Contribution ID: **88** Type: **Poster**

The CBM-TRD Cluster Finder

Tuesday, 20 September 2022 16:40 (20 minutes)

This contributions presents the implementation of the CBM-TRD cluster finder. The cluster finder is implented with Vitis HLS in an FPGA.
The CBM experiment at FAIR will focus on rare probes of the QCD phase diagram at high net-baryon densities.
The free streaming DAQ has to process up-to 2 TB/s of raw data. This data undergo online event selection, where 4D track reconstruction is necessary. To accelerate the online event selection, the data is preprocessed in the FPGA.
I will demonstrate how HLS can be utilized to implement a high-throughput cluster-finder capable of processing up-to six 4.8 GB/s GBT-Links.

Summary (500 words)

Traditionally FPGA firmware was developed solely with Hardware Description Languages (HDL) such as Verilog or VHDL.
However, with the steady improvements of tools like Vitis HLS (High Level Synthesis) it is now possible to write parts of the firmware with higher level languages like C++.
Using HLS allows faster development cycles, easier code reuse and, most importantly, to efficiently write complex algorithms for the FPGA.
The Compressed Baryonic Matter (CBM) experiment at the Facility for Antiproton and Ion Research (FAIR) will investigate the QCD phase diagram at high net-baryon densities.
The experiment employs a free streaming data acquisition with self-triggered front-end electronics (FEE).
At interactions rates of up to 10 MHz the data acquisition (DAQ) chain has to process very high data loads of up-to 2 TB/s. As such a large amount of raw data can't be written to storage, the data has to be filtered online to interesting events. The online event selection has to achieve 4D tracking in real time. In order to reduce the computational load of the compute cluster some of the reconstruction can be moved to the read-out FPGA.

The CBM-TRD is equipped with the SPADIC front-end ASIC. The SPADIC allows for an oscilloscope-like sampling of the detector signals. With up-to 32 samples and a sampling rate of 16 MHz.
Additionally, the ASIC features a forced neighbour readout logic, which allows to read out pads adjacent to the pad, which fulfilled the trigger logic without lowering the threshold.
The SPADIC marks the trigger type of the data, which aids the cluster finding algorithm, as each cluster is surrounded by two forced neighbour triggers.

The cluster finder can be split into several parts: The pad sorter, the cluster aggregation and the cluster feature extraction.

The pad sorter prepares the data for the cluster aggregation. The sorting takes the data from three merged and time-sorted GBT-links and outputs a stream sorted according to the pad configuration of the detector.
The cluster aggregation combines the individual trigger messages into a cluster. Each cluster starts and ends with a forced neighbour trigger message, making it easy to combine messages to a cluster.
In the next step the cluster features are extracted, such as the cluster charge and the cluster position.
The cluster finder is fully pipelined with an initiation interval of 1 and operates at a frequency of 200 MHz.

HLS excels at auto-pipelining compute heavy algorithms and streaming applications. This makes it an excellent tool to quickly and efficiently develop high-throughput designs, which are a requirement for the high

data rates that are expected within CBM.

Primary author: SCHLEDT, David (IRI - Frankfurt University)

Co-author: KEBSCHULL, Udo Wolfgang (Goethe University Frankfurt (DE))

Presenter: SCHLEDT, David (IRI - Frankfurt University)

Session Classification: Tuesday posters session

Track Classification: Programmable Logic, Design Tools and Methods