Documentation

ATLAS Level-1 Calorimeter Trigger Upgrade

Jet Feature Extractor (jFEX)   
Prototype

**Sebastian Artz, Stefan Rave, Elena Rocco, Ulrich Schäfer,**

**Version: 1.01**

**July 2016**

**Based on the 7-module mapping w/o extended environment**

Contents

[1 Related Documents 3](#_Toc456094672)

[2 Conventions 3](#_Toc456094675)

[3 Introduction 4](#_Toc456094676)

[3.1 Overview of Phase-1 System 5](#_Toc456094679)

[3.2 Overview of Phase-2 system 6](#_Toc456094680)

[4 Functionality 7](#_Toc456094681)

[4.1 Real-Time Data Path 8](#_Toc456094683)

[4.1.1 Input Data Granularity 8](#_Toc456094684)

[4.1.2 Feature Identification Algorithms 8](#_Toc456094685)

[4.1.3 Processing Area 10](#_Toc456094686)

[4.1.4 Output Bandwidth 12](#_Toc456094692)

[4.2 Readout Data Path 13](#_Toc456094693)

[4.3 Latency 15](#_Toc456094699)

[4.4 Error Handling 15](#_Toc456094700)

[4.5 Interface to TTC 16](#_Toc456094701)

[4.6 Slow Control 16](#_Toc456094702)

[4.7 Environment Monitoring 17](#_Toc456094703)

[4.8 Commissioning and Diagnostic Facilities 17](#_Toc456094704)

[4.9 ATCA form factor 17](#_Toc456094707)

[5 Data formats 18](#_Toc456094708)

[Input data 18](#_Toc456094709)

[5.1 18](#_Toc456094710)

[5.1.1 Calorimeter Data Format 18](#_Toc456094711)

[5.2 Real-Time Output Data 19](#_Toc456094712)

[Readout data 21](#_Toc456094713)

[5.3 21](#_Toc456094714)

[6 Implementation 22](#_Toc456094717)

[6.1 Input Data Reception and Fan Out 22](#_Toc456094718)

[6.2 Processor FPGA 23](#_Toc456094719)

[6.2.1 General-purpose I/O 24](#_Toc456094720)

[6.2.2 Readout 25](#_Toc456094721)

[6.2.3 Control 26](#_Toc456094722)

[6.3 Clocking 27](#_Toc456094723)

[6.3.1 Global clock (GCK\_CLK) 28](#_Toc456094724)

[6.3.2 Input clock (CLK\_MGT1) 29](#_Toc456094725)

[6.3.3 Output clock (CLK\_MGT2) 29](#_Toc456094726)

[6.3.4 ROD clock (CLK\_MGT3) 30](#_Toc456094727)

[6.3.5 IPBus clock (CLK\_MGT4) 30](#_Toc456094728)

[6.3.6 TTC command clock (EXT\_TTC\_CLK) 30](#_Toc456094729)

[6.3.7 Configuration clock (EMCCLK) 31](#_Toc456094730)

[6.3.8 FELIX clock (MGT5\_CLK) 31](#_Toc456094731)

[6.3.9 CPLDs 31](#_Toc456094732)

[6.4 FPGA configuration 31](#_Toc456094737)

[6.5 The IPM Controller 32](#_Toc456094738)

[6.6 Power Management 32](#_Toc456094739)

[6.7 Front-panel Inputs and Outputs 33](#_Toc456094740)

[6.8 Rear-panel Inputs and Outputs 34](#_Toc456094741)

[6.8.1 ATCA Zone 1 34](#_Toc456094743)

[6.8.2 ATCA Zone 2 34](#_Toc456094744)

[6.8.3 ATCA Zone 3 35](#_Toc456094745)

[6.9 LEDs 35](#_Toc456094746)

[6.10 Instrument Access Points 36](#_Toc456094747)

[6.10.1 Set-Up and Control Points 36](#_Toc456094748)

[6.10.2 Signal Test Points 36](#_Toc456094749)

[6.10.3 Ground Points (TODO: ask Bruno, update) 36](#_Toc456094750)

[7 Front-Panel Layout (TODO: update) 37](#_Toc456094754)

[8 Programming model 37](#_Toc456094755)

[8.1 Guidelines 37](#_Toc456094756)

[8.2 Register Map (TODO: PDR status) 37](#_Toc456094757)

[8.3 Register Descriptions 38](#_Toc456094758)

[9 Glossary 38](#_Toc456094759)

[10 Document History 40](#_Toc456094761)

# Related Documents

1. ATLAS TDAQ System Phase-I Upgrade Technical Design Report, CERN‑LHCC‑2013‑018, <http://cds.cern.ch/record/1602235/files/ATLAS-TDR-023.pdf>
2. L1Calo Phase-I Hub Specification *(not yet available)*
3. L1Calo Phase-I ROD specification *(<https://twiki.cern.ch/twiki/pub/Atlas/LevelOneCaloUpgradeModules/Hub-ROD_spec_v0_9.pdf>)*
4. L1Calo Phase-I eFEX Specification *(<https://twiki.cern.ch/twiki/pub/Atlas/LevelOneCaloUpgradeModules/eFEX_spec_v0.2.pdf>)*
5. L1Calo Phase-I Optical plant Specification *(not yet available)*
6. ATCA Short Form Specification, <http://www.picmg.org/pdf/picmg_3_0_shortform.pdf>
7. PICMG 3.0 Revision 3.0 AdvancedTCA Base Specification, *access controlled*, <http://www.picmg.com/>
8. L1Calo High-Speed Demonstrator report *(<https://twiki.cern.ch/twiki/pub/Atlas/LevelOneCaloUpgradeModules/HSD_report_v1.02.pdf>)*
9. Development of an ATCA IPMI controller mezzanine board to be used in the ATCA developments for the ATLAS Liquid Argon upgrade, http://cds.cern.ch/record/1395495/files/ATL-LARG-PROC-2011-008.pdf

# Conventions

The following conventions are used in this document.

A programmable parameter is defined as one that can be altered by slow control, for example, between runs, not on an event by event basis. Changing such a parameter does not require a re-configuration of any firmware.

Where multiple options are given for a link speed, for example, the readout links of the jFEX are specified as running up to 10 Gb/s, this indicates that the link speed has not yet been fully defined. Once it is defined, that link will use a single speed. All links on the jFEX will run at a fixed speed in the final system.

In accordance with the ATCA convention, a crate of electronics here is referred to as a shelf.

Where the term jFEX is used here, without qualification, it refers to the jFEX module. The jFEX subsystem is always referred to explicitly by that term.

# Introduction

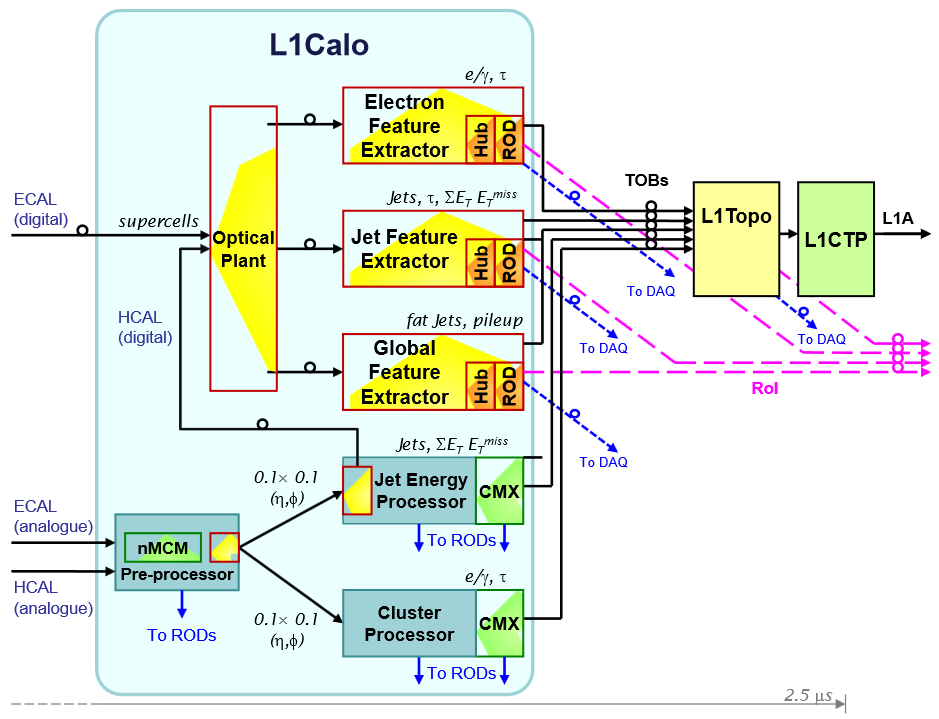
This document describes the jet Feature Extractor (jFEX) module of the ATLAS Level‑1 Calorimeter Trigger Processor (L1Calo) [1.1] . The jFEX is one of several modules being designed to upgrade L1Calo, providing the increased discriminatory power necessary to maintain trigger efficiency as the LHC luminosity is increased beyond that for which ATLAS was originally designed.

The function of the jFEX module is to identify, in the data received from the electromagnetic and hadronic calorimeters, large energy deposits indicative of jets and τ particles. Global parameters like total transverse energy and missing transverse energy are also calculated. The jFEX does this using algorithms of greater complexity and data of finer granularity than those processed by the current L1Calo system. At the same time larger jets can be calculated than those possible in the Phase-0 system.

The jFEX will be installed in L1Calo during the long shutdown LS2, as part of the Phase-1 upgrade, and it will operate during Run 3. It will remain in the system after the Phase-2 upgrade in LS3, and will operate during Run 4, at which time it will form part of L0Calo. The following sections provide overviews of L1Calo in Run 3 and L0Calo in Run 4. Where the requirements on the jFEX are different in these two runs, it is the most demanding requirements that are used to specify the jFEX. Where these arise from Run 4, this is stated explicitly in the text.

This is a specification for a prototype jFEX. This prototype is intended to exhibit the full functionality of the final module, plus some additional functionality for research purposes, such as the ability to handle different speeds of multi-Gb/s link. All links specified for the prototype as running at 12.8 Gb/s will also support running at 6.4 Gb/s, 9.6 Gb/s and 11.2 Gb/s. This will be used to identify the best speed for the finalised system, which will run at a fixed speed. The other differences between prototype and final functionality are noted in the appropriate sections. Excepting these, the functionality described here can be regarded as that of the final jFEX.

## Overview of Phase-1 System



1. The L1Calo system in Run 3. Components installed during LS2 are shown in yellow /orange.

In Run 3, L1Calo contains three subsystems installed prior to LS2 (see Figure 1):

* the Pre-Processor, which receives shaped analogue pulses from the ATLAS calorimeters, digitises and synchronises them, identifies the bunch-crossing from which each pulse originated, scales the digital values to yield transverse energy (*E*T), and prepares and transmits the data to the following stages;
* the Cluster Processor (CP) subsystem, comprising Cluster Processor Modules (CPMs) and Common Merger Extended Modules (CMXs), which identifies isolated e/γ and τ candidates;
* the Jet/Energy Processor (JEP) subsystem, comprising Jet/Energy Modules (JEMs) and Common Merger Extended Modules (CMXs), which identifies energetic jets and computes various energy sums.

Also installed prior to LS2 is the Topological Processor (L1Topo), which receives data from L1Calo and L1Muon and applies topological algorithms and kinematic cuts.

Additionally, it contains the following two subsystems, installed as part of the Phase-1 upgrade in LS2:

* the eFEX subsystem, comprising eFEXs, and Hub modules [1.2] with Readout Driver (ROD) daughter cards [1.3] , which identifies and counts isolated e/γ and τ candidates, using data of finer-granularity than does the CP subsystem;
* the jet Feature Extractor (jFEX) subsystem [1.4] , comprising jFEX modules and Hub modules with ROD daughter cards, which identifies energetic jets, large τ candidates and computes various energy sums, using data of finer-granularity while handling larger jet windows than does the JEP subsystem. The increased granularity also allows for more flexible and more complex jet algorithms.
* the global Feature Extractor (gFEX) [TODO] subsystem, which identifies calorimeter trigger features requiring the complete calorimeter data.

In Run 3, the electromagnetic calorimeter (ECAL) electronics provide L1Calo with both analogue signals (for the CP and JEP) and digitised data (for the eFEX and jFEX subsystems). From the hadronic calorimeters (HCAL), only analogue signals are received. These are digitised on the JEP and output optically towards the eFEX and jFEX subsystems. Initially at least, the eFEX and jFEX subsystems operate in parallel with the CP and JEP subsystems. Once the outputs of the eFEX and jFEX have been validated, removing the CP and JEP systems from L1Calo is an option, excepting that section of the JEP used to provide hadronic data to the FEX subsystems.

The optical signals from the JEP and ECAL electronics are sent to the FEX subsystems via an optical plant [1.5] . It breaks apart the fibre bundles, duplicates signals and regroups them, changing the mapping from that employed by the ECAL and JEP electronics to that required by the FEX subsystems.

The outputs of the FEX (plus CP and JEP) subsystems comprise Trigger Objects (TOBs): words which describe the location and characteristics of any candidate trigger objects found. These words are transmitted to L1Topo via optical fibres. There, they are merged over the system and topological algorithms are run, the results of which are transmitted to the Level-1 Central Trigger Processor (CTP).

The FEX subsystems are implemented using the ATCA standard [1.6] [1.7] . The jFEX comprises one shelf of 7(6) jFEX modules. L1Topo comprises up to four identical modules, each of which receives a copy of all data from all the jFEX modules.

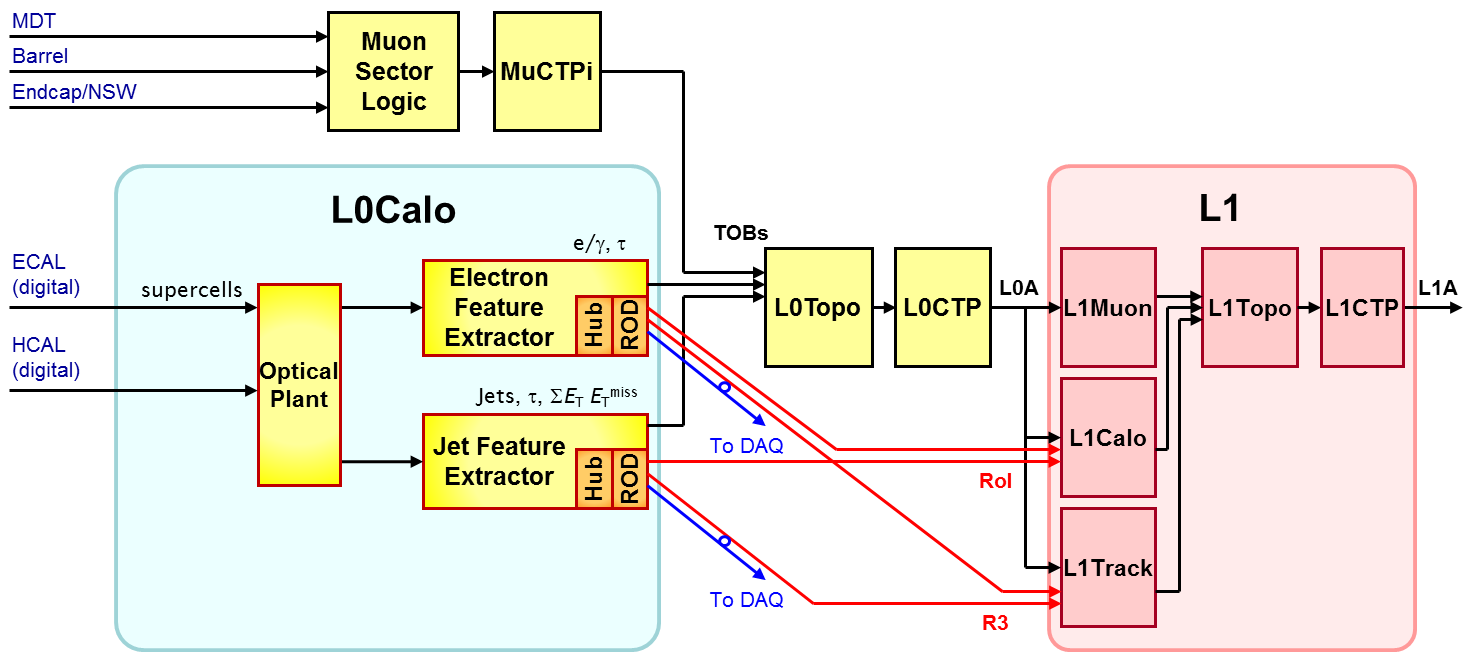
As for the other L1Calo processing modules, on receipt of an L1A the jFEX subsystem provides Region of Interest (RoI) and Readout data to Level‑2 (EF?) and the DAQ system respectively. Each jFEX module outputs these data on to the shelf backplane, and two Hub modules in each shelf aggregate the data and implement the required ROD functionality (via a daughter board). Additionally, the Hub modules provide hubs on the TTC, control and monitoring networks.

## Overview of Phase-2 system

In the Phase-2 upgrade, the entire calorimeter input to L1Calo (from both the ECAL and the HCAL) will migrate to the digital path, rendering the Pre-Processor, CP and JEP subsystems obsolete. The eFEX and jFEX subsystems will thus form the front-end of the L1Calo system at Phase-2. Hence, not only must they be designed for adequate performance at Phase-1, they must also be compatible with the Phase-2 upgrade.

The Phase-2 upgrade will be installed in ATLAS during LS3. At this point, substantial changes will be made to the trigger electronics: the entire calorimeter input to the trigger (from both the ECAL and the HCAL) migrates to the digital path; the latency available to the Level-1 trigger is greatly increased; L1Track is introduced and requires seeding. To meet these new opportunities and demands, L1Calo is split into L0Calo (a Level-0 calorimeter trigger) and L1Calo.

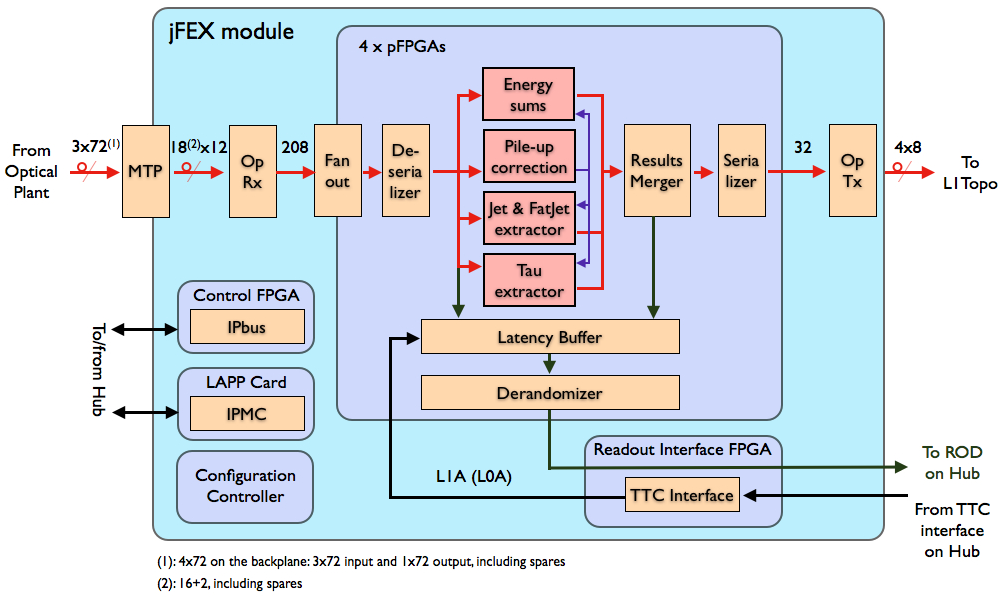
The remaining parts of the previous L1Calo system, the FEX subsystems remain, but they now constitute L0Calo (see Figure 2). They supply L0Topo (and thence L0CTP) with real-time TOB data and, on receipt of an L0A, they supply readout data to the DAQ system, plus RoI data to L1Calo and L1Track.



1. The L0/L1Calo system in Run 4. The new L1 system is shown in red /pink. Other modules (yellow /orange) are adapted from the previous system to form the new L0Calo.

# Functionality

Figure 3 shows a block diagram of the jFEX. The various aspects of jFEX functionality are described in detail below. Implementation details are given in section 6.



1. A block diagram of the jFEX module. Shown are the real-time and readout data paths. With the exception of the L1A, control and monitoring signals are omitted for simplicity. (Todo: updated picture)

## Real-Time Data Path

### Input Data Granularity

The jFEX receives data from all calorimeter parts. From ECAL and HCAL data are received as single *E*T values from each 0.1 × 0.1 (η, φ) trigger tower, summed in depth, covering the central region |η| < 2.5. The region 2.5 < |η| < 3.2 is covered in 0.2 × 0.2. The FCAL region of 3.2 < |η| < 4.9 is divided in up to 12 η-bins, depending on the FCAL layer, with a binning width of 0.4 in φ. Spare bandwidth for an increased granularity in the FCAL region after the Phase-II upgrade is available in the jFEX modules covering the outer η regions.

### Feature Identification Algorithms

The jFEX system examines data from the Electromagnetic and Hadronic Calorimeters, within the range |*η*| < 4.9, to identify energy deposits characteristic of hadronic jets. The data from the range |*η*| < 2.5 is also used to identify energy deposits from τ particles, which are larger than the windows used by the τ identification algorithms on the eFEX system. Combining the data from all jFEX modules, the global parameters and are calculated.

Multiple versions of the sliding window algorithm are implemented in parallel, to achieve the best result in identifying jets from energy deposits in both calorimeters.

Physics studies to determine the optimal algorithms are on-going. The following descriptions are indicative; whilst the precise details are undefined, the overall structure and complexity of the algorithms are understood. A more detailed description will follow once the details are known.

The algorithm employed for the purpose of identifying large energy deposits from τ particles, measures the diffuseness of the deposits, thus distinguishing those produced by τ particles from the more diffuse deposits typical of jets interacting in the ECAL. The algorithm can be divided into two steps: the first step is to seed the process of finding τ candidates by searching for characteristic shower shapes and applying a hadronic veto, followed by measuring the energy of the candidate particles.

The process of finding jet candidates is based on the sliding window algorithm. Energy deposits are summed around a central trigger tower over a small area. If this sum is a local maximum, compared to its 8 neighbours, the central trigger tower is considered as the position of a jet candidate. Once a candidate is found, a bigger area around the position is summed, to calculate the transverse energy. In the JEP system the size of the seeding area is 0.4 × 0.4 (η × φ) with up to 0.8 × 0.8 for the energy calculation. The jFEX system can calculate candidates with up to 1.7 × 1.7. As a further improvement non-square jet windows are feasible, due to the finer granularity. Also weights can be applied to each individual trigger tower within a jet window. The values that provide the best trigger performance are yet to be determined by studies.

In parallel to the regular jet finding an additional algorithm is used to identify heavily boosted objects, like top quarks, which decay into several, separate jets, which is similar to the algorithm used on the gFEX. For this purpose the sliding window algorithm can be extended to a maximum size of 2.2 × 2.2, using a granularity of 0.2 × 0.2. For the inner region of 1.8 × 1.8 the finer granularity can be used to identify a substructure. The detailed mechanism for finding these “fat jets”, as well as the identification of its sub structure, is a matter of ongoing studies.

In addition to the algorithms mentioned above recent development on triggering algorithms, like the so called jet-without-jet algorithms [ref], are being explored to be implemented inside the jFEX.

For those trigger candidates that pass the tests above, Trigger Object words (TOBs) are generated and passed to the merging logic (see below). These TOBs describe the location and type of the candidate and its energy.

At the edges of the η range of the calorimeter, data for the full search windows are not available. In these instances, modified algorithms are applied (the details of which are to be defined).

Beside these TOBs, the parameters and are estimated on each jFEX module for a subset of the detector and sent to L1Topo to be merged to a global value.

The increased instantaneous luminosity of the LHC in Run 3 and Run 4 will produce unprecedented levels of pile-up. To compensate for this, event-by-event energy corrections, which cope with the fluctuations of pile-up energy deposits, are determined and applied on each jFEX module. These corrections are viable for jet finding, as well as for calculations.

In order to calculate these energy corrections each Processor FPGA calculates sub-sums using the data covering its core region in φ and all towers including the environment in η. These pile-up sums are divided in eta bins and transmitted between all Processor FPGAs. The calculation of the energy correction can be done on each Processor FPGA concurrently.

### Processing Area

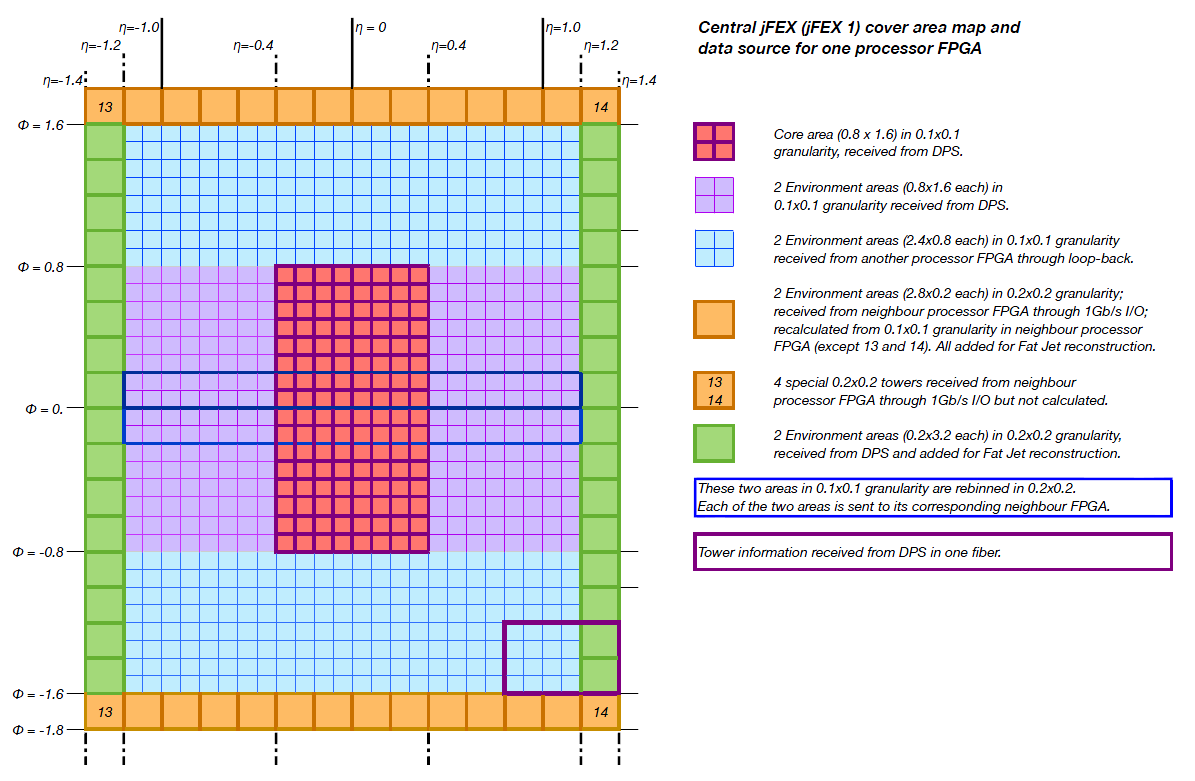
The feature identification algorithms used by the jFEX define a core area, over which TOB candidates are found, and a larger environment area, from which the algorithms need to receive the data. As the core areas examined by neighbouring instances of each algorithm are contiguous, the environment areas overlap. In the central region each module covers the whole φ-ring over 0.8 in η as the core area. Five modules are required for |η| ≤ 2.0. The region within 2.0 < |η| < 4.9 is covered by a single module for each side of the detector. This is possible due to the coarser granularity and missing environment at the outer end. Contiguous to its core area any module receives data in 0.1 × 0.1 granularity (or as fine as available) for 0.8 in η in each direction and an additional row in 0.2 × 0.2 granularity beyond this, to support the identification algorithm for fat jets. Hence a single module in the central region receives data from a region of 2.8 × 6.4 (η × φ).

In total, seven jFEX modules are required to process all of the data from the calorimeters within the range |η| < 4.9. The hardware on all jFEX modules is the same; the differences in core area processed are implemented via firmware.

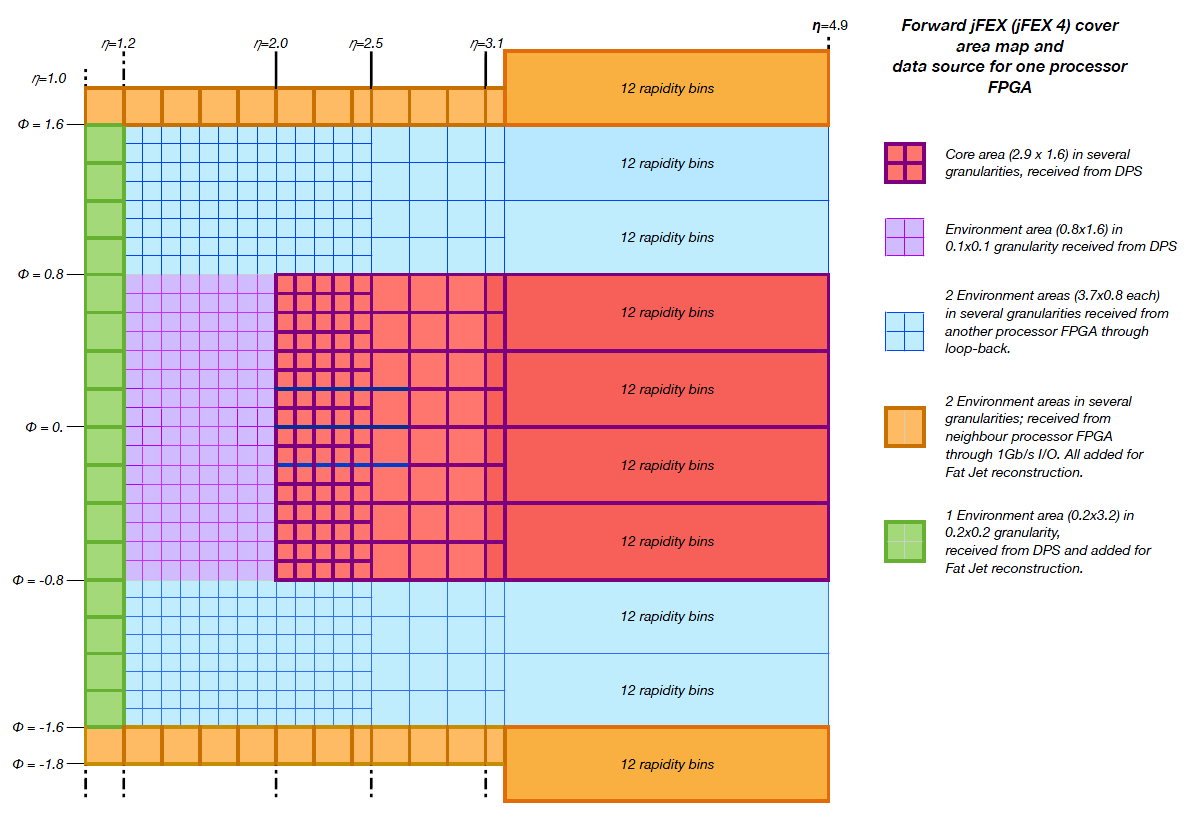
Except the data from the FCAL region, all input signals received by the jFEX system require a 3-way fan out from the DPS, at the DPS.

On each jFEX module there are 4 Processor FPGAs. Each of them covers a quarter of the modules core area in φ. Combined with the environment of 0.8 in both, η and φ, a single FPGA receives data from a total of 2.4 × 3.2 (η × φ) in a granularity of 0.1 × 0.1. The extended environment increases the covered area to 2.8 × 3.6 (η × φ), using a coarser granularity in the outer region.

Each jFEX module receives only one copy of the data from every trigger tower. The required duplication for overlapping areas is handled internally as shown in Figure 4 and Figure 5. The core area (displayed in red) and the environment within the same φ range (shown in violet) are received directly from the source. This information is duplicated early in the MGT data path and retransmitted via PMA “loopback” to the neighbouring FPGAs. The duplicated data are shown in light blue. The extended environment shown in green is transmitted via the same links that carry the data for the fine granularity. Each input contains 16 trigger towers in fine granularity, arranged in 0.4 × 0.4, and 2 additional cells covering 0.2 × 0.4, covering the same φ range while being adjacent in η to the fine granularity. Due to the limited η coverage per module in the TREX, data from the Tile calorimeter is sent on separate links as 8 towers covering 0.2 x 1.6 in total. The extended environment shown in orange is transmitted from the neighbouring FPGAs via low latency 1Gb/s differential links. These data in a coarser granularity are used to extend the biggest possible jet window to 2.2 × 2.2.



1. The input data for a single Processor FPGA on a module covering the central region, including granularity and source. The red area shows the core area, while the blue and violet shows the environment area. The orange and green shows the extra environment area added for fat jets feature extraction.



1. The input data for a single Processor FPGA on a module covering the forward region, including granularity and source. The red area shows the core area, while the blue and violet shows the environment area. The orange and green shows the extra environment area added for fat jets feature extraction.

### Output Bandwidth

The real-time output data of the jFEX are transmitted to the L1Topo system. This system comprises multiple, identical modules, and the jFEX is assumed to transmit the same data to each module. Furthermore, each L1Topo module houses two FPGAs, and, ideally, the jFEX should transmit the same data to each L1Topo FPGA, to maximise the flexibility and minimise the latency of L1Topo.

Due to bandwidth constraints at the L1Topo input, the real-time output of the jFEX is carried on a single fibre per jFEX FPGA to each L1Topo FPGA. The exact format of the jFEX TOBs is yet to be defined, but they are estimated to be approximate 40 bits in size. Due to the symmetry in the jFEX system each jFEX FPGA has one MiniPOD transmitter with 12 available channels for the real-time output. This can be used to cover an increased number of L1Topo modules or an increased number of available input links compared to the currently planned system.

If the available bandwidth per FPGA is not sufficient, parallel IO links can be used to merge the TOBs within the four FPGAs of each module. This option requires an increased latency compared to the direct output from each processor separately.

## Readout Data Path

On receipt of an L1A signal, the jFEX provides data to a number of systems: in Run 3, it provides RoI data to Level-2; in Run 4, it provides RoI data to L1Track and L1Calo (the jFEX being part of L0Calo in Run 4); in both Run 3 and Run 4, it provides data to the DAQ system. Collectively, these data are referred to here as readout data.

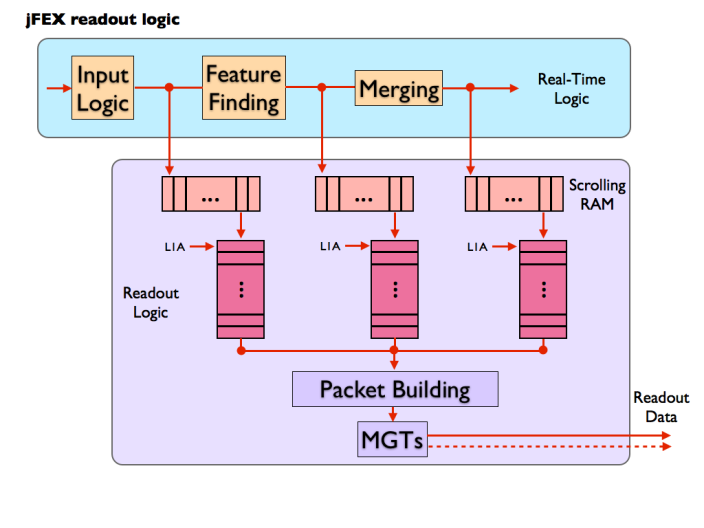
The jFEX outputs a single stream of readout data, which contains the superset of the data required by all of the downstream systems. In Run 3, these data are transmitted across the crate backplane to a ROD. In Run 4, there are two RODs per crate and the jFEX transmits identical readout data to both RODs. It is the RODs that are responsible for formatting the data as required by the downstream systems, and handling the multiple interfaces.

For each event that is accepted by the Level-1 trigger, the jFEX can send three types of data to the readout path: final TOBs, expanded TOBs (XTOBs) and input data. The final TOBs are copies of those transmitted to L1Topo. In normal running mode these are the only data read out. The XTOBs are words that contain more information about trigger candidates than can be transmitted on the real-time data path (see section 5.2). They are extracted from the real-time path before sending the TOBs to L1Topo and therefore, as the number of TOBs in the real-time path may be reduced due to the limited bandwidth, the number of XTOBs may be larger than the number of TOBs. To minimise the amount of readout data generated, XTOBs are not normally read out. However, this functionality can be enabled via the slow control interface. This cannot be done dynamically for individual events.

The input data comprise all data received from the calorimeters. They are copied from the real-time path after serial-to-parallel conversion and after the CRC word has been checked. There are a number of programmable parameters, set via slow control, that determine which input data are read out. These are as follows.

* The Input Readout mode: by default, only input data from fibres that have generated an error are read out. However, the readout of data received without error can also be enabled.
* The Input Channel Mask: the read out of individual channels of input data, from individual FPGAs, can be disabled. A channel here means the data received at an FPGA from one fibre. Most of the input data received by a single Processor FPGA are redundant copies, created because of the need to fan out data between the FPGAs and modules. The Input Channel Mask provides a way of stripping redundant channels from the jFEX readout. It also allows data from permanently broken links to be excluded from the readout process.
* The Input Readout Veto: this veto is asserted for a programmable period (0-256 ticks) after the read out of any Input Data. It provides a means of pre-scaling the amount of Input data read out, preventing it from overwhelming the readout path.

The mechanism for capturing readout data is illustrated in Figure 6. For every bunch crossing all input data, intermediate and final TOB data are copied from the real-time path and written to scrolling, dual-port memories. They are read from these memories after a programmable period, of up to 3 μs. At this point they are selected for readout if they meet both of the following criteria: an L1A pertaining to them is received, and they are enabled for readout by the control parameters described above. Otherwise, they are discarded.



1. A functional representation of the jFEX readout logic.

For each L1A, data from a time frame, programmable via control parameters, can be read out. The selection of data for read out is a synchronous process with a fixed latency, and it is the period for which data are held in the scrolling memories that determines the start point of this time frame. The correct value must be determined when commissioning L1Calo (it should correspond to the period from when the data are copied into the scrolling memories, to when an L1A pertaining to those data is received at the jFEX, plus or minus any desired offset in the time frame). The jFEX hardware allows the read out of overlapping time frames. At low rates (including everything before Run 4) the jFEX does not expect to read out overlapping time frames, still the firmware will be able to provide this if required. At high rates, the read out of overlapping time frames will be more likely, but the frame length and trigger rate will need to be controlled carefully to prevent buffer overflow.

It is possible that for a BC there will be no TOB data (or XTOB data when enabled) to be captured. In such cases a control word is inserted into the readout path to indicate this. This word, which is used for flow-control, is internal to the jFEX; it is not passed to the ROD. (TODO: how to handle this w/o merger?)

Data that are selected for readout are written to FIFOs, where they are stored before transmission to the ROD. This storage is necessary for two purposes: first, data are sent to the ROD in formatted packets, requiring some data to be stored as the packet is built; secondly, the peak rate at which readout data are captured by the jFEX exceeds that at which they can be transferred to the ROD.

All of the readout logic described above is implemented in each Processor FPGA. The data are built into packets and transmitted, via the shelf backplane, to a ROD (in Run 3) or two RODs (in Run 4, each ROD receiving a copy of the same data). Six links, each running at up to 10 Gb/s, carry the data to a ROD. Each of the Processor FPGAs is connected via one of these links per ROD. The FPGAs U1 and U2 are connected with one additional per ROD each. This extended bandwidth can be used if the bandwidth using one link per FPGA is not sufficient. To balance the data sent from FPGAs with a different number of ROD links, the data can be shared using parallel I/O links or a bigger part of the redundant input data can be sent from those FPGAs with a higher link count, if this option is enabled.

The transfer of data from the FIFOs to the ROD(s), is initiated whenever the FIFOs are not empty. There is no backpressure asserted from the ROD(s) to pause transmission.

Table 1 shows an estimate of the maximum readout bandwidth required of the jFEX. This maximum case occurs in Run 4, where readout is initiated by the L0A signal at a rate of 1 MHz. However, only the TOB data need to be read out at this rate. In normal operation, the input data are only read out at a pre-scaled rate for monitoring purposes. They are also read out if an error is detected, but if that error is persistent, those data are also pre-scaled. Thus, for the input data, a maximum readout rate of 50 KHz is acceptable. The number of bunch crossings from which data is read out after an L1A (L0A) can be set via control parameters. For normal operation a window size of three bunch crossings is assumed in this calculation.

Readout of the XTOB data is optional. The calculation shown in Table 1 assumes a pre-scaled rate of 500 KHz. The number of XTOBs in this calculation is based on the number of TOBs that can be sent in the real time data path from a single Processor FPGA. The maximum number of generated XTOBs depends on the exact implementation of the algorithms and is thus not yet known. Based on the available bandwidth, the maximum rate for readout of XTOBs can be calculated, once details for the algorithms are known.

| Data | No. Chan. | Bits / Chan. / BC  (post 8b/10b) | BC / Event | Trigger Rate / KHz | Bandwidth / Gb/s |
| --- | --- | --- | --- | --- | --- |
| Input data | 416 | 320 | 3 | 50 | 19.97 |
| XTOBs | 96 | 80 | 3 | 500 | 11.52 |
| TOBs | 4 | 320 | 3 | 1000 | 3.84 |
| **Total** |  | | | | **35.33** |

1. An estimate of the maximum readout bandwidth required for a jFEX module. For XTOBs, a channel equals an XTOB; for the other types of data, a channel equals a fibre. (TODO: comment 43: WHY????)

## Latency

A breakdown of the estimated latency of the real-time path of the jFEX is given in the ATLAS TDAQ System Phase-1 Upgrade Technical Design Report [1.1] .

## Error Handling

The data received by the jFEX from the Calorimeters are accompanied by a CRC code. This is checked in the Processor FPGAs, immediately after the data are converted from serial, multi-Gb/s streams into parallel data. If an error is detected, the following actions are performed:

* All data to which a detected error pertains are suppressed (i.e. set to zero) on the real-time path. They are passed to the readout path as received.
* The Error Check Result for the current clock cycle is formed from the ‘OR’ of all error checks for the current bunch crossing.
* The Input Error Count is incremented for any clock cycle where there is at least one error in any input channel.
* A bit is set in the Input Error Latch for any channel that has seen an error. These bits remains set until cleared by an IPBus command.
* The global Input Error bit is formed from the ‘OR’ of all bits in the Input Error Latch.

The Error Check Result, Input Error Count, Input Error Latch and Input Error bit can all be read via IPBus. A single IPBus command is provided to clear all of these registers. The Error Check Result and Input Error Count are included in the readout data for the current bunch crossing. The jFEX does not generate any other external error signal, so data monitoring or regular hardware scanning must detect an error condition.

## Interface to TTC

TTC signals are received in the jFEX shelf in the Hub-ROD module. There, the clock is recovered and commands are decoded, before being re-encoded using a local protocol (to be defined). This use of a local protocol allows the TTC interface of the shelf to be upgraded without any modification of the jFEX modules.

The jFEX module receives the clock and TTC commands from the Hub-ROD via the ATCA backplane. It receives the clock on one signal pair and the commands on a second (see section 6.8 for details).

## Slow Control

The slow control functions of the jFEX are implemented on a Zynq SoC which is placed on an extension mezzanine, mounted via an FCI Gig-Array connector. On a first iteration of this mezzanine a PicoZed board is used with a limited connectivity to the Zynq compared to later iterations.

An IPBus interface is provided for high-level, functional control of the jFEX. This allows, for example, algorithmic parameters to be set, modes of operation to be controlled and spy memories to be read.

IPBus is a protocol that runs over Ethernet to provide register-level access to hardware. Here, it is run over a 1000BASE-T Ethernet port, which occupies one channel of the ATCA Base Interface. On the jFEX there is a local IPBus interface in every FPGA, plus the IPMC. These interfaces contain those registers that pertain to that device. The Zynq implements the interface between the jFEX and the shelf backplane, routing IPBus packets to and from the other devices as required. The Zynq also contains those registers which control or describe the state of the module as a whole. For those devices such as MiniPODs, which have an I2C control interface, an IPBus-I2C bridge is provided.

## Environment Monitoring

The jFEX monitors the voltage and current of every power rail on the board. It also monitors the temperatures of all the FPGAs, of the MiniPOD receivers and transmitters, and of other areas of dense logic. Where possible, this is done using sensors embedded in the relevant devices themselves. Where this is not possible, discrete sensors are used.

The voltage and temperature data are collected by the jFEX IPMC, via an I2C bus. From there, they are transmitted via IPBus to the ATLAS DCS system. The jFEX hardware also allows these data to be transmitted to the DCS via IPMB and the ATCA Shelf Controller, but it is not foreseen that ATLAS will support this route.

If any board temperature exceeds a programmable threshold set for that device, IPMC powers down the board payload (that is, everything not on the management power supply). The thresholds at which this function is activated should be set above the levels at which the DCS will power down the module. Thus, this mechanism should activate only if the DCS fails. This might happen, for example, if there is a sudden, rapid rise in temperature to which the DCS cannot respond in time.

## Commissioning and Diagnostic Facilities

To aid in module and system commissioning, and help diagnose errors, the jFEX can be placed in Playback Mode (via an IPBus command). In this mode, real-time input data to the jFEX are ignored and, instead, data are supplied from internal scrolling memories. These data are fed into the real-time path at the input to the feature-extracting logic, where they replace the input data from the calorimeters.

Optionally, the real-time output of the jFEX can also be supplied by a scrolling memory. It should be noted that, in this mode, the jFEX will process data from one set of memories, but the real-time output will be supplied by a second set of memories. Depending on the content of these memories, this may result in a discrepancy between the real-time and readout data transmitted from the jFEX.

In Playback Mode the use of the input scrolling memories is mandatory, the use of the output scrolling memories is optional, and it is not possible to enable Playback Mode for some channels but not others. Playback Mode is selected, and the scrolling memories loaded, via the slow control interface. The scrolling memories are 256 words in depth.

In addition to the above facility, numerous flags describing the status of the jFEX can be read via the slow control interface (see section 8). Access points are also provided for signal monitoring, boundary scanning and the use of proprietary FPGA tools such as ChipScope and IBERT.

## ATCA form factor

The jFEX is an ATCA module, conforming to the PICMG® 3.0 Revision 3.0 specifications.

# Data formats

The formats of the data received and generated by the jFEX have yet to be finalised. Those defined here are working assumptions only.

## Input data

The jFEX modules receive data from the calorimeters on optical fibres. For the region |η| < 2.4 each fibre carries the data for 16 adjacent triggers towers, i.e. an area of 0.4 × 0.4 (η × φ). The data from the calorimeters within 2.4 < |η| < 3.2 can be sent using one fibre per 0.4 in φ, due to the coarser granularity of only 12 trigger towers. This also leaves spare capacity to include the additional information from the Tile-HEC overlap, which cannot be included in the fibres of the central region. The Processor FPGAs covering 0.4 < |η| < 1.2 do not receive these fibres. The corresponding modules receive data from the overlap region on separate fibres. The complete data from the FCAL, covering all three layers, is carried on three fibres per 0.8 in φ. An increased granularity up to a factor of 4 in the first FCAL layer, provided by the possible FCAL upgrade during LS3, the sFCAL, can be received using spare links on the modules covering the outer η regions. An increased granularity by more than a factor 4 would require additional jFEX modules.

These mappings are independent of the line rate of the optical fibres between 9.6 Gb/s and 12.8 Gb/s. The data format and content, however, are dependent upon the line rate within this range. For the highest rate of 12.8 Gb/s, the data are encoded as specified below. Further study is required to establish the optimal bandwidth.

The jFEX system will be required to be compatible with tile input data coming from two different systems: in Run 3 it receives them from the TREX; in Run it accepts them from the Tile Cal DPS. Due to the option to send BCMUXed data, the required input bandwidth during Run 3 is only about half of the bandwidth required in Run 4. Therefore the scenario used for these calculations is always the most demanding: the setup after the Phase-II upgrade. This might result in some spare links during Run 3 but has no impact on the design of the jFEX.

Data from the calorimeters are transmitted to the jFEX as continuous, serial streams. To convert these streams into parallel data, the jFEX logic must be aligned with the word boundaries in the serial data. The scheme for achieving this is yet to be defined, but there are a number of possible mechanisms. For example, boundary markers can be transmitted during gaps in the LHC bunch structure. These markers are substituted for zero data and are interpreted as such by the jFEX trigger-processing logic. Periodic insertion of such markers allows links to recover from temporary losses of synchronisation automatically.

### Calorimeter Data Format

* Up to 13-bit data are provided for each of the 16 trigger tower at 12.8Gb/s input speed.
* Up to 13-bit data are provided for each of the two 0.2 × 0.2 (η × φ) cells at 12.8 Gb/s input speed.
* Bit depth limited to 12 bit at 11.2 Gb/s input speed.
* A 10-bit cyclic redundancy check is used to monitor transmission errors.
* 8b/10b encoding is used to maintain the DC balance of the link and ensure there are sufficient transitions in the data to allow the clock recovery.
* Word-alignment markers (8b/10b control words) are inserted periodically, as substitutes for zero data.

Using 8b/10b encoding, the available payload of a 12.8 Gb/s link is 256 bits per bunch crossing (BC). The above scheme uses up to 244 bits (data from 18 cells, plus a 10-bit CRC). The remaining 12 bits/BC are spare. The order of the data in the payload is not yet defined.

## Real-Time Output Data

The Real-time output of the jFEX comprises TOBs, each of which contains information about a jet or τ candidate, such as its location and the deposited energy. Figure 7, Figure 8 and Figure 9 show the draft format of the jet TOB, fat jet TOB and τ TOB respectively. Besides these candidates the global values, and , are transferred to L1TOPO. They are sent separately as , and as 13-bit energy values.

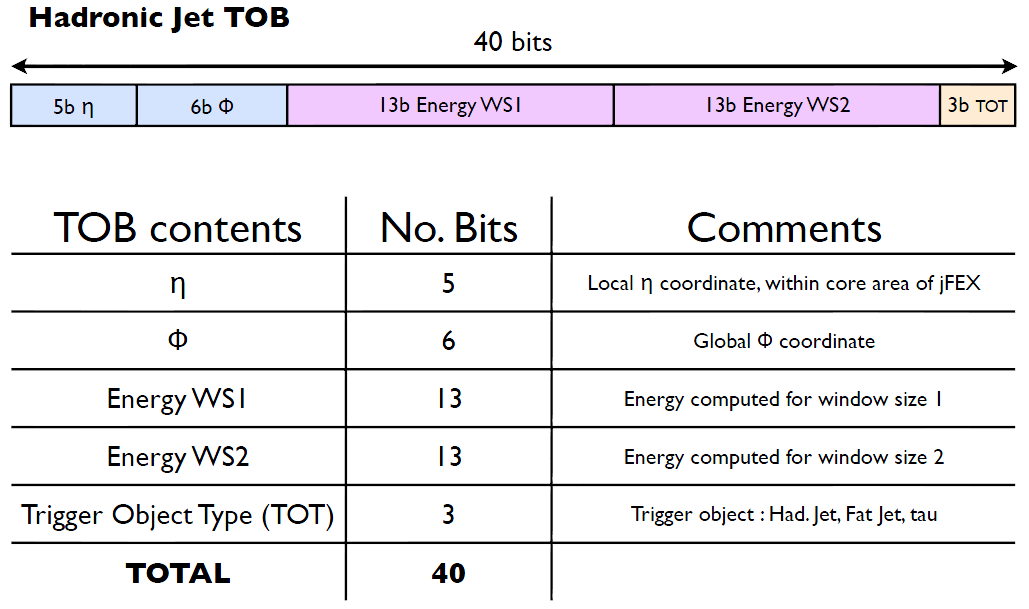
Due to multiple jet finding algorithms, the jet TOBs include two energies. The results from two algorithms, which are based on the same seeding procedure, can be compressed into one TOB. The sizes of the remaining TOBs are adjusted to match the jet TOBs.

The TOBs are transmitted to L1Topo on optical fibres. The line rate and protocol used for this transmission is the same as that used to transmit data from the calorimeters to the jFEX. The baseline specification is thus as follows.

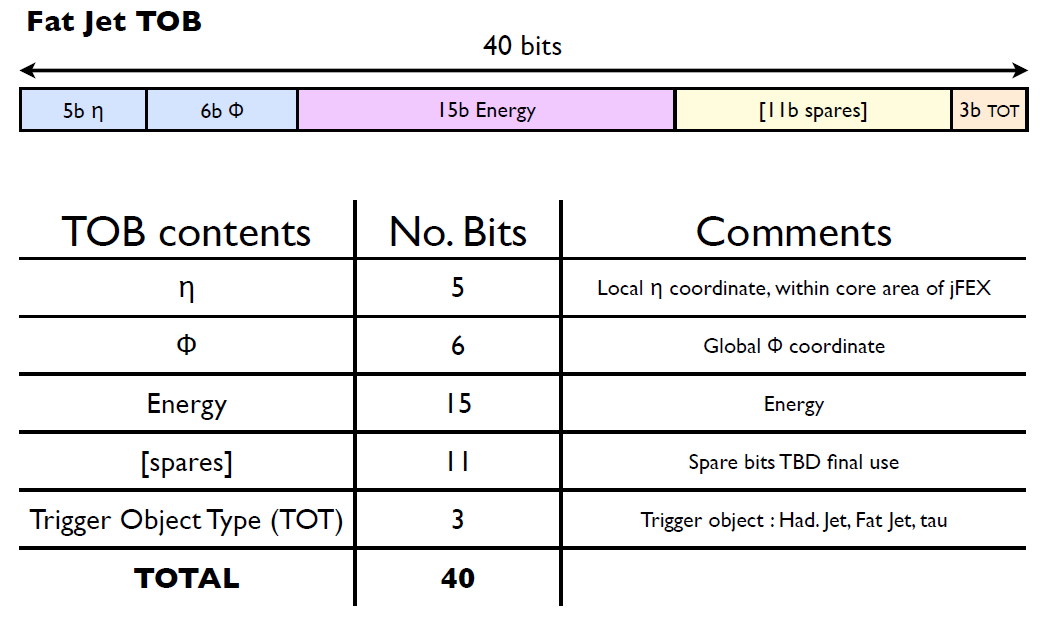
* The data are transferred across the optical link at a line rate 12.8 Gb/s.
* 8b/10b encoding is used to maintain the DC balance of the link and ensure there are sufficient transitions in the data to allow the clock recovery.
* Word-alignment markers (8b/10b control words) are inserted periodically, as substitutes for zero data.

For TOBs of 40 bits, four links at the given specifications allow a maximum of 20 TOBs and the global values to be transmitted to L1Topo per bunch crossing.

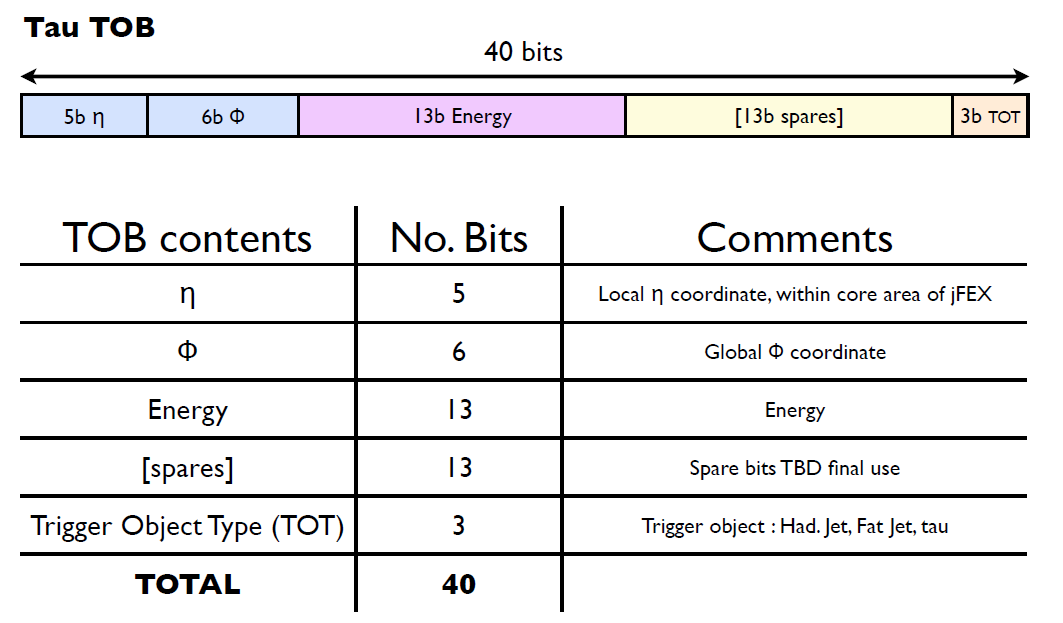
Should the specification of the jFEX inputs change, the specification of the real-time outputs will be updated to match. (Using a common line rate and encoding scheme enables the output data to be looped back to the inputs for diagnostic purposes.)



1. Draft jet TOB Format.



1. Draft fat jet TOB Format.



1. Draft τ TOB format.

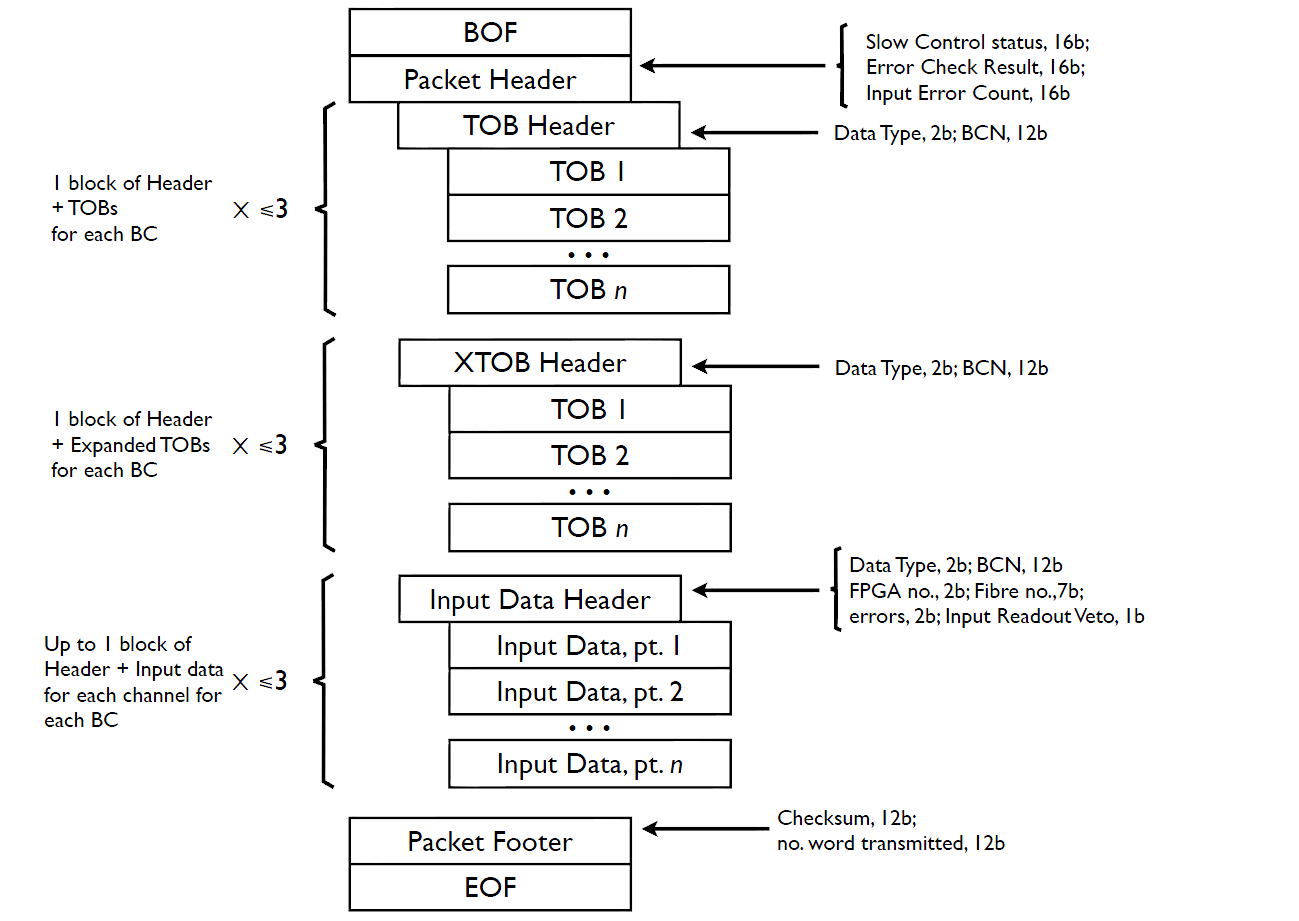
## Readout data

On receipt of an L1A, the jFEX transmits to the ROD a packet of data of the format shown in Figure 10. This packet contains up to three types of data: TOBs, XTOBs and input data (see below). The TOBs are exact copies of those output from the jFEX on the real-time path. The input data are copies of the calorimeter data as received in the Processor FPGAs. The XTOBs are words that contain more information about trigger candidates than can be transmitted on the real-time data path. If the readout of XTOBs is enabled, any TOB in the readout data will have a corresponding XTOB. The readout data may also contain XTOBs for which there is no corresponding TOB. Such XTOBs describe trigger candidates for which TOBs have not been transmitted to L1Topo because of the input bandwidth limit of that module. The exact format of the XTOBs is yet to be determined. Preliminary assumptions introduce a width of up to 64 bits.

The data in the readout packet are from a programmable window of bunch crossings. The size of this window is the same for all types of data and is limited by the available memory in the Processor FPGAs. The size can be set via control parameters.

Within the packet, the data are organised first according to type, and then according to bunch crossing. Headers mark the boundaries between data types and bunch crossings. Not every type of data is necessarily present in a packet. If a data type is absent, then the headers for that data are also absent. In the extreme, the packet may contain no data, in which case just the packet header and footer are transmitted.

The jFEX readout packets are transmitted to the ROD via six links at up to 10 Gb/s per link, using a link-layer protocol that is to be defined.



1. A provisional format for a readout data packet.

# Implementation

The description of the implementation is based on jFEX modules in the central region. Details of the implementation differ on modules covering the outer regions, due to changes in the input data granularity and the η coverage.

This is meant to be a functional description. Details of the pins used for a certain connection can be found in the schematics.

## Input Data Reception and Fan Out

The jFEX receives data from the calorimeters via optical fibres. Each fibre carries data from an area of 0.4 × 0.4 (η, φ). In order to cover an area as described in section 5.1, a single jFEX module must receive data on up to 192 fibres. Two modules require up to 16 (TODO: 0.2x0.2 not included in description) additional links, carrying the data from the Tile-HEC overlap, making a total of 208.

The input fibres to the jFEX are organised into 20 ribbons of 12 fibres each. They are routed to the jFEX via the rear of the ATCA shelf, where a rear transition module provides mechanical support. Optical connections between the fibres and the jFEX are made by up to two 72-way (hadronic data including overlap and spares) and two 48-way (e.m. data) Multi-fibre Push-On/Pull-Off (MPO) connectors, mounted in Zone 3 of the ATCA backplane. These connectors allow the jFEX to be inserted into, and extracted from, the shelf without the need to handle individual ribbon connections.

On the jFEX side of the MPO connectors, 20 optical ribbons (each comprising 12 fibres) carry the signals to 20 Avago MiniPOD receivers. These perform optical to electric conversion. They are mounted on board, around the Processor FPGAs, to minimise the length of the multi-Gb/s PCB tracks required to transmit their output.

Each of the received signals is transmitted to two of the four Processor FPGAs. The Processor FPGA, which has a core region that covers the region in φ from which the data on a fibre is originated, receives the incoming signal. The data is retransmitted to one of the neighbouring Processor FPGAs via “PMA loopback”. Once the signal has been received by the FPGA and equalisation has been performed, but before the signal has been decoded, it is sent from the high-speed receiver to the paired high-speed transmitter. There is a latency penalty of >25 ns and some degradation of signal quality associated with this method. The jFEX must therefore handle upwards of 400 multi-Gb/s signals.

For the connection from the MiniPODs to the FPGAs, the GTH transceivers are used due to their reduced power consumption compared to the GTY version. On the GTY, side where the looped-back data is received, only a small fraction of the MGTs is required to also use the transmitter for special links described below.

A block of 3 quads starting from bank 219 is connected to each of the 5 MiniPODs per FPGA (RX side) and looped to the same neighbouring FPGA (TX). The transmitting part of the third block is split into 5 links to each neighbouring FPGA and 2 links to MMCX test points. The jFEX is able to receive data on 232 fibres using loop-back only for duplication. If more input bandwidth is required 8 additional links can be used that are not included in the loop-back scheme. This data can be shared on spare parallel-IO links between the FPGAs if required.

The input mapping therefore is required to have all links containing data of a φ-octant (0.8) on one optical ribbon for hadronic and electromagnetic calorimeter respectively.

The MGTREDCLKs provided for these links are described in the clocking section.

## Processor FPGA

There are four Processor FPGAs on each jFEX module. The functionality they implement can be grouped into real-time, readout and slow-control functions. All Processor FPGAs on a jFEX module have the same functionality. The differences between the Processor FPGAs on different modules are caused by the varying core areas covered by a certain module and are implemented via different firmwares.

Every Processor FPGA performs the following real-time functions.

* It receives, from MiniPOD optical receivers and via PMA loop-back from its neighbouring FPGAs, up to 118 inputs of serial data at up to   
  12.8 Gb/s and additional data for the extended environment from neighbouring FPGAs on 1Gb/s differential links. These carry data from the calorimeters, from an environment of 2.8 × 3.6 (3.9 × 3.6 in forward region).
* Optionally, it sends (and receives) the extended environment in φ in 0.2 × 0.2 granularity to (and from) its neighbouring FPGAs.
* It applies the feature-identification algorithms described in section 4.1.2 to the calorimeter data, to identify and characterise jet and τ objects and calculate global values.
* For each jet and τ object found, it produces a TOB, as described in section 5.2.
* It prioritises the TOBs, and if the number it has found exceeds the number that can be transmitted L1Topo in one BC, the excess TOBs are supressed.
* Optionally excess TOBs can be sent to the other Processor FPGAs if the additional latency is available.
* It transmits its TOB results and global values to L1Topo via up to 12 high speed links.

Each Processor FPGA can process a core area of calorimeter data of 0.8 × 1.6 (2.9 × 1.6 in the forward regions). The differences between the modules depending on their covered η range are implemented via firmware. The hardware is the same for all modules.

On the readout path (described in section 4.2), each Processor FPGA performs the following functions.

* The Processor FPGA records the input data and the TOBs generated on the real-time path in scrolling memories, for a programmable duration of up to 3 μs.
* On receipt of an L1A, it writes data from the scrolling memories to the FIFOs, for a programmable time frame. This is only done for those data enabled for readout by the control parameters.
* The Processor FPGA transmits data from the readout FIFOs to the ROD module, using an electrical connection on the backplane.

For slow control and monitoring, each Processor FPGA contains a local IPBus interface, which provides access to registers and RAM space within the FPGAs.

The Processor FPGA is a Xilinx XCVU190. The dominant factor in the choice of device is the available number of multi-Gb/s receivers, low latency parallel links and logic resources.

Of the 120 high speed links available in the XCVU190, depending on the covered η range, 80 to 104 are used in the currently planned mapping scheme. Some of the remaining links are used for slow control functions as described below. The spare resources can be used to increase the input bandwidth if required.

### General-purpose I/O

Regarding general-purpose I/O, of the 448 pins available 408 pins can be used for differential links as limited by the FPGA package. There are 8 banks of 52 and a ‘half-bank’ with 26 pins. Each full bank can support up to 24 (12 for the half bank) differential links and 4 single ended links. A subset of the differential links can also be used as (differential) input for global clocks (GC) to the FPGA. Each pin of a differential pair can also be used as a single ended link. All Banks are operated with 1.8V. The differential pairs are designed for LVDS, the single ended links are mainly assumed to use LVCMOS. The expected transmission speed on differential links is around 1 Gb/s.

Since the links between FPGAs are supposed to be used in both directions with, at least in most cases, each pin being connected to the exact same pin of the other FPGA , careful firmware planning is required to avoid implementing both ends of a link as receiver or transmitter. This will lead to a slightly different firmware on the input and output end. The main functionally inside each FPGA for one module remains identical.

According to the Xilinx documentation on the package, the full banks are named as 61, 62, 63, 65, 67, 70, 71 and 72 and the half bank as 68. In a preliminary package there was an additional bank 66 available of which two differential inputs, only available as GC inputs, remain in the current version. Bank 65 has some special use pins that are required for configuration of the FPGA. Therefore only a subset of this bank is available as parallel IO.

Each Processor FPGA has 72 differential links each of its neighbouring FPGAs (combined for both directions of communication). Clockwise (U1, U2, U4, U3), the following banks (e.g. from U1) are connected to the corresponding bank of the next FPGA (e.g. U2):

* Bank 63 to Bank 61
* Bank 67 to Bank 72
* Bank 70 to Bank 71

Diagonally, the Banks 62 (24 differential links) and 65 (18 differential links – detailed information which pairs are available can be found in the schematics on page 86, using an identical scheme for all four FPGAs or on the bottom of page 5 in the block diagrams) are connected between U1-U4 and U2-U3 respectively.

The half-bank 68 is used to support 24 single ended links to the mezzanine. This connection was initially designed to be used for a parallel IO implementation of the IPBus. If the version utilizing high speed links is used, these links are available as spares. Since they are routed to the mezzanine connector, they can also be used to extend the connectivity between Processor FPGAs. These links would be required to be operated at a significantly lower speed compared to the differential links.

12 single ended links are connected to the CPLD controlling the clock path (as described below) from each FPGA. For this connection the pins, which can only support single ended links, from the banks 61, 62 and 67 are used.

Four single ended links from Bank 71 are connected to MMCX connectors for testing and debugging.

Two single ended links from Bank 72 are connected to status LEDs.

### Readout

With regard to the readout path (which is described in section 4.2), the functions of each Processor FPGA are as follows.

* It monitors status flags from the readout FIFOs.
* If the FIFOs contain the data for at least one complete event, the FPGA initiates the building of a readout packet.
* It reads from the FIFOs all data required for that event, as indicated by the control parameters (Input Readout Mode, Input Readout Veto, etc.).
* It builds the readout event packet.
* It transmits packets of readout data to the ROD, via the shelf backplane, from one multi‑Gb/s transceiver.
* FPGAs U1 and U2 each have one additional link to the ROD. This can be used to increase the read-out bandwidth if required.
* In Phase-II, identical copies are sent to each of the two RODs

The ROD is connected via the backplane using Fabric Channel 1(and 2 for the second ROD in Phase-II) on connector J23/P23 according to the ATCA specifications. In the regular specifications only 4 output links are defined for each Fabric Channel. Since the jFEX only required two inputs, the two remaining inputs are used for the extended connection on U1 and U2.

On the FPGA end, GTY quad number 133 (TX only) is used for this connection. As clocking source for this connection, MGTREFCLK1\_133 is provided with a dedicated clock (MGT3\_CLK) that can be set to a multiple of the LHC clock, independently of the clock used on the real-time data path.

The maximum data rate out of each module on the readout path, aggregated over the six transmitters and ignoring framing information, is 60 Gb/s. Assuming 8B/10B encoding, this gives a payload of 48 Gb/s.

### Control

The Zynq provides the interface between the jFEX and the IPBus network on the backplane. It contains those control and status registers that concern the operation of the jFEX as whole (i.e., all registers not specific to one particular FPGA), plus a switch to direct IPBus traffic to and from those registers and IPBus-accessible RAM blocks that are implemented in the other FPGAs.

In addition to high-level control, IPBus provides a pathway for transmitting environment data (on temperatures and voltages) from the jFEX to the ATLAS DCS system. This traffic is also handled by the Zynq, which routes packets between the DCS system and the IPM Controller on the jFEX.

The control signals and feedback are transferred between Zynq and the Processor FPGA via a bidirectional high speed link. On the FPGA end, this links is connected to quad number 126. A dedicated reference clock (MGT4\_CLK) is provided to MGTREFCLK1\_126 as described below. If an implementation of the IPBus on parallel IO is preferred, the connection from Bank 68, as described above, can be used.

If there is no Zynq available to support the IPBus connection, U1 has additional reference clock inputs to be able to use a single link of the real time data path (input and output) to set up an IPBus connection in the testing phase. This scheme is not foreseen to be used in regular operation. MGTREFCLK1\_221 and MGTREFCLK0\_121 are provided with the same clock as mentioned above. In this mode U1 can be implemented as an IPBus master, using up to two links rerouted from the mezzanine to other FPGAs.

With regard to the TTC interface, TTC commands and information are transmitted within the jFEX shelf using a local protocol. This isolates the jFEX module from any changes in the ATLAS TTC system between Run 3 and Run 4. The local TTC protocol is yet to be defined, but it is estimated to require much less bandwidth than the 1 Gb/s allocated. The functionality of the Zynq with respect to the TTC interface is as follows.

* Control commands, such as Event Counter Reset, Bunch Counter Reset and L1A, are received from the shelf backplane via a single link of 1 Gb/s.
* These commands are decoded and passed to the relevant hardware on the jFEX via individual control lines.
* Any information requested from the jFEX hardware is received by the Merger FPGA on individual status lines. There it is built into a packet and transmitted over the shelf TTC network.

If during the testing phase there is no Zynq available, the TTC commands can be rerouted from the mezzanine to one of the FPGAs using the second MGT connection available on Bank 126. The remaining FPGAs will receive these commands on a parallel IO connection.

## Clocking

There are two types of clock sources on jFEX: on-board crystal clocks and the LHC TTC clock, received from the ATCA backplane. These clock sources are fed via the clocking circuitry to four FPGAs. The 40.08MHz TTC clock is assumed to have too much jitter to drive multi-Gb/s links directly. A PLL chip is therefore used to clean up the jitter on this clock. To facilitate standalone tests of the high-speed serial links on the jFEX, an on-board crystal clock of 40.08MHz is also provided. From the input of 40.08 MHz the PLL chip can generate clocks of frequency *n* × 40.08 MHz within a certain range. This flexibility allows the multi-Gb/s links on the jFEX to be driven at a large range of different rates. There are three different multiples of the TTC clock that are required for the high speed line rates to be tested:

* *9.6 Gb/s: n = 6 (or 3)*
* *11.2 Gb/s: n = 7*
* *12.8 (6.4) Gb/s: n = 4 (or 8)*

The Si5338 was recommended by CERN and was therefore chosen for the prototype. The original TTC clock or a jitter-cleaned version of this is also sent to the extension mezzanine.

A detailed description of the different clocking trees is given in the following subsections. I wanted to add one more sentence here… (TODO)

Every clock tree can use either a clock from the mainboard, supplied by the on board jitter cleaner or in some cases by a local crystal oscillator, or a clock coming from the mezzanine. An alternative PLL chip, the Si5345, is already planned to be tested on the mezzanine for better jitter performance while offering the same features with more output channels compared to the Si5338. If the new chip shows good results in the tests, it is considered an option to use the Si5345 on the jFEX mainboard in production.

In all clock trees with multiple input options, the selection of the required input is driven by the cCPLD. Since Bank 2 from this cCPLD is supplied with 3.3V while the clocking chips usually expect 2.5V, a 4.7kΩ pull-up resistor to 2.5V is connected to all relevant selection inputs. The cCPLD is only expected to pull the signal to ground for logic low and being disabled as an output if logic high is required.

### Global clock (GCK\_CLK)

The global clock (GC) is meant to drive the internal logic of the FPGAs and the Zynq. There are two parallel inputs used for each FPGA: the two remaining differential inputs of Bank 66, expected to be driven with 2V5 LVDS signals.

The main clock tree has a multiplexer chip (SY56028XR) with four inputs as its first input stage. The four inputs are:

* IN3: LVPECL 40.0787 MHz oscillator (default)
* IN2: CML backplane input from Hub slot 2
* IN1: CML backplane input from Hub slot 1
* IN0: LVDS clock from U1

The oscillator connected to IN3 is provided for stand-alone operation.

IN1 receiving the clock from Hub slot 1 is expected to be the clock input in normal operation and during integrated tests with an external clock source. IN2 is also available as a possible back-up to use a clock coming from Hub slot 2 if a signal from Hub slot 1 cannot be received for any reason.

IN0 receives an LVDS clock converted from a single ended CMOS output from U1. This option was included to provide a synchronous clock for MGT1 (described below) and GC if the jitter cleaner is not working. In this case an oscillator can be used for MGT while U1 sends a recovered clock out of a general purpose IO pins which is converted to an LVDS signal close to the FPGA and sent to the GC input multiplexer.

The final stage of the GC is a 2:8 (4 FPGAs, mezzanine, cCPLD, loop back, one unused output) fan-out chip receiving the raw clock on IN1 as default and a jitter cleaned version of this clock on IN0. There is an option to loop one output of the fan-out chip back to the jitter cleaner to activate a zero-delay mode in the jitter cleaner to provide a stable phase for all TTC based clocks. If this mode does not work properly with another PLL chip in between, the link from the jitter cleaner to the fan-out chip can be disconnected by removing a 0 Ω resistor and connected directly to its own input. This mode only allows to have a non-jitter cleaned GC on the jFEX mainboard (or using another chip on the mezzanine) but might be required for the MGT clocks. The jitter cleaner is not intended to be used without a feedback input. For a use with no feedback input, IN6 should be tied to ground which can only be done using a cable since there is no such connection provided on the PCB (same for IN4).

One output of the fan-out chip is converted from LVDS to CMOS and connected to single ended clock input (GCK2) of the cCPLD to allow for internal logic in this device to be operated at the same speed as the connected FPGAs.

A backup for the GC is coming from the mezzanine, being routed directly to each FPGA. This input can be used if a different clock is required (e.g. for testing) or if the non-jitter cleaned TTC clock is not usable and the on board jitter cleaner is not sufficient to provide a stable clock.

### Input clock (CLK\_MGT1)

The clock for the real time data input is provided to all MGTs in all FPGAs - directly or indirectly. The UltraScale architecture has the option to forward MGT reference clocks from the receiving quad to up to two neighbouring quads in either direction (except for stacked silicon borders) with some limitation on the maximum number of concurrently routed clocks. Since this is a new feature in the UltraScale series compared to the possibility to forward a clock only to direct neighbours in Virtex 7 devices, there was no way to test the reliability of this extended forwarding. Therefore, the jFEX clocking scheme supplies the reference clock to one out of three quads without relying on the untested feature.

The input stage for the MGT1 clock consists of two SY56028XR multiplexers. One of them (MUX1) has three oscillators and a footprint for a programmable clock (which was not available from distributers for assembly). The other one (MUX2) is connected to OUT0 of the jitter cleaner with the ability to be programmed to various multiples of its input clock and a signal

Following to the multiplexers are two stages of SY58034 2:6 fan-out chips feeding all required clock signals to four FPGAs. As default MUX1 with the 160.3148 MHz oscillator is providing the clock signal.

All clocks (except for the LVPECL oscillators) are expected to be 2.5V CML signals.

### Output clock (CLK\_MGT2)

The output clock MGT2 for real time transmission to L1Topo has two input options:

* AC coupled LVPECL signal from the mezzanine (default) (TODO: cross check with Bruno for LVPECL vs CML)
* CML AC coupled signal from OUT1 of the jitter cleaner

There is no PLL chip purely as a multiplexer but two stages of 2:6 SY58034 fan-out chips with these two clock signals connected to the first stage.

The clock is provided to MGTREFCLK1\_119, MGTREFCLK1\_120, MGTREFCLK1\_121 and MGTREFCLK1\_128. The first three quads (119, 120 and 121) are connected to the transmitter MiniPOD while quad 128 has two connections to test points for oscilloscope measurements with different track lengths.

The FPGAs are supplied with a separate clock for the outputs to allow for a different line rate compared to the input. Also a clock supplied to every quad is required to test the highest speeds of the GTY MGTs since line rates above 16 Gb/s cannot be used with a forwarded reference clock.

### ROD clock (CLK\_MGT3)

The output clock MGT3 for read-out data transmission to ROD module(s) has two input options:

* AC coupled LVPECL signal from the mezzanine (default) (TODO: cross check with Bruno for LVPECL vs CML)
* CML AC coupled signal from OUT2 of the jitter cleaner

There is no PLL chip purely as a multiplexer but a single stage of a 2:6 SY58034 fan-out chip with these two clock signals connected to its input.

### IPBus clock (CLK\_MGT4)

The IPBus MGT4 for the serial link version of the IPBus implementation has two input options:

* AC coupled LVPECL signal from the mezzanine (default)
* AC coupled LVPECL signal from a local oscillator

On the jFEX, the protocol between the IPBus master and the IPBus slaves runs using this clock. Hence, the jFEX module control function over IPBus is independent from the TTC clock domain.

The local oscillator can be used in case one of the FPGAs is implemented as IPBus master. The local clock signal cannot be routed to the mezzanine. For the default use case of an IPBus master implemented in the Zynq, a separate oscillator on the mezzanine should be used.

The clock is provided to MGTREFCLK1\_126. For U1 only, this clock is also routed to MGTREFCLK1\_221 and MGTREFCLK0\_121 to provide a clock for an optical Ethernet connection without using the Zynq.

### TTC command clock (EXT\_TTC\_CLK)

The TTC command clock is routed directly from the mezzanine to each FPGA as MGTREFCLK0\_126. There are no fan-out chips or multiplexers available.

This clock is supposed to be used for either retransmitted TTC commands coming from the Zynq or the HUB sourced signal from the backplane rerouted via the mezzanine to one of the FPGAs.

### Configuration clock (EMCCLK)

The (optional) configuration clock EMCCLK is routed as an LVDS signal from the mezzanine to each FPGA separately with a local LVDS to CMOS conversion for the single ended input in Bank 0.

This clock can be used for most configuration modes to replace the internal clock on the CCLK output.

The maximum clock speed is 150 MHz according to Xilinx documents.

### FELIX clock (MGT5\_CLK)

A local 125 MHz oscillator is connected to MGTREFCLK1\_226 on U1 to provide a dedicated clock to receive FELIX commands without a mezzanine, independently of the currently used IPBus setup.

### CPLDs

Additional to the GC copy (as described above) the cCPLD also receives a CMOS clock signal from U1 on its GCK1 input.

Both CPLDs receive a 1 kHz single ended clock on GCK0 which is expected to be used for most of the control functions.

## FPGA configuration

There are multiple options to configure the FPGA. The most ‘out-of-the-box’ way is via JTAG. In the default setup after production all FPGAs are connected to a single header in a daisy chain. This allows for a single connector to be used for all FPGAs in the initial boundary scan. To increase configuration speed the chain can be split (by replacing 0 Ω resistors) into separate parts for U1 and U2 respectively. U3 and U4, which are not expected to be populated on the prototype, are not separable on this board.

Additional to the expectedly rather slow configuration using JTAG, which also requires an external computer to be connected to the corresponding headers, the required connectivity for master SPI or SelectMAP configuration (according to Xilinx UG570) with up to 8 bit data width to the mezzanine is available.

For a combination of master SPI and SelectMAP configuration a 24 bit wide bus is routed from each FPGA to the mezzanine. There are multiple options on how to use this connection from which can be chosen by designing the mezzanine accordingly. Details on the required design can be extracted from Xilinx documentation.

The bit streams will be stored on the mezzanine. Updates of the bit streams can be updated via IPBus, depending on the mezzanine, via the FPGAs or the Zynq.

Re-configuration of the FPGAs can be initiated via IPBus and via the low-level management IPMI bus. The FPGAs can be reconfigured separately.

## The IPM Controller

For the purposes of monitoring and controlling the power, cooling and interconnections of a module, the ATCA specification defines a low-level hardware management service based on the Intelligent Platform Management Interface standard (IPMI). The Intelligent Platform Management (IPM) Controller is that portion of a module (in this case, the jFEX) that provides the local interface to the shelf manager via the IPMI bus. It is responsible for the following functions:

* interfacing to the shelf manager via dual, redundant Intelligent Platform Management Buses (IPMBs), it receives messages on all enabled IPMBs;
* negotiating the jFEX power budget with the shelf manager and powering the Payload hardware only once this is completed (see section 6.6);
* managing the operational state of the jFEX, handling activations and deactivations, hot-swap events and failure modes;
* implementing electronic keying, enabling only those backplane interconnects that are compatible with other modules in shelf, as directed by the shelf manager;
* providing to the Shelf Manager hardware information, such as the module serial number and the capabilities of each port on backplane;
* collecting, via an I2C bus, data on voltages and temperatures from sensors on the jFEX, and sending these data, via IPBus, to the Merger FPGA;
* driving the ATCA-defined LEDs.

The jFEX uses the IPMC mezzanine produced by LAPP as the IPM Controller [1.9] . The form factor of this mezzanine is DDR3 VLP Mini-DIMM.

As a backup all required links for the IMP controller are also routed to the mezzanine to provide the option to replace the on board card by placing a connector on the mezzanine. Only one IMPC card is allowed to be operated on the jFEX at any time. The option of using an IPMC card on the mezzanine is not foreseen to be available in production, due to the limited slot height for single slot ATCA boards.

## Power Management

With regard to power, the hardware on the jFEX is split into two domains: Management hardware and Payload hardware. The Management hardware comprises the IPM Controller plus the DC-DC converters and the non-volatile storage that this requires. By default, on power up, only the Management hardware of the jFEX is powered (drawing no more than 10 W), until the IPM Controller has negotiated power-up rights for the Payload hardware with the shelf manager. This is in accordance with the ATCA specification. However, via a hardware switch it is also possible to place the jFEX in a mode where the Payload logic is powered without waiting for any negotiation with the shelf controller. This feature, which is in violation of the ATCA specification, is provided for diagnostic and commissioning purposes.

Power is supplied to the jFEX on at -48V DC feed. A PIM4000 convertor accepts this feed and provides a power supply of 3.3 V to the Management hardware (3V3\_UP). The 48V supply is being monitored and forwarded to a QBVW033A0B, which supplies 12V to the Payload hardware. This 12V supply is stepped down further, by multiple switch-mode regulators, to supply the multiplicity of voltages required by the payload hardware.

On power-up of the Payload hardware, the sequence and timing with which the multiple power rails are turned on are controlled by the sCPLD, which is supplied by a local DC-DC converter from management power. First supplies to be powered up are the board wide available 2.5 V and 3.3 V, which are also used as management power in the remaining power supplies. The sequencing of the separate FPGA input voltages are to be done according to the Xilinx specifications to minimize the current drawn during power up. Where possible, with respect to voltage level and sequencing requirements, power planes are combined to reduce the requirements of the power planes in the layer stack up while providing the largest possible planes to reduce power loss on the PCB.

All FPGA power supplies are located on a power mezzanine which can be replaced to test different types of power supplies. The first version of this mezzanine uses MAX20751 converters as supplies with low voltage but high currents like VCCINT. For higher voltages like 1.8V VCCINT board level 3.3V, MAX15301 converters are used. For the FPGA VCCAUX voltages, with low expected current drawn, MIC68400 converters are used.

All assumptions on power requirements are derived from the Xilinx Power Estimator spreadsheets.

Excluding the optional exception noted above, the jFEX conforms to the full ATCA PICMG® specification (issue 3.0, revision 3.0), with regard to power and power management. This includes implementing hot swap functionality, although this is not expected to be used in the trigger system.

## Front-panel Inputs and Outputs

The following bi-directional control interfaces are available on the front panel. See section 6.10 for the use of these interfaces.

* JTAG Boundary Scan. The optimum physical form factor for this interface is to be identified.
* 1G Ethernet socket.

## Rear-panel Inputs and Outputs

### ATCA Zone 1

This interface is configured according to the ATCA standard. The connections include

* -48V power supply
* hardware address,
* IPMB ports A and B (to the Hub module),
* shelf ground,
* logic ground.

Figure 11 shows the backplane connections between the jFEX and the Hub module, which are located in Zones 1 and 2 of the ATCA backplane. See the ATCA specification for further details.

### ATCA Zone 2

#### Base Interface

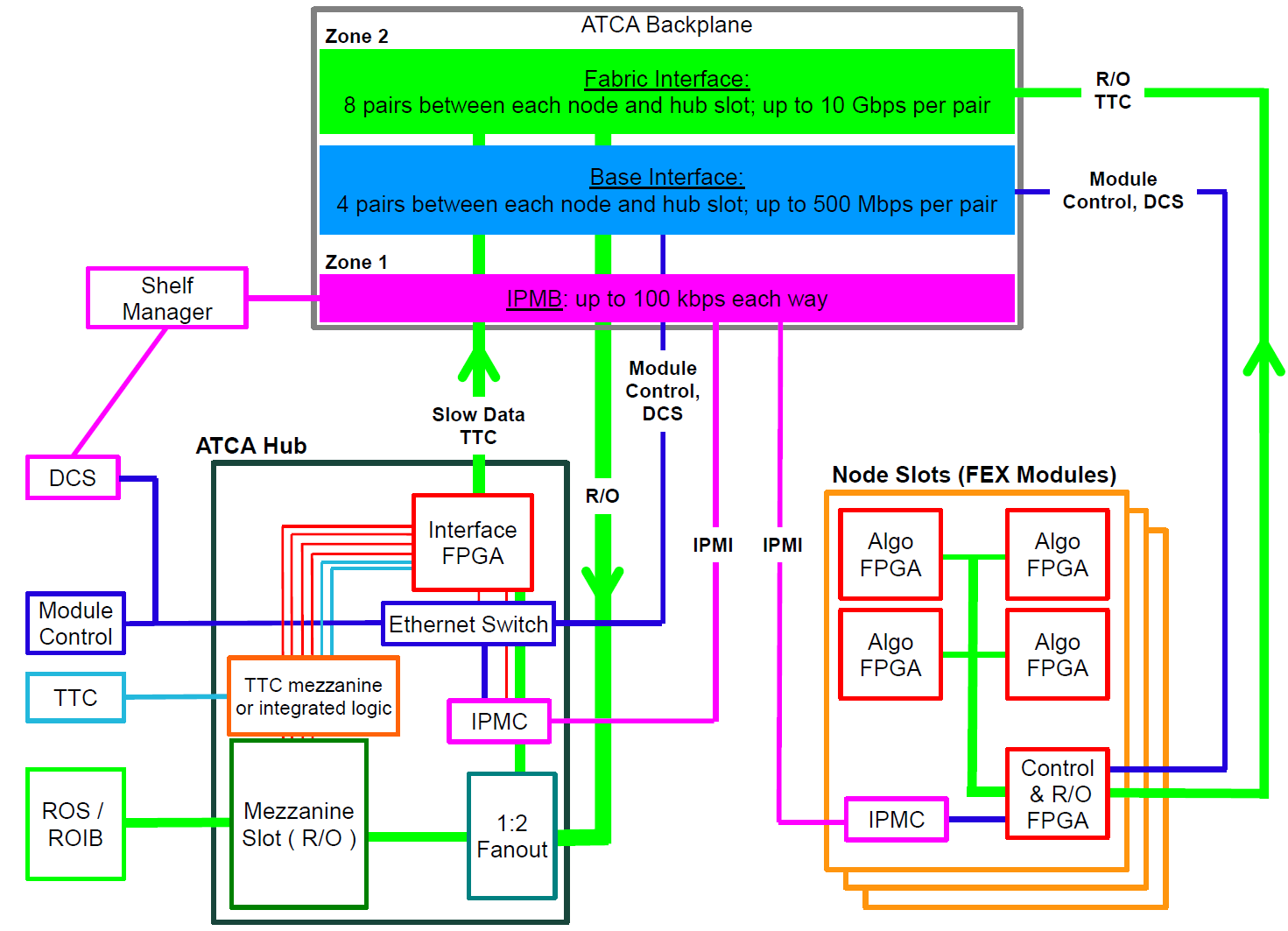
The Base Interface comprises eight differential pairs. Four of these are connected to hub slot one and are used for module control, the other four are connected to hub slot two and are used to carry DCS traffic. Both of these functions are implemented using IPBus, running over 1G Ethernet links.

#### Fabric Interface

The Fabric Interface comprises 16 differential signal pairs, eight of which are connected to hub slot one, and eight of which are connected to hub slot two. Those signal pairs connected to hub slot one are used as follows:

* One signal pair is used to receive the TTC clock.
* One signal pair is used to receive decoded TTC commands, plus near real-time signals such as ROD busy. The protocol is to be defined. The link speed does not exceed 10 Gb/s.
* Six signal pairs are used to transmit readout data. The link speed does not exceed 10 Gb/s. Two out of these six signal pairs are used as receivers in standard ATCA backplanes. They are inverted to increase the possible readout bandwidth.

Those signal pairs connected to hub slot two are reserved for the same functions as above. Potentially, this allows redundant connections to be made to this hub slot. However, the firmware necessary to drive and receive data to and from the Fabric Interface of hub slot two is undeveloped.



1. The ATCA backplane connections between the jFEX and the Hub module. (TODO: update or remove)

### ATCA Zone 3

ATCA zone houses two 48-way and two 72-way optical MPO connectors. They house a total of up to 240 fibres, carrying data from the calorimeters to the jFEX (see section 6.1). At the rear of the MPO connectors, optical fibres carry data from the calorimeters to the jFEX via the L1Calo Optical Plant. These fibres are supported in the jFEX shelf by a (passive, mechanical) rear transition module (RTM). On the jFEX side of the connectors, fibre ribbons carry the calorimeter data to MiniPOD receivers, mounted on-board. The optical connections are made on the insertion of the jFEX into the shelf, and broken on its extraction.

## LEDs

All LEDs defined in the ATCA specifications are located on the jFEX front panel. In addition, further status LEDs are provided on either the front panel or the top side. These indicate functions like power, Done signals, L1A receipt und further LEDs for diagnostic purposes for all FPGAs.

## Instrument Access Points

### Set-Up and Control Points

The following interfaces are provided for the set-up, control and monitoring of the jFEX. They are intended for commissioning and diagnostic use only. During normal operation it should not be necessary to access the jFEX via these interfaces.

* The JTAG Boundary Scan port: via this port a boundary scan test can be conducted, all FPGAs on the jFEX can be configured, the configuration memory of the Configurator can be loaded and the FPGA diagnostic/evaluation tool ChipScope can be run, including for IBERT tests. This port is on the front panel.
* The 1G Ethernet port: this port provides an auxiliary control interface to the jFEX, over which IPBus can be run, should there be a problem with, or in the absence of, an IPBus connection over the shelf backplane. It is on the front panel and connected to the Merger FPGA.
* The RS232 port: this port provides a control interface of last resort, available if all others fail. It is mounted on the top side of the module and connects to the Merger FPGA. Firmware to implement this interface will only be developed if needed.

### Signal Test Points

Due to the sensitive nature of multi-Gb/s signals, no test points are provided on PCB tracks intended to carry multi-Gb/s data. If such signals need to be examined, this must be done via firmware. Test points are placed on a selection of those data and control tracks that are not operating at multi-Gb/s.

For each FPGA, spare, general-purpose IO pins are routed to headers. Furthermore, spare multi-Gb/s transmitters and receivers are routed to SMA sockets. With appropriate firmware these connections allow internal signals, or copies of data received, to be fed to an oscilloscope, for example, or driven from external hardware.

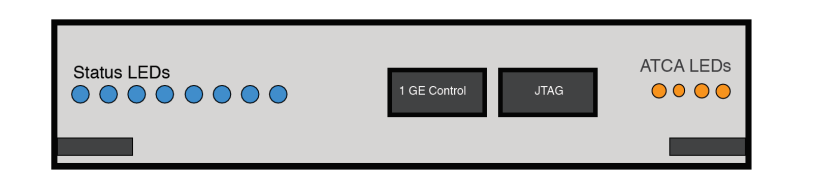
Details on these test points can be found in the general description of the MGT links and general purpose IO sections respectively.

### Ground Points (TODO: ask Bruno, update)

At least six ground points are provided, in exposed areas on the top side of the module, to allow oscilloscope probes to be grounded.

Additional ground points are provided on 3-pin I²C connectors meant to connect to an I²C-USB-converter or a separate board.

# Front-Panel Layout (TODO: update)



1. Preliminary front panel layout (not to scale).

Figure 12 shows a preliminary template for the front panel layout of the jFEX. Shown are the JTAG port for boundary scanning and FPGA access, an auxiliary Ethernet control port, status LEDs and the ATCA extraction/insertion handles. These components are not drawn to scale.

# Programming model

## Guidelines

The slow-control interface of the jFEX obeys the following rules.

* The system controller can read all registers; there are no ‘write only’ registers.
* Three types of register are defined: Status Registers, Control Registers and Pulse Registers.
* All Status Registers are read-only registers. Their contents can be modified only by the jFEX hardware.
* All Control Registers are read/write registers. Their contents can be modified only by system controller. Reading a Control Register returns the last value written to that register.
* All Pulse Registers are read/write registers. Writing to them generates a pulse for those bits asserted. Reading them returns all bits as zero.
* Attempts to write to read-only registers, or undefined portions of registers, result in the non-modifiable fields being left unchanged.
* If the computer reads a register (e.g. a counter) which the jFEX is modifying, a well-defined value is returned.
* The power-up condition of all registers bits is zero, unless otherwise stated.

## Register Map (TODO: PDR status)

The full register map will be developed during the design process and documented here. The following is an incomplete list of the requirements that have been identified thus far.

* Programmable parameters:
  + Enable intermediate TOB readout
  + Input error registers
  + Input mask bits
  + Input Readout Mode
  + Input Readout Mask
  + Input Readout dead time length
  + Programmable input delay
  + Readout frame length
  + Readout Offset, Input data
  + Readout Offset, Intermediate Data
  + Readout Offset, Final TOBs
* Status words:
  + The Error Check Result,
  + Input Error Count,
  + Input Error Latch
  + Input Error Bit Mask
  + Readout parameters
* Memory access:
  + All dual-port RAM on readout path
  + All FIFOs on readout path (non-destructive, random access).
  + Playback and spy buffers, memory mapped

## Register Descriptions

This section is a place holder, to be completed during the design process.

# Glossary

|  |  |
| --- | --- |
| ATCA | Advanced Telecommunications Computing Architecture (industry standard). |
| BC | Bunch Crossing: the period of bunch crossings in the LHC and of the clock provided to ATLAS by the TTC, 24.95 ns. |
| BCMUX | Bunch-crossing multiplexing: used at the input to the CPM, JEM (from Phase 1) and eFEX, this is a method of time-multiplexing calorimeter data, doubling the number of trigger towers per serial link. |
| CMX | Common Merger Extended Module. |
| CP | Cluster Processor: the L1Calo subsystem comprising the CPMs. |
| CPM | Cluster Processor Module. |
| DAQ | Data Acquisition. |
| DCS | Detector Control System: the ATLAS system that monitors and controls physical parameters of the sub-systems of the experiment, such as gas pressure, flow-rate, high voltage settings, low-voltage power supplies, temperatures, leakage currents, etc. |
| ECAL | The electromagnetic calorimeters of ATLAS, considered as a single system. |
| eFEX | Electromagnetic Feature Extractor. |
| FEX | Feature Extractor, referring to either an eFEX or jFEX module or subsystem. |
| FIFO | A first-in, first-out memory buffer. |
| FPGA | Field-Programmable Gate Array. |
| HCAL | The hadronic calorimeters of ATLAS, considered as a single system. |
| IPBus | An IP-based protocol implementing register-level access over Ethernet for module control and monitoring. |
| IPMB | Intelligent Platform Management Bus: a standard protocol used in ATCA shelves to implement the lowest-level hardware management bus. |
| IPM Controller | Intelligent Platform Management Controller: in ATCA systems, that portion of a module (or other intelligent component of the system) that interfaces to the IPMB. |
| IPMI | Intelligent Platform Management Interface: a specification and mechanism for providing inventory management, monitoring, logging, and control for elements of a computer system. A component of, but not exclusive to, the ATCA standard. |
| JEM | Jet/Energy Module. |
| JEP | Jet/Energy Processor: the L1Calo subsystem comprising the JEMs. |
| jFEX | Jet Feature Extractor. |
| JTAG | A technique, defined by IEEE 1149.1, for transferring data to/from a device using a serial line that connects all relevant registers sequentially. JTAG stands for Joint Technology Assessment Group. |
| L0A | In Run 4, the Level-0 trigger accept signal. |
| L0Calo | In Run 4, the ATLAS Level-0 Calorimeter Trigger. |
| L1A | The Level-1 trigger accept signal. |
| L1Calo | The ATLAS Level-1 Calorimeter Trigger. |
| LHC | Large Hadron Collider. |
| MGT | As defined by Xilinx, this acronym stands for Multi-Gigabit Transceiver. However, it should be noted that it denotes a multi-gigabit transmitter–receiver pair. |
| MiniPOD  MicroPOD | An embedded, 12-channel optical transmitter or receiver.  An embedded, 12-channel optical transmitter or receiver, smaller compared to the MiniPOD. |
| MPO | Multi-fibre Push-On/Pull-Off: a connector for mating two optical fibres. |
| PMA | Physical Media Attachment: a sub-layer of the physical layer of a network protocol. |
| ROD | Readout Driver. |
| RoI | Region of Interest: a geographical region of the experiment, limited in *η* and *φ,* identified by the Level-1 trigger (during Run 3) as containing candidates for Level-2 trigger objects requiring further information. In Run 4, RoIs are used in the same between the Level-0 and Level-1 triggers. |
| Shelf | A crate of ATCA modules. |
| SMA | Sub-Miniature version A: a small, coaxial RF connector. |
| Supercell | LAr calorimeter region formed by combining ET from a number of cells adjacent in *η* and*φ*. |
| TOB | Trigger Object. |
| TTC | The LHC Timing, Trigger and Control system. |
| XTOB | Extended Trigger Object. A data packet passed to the readout path, contained more information than can be accommodated on the real-time path. |

# Document History

|  |  |
| --- | --- |
| **Version** | **Comments** |
| 0.0 | Internal circulation |
| 0.1 | L1Calo circulation |
| 0.2 | PDR draft |
| 1.0 | Detailed documentation of the prototype, first draft |
|  |  |