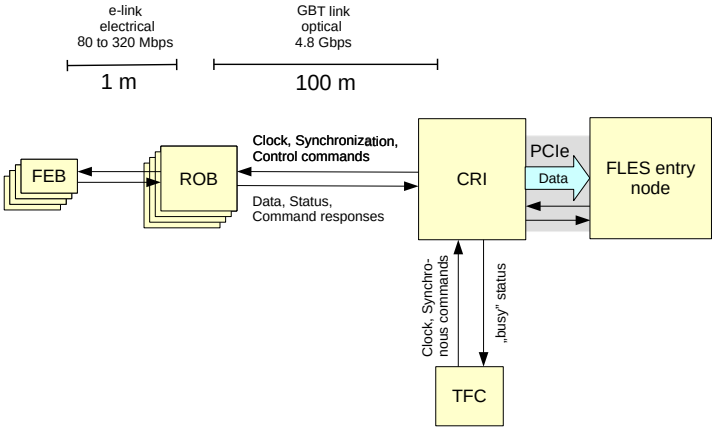
CRI firmware for CBM DAQ

Presenter: Wojciech M. Zabołotny[1],

[1]Institute of Electronic Systems, Warsaw University of Technology

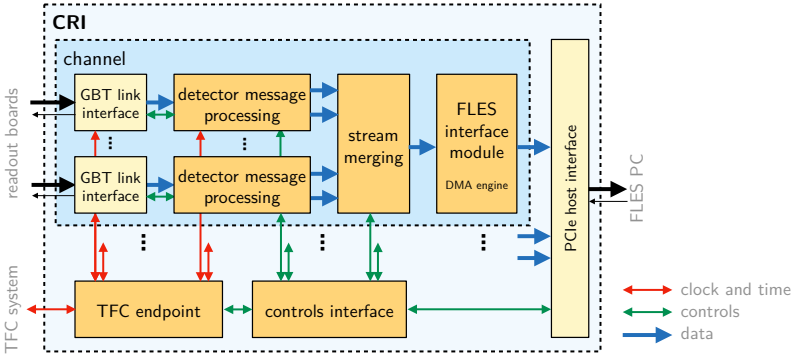
PANDA FEE/DAQ Workshop 25-26 June 2026

Overall view of the CBM DAQ



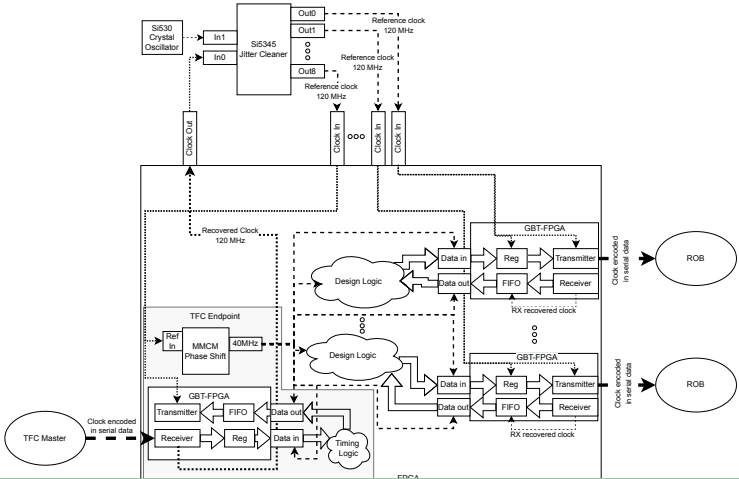
The 1st version of CRI - BNL-712 [1]

Overview of the CRI firmware

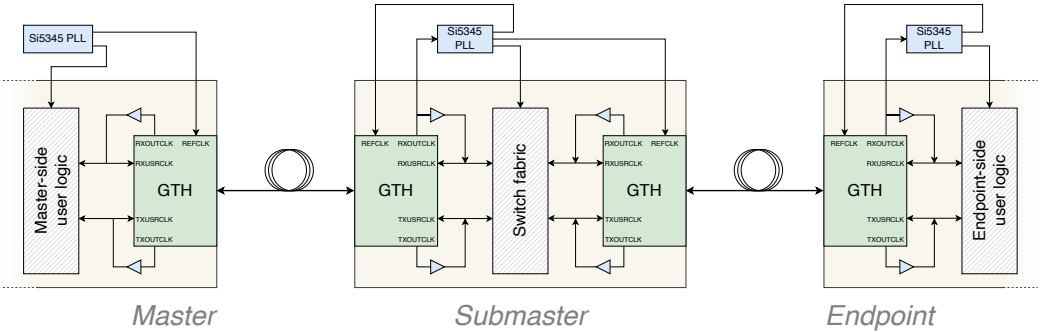


Source: [2], Figure 5.1

Usage of GBT-FPGA in CRI

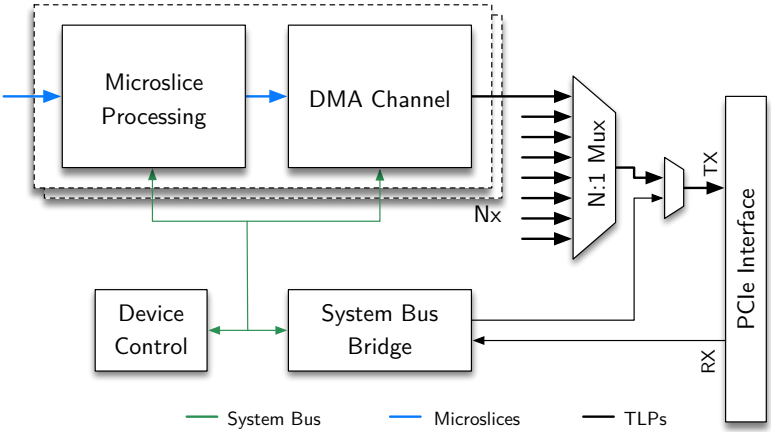


Implementation of TFC (by KIT)

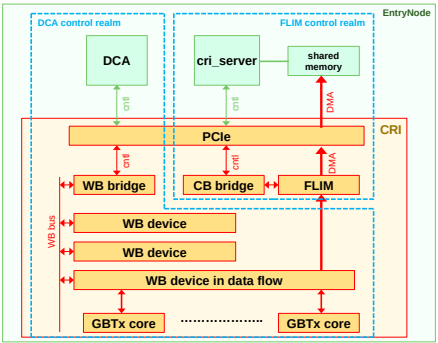


- The Timing and Fast Control (TFC) system (developed by KIT) provides distribution of reference clock and time, and fast control path ([2], Chapter 6).

Implementation of FLIM (by FIAS) [3]

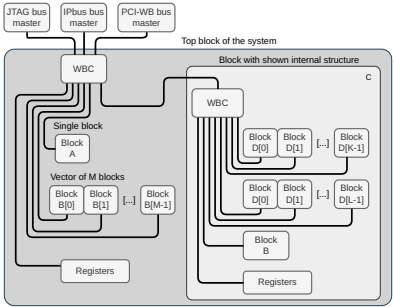


Organization of the Control Bus in CRI



- CRI is controlled via PCIe interface.
- The same interface is also used for transferring the detector data via DMA.
- Therefore, the control is implemented independently for the DMA engine and for the rest of the CRI FW.

Organization of the Control Bus in CRI

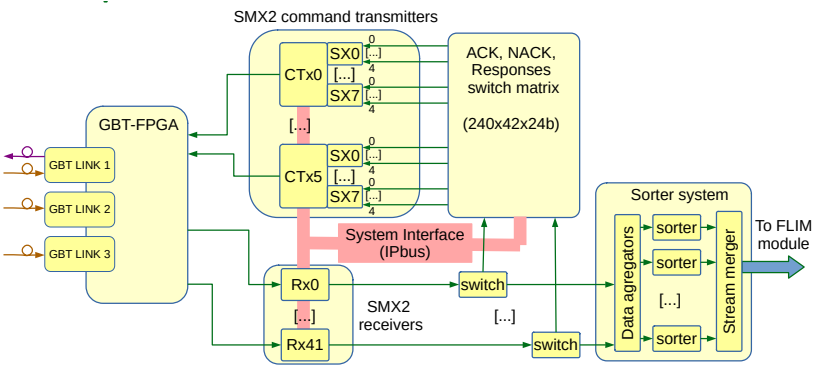


- The main part of the CRI FW is controlled via Wishbone bus.
- The address space is organized hierarchically and managed with AGWB [4].
- AGWB automatically implements bus hierarchy and registers in VHDL together with software structures and routines, based on the system description in XML.
- At the beginning of the address space there is a special Zeropage area supporting error detection and alarm system.

Control of FEE ASIC in STS

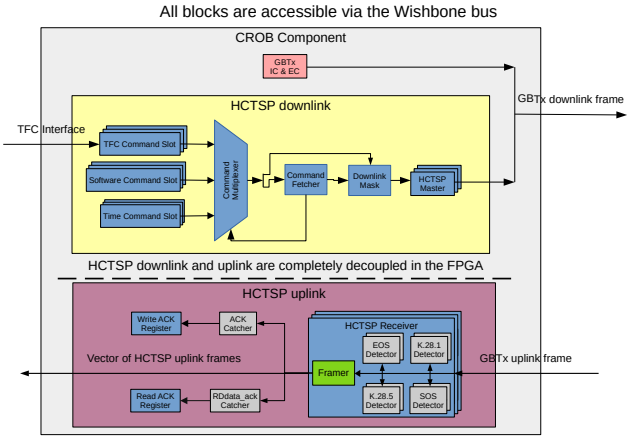
- For STS detector a special SMX ASIC has been developed [5].
- The ASIC implements a dedicated HCSTP protocol [6].
- The HCSTP protocol was reused also in other ASICs (e.g. SPADIC used by TRD).
- The protocol ensures constant latency downlink communication needed for FEE synchronization.

Implementation of SMX controller in old DPB



- Use of slow control interface (IPbus) required implementation of full controller in FPGA
- Routing of command responses consumed significant resources

Implementation of HCSTP

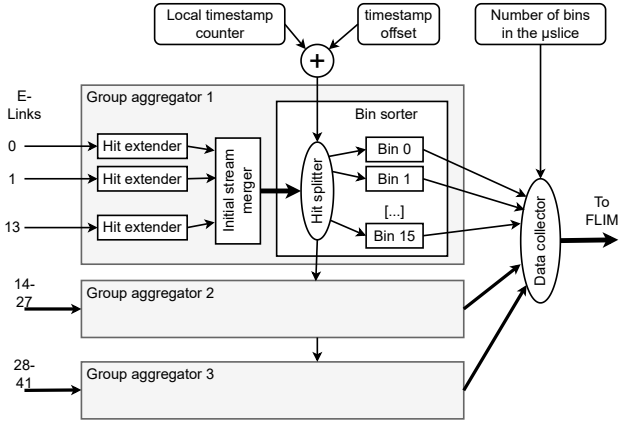


- Use of faster PCIe interface enabled splitting TX and RX parts of controller
- Control handshake is fully handled by software

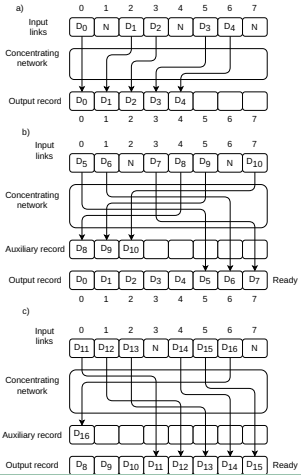
Data concentration in the CBM STS DAQ

- The data concentration in the CBM STS DAQ was evolving during the preparation of the experiment.
- The initial attempt - the full data sorting [7]. It had problems due to beam intensity fluctuations, and possible data corruption due to transmission errors.
- The second attempt - application of the bin sorter. It was resilient to corrupted data. It better handled beam intensity fluctuations. However, the amount of data loss was significant.
- The third attempt - assigning the data to microslices according to their arrival time.

Data concentration based on bin sorter



Data packing into output words



- The data from multiple input channels are delivered as non-dense streams.
- They should be packed into the wider words accepted by the DMA engine without wasting space.
- That packing was a problem in first implementations of the data concentration [8].
- Finally it has been solved by using the specially designed interconnection networks [9, 10]

Software controlling the CRI

- AGWB enables easy interfacing with various languages including Python and C++.
- The initial development was done mainly in Python due to convenience of interactive work.
- Certain operations done on the hardware require proper synchronization and performing them as transactions.
- For that purpose the DCA (Device Control Agent) was written in C++ by Walter Müller ([2], Chapter 5.4).
- Thanks to the Python bindings it is still possible to develop and debug user software in Python.
- Critical parts are implemented in C++ and performed in a carefully synchronized way.

Alternative readout board – GERI

- Using CRI is not possible in all institutes participating in preparation of CBM experiment.
- An alternative solution has been prepared with a GERI board based on a standard commercial TEC0330 board. A significant part of development was done in the framework of EU H2020 EURIZON project.
- The alternative readout chain may use a dedicated GBTx emulator (GBTxEMU) [11] instead of restricted GBTX ASICs.
- The GERI platform uses a simple HLS-implemented DMA engine [12], prepared specifically for small installations (e.g. single PC or small cluster).
- That alternative readout chain also will be used for CBM components testing setups, and is going to be used by other experiments.

Bibliography I

[1] Alexander Paramonov. FELIX: the Detector Interface for the ATLAS Experiment at CERN. *EPJ Web of Conferences*, 251:04006, 2021. DOI: 10.1051/epjconf/202125104006.

[2] CBM Collaboration. *Technical Design Report for the CBM Online Systems – Part I, DAQ and FLES Entry Stage*. GSI Helmholtzzentrum fuer Schwerionenforschung, GSI, Darmstadt, 2023. DOI: 10.15120/GSI-2023-00739.

[3] Dirk Hutter. *An input interface for the CBM first-level event selector*. doctoralthesis, Universitätsbibliothek Johann Christian Senckenberg, 2021. https://publikationen.ub.uni-frankfurt.de/frontdoor/index/index/year/2021/docId/59159

[4] Wojciech M. Zabołotny, Marek Gumiński, Michał Kruszewski, and Walter F. J. Müller. Control and Diagnostics System Generator for Complex FPGA-Based Measurement Systems. *Sensors*, 21(21):7378, November 2021. DOI: 10.3390/s21217378.

Bibliography II

[5] K. Kasinski, R. Kleczek, and R. Szczygiel. Front-end readout electronics considerations for Silicon Tracking System and Muon Chamber. *Journal of Instrumentation*, 11(02):C02024–C02024, February 2016. DOI: 10.1088/1748-0221/11/02/C02024.

[6] K. Kasinski, R. Szczygiel, W. Zabolotny, J. Lehnert, C.J. Schmidt, and W.F.J. Müller. A protocol for hit and control synchronous transfer for the front-end electronics at the {CBM} experiment. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 835:66 – 73, 2016. DOI: https://doi.org/10.1016/j.nima.2016.08.005.

[7] Wojciech M. Zabołotny. Dual port memory based Heapsort implementation for FPGA. In *Proc. SPIE*, volume 8008, pages 80080E–80080E–9, Wilga, Poland, June 2011. DOI: 10.1117/12.905281.

[8] Marek Gumiński, Wojciech Marek Zabołotny, Adrian Paweł Byszuk, and Krzysztof Poźniak. Sorting of STS-XYTER2 data for microslice building for CBM experiment. In *Proceedings of Topical Workshop on Electronics for Particle Physics — PoS(TWEPP2018)*, page 143, Antwerp, Belgium, May 2019. Sissa Medialab. DOI: 10.22323/1.343.0143.

Bibliography III

[9] Marek Gumiński, Michał Kruszewski, Bartosz Marek Zabołotny, and Wojciech Marek Zabołotny. Beneš Network-Based Efficient Data Concentrator for Triggerless Data Acquisition Systems. *Electronics*, 12(6):1437, March 2023. DOI: 10.3390/electronics12061437.

[10] Wojciech Marek Zabołotny. Scalable Data Concentrator with Baseline Interconnection Network for Triggerless Data Acquisition Systems. *Electronics*, 13(1):81, December 2023. DOI: 10.3390/electronics13010081.

[11] W.M. Zabołotny et al. GBTX emulator for development and special versions of GBT-based readout chains. *Journal of Instrumentation*, 16(12):C12022, December 2021. DOI: 10.1088/1748-0221/16/12/C12022.

[12] Wojciech M. Zabołotny. Versatile DMA Engine for High-Energy Physics Data Acquisition Implemented with High-Level Synthesis. *Electronics*, 12(4):883, February 2023. DOI: 10.3390/electronics12040883.

Thank you for your attention!