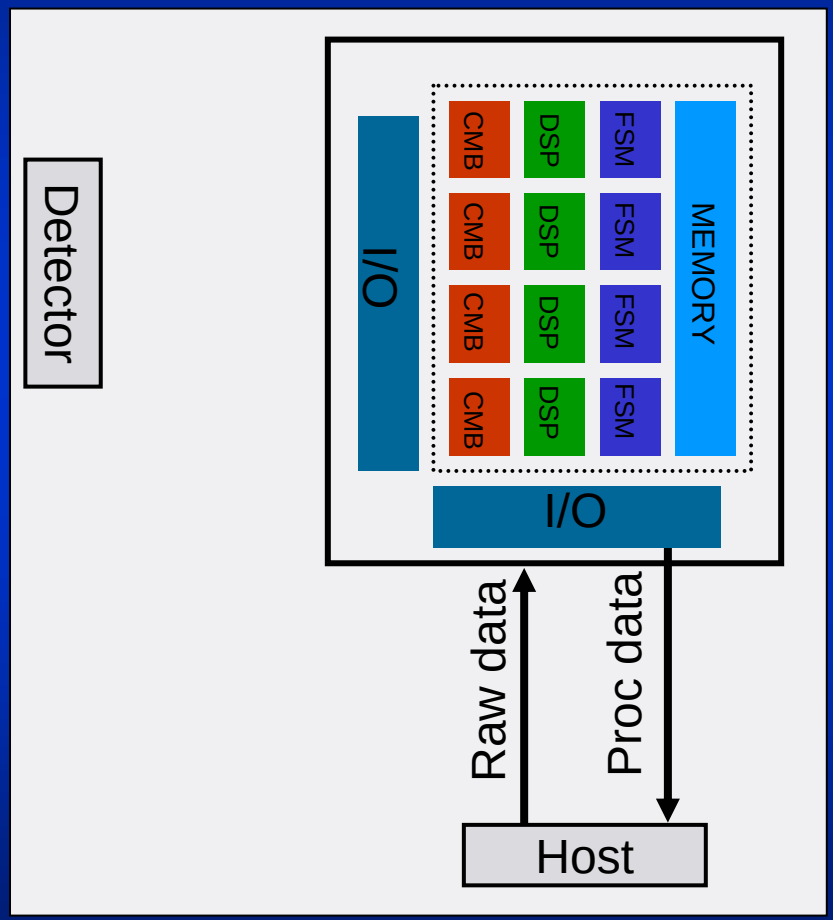
- „Online mode“ : FPGA acts as a pre-processor
- „Offline mode“ : FPGA acts as a co-processor

Pre-processor Mode



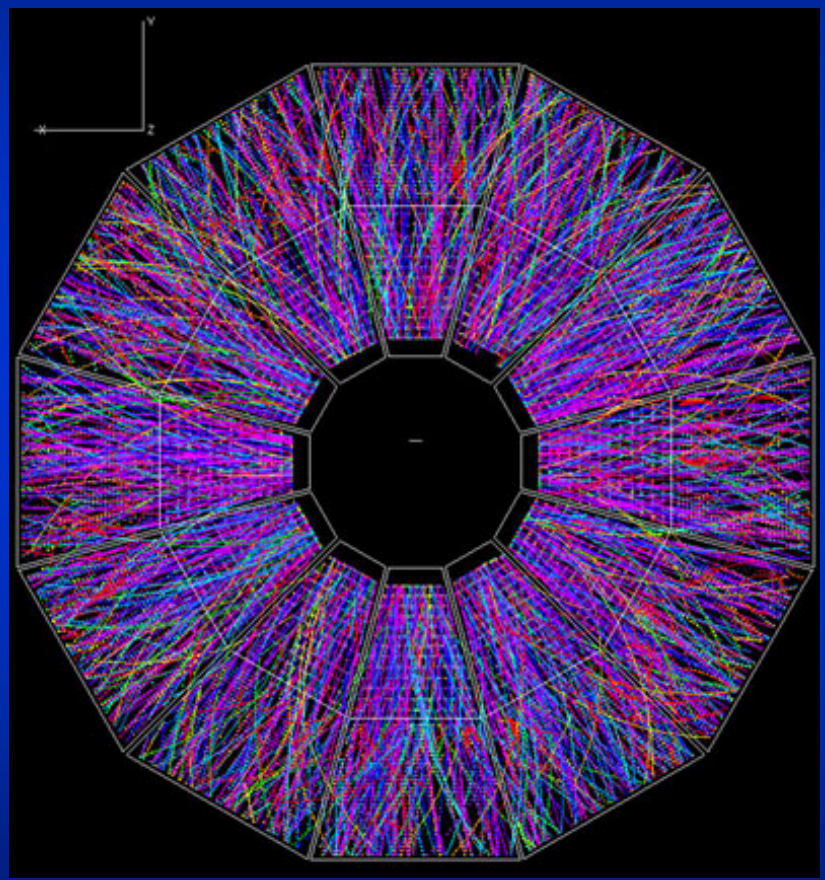
- Data is „injected“ from the detector
- Data is preprocessed:
  - Formater
  - Filter
  - Transformations (i.e.Hough)
  - Pattern recognition
  - Compression
- Logic can be designed for a special task.
- Logic can be massive parallel
- Raw and/or processed data is send to the host

Co-processor Mode



- Data is „injected“ from the host
- Existing resources can be re-used
- FGPA can be reprogrammed with the logic that is needed for a special operation
- Processed data is send back to the host, i.e. over DMA

FPGAs in ALICE – TPC datachain

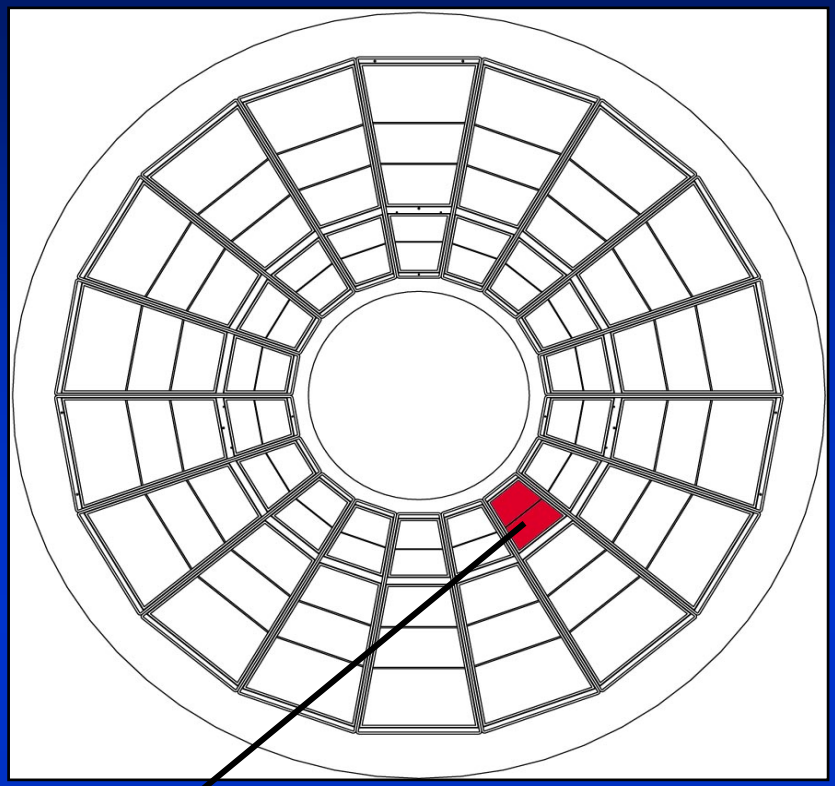


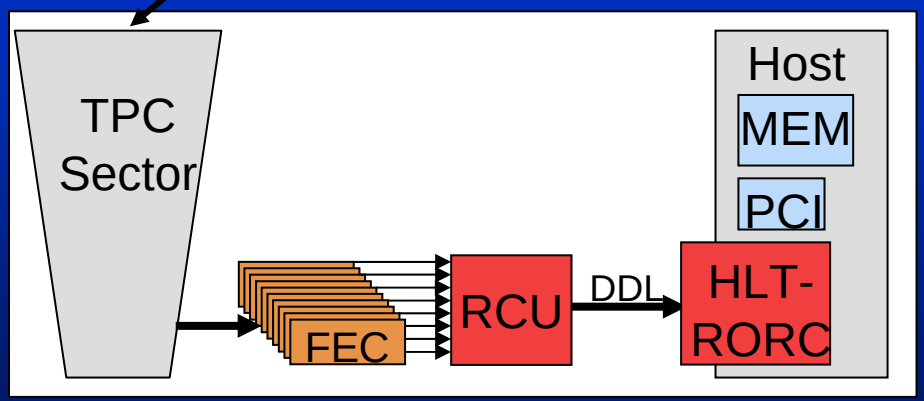
- Future High-Energy-Physics will produce vast amounts of data.
- ALICE TPC :
  - ~ 75 MByte/event
  - ~ 200 Hz eventrate
  - ~ **15 GByte/s**
- 2 major challanges from the computational view:
  1. reduce amount of data
  2. trigger for special events

  => data needs to be pre-processed

but: mind the physics!

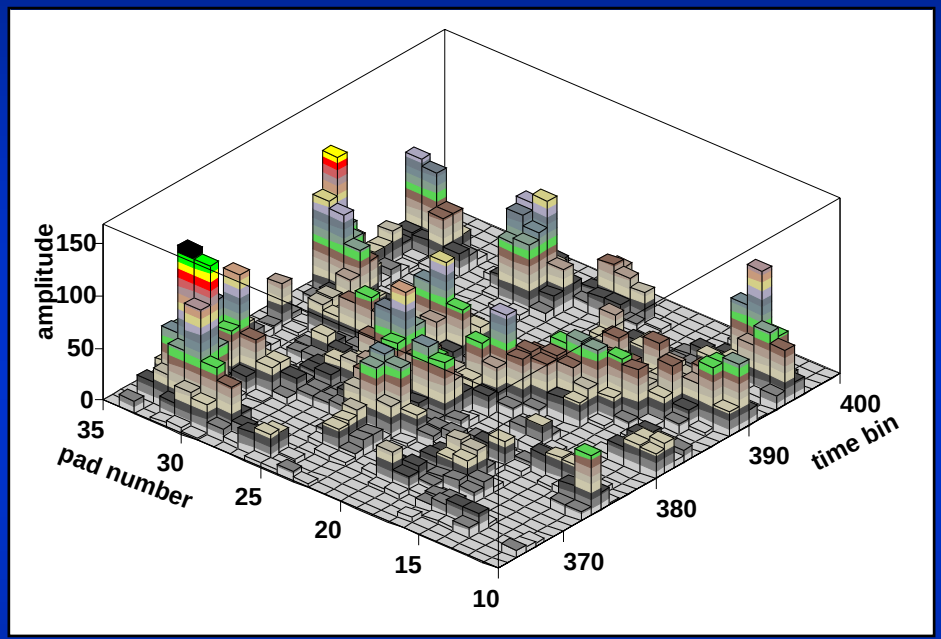
Datapath





- TPC is divided into 216 patches
- Each patch sends it‘s data over a optical fiber to a ReadOutReceiver Card(DAQ-/HLT-RORC)
- HLT-RORC is a custom made PCI card equipped with an optical receiver and a FPGA
- max. datarate coming from the FEEs is 200 MByte/patch

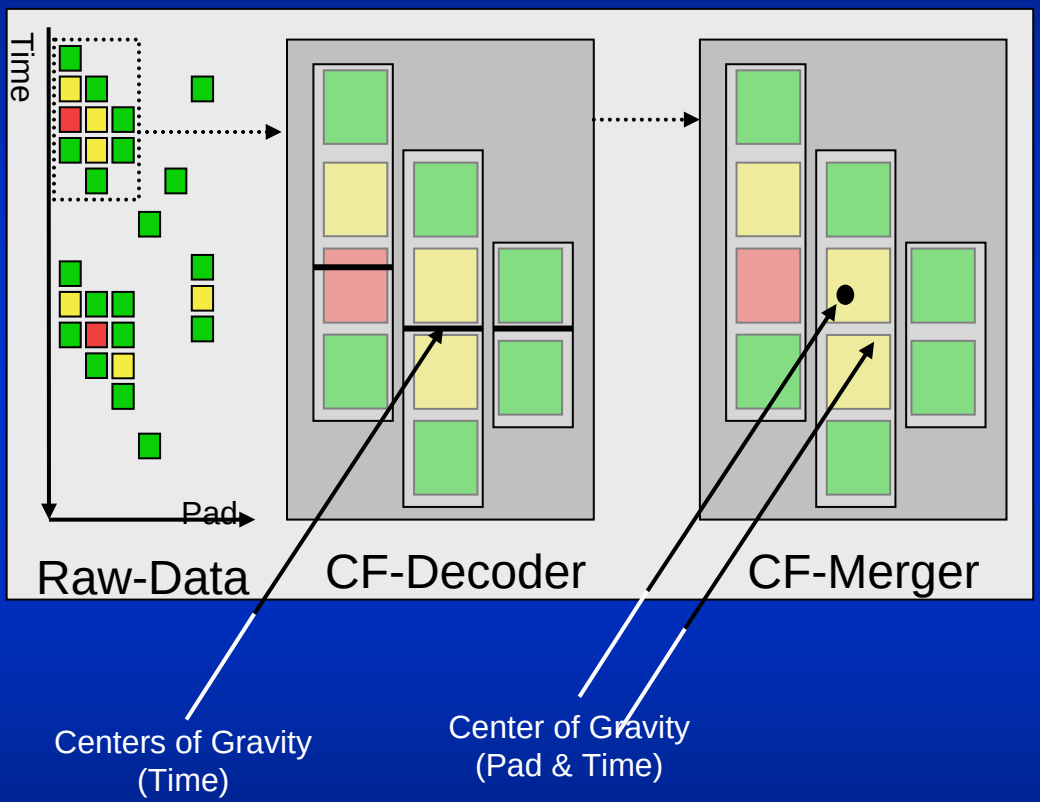
Data and Clusterfinding



Simulated ALTRO-data for one row

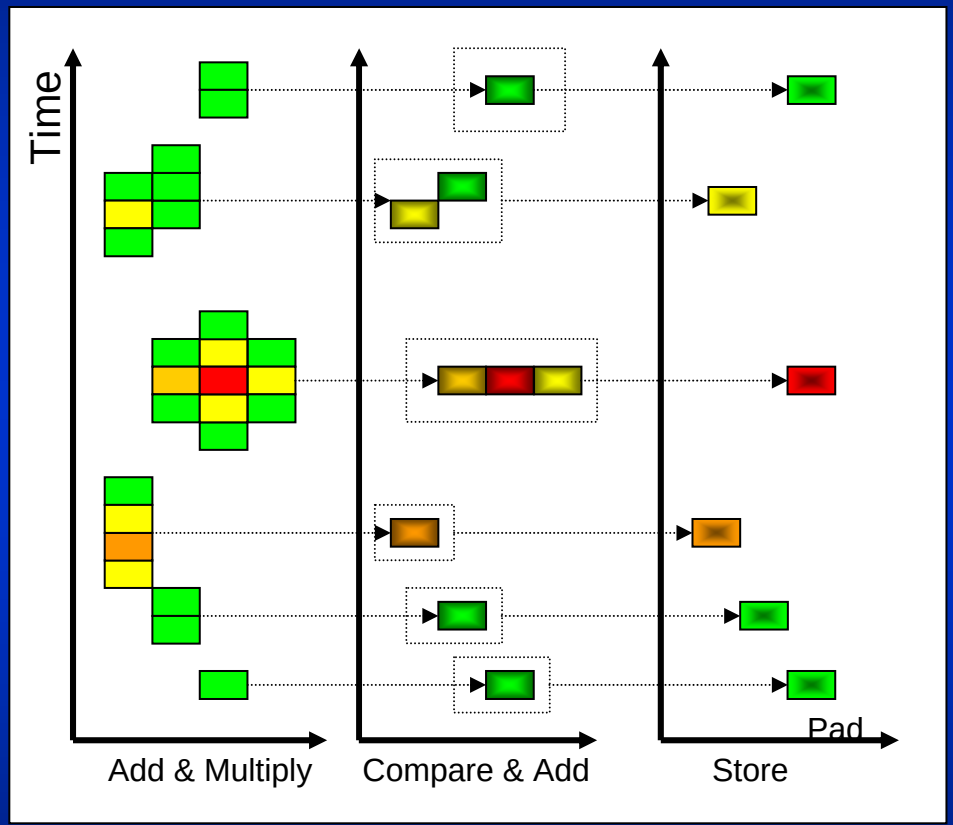
- Each datapoint is represented by 4-dimensional vector
  (row, pad, time, charge)
- Processing can by done for each row independently:
  Clusterfinding is a parallel process
- Clusterfinder operates on
  (pad, time, charge)
- Main operations: add, multiply, compare, store
- Main operations can be done in parallel

CF-Algorithm I



- Data is coming in for each pad in reverse time direction.
- Clusterfinding is done by calculating the center of Gravity in Pad- and Time- direction (STAR-algorithm)
- First the centers of gravity are calculated on each pad in time direction
- Second the adjacent pads are merged if the centers of gravity in time direction are within a defined range.

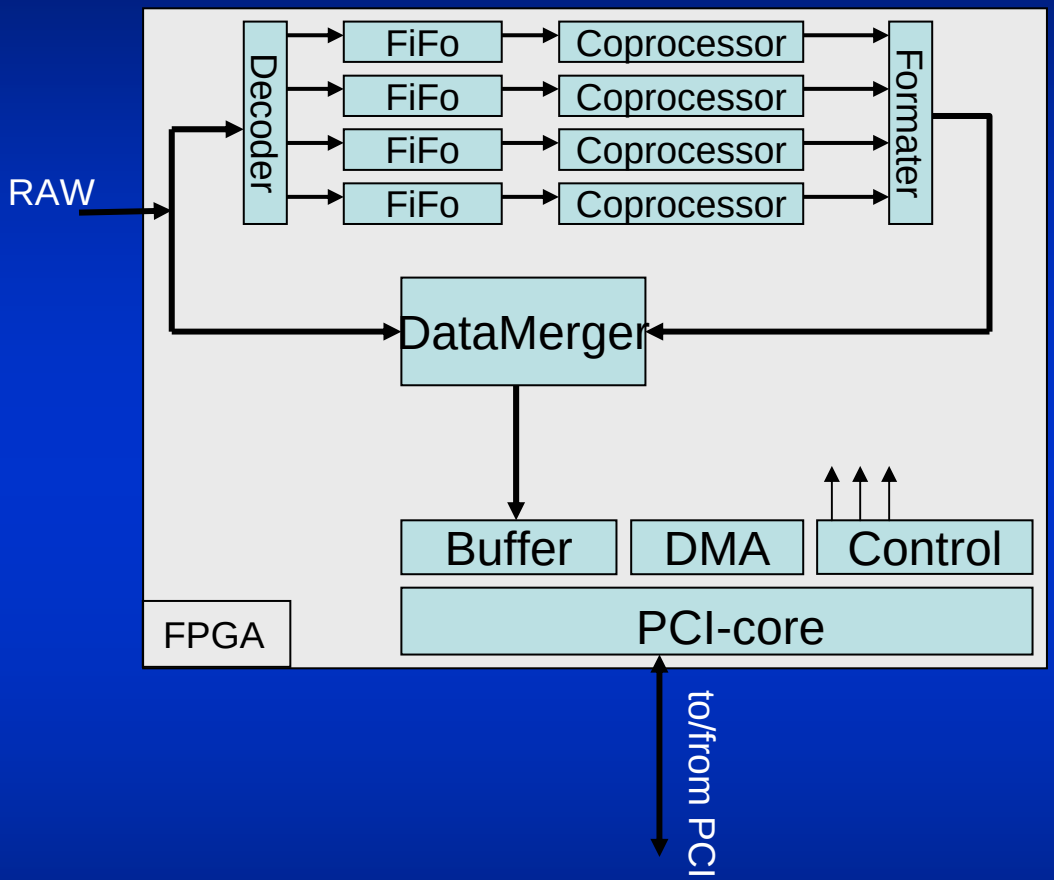
CF-Algorithm II



- 1.stage: CF-DECODER
  CF calculate the weighted time, the sequential charge and the total charge.
- 2.stage: CF-MERGER
  The values are compared and merged if they correspond
- 3.stage: DATA-MERGER
  The clusters are written to a memory

RAW ⟶ Decoder ⟶ Merger ⋯⋯▸ Data-Merger

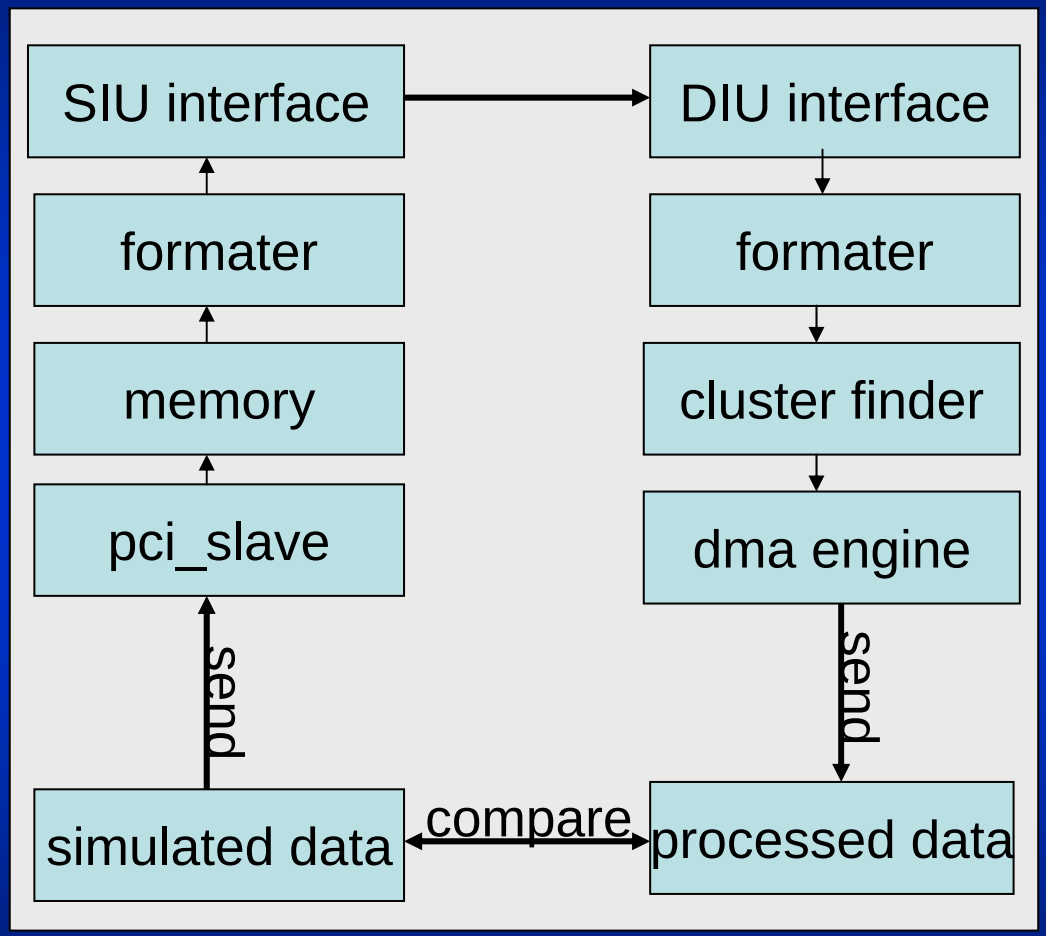
Parallel Pre-Processing



- Each CF operates locally on one row.
- CF can operate in parallel
- 2 datastream: one raw data, one processed data

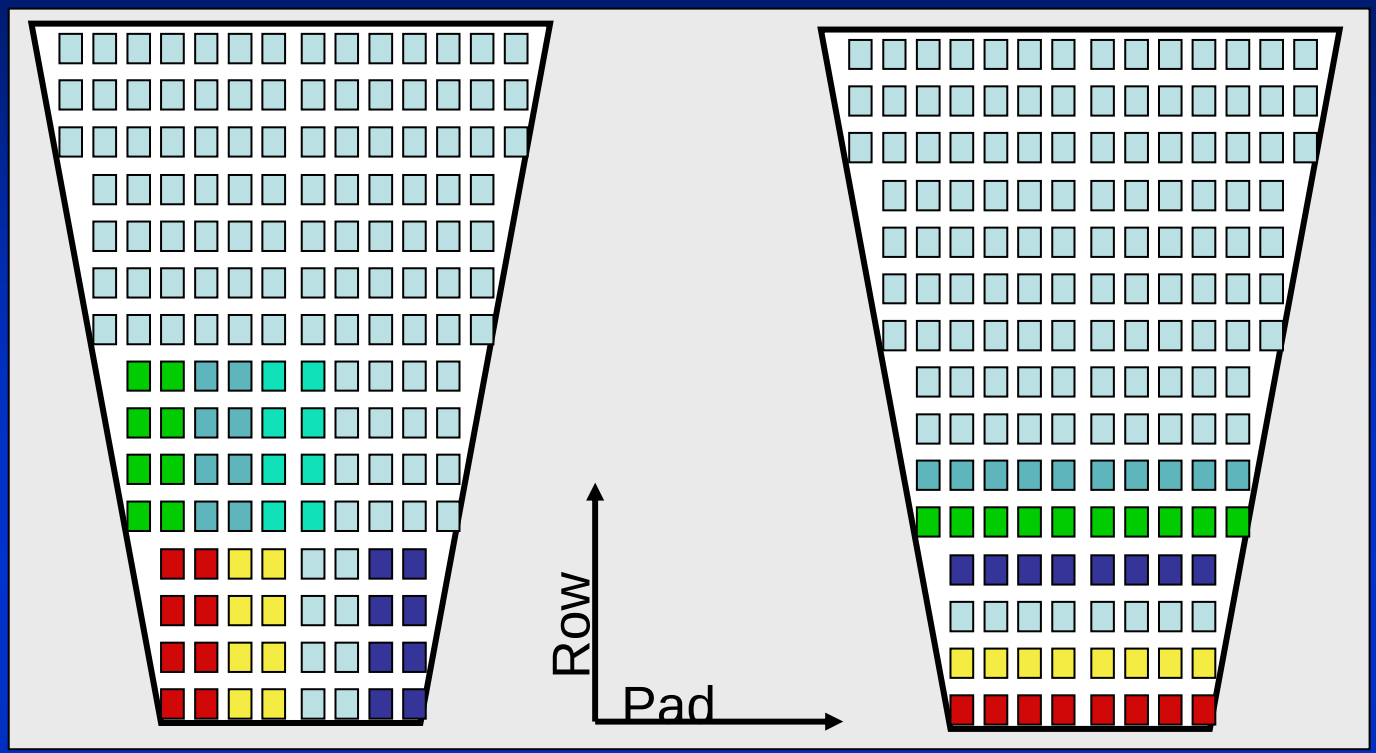
- After processing they are merged into one stream and send over PCI DMA transfers to the main memory of the Host.

Hardware testbench



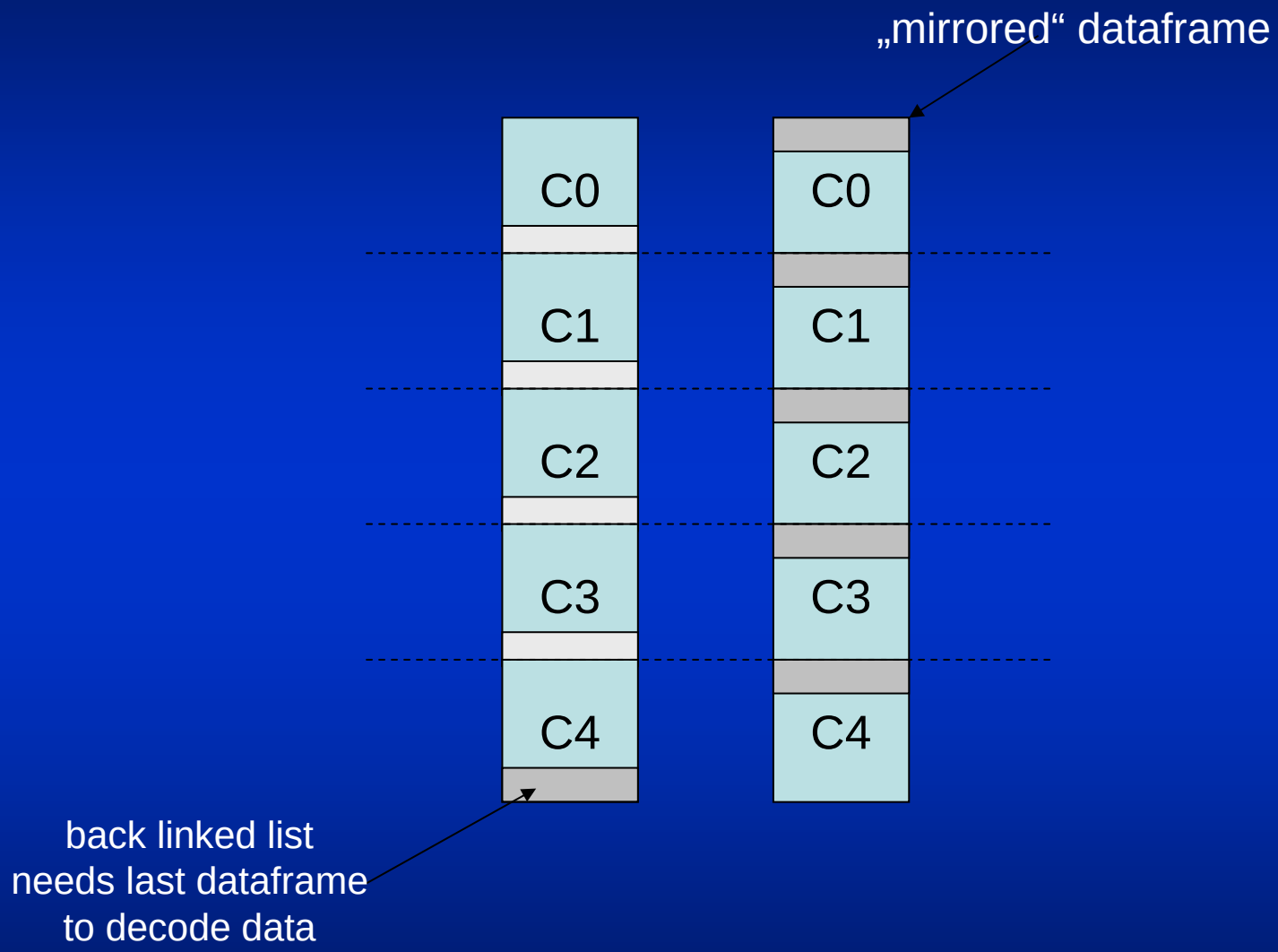
- simulated data is send to a PCI card and stored into a memory

Readout with LUT



FEC_0 Channel_0 FEC_0 Channel_1 FEC_0 Channel_2 .... FEC_F Channel_E FEC_F Channel_F	Look-up table	FEC_0 Channel_0 FEC_1 Channel_4 FEC_2 Channel_6 .... FEC_6 Channel_D FEC_9 Channel_A

Reverse backlinked list



Summary

- FPGAs are widely used for data transportation
- FPGAs provide powerful DSP and arithmetic functionality

  => processor located directly in the data chain
- Calculations can be done highly parallel
- Logic can be customized and adapted to the requirements AND changed within a few microseconds
- In Offline mode the logic normaly dedicated to readout can be re-used for Co-processing.

Cluster-Finder:

- A first design exists and has been checked with simulated data. Also the preprocessed data was checked against the data produced by the C-algorithm that is used for clusterfinding in STAR.
- Design is currently redesigned to improve speed and accuracy.

  OLD CF-Decoder : 570 LogicCells, fmax=40MHz, relative scale

  NEW CF-Decoder: 550 LogicCells, fmax>100MHz, absolute scale