TDAQ Phase-II Upgrade Project

Forward Feature EXtractor (fFEX) User Requirements, Specifications and Preliminary Design Document

Abstract

This User Requirements and Specification Document describes the user requirements and the preliminary specification of the ATLAS fFEX, which is part of the L0Calo trigger system and acts as a forward-region extension of the Phase-1 Level-1 calorimeter trigger feature extractors. This trigger sub-system shall efficiently find and identify jets and electromagnetic objects (dominantly electrons/positrons), and shall report these to the L0Global Trigger.

ATLAS TDAQ Phase-II Upgrade Project	
ATLAS Doc:	ATL2-D-MG-xxx
EDMS Id:	xxx
EDMS Url:	https://edms.cern.ch/document/xxx
Version:	1.1
Created:	April 16, 2021
Last modified:	March 4, 2022

Prepared by:	**Checked by:**	**Approved by:**
A. Álvarez Fernández, U. Schäfer, S. Tapprogge		

15 [INTENTIONALLY BLANK PAGE]

Revision	Date	Description
0.1	4/21/2021	Initial draft copied from PFM.

Revision History

Table of Contents

1. Introduction

1.0.1 Purpose of the document

This document describes the user requirements and the specifications of the ATLAS fFEX. The purpose is to collect and provide the specifications on the hardware, firmware and software that will be provided by the responsible developing group to the users. It will also act as a reference for the users of the fFEX module.

ADD comment on preliminary h/w and f/w design HERE

This document is a deliverable to the combined Specification and Preliminary Design Review foreseen for the fFEX.

1.1 Scope

This document covers the requirements on and the specification of the functionality for the fFEX hardware, firmware and software to be delivered. The hardware requirements concern

- general requirements for operation: the PCB form factor, power requirements and cooling
- interfaces to external modules via the optical input and output modules for the real-time and read-out path and the connector types
- the processing and control FPGAs
- other required components, such as external memories and debugging components
- options for loading firmware configurations

The firmware requirements concern

- firmware that will be delivered to enable and repeat basic functionality tests of the optical modules and FPGAs
- firmware blocks that are part of the debugging infrastructure (in form of spy memories in the processing FPGA)

The software requirements concern

- basic software for loading firmware configurations
- basic software for operating spy memories
- frameworks for bit-wise simulation of algorithms with defined interfaces, test pattern generation and playback through playback memories, and analysis of spy memory data with simulated data

Some basic, essential firmware and software will be provided when delivered to the developing groups, but they will evolve with the development of the GCM. The evolution of the firmware and software is considered to be non-essential for the initial delivery of the fFEX.

It will also describe possible modes of operation of the fFEX hardware for developers that are foreseen with the here presented requirements.

1.2 Definitions, acronyms and abbreviations

The acronyms follow closely the acronyms used in [1].

1.2.1 Glossary

CTP CTPCentral Trigger Processor. 3

DCS Detector Control System. 4

FELIX Front-End LInk eXchange. 3, 4, 12, 15, 21

fFEX forward Feature EXtractor. i, 1, 3–5, 12–15, 17–23, 26–28

GCM Global Common Module. 1, 4, 15

jFEX jet Feature EXtractor. 23

JTAG Joint Test Action Group (developer of IEEE Standard 1149.1-1990). 13, 14, 18

L0Calo Level-0 Calorimeter Trigger (Run 4 and beyond). 3

L0Muon Level-0 Muon Trigger System (Run 4 and beyond). 3

LASP LAr Signal Processor. 4

PCB Printed Circuit Board. 1

PDR Preliminary Design Review. 5

TDAQ Trigger-DAQ. i

TIP Trigger Input signals as they enter the CTP before the trigger logic. 4

TTC Trigger, Timing, and Control system. 3, 4

1.3 References

This section should provide a complete list of all the applicable and reference documents, identified by title, author and date. Each document should be marked as applicable or reference. If appropriate, report number, journal name and publishing organisation should be included.

[1] The ATLAS Collaboration, *ATLAS Trigger and Data Acquisition Phase-II Upgrade Technical Design Report*, Tech. Rep. CERN-LHCC-2017-020; ATLAS-TDR-029, CERN, Geneva, Dec, 2017. https://cds.cern.ch/record/2285584. 1

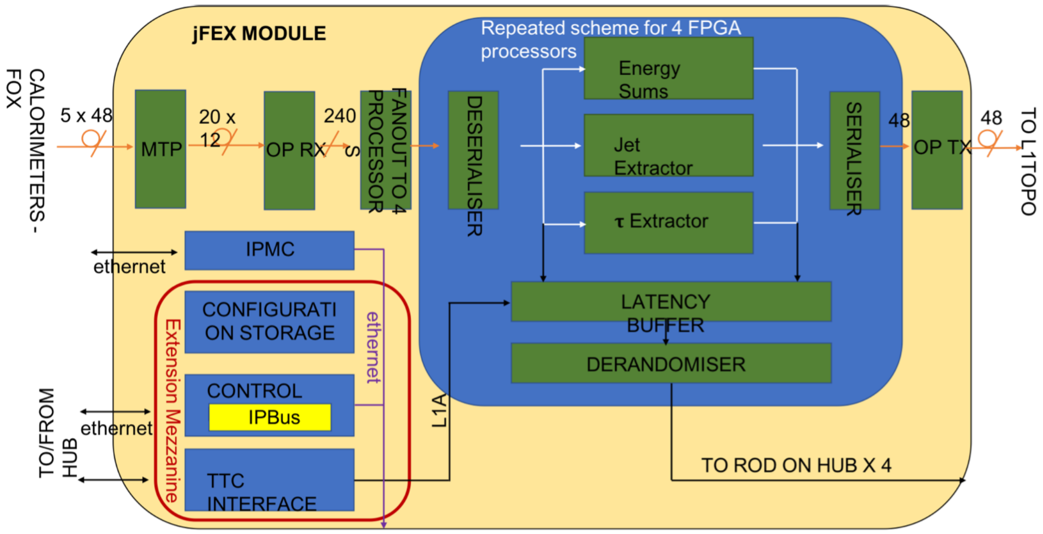

Figure 2.1: Functionaliy of the fFEX modules - PLACEHOLDER - taken from jFEX PRR - NEEDS TO BE REPLACED

.

2. General (Functional) Description

The fFEX (Forward Feature EXtractor) is part of the Phase-II upgrade of the ATLAS Trigger and Data Acquisition (TDAQ) system for the High Luminosity upgrade of the Large Hadron Collider (HL-LHC). The expected peak luminosities will be $\mathcal{L} = 7.5 \times 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$, corresponding to approximately 140 inelastic proton-proton collisions per bunch crossing. It is foreseen to collect $\mathcal{L} = 300 - 350\,\mathrm{fb}^{-1}$ per year and a total integrated luminosity up to $4000\,\mathrm{fb}^{-1}$.

To meet the requirements on the trigger system at the first stage of event selection, a baseline architecture with a single-level hardware trigger that features a maximum rate of $1\,\mathrm{MHz}$ and $10\mu\mathrm{s}$ latency is foreseen. The hardware-based Level-0 Trigger system is composed of the Level-0 Calorimeter Trigger (L0Calo), the Level-0 Muon Trigger (L0Muon), Global Trigger and the Central Trigger sub-systems: In the L0Calo subsystem, the Phase-I calorimeter feature extraction trigger processors are extended by a new forward Feature EXtractor (fFEX) to reconstruct forward jets and electrons. The Global Trigger will replace and extend the Run 2 and Phase-I Topological Processor, by accessing full-granularity calorimeter information to refine the trigger objects calculated by L0Calo, perform offline-like algorithms, and calculate event-level quantities before applying topological selections. The final trigger decision is made by the Central Trigger Processor (CTP), which can apply flexible prescales and vetoes to the trigger items. The CTP also drives the Trigger, Timing and Control system network to start the readout process of the detectors.

The hardware implementation of the fFEX consists of the following components: TBD

2.1 Perspective

2.2 General Capabilities

In standalone mode the input and output data is delivered/recorded from/in internal spy memories.

In any mode the fFEX can be connected to a FELIX module to receive the clock, configuration and per event TTC information and send the readout and trigger information via optical fibres. The fFEX can also be run independent of a FELIX module providing a clock internally and storing the readout data in internal spy memories.

The firmware is structured in firmware blocks, that are also foreseen for the GCM. Since the development of these firmware blocks is the main purpose of the fFEX, initially the only frame blocks, as well as blocks for testing and debugging will be provided. The main basic block is the low-level frame firmware for the management of FPGA interfaces, reference clocks, I/O interfaces, and the general infrastructure. For the trigger algorithms a basic trigger framework that will handle input and output (real-time and readout) around a mock algorithm will be provided, which will serve as a development starting point. Spy and plackback memories using FPGA memory blocks will aid the development and testing of the trigger, data aggregation and multiplexing firmware. The memories are also capable of comparing incoming data with pre-stored patterns for real-time comparison of the data.

The provided software will focus on the control and configuration of the FPGA and the management of the spy and playback memories. Frameworks will be provided that create test patterns for the firmware tests, handle the spy and playback memories, provide bit-wise simulation of algorithms using the mock algorithm and a framework that supports the analysis of the spy memories and the output of the simulation. The software blocks will be reused for the general TDAQ software and on- and off-line simulation or even be part of it.

2.3 Interfaces

Interfaces to the following systems should be supported by the fFEX:

- Real-time data source systems (the same as the source systems for the GCM):
  - the calorimeter front-end electronics of the LArg (LAr Signal Processor (LASP)) will send full-granularity cell energies every LHC bunch crossing via optical fibres
- L0Global Trigger where the results of the algorithms will be sent every LHC bunch crossing
- FELIX, to which TIPs, trigger algorithm results, and error reporting are sent via dedicated optical read-out fibres and from which clock, configuration and per-event TTC information are received through dedicated optical read-out fibres
- Configuration and control systems, possibly via IPBUS
- DCS for basic environmental monitoring and control

The data is expected to be correctly formatted. The optical fibres are connected via standard MTP connectors to the optical modules. The optical input and output modules are configurable to match the desired interface link-speeds depending on which layer the fFEX emulates.

2.4 General Constraints

2.5 Operating Environment

fFEX can be operated standalone without any external source/sink systems. The inputs to the firmware is provided through internal input spy memories (designed as memory blocks in the firmware) and the output to external systems is captured in output spy memories. It can also be interfaced to several external real-time data source/sink systems as well as a Phase-II FELIX for the readout as listed in Interfaces.

For ease of operation the fFEX will conform to an ATCA form-factor with required infrastructure to support operation in an ATCA shelf (power and sensor management via IPMI and requirements). Since developing sites might not be equipped with a ATCA shelf, operation without a shelf is also possible overriding the management features. However, power and environmental management should be provided by the user.

2.6 Assumptions and Dependencies

2.7 Development tools

For the development the following tools are foreseen. Most tools follow standard development tools either used and developed by the hardware manufacturer or tools that are commonly used at CERN.

- git versioning system as the default versioning tool at CERN. This will be used for all the revelant files (schematics, firmware and software) used in the design phase. The firmware and software git repositories will also receive major updates when the fFEX firmware/software reaches an important development step.
- Cadence for schematics development and layout
- Vivado for firmware development and synthesis when the choice of FPGA falls to Xilinx FPGAs
- C++, Python, Java and TDAQ framework for the control and simulation software

2.8 Verification plan

The verification plan will be described in detail in the PDR document, but it will contain the

- verification of the layout after schematics design and before prototype production
- verification of power requirement with powering tests
- verification of the input/output capabilities and data integrity test with loop-back tests and prototype firmware
- verification and functionality tests of the initial firmware through data integrity checks on optical links and on spy/playback memories, and latency checks
- functionality tests of software frameworks using the mock algorithm: test pattern generation, control and readout of playback and spy memories, verification of the simulation and the analysis tools

3. Requirements

Adrian: this is a different version for requirements (v2), but the old one still exists

3.1 Algorithm requirements

- fFEX to efficiently identify and trigger on electron candidates with pseudorapidities $|\eta| > 2.5$
- fFEX to efficiently identify and trigger on jet candidates with pseudorapidities $|\eta| > 3.2$
- fFEX to provide energy sums (E_x, E_y, E_T) for the region $|\eta| > 3.2$ as well as for $3.2 > |\eta| > 2.5$)
- fFEX needs the necessary processing power to exploit the full granularity of the calorimeter information with more complex algorithms (beyond simple sliding window / energy sums) in order to address the more challenging HL-LHC forward phase space region

3.2 Interface requirements

- fFEX to receive cell energy information in full granularity (both transverse and longitudinal) for the region abs(eta) > 2.5 (i.e. from FCAL, EMEC-IW and HEC-IW)
- All Input data transmitted to fFEX shall be written to the readout by the source system (i.e. the LASP) for all L0 accepted events
- fFEX to receive cell energy information in full longitudinal granularity, pre-summed to the transverse granularity of EMEC-IW) for the region 2.5 > abs(eta) > 2.2 (i.e. EMEC-OW and HEC-OW)
- fFEX to receive cell (transverse) energy using 11 bits
- fFEX to transmit TOB information for electron candidates, jet candidates and energy sums to L0global
- The fFEX shall receive and transmit data to the Phase-II FELIX system serving as a receiver for clock, configuration and TTC information as well as a data source for the readout information for the TIP, trigger decision and status and error flags.
- fFEX to have jTAG interface as one option to load/configure FPGAs
- fFEX to have (IPBus?) interface to remotely configure FPGAs Uli - do we think we need that ? not possible with TopoFex derived scheme. alternatively we might require flash update via IPbus (planned for TopoFex, not yet implemented)

3.3 Hardware requirements

- The fFEX shall be able to receive or send real-time data via optical fibres at link speeds up to $25.8\,\mathrm{Gb/s}$.
- one fFEX module must not dissipate more than 400 W of power (maximum) and shall consume more than 300 W ?? under normal load. The power modules have to be able to provide the required power under maximum load.
- fFEX to have (IPBus?) interface to remotely configure FPGAs
- Passive cooling solutions (heat sinks and sufficient clearance) shall be provided, such that the cooling while operating in an ATCA shelf is not hindered. The passive cooling solutions can be augmented by the cooling solutions provided by the user when operating outside of an ATCA shelf.

- fFEX modules shall be designed using an ATCA form factor for the PCB. The fFEX shall also provide corresponding power and sensor management functionality.
- fFEX module shall be able to be operated in stand-alone mode (without any external modules), making use of playback/spy memories for real time data path testing
- specify/comment on fFEX module latency value(s) ?

3.4 Firmware requirements

- Firmware for basic functionality/integration test shall be provided for initial test of data integrity over the optical links (e.g. IBERT for loop-back tests), and functionality tests of configuration and control of the FPGAs.
- fFEX frame Firmware shall include infrastructure and command/control firmware. This includes the firmware blocks that interface to the MGTs, de/encoding and de/serialisation of the input/output data and that provide the spy memory functionality.
- The update policy of the firmware development tools will be centralised, in order to keep the development sites in sync. The firmware development will use a repository accessible by all developers.
- The final design workflow will be script-based, to minimize the use of Graphical User Interface-based design options.

3.4.1 Algorithmic firmware requirements

- As one baseline algorithm, a sliding window for the EM trigger shall be implemented (with energy sums used to provide discrimination based on isolation variables)
- Refined EM trigger algorithms shall determine quantities to characterize the shower shape
- as one baseline algorithm, a sliding window for the jet trigger shall be implemented (for at least two different jet size parameters, e.g. R = 0.4 and R = 1.0)
- Double counting of single EM trigger objects shall be excluded
- Double counting of single JET trigger objects shall be excluded
- flagging of overflow conditions ?

3.4.2 Infrastructure firmware requirements

- Firmware shall be provided for initial testing of optical links and of FPGA configuration/control functionality
- Firmware shall provide „infrastructure“ capabilities/functionality (e.g. interface to MGTs, de-/encoding, de/serialization, playback/spy memories (both for real time and readout data paths)

3.5 Readout requirements

- fFEX shall be able to be read out at the maximum average L0 trigger rate of 1 MHz
- the readout data shall include the TOB information sent to L0global and possibly (in case of overflow) those TOBs found, but not sent
- for debugging purposes, it shall be possible (at rates lower then 1 MHz) to read out the input data received from LASP

3.6 Control requirements

- adequate firmware for basic functionality/integration tests shall be provided

3.7 Configuration requirements

- The fFEX provides the possibility to load the firmware from SPI memories and via JTAG.
- An option for remote configuration of the FPGA via a ZYNQ chip or IPBus shall be provided.
- Software shall be provided that loads the firmware (remote configurations).
- fFEX software shall be able to configure and setup the board.
- fFEX software shall able to operate the spy and playback memories.

3.8 Monitoring requirements

-

3.9 Software requirements

- Software to configure and control the fFEX module as well as the firmware, to control/write/read the playback/spy memories and to create test patterns shall be provided
- bitwise simulation framework shall be provided, e.g. to compare generated, recorded and simulated test patterns
- Software development shall make use of a central repository

4. Specifications

4.1 Hardware specifications

4.1.1 System architecture

The fFEX is a double width ATCA module, fully compliant to ATCA specifications. It receives data from the calorimeters (LAr) on the real-time data path (RTDP) on optical fibres through the front panel on 24-way MTP/MPO connectors. The fixed fibre assembly gangs together two 12-way FireFly optical-electrical converters, which send their electrical output signals to the MGT receivers of two FPGAs of type XCVU13P-L2FLGA2577E. The input link count is 96 links per FPGA, transmitting at 25.78125 Gb/s. After processing, the results are sent out of the processor FPGAs on 24 MGT links each, to 12-way FireFly optical-electrical converters, and sent to L0Global on optical fibres. Again, two converters each are ganged together into a 24-way MTP/MPO connector on the front panel.

Module control is exerted by a controller mezzanine (UltraZed), based on a Xilinx Zynq device. The controller is linked to the ATLAS TDAQ via two Ethernet connections located on the front panel. The controller is interfaced to the processor FPGAs by 4 MGT links each

.

ATCA level control and environmental monitoring is done via a CERN IPMC module. It is interfaced to ATCA backplane zone 1 and communicates to main power control and payload via status lines and I2C links.

The fFEX is linked to the ATLAS FELIX system for the purpose of clock and timing data distribution and readout. The physical interface goes via a single 12-way MTP/MPO connector on the front panel, with fibre connection to a 4+4-way bidirectional FireFly module that is electrically linked to the module controller. The module controller is in charge of clock recovery, timing data decoding and distribution, and ROD functionality.

The power supply concept (PIM / primary brick / local regulators on a panel) is being taken over from the Phase-1 modules. Power supplies are being monitored and controlled mainly via I2C and this information needs to be interfaced to the module controller and/or IPMC.

Further environmental monitoring is possible from opto-devices and various temperature, current, and voltage monitoring points.

4.1.2 Processing FPGAs

Each fFEX module contains 2 FPGAs, and there are 4 modules in total, 2 on each hemisphere. Since each FPGA covers one quadrant in phi, a 100% duplication of all data is provided upstream by neighbouring octants on each side. The chosen FPGAs are of the type XCVU13P-L2FLGA2577E (Virtex Ultrascale+). The number of input MGT links per FPGA that will be used is 96 (8 times 12) and the number of output MGT links 24 (2 times 12).

4.1.3 Optical modules

Samtec FireFly modules are used for the optical-electrical conversion. 12-way optical buses from each opto-electrical device are routed into 3 quads of the FPGA. Ideally the 3-quad circuit should be pre-routed and re-used if possible and if beneficial in the course of module design. The natural segmentation at FPGA level is 4 super logic regions (SLR), each containing 4 quads on each side of the FPGA. For the real-time inputs and outputs, each opto-device will use 3 quads of the same SLR, meaning that all the SLRs will be used for the inputs. For the outputs, the two central SLRs will be occupied with one opto-device in each, allowing for the separation between jet and electromagnetic algorithms. The real-time path (96 inputs, 24 outputs) will use a 25.78125 Gb/s crystal-based transmission, with synchronization to LHC clock.

4.1.4 Clock system

Clock, TTC data, and readout handling is implemented on the fFEX and its UltraZed controller mezzanine. The fFEX is interfaced directly to the FELIX system via optical links using two 4+4 channel bidirectional FireFly modules, one on each side of the FPGA. The MGT links used for this purpose are located in the two central SLRs, and those incoming links are routed into one MGT quad of the control mezzanine each. The transmitter lines of those quads are the readout links back to FELIX. Data rates are assumed to be up to 10.24 Gb/s or 25.78125 Gb/s for readout (TX) and 2.56 Gb/s on the timing channel. The MGT reference clock must be supplied from a 40.08 MHz crystal oscillator. The recovered clock is routed out of the FPGA into a jitter cleaner for MGT reference clock supply to all other MGT instances on the board. There is a fanout in several multiples of the LHC clock frequency.

4.1.5 Controller

The module controller (baseline) is based on an UltraZed board and comprises the following functionality:

- Interface to FELIX
  - Clock recovery and timing data distribution
  - Readout driver functionality
- Ethernet access via Zynq PS
- Ethernet access via IPbus
- Module controller interfacing to
  - Controller internal register set
  - Processor register set
  - Extended board-level environmental monitoring and low level control via I2C / SPI

The controller will run Linux on the Zynq PS and requires mass storage (SD card etc.). At current we assume the module controller, along with a bit of ancillary logic, to sit on a RTM.

4.1.6 Environment monitoring

4.1.7 Power distribution

The power supply scheme is being taken over from the Phase-1 modules, and it consists of these three elements:

- A Power Integrated Module (PIM), which implements the power supply at board level.
- A convertor brick that supplies the payload, converting -48V to 12V.
- Point-Of-Load (POL) regulators, which are partially aggregated on pluggable mezzanines.

Power sequencing and monitoring is achieved with the help of a power controller, with the current baseline model being ADM1066ASUZ. The power supplies are being monitored and controlled via I2C and this information needs to be interfaced to the module controller and/or IPMC.

4.1.8 Thermal management

4.2 Interfaces

4.2.1 Interface with LAr

4.2.2 L0Global interface

The results are transmitted out of the processor FPGAs on 24 MGT links each to 12-way FireFly electrical-optical converters. These converters are grouped together in pairs into two 24-way MTP/MPO connectors on the front panel per module, which sent their signals to L0Global (48 links per module).

4.2.3 FELIX interface and Readout

The fFEX is linked to the FELIX system for the purpose of clock and timing data distribution and readout. The physical interface goes via a single 12-way MTP/MPO, with fibre connection to a 4+4-way bidirectional FireFly module that is electrically linked to the module controller. The module controller is in charge of clock recovery, timing data decoding/distribution, and ROD functionality.

4.2.4 Protocols and data formats

4.3 Firmware specifications

Adrian: TBD which parts go here and which ones go in chapter 5. Let's just start writing everything in chapter 5 and we will think what to move in here later.

insert reference to FPGA description here (avoid duplication of information)

4.3.1 Firmware architecture overview

4.3.2 Input data/signals and format

4.3.3 Infrastructure firmware

4.3.4 Algorithmic firmware

4.3.5 Output data/signals and format

Has to cover both real time data (TOBs to L0Global) as well as the readout data path (xTOBs ...)

4.3.6 Control firmware

ST statements on expected FPGA resource usage here (separate subsection) and/or in Chap. 5 ??

4.3.7 Testing and integration

4.3.8 Firmware development: tools and organisation

4.4 Software specifications

ST do we want to write something about software at all, here and in Chap. 5?

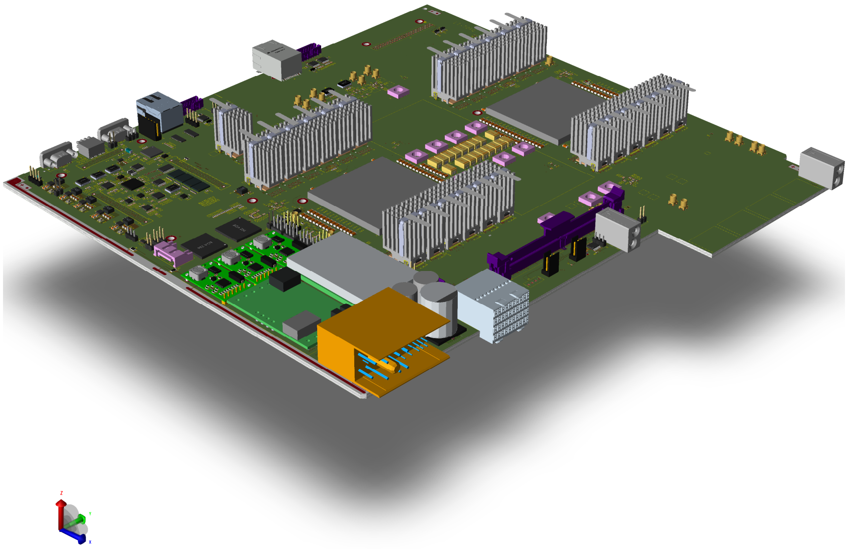

Figure 5.1: Structure of the fFEX Prototype

5. Preliminary Design

5.1 Hardware Implementation

5.1.1 System architecture

The fFEX prototype is designed in an ATCA form factor, however, it foresees the possibility of a standalone operation. The structure of the board can be seen in Figure 5.1.

The main building blocks are the following:

- Two processing FPGAs
- 8 Samtec FireFly modules per FPGA for real-time data path
- 2 Samtec FireFly modules per FPGA for the interface to FELIX
- UltraZed board with Zynq UltraScale+
- IPMC
- Power mezzanines
- DDR4 RAMs

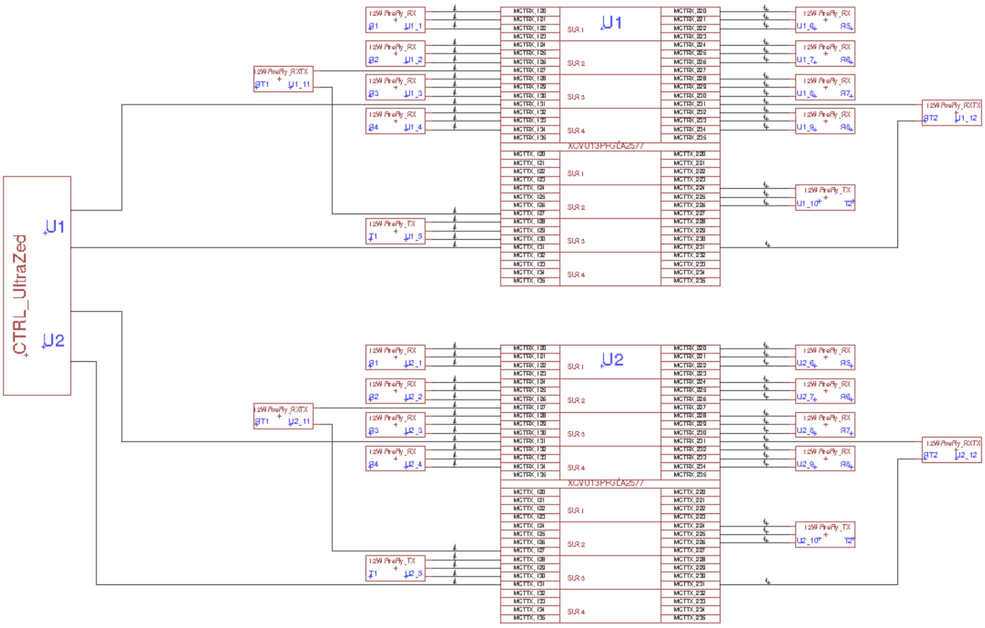

Figure 5.2: MGT connections on the Processor FPGA of the fFEX prototype

5.1.2 Processing FPGAs

5.1.2.1 Block diagram and floor plan

The fFEX prototype includes two FPGAs, so that a representative processing power can be provided and firmware can be developed. The largest FPGA type to be used for the fFEX prototype is the Virtex Ultrascale+ VU13P (`xcvu13pflga2577`).

There are 32 GTY quads available on the Processor FPGA of the fFEX prototype, providing the following connectivity:

- Connection to optical modules for real-time data path (96 RX, 24 TX)
- Connection to FELIX (8 RX, 8 TX)
- UltraZed connection (4 RX, 4 TX)
- IPBus connection: Zynq - MUX / GEP (-) TBC
- Debug (-) TBC
- Spare (20 RX, 92 TX) TBC

A corresponding block diagram can be seen in Figure 5.2.

The complete floor plan of the Processor FPGA can be seen in Figure 5.3. "OM" corresponds to Optical Module connection. Connections to a single optical module are grouped with braces. Connections to the SoC and DDR4 RAMs are shown as well as dedicated IPBus and debug lines. Adrian: description to be updated

5.1.2.2 FPGA configuration

While the processing FPGAs are accessible through their JTAG port at any time, the configuration bitstream required at any power-up is meant to be provided by local storage. The configuration mode used is Master SPI with the processing FPGAs having their own configuration flash memory.

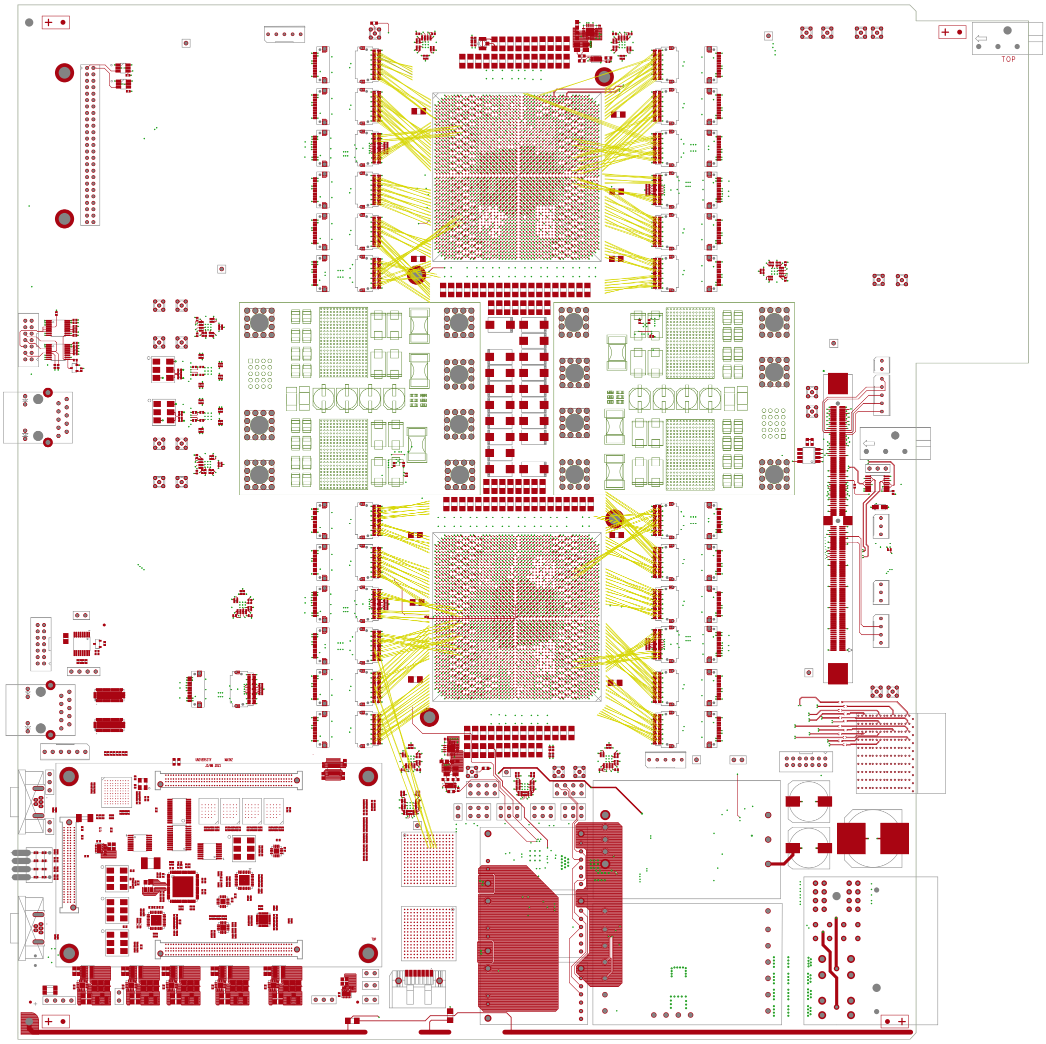

Figure 5.3: The floor plan of the Processor FPGA of the fFEX prototype

The update process can be triggered and controlled from either the control FPGA or via JTAG. The control FPGA itself will in any case be configured from a small SPI flash chip, which due to smaller capacity and rare updates, is assumed to be a rather painless update operation. For the control FPGA, in-situ (live) updates are possible due to the use of a Xilinx-provided fall-back / golden image scheme.

5.1.2.3 FPGA Multiboot

The FPGA MultiBoot and fall-back features are used on the fFEX board to support updating bitstream images dynamically in the field. The FPGA MultiBoot feature enables switching between images on the fly. On this method, the flash memory holds two images:

- Golden image: previously validated image, which includes only basic slow-control functions implemented;
- Multiboot or user image: working image to be used during normal operation. If an error occurs during loading of the MultiBoot image from the upper address space, the fall-back circuitry triggers the golden image to be loaded.

The FPGA MultiBoot feature allows the IPBus communication to the fFEX to be always maintained, in case the user image is corrupted. Additionally, it allows the image to be re-written via IPBus-SPI bridge.

5.1.3 Optical modules

The processor FPGAs interface with input and output fibres via electro-optical transceivers. Three options of the optical modules configurations were considered for the fFEX prototype as potential candidates:

- 8 RX MiniPOD modules (`AFBR-824FH1Z`), 8 TX MiniPOD modules (`AFBR-814FH1Z`).
- 24 bidirectional 2x4 lane $28\,\mathrm{Gb/s}$ FireFly modules (`NR-ECUO-B04-28-025-0-5-1-1-01`)
- 8 bidirectional 2x12 lane $28\,\mathrm{Gb/s}$ BOA modules (`FBOTD25MT3C00`)

Maximum data rate of MiniPODs is $14\,\mathrm{Gb/s}$ per lane. Data rate per lane foreseen for the GCM is up to $25.78125\,\mathrm{Gb/s}$. Therefore, faster optical modules, such as Samtec FireFly or Finisar BOA were considered as the preferred candidates.

As part of an R&D for the Phase-II Global Trigger System, a Technological Demonstrator board has been designed with an FPGA and high-speed optical modules implemented on-board (Figure 5.4). The Demonstrator is designed in a custom ATCA form factor with a number of design blocks which can be evaluated and reused for the GCM. The central part of the board is the Xilinx Virtex Ultrascale+ 13P FPGA, which is connected to two 28G 2x4-lane bidirectional Samtec FireFly modules, one 28G 2x12 bidirectional Finisar BOA module and six MiniPODs.

Performance of the high-speed optical modules and the FPGA has been evaluated with long-run link tests. An Integrated Bit Error Ratio Test (IBERT) loopback test has been performed for the Finisar BOA optical module. In the test 12 transmitter links of the optical module were looped back to 12 receiver links of the same module with a help of a 24 to 2x12-fiber Y-cable and a 12-fiber trunk cable.

During a day-long IBERT test run at 25.65 Gb/s with a 31-bit PRBS pattern used a 1.9E-15 BER has been reached. All 12 links are functional, and no bit errors have been detected. A typical eye diagram, obtained using a low power mode of the GTY receiver, is shown in Figure 5.5. With an open area and an open UI of 6656 and 44.44 % respectively, a good performance of the Finisar BOA optical module is achieved.

Similar tests have been performed with the Samtec FireFly optical module as well. A very good performance of the module has been achieved and no bit errors have been detected. Lower data rates are supported as well in order to be compatible with Phase-I link speed.

5.1.4 Clock system

FELIX provides the LHC clock from the TTC system for the fFEX. The signal is recovered with the help of the control FPGA and distributed to one of the inputs of the jitter cleaner chip `Si5345` placed on the main board. A dedicated local clock oscillator is used for the recovery of the system clock. A chain of fanout chips takes care of the recovered system clock distribution to the processor FPGA as well as back to the control FPGA, where the system clock is needed for transmitting data and BUSY signals back to FELIX.

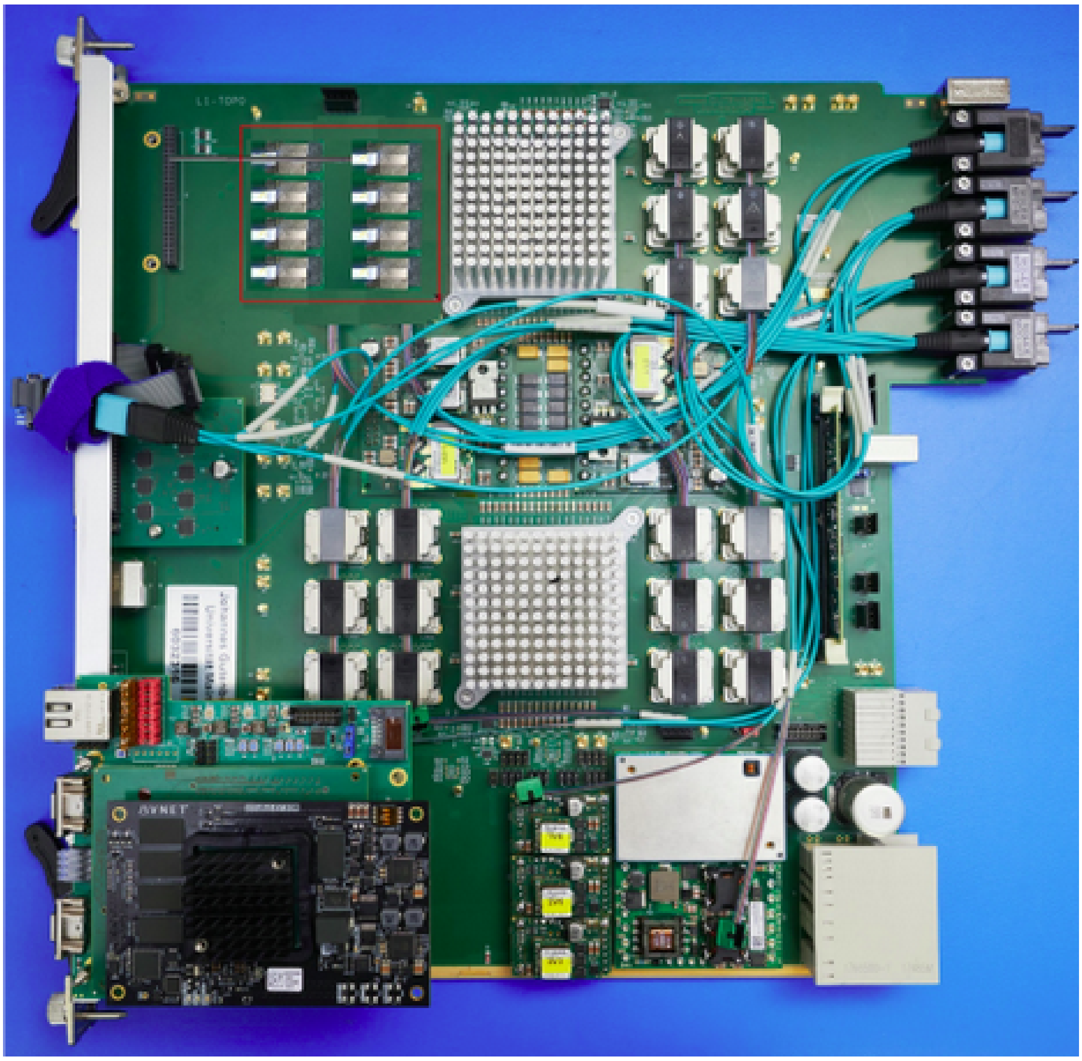

Figure 5.4: Global Trigger Technological Demonstrator hardware overview

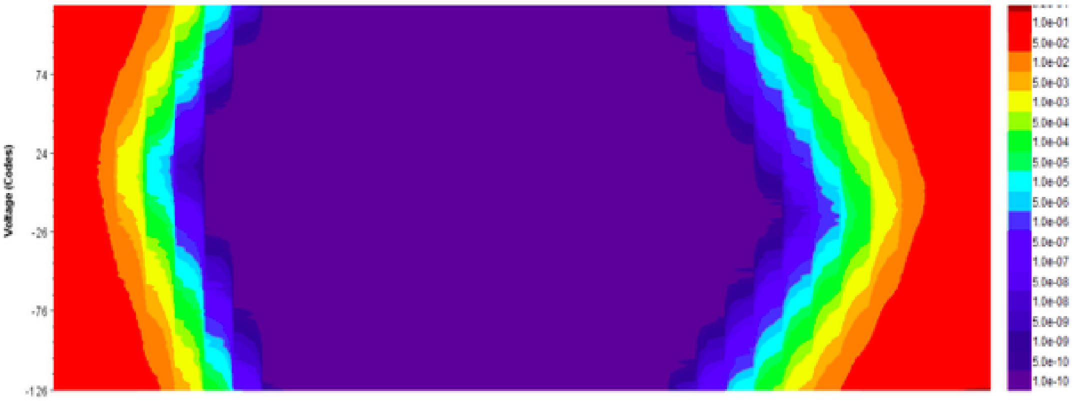

Figure 5.5: Finisar BOA IBERT loopback test: a typical eye diagram

Adrian: this part about the recovered clock needs changes => The default input of the `Si5345` uses the recovered system clock, being all outputs derived from it. Additionally, one local crystal and two SMA

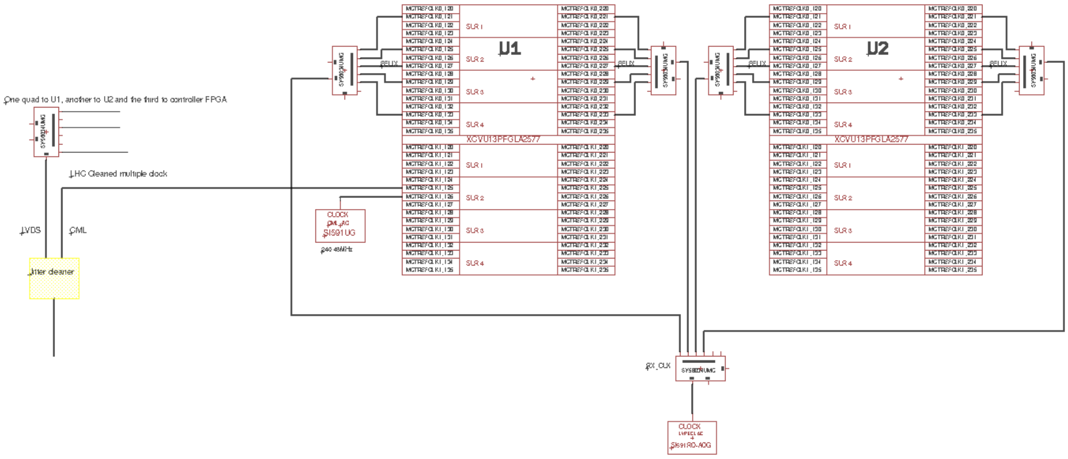

Figure 5.6: Clocking of the fFEX prototype (Adrian: to be completed)

connectors (differential input) are connected to the remaining inputs of the jitter cleaner. A backplane (Zone 2) clock connection is available as well. It is present for hardware test and debug purposes only at the current prorotype stage of the fFEX development. It is not required for the firmware development.

From its inputs the PLL chip can generate clocks of various multiples of the input frequencies. This flexibility allows the multi-Gb/s links on the fFEX prototype to be driven at a large range of different rates. The `Si5345` settings are accessible via I2C.

Various reference clocks are implemented on the fFEX prototype:

- A 156.25 MHz crystal clock
- A 240.48 MHz crystal clock
- A jitter-cleaned multiple of the LHC clock
- An additional jitter-cleaned multiple of the LHC clock as a spare
- A global clock, jitter-cleaned, 40.08 MHz
- A global crystal clock

The global clock tree is present as well in order to synchronize programmable logic between the processor and the control FPGA. Two dedicated IPBus reference clock trees are also implemented. A corresponding block diagram can be seen in Figure 5.6.

The MGTs reference clock scheme is designed to minimize the number of signals routed on the PCB. Additionally, two constraints should be respected:

- Based on the the GTY user guide the reference clocks for a QUAD can also be sourced from up to one QUAD below or above for data rates between 16.375 Gb/s and 28.21 Gb/s. No crossing of clocking signals between different SLRs is allowed. Therefore we can use the top 3 quads of each SLR and each side for one optical device and supply only the central one of those with the respective reference clock. The remaining fourth quad should be supplied from a separate chain.
- The IPbus slow control can either run from an on-board 125 MHz crystal clock, or from an output of the `Si5345` chip. This allows the fFEX prototype module control function over IPBus to be independent or not from the system clock.

5.1.5 Controller

The control block provides many of the (non-real-time) services required on the fFEX prototype. It hosts mainly module control, clock/control and configuration circuitry. It also provides initialization circuitry for the FPGA and acts as an interface to environmental monitoring devices. It is based on an UltraZed board located in the RTM.

The functionalities of the module controller are:

- Hosts the control FPGA of the fFEX prototype
- Hosts the MasterSPI configuration circuitry for processor and control FPGAs
- Clock cleaning and distribution
- Monitoring and slow control
- IPbus master
- Interface to FELIX (Incl. TTC clock recovery)

The "intelligent" module controller is an FPGA which handles incoming IPbus requests and forwards the data and control packets to the processor via MGT links.

The configuration of the processor FPGA is controlled from the control block. To that end all signal lines required are routed to that block. The configuration mode used is Master SPI. Special care is taken once routing the CCLK signal on the fFEX prototype as it is considered as critical by Xilinx.

Environmental data (voltages, currents, temperatures) are collected on the board by I2C based sensors, and routed to the control block via the bidirectional I2C buses. Parameters in the respective devices are set in the same way. Data are originating from dedicated monitoring chips, or from monitor/control interfaces available in core functionality devices, e.g. FireFlies. They are routed into the control FPGA with an optional breakout onto headers. The control FPGA allows for access to these data via IPbus. The status/control data exchanged that way are complementary to the IPMC data.

The IPBus communicates with its control PC(s) via an Ethernet Phy chip. The chip type chosen is VSC8221. It is an electrical Ethernet (1000BASE-T) to SGMII device. The SGMII link is connected to an MGT link of the control FPGA. The 1000BASE-T port is located on the front panel. This link is AC-coupled with series capacitors. Magnetics (transformers) are not required due to the choice of Phy chip, which is specifically designed (voltage mode drivers, internal biasing) to support magnetics-free links.

Additionally, the control FPGA processes the input data and clock from the TTC system via FELIX and the output data to FELIX. The optical interface to FELIX is implemented with the use of two bidirectional Samtec FireFly modules.

In addition to the internal Random Access Memory (RAM) the control FPGA is interfaced with large RAM storage to allow for long-latency buffering of the input and output data.

ZYNQ Ultrascale+ XCZU719EG-1FFVD1760E device is used as the control FPGA. The control block does not populate a hybrid SoC on its own, instead it is interfaced through a daughter system on a module (SoM) from Avnet called UltraZed-EV SOM, based on a ZYNQ device. The UltraZed is plugged onto the main board.

5.1.6 Environmental monitoring

The IPMC module is a standard component in ATCA boards for hot swap power management and sensor monitoring. The fFEX prototype uses the CERN IPMC which follows the functions defined by the ATCA rev. 3.0 standard. It can be accessed through an Ethernet interface, a serial interface or JTAG.

The hot swap functionality is controlled by a handle switch which is located in the front on the underside of the board. The position is adjusted such that locking the board into an ATCA shelf into its correct position with side clamps will press the handle switch and trigger the power-up of all power regulators through a signal on the `ATCA_PWR_CTRL` lines. The IPMC hot swap functionality can also be overridden by a jumper which allows direct power-up of the board.

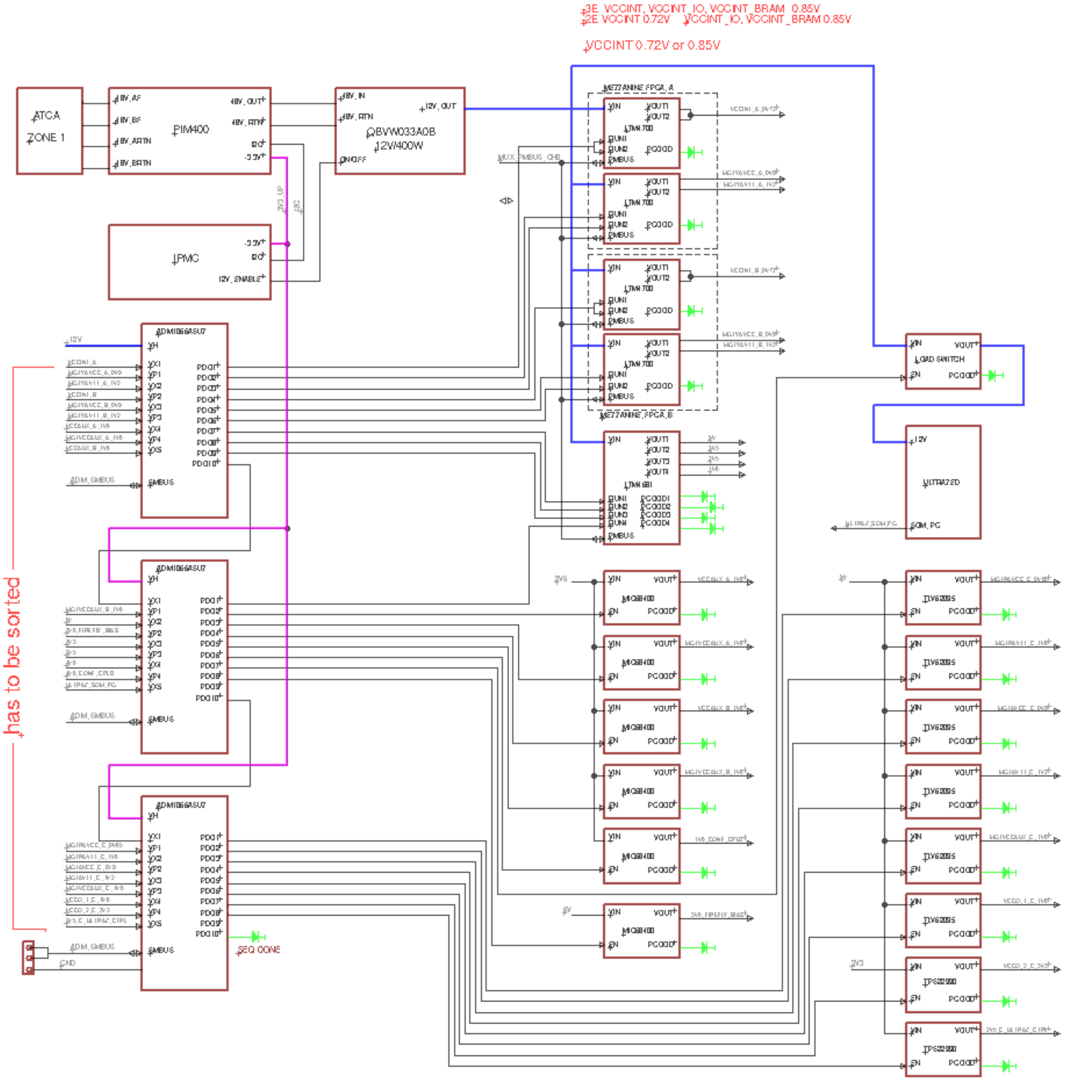

Figure 5.7: Main power circuitry of the fFEX prototype

596 5.1.7 Power distribution

597 5.1.7.1 Power management in ACTA shelf

598 The setup of the board is the same as for the standalone verification. Inserting the board into a running shelf 599 will connect the 48V power connector and locking the board in position will activate the other power regulators. 600 With the shelf manager the board can be switched on and off now. For standalone operation the activation of 601 the power regulators through the handle switch can be overridden.

602 5.1.7.2 Power management on fFEX Board

603 Power is supplied to the fFEX prototype on dual, redundant -48V DC feed through the ATCA Zone 1 connector. 604 A PIM4000 converter accepts this feed, providing: 3.3 V supply for the hardware management (3V3-UP); 48 605 V (monitored) power supply used by the isolated DC/DC converter QBVW033A0B to generate a 12 V output. 606 This 12 V supply is stepped down further, by multiple non-isolated DC/DC power modules, mainly placed on 607 mezzanines, to supply the multiplicity of voltages required on the fFEX prototype board (Figure 5.7).

608 On power-up of the payload hardware, the sequence and timing with which the multiple power rails are

turned on is controlled by a CPLD based on the Power Good signals of each DC/DC converter. First supplies to be powered up are the board wide available 2.5 V and 3.3 V. The sequencing of the separate FPGA input voltages, according to the Xilinx specifications is:

- VCCINT, VCCINTIO, MGTAVCC;
- MGTAVTT, VCCAUX, MGTVCCAUX, VCCAUXIO;
- VCCIO;

According to the Virtex UltraScale+ FPGA Data Sheet, the operating voltages of a device with speed grade -1 are:

- VCCINT & VCCINTIO: 0.85 V;
- MGTAVCC: 0.90 V;
- MGTAVTT: 1.20 V;
- VCCAUX, MGTVCCAUX, VCCAUXIO & VCCIO: 1.80 V

Table 5.1 shows the power estimation for FPGA voltage rails on the fFEX prototype. All assumptions on power requirements are derived from the Xilinx Power Estimator spreadsheets.

Name	Voltage	Current Estimation
VCCINT	0.85 V	46 A
MGTAVCC	0.9 V	12.3 A
MGTAVTT	1.2 V	28 A
VCCAUX	1.8 V	1.1 A
MGTVCCAUX	1.8 V	1.2 A
VCCIO	1.8 V	0.3 A

Table 5.1: fFEX prototype power estimation for FPGA voltage rails – NUMBERS TO BE UPDATED

5.1.7.3 Power mezzanines

The power supplies for the processor FPGA are mainly located on mezzanines. This approach allows a careful evaluation of ripple noise and stability of each design, before connecting the mezzanines to the fFEX prototype and powering up the board.

The fFEX prototype Power Mezzanines are based on the TDK-Lambda iJX series of non-isolated DC/DC converters. Three main factors were taken into account once making this selection:

- High current consumption requirement by the Virtex Ultrascale+;
- The MGTs on the Virtex Ultrascale+ are very sensitive to noise in its power rails (MGTAVCC, MGTAVTT and MGTVCCAUX) requiring a maximum 10 mVpp of noise on the FPGA power pins over the band from 10 kHz to 80 MHz;
- Out-of-the-box monitoring tool for output voltages, currents and temperatures (on-board PMBus monitoring).

For the VCCINT, the iJB (max. 60 A output current) is used, while MGTAVCC, MGTAVTT and VCCIO uses the iJA (max. 35 A output current). The iJA DC/DC converter is also used to supply the board level voltages 3.3 V and 2.5 V. The FPGA auxiliary voltages (VCCAUX and MGTVCCAUX) supplies, are an exception on this mezzanine approach. As they have a much lower power consumption, the linear regulators MIC68400 are used being placed directly on the fFEX prototype and not on mezzanines.

Two different mezzanine types are used for the fFEX prototype: the FPGA Power Mezzanine equipped with two iJA for MGTAVCC & MGTAVTT and one iJB for VCCINT and the Board Level Voltage Mezzanine equipped with one iJB, for VCCIO, 3.3 V and 2.5 V. A third mezzanine called Power Panel is used to make the interface between the FPGA Power Mezzanine and the main board, being it a passive board. The Board Level Voltage Mezzanines are directly connected to the fFEX prototype.

The processor FPGA is supplied by a single FPGA Power Mezzanine and two separate MIC68400. The VCCIO is supplied by a single Board Level Voltage Mezzanine, so as the 3V3 and 2V5. In total the fFEX prototype board hosts one FPGA Power Mezzanine and three Board Level Voltage Mezzanines.

For safety, apart from the over-voltage/current protection mechanism intrinsic to the iJX DC/DC regulators, the fFEX prototype Power Mezzanines also include an over-voltage protection circuit (crowbar), controlled by LTC1696 chip.

5.1.8 Thermal management

In order to allow for proper cooling, a heat sink is pasted on the processor and the internal FPGA temperature is always monitored.

The fFEX system can be hosted in a standard ATCA shelf that uses vertical flow with water chillers sandwiched between shelves. The cooling capacity per such a shelf is 450W: 400W in the front space and 50W on the backplane.

5.2 Interfaces

5.2.1 Interface with LAr

5.2.2 L0Global interface

5.2.3 FELIX interface and Readout

FELIX provides the input data and clock from the TTC system for the processor FPGA. The readout output from the processor FPGA is interfaced to FELIX as well. The routing of the input data and clock from the TTC system as well as the routing of the readout output from the processor FPGA is implemented via the control FPGA, connected between FELIX and the processor FPGA. The optical interface to FELIX is implemented with the use of one bidirectional FireFly module. The lpGBT protocol, used for TTC distribution from FELIX, is supported with all the main hardware requirements fulfilled: data rates up to 10.24 Gb/s, 40 MHz reference clock, optical input and output links.

5.2.4 Protocols and data formats

5.3 Firmware Implementation

5.3.1 Firmware architecture overview

insert reference to FPGA description here (avoid duplication of information)

5.3.1.1 Calorimeter coverage

5.3.2 Input data/signals and format

5.3.3 Infrastructure firmware

The basic low-level infrastructure firmware block handles the distribution of reference clocks, operation of the optical links, data decoding and serialisation/deserialisation of either the data from the preceding layer or the data to the proceeding layer. It has defined interfaces to the functional block of the firmware allowing for a modular and independent development of the actual functionality. Initially only a dummy algorithm is provided that demonstrates how the low-level frame interacts with the functional block.

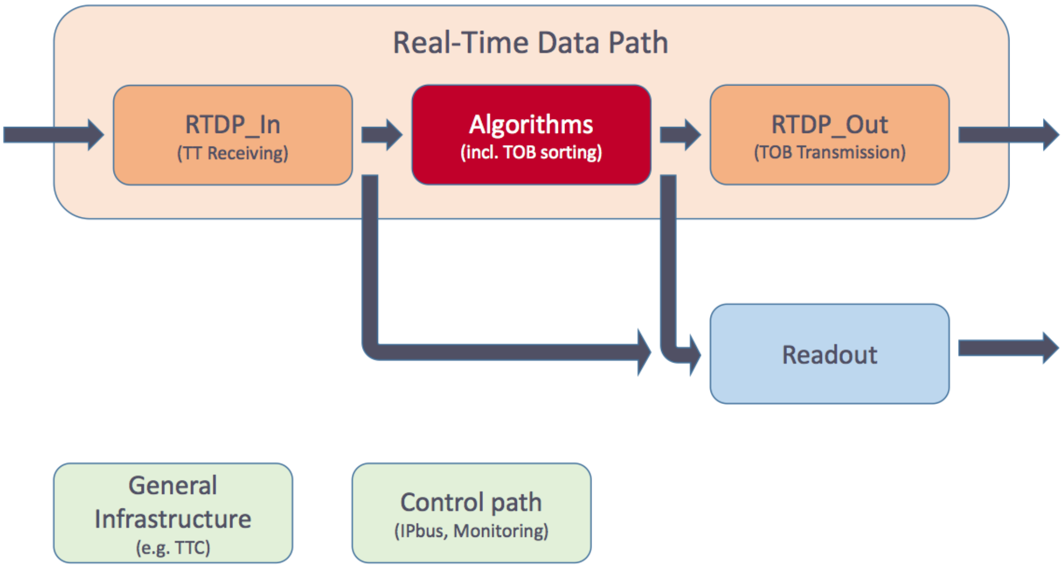

Figure 5.8: Overview of the firmware components for the realtime data path - PLACEHOLDER - taken from jFEX PRR - NEEDS TO BE REPLACED

.

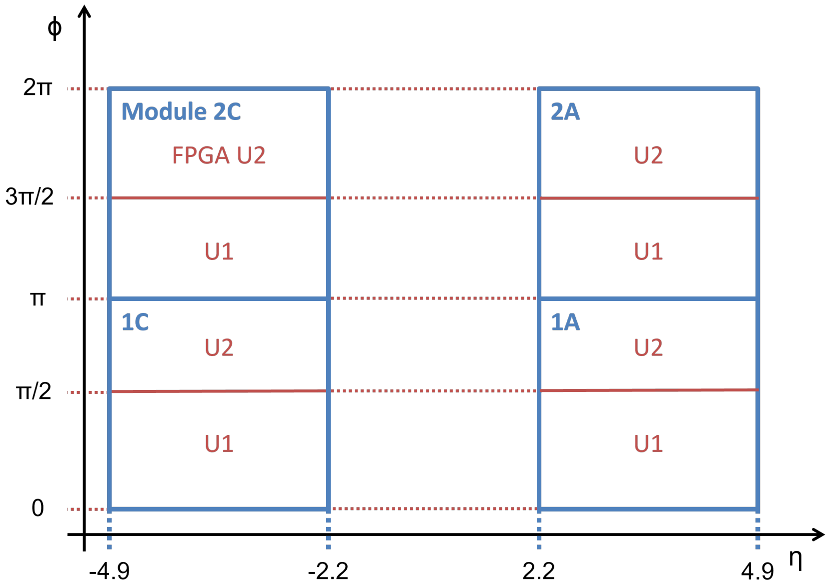

Figure 5.9: Calorimeter coverage of the fFEX modules (blue boxes) and their respective FPGAs (red separation lines).

.

5.3.4 Algorithmic firmware

5.3.5 Functional description

This section provides a detailed description of the individual modules in the fFEX firmware following the block diagram in 5.11.

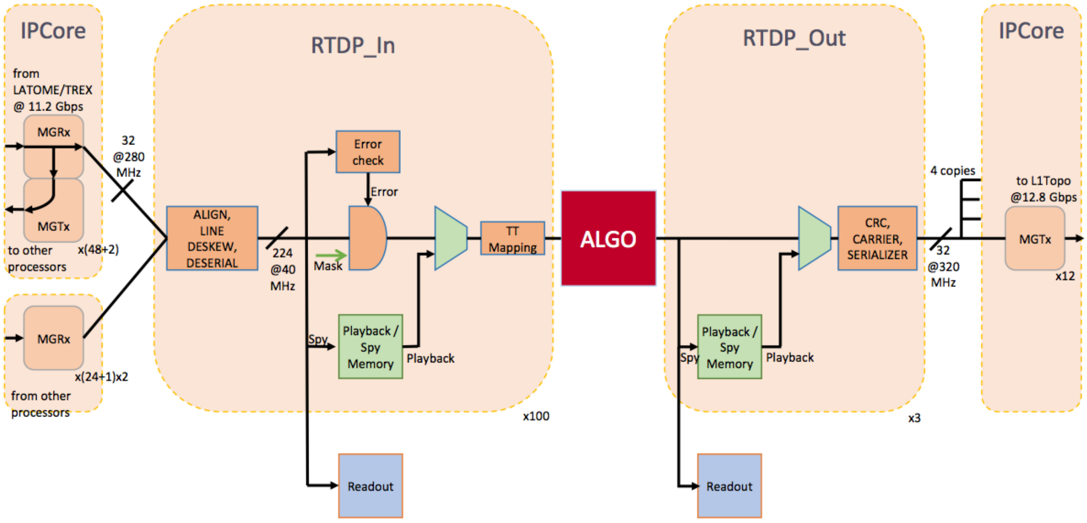

Figure 5.10: Overview of the infrastructure firmware components for the realtime data path - PLACEHOLDER - taken from jFEX PRR - NEEDS TO BE REPLACED

.

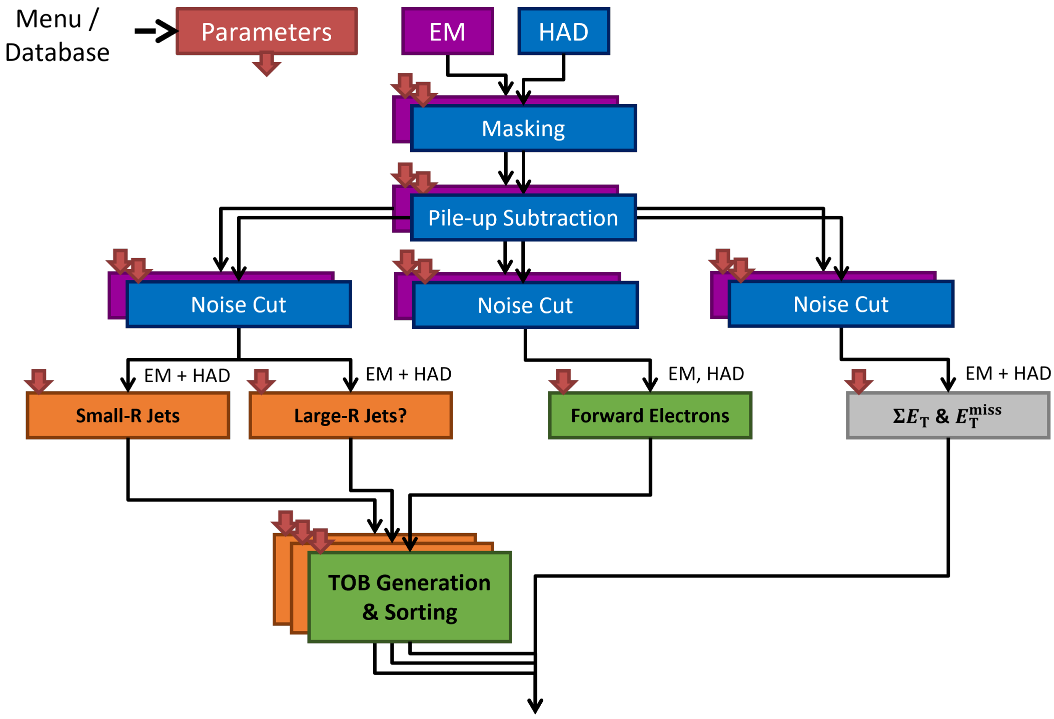

Figure 5.11: Top-level block diagram of the fFEX algorithm firmware.

.

5.3.5.1 Forward jet algorithms

The fFEX jet algorithms will benefit from already existing code in the jFEX firmware repository. The big difference lies in the finer input granularity, especially from the FCal elements, where the cells are not equally spaced. The list-based implementation that was originally developed for jFEX enables a smooth transition across regions with different and/or irregular granularity. These lists are generated with the help of a Python

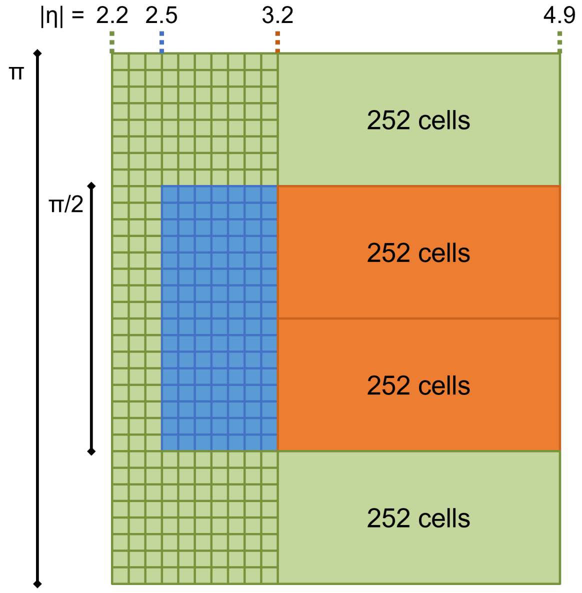

Figure 5.12: EM environment/granularity of a single processor FPGA ($\eta \leftrightarrow$, $\varphi \updownarrow$). The blue/orange area marks the core region, while the green area represents the overlap with neighboring FPGAs. EM data are received from 3-4 layers in $2.2 < |\eta| < 2.5$, from 2 layers in $2.5 < |\eta| < 3.2$ and from 1 layer beyond.

.

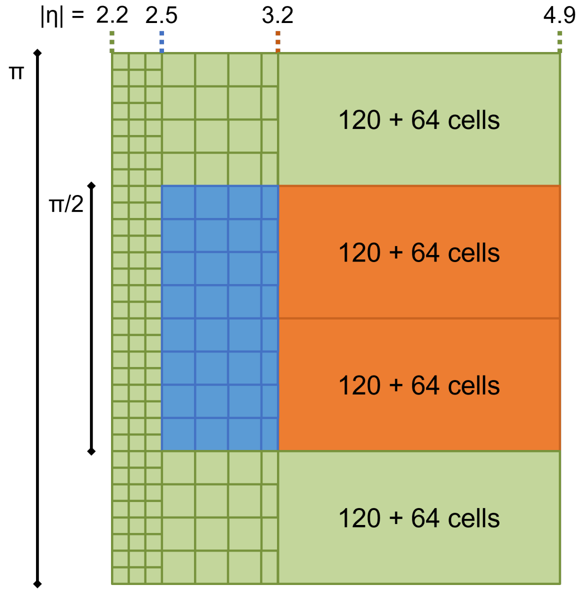

Figure 5.13: HAD environment/granularity of a single processor FPGA ($\eta \leftrightarrow$, $\varphi \updownarrow$). The blue/orange area marks the core region, while the green area represents the overlap with neighboring FPGAs. HAD data are received from 4 layers in $2.2 < |\eta| < 2.5$, from 4 layers in $2.5 < |\eta| < 3.2$ and from 2 layer beyond.

.

script taking advantage of floating point precision. They provide information about the number of sums to be calculated as well as the length of each sum itself. Thus, each list represents a kind of construction manual for the synthesis tool. Changes to a list only requires a new synthesis run, but no change to the firmware code.

The jet algorithms are based on the *sliding window* approach, which can be highly parallelized:

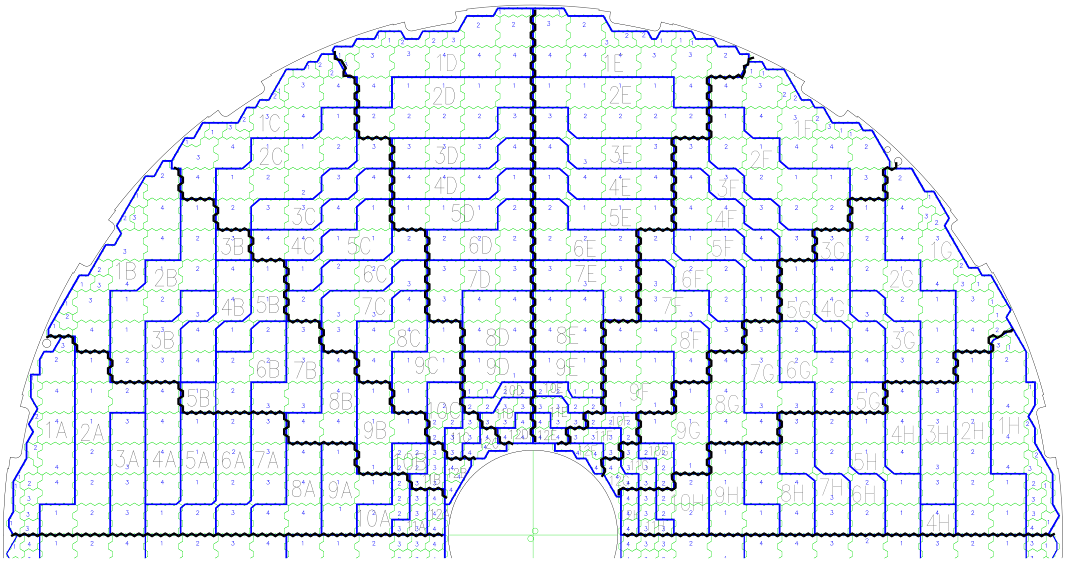

Figure 5.14: Geometry of FCal layer 1 (EM layer) for $\Delta\varphi = \pi$ in the $x-y$ plane ($+x \leftarrow$, $+y \uparrow$, $+z \oplus$). The distance of SCs in φ is approximately 0.4 (black separation lines). Blue lines form individual SCs along η, while green lines separate underlying cells.

.

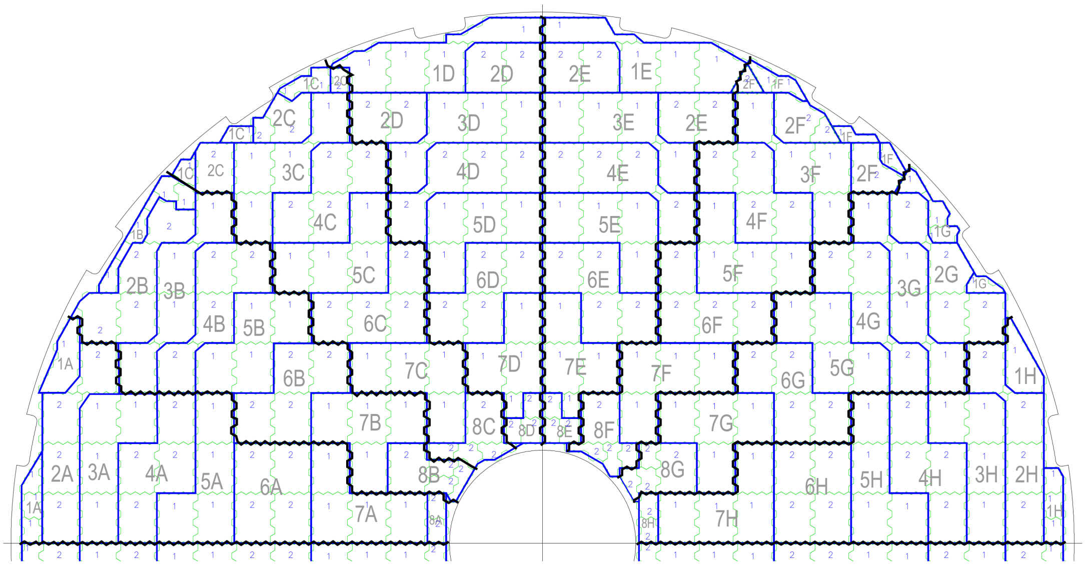

Figure 5.15: Geometry of FCal layer 2 (1st HAD layer) for $\Delta\varphi = \pi$ in the $x-y$ plane ($+x \leftarrow$, $+y \uparrow$, $+z \oplus$). The distance of SCs in φ is approximately 0.4 (black separation lines). Blue lines form individual SCs along η, while green lines separate underlying cells.

.

Identification of Local Maxima (Seeding): The geometric center of each cell that is in the finest layer and within the core region of an FPGA is considered to be an RoI candidate. The sum of all cells within $\Delta R < 0.2$ around a given center, referred to as *seed*, is tested for a local maximum. Neighboring (and overlapping) seeds within a *search window* of $\Delta R < 0.3$ around the seed being tested are compared. To resolve ambiguities in case of equal energies, a mixture of 'greater than' and 'greater than or equal' conditions is used according to

jFEX parameter - MW

jFEX parameter - MW

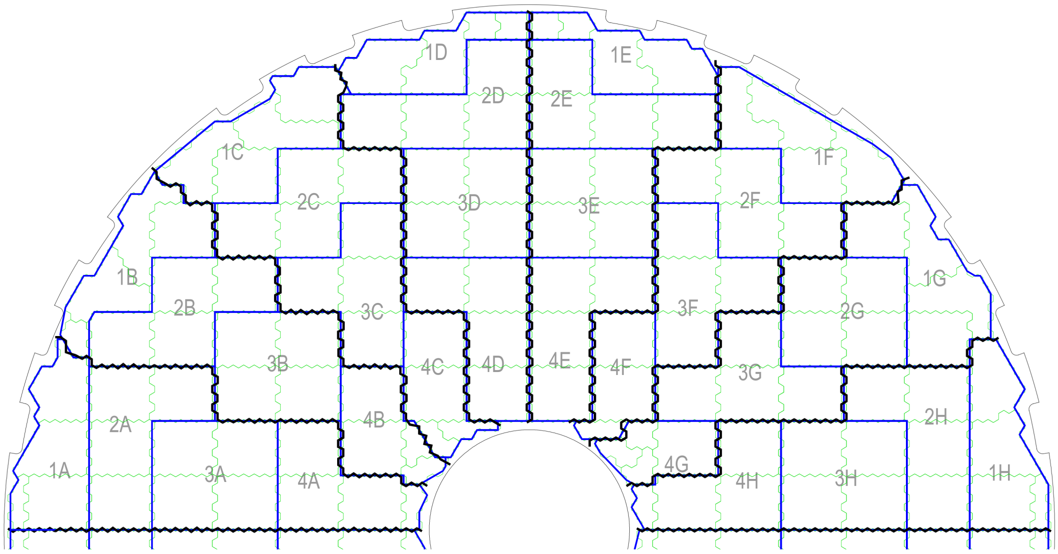

Figure 5.16: Geometry of FCal layer 3 (2nd HAD layer) for $\Delta\varphi = \pi$ in the $x-y$ plane ($+x \leftarrow$, $+y \uparrow$, $+z\oplus$). The distance of SCs in φ is approximately 0.4 (black separation lines). Blue lines form individual SCs along η, while green lines separate underlying cells.

.

$$\text{cond.} = \begin{cases} \geq & \text{if} \quad (\Delta\eta + \Delta\varphi < 0) \vee \left[(\Delta\eta + \Delta\varphi = 0) \wedge (\Delta\eta < 0)\right] \\ > & \text{else} \end{cases}, \tag{5.1}$$

where $\Delta\varphi$ and $\Delta\eta$ are calculated w.r.t. φ_{RoI} and η_{RoI}, respectively. Only the geometric center of a given cell defines whether it is part of a sum.

Jet Energy Summation (Clustering): The jet cluster energy is the sum of all cells within $\Delta R < 0.4$.

jFEX parameter - MW

5.3.5.2 Forward EM algorithms

The EM algorithm for the fFEX will share similar design principles as the jet algorithm and will be based upon a modified *sliding window* approach.

Identification of Local Maxima (Seeding): Local maxima are identified in the same way as they are for jets.

- $\Delta R(\eta,\phi) < 0.1$ for EM core in EMEC/HEC, $\Delta R(\eta,\phi) < 0.2$ for isolation
- $\Delta R(x,y) < 6cmx5.2cm$ (2 cells) for EM core in FCAL, $\Delta R(x,y) < 12cmx10.4cm$ for isolation (seen slightly better performance compared to using eta x phi).
- Handling overlap between EMEC/FCAL. So far: calculate local maxima separately, then check local maxima between EMEC/FCAL + give priority to EMEC.

Consider utilizing smaller cell sizes towards higher eta - JB

Electron energy and shape:

- Core energy = Sum over 2x2 cells in EM depth
- EM Isolation energy = Sum over 4x4 ring of cells in EM depth
- HAD Isolation energy = Sum over 4x4 cells in HAD depth
- Promising shower shape variables: Fraction of max. energy cell and depth of shower barycentre

Need to specify whether to use full or split HEC cells, so far split up to match EMEC window. - JB

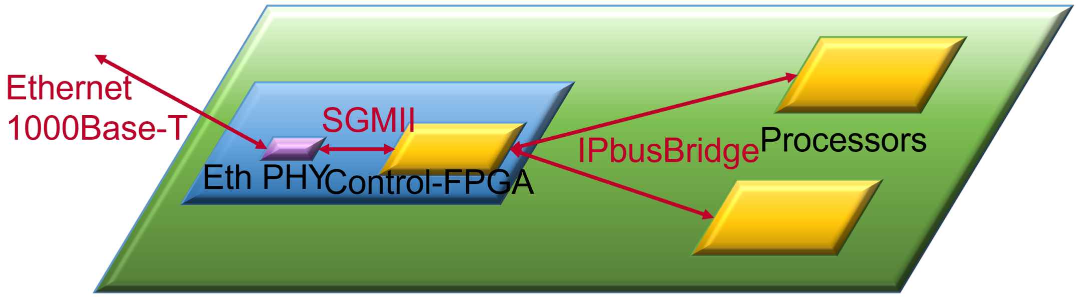

Figure 5.17: Overview of the control firmware components for the real-time data path.

.

5.3.6 Output data/signals and format

5.3.6.1 Saturation and overflows

All calculations within fFEX are performed at best (available) energy resolution and by extending bit lengths to avoid (uncontrolled) arithmetic overflows. When generating TOBs, energy values have to be truncated; if an overflow occurs during truncating, the corresponding energy value is set to full-scale (= saturation).

Naive assumption, similar to jFEX - MW

5.3.7 Control firmware

The firmware for the command/control FPGA controls the board and processing FPGA and monitoring (via IPBus), readout data, TTC command and clock forwarding with firmware blocks that match the sending and receiving blocks in the processing FPGA. The initially available low-level block concerns reference clocks, I/O interfaces and general infrastructure, inherited from the Phase-I jFEX/Topo control firmware, as described in Sec. 5.3.9.1.

5.3.8 Testing and integration

5.3.9 Firmware development: tools and organisation

5.3.9.1 Firmware reuse

The fFEX module reuses to a large extent the Phase-I jFEX/Topo infrastructure firmware. This firmware is designed to be as generic as possible, which means, that the settings for the MGTs, number of channels/quads/SLRs, IPbus register map, etc. are taken from the database to configure the firmware. TCL scripts generate required VHDL code, when third-party code (IPCores) don't provide uniform interfaces needed for generic usage.

The fFEX hosts a 13P device in an A2577 package as a single processor FPGA. Thus, the firmware reuse needs to consider a migration from a 9P to a 13P device. However, to re-configure the common jFEX/Topo infrastructure firmware to a 13P device would just require other database settings, but no further firmware development - since the same clocking scheme (only one system-wide clock domain, 32-bit data buses to/from the MGTs) is used and the FPGA is from the same FPGA family (Ultrascale+ series with GTY MGTs).

Finisar BOA optical modules are used on the fFEX prototype, while MiniPODs are hosted on the jFEX/Topo modules. Nevertheless, the kind of optical modules connected to the FPGA is not relevant for the firmware. The only case which would make firmware changes and probably another synchronization scheme necessary is when wider data buses to/from the MGTs are required by the line rate.

IPBus support is implemented in the fFEX hardware in the same way as it is done on the jFEX. The same UltraZed module is used to implement the control part. Thus, no firmware changes are required and the IPbus master is placed in the control FPGA and its IPbus bus is bridged via MGT link to the processor.

The biggest part of the FPGA constraints concerns the MGT and MGT refclk pins. These are generated by TCL scripts, so manual work on the constraints is only needed for special pins like system clock or parallel-IO pins.

The number of SLRs can be adjusted via the configuration database.

The spy memories support is generic and, thus, is reused within the fFEX firmware. Both spy memory operation modes are supported.

5.4 Software Implementation

At this stage the software has some basic functionality to configure the FPGAs and control the fFEX module in standalone via IPBus. It will be based on available software libaries for IPBus communication. A register map that can be updated with the progressing firmware is used as an abstraction layer between the actual registers and the functionality in the software.

The control part of the software includes methods to set the different operating modes of the processing and control FPGA, e.g. which are the sources or sinks, either modules connected to the optical modules or spy memories. Reading and writing from/to the spy memories is also supported with the basic software. Some basic tools are also provided that help to compare expected bit patterns with recorded.

The software can evolve with the development of the fFEX firmware. A complete simulation framework with defined input and output data formats and implemented algorithms can make use of the provided tools to operate the spy memories, and the algorithms can be debugged and developed. Finally, the software can be integrated into the phase-II TDAQ software to allow for operation with other modules in the partition. In addition, online simulation and testing with predefined patterns is possible for quick or extensive firmware validation.