

Dissertation  
submitted to the  
Combined Faculty of Natural Sciences and  
Mathematics  
of the Ruperto-Carola-University of Heidelberg, Germany  
for the degree of  
Doctor of Natural Sciences

Put forward by  
Tigran Mkrtchyan

born in: Yerevan

Oral examination: 18-01-2023





**Extension of the L1Calo PreProcessor  
System for the ATLAS Phase-I Calorimeter  
Trigger Upgrade**

Referees: Prof. Dr. Hans-Christian Schultz-Coulon  
Prof. Dr. Ulrich Uwer



## Zusammenfassung

Der hardwarebasierte Level-1-Kalorimeter-Trigger (L1Calo) des ATLAS-Experiments am Large Hadron Collider (LHC) wurde für die Run-3 Periode der Datenaufnahme verbessert. Durch neue und anspruchsvolle Algorithmen wird das Upgrade die Leistung des Triggers in einer herausfordernden, hohen Pile-Up Umgebung erhöhen und dabei die niedrigen Trigger Schwellwerte beibehalten.

Die Tile Rear Extension (TREX)-Module sind die neueste Erweiterung des L1Calo PreProcessor-Systems. Mit modernsten FPGAs und optischen Hochgeschwindigkeits-Transceivern liefern die TREX-Module digitalisierte hadronische Transversalenergien aus dem ATLAS Tile Calorimeter alle 25 ns an die neuen Feature Extractor-Prozessoren (FEX). Außerdem sind die Module so gestaltet, dass die Kompatibilität mit den ursprünglichen Trigger-Prozessoren erhalten bleibt. Das System aus 32 TREX-Modulen wurde entwickelt, produziert und erfolgreich in ATLAS installiert. Diese Arbeit beschreibt die funktionale Implementierung der Module und die detaillierte Integration und Inbetriebnahme in den ATLAS-Detektor.

## Abstract

For the Run-3 data-taking period at the Large Hadron Collider (LHC), the hardware-based Level-1 Calorimeter Trigger (L1Calo) of the ATLAS experiment was upgraded. Through new and sophisticated algorithms, the upgrade will increase the trigger performance in a challenging, high-pileup environment while maintaining low selection thresholds.

The Tile Rear Extension (TREX) modules are the latest addition to the L1Calo Pre-Processor system. Hosting state-of-the-art FPGAs and high-speed optical transceivers, the TREX modules provide digitised hadronic transverse energies from the ATLAS Tile Calorimeter to the new feature extractor (FEX) processors every 25 ns. In addition, the modules are designed to maintain compatibility with the original trigger processors. The system of 32 TREX modules has been developed, produced and successfully installed in ATLAS. The thesis describes the functional implementation of the modules and the detailed integration and commissioning into the ATLAS detector.



# Contents

<b>Abstract</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Theoretical Background</b>	<b>5</b>
2.1 The Standard Model of Particle Physics . . . . .	5
<b>3 The LHC and the ATLAS Experiment</b>	<b>9</b>
3.1 The Large Hadron Collider . . . . .	9
3.2 The ATLAS Detector . . . . .	12
<b>4 The Trigger and Data Acquisition of ATLAS</b>	<b>19</b>
4.1 The Level-1 Trigger System . . . . .	20
4.2 The Central Trigger Processor . . . . .	25
4.3 The DAQ and the Higher-Level Trigger system . . . . .	25
<b>5 The Upgrade of the Level-1 Calorimeter Trigger</b>	<b>27</b>
5.1 Motivation for the upgrade . . . . .	27
5.2 New L1Calo Trigger system . . . . .	28
<b>6 The L1Calo PreProcessor and the TREX</b>	<b>37</b>
6.1 Hardware design, from prototyping to production . . . . .	37
6.2 Adapting the PPMs for TREX compatibility . . . . .	42
6.3 Control, configuration of the TREX and the interface with the PreProcessor	44
6.4 Clocking and trigger information . . . . .	51
6.5 The Real-time data path . . . . .	53
6.6 The Readout data path . . . . .	63
6.7 The System-On-Chip and Monitoring . . . . .	70
<b>7 Functional Tests</b>	<b>85</b>
7.1 Test software suite . . . . .	85
7.2 Acceptance tests . . . . .	85

---

7.3	Power and thermal measurements . . . . .	87
7.4	Clock stability measurements . . . . .	88
7.5	Latency measurements . . . . .	89
7.6	LVDS transmission measurements . . . . .	94
7.7	Signal integrity of the high-speed transmission . . . . .	95
<b>8</b>	<b>Installation, Commissioning and First Data</b>	<b>99</b>
8.1	Installation in ATLAS . . . . .	99
8.2	Connectivity and Fibre Mapping . . . . .	103
8.3	Integration with the legacy trigger systems . . . . .	106
8.4	Integration with the FEX systems . . . . .	108
8.5	Interfaces to the DAQ . . . . .	111
8.6	Performance and first data . . . . .	113
<b>9</b>	<b>Conclusions</b>	<b>119</b>
<b>A</b>	<b>Glossary and conventions</b>	<b>123</b>
	<b>List of Figures</b>	<b>126</b>
	<b>List of Tables</b>	<b>134</b>
	<b>Bibliography</b>	<b>136</b>
	<b>Acknowledgements</b>	<b>143</b>

# Chapter 1

## Introduction

At the Large Hadron Collider (LHC) protons are accelerated and collided at unprecedented centre-of-mass energies of up to  $\sqrt{s} = 13.6$  TeV and instantaneous luminosities of around  $\mathcal{L} = 10^{33} \text{ cm}^{-2} \text{ s}^{-1}$ .

Situated around one of the interaction points is the ATLAS detector experiment, which studies the physics processes based on the slew of secondary particles resulting from the pp collisions. The sheer amount of data produced by the detector from the high interaction rate and the limitations of mass storage technology make a trigger system an absolute necessity in ATLAS. The trigger system searches through the dominating QCD background to find and select rare physics events of interest.

The ATLAS Level-1 Trigger is a fast, hardware-based system that performs decisions based on partial detector information and reduces the event rate from 40 MHz down to 100 kHz. The Level-1 Calorimeter Trigger exploits the fast nature of calorimeter signals to analyse the energy deposits and build high-level trigger candidates based on the shower shape development.

As the operating conditions of the LHC grow harsher in Run-3 with higher energies and an increased number of pp interactions, the ATLAS trigger system is undertaking a major upgrade. Utilising novel Field Programmable Gate Array (FPGA) -based technologies, the fully digital trigger system implements sophisticated object-finding and clustering algorithms.

To supply the new trigger algorithms with digitised hadronic energy deposits from the Tile Calorimeter, the Tile Rear Extension (TREX) module for the L1Calo PreProcessor (PPr) has been developed and commissioned in ATLAS. Equipped with advanced FPGAs and high-speed optical transceivers, the TREX system provides digitised hadronic transverse energy results optically to the upgraded trigger processors every 25 ns. The data provided by the TREX improves the identification of the trigger candidates, leading to higher efficacy. In parallel, the system maintains compatibility with the existing trigger processors, used in Run-1 and Run-2.

The thesis discusses the development, testing, installation and commissioning of the TREX modules in ATLAS. Chapter 2 briefly describes the Standard Model. The LHC accelerator complex and the ATLAS detector are summarised in Chapter 3. Chapter 4 gives an overview of the ATLAS Trigger and Data Acquisition System. An overview of the new Level-1 Trigger system and its expected performance is given in Chapter 4. The functionality of the Tile Rear Extension modules is covered in Chapter 6. Chapter 7 presents the functional tests of the TREX modules. The installation in ATLAS and the operation of the final system are discussed in Chapter 8. The conclusions are presented in Chapter 9. A list of common naming conventions is available in the Glossary [A](#).

## Author's Contributions

As with large collaborations such as ATLAS, being a part of the Level-1 Trigger and calorimeter communities, the author received invaluable support and infrastructure to integrate the TREX modules in ATLAS. The work presented in this thesis has been carried out from July 2019 until November 2022.

The author's contributions started with developing test firmware and software for the prototype and pre-production TREX modules. Soon after, the author became the main firmware developer and system responsible, maintaining the designs for all FPGAs on the TREX and also on the PreProcessor Modules (PPMs)<sup>1</sup>. Throughout the thesis, the firmware functionality was developed in all aspects, while also preparing companion test and control software.

Multiple test procedures were designed and carried out for the TREX, which majorly contributed to the schematic and layout improvements for the final production version of the TREX hardware.

Once the final hardware modules arrived, all of the acceptance tests for the 40 boards and the assembly of the mechanics were conducted. The next contribution was the installation of the modules in ATLAS and configuring them to an operational state.

Most importantly, the author was responsible for fully integrating the TREX modules with the existing and upgraded DAQ and trigger systems by designing and improving the firmware functionality of all FPGAs.

In parallel, the author carried out the full development of the System-on-Chip firmware, operating system and the hardware monitoring framework.

In addition, as part of the ATLAS operations, the author became the main responsible and first responder regarding any issues of the PPr, for both PPMs and TREX systems. The contributions continued by analysing the first Run-3 collision data to validate the complete behaviour and functionality of the installed system.

---

<sup>1</sup>Abbreviations are available in the Glossary [A](#)



---

Overall, the author brought the TREX system from a concept to complete operation as part of the ATLAS detector in its entirety with stable and error-free running during pp collisions.

At the end of the thesis, the author also started developing the FPGA firmware and designing test routines for the hardware prototype of the Tile Calorimeter PreProcessor trigger interface (TDAQi) of the Phase-II upgrade for the High-Luminosity LHC. In relation to this, the Tile Calorimeter response to isolated hadrons using the Phase-II electronics in a testbeam environment was analysed.



# Chapter 2

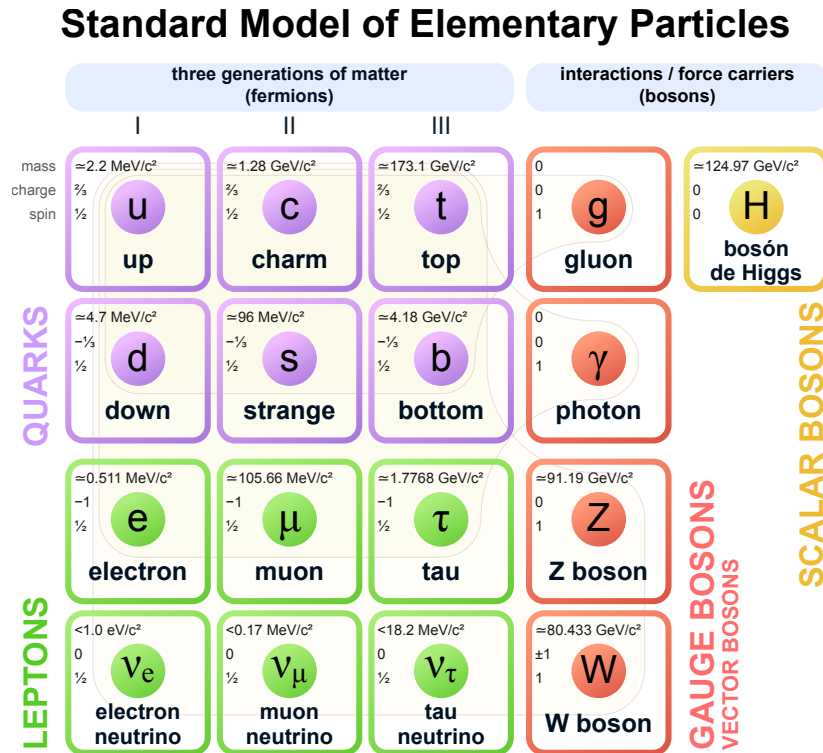
## Theoretical Background

### 2.1 The Standard Model of Particle Physics

Leptons, quarks, and gauge bosons are three different types of fundamental elementary particles that interact with one another in three different ways: electromagnetic (EM), weak, and strong interactions. This ensemble of quantum field theories is known as the Standard Model (SM). Quantum Electro-Dynamics (QED), which makes predictions that have been confirmed by a large number of experimental results up to extremely high precision, governs electromagnetic interaction. It is unified with weak interaction theory, and the two are frequently referred to as the Electro-Weak (EW) sector of the Standard Model. The interactions between quarks and gluons carrying colour charge are mediated via the strong force that is described by Quantum Chromo-Dynamics (QCD).

Together QCD and the Glashow-Weinberg-Salam (GWS) models form the SM, which is symmetric under the Poincaré-group of special relativity as well as the combined gauge groups:  $SU(3)_C \otimes SU(2)_L \otimes U(1)_Y$ , where  $C$  stands for colour,  $L$  for left-handed and  $Y$  for hypercharge.

Quarks are fermions with a spin- $\frac{1}{2}$  that carry a fractional electrical charge and colour. They take part in each of the three groups of interactions. The Six quarks up (u), down (d), charm (c), strange (s), top (t), and bottom (b) are divided into three generations. Only the u and d quarks from the first generation are the fundamental components of ordinary matter. They make up the neutrons (udd) and protons (uud) necessary to build any atomic nucleus. Each quark has an antiquark that has identical mass, but differs in quantum numbers such as electrical and colour charges, which are the opposite. All hadrons, including nucleons, unstable mesons, and hyperons, are made up of quarks and antiquarks.



**Figure 2.1.** The Standard Model of Elementary Particles [1]

Fig. 2.1 presents the list of particles included in the SM. For each particle the mass, electrical charge and spin values are given.

The leptons  $e$ ,  $\mu$  and  $\tau$  are also spin- $\frac{1}{2}$  fermions with an integer electrical charge and no colour. The three associated neutrinos,  $\nu_e$ ,  $\nu_\mu$ , and  $\nu_\tau$  are electrically neutral and only participate in weak interactions, whereas the charge leptons also partake in EW interactions. Leptons are also divided into three generations and have anti-leptons with an opposite electric charge. The only component of ordinary matter is the electron, which together with nuclei forms all atoms in the universe.

The carriers of the fundamental interactions are the gauge bosons. The electromagnetic interaction is transmitted via a massless spin-1 boson, the photon ( $\gamma$ ), whereas the weak interaction is transmitted via the exchange of the massive  $W^\pm$  and  $Z$  spin-1 bosons. The strong force is carried by eight gluons ( $g$ ), neutral, massless vector bosons that carry colour charges.

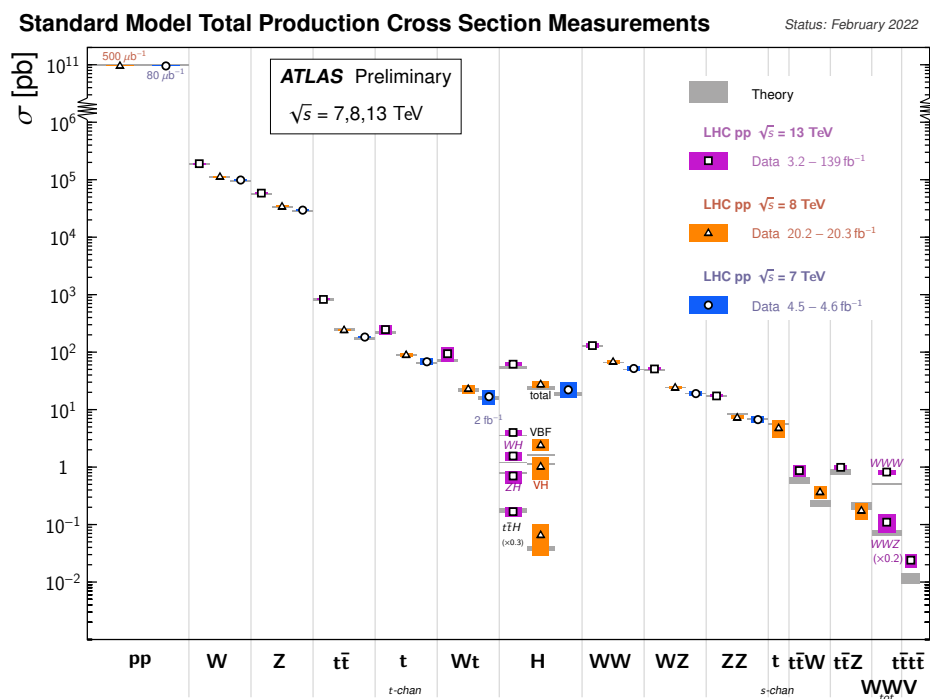
The Higgs-Englert-Brout mechanism of spontaneous symmetry breaking in the EW sector generates the masses of fermions (potentially excluding neutrinos) as well as the masses of the weak gauge- and the Higgs-boson. It was predicted back in the 1960s [2].

The Higgs boson associated with this measurement has been discovered by ATLAS [3] and CMS [4] at the LHC in 2012.

The experimental development of the SM has advanced quickly over the past few

decades. However, there are still fundamental questions left unanswered, such as the hierarchy problem, matter-antimatter imbalance, dark matter, etc. These are the subjects of ongoing research, and different Beyond-SM (BSM) extension models are under study which could explain these phenomena. No signs of BSM physics have been observed so far except for neutrino oscillations. Nevertheless, SM theory has done an excellent job of describing a plethora of experimental evidence.

Fig. 2.2 presents the comparison of total production cross-sections measured by the ATLAS detector and predicted by the SM. As the centre-of-mass energy increases, so does the production cross-section, which can be easily seen from the  $W$ ,  $Z$ ,  $t\bar{t}$  cross-sections. With Run-3, the additional recorded data is set out to enhance BSM searches and allow for high-precision measurements of SM parameters.



**Figure 2.2.** The total production cross-sections predicted by the SM and measured by the ATLAS detector [5].



# Chapter 3

## The LHC and the ATLAS Experiment

### 3.1 The Large Hadron Collider

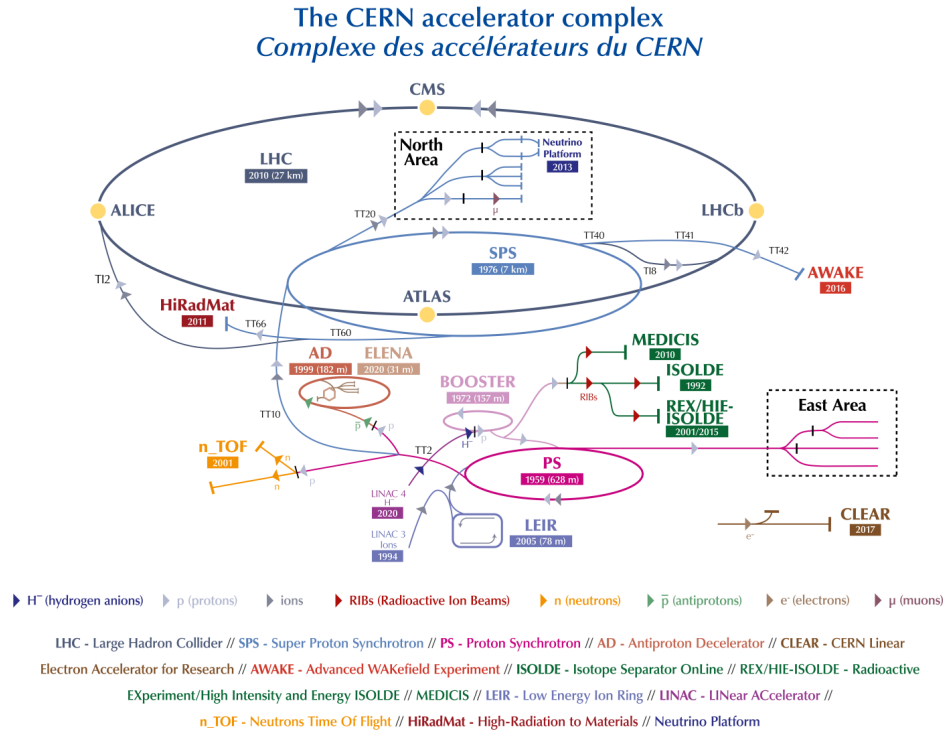
The Large Hadron Collider (LHC) [6] is the most powerful particle accelerator in the world, located at the European Organisation for Nuclear Research (CERN<sup>1</sup>) in Geneva, Switzerland. It is placed in a circular tunnel between 45m and 170m underground and boasts a circumference of 27 km. The LHC is designed for colliding beams of protons and heavy ions at unprecedented energies. It is an invaluable tool for probing a large phase-space in particle physics.

Two parallel beam pipes pass through 392 quadrupole and 1232 niobium-titanium dipole superconducting magnets, focusing and bending the particle beams in opposite directions. Additionally, several thousand corrector magnets are placed to precisely focus the beams near the interaction points. The beam acceleration is performed with 8 superconducting radio-frequency (RF) cavities operated at a frequency of 400.8 MHz, reaching record-breaking centre-of-mass energies of 13.6 TeV.

The LHC beams collide at four interaction points along the ring, where four major experimental detectors are located to measure the physics processes ensuing from the collisions. Two of the physics experiments are the general-purpose detectors A Toroidal LHC ApparatuS (ATLAS) [7] and Compact Muon Solenoid (CMS) [8]. They are designed for studying p-p collisions in the Terascale range. The A Large Ion Collider Experiment (ALICE) [9] detector is dedicated to heavy ion collisions, which allow for studying the physics of strong interactions and quark-gluon plasma. The fourth experiment, called the Large Hadron Collider beauty (LHCb) [10], is designed for precision measurements of CP violation, B-meson decays and matter-antimatter asymmetries.

---

<sup>1</sup>Derived from Conseil Européen pour la Recherche Nucléaire



**Figure 3.1.** The CERN accelerator complex [11].

Fig. 3.1 illustrates the series of accelerator machines with progressively higher energies. The source of the protons is negatively charged hydrogen ion gas, which is stripped of its electrons through an electric field and accelerated by the LINAC4 linear accelerator. The beam of protons is then passed through multiple pre-acceleration circular accelerators, first through the BOOSTER, then the Proton-Synchrotron (PS) and finally the Super Proton-Synchrotron (SPS). The SPS brings the energy up to 450 GeV, at which point the beam is injected into the LHC ring.

The Run-1 period of the LHC beam operation began in 2010 with a ramp-up of the beam energy to 3.5 TeV. The beam operated at the same energy until November of 2011. In 2012 the beam energy increased to 4 TeV and delivered collisions until the end of Run-1, which was December 2012. After successful data-taking, the LHC accelerator complex was shut down for upgrades and maintenance that lasted until 2015. During this time, the detectors prepared for Run-2 with upgrades of their own. In 2015, Run-2 started successfully with LHC beam energies of 6.5 TeV. The machine operated until the end of 2018, delivering collisions far beyond its design. To prepare for Run-3 of the LHC, the detectors started an upgrade campaign in 2019. For ATLAS, it is referred to as the Phase-I upgrade. This upgrade went on for three years until July of 2022 when Run-3 began. Run-3 is planned to operate for four years until 2026, after which the upgrade for High-Luminosity LHC will commence, referred to as Phase-II.



LHC Runs	Operating years	ATLAS Upgrade
Run-1	2010-2012	-
Run-2	2015-2018	Phase-0 (LS1)
Run-3	2022-2026	Phase-I (LS2)
Run-4+	2029-	Phase-II (LS3)

**Table 3.1.** The naming conventions of LHC runs, the years of operation and the corresponding ATLAS upgrade phase.

### The LHC bunch structure

The two LHC proton beams are ordered into bunches containing up to  $10^{11}$  particles. They circulate at a frequency of  $f_{orbit} = 11.245$  kHz, where one revolution around the LHC is called an *orbit*. The nominal bunch spacing is 25 ns, which corresponds to a  $f_{BC} = 40.08$  MHz bunch clock. The harmonic number of the LHC constrains the number of possible slots, where a bunch can be assigned:

$$h = \frac{f_{BC}}{f_{orbit}} = 3564 \quad (3.1)$$

This means that per orbit, there are 3564 potential *Bunch-Crossing* (BC) slots where a collision between the opposite travelling beams can occur. Each BC is assigned with a unique number from 0 to 3563, commonly referred to as *Bunch-Crossing Identifier* (BCID). Even though there are 3564 BCs, the fill pattern is not necessarily uniform. There are BCs where both, one or neither bunches of the two beams are *filled*. The maximum occupancy of the filled bunches is 2808 within one orbit. This is due to operational restrictions and standards that permit steady and secure beam operation. A gap of 120 BCs is dedicated to the beam extraction kicker magnet to extract the beam out of the LHC. This occurs at the end of the orbit and is called the abort gap. There are further gaps for calibration purposes and the LHC ensures none of the bunches within this gap contain protons.

During data-taking, the protons are typically distributed in a train structure with alternating filled and empty bunches. When the bunches of both beams are filled for a particular BC, then this BC is called *paired*. When only one beam contains protons, that BC is called *unpaired*.

### Luminosity and Pileup

The performance of the LHC is quantified through the delivered Luminosity and the  $\sqrt{s}$  centre-of-mass energy. The luminosity can be defined as a first approximation:

$$\mathcal{L} = \frac{f_{orbit} n_b N^{(1)} N^{(2)}}{4\pi\sigma^{(1)}\sigma^{(2)}} \quad (3.2)$$

where  $N^{(1)}$  and  $N^{(2)}$  are the number of protons in each bunch and  $n_b$  is the number of filled bunches. The transverse beam sizes of each beam at the interaction point are  $\sigma^{(1)}$  and  $\sigma^{(2)}$ .

The number of collision events can be expressed as a function of the luminosity and the cross section for a given physics process. Taking the total cross section  $\sigma_{tot}$  for any SM pp interaction, the event rate will be:

$$\frac{dN}{dt} = \sigma_{tot} \mathcal{L} \quad (3.3)$$

Therefore, the mean number of interactions per collision  $\langle \mu \rangle$  can be expressed as:

$$\mu = \frac{\sigma_{tot}}{n_b f_{orbit}} \mathcal{L} \quad (3.4)$$

Through the LHC operation runs the mean number of interactions per collision has varied from an average of  $\langle \mu \rangle \simeq 13$  up to  $\langle \mu \rangle \simeq 33$ . This indicates that the hard scatter event is contaminated with additional soft interactions, which is known as pileup.

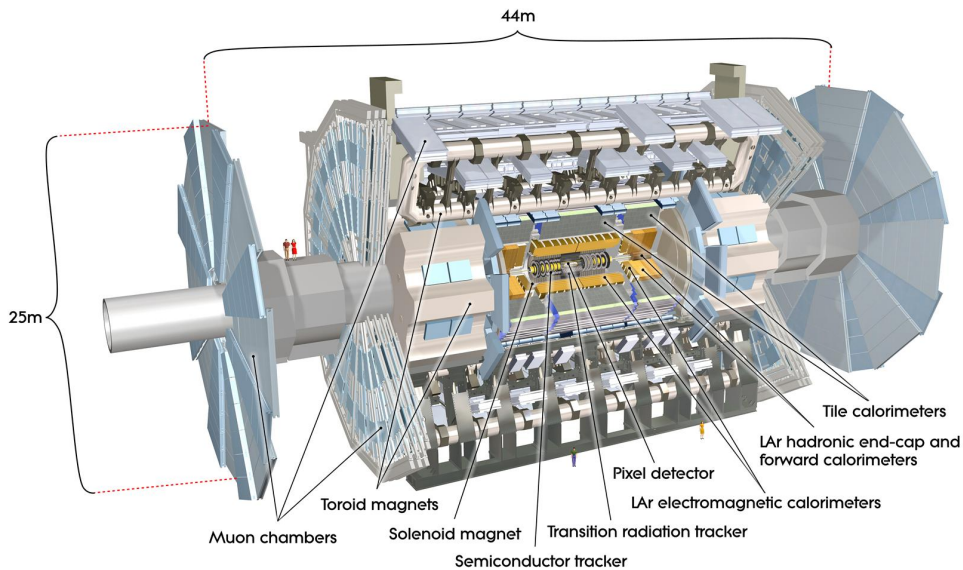
## 3.2 The ATLAS Detector

The ATLAS detector is a general-purpose detector located in one of the four experimental caverns 100 m below ground. It consists of a cylindrical *barrel* around the centre of the interaction point, creating a rotational symmetry around the beam axis. Two endcaps are attached to the barrel, one on each side.

The detector itself contains multiple layers of sub-detectors. Starting from the interaction point, the first subsystem is the Inner Detector, dedicated to particle track reconstruction and momentum measurements.

The next subsystems are the Electromagnetic and Hadronic Calorimeters, which measure the energy and position of the incident particles. The Muon Spectrometers surround the calorimeter system, composed of toroidal air cores and gaseous chambers.

An overview of the ATLAS detector is depicted in Fig. 3.2. It has a length of 46 m and a diameter of 25 m. The weight of the entire detector is approximately 7000 tonnes.



**Figure 3.2.** The ATLAS Detector [12].

### 3.2.1 Coordinate System

The coordinate system used in ATLAS is right-handed with the p-p interaction point taken as its origin. The  $z$ -axis points in the direction along the beamline, while the  $x$ -axis is pointing radially towards the centre of the LHC ring and the  $y$ -axis points in the vertical direction. The azimuthal angle  $\phi$  spans from 0 to  $2\pi$  in the transverse  $x$ - $y$  plane and the polar angle is defined from 0 to  $\pi$  relative to the  $z$ -axis. In particle detectors, typically, the pseudorapidity is used instead of the polar angle to describe the angle of the particle relative to the beam:

$$\eta = -\ln \left( \tan \frac{\theta}{2} \right). \quad (3.5)$$

At high energies, where the mass of the particle becomes negligible ( $m \ll \mathbf{p}$ ), the pseudorapidity approximates to the rapidity:

$$y = \frac{1}{2} \ln \left( \frac{E + p_z}{E - p_z} \right), \quad (3.6)$$

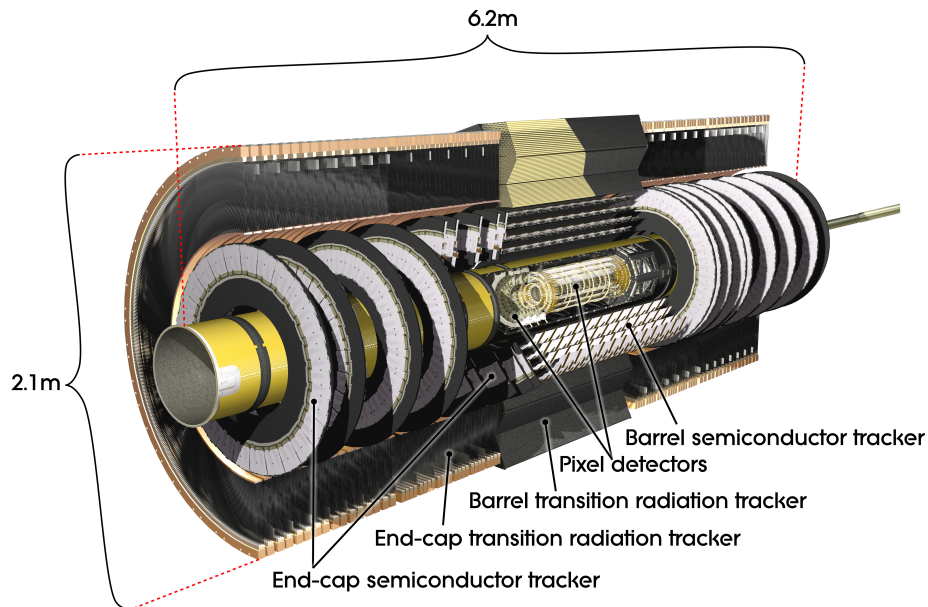
where the  $p_z$  is the momentum component in the  $z$ -direction.

### 3.2.2 Inner Detector

The Inner Detector is the first point of detection in ATLAS, which is used for track reconstruction of the charged particles produced by the p-p collisions and for identifying the primary collision vertex along with decay vertices of short-lived particles. It is immersed

in a 2 T magnetic field parallel to the beamline, which curves the path of the entering particles and allows to measure their momenta.

Fig. 3.3 shows an overview of the Inner Detector, which is made up of three different detector layers. In the barrel region, the layers are stacked up in concentric cylinders around the beam axis, while in the endcap region the detectors are arranged in disks perpendicular to the z-axis. The inner layer houses two high-resolution detectors, the Pixel Detector and the Semiconductor Tracker (SCT), while the outer layer is made up of the Transition Radiation Tracker (TRT). The acceptance for the Pixel and SCT detectors is  $|\eta| < 2.5$ , while for the TRT it is  $|\eta| < 2.0$ .



**Figure 3.3.** A cross-section of the ATLAS Inner Detector depicting the Pixel Detector, Semiconductor Tracker and the Transition Radiation Tracker [13].

The Pixel Detector is closest to the interaction point, just 3.3 cm away from the beam pipe. It consists of 4 layers of silicon pixels, with a pixel size of  $50 \times 400 \mu\text{m}^2$  in the outer layer and  $50 \times 250 \mu\text{m}^2$  in the innermost layer. Particles traversing through the pixels cause ionisation that is localised in the medium. The produced signals are used to measure the position of the trajectory with an intrinsic precision of around  $10 \mu\text{m}$ . With over 92 million channels, the Pixel Detector offers very high-granularity measurements of the vertex resolution in order to separate between production and decay vertices.

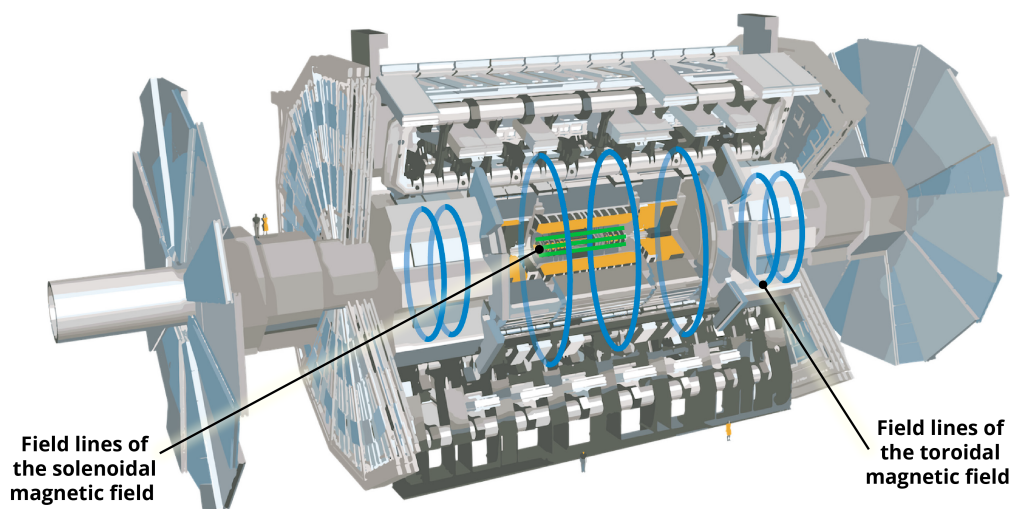
The SCT surrounding the Pixel Detector is used for track detection and reconstruction. It provides radially eight points in space per track, which add to the measurements of the particle momentum, impact parameter and vertex position. It is made up of over 4088 two-sided modules and over 6 million *micro-strips*, which provide a precision of up to  $25 \mu\text{m}$ .

The final layer of the Inner Detector is the TRT. It consists of 350000 thin-walled

drift tubes or straws. Each of the straws has a surface diameter of  $4\text{ mm}$ , made of conductive-coated Kapton. A  $30\ \mu\text{m}$  gold-plated tungsten wire is placed in the centre and the straw is filled with a gas mixture of 70% Xe, 27% CO<sub>2</sub> and 3% O<sub>2</sub>. The TRT combines the principle of ionisation and the transition radiation effect for particle identification and track reconstruction. As a relativistic particle crosses the boundary between materials with different dielectric constants, it emits photons proportional to the Lorentz factor. The wavelength of the radiated photons is proportional to the mass of the incident particle, which allows differentiation between lighter and heavier particles.

### 3.2.3 Magnet System

The ATLAS magnet system consists of four superconducting magnets, that bend the trajectories for particle identification and momentum measurements. The central solenoid magnet provides a powerful magnetic field of 2 T in the barrel region. The three outer toroid magnets supply the Muon Spectrometer with a field between 0.5 T and 1 T in magnitude.



**Figure 3.4.** The ATLAS Magnet System. The solenoidal magnetic field lines are illustrated in green, the toroidal magnetic field lines in blue [14].

Fig. 3.4 illustrates the magnetic field lines within the ATLAS detector. The magnetic field from the solenoid, shown in green, is contained in the central barrel region between the Inner Detector and the electromagnetic calorimeter, while the toroidal fields are present in the outer barrel and endcap regions of the muon spectrometer, forming a cylindrical magnetic field around the beam pipe.

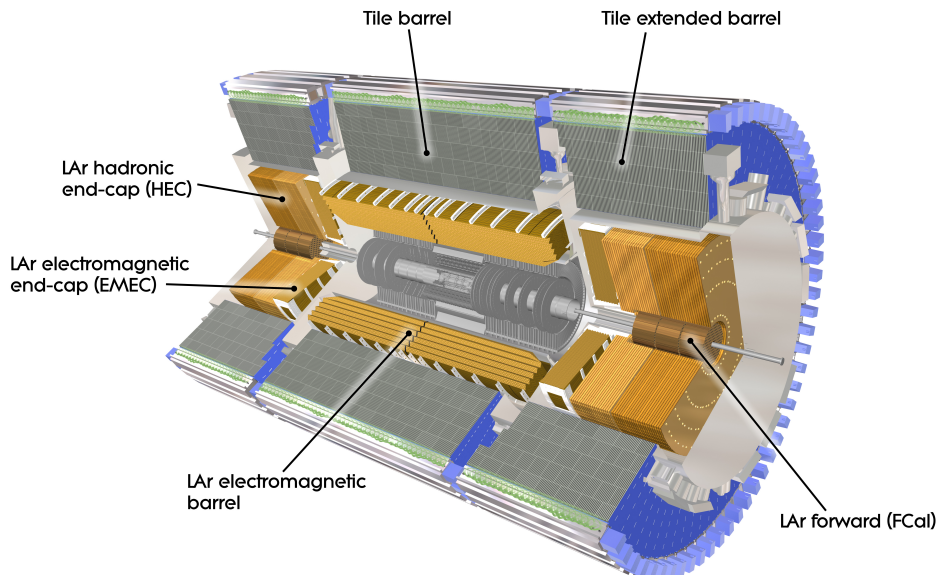
### 3.2.4 Calorimetry

ATLAS employs calorimetry to measure the energies and positions of charged and neutral particles produced by the colliding protons. The calorimeter system is composed of an electromagnetic and a hadronic calorimeter. The electromagnetic calorimeter is designed for containing electrons and photons through their electromagnetic interaction, such as bremsstrahlung and pair-production, while the hadronic calorimeter measures the energy of mostly hadrons through electromagnetic and strong interactions.

Both calorimeters are sampling calorimeters made up of alternating layers of dense material, called absorbers, and an active medium which produces a signal that can be detected. The incident particle interacts with the absorber material, lowering the energy and producing showers of secondary particles. To account for the energy loss in the absorber material, a *sampling fraction constant* is calculated during beam tests with electron and pion beams.

The sampling calorimeters have a lower energy resolution compared to homogenous calorimeters, where the whole calorimeter is composed of an active medium. However, sampling calorimeters have an advantage in the longitudinal and lateral segmentation, offering much higher spatial resolution and better particle identification through the shower shape.

Due to the non-compensating nature of the calorimeters, the overall calorimeter response for hadrons is lower than for particles interacting purely electromagnetically.



**Figure 3.5.** The ATLAS Calorimeters. LAr is shown in yellow-gold and the Tile in grey colour [15].

An overview of the ATLAS calorimeters is presented in Fig. 3.5. The electromagnetic



and hadronic calorimeters cover the region of  $|\eta| < 4.9$  and a full  $2\pi$  in  $\phi$ .

### The Liquid Argon Calorimeter

ATLAS uses lead/liquid-Argon (LAr) [16] for the barrel electromagnetic barrel (EMB) and endcap (EMEC) calorimeter, while the hadronic endcaps (HEC) use copper and the forward calorimeters (FCAL) use tungsten plates as absorbers. The barrel region covers a range of  $|\eta| < 1.475$  while the endcaps cover  $1.375 < |\eta| < 3.2$ . The HEC covers  $1.5 < |\eta| < 3.2$  and FCAL  $3.1 < |\eta| < 4.9$ .

The accordion-shaped lead absorber plates in the central barrel region provide full azimuthal symmetry without cracks in  $\phi$ . The volume between the lead absorber plates is filled with liquid argon and submerged within are Kapton-Copper electrodes. The induced electromagnetic showers ionise the LAr atoms. The electrodes placed within the active medium collect the ionisation charges which produce a detectable signal.

### The Tile Calorimeter

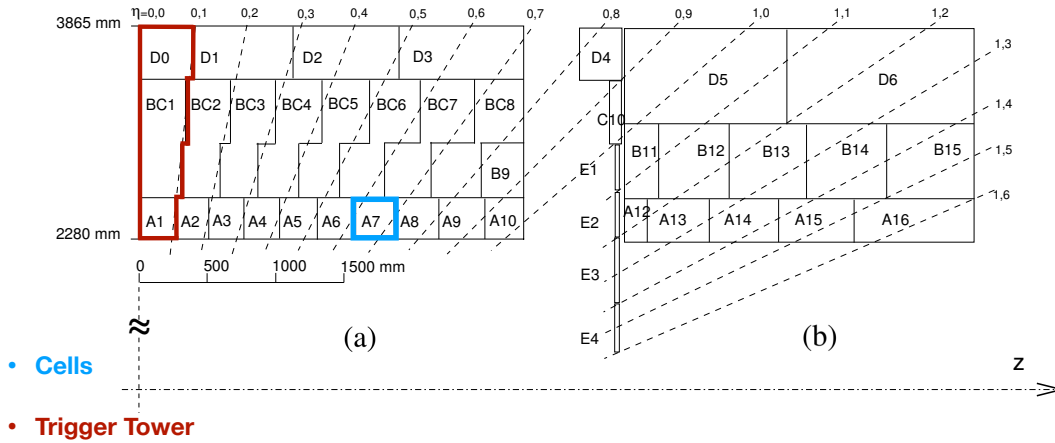
The Tile hadronic calorimeter (TileCal) [17] is situated in the central region of the detector, divided into a long barrel (LB) and an extended barrel (EB) modules on each side. The LB has a coverage of  $|\eta| < 0.8$ , while the EBs cover  $0.8 < |\eta| < 1.7$ . Both the LB and the EBs consist of 64 trapezoidal wedges with full  $\phi$  coverage, therefore each module has an azimuthal size of  $\Delta\phi = 0.1$ .

TileCal uses steel as absorbers and scintillating tiles as the active medium. The showers induced by the incident particles cause light production in the scintillators. The light is collected from opposite sides of each tile via two wavelength-shifting fibres and directed to a pair of photo-multiplier tubes (PMTs).

The PMTs convert the photons to an amplified electrical signal that is directly proportional to the energy deposited in the scintillator. The segmentation of each Tile module is done by grouping the wavelength-shifting fibres of corresponding tiles into two bundles, resulting in a cell granularity of  $\Delta\eta = 0.1$

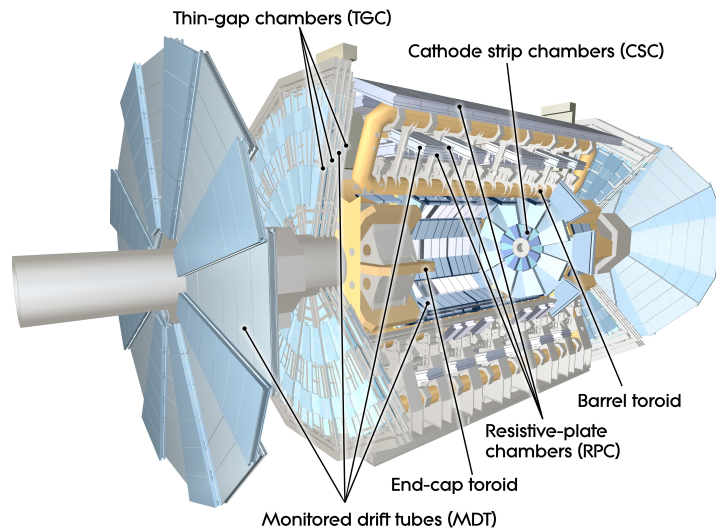
Radially, the barrel is segmented into 3 sampling depths with interaction lengths of 1.4, 3.9 and 1.8. The fast PMT signals are amplified and shaped on-detector, in order to mitigate aliasing effects during digitisation.

For the Level-1 Calorimeter trigger, the analogue PMTs signals of cells are summed radially, creating reduced granularity trigger-towers and sent to the PreProcessor system, which will be discussed in Chapter 4. Fig. 3.6 illustrates the TileCal segmentation, where a cell is outlined in blue, while a trigger-tower is shown in red.



**Figure 3.6.** Segmentation of the Tile Calorimeter for the Long Barrel (a) and the Extended Barrel (b) [18].

### 3.2.5 Muon Spectrometers



**Figure 3.7.** A cut-out of the ATLAS detector illustrating the Muon Spectrometers [19].

The ATLAS Muon Spectrometers form the last layer of the detector and are divided into two groups of gaseous detectors.

The first group consists of Monitored Drift Tubes (MDTs) and Cathode Strip Chambers (CSCs) for measuring the momenta of the particles that manage to pass through the calorimeters, with their trajectories curving under the toroidal magnetic fields.

The second group is made up of Resistive Plate Chambers (RPCs) in the central barrel region and Thin Gap Chambers (TGCs) in the endcaps. The fast gaseous detectors measure the particle coordinates for triggering purposes.

Fig. 3.7 presents the four different types of muon detectors in ATLAS. The high-precision tracking MDTs and CSCs cover the region of  $|\eta| < 2.7$ . The RPCs and the TGCs, used for the trigger system, have an acceptance of  $|\eta| < 2.4$ .



# Chapter 4

## The Trigger and Data Acquisition of ATLAS

The ATLAS detector produces a tremendous amount of data at a rate of 40 MHz and an event size of roughly 1.5 MBs. Storing all of the data is impossible due to bandwidth and storage limitations, thus a trigger system is implemented to reduce the rate by selecting events with physics signatures of interest.

The Trigger and Data Acquisition (TDAQ) system [20] of the ATLAS experiment is tasked with treating the huge data flow produced by the subdetectors. The trigger system consists of a hardware-based first-level (L1) trigger and a higher-level software-based filter. The Level-1 Trigger is based on custom and commercial electronics and processes the data with strict latency requirements.

Each sub-detector front-end contains a pipelined memory buffer, where the incoming event data is placed at a rate of 40 MHz. These buffers are large enough to store approximately  $2.5 \mu s$  worth of data, hence the trigger decision from the Level-1 must arrive within that time constraint. The data remains in the pipeline and awaits a trigger decision by the Central Trigger Processor (CTP) [21] in the form of a Level-1 Accept (L1A) signal that is distributed to all sub-detectors. The maximum rate of the L1A is around 100 kHz, causing 400 times reduction in the rate.

After the event passes the conditions imposed by the L1 Trigger System, the DAQ system is responsible for aggregating the transfer of event data from the detector front-ends, including the L1 processing hardware, to the HLT where the rate is further reduced to 1 kHz and written to permanent storage.

Fig. 4.1 outlines the ATLAS TDAQ system. It can be divided into four parts, the Level-1 Trigger, Detector Readout, the High-Level-Trigger and the DataFlow. The rectangles encircled in red showcase new and upgraded hardware and software added for Run-3.

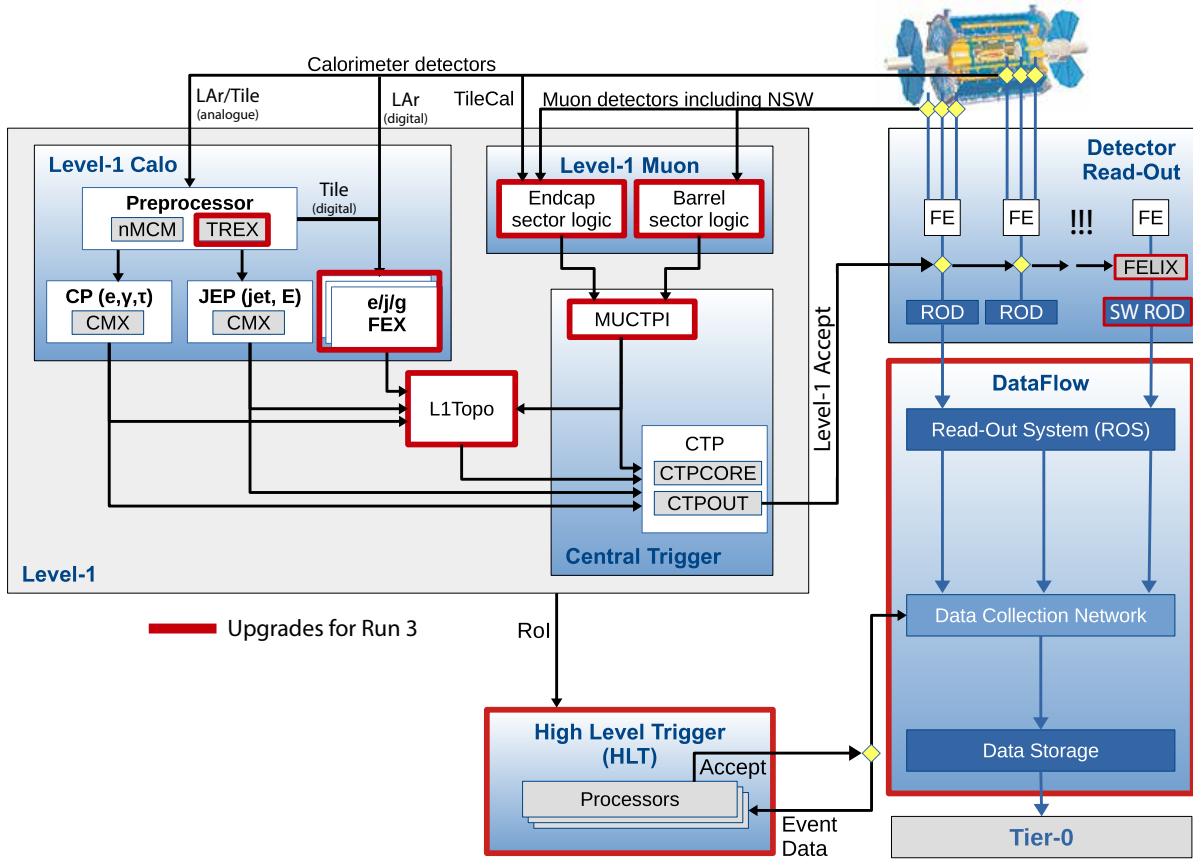


Figure 4.1. Overview of the ATLAS TDAQ system, taken and modified from [22].

## 4.1 The Level-1 Trigger System

The ATLAS Level-1 Trigger is made up of multiple sub-systems that search for events with high-energy electrons, photons, muons and jets. They use reduced granularity data from the calorimeter and muon systems to perform fast identification of various physics signatures. Once the objects have been identified, their information is collected and sent to the CTP, where the results are compared to 256 pre-defined trigger conditions.

For Run 3, both the calorimeter- (L1Calo) and muon-based (L1Muon) trigger systems have undergone major upgrades to improve the trigger efficiency. The upgrade of the L1Calo trigger is presented in Chapter 5.

### 4.1.1 The Level-1 Muon Trigger

The L1Muon trigger uses the RPC and TGC chambers from the Muon Spectrometer system to identify muon candidates and their transverse momenta. The  $p_T$  of the muon candidate is compared to six programmable thresholds and assigned to the correct bunch crossing. Each of the 208 muon trigger sectors transmits data on muon candidates to the ATLAS Muon-to-CTP interface (MUCTPI) [23], which then determines the muon candidate multiplicity for each of the six transverse momenta ( $p_T$ ) thresholds and delivers

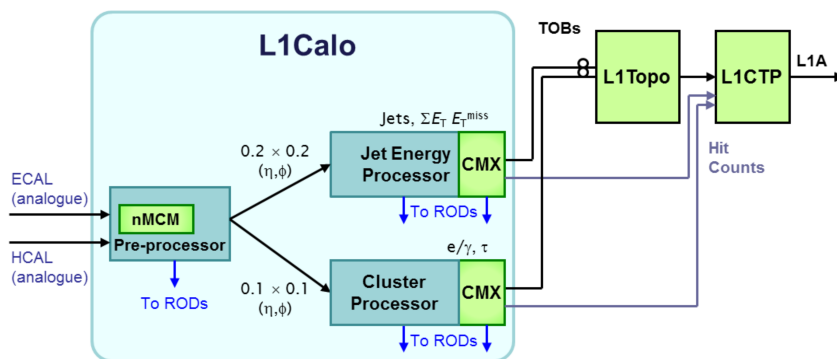
it to the CTP.

### 4.1.2 The Level-1 Calorimeter Trigger

The ATLAS Level-1 Calorimeter (L1Calo) Trigger [24] is a pipelined system that receives around 7200 analogue signals from the electromagnetic and hadronic calorimeters. Both calorimeters have about 200000 cells in total. In order to process the full calorimeter granularity, it would require a large amount of cabling and an increase in the scale of the L1Calo system. Therefore, a reduced-granularity trigger-tower approach is used that lowers the number of input channels to about 7200.

The analogue pulses are digitised, synchronised and calibrated to transverse energy  $E_T$ . Based on the calorimeter  $E_T$  particle candidates are formed and their information is passed to the Level-1 Topological Trigger (L1Topo) and the CTP.

This section covers the legacy trigger system used in Runs 1,2 and also in Run-3. The upgrade system runs alongside the legacy, which is discussed in Chapter 5.



**Figure 4.2.** Overview of the L1Calo System in Run-2 [20].

Fig. 4.2 depicts the data flow through L1Calo. The green colour indicates upgrades performed in LS1, in preparation for Run-2.

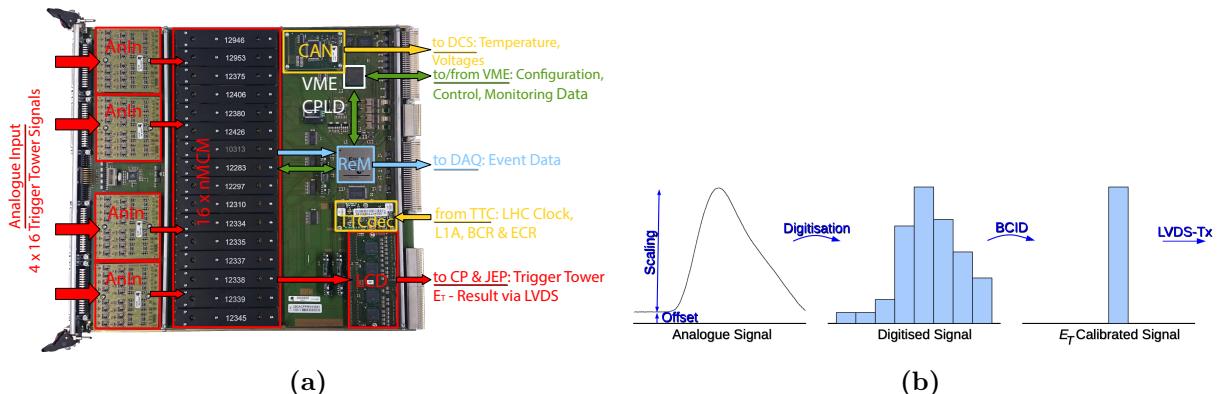
#### PreProcessor

The L1Calo PreProcessor system is composed of 124 PreProcessor Modules (PPM) organised into eight VME<sup>1</sup> crates. Six crates process the analogue signals from the LAr calorimeter, while the other 2 are dedicated to Tile.

The PPM is a highly modular system, a base plane (carrier) holds 16 so-called Multichip-module (MCM) mezzanines. While the carrier itself remained unchanged since Run-1, the MCMs were upgraded for Run-2, called the new Multichip-Modules (nMCMs) [25], from an Application Specific Integrated Circuit (ASIC)-based design to an FPGA. The choice of the FPGA was a Xilinx Spartan-6 [26] with plenty of room to

<sup>1</sup>Versa Module Europa

accommodate the ASIC design and add a new layer of functionality improvements.



**Figure 4.3.** The PreProcessor Module in Run-2 (a) and its functionality (b), taken and modified from [27].

The calorimeter signals arrive in the form of bipolar differential signals. To prepare them for digitisation, first the signals are conditioned and converted to single-ended via the so-called An-In boards. Once the signals are ready to be digitised, they're used as an input for the nMCMs.

Every nMCM receives 4-channels as input, each of them corresponding to a trigger-tower covering a region of  $0.1 \times 0.1$  in  $\eta \times \phi$ . The digitisation is performed with 10-bit flash Analog-to-Digital Converters (ADCs) at a sampling frequency of 80 MHz, synchronous to the LHC clock. An adjustable fine-timing scheme allows the sampling clock phase to be shifted with steps of 1.042 ns.

After digitisation, a coarse timing procedure aligns the data and compensates for the signal propagation time due to cable length differences between the various parts of the calorimeter. It is realised through a First-In-First-Out (FIFO) memory buffer that has a maximum programmable depth of 16 Bunch-Crossings (BCs).

Once the ADC output data is aligned, a bunch-crossing identification logic assigns the energy deposition to the corresponding LHC bunch-crossing. The algorithm is performed through the use of Finite Impulse Response (FIR) Filters, where weighted sums are built over five ADC samples at a 40 MHz frequency<sup>1</sup>:

$$f_i = \sum_{j=0}^5 c_j A_{i-2+j} \quad (4.1)$$

where the  $c_j$  are the coefficient weights,  $A_i$  is the ADC sample at the given BC and  $f_i$  is the filter output at BC  $i$ . To calculate the filter output, the ADC samples of the two previous and two next BCs are used. The set of coefficients used corresponds to an

<sup>1</sup>Every other sample is taken from the digitisation at 80 MHz

Autocorrelation Filter scheme, following the pulse shape, where the signal-to-noise ratio is increased for the peak sample.

The peak finder algorithm [28] monitors the output of the FIR filters for a local maximum condition:

$$f_{i-1} < f_i \geq f_{i+1}. \quad (4.2)$$

When the condition is met, the FIR filter output is classified as the amplitude of the reconstructed pulse. Through the use of a programmable Look-Up-Table (LUT) the filter output is calibrated to transverse energy  $E_T$  in the electromagnetic scale. The resulting  $E_T$  value is 8-bits in length, with a range from 0 to 256 GeV and a resolution of 500 MeV per LSB for electromagnetic candidates.

The transverse energy, resulting from the bunch-crossing identification, is transmitted to the Cluster and Jet-Energy Processors for further processing, via electrical Low-Voltage Differential Signals (LVDS).

The upgrade of the PPr for Run-3 is described in Chapter 6.

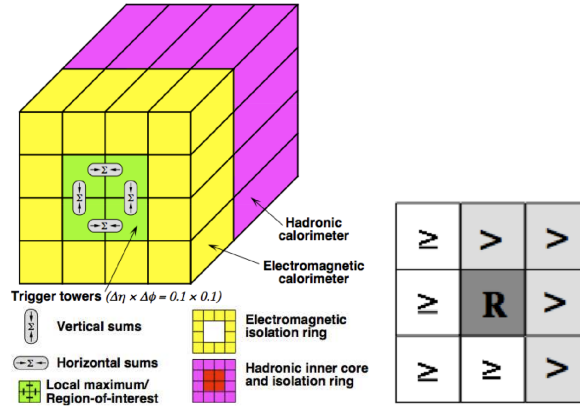
## Cluster Processor

The CP system consists of 56 Cluster Processor Modules (CPM), distributed into four VME crates.

Energy clusters deposited in the calorimeters that correspond to isolated electron, photon, or hadronically decaying tau candidates are identified through the CP system. The CP system receives the digitised 8-bit  $E_T$  results from the PPr with a resolution of  $\Delta\eta \times \Delta\phi = 0.1 \times 0.1$ . Two algorithms are used to perform the identification, one for  $e/\gamma$  and one for  $\tau/hadrons$ . Both algorithms are based on a  $4 \times 4$  trigger-tower window in the electromagnetic and hadronic layers. The window contains a *core*, made up of the central  $2 \times 2$  towers, surrounded by an isolation ring. Within the core, four possible  $E_T$  sums are computed with neighbouring trigger-towers and evaluated against programmable thresholds.

The algorithm looking for  $e/\gamma$  candidates applies a selection where the induced shower is localised within two neighbouring trigger-towers and no energy deposits are detected in the hadronic layer. Here, the isolation ring is used to identify narrow electromagnetic showers. The  $\tau/hadrons$  algorithm uses the same core as the other algorithm but adds the contribution from the core of the hadronic layer.

In order to avoid double counting the same trigger object  $E_T$  sums of both the electromagnetic and hadronic cores are compared to eight equally sized areas within the  $4 \times 4$  window. After comparing all 8 regions, the core with the local maximum is identified as the trigger object and the corresponding  $\eta - \phi$  coordinate is sent to the HLT as a Region of Interest (RoI).



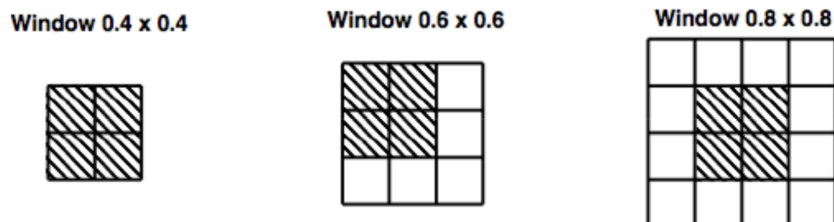
**Figure 4.4.** The CP algorithm (left) depicting the core (local maximum) and the isolation and the definition of the RoI (right) [24].

Fig. 4.4 illustrates the CP trigger algorithms on the left-hand side, where the core region is depicted in green and the isolation ring in yellow. The right-hand image presents the eight neighbouring regions where a local maximum is searched for.

### Jet-Energy Processor

The JEP system is made up of 32 Jet/Energy Processor Modules (JEM) within two VME crates.

The JEP algorithms search for jets and calculate sums such as the total, missing and jet-sum transverse energies. The data received from the PPr is a coarser 9-bit jet-sum with a resolution of  $\Delta\eta \times \Delta\phi = 0.2 \times 0.2$ . The JEP employs a sliding window algorithm, similar to the CP, with 3 different sizes and moves within the calorimeter coverage by single steps in  $\eta - \phi$ . At each step, jet-elements are built by summing the jet-sums with sizes of  $2 \times 2$ ,  $3 \times 3$  and  $4 \times 4$  and the summed  $E_T$  is then compared to programmable thresholds. In this case, the RoI is also based on the region with a local maximum and sent to the HLT.



**Figure 4.5.** The three sizes of the sliding window of the JEP algorithm [24].

Fig. 4.5 depicts the three sliding window sizes employed by the JEP algorithm.

### Level-1 Topological Trigger

Particle candidates identified by the L1Calo and L1Muon systems are sent to the Level-1 Topological (L1Topo) trigger systems, where selection criteria, based on the event topology, are used to make a trigger decision. The criteria include cuts to the invariant mass or to the angular spacing between particle candidates.

## 4.2 The Central Trigger Processor

The final step in the Level-1 trigger system's processing is the Central Trigger Processor (CTP). It gathers digital trigger signals from several forward detectors, the MUCTPI, L1Calo, L1Topo and time-aligns them. In addition, it manages deadtime, prescales, and implements the trigger logic setup in accordance with the physics trigger menu.

The CTP applies dynamically adjustable prescales to each trigger item in order to regulate rates, especially for trigger items which have low energy thresholds. A prescale factor of  $n$  means that only every  $n$ 'th event of that trigger item is accepted. This serves as a reduction factor of the trigger rate. Unprescaled trigger items are particularly significant since the adoption of a prescale imparts a statistical bias to the trigger selection. Unprescaled triggers with the lowest energy thresholds are primarily used as the main trigger during data analysis.

Furthermore, the CTP receives the timing signals from the LHC machine, such as the 40.08 MHz clock and distributes them to all detector sub-systems through the network of ATLAS Local Trigger Interface (ALTI) [29] modules.

The ALTI system is also used to distribute the trigger signal generated by the CTP, called the Level-1 Accept signal, and other Timing, Trigger and Control (TTC) commands to the sub-detectors.

## 4.3 The DAQ and the Higher-Level Trigger system

Upon the issuance of an L1A, all of the detector data that was previously stored in pipeline memories is moved to a different set of storage elements called read-out buffers (ROBs), where it remains until it is validated or ignored by the HLT.

For Run 3, the ATLAS readout system is upgraded with two main components [30]. The first is the Front-End Link EXchange (FELIX) system, which is made up of PCIe FPGA cards and high-speed network interfaces, housed in commodity servers. The FELIX system acts as a router between the detector sub-systems and the switched Data Collection Network. The second component is the Software ReadOut Driver (SW ROD),

which performs data aggregation, buffering, detector-specific processing, and routing to the HLT.

The HLT consists of a computing farm using commodity servers. There are around 40000 physical cores which analyse the events triggered by L1 and reduce the output rate further to around 1.5 kHz. The filtered events are then written to mass storage. The HLT software has seen a massive upgrade for Run 3, in particular, the software is now multi-threaded, which brings improvements to memory consumption and performance. [31]



# Chapter 5

## The Upgrade of the Level-1 Calorimeter Trigger

### 5.1 Motivation for the upgrade

The ATLAS recorded integrated luminosity is estimated to be at  $300 \text{ fb}^{-1}$  by the end of Run-3. The collected dataset will allow for precision measurements in the Higgs production and for many Standard Model physics processes. It will also offer a large phase-space for Beyond-Standard Model (BSM) searches. To enrich the recorded data with a wide coverage of physics processes, the L1 Trigger must be able to search and identify many interesting physics signatures with high efficiency.

The decays of electroweak particles like the W, Z, and Higgs bosons, which have masses between 80 and 125 GeV, span a significant portion of the trigger phase space. These electroweak particles primarily decay to leptons and jets, producing detector objects that typically have a transverse momentum greater than 30 GeV. The most useful Level-1 trigger signatures for electroweak-scale particles are that of isolated electrons and muons, which predominantly originate from the decay products of W and Z bosons.

One of the main objectives of the Phase-I upgrade is to maintain low- $p_T$  trigger thresholds at Level-1 for single electrons and muons. Maintaining unrescaled Level-1 triggers on the low- $p_T$  objects will increase the data samples for high precision and new physics measurements. It will also yield data samples that can be studied in more detail, leading to physics results with low systematic uncertainty.

The ability to perform physics measurements with electroweak-scale particles that result in jets, missing transverse momentum and hadronically decaying tau leptons is also a key objective of the Phase-I upgrade. In comparison to muons and electrons, signatures of these objects are less distinct due to their shower shapes within the calorimeter, discussed further in this chapter. In order to achieve rates that can be supported by the ATLAS TDAQ system, it is preferable to combine numerous trigger objects at Level-1

(including very low- $p_T$  electrons and muons, if suitable) in order to trigger on these decay products of electroweak-scale particles effectively.

## 5.2 New L1Calo Trigger system

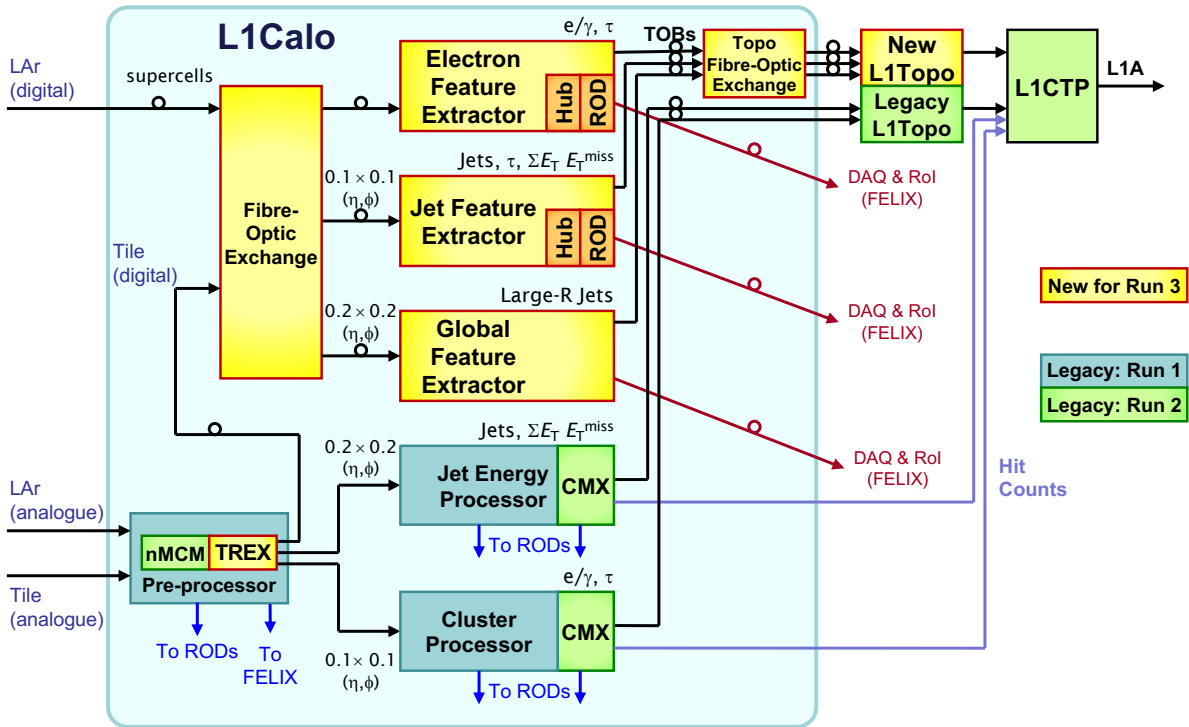
The original L1Calo design, using over 7000 analogue inputs of coarse granularity calorimeter information, was very successful in the first 10 years of LHC operation, which already provided integrated luminosities beyond design.

However, the expected higher luminosities in future LHC operation will compromise the efficacy of the original hardware, and so an upgrade of the system has been built and installed during the recent LHC shutdown period. The basis of the improvement was to increase the level of detail of information available to the trigger, with more granular information both in longitudinal position and calorimeter depth.

In particular, this allows more sophisticated algorithms to be used based on shower shapes, while also aiding energy resolution in a higher pile-up environment. The higher data rate needed to transmit this additional information (approximately a factor of 10 in the electromagnetic layer) necessitated the use of a new digital trigger signal path which is integrated into the calorimeter outputs, replacing the old analogue path with digital signals at 40 MHz transferred on optical links. The algorithmic part of the Level-1 processing is performed on three Feature Extractors (FEX) which specialize in identifying different physics signatures.

The front-end electronics of the LAr has been upgraded during the Phase-I period, to provide L1Calo with fine-granularity digital information (SuperCells) [32] via optical fibres running at 11.2 Gb/s, in addition to the already existing analogue signal path. For each trigger-tower, ten so-called SuperCells with an area as narrow as  $\Delta\eta \times \Delta\phi = 0.025 \times 0.1$  in the front and middle layers, provide  $E_T$  information from the four longitudinal calorimeter layers. The digital data is sent to the FEX subsystems via an optical plant, called Fibre Optics eXchange (FOX) [33], which rearranges the input fibres to the mapping required by the FEX systems.

The Tile Calorimeter continues to send only analogue signals to L1Calo in Run-3. An upgrade of the Tile front-end electronics, to provide L1Calo with digitised data via optical fibres, is planned only for the Phase-II upgrade taking place after Run-3. To accommodate these plans, Tile Rear Extension (TRES) modules were built and installed in the two PreProcessor crates that process analogue signals from Tile Calorimeter, to extract copies of the digitised hadronic  $E_T$  values from the legacy trigger data path to CP and JEP, and to transmit them optically at 11.2 Gb/s to the FEXes.



**Figure 5.1.** Schematic of the L1Calo system in Run-3. The Phase-I upgrade modules are outlined in yellow. The existing legacy system is depicted alongside the new system in blue and green. Taken and adapted from [20].

The design of the new Level-1 Calorimeter Trigger is shown in Fig. 5.1.

The original (legacy) trigger system, shown in blue, is still maintained in place for the start of LHC Run-3, with necessary connections between the two, so that initial data-taking for physics purposes is not compromised while the new system is being understood and tuned for best performance. The Phase-I upgrade items are indicated in yellow.

### 5.2.1 Overview

The FEX processors are built in the form of Advanced Telecommunications Computing Architecture (ATCA) blades, hosting multiple Field Programmable Gate Arrays (FPGAs) and high-speed optical transceivers. The modules perform fast calculations to create higher-level trigger objects (TOBs) based on the calorimeter input. There are three flavours of FEX systems:

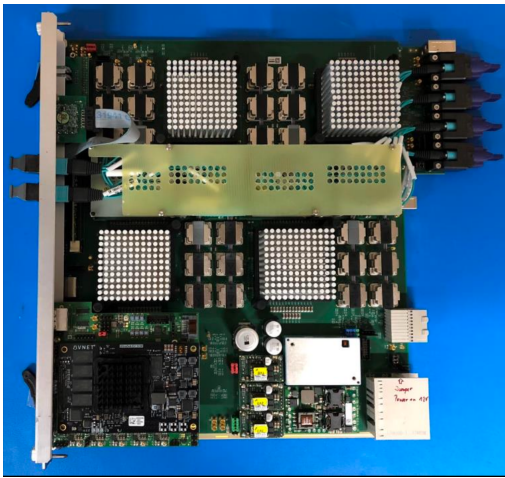
- electron Feature EXtractor (eFEX)
- jet Feature EXtractor (jFEX)
- global Feature EXtractor (gFEX)

A new Phase-I Level-1 Topological Trigger system is included in the upgrade, consisting of

three ATCA-based modules. Fig. 5.2 shows the production versions of the eFEX, jFEX and gFEX modules.



(a) eFEX module



(b) jFEX module

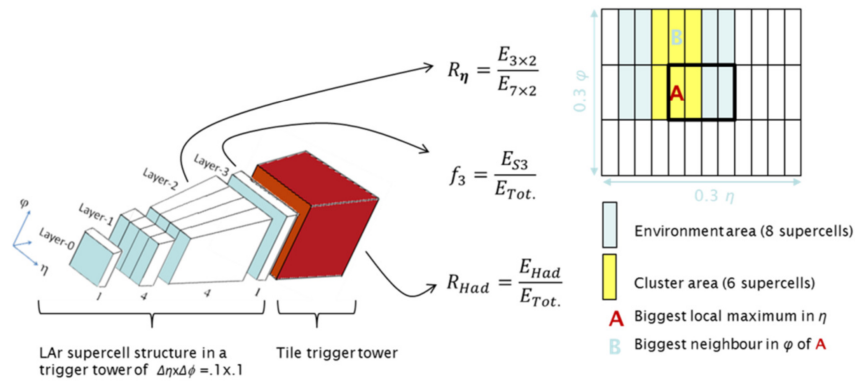


(c) gFEX module

**Figure 5.2.** Single modules of the eFEX, jFEX and gFEX [34].

### Electron Feature Extractor

The electron Feature Extractor (eFEX) processes the finer-granularity SuperCell trigger information of the electromagnetic calorimeter in order to find narrow electron, photon or tau-like showers, using the full depth and spatial information of the shower development to better distinguish, and reject the dominant jet background. It consists of 24 individual ATCA-based modules running entirely independently and in parallel in order to handle smaller blocks of the central detector coverage. The eFEX system covers an area of  $|\eta| < 2.5$  and the whole  $\phi$  range and receives SuperCells from LAr and  $0.1 \times 0.1$  trigger-towers from Tile.



**Figure 5.3.** Overview of the eFEX trigger algorithm [35] with the calorimeter layer segmentation depicted on the left and the  $R_\eta$  condition on the right. The  $R_\eta$  cluster in Layer-2 is illustrated in yellow and the environment in blue. The seed cell  $A$  is the local maximum in  $\eta$ ,  $B$  is the highest neighbour of  $A$  in  $\phi$ .

The  $e/\gamma$  algorithm [36] improves upon its CP counterpart, utilising the finer granularity cells from the electromagnetic calorimeter to identify narrow electromagnetic showers. A seed-finder algorithm scans the calorimeter cells in a window of  $0.3 \times 0.3$  ( $\eta \times \phi$ ) to determine the starting point of the cluster reconstruction. Within each window, the four central cells are compared to the surrounding area to find a local maximum. This comparison is done in parallel for the 36 cells.

Once the cluster with the highest  $E_T$  is found and classified as a particle candidate, two parameters are calculated for vetoing significant  $E_T$  contributions around the particle candidate:  $R_\eta$  and  $R_{had}$ .  $R_\eta$  is defined as the isolation of an electron candidate in the cluster in Layer-2 of the electromagnetic calorimeter. This is where the highest energy deposit occurs.

$$R_\eta = 1 - \frac{E_{cluster}}{E_{Env} + E_{cluster}} \quad (5.1)$$

Fig. 5.3 illustrates a schematic overview of the  $R_\eta$  parameter. The  $E_{cluster}$  is depicted in yellow and the surrounding area,  $E_{env}$ , in blue. The seed cell indicated with the letter  $A$  has the highest  $E_T$  and is classified as the local maximum. The highest neighbour is indicated with the letter  $B$ .

The second parameter  $R_{had}$  is defined as the ratio between the energy deposited in the hadronic calorimeter and the total energy. The electron candidates are expected to have a small  $R_{had}$  due to the showers induced by electrons being confined in the electromagnetic layer.

$$R_{had} = \frac{E_{had}}{E_{had} + E_0 + E_1 + E_2 + E_3} \quad (5.2)$$

The  $\tau$  algorithm [36] involves identifying narrow electromagnetic and hadronic clusters

and separating those from potential jet background. The seeding algorithm scans all towers in  $\eta - \phi$  and also longitudinally through the electromagnetic and hadronic layers. Once a seed with the highest  $E_T$  is found, the clustering is performed to find the energy of the tau candidate and the shower development in  $\phi$ . Due to the taus inducing narrower showers than jets, an isolation condition is applied to improve the efficacy of the algorithm.

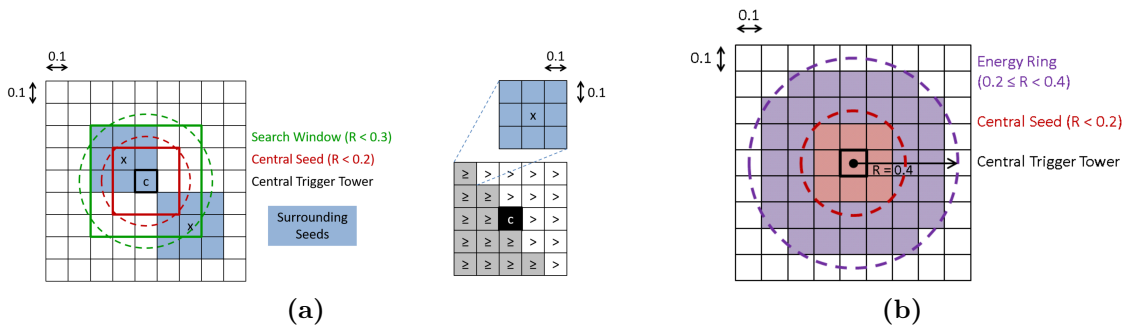
### Jet Feature Extractor

The jet feature extractor (jFEX) does not require the full granular information, and so consists of only 6 similar modules, assessing jet-like objects but with greater flexibility than the original JEP system for making regional corrections for pile-up effects.

In the central region of the calorimeters, the jFEX receives trigger-towers of  $0.1 \times 0.1$  in  $\eta - \phi$  from both LAr and Tile. However in the forward region, the granularity becomes coarser and  $\eta$ -dependant.

The jFEX small-radius jet algorithm [36] is based on a sliding window algorithm with a size of  $0.9 \eta \times 0.9 \phi$ . Around each trigger-tower within the window,  $0.3 \eta \times 0.3 \phi$ -sized sums are built to form so-called *seeds*. To identify the local maximum, the seeds inside a  $0.5 \eta \times 0.5 \phi$  search window are compared. In parallel to the seeding, jet energy sums are built around each trigger-tower by summing all towers within a radius of  $R^1 < 0.4$ .

Fig. 5.4 (a) illustrates the seeding algorithm used to identify local maxima. (b) presents the definition of a jet candidate with a radius of  $R = 0.4$ .

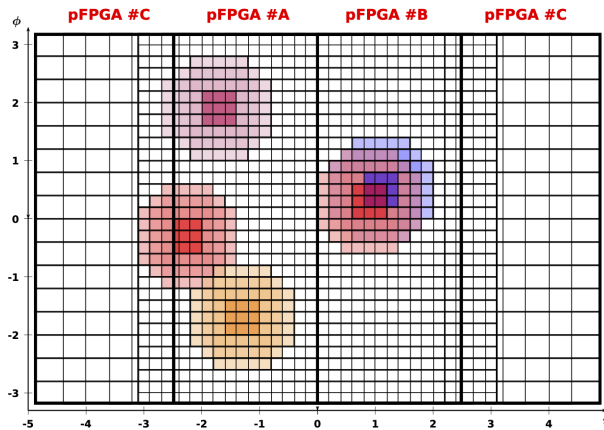


**Figure 5.4.** The diagram of the jFEX seeding algorithm (a) and the definition (b) of a jFEX jet candidate with a radius of  $R = 0.4$  [36].

The jFEX tau algorithm [36] uses a larger area of up to  $1.7 \eta \times 1.7 \phi$  compared to the eFEX, therefore a larger isolation can be calculated which complements the eFEX algorithm. In addition to the jet and tau algorithms, the jFEX computes two global variables, the sum of the total transverse energy  $\sum E_T$  and  $E_T^{miss}$ .

### Global Feature Extractor

<sup>1</sup>Definition of  $R = \sqrt{\Delta\eta^2 + \Delta\phi^2}$



**Figure 5.5.** Example event with large-R jet candidates spanning across multiple FPGAs of the gFEX module [37].

The global feature extractor (gFEX) is a single module processing data from the whole detector at a coarser granularity of  $0.2 \times 0.2$  in  $\eta - \phi$ , also identifying jets and measuring missing energy, but being a single module, has the capacity to make full event-level corrections and identify large jet objects.

The large-R jet objects [36] are formed by summing of 69 towers from the electromagnetic and hadronic layers. Once the large-R jet object is formed, a pile-up density  $\rho$  is calculated and multiplied by the number of towers making up the jet object. This value is then subtracted from the jet energy. The pileup density is calculated event-by-event by dividing the sum of the tower  $E_T$  values below a programmable threshold by the number of towers that have an  $E_T$  below this threshold:

$$\rho = \frac{\sum_i^N E_{T,i}}{N} \quad (5.3)$$

The jet object with the highest  $E_T$  out of 32  $\phi$ -bins is transmitted to the L1Topo for a trigger decision.

Fig. 5.5 illustrates a simulated example of the large-R jet formation in gFEX. The  $\eta - \phi$  space is segmented into three regions, processed by separate FPGAs.

For the  $E_T^{miss}$  calculation, the "Jets without jets" algorithm is implemented in the gFEX, further described in [38].

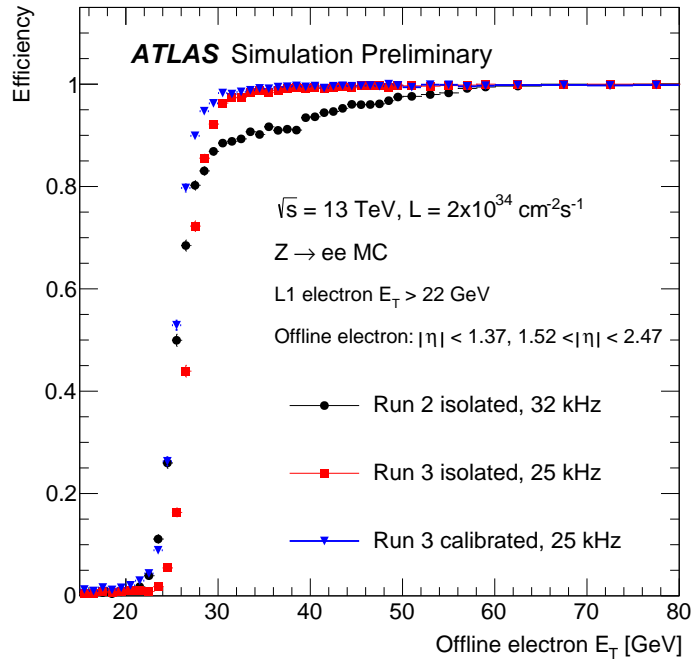
### Phase-I Level-1 Topological Trigger

The results from these FEX modules will be further assessed by a new topological trigger system, which can apply flexible algorithms from the multiplicity counting up to complex multiple object-based topological algorithms.



## 5.2.2 Expected performance

The new algorithms have been integrated as part of the ATLAS offline software framework. The algorithms performed by the FEXes are integrated into the offline simulation of the L1Calo system. Therefore they can be simulated to gauge the expected performance and behaviour for Run-3.

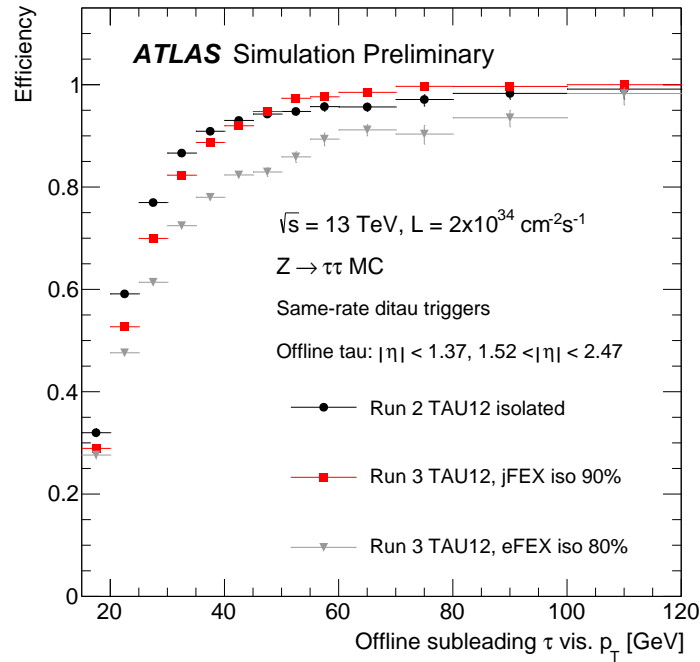


**Figure 5.6.** Single-electron efficiency comparison between Run-2 and the Run-3 eFEX [39].

An example of the expected single-electron trigger efficiency is depicted in Fig. 5.6. The results are based on Monte Carlo simulation of the  $Z \rightarrow ee$  process. The performance of the new Run-3 trigger is compared to the existing Run-2 electron trigger, and also to offline reconstructed electron candidates that satisfy a likelihood-based identification and gradient isolation.

The Run-3 isolation thresholds were tuned to give the lowest rate while introducing only a 2% inefficiency for electrons passing the Level-1 energy threshold. The isolation requirement is not applied for clusters with  $E_T > 50(60)$  GeV in the Run-2 (3) trigger. A threshold of 22 GeV is used for the Run-2 (black) and uncalibrated Run-3 (red) triggers. The improved performance of the Run-3 trigger results in smaller rate and improved efficiency. A layer- and  $\eta$ -dependent calibration is introduced (blue) to compensate for varying detector response. The threshold on the calibrated cluster energy is chosen to produce the same rate as the uncalibrated trigger, resulting in an improved efficiency.



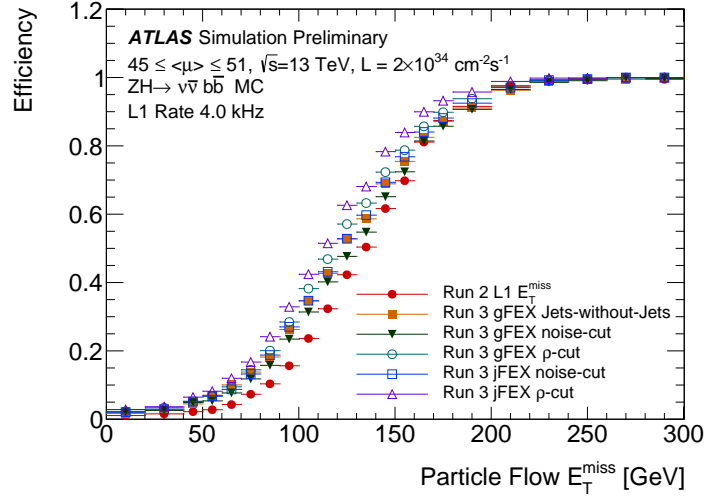


**Figure 5.7.** Di-tau trigger efficiency comparison between Run-2 and the Run-3 eFEX and jFEX [39].

For tau-based algorithms, Fig. 5.7 shows the comparison of the di-tau trigger efficiency between Run-2 and the Run-3 eFEX and jFEX. The efficiencies are derived from  $Z \rightarrow \tau\tau$  Monte Carlo simulation samples, with a 20 (12) GeV threshold on the leading (subleading) tau, with respect to the offline reconstructed tau candidates. The energy threshold corresponds to the primary Run-2 di-tau trigger, without the additional topological selection applied.

The Run-3 isolation thresholds were tuned to produce the same rate as the Run-2 trigger. Run-3 taus are reconstructed in eFEX, and the isolation requirement is computed from surrounding energy as seen in eFEX (grey) or jFEX (red).

The efficiency of the new missing transverse momentum ( $E_T^{miss}$ ) algorithms of the Run-3 jFEX and gFEX are presented in Fig. 5.8. A comparison is performed with the Run-2  $E_T^{miss}$  trigger. The results are based on a  $ZH \rightarrow \nu\nu bb$  Monte Carlo simulation sample, and the efficiency is computed with respect to the offline  $E_T^{miss}$  with a Tight selection [40]. The thresholds are tuned to give a L1 rate similar to Run-2.



**Figure 5.8.**  $E_T^{miss}$  algorithm efficiency comparison between Run-2 and Run-3 jFEX and gFEX [39].

The noise-cut algorithm computes  $E_T^{miss}$  from the vector sum of all towers with  $E_T$  above an  $\eta$ -dependent threshold. The  $\rho$ -cut algorithm computes  $E_T^{miss}$  from the vector sum of all towers with  $E_T$  above a threshold depending on  $\eta$  and corrected for the local per-event pileup density ( $\rho$ ). The jets without jets algorithm computes  $E_T^{miss}$  based on a linear combination of soft and hard contributions to the  $E_T$  of all towers.

The Run-3 algorithms have steeper turn-on curves at earlier  $E_T^{miss}$  values, which shows better expected performance from the FEX processors compared to the Run-2  $E_T^{miss}$ .

# Chapter 6

## The L1Calo PreProcessor and the TREX

The functionality of the L1Calo PreProcessor (PPr) and its upgrade for Run-3 was covered in Sec. 4.1.2. This chapter describes the hardware of the upgraded PPr system with the addition of the Tile Rear Extension (TREX) modules and the functional aspects that treat the incoming calorimeter signals.

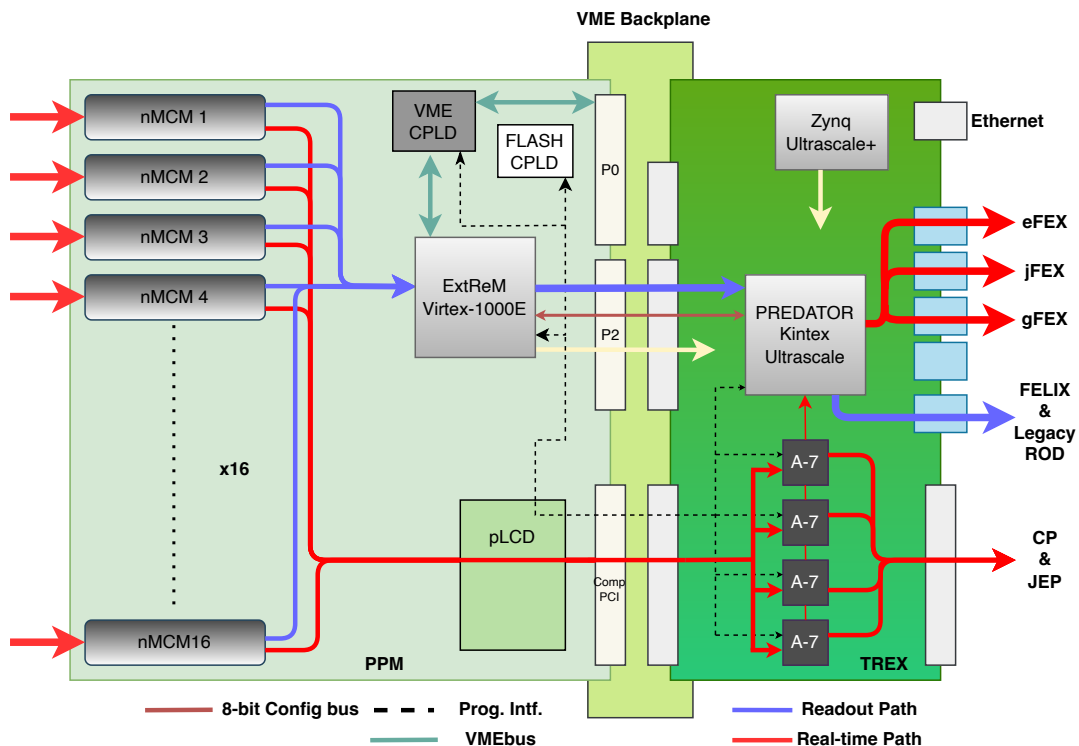
The initial development of the TREX started in 2015 when a conceptual design was proposed to create an interface that would allow the PPr system to provide digital optical data to the Feature Extractor (FEX) processors of the upgrade system, described in Sec. 5.2.1, while maintaining all of the older interfaces with the legacy trigger processors. As a prototype, the first iteration of the hardware was produced in 2017, after which it underwent two more iterations of design changes in 2019 and in 2020 respectively, before being installed in ATLAS.

The work described here started in 2019, at a time when the TREX module was still in its early development phase. The second iteration of the modules was under production and most of the functionality was not complete nor fully tested.

### 6.1 Hardware design, from prototyping to production

The TREX is a VME-compliant 9U rear transition module hosting multiple FPGAs designed specifically for fast data processing and high-speed optical transmission of the calorimeter signals. Although it is highly specialised, the module can be repurposed for other demanding processing and networking tasks, thanks to the onboard high-power FPGA coupled with fast optical transceiver modules. The hardware design revolves around five key criteria that the system must fulfil:

- Fast data formatting
- Optical outputs for the Phase-I trigger and DAQ system
- Compatibility with the existing (legacy) DAQ and Trigger processors
- Low and deterministic latency
- High redundancy and configurability



**Figure 6.1.** A diagram illustrating the various data paths of the PPM and TREX.

The processing elements are listed below along with their firmware design name:

- 4x Artix-7 FPGAs [41]- Data-IN-Out (DINO)
- 1x Kintex Ultrascale FPGA [42] - PREprocessor Data CollectOR (PREDATOR)
- 1x Zynq Ultrascale+ Multi-Processor System-On-Chip (MPSoC) [43] - Global Loader And Monitor (GLAM)

In addition to the FPGAs, the latest production TREX hosts four 12-way optoelectrical Samtec FireFly [44] transmitter modules, one bi-directional 4-way module, and an optional 12-way receiver. Each of the PREDATOR and four DINO FPGAs performs specialised tasks, therefore five individual firmware designs are created and maintained. Integrating the TREX modules into the existing PPr system, however, necessitated also

a re-design of the Virtex-1000E (ExtReM) control FPGA firmware on the PreProcessor Modules (PPMs) as well as hardware modifications.

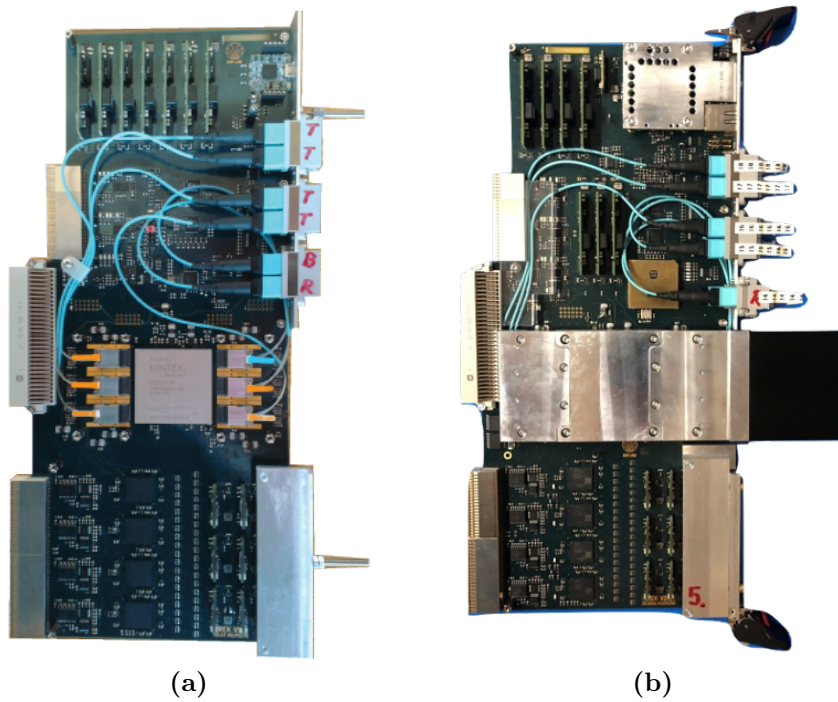
In order to visualise the combined system, a schematic diagram depicting a PPM coupled with a TREX module is shown in Fig. 6.1. The data flow through the system is highlighted in red and blue colours, while the rest of the connections show the configuration paths. The complete L1Calo PPr system for the Tile Calorimeter is composed of 32 PPMs and TREX modules in 2 VME crates.

Throughout the development cycle, the TREX design has undergone three iterations of changes. At each step, the hardware has been thoroughly scrutinised.

The first iteration, called the **prototype V1.0**, gave tremendous insight into the high-speed transmission architecture. However, there were design issues in the power distribution and in the routing of the high-speed signals. Whenever large current draws occurred, the DC-DC converters located further away from the FPGA, resulted in large ripples and drops across the core voltage as well as the voltage supplied to the high-speed transceivers. Routing of the high-speed lanes through VIAs created stubs, which acted as antennas and degraded the signal integrity. A fully assembled V1.0 prototype is shown in Fig. 6.2 (left).

The second version, referred to as the **pre-production V2.0**, improved the power delivery to the processing FPGA by moving the DC-DC converters closer to the FPGA and by an increase of the output voltage. The FPGA model was changed to a more power-efficient and package-compatible variant with a lower number of high-speed transceivers, which improved the thermal performance while maintaining the required number of transceivers. The I2C bus and clocking network were also changed for the Artix-7 FPGAs. The final change came with the introduction of the carrier functionality for the Zynq MPSoC. While the module was functional in most aspects, it suffered from larger capacitive couplings in long traces, in particular, between the ExtReM FPGA and the Zynq which caused unstable behaviour during data transfers.

The pre-production module is depicted in Fig. 6.2 (right) equipped with an improved heatsink.



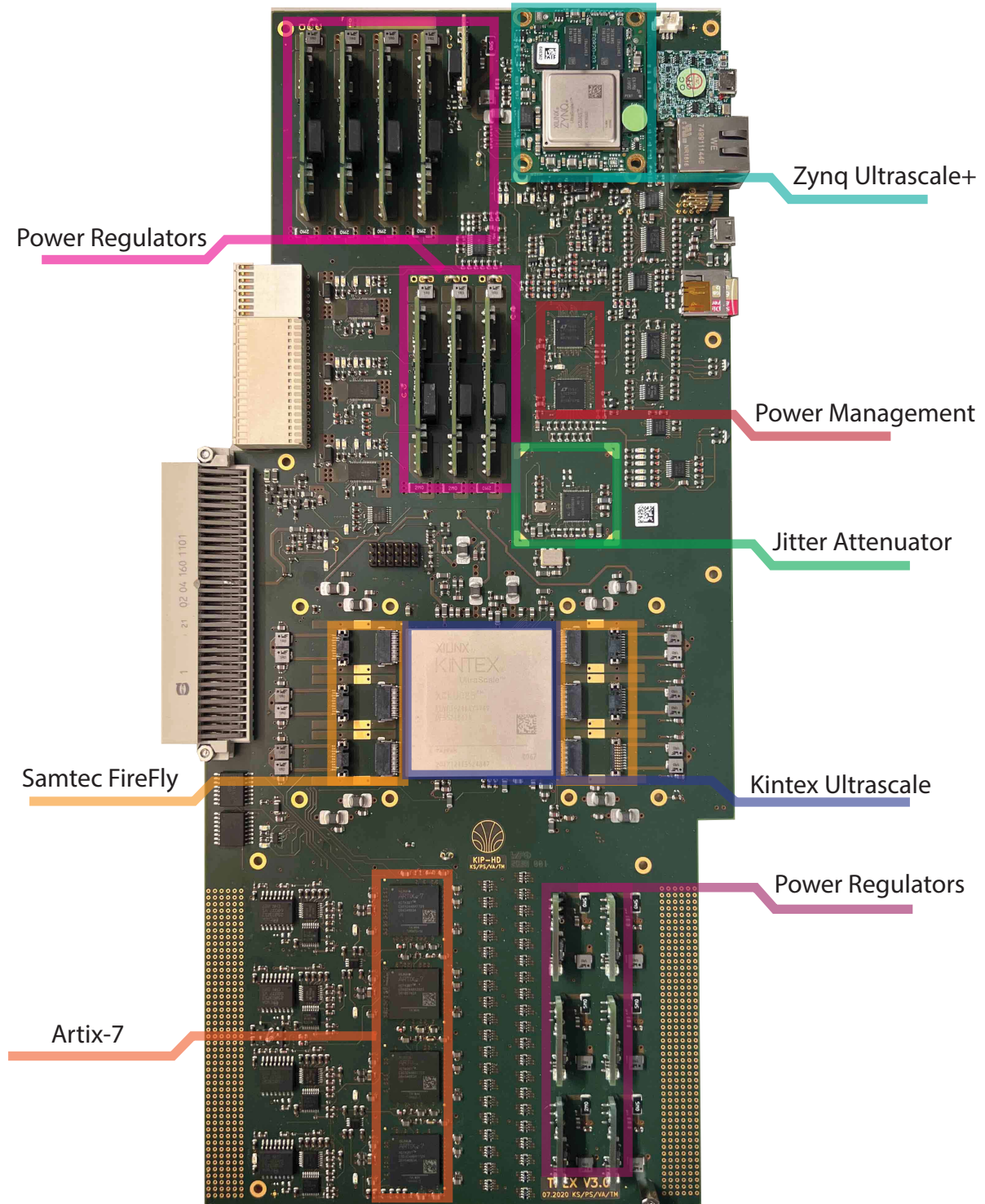
**Figure 6.2.** The first TREX prototype (a) is depicted with the FPGA and optical transceivers exposed. The pre-production (b) is shown with the latest custom heatsinks.

The third and final design, the **production V3.0**, shown in Fig. 6.3, brought a multitude of improvements to the overall stability of the module, most of which were identified during rigorous testing of V2.0. The power logic saw changes as well, by adding a feedback signal from the PPM, any leakage current flow could be controlled when the module is powered off.

The clock tree was improved with better distribution to the processing FPGA and the Zynq. Further changes were made in the Zynq and the programming interface. The final iteration fixed all issues that could've caused instabilities during the operation period in ATLAS.

The large-scale production of 40 TREX modules posed challenges as, during the acceptance tests, the majority of the assembled boards had a short circuit inside the second Artix-7 FPGA. The issue was identified through thermal imaging and showed a short-to-ground for the 2.5 V bank. It is suspected to have been caused due to thermo-mechanical stress asserted during the assembly process. A reworking campaign was started, where all the affected FPGAs were manually extracted and replaced. It was a major undertaking with a high yield rate.





**Figure 6.3.** A partially assembled production (V3) TREX without a heatsink and CompactPCI connectors and FireFly transceivers. It is used to showcase the various components of the TREX.

Powering the TREX modules is done through the VME crate power supply and is directly coupled to the PPM power circuitry. 5.0 V and 3.3 V are used as input voltages to derive all necessary bus and FPGA voltages. A configurable load-balancing mechanism is implemented, which allows for regulation of the current draw between the voltage sources. Two power management units supervise and control the DC-DC converters and issue a power sequencing routine during power on and off.

## 6.2 Adapting the PPMs for TREX compatibility

The PreProcessor module underwent both hardware and firmware adjustments to achieve compatibility with the TREX module.

### Hardware Changes

The first hardware change is the replacement of the LVDS Cable Driver (LCD) mezzanine card with a passive bridge (pLCD). The LCD card houses four Virtex-II FPGAs and a pre-emphasis circuit that drives the incoming LVDS streams from the nMCMs to the CP and JEP processors. The LCD functionality is moved over to the TREX, where the four Artix-7 FPGAs perform the required additional data duplication for the FEX processors. The pLCD bridge remaps the LVDS signals to the new FPGAs through the Compact PCI connector on the PPM. It also enables the TREX to use the programming interface and multiple control signals from the VME CPLD<sup>1</sup> located on the PPM. Further details on the programming interface are described in Sec. 6.3.4

The next set of changes include the removal of resistors along the differential signals, which would limit the bandwidth on the LVDS streams. This is done to achieve higher transfer speeds in the future if needed. Furthermore, the production series of the TREX modules require a static signal from the PPM power manager circuit, which is fed through an external wire to one of the P2 connector pins. The signal indicates the power state of the PPM. The corresponding power logic of the TREX board uses the indicator to control any leakage currents flowing into the board when the main PPM rail is powered off. All 32 modules in the TileCal crate have been modified with these changes.

### CPLD changes

With the introduction of the pLCD card, the programming and configuration signals that were used for the Virtex-II FPGAs are repurposed and routed to the TREX. The VME CPLD firmware is modified to show the configuration status of the TREX FPGAs and to control the programming interface.

---

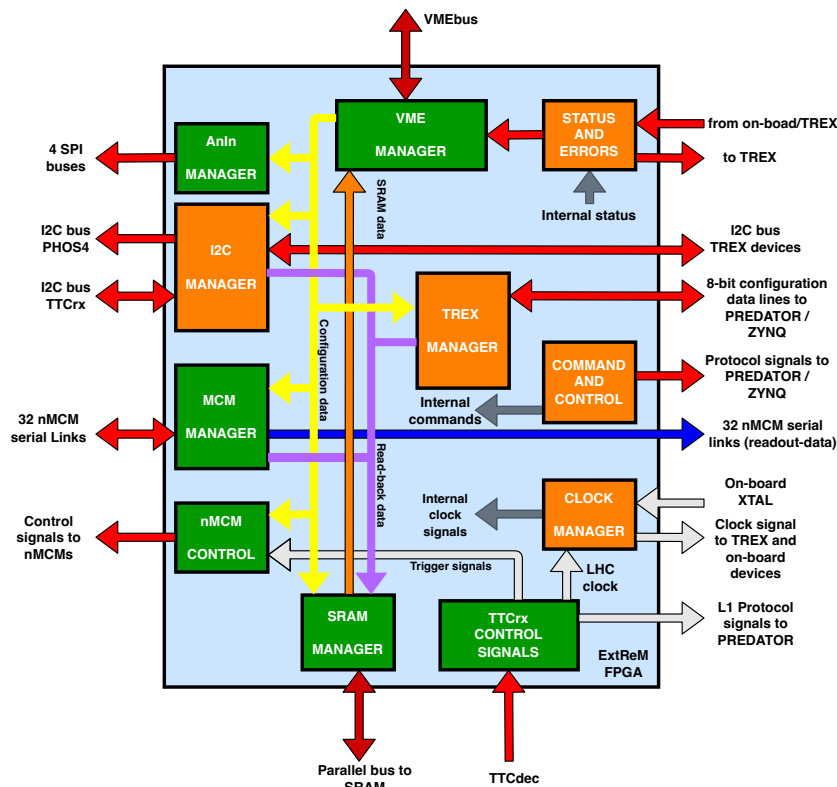
<sup>1</sup>Complex Programmable Logic Device



An additional change is performed for the V3.0 TREX modules, where the programming interface is changed from a uni-directional, write-only bus to a bi-directional interface. The change gives direct read and write access to the TREX flash memory modules via the VMEbus. The procedure is described in Sec. 6.3.4.

### The ExtReM FPGA

In Run-1 and 2, the Readout Manager FPGA [45], located on the PPM was responsible for the control of the nMCMs, rate metering and histogramming. In addition, it was also performing the collection and processing of event data to be read-out to the ATLAS DAQ system. With the addition of the TREX to the PreProcessor system, the functionality of the FPGA is modified for managing the TREX, hence aptly named the Extended Readout Manager(ExtReM). It is acting as a communications hub between the Single-Board Computer (SBC) and the TREX. The VME commands issued by the SBC are received and decoded on the ExtReM FPGA. Based on the destination address, the data is passed to TREX. The response from the destination module is also processed and sent back to the SBC.



**Figure 6.4.** A schematic overview of the ExtReM firmware. The orange-coloured logic blocks illustrate the new or modified functionality for the TREX.

The ExtReM logic blocks are depicted in Fig. 6.4. Multiple new management logic circuits are added to the design, the internal clock distribution is also modified to provide

the TREX module with the 40.08 MHz LHC clock. The status and error handling segments have new registers for the new module. The removal of the readout processing block freed-up resources for implementing the functionality required for the TREX.

## 6.3 Control, configuration of the TREX and the interface with the PreProcessor

### 6.3.1 Communication between the PPM and TREX

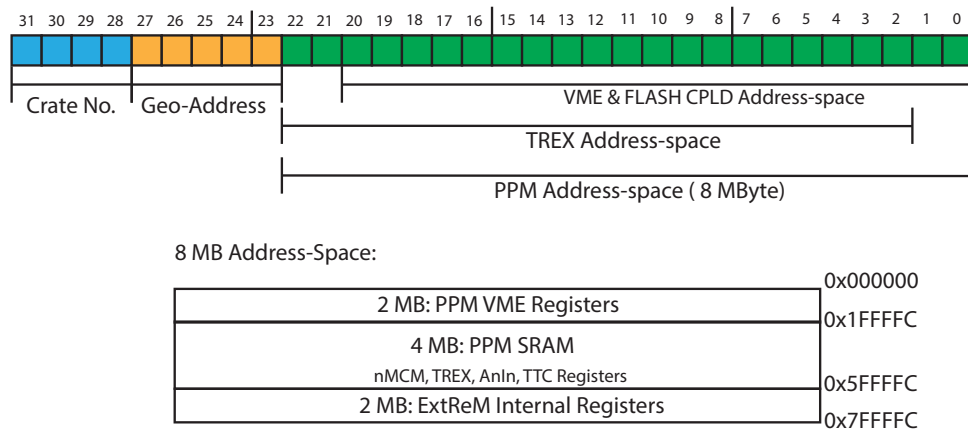
There are three methods for interacting with the TREX via VME, each interface serves a specific purpose:

- **Custom 8-bit protocol.** A custom FPGA to FPGA protocol that connects the processing (PREDATOR) FPGA and the Zynq to the PPM. It is used for loading and retrieving configuration parameters required for initialising the devices and preparing them for data-taking operations.
- **Inter-Integrated Circuit (I2C) bus** A synchronous, serial communication bus dedicated to accessing onboard programmable devices. Hardware control and collection of monitoring information are performed via the I2C protocol.
- **8-bit Programming interface** The programming interface is a multipurpose 8-bit wide data bus, dedicated to programming the FPGAs and their flash memory modules. Multiple transfer protocols are initiated over the bus, such as QSPI for the flash memories and SelectMAP [46] for the FPGAs.

The interfaces are described in more detail in Sec. 6.3.2.

#### Register map of the TREX

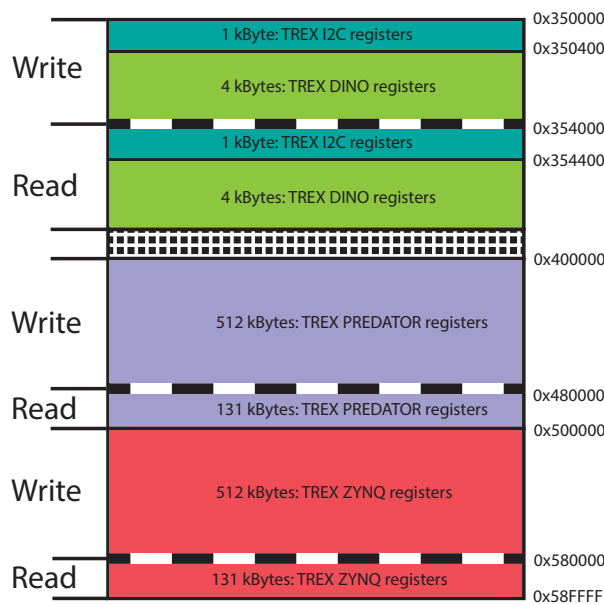
Before the data transfer protocols can be initiated, a dedicated TREX register map is required within the VME address space. The register model implemented for the PreProcessor uses a 32-bit addressing mode with 32-bits data (A32/D32). For every PPM, 8 MBytes of memory is allocated, which is divided into three sub-spaces. This is shown in Fig. 6.5.



**Figure 6.5.** The VME address-space of the PPM and TREX.

The first 2 MB are dedicated to the control and status registers of the PPM CPLDs. The next 4 MBs are allocated for mapping an entire on-board static RAM (SRAM) module to extend the ExtReM memory capabilities and access the registers of on-board devices such as the TTCdec mezzanine, the An-Ins and the nMCMs. In Run-2, it was used to carry out large block transfers between the ExtReM and the nMCMs while also providing storage space for read-back purposes of other peripherals. The remaining 2MB are designated for internal registers within the ExtReM FPGA.

The SRAM occupancy was at 50%, which meant that the free 2 MB could be used for mapping all the TREX registers and memories. Since all TREX registers are based on 32-bits, the address space is 4-byte aligned exactly as for the ExtReM. This means that the first two LSBs of the address bus are never used and can be ignored.



**Figure 6.6.** The address-space and the divisions for the TREX components.

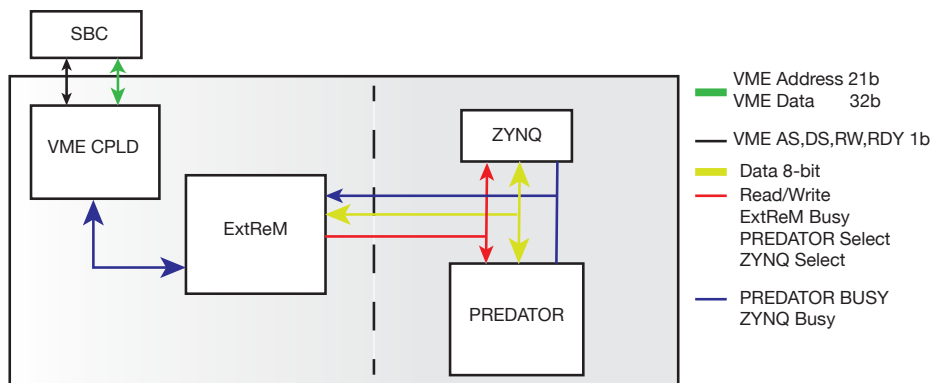
The allocated TREX address space is divided further into multiple read and write

sectors, with each FPGA having its own sub-address space. The sub-division is represented in Fig. 6.6. Each of the register sectors has read and write counterparts. The read address is always shifted by a fixed offset from the write address. By interacting with the dedicated locations, the operation type is encoded within the address. This signals the TREX whether it is a read or write operation.

### 6.3.2 8-bit custom protocol

The main communication for reading and writing registers in the processing FPGA is realised by a synchronous, custom 8-bit interface. 8 parallel data lanes and a set of 6 control signals which perform the communication handshake are shared between the ExtReM, PREDATOR and Zynq FPGAs.

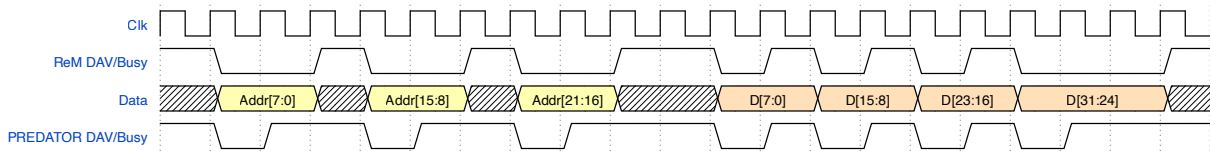
Fig. 6.7 illustrates the control interface of the TREX via VME, where the command issued by the Single-Board Computer is passed through the PPM. The data lanes used by the 8-bit custom protocol are colour-coded and also show the direction of the data.



**Figure 6.7.** Control interface from VME to the TREX FPGAs. The data passes through decoding stages in the VME CPLD and the ExtReM before being forwarded to the TREX FPGAs via a custom 8-bit interface.

The initiator of the transfer is the ExtReM FPGA, started by the SBC issuing a specific request through the VME bus. At the start of a write cycle, the ExtReM FPGA drives the corresponding select signal low to indicate which device should acknowledge the transfer. The transferred data contains the 32-bit VME configuration data and the 21-bit VME configuration address, where the address itself indicates whether a read or write operation should be performed. The destination device decodes the configuration address and writes the data to the designated register or memory location.

A typical write cycle is depicted in Fig. 6.8, the Busy signal here has a double meaning. When sending data the ExtReM drives the Busy signal low to indicate data availability, when receiving data, the same signal is driven to low by the receiving end to indicate that the data is being processed.



**Figure 6.8.** Timing diagram depicting a write cycle of the custom 8-bit protocol.

When a read cycle is initiated, only the VME configuration address is transferred to the destination device and the VME cycle is acknowledged. However, the Finite-State-Machine (FSM) on the ExtReM polls until the 32-bit data are read from the destination device and written to the SRAM on the PPM at the same memory address location as the configuration address. The status of the transfer can be polled via the SBC, indicated by the busy signals. Once the 32-bit data word is written in the SRAM, the busy signals are de-asserted and the data can be read by the end-user through another VME cycle. This method of performing a read cycle is required since the data transfers over this interface can take longer than the time-out delay of the VMEbus, which in turn causes VME bus errors. To avoid errors, the ExtReM FPGA always acknowledges the VME requests and flags the status of the transfer in the internal status registers.

As it is a synchronous protocol, it requires both the source and destination FPGAs to use a common clock, such as the LHC clock, described in Sec. 6.4.

### 6.3.3 I2C Controller and on-board devices

Apart from the 8-bit configuration interface, the TREX relies on a second interface for configuration and monitoring via the Inter-Intra Circuit protocol (I2C). This interface allows the user software to access the TREX I2C data bus and the devices attached:

- 4x 9-bit non-volatile I/O expanders [47] - static I/O signals used for configuration
- 2x 16-bit volatile I/O expanders [48] - status indicators showing module health
- Si5345 Jitter Cleaner and attenuator [49] - clean clock supply to the FPGA
- Si570 Programmable oscillator [50] - Standalone multi-purpose crystal oscillator
- 2x Power Management Units [51] - For power sequencing and monitoring
- 2x Monitoring ADCs [52] - Monitoring of currents and voltages
- The PREDATOR (and FireFly modules) and four DINO FPGAs

There are two main ways for the end-user to access the TREX I2C bus: via VME through the ExtReM FPGA, similar to the 8-bit configuration interface or through the Zynq via Ethernet. Both the ExtReM and the Zynq implement an I2C master core logic

block, hence can act as masters of the bus. Bus arbitration is, therefore, necessary to ensure only one device is accessing the bus at a time.

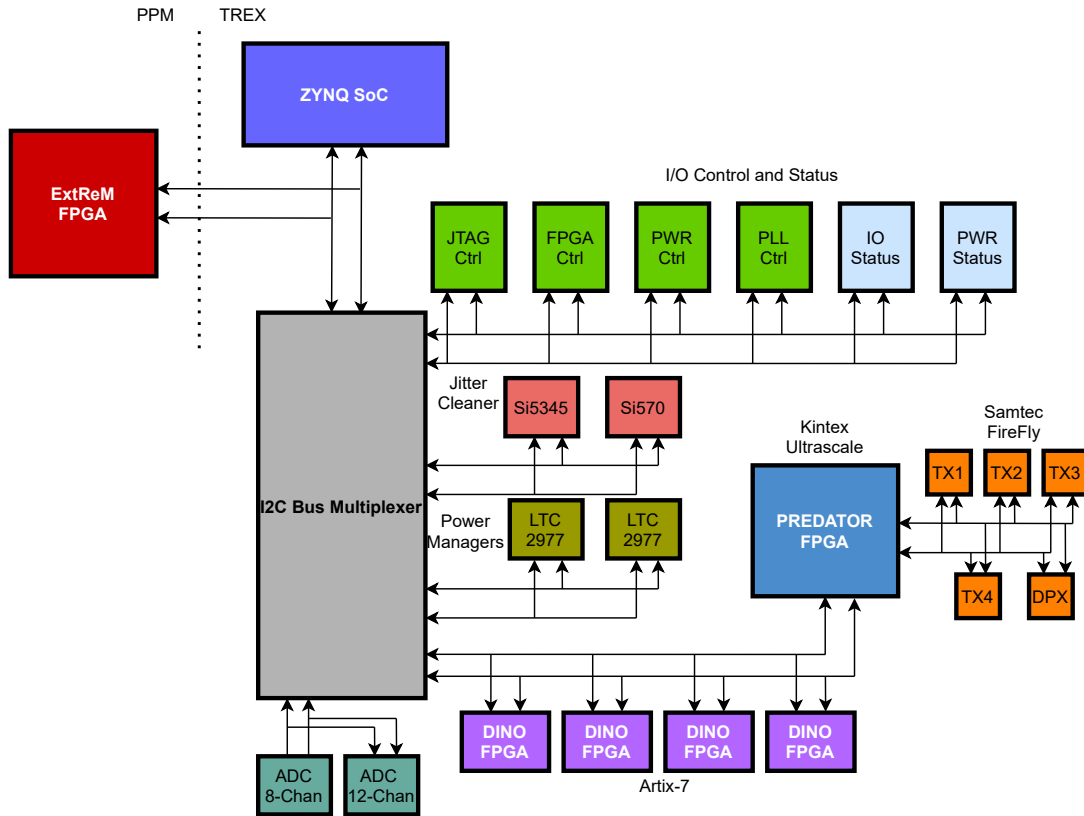
Two open-drain devices control the I2C data bus, with an enable bit set by the ExtReM and Zynq logic, respectively. The presence of the enable bit ensures a clean way to disconnect one or the other bus master in case the controller logic locks up and hinders access to the bus. Upon power-up, the Zynq is set as the master to collect monitoring information for the ATLAS Detector Control System (DCS). When the ExtReM requires access to the bus, a procedure is set in place to handle the arbitration: Dedicated control registers are defined in the ExtReM which handle the arbitration handshake. The control software issues the VME command for the Zynq to release the bus and flag the ExtReM as master. The ExtReM issues the request through one dedicated control signal to the Zynq and waits for a response.

Upon receiving the request, the Zynq completes any ongoing I2C transfers, asserts a busy to the monitoring framework application, discussed further in Sec. 6.7.3, and indicates to the ExtReM that the bus is released. In turn, the ExtReM flags itself as master and can perform the I2C read/write commands issued by the control software. When the VME actions are finished, the ExtReM releases the bus and signals the Zynq, which is continuously polling the status, that it may resume the operation on the bus.

The ExtReM FPGA also has the option to take the master by force in case of an emergency, this has been rarely used in the test-bench environment and never during standard operation in ATLAS.

In the implementation of the final functional design, the default I2C bus master is defined to be the Zynq, which periodically accesses all devices and collects environmental information for monitoring purposes to be sent to the DCS. It is strictly performing read operations. Any configuration change, be it programming the Oscillators or changing the FPGA configuration modes is strictly done via the VME run control application through the ExtReM.

The six FireFly transceivers include an I2C interface as well, however, in contrast to all other onboard devices, the modules are not connected directly to the I2C bus. The communication is performed through the PREDATOR FPGA, in which the logic allows the FireFly communication either via the 8-bit configuration interface or via local I2C.



**Figure 6.9.** The schematic overview of the TREX I2C bus with all connected on-board devices.

Fig. 6.9 illustrates the TREX I2C bus structure. A programmable bus multiplexer allows switching between the master devices and the groups of various on-board components.

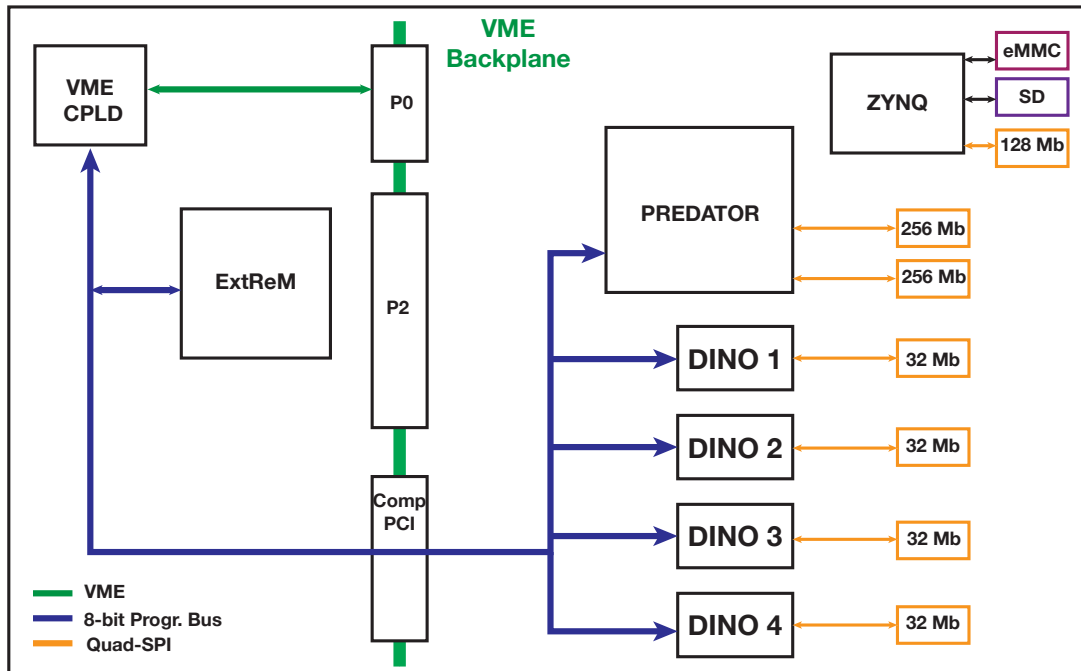
### 6.3.4 Programming interface

During operation in ATLAS, physical access to the modules is not always available, hence multiple, redundant interfaces are needed for programming the FPGAs and their flash memories. Each FPGA includes three different methods for loading the firmware bitstreams. The PREDATOR and four DINO FPGAs can be programmed via the following interfaces:

- NOR-based flash memories using the Quad Serial Peripheral Interface (QSPI) protocol
- JTAG - via USB or Ethernet using the Zynq as a JTAG server
- Software loading via VME using the SelectMAP protocol

For the Zynq SoC, the configuration can be performed via the SD card, QSPI and JTAG. Detailed booting procedure is discussed in Sec. 6.7.2

Fig. 6.10 depicts the programming bus connecting the VME CPLD to the TREX FPGAs through the Compact PCI connector. The flash memories attached to each FPGA are shown in orange.



**Figure 6.10.** A diagram depicting the available FPGA programming methods via the VMEbus.

## QSPI

Each FPGA is connected to a NOR-based flash memory module, which enables fast configuration of the FPGA with the firmware binary stored within it. The interface for access and transfer of data between the flash memory and the FPGA uses the Quad Serial Peripheral Interface (QSPI) protocol.

The SPI flash memories attached to the DINO FPGAs are 32 Mbit in size, allowing for a single firmware binary to be stored. For the PREDATOR FPGA larger, dual 256 Mbit flash memory modules are used. The firmware binary is split into 2 halves, where each half is stored on one of the flash memory modules. In this configuration, with a total of 8 parallel data lines, the firmware binary can be loaded twice as fast.

The Zynq SoC also comes with a 128 Mbit flash memory module, although the SD card boot method is preferred, the QSPI flash memory module brings redundancy to the table in case of a failure of the SD card. It is described in more detail in Sec.6.7.2.

Programming the flash memories via VME is done through direct emulation of the QSPI protocol in the software. The data is passed through the FPGA, such that a direct communication line between the SBC and the memories is established.



## JTAG

The JTAG method is used mainly for fast FPGA configuration, updating the flash memory modules and firmware logic analysis in the laboratory environment. The interface can be accessed either via a USB module or through the Zynq which acts as a hub and transfers the commands to the selected FPGA. In-detail description of the JTAG interface is presented in the 6.7.3 section.

### Software loading and updating the flash memories

The software loading method is a way to directly transfer the firmware binary file to either the FPGA or the flash memory module via the VMEbus. This method interacts directly with the software, hence the protocols used for the transfers need to be implemented in the software. The corresponding protocols are the SelectMAP for the FPGA and the Quad-SPI for the flash memory respectively.

The programming of the 4 DINO FPGAs uses a cascade model. The selection signal of the FPGA is passed to the next one once the upstream DINO FPGA is successfully configured. This applies to the first, second and third DINO FPGA.

After configuring the FPGA with the firmware binary, the 8 dedicated programming lanes become available for free use. This point is crucial as it allows the same programming lanes to be routed internally through the FPGA and out to the flash memory modules. In the case of the DINO memories, 4 lanes are used.

For the dual memories of the PREDATOR, the 8 lanes are split with 4 routed to one memory module and the other 4 to the second module. Having 4 lanes per memory increases the programming speed significantly due to the 4 times increase in the bandwidth compared to the traditional SPI transfer. In particular, the flash memories for the PREDATOR FPGA are programmed in parallel. Using a single VME cycle, the data is written to both memories at the same time.

## 6.4 Clocking and trigger information

There are two input options for the TREX to receive the LHC clock and the Timing, Trigger and Configuration (TTC) commands:

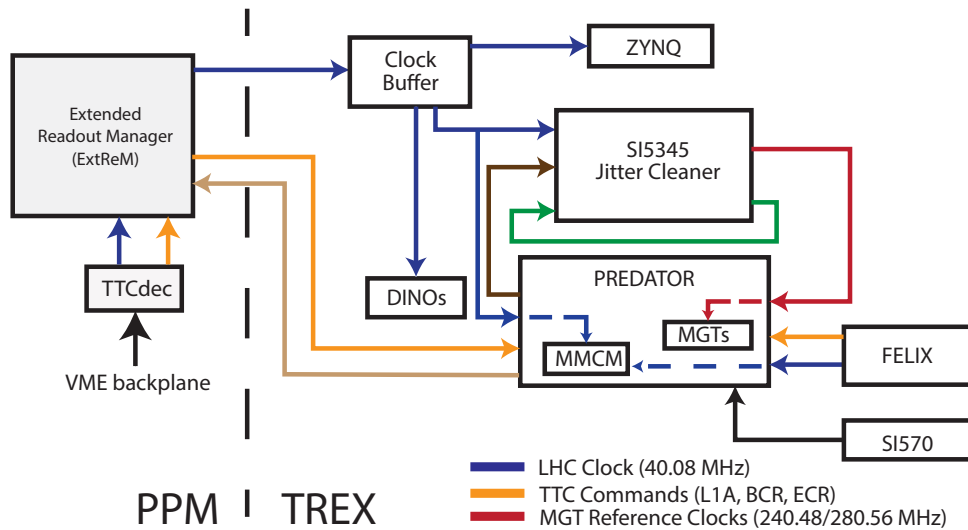
- Legacy System: Via the same source as the PPM, through the TTCdec mezzanine hosting a TTCrx chip [53].
- Phase-I: Recovery of the clock from the GBT [54] data frames from FELIX.

The legacy TTC path uses the existing infrastructure of how the PPMs receive the LHC clock and the TTC commands. Optical signals encoded in a 160MHz format are fed

from the ALTI [29] to the Timing Control Module (TCM). The TCM, located in the last slot of the VME crate, converts the optical signal into differential signals and propagates them to each PPM via a dedicated bus on the VME backplane. The differential signals are caught by the TTCdec mezzanine and decoded.

Decoding the input yields the following signals of interest:

- **LHC Clock (40.08 MHz):** The LHC clock is synchronised with the LHC RF frequency ( 400.8 MHz) and distributed to all sub-detectors.
- **Level-1 Accept Signal (L1A):** The L1A is issued by the Central Trigger Processor, signalling all detector sub-systems to transfer the event data to the DAQ system.
- **Bunch Counter Reset Signal (BCR):** Issued once every LHC turn ( $89 \mu\text{s}$ ), the BCR signal synchronises the local Bunch-Crossing Identifier (BCID) counters of the sub-detectors.
- **Event Counter Reset Signal (ECR):** The ECR is issued periodically to clear the Level-1 ID counters of the sub-detectors, typically every 5 s during data-taking.



**Figure 6.11.** The clock and TTC distribution network on the PPM and TREX.

The LHC clock and the TTC command signals are fanned out inside the ExtReM FPGA, where one single-ended copy of each signal is distributed to the TREX module via the P2 backplane connector. The TTC command signals are routed directly to the PREDATOR FPGA, while the LHC clock is used as input into a fan-out clock buffer. The fan-out buffer output drives four LVDS copies of the clock pulse to the following on-board devices:

- Si5345 Jitter cleaner-attenuator

- PREDATOR FPGA
- The Zynq SoC
- DINO FPGAs

Fig. 6.11 illustrates the clocking network of the PPM and TREX modules. The on-board Si570 programmable oscillator provides a clean 200 MHz clock for the GLink transmission to the legacy ROD at 800 Mbps, which will be covered further in the chapter.

The second source for receiving the LHC clock and TTC signals comes from FELIX. Here, a 240.48 MHz clock is recovered from the GBT input data frame. The clock signal is synchronous to the 40.08 MHz LHC clock. The TTC commands are included in the GBT data.

Seven outputs of the Jitter Cleaner Phase-Locked Loop (PLL) provide the reference clocks to the GTH<sup>1</sup> transceivers of the PREDATOR FPGA. The Xilinx transceiver architecture combines the transceivers in groups of four, which are called *quads*, that make multi-channel implementations easier. The GTH quads in the PREDATOR FPGA can distribute the clock to two neighbouring quads above and below, located in the same Super Logic Region<sup>2</sup> and physical location.

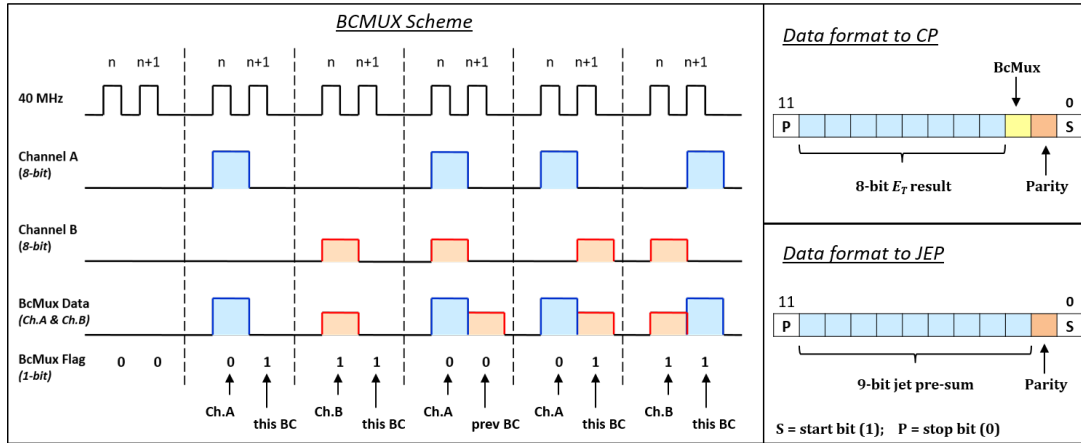
## 6.5 The Real-time data path

The path of signals received from the calorimeters, processed and transmitted further down the trigger chain to the CTP at the rate of the LHC clock (40.08 MHz) is called the real-time path. The corresponding hardware and firmware design is built with low and deterministic latency in mind. The TREX introduces high-speed transceivers along the trigger path which require special attention for optimizing the latency and maintaining a deterministic nature.

On the TREX, the real-time data arrives in the form of 48 LVDS serial streams after digitisation and processing on the nMCMs. 32 of the serial links provide 8-bit  $E_T$  results, with a granularity of  $\Delta\eta \times \Delta\phi = 0.1 \times 0.1$  and a resolution of 500 MeV per LSB. The other 16 links provide coarser 9-bit jet sums with a granularity of  $\Delta\eta \times \Delta\phi = 0.2 \times 0.2$  and a resolution of 1 GeV per LSB. The saturation point is 127.5 GeV for the first set of links and 255 GeV for the latter.

<sup>1</sup>Xilinx Gigabit Transceiver type H (up to 16 Gbps)

<sup>2</sup>Devices which use Stacked Silicon Interconnect Technology (SSIT) have multiple dies packaged together. Each die is referred to a Super Logic Region



**Figure 6.12.** The Bunch-Crossing Multiplexing (BCMUX) scheme illustrating the possible combinations of the data contents and the BCMUX flag, and the 12-bit data frame of the real-time path sent to the CP and JEP. Adapted from [55].

### The output to CP/JEP

To reduce the number of input links for the CP, mentioned in Sec. 4.1.2, the nMCM channels with the same  $\eta$  coordinate are paired together with a procedure called Bunch-Crossing Multiplexing (BCMUX) [28]. As each nMCM has 4 channels, the BCMUX scheme is used to lower the amount of LVDS links from 4 to 2 per nMCM.

The underlying concept of the BCMUX routine relies on the fact that when the bunch-crossing identification algorithm on the nMCM determines a non-zero  $E_T$  result for a given BC, the BC that follows after will contain a zero result in that same channel. Using two consecutive BCs, the same LVDS channel is used to multiplex two non-zero channels from different nMCM channels. For each BC, the data format for the CP output includes a BCMUX flag next to the 8-bit  $E_T$  result. It indicates to which BC the  $E_T$  result belongs. An odd parity bit is added for transmission error detection and the 10-bit data is encompassed with start and stop synchronisation bits, forming the 12-bit transmission frame.

The output for the JEP, mentioned in Sec. 4.1.2 is formed by summing the 8-bit  $E_T$  values of all four nMCM channels. The summing is performed by one additional clock cycle, and the resulting 10-bit value is truncated to 9 bits by truncating either the Most Significant Bit (MSB) or Least Significant Bit (LSB), resulting in a 9-bit  $0.2 \times 0.2 \sum E_T$  value. An odd parity bit is added here as well and the 10-bit result is wrapped with the synchronisation bits, forming the 12-bit frame.

The BCMUX scheme and the output formats are visualised in Fig. 6.12.

## Output to the Feature Extractors (FEXes)

The real-time data path is the most critical functionality performed on the PPM/TREX, with the most stringent requirements applied. The data flow and processing are optimised such that the physical signal traces are kept as short as possible and the sequential logic uses the least number of clock cycles to perform the logical operations.

In Table 6.1 the optical outputs to the FEX processors are summarised.

Nr. Opt. Links	FEX	Data	Resolution $\Delta\eta \times \Delta\phi$	Speed
Up-to 12	eFEX	8-bit $E_T$	$0.1 \times 0.1$	11.2 Gbps
12	jFEX	8-bit $E_T$	$0.1 \times 0.1$	11.2 Gbps
1 + 1	gFEX	9-bit $\sum E_T$	$0.2 \times 0.2$	11.2 Gbps

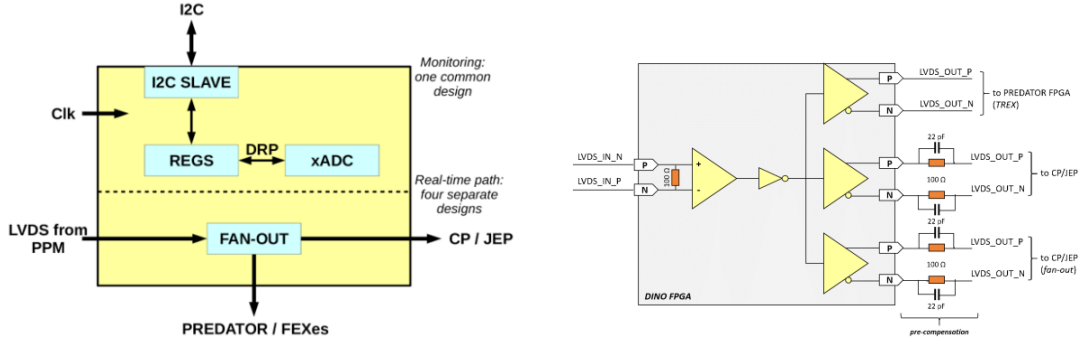
**Table 6.1.** Overview of the real-time output to the FEX processors

### 6.5.1 Data duplication and electrical transmission

The real-time data arrives on TREX at the 4 DINO FPGAs. The first 2 FPGAs process the signals with finer granularity going to the CP, while the other 2 FPGAs treat the signals with the coarser granularity directed to the JEP. Each DINO is tasked with duplicating the incoming LVDS signals and sending one copy to the PREDATOR FPGA and another to the legacy systems downstream.

A general overview of the signal processing of the DINO FPGA design is depicted in Figure 6.13. The input LVDS signals are internally terminated, converted to single-ended and fanned out into separate FEX and CP/JEP streams, with additional duplicate outputs of certain CP/JEP links to due overlap regions. Before going out of the FPGA fabric, the duplicated signals are converted back to LVDS and driven out.

Any additional routing within the FPGA fabric adds to the overall latency of the system. Therefore, to minimise the delays, stringent constraints are set for the internal FPGA routing. Sequential logic is avoided for the LVDS signal processing as it would introduce further delays in signal propagation.



**Figure 6.13.** Functional overview of the DINO FPGA design. The design is segmented into a monitoring, control block and LVDS data duplication. The LVDS streams are separated with copies transmitted to the PREDATOR FPGA and to the CP/JEP subsystems. Certain channels are duplicated for CP/JEP due to overlap regions.

As each DINO FPGA receives a different number of LVDS inputs, 4 separate flavours of the firmware are maintained. The monitoring portion of the firmware is running in the LHC clock domain (40 MHz) which drives the I2C slave interface logic and the system monitoring core.

### 6.5.2 De-serialisation, delays and bitslip logic

The first part of the data treatment in the PREDATOR FPGA is the capturing, de-serialising and decoding of the LVDS streams carrying the transverse energies. At the input, the LVDS streams are converted into single-ended signals and aligned to the 240 MHz sampling clock, which is synchronous to the incoming data. The alignment involves delaying the input signal with respect to the sampling clock of the deserialiser, such that the sampling points (positive or negative edge) don't fall into transition regions, where the state of the bit is metastable. The LVDS transmission is running at a speed of 480 Mbps which corresponds to a data bit length of 2.08 ns.

The delay component consists of up to 512 taps with a varying resolution between 2.5 ps to 15 ps depending on the configuration. In the case of the PREDATOR FPGA, the average resolution of a tap across all input links can be calculated:

$$\langle D_T \rangle = \frac{1}{N_{links}} \sum_{i=1}^{N_{links}} \frac{D_0}{T_i - T_{clock}} [ps] \quad (6.1)$$

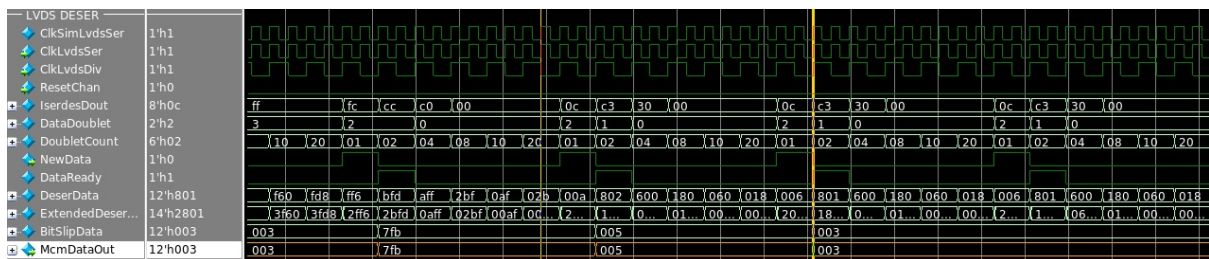
where  $D_0$  is the initial delay value in ps, set at the start.  $T_i$  corresponds to the number of taps used for the corresponding link and  $T_{clock}$  is the difference between the input data and the sampling clock paths that must be accounted for during the calculation. The  $N_{links}$  corresponds to the number of LVDS links, in the case of the links carrying  $0.1 \eta \times 0.1 \phi$  8-bit  $E_T$  values is 32, while for the links carrying  $0.2 \eta \times 0.2 \phi$  9-bit jet-sums

it is 16.

A self-calibration routine is enabled for correcting the clock-to-data skews at the input of the deserialiser. The calibration also takes into account voltage and temperature compensation in order to keep the delay value fixed over time. For each link, the initial delay value  $D_0$  is set to 500 ps and after the calibration procedure, the number of taps  $T_i$  used to achieve this delay value is read out from the delay control logic. Subtracting the  $T_{clock}$ , which is on average 54 taps, an average tap width  $\langle D_T \rangle$  of 4.5 ps is obtained. After obtaining the tap width, the delay value is tuned step-by-step to achieve alignment.

The serial-to-parallel conversion is performed with 6 data captures in the double-data-rate mode (DDR), in this mode, the data bits are sampled with both the positive and negative edge of the clock. Before the deserialisation of the LVDS data, a synchronisation routine must be performed to identify the incoming pattern from the nMCMs. The 12-bit synchronisation pattern comprises 6 zeroes and 6 ones, which indicate the start and the end of the data frame. The 12-bit data is shifted in a barrel register for every clock cycle until the synchronisation pattern is detected. Upon detection, the pattern is compared to the value of 0x3F 128 consecutive times for all 48 links. If any of the links don't pass the synchronisation after 128 iterations, the status is reported back to the user as an error. The synchronisation procedure is carried out during the configuration stage, before the start of data-taking.

The 32 LVDS streams that carry 2  $E_T$  values on a single physical link, described as BCMUX in Sec 6.5 are de-BCMUXED after deserialisation to retrieve all 64 channels. The parity of the payload is also calculated and compared to the odd parity bit. In the case where the parity does not match, a link error flag is set to indicate transmission errors to the FEX processors, otherwise, it is dropped from the processing chain.



**Figure 6.14.** A behavioural simulation of the TREX deserialisation block of the incoming LVDS streams from the nMCMs.

Fig. 6.14 illustrates the timing diagram for the deserialisation of the incoming LVDS streams. The results are obtained from behavioural simulation of the logic blocks.

### 6.5.3 Data treatment and formatting

A single PPM with a TREX processes the analogue pulses from an  $\Delta\eta \times \Delta\phi = 0.4 \times 1.6$  region of TileCal. The formatting for the FEX processors is unique and requires individual packing of the data into a frame which is transmitted via high-speed optical fibres. Based on the transmission link speed, the formatting of the data is performed with a clock speed of 280.56 MHz. This corresponds to 7 clock cycles per single bunch crossing (40.08 MHz) and translates to a single data frame containing seven 32-bit words.

In Fig. 6.15 the format is shown as the output format to the eFEX, jFEX and gFEX. Each subsystem has slight differences in data ordering and format. For the eFEX, the data fields have a length of 10-bit, while for the jFEX processor, 12-bits are dedicated. The lower 8-bits are reserved for the transverse energy result with the 9th bit containing a link error flag. The remaining upper MSBs are available in case additional information is needed in the future. The data fields for eFEX and jFEX are numbered in increasing order according to the PPM channel numbering defined during previous operations in Run-1 and Run-2 and follow a specified mapping in  $\eta - \phi$  coordinates. The gFEX data frame follows the nMCM number in increasing order.

The 7th 32-bit word in the data frame is reserved for alignment, synchronisation and error detection. It includes an 8-bit comma character (K28.5) for alignment in the 8b/10b encoding, a 12-bit bunch-crossing identification (BCID) number and a 9-bit Cyclic Redundancy Check (CRC) [56] which is calculated and appended before the data is shipped out.



**Data Frame [@11.2 Gbps]**

W #	b 31	b 30	b 29	b 28	b 27	b 26	b 25	b 24	b 23	b 22	b 21	b 20	b 19	b 18	b 17	b 16	b 15	b 14	b 13	b 12	b 11	b 10	b 9	b 8	b 7	b 6	b 5	b 4	b 3	b 2	b 1	b 0
0	BC [4:3]			DATA1 [9:0]									DATA0 [9:0]									BC [6:5]			K28.5							
1	DATA4 [9:0]									DATA3 [9:0]									DATA2 [9:0]													
2	DATA7 [9:0]									DATA6 [9:0]									DATA5 [9:0]													
3	DATA10 [9:0]									DATA9 [9:0]									DATA8 [9:0]													
4	DATA13 [9:0]									DATA12 [9:0]									DATA11 [9:0]													
5	DATA15 [9:0]									DATA14 [9:0]																						
6	CRC						BCID [2:0]									BCID [11:0]																

W #	b 31	b 30	b 29	b 28	b 27	b 26	b 25	b 24	b 23	b 22	b 21	b 20	b 19	b 18	b 17	b 16	b 15	b 14	b 13	b 12	b 11	b 10	b 9	b 8	b 7	b 6	b 5	b 4	b 3	b 2	b 1	b 0
0	DATA2 [7:0]							DATA1 [11:0]							DATA0 [11:0]																	
1	DATA5 [11:8]				DATA2 [11:8]				DATA4 [11:0]				DATA3 [11:0]																			
2	DATA5 [7:0]							DATA7 [11:0]							DATA6 [11:0]																	
3	DATA10 [7:0]							DATA9 [11:0]							DATA8 [11:0]																	
4	DATA13 [11:8]				DATA10 [11:8]				DATA12 [11:0]				DATA11 [11:0]																			
5	DATA13 [7:0]							DATA15 [11:0]							DATA14 [11:0]																	
6	CRC						BCID_LOW [6:0]						BCID_HIGH [11:7]						K28.5													

W #	b 31	b 30	b 29	b 28	b 27	b 26	b 25	b 24	b 23	b 22	b 21	b 20	b 19	b 18	b 17	b 16	b 15	b 14	b 13	b 12	b 11	b 10	b 9	b 8	b 7	b 6	b 5	b 4	b 3	b 2	b 1	b 0
0	DATA2 [7:0]							DATA1 [11:0]							DATA0 [11:0]																	
1	DATA5 [3:0]			DATA4 [11:0]							DATA3 [11:0]							DATA2 [11:8]														
2	DATA7 [11:0]							DATA6 [11:0]							DATA5 [11:4]																	
3	DATA10 [7:0]							DATA9 [11:0]							DATA8 [11:0]																	
4	DATA13 [3:0]			DATA12 [11:0]							DATA11 [11:0]							DATA10 [11:8]														
5	DATA15 [11:0]							DATA14 [11:0]							DATA13 [11:4]																	
6	CRC						BCID_LOW [6:0]						BCID_HIGH [11:7]						K28.5													

**Figure 6.15.** The TREX data formats prepared for each type of FEX processor. The data fields are arranged according to FEX requirements and the tower granularity.

On a predefined BCID within the LHC orbit, where no collisions occur, the TREX provides align frames which contain auxiliary information rather than transverse energies. The transmission of an align frame simplifies the fibre and channel mapping on the destination modules. The auxiliary information includes metadata about the source device such as the serial number, geographical location and the intended destination.

Fig. 6.16 presents an example of the align frame data contents. Capturing the frame on the FEX processors allows for tracing the full data path and identifying any mismatching issues between the modules.

**Align Frame [@11.2 Gbps]**

W #	b 31	b 30	b 29	b 28	b 27	b 26	b 25	b 24	b 23	b 22	b 21	b 20	b 19	b 18	b 17	b 16	b 15	b 14	b 13	b 12	b 11	b 10	b 9	b 8	b 7	b 6	b 5	b 4	b 3	b 2	b 1	b 0
0	TREX_ID [7:0]							FEX_ID [1:0]	FIBER_ID [5:0]					BCID [11:0]																		
1	TREX_SRC_ID [31:0]																															
2																																
3																																
4																																
5																																
6	CRC						BCID_LOW [6:0]						K28.0						K28.5													

**Figure 6.16.** An example of the align frame issued by the TREX and its formatting. The second 32-bit word contains geographical information for identifying the TREX origin.

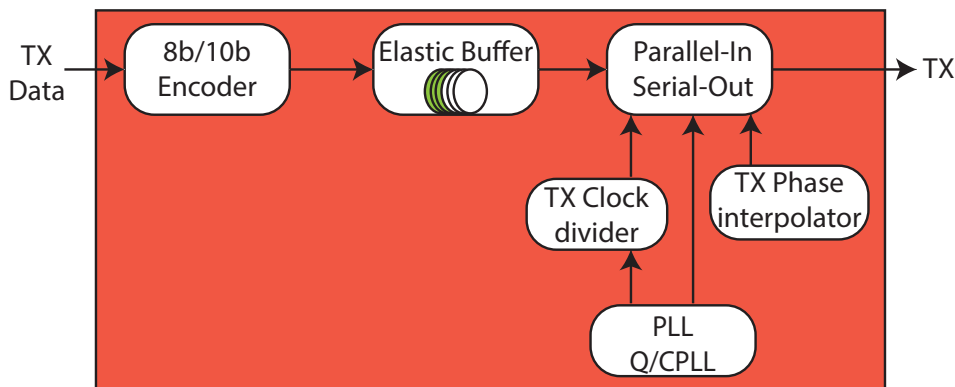


Fig. 6.17 shows the byte ordering of the encoder going from 32-bit input data mapped to 40-bits coded data.

## Transmission

The transceiver logic is the final block within the processing chain in the PREDATOR FPGA, where the high-speed serialised data is driven out of the FPGA to the opto-electrical FireFly transmitter. The formatted data passes through the 8b/10b encoder, then goes through a Clock-Domain Crossing (CDC) from the 280.56 MHz domain into the clock domain of the high-speed serialiser using an elastic buffer.

Fig. 6.18 show a simplified diagram of the data flow within the transmitter interface logic. The transmitter PLL receives as input a reference clock of 280.56 MHz, synchronous to the LHC clock and outputs a high-frequency bit clock. The high-speed serialiser, called the Parallel-In to Serial-Out (PISO) register, uses 2 clocks. The first clock is the bit clock of 11.2 GHz and another divided-down clock for clocking in the parallel data. However, this divided-down clock is not necessarily synchronous to the reference clock and can introduce latency uncertainties when performing the CDC. To achieve a deterministic latency in the transmitter interface, the technique described in [58] is implemented in the FPGA.



**Figure 6.18.** Simplified schematic overview of the high-speed transmitter interface implemented in the PREDATOR FPGA. The data prepared for transmission undergoes 8b/10b encoding and a clock domain crossing into the serialiser clock. At the end of the logic block, the data is serialised and driven out.

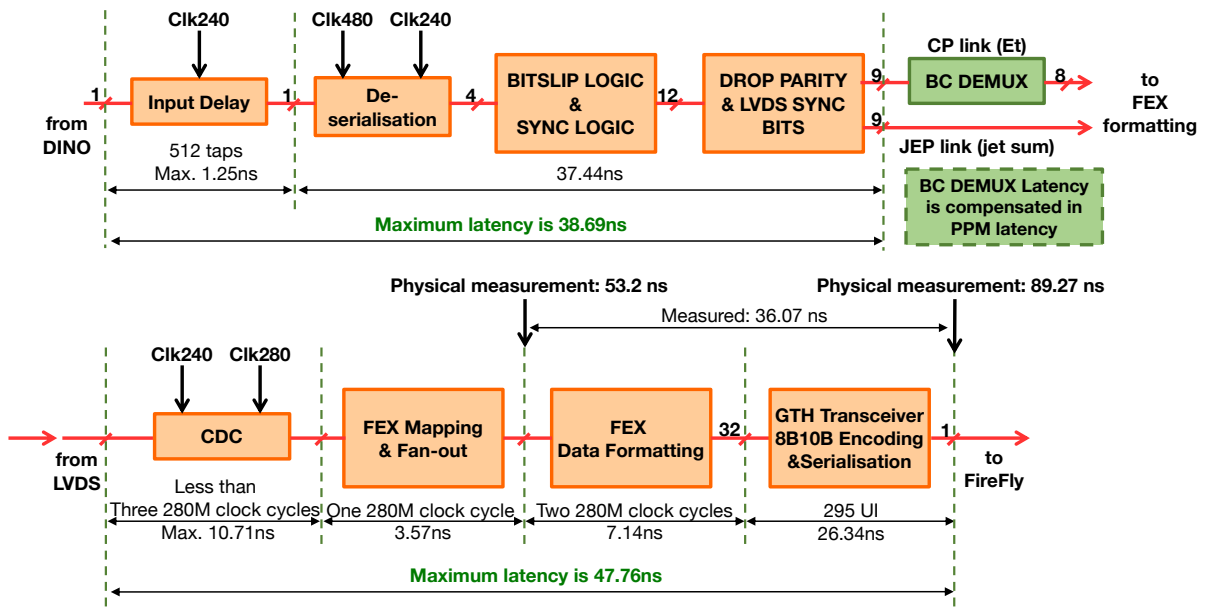
### 6.5.5 Latency

The latency introduced by the physical routing of the signals, the optical transmitters and the fibres used to carry the trigger information from module to module is fixed. However, the processing latency within the FPGA is design-dependent. The choice of sequential logic, and which clock domain frequency to process the data determines the internal

latency. As described in Sec. 6.5, the trigger path is time-critical (time sensitive) and its latency through the system must be kept at a minimum while performing all necessary processing tasks.

The latency is presented in units of nanoseconds and BCs, as the trigger operates in the LHC clock domain. The values are calculated using simulation and validated with real measurements. From the calorimeter inputs on the PPM up to the input of the PREDATOR FPGA the signal travel time takes **387 ns** or **15.5 BCs**.

Fig. 6.19 shows the real-time data path through the processing blocks of the PREDATOR FPGA. The first processing logic step that introduces **38.69 ns** of latency is the parallelisation of the incoming LVDS streams. The signal processing clock domain frequency is 240 MHz, which allows to perform 6 clock cycles of operations within one BC. The BC de-multiplexing is compensated within the PPM latency since the jet-sum links, which are not BCMUXED, require one additional clock cycle at the summing stage in the nMCMs and therefore arrive later. This time difference is used in the PREDATOR to perform the de-multiplexing operation.



**Figure 6.19.** The real-time data flow in the PREDATOR FPGA and the estimated latency values. The values shown in green are obtained from the simulation of the logic. Values obtained from physical measurements are highlighted at two stages and show the latency starting from the LVDS input stage up to the measurement point.

The next set of processing logic is operated at the FEX clock frequency of 280 MHz. The parallel data of 64 channels go through a clock domain crossing (CDC) which takes three 280 MHz clock cycles. The CDC, mapping, formatting and high-speed serialisation take **47.76 ns**. In total, the PREDATOR latency is estimated from simulation to be **86.45 ns** or **3.5 BCs**. Adding the values to the PPM latency shows that the complete system is well within the required **480 ns** latency envelope. The measurements of the

physical latency of the PPM and TREX is presented in Sec. 7.5.

## 6.6 The Readout data path

All ATLAS sub-detectors store their corresponding event data in pipeline memories and wait for a positive trigger decision by the Central Trigger Processor (CTP), in the form of a L1-Accept signal. Upon receipt of the L1A, the event data is transferred to the ATLAS DAQ for combined event building. If the L1A does not arrive in a specified time, the data in the pipelines is overwritten with new events and the original event is lost.

This also applies to the PreProcessor, where the event data is used for monitoring, verification of the trigger and calibration purposes. The event data is stored in pipelined memory buffers located inside the nMCMs and contains a configurable amount of data such as the number of raw flash ADC slices, values,  $E_T$  results that were transmitted to the downstream trigger processors, pedestal correction values and error flags for the event.

The TREX maintains the transmission to the legacy Readout Drivers (RODs), but also a new interface to the Front-End Link Exchange (FELIX) is introduced. The readout data is provided via both interfaces in parallel, but the FELIX interface is the baseline for all upgrade modules. This will also ensure support and longevity throughout Run-3 as the RODs are not actively supported anymore and the DAQ is moving to a more streamlined solution across all ATLAS subsystems. At a later stage during Run-3, only the FELIX interface will remain active. The individual paths are described in more detail below.

### 6.6.1 Legacy Readout-Driver Interface

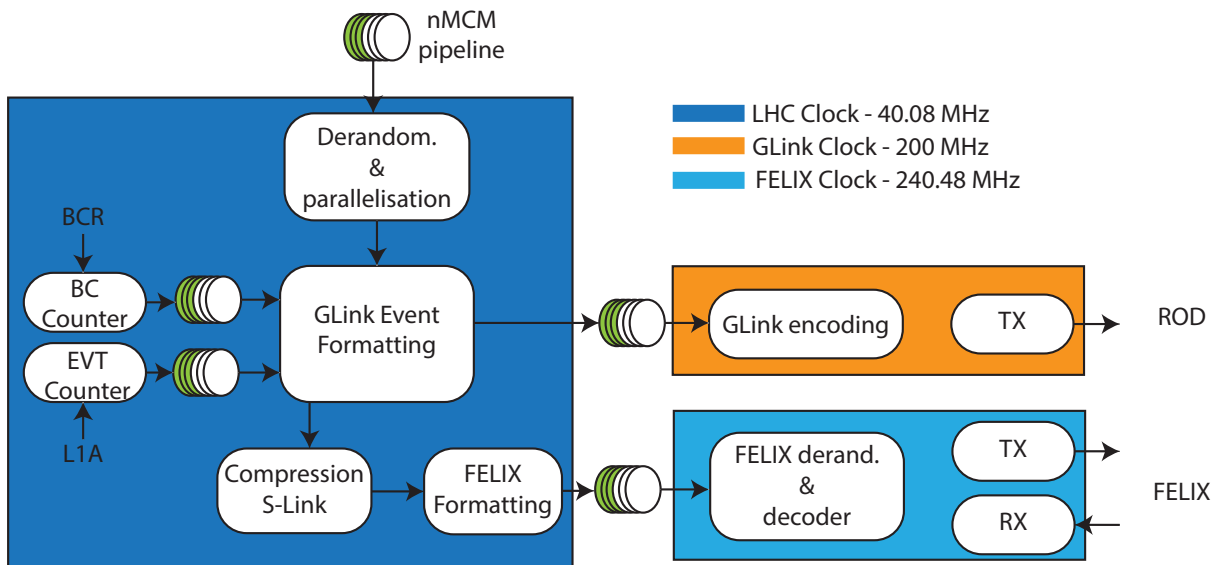
In Run-2, the data collection and processing were done on the ExtReM FPGA. The event data stored in each nMCM pipeline buffer was packed and transmitted via 2 serial streams to the ExtReM. In total, there are 32 serial streams with each stream carrying data from 2 channels. The data was formatted, packed and shipped to the GLink Transmitter Modules - Optical Tx (RGTM-Os) card slotted in the back of the PPM. The Agilent (HP) Chipset [59] on the RGTM-O encoded the data into the GLink protocol and sent it to the legacy Readout-Drivers via a single optical link running at 800 Mbps.

For Run-3, the readout functionality is moved to the TREX, with the serial lines routed through the ExtReM directly to the PREDATOR FPGA. As the RGTM-O card is removed to accommodate the TREX modules, the encoding and transmission functionality of the Agilent Chipset also needed to be implemented in the PREDATOR firmware design.

The incoming serial data is divided into 16 parallel GLink channels, that correspond to single data lanes. The data is placed into derandomiser buffers to cope with the L1A rate fluctuations during the transfer chain to the downstream DAQ system, minimising the system dead-time. Each channel block processes data from 2 input streams and as the data is multiplexed, it amounts to 4 nMCM channels per GLink channel.

As the event format has been kept identical as it was in Run-2, the event data is packed into the ROD-specific format, referred to as the GLink Event Data Format. It comprises of 16 individual serial streams that contain a locally computed 12-bit BCID, error flags, parity bits and data from the nMCM channels.

When receiving an L1A signal, the triggered event is tagged with a unique identifier. For that purpose, two quantities are used - the Level-1 ID and the BCID. These quantities are independently computed both on the nMCMs and the TREX based on the incoming TTC signals described in Sec. 6.4. As the nMCMs are also counting the same quantities and sending the 4 LSBs to the TREX, a comparison is performed between the two counts. In case of mismatches, the corresponding error flags are set to indicate faults in the synchronisation. For each triggered event, the L1ID and BCID quantities are written into FIFOs and read out during the GLink formatting step. They required careful tuning of the architecture and width to properly align with the incoming nMCM data.



**Figure 6.20.** A simplified schematic diagram of the TREX readout path implemented in the FPGA. The three processing clock domains are presented in different colours. Between each clock domain a buffer symbol indicates the crossing.

### GLink encoding

The second stage is preparing the data for the transmission. First, there is a CDC from the LHC clock to the 200 MHz free-running clock domain that the transmission logic

uses. Afterwards, the 16-bit wide data stream is encoded into 20-bits for the GLink transmission, following the protocol described in [59] and sent out to the ROD modules.

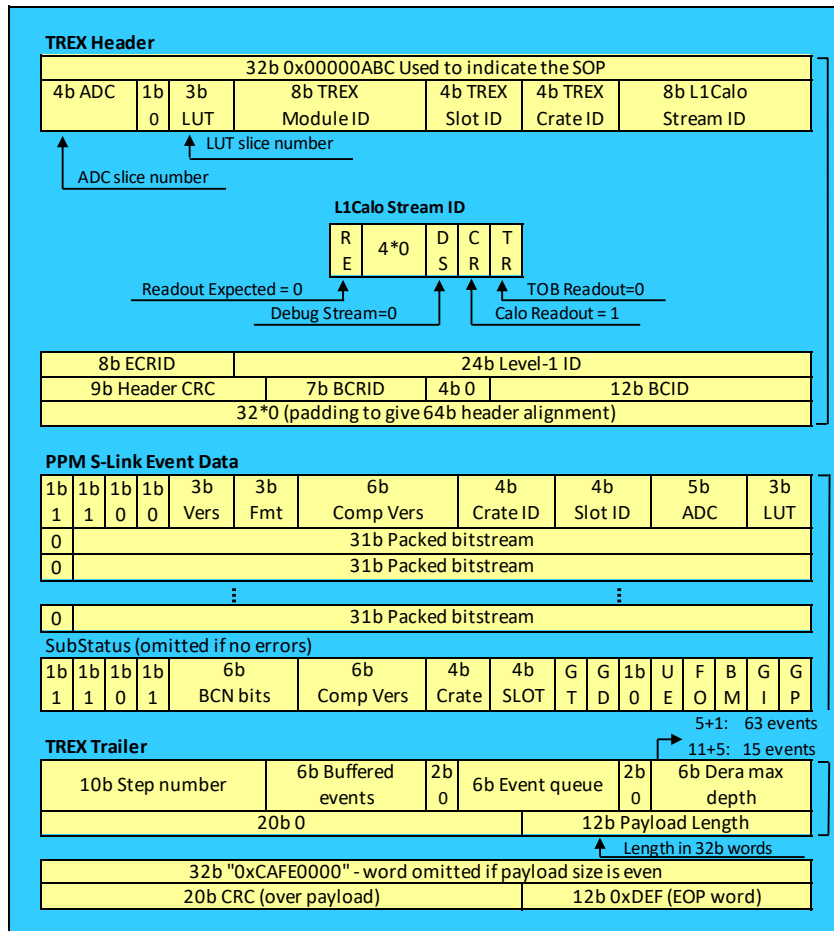
### 6.6.2 FELIX interface

The FELIX aims to replace the custom readout hardware with more streamlined and commercial components across all subsystems of ATLAS. It receives the data from the sub-detectors via high-speed optical links running at either 4.8 Gbps (GBT) or 9.6 Gbps (FULLMODE) and transfers it to the software-based Readout Driver (SW ROD) [60]. For the TREX system, the transmission is performed via the 9.6 Gbps link speed with 8b10b encoding, similar to the transmitter interface of the real-time path.

To reduce the efforts on the DAQ, such as re-writing new software tools, as e.g. bytestream decoders for a new data format, the packets sent to the FELIX and subsequently to the SW ROD maintain the core legacy data format within the payload. This way the event information can be extracted using existing software tools, regardless of whether a new interface is used or not.

The simplified data flow diagram of the FELIX path is presented in Fig. 6.20, where the various clock domains are highlighted in individual colours. Compared to the legacy ROD path, the data to FELIX-SW ROD undergoes an additional processing stage.

On the legacy ROD hardware, the data gets compressed to minimise the overall size within the ATLAS event. Since the FELIX path bypasses the ROD, the compression needs to be realised either in the software or in the TREX hardware. During the first attempts, the compression was done as part of the SW ROD custom processing routine, so that the event data in the FELIX packet could remain in the GLink Event format.

**TREX Readout Packet to FELIX**

**Figure 6.21.** The formatting of the TREX data sent to the FELIX. It is composed of four 32-bit header words, which include metadata describing the event such as the Level-ID and the BCID of the triggered event, the number of transmitted read-out ADC and Look-Up Table (LUT) values and also geographical information. The S-Link event data starts with a header of its own, followed by the dynamically sized and compressed event data. The TREX trailer is attached at the end, containing further auxiliary information and status flags.

However, the compression algorithm proved to be quite CPU intensive, which affected the readout performance of the system. It was therefore decided, to implement the compression algorithm within the TREX. The compressed data, referred to as the S-Link format [61], would then be placed into the FELIX payload instead of the GLink format. The FELIX collects and passes the received data from the TREX to the SW ROD, where the custom processing routine strips the payload from additional headers, leaving only the S-Link formatting for the ATLAS event builder to process.

Fig. 6.21 depicts the ordering of the packet sent to the FELIX. It is composed of 32-bit words, the first five words make up the TREX header. Here, the first word includes hardware metadata such as the module identifier, geographical address and the readout configuration. The second and third words contain the Level-1 ID and the BCID, which are used to synchronise the data with the timing information received through the TTC

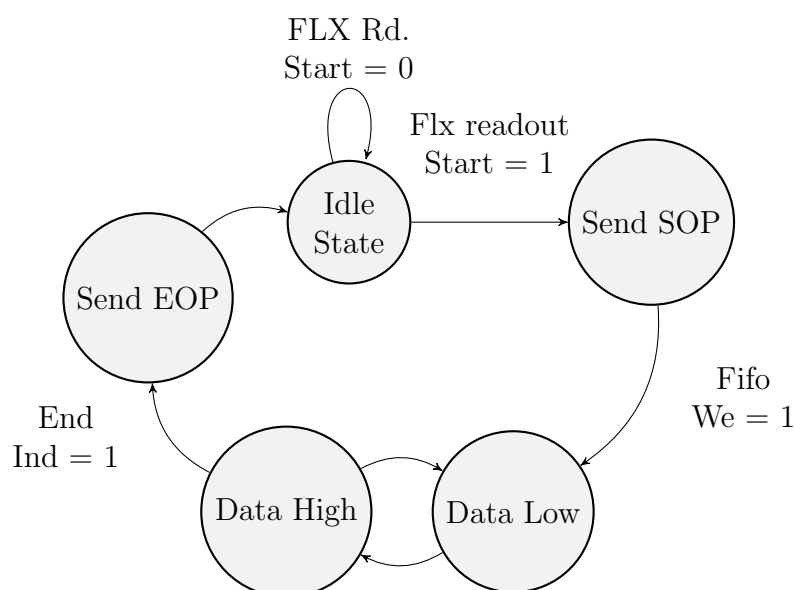


network. The payload follows the header with a variable number of 32-bit words. At the end of the packet, a trailer is attached containing error flags and further diagnostic information and the result of a CRC-20 calculation.

After performing the compression, the output data is stored in a CDC buffer to cross from the LHC clock domain over to the 240.48 MHz domain in which the data is formatted and shipped. The buffer is 64-bits wide, and the FSM responsible for the CDC packs two 32-bit words per single LHC clock cycle. A consequence of this is that the entire FELIX packet always contains an even number of 32-bit words.

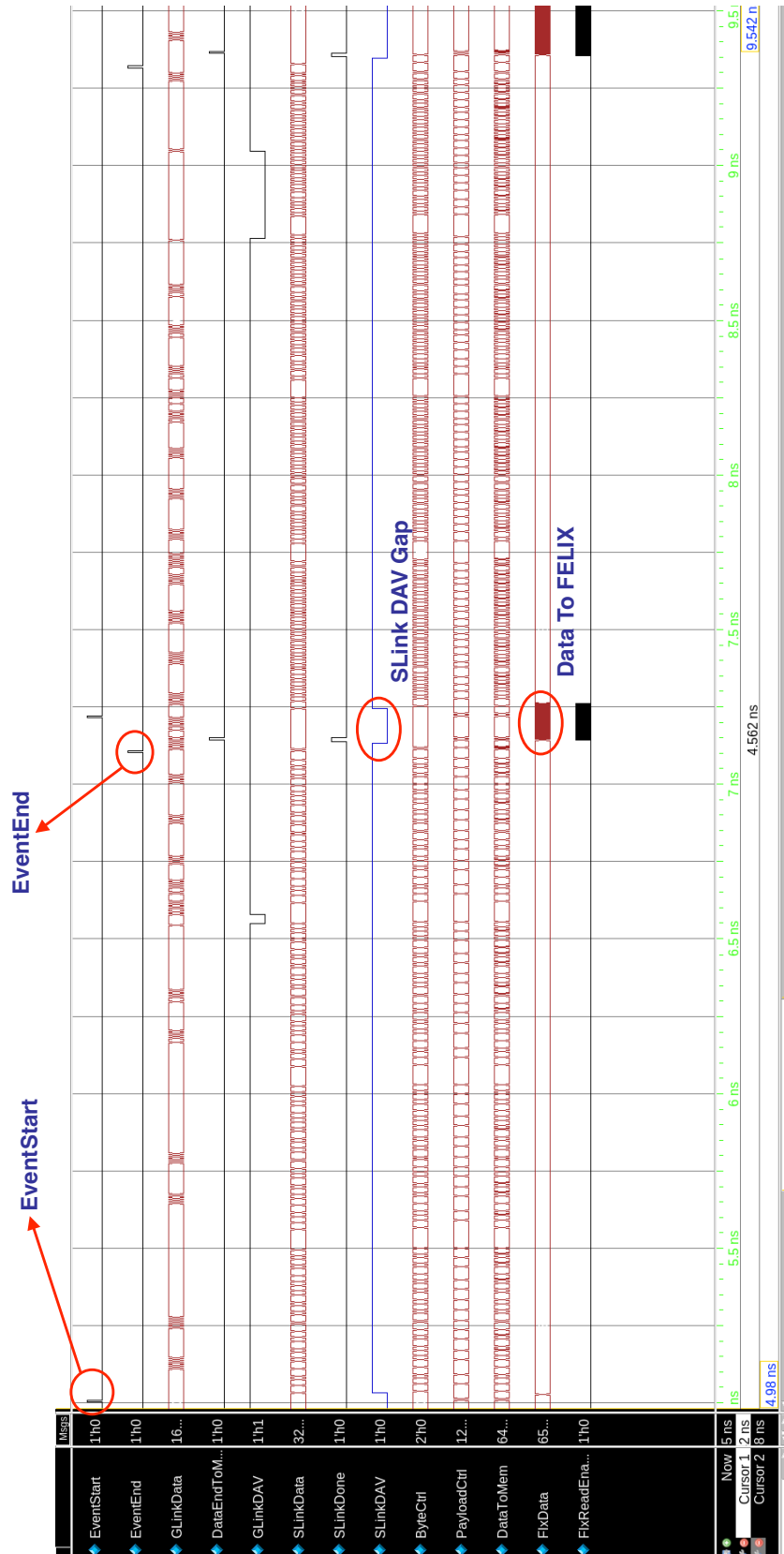
Fig. 6.23 presents the data flow in time, for two consecutive events at a trigger rate of 100 kHz. The *GLinkDAV* (data available) flag shows the processing of the data into the GLink format. Once a single event is fully formatted, the flag is driven low. The output *GLinkData* is fed to the S-Link formatting, where the start of the next stage of processing is indicated by the *EventStart* flag. A similar *SlinkDAV* indicates the state of the formatting. The *ByteCtrl* and *PayloadCtrl* control counters manage the FSM that packs the *SlinkData* into the CDC memory buffer.

As soon as the event formatting is done, the *EventEnd* flag is raised. A few more clock cycles are required to attach a trailer in the *SlinkData* and once completed, the *SlinkDone* flag is raised. The done signal is propagated to the FSM in the FELIX formatting logic, running at 240.48 MHz. The done flag acts as the start signal to read from the CDC memory, indicated by the signal *FlxReadEnable*. The *DataToMem* shows the data written into the memory at 40 MHz, while the *FlxData* is read from the same memory at the higher clock speed. The fast data processing occurs within the *SlinkDAV* gap between two events.



**Figure 6.22.** Finite-State-Machine for initiating the transmission to the FELIX.

Fig. 6.22 illustrates one of the FSMs after crossing to the higher 240.48 MHz clock domain. The generation of the signal that initiates the FSM from the idle state is coupled with the flag which indicates the completion of the event data compression. Before sending the core data, a Start of Packet (SOP) 32-bit word is sent to the FELIX, which acknowledges the transfer. Once the data is fully sent, an accompanying End-of-Packet (EOP) word is issued, terminating the transmitted message. A message contained within the SOP and EOP is called a FELIX chunk.



**Figure 6.23.** Timing diagram illustrating the processing of events at a trigger rate of 100 kHz. The diagram shows the data flow through the GLink formatting, compression and packing into the FELIX format. The start and end of processing an event are indicated by the *EventStart* and *EventEnd* flags. The gap between two events is indicated when the *SlinkDAV* is driven low. During this gap period, the readout to FELIX is initiated and shown in *FlxData*.

## 6.7 The System-On-Chip and Monitoring

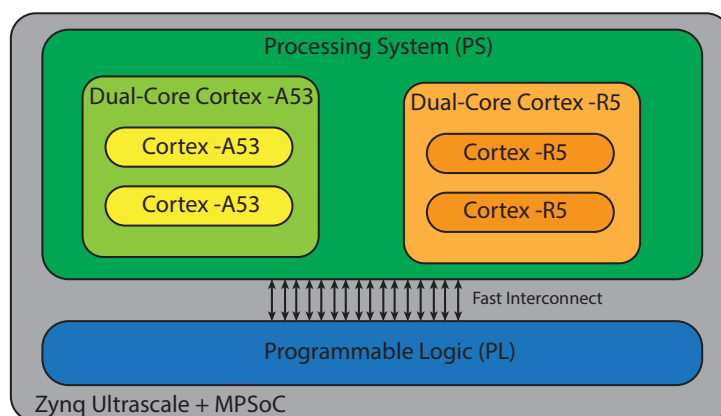
Lately, embedded devices have taken off in popularity due to their multi-functional purpose, low power consumption and fast interconnects.

The term SoC is quite broad and can refer to many variations of the components on the same die. In the context of this thesis, an SoC is defined as a device which integrates Programmable Logic (FPGA) with a 64-bit ARM Cortex [62] based Processing System, on-die memory and fast interconnects.

The main function of the TREX System-On-Chip (SoC) is to gather environmental data, assess the overall health status of the board and transmit the information to the ATLAS Detector Control (DCS) System [63]. The ATLAS DCS is an industrial Supervisory Controls and Data Acquisition (SCADA) system which is responsible for the continuous monitoring and supervision of the detector equipment. Each individual sub-component in the ATLAS detector is carefully monitored to ensure stable data-taking and quick action in case of a hardware failure, hence the TREX modules are required to propagate their health information to the DCS.

For monitoring, configuration and communication access to each TREX module, an SoC device is suitable for acting as a second and independent compute unit in addition to the crate SBC. Each module can be directly accessed utilising an Ethernet-based protocol or via the VMEbus, giving it more flexibility and redundancy.

The Multi-Processor System-On-Chip (MPSoC) chosen for the TREX includes a dual-core ARM Cortex-A53 Application Processing Unit (APU) and a dual-core ARM Cortex-R5 Real-Time Processing Unit (RPU). A simplified diagram of the Zynq SoC structure is depicted in Fig. 6.24.



**Figure 6.24.** A simplified diagram of the PL and PS sectors of the TREX Zynq.

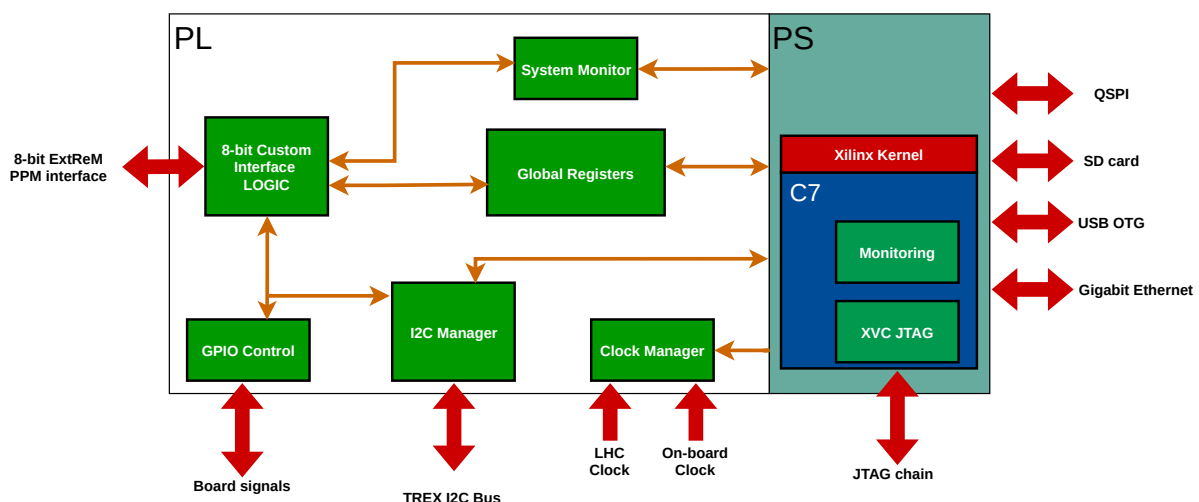
In most cases, the SoC can be found in a System-On-Module (SoM) format. The SoMs are small PCBs which house the SoC and any external peripherals that are required to operate it, such as external memory and storage devices, oscillators for generating the

system clocks and power regulators.

The TREX is designed to house an SoM with a form factor of  $4 \times 5$  cm using rugged connectors. The SoM of choice is the TE0820 manufactured by Trenz Electronics [64], which hosts a Xilinx Zynq UltraScale+ XCZU3CG-1SFVC784E, 2 GBytes of DDR4 SDRAM, a 128 MByte QSPI-based flash memory and an 8GByte eMMC storage module.

The Operating System (OS) with user applications, coupled with custom protocols implemented in the Programmable Logic maximises the available communication interfaces. A list of the requirements is presented below:

- Interface to the PPM for communication and programming
- Access to all on-board components for gathering environmental data
- Configuration and programming of onboard FPGAs - the TREX module should be fully programmable when VME access is not available
- Power management and intelligent switching for powering down the core power rails
- Redundant access to monitoring data via VME or through Ethernet
- Socket-based user applications for accessing the device over the network.
- Software monitoring framework serving data to the ATLAS Detector Control System



**Figure 6.25.** Schematic overview of the logic blocks of the TREX Zynq PL and PS.

Fig. 6.25 illustrates an overview of the various logic blocks of the TREX Zynq PL and their IO interfaces.

### 6.7.1 The Programmable Logic

On the FPGA side of the SoC protocols specialised for communicating with the PPM and the onboard devices are implemented.

The design of the PL sector takes advantage of the wide selection of programmable input and output signals connected to the onboard components and logic circuits. For the data transfer between the ExtReM FPGA on the PPM, the same 8-bit custom protocol is implemented which has been described in Section 6.3.2. General Purpose Input/Output (GPIO) signals provide board control and status flags to and from the Zynq.

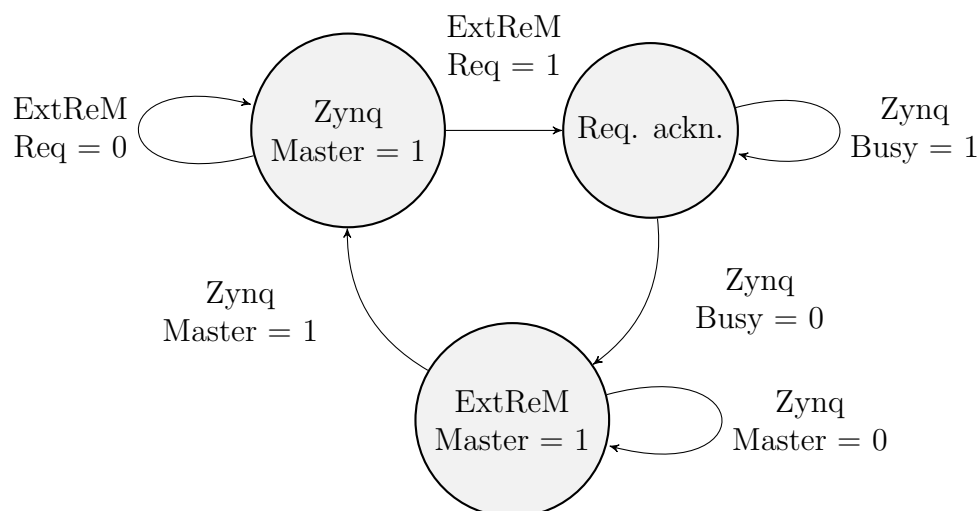
#### I2C bus access and control

For gathering the monitoring data, the Zynq must have unconstrained access to the TREX I2C bus. Therefore, three methods of communication interfaces are implemented in the Zynq PL. The first two methods use a wishbone-compliant I2C core written in HDL [65] for collecting the data:

- Through the 8-bit custom protocol - the I2C command is forwarded by the ExtReM FPGA to the I2C master core in the Zynq PL.
- The PS initiates the I2C transfer FSM which communicates with the I2C master core in the Zynq PL.
- The PS communicates directly through an internal driver with the I2C bus.

The access is ensured by a Finite State Machine which arbitrates the master functionality of the bus between the ExtReM FPGA and the Zynq, depicted in Fig. 6.26.

The procedure ensures that all ongoing transfers are completed before the hand-off, as an interrupted transfer can cause the bus to become unresponsive. 3-options of access redundancy are in place, eliminating single points of failure for the monitoring of the health status.



**Figure 6.26.** Finite-State-Machine for arbitrating the I2C bus master between the ExtReM and the Zynq.

### Clocking network

The clock network of the Zynq comprises of 3 clocks from independent sources:

- LHC Clock - Used as a synchronous system clock for the PL
- Standalone 33.33 MHz crystal oscillator clock which drives the PS
- Si5338A Programmable oscillator for driving the PL in case the LHC clock is not available - programmed with a frequency of 40 MHz

When the modules are powered up, the LHC clock becomes available only after the ExtReM FPGA has been configured with a firmware bitstream. To start the initialisation and monitoring procedure on the Zynq, the external oscillator is used as the system clock. Upon detecting the LHC clock, the system clock source is switched. The synchronous LHC clock is required to carry out data transactions over the custom 8-bit protocol.

### Power control

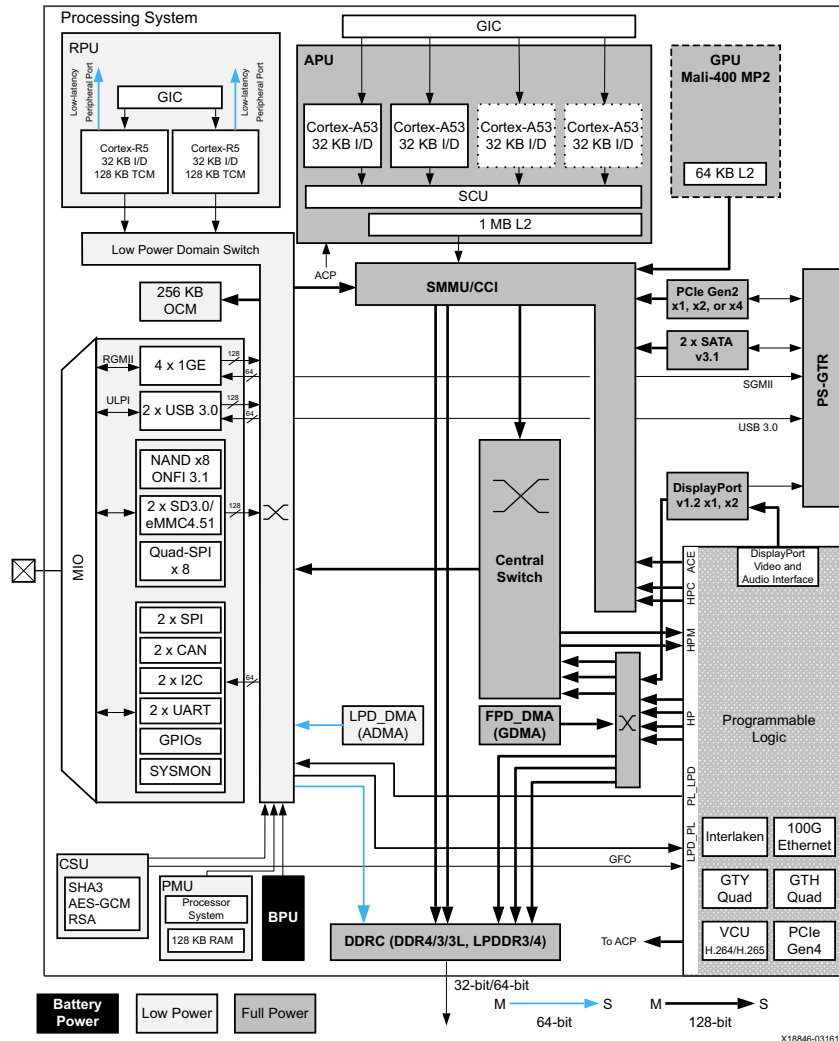
The Zynq implements multiple options for controlling the TREX power state. The first option is through the onboard LTC2997 power management unit. As this device interfaces with the I2C bus, the Zynq can program and retrieve the power state information. It has also the ability to shut off the power DC-DC rectifier units through the power manager.

The second option of power control is via the non-volatile IO expander unit, using the I2C protocol. However, in this case, changing the state of the digital IO pins in the expander controls directly the rectifiers, bypassing the power management unit altogether.

The third and classified as a last resort method is a direct output connection from the Zynq to the power switching circuit. Here the user can choose to control both the

rails powering the FPGAs and the 3.3 V supply which powers the I2C bus. The power regulator of the Zynq is always kept in an operational mode and can be used to perform a complete power cycle of the board if necessary.

The last option available is the power control of the Zynq module itself. This can be performed only via the PPM, through the ExtReM FPGA. In case of a failure on the Zynq, which leaves the devices unresponsive through the ethernet-based communication, the Zynq configuration can be reloaded via VME commands.



**Figure 6.27.** The schematic overview of the Zynq PS [66]. The various processing and IO units are illustrated along with their internal interconnects.

## 6.7.2 The Processing System

The dedicated silicon element on the Zynq is the Processing System (PS). It is made up of a set of processing units, interconnects, on-chip memory resources and interfaces. The block diagram of the PS architecture is shown in Fig 6.27. The Cortex-A53 cores operate at frequencies of up-to 1.5 GHz and each have 32 KBytes of fast Level-1 cache memory.



## The Bootloader and the Kernel

The booting procedure of the Zynq is performed through multiple stages, with each step customised for the TREX use case. The stages are described below in sequential order during boot-up:

- **The Platform Management Unit (PMU).** The first element to be started is the PMU, which serves as a power and security manager. It is responsible for controlling the power of the peripherals, managing the clocking, reset networks, and controlling the wake-up and sleep modes based on certain triggers.
- **The Configuration and Security Unit (CSU).** The CSU reads the first-stage boot loader binary code and loads it into the on-chip memory (OCM). The boot mode i.e. the location from where to read the First-Stage Boot Loader (FSBL) is determined by external pins on the Zynq. Typical FSBL read options include JTAG, USB (Device Firmware Upgrade) protocol, SD card, eMMC, or from a QSPI memory module. In the case of the TREX module, the boot mode is set to the SD card by default, but it can be dynamically changed between the SD card or the QSPI via the IO expander module. The JTAG option is always available regardless of the state of the pins. The internal BootROM procedure looks for a valid boot header in the binary stream called the "Golden Image Search", where an identifier string "XLNX" is searched along with a valid header checksum. For the SD card boot method, a FAT16 or FAT32 filesystem is required. Once a valid bootloader image is found and loaded into the memory, the instructions can be executed by the Application Processing Unit (APU).

The CSU is implemented as a triple-redundant embedded processor with ROM and RAM attachments. For security, the CSU has a cryptographic interface containing multiple encryption methods. This is available for applications that require authentication and tamper-proof requirements. The final interface is the processor configuration-access port (PCAP) which is used to configure the PL FPGA directly from the FSBL. This means the FPGA firmware binary can be stored in a memory location after the FSBL inside the SD card or the QSPI storage module and doesn't require a separate configuration method.

- **The First-Stage Boot Loader (FSBL).** The FSBL is executed by the APU and configures the entire system. Once the FSBL is executed, the configuration of the PL is initiated through the PCAP interface found on the CSU. Loading of the ARM Trusted Firmware (ATF) and the second-stage bootloader or a bare-metal application from the non-volatile memory into the active DDR memory follows next. Custom user code can be executed to program onboard peripherals such as

external clock generators or the initialisation of the Ethernet PHY which is needed during the next stages of the boot sequence and in operation.

- **The Second-Stage Boot Loader (U-Boot).** U-Boot is an open-source boot loader for embedded processors with various architectures [67]. It is the last stage before the operating system and gives the user more configuration options with wider access to the attached peripherals. A device tree configuration file is read by the bootloader, which is a data structure describing the embedded hardware i.e. the CPU type and the number of cores, memory resources and the mapped IO, access location within the memory space and command line arguments for the Kernel. Since it is quite universal, the data format for the device tree can be used to describe custom user elements and enable peripherals during boot time.

The devices are available upon loading the device tree within the bootloader environment, however, it is also passed to the Kernel at the Linux run time. With the device tree loaded, as an example, the Ethernet interface becomes available and can be used to download the Kernel image from a dedicated server using the BOOTP protocol instead of a locally installed image. This is suited for larger-scale systems with many nodes that need to be booted. It saves space and makes upgrading the Kernel image more convenient. The network booting option was used for the TREX in the laboratory test-bench environment, but the preferred method for running in ATLAS is that each TREX module includes a full image on the local storage medium. With the requirement that, even in the case of a network failure, the modules need to boot undisturbed and provide monitoring data.

- **The Linux Kernel.** For building a custom Linux kernel, open-source projects from Xilinx [68] and Yocto [69] have been considered. Both platforms offer a framework to create a custom kernel suitable for the current use case of the embedded device. The TREX uses a custom kernel based on version 4.19 of the official Xilinx kernel. This decision is made due to the choice of the Linux distribution, which is described in the next section. The custom kernel brings access for the necessary drivers to interface with the TREX I2C bus and the interconnection pins to the PL (FPGA) side.

## The Root Filesystem

At the time of writing this thesis, the CERN computing infrastructure is based on the CentOS 7 (C7) Linux distribution [70]. From local desktop computers to the Worldwide LHC Computing Grid, all nodes use CentOS 7 with CERN-specific customisations (CC7). Therefore the ATLAS software, both for hardware control and data analysis is written for this distribution.

Taking this into account, it creates a baseline for choosing the C7 distribution for the SoCs. While the x86 architecture is fully supported, compiling a version for embedded systems has a few caveats. One example is the Kernel support. The TREX SoCs are running on a CentOS 7-based root filesystem coupled with a custom Xilinx-based kernel. C7 is a stable release with a 10 year support cycle. The kernel was fixed at version 3.10 at the time of its release, which did not include support for any ARM-based processors, in particular the 64-bit *aarch64* architecture. ARM support was introduced with kernel version 4.0 and onwards and adding support retroactively to an earlier kernel version was not an option.

However, using a custom kernel with ARM support and the C7 root filesystem proved to be a reliable solution for operating the TREX Zynq modules in ATLAS. The C7 root filesystem provides the full feature set such as systemd and the yum, dnf package manager. This means that applications are cross-compiled on a more powerful *x86\_64* node and the resulting binary executables and shared libraries are transferred to the Zynq for execution.

The advantage of using C7 on both the x86 and aarch64 systems is that all dependencies and packages can be maintained at the same version, such that during run time no dependency version mismatches will occur. The other advantage is the dnf package manager that can fetch and install pre-compiled aarch64 packages from remote repositories which are one-to-one compatible with their x86 counterparts.

## Boot methods

Now that the individual steps of the booting process have been covered, the implemented boot methods can be discussed. During operation, a reliable and redundant booting mechanism is required. In the case of single-event upsets or faults on the storage medium, a second independent option must be considered.

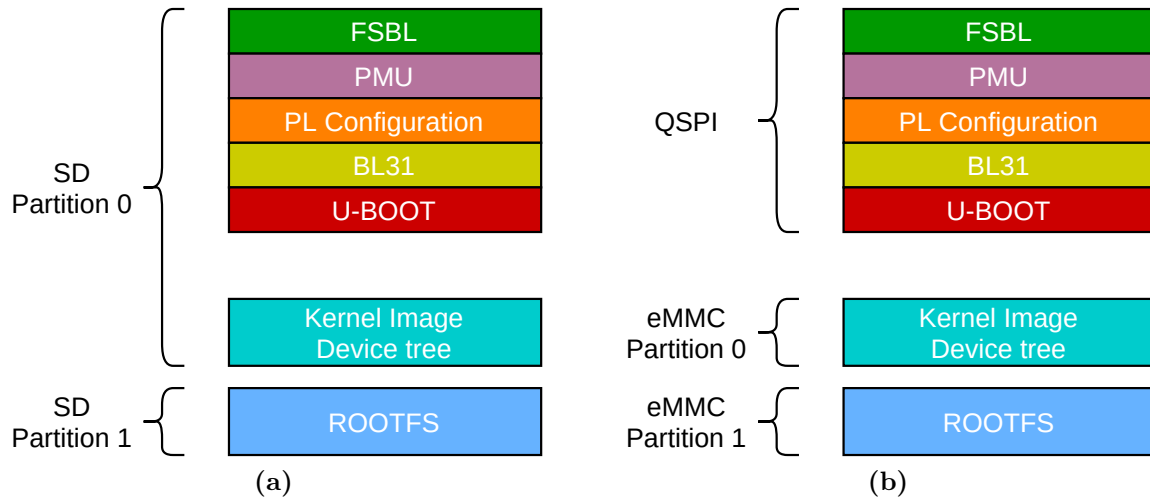
The TREX uses the SD card medium as the baseline for configuring the PL and loading the operating system. The SD card is partitioned into 2 filesystems. The first partition called *boot* uses the FAT32 format and has a size of 1GB. Herein a binary BOOT.BIN file is stored which contains the FSBL, PMU, ATF and U-boot and the firmware bitstream for the PL, while a separate image called *Image.ub* contains the device tree and the Linux kernel. The second partition uses the rest of the available space on the SD card and houses the root filesystem.

As a fallback solution, in case of a failure of the SD card a second boot option is in place. This is to ensure continued monitoring of the device's health status when there is an ongoing data-taking run and physically swapping the SD card is not feasible at that moment.

Via VME, the Zynq boot method can be changed by writing to the IO expander.

After a power cycle, the Zynq will pick up the changed boot pin configuration and start booting from the onboard QSPI flash memory. As the flash memory is rather small and can contain only the FPGA firmware and the bootloaders, the kernel image is stored on the first partition of the eMMC storage module, while the root filesystem is placed on the second partition.

Fig. 6.28 presents the two booting methods of the TREX Zynq SoC. The baseline and fallback solutions with their corresponding storage locations.



**Figure 6.28.** The TREX Zynq booting methods for loading the PL bit stream and the operating system. The nominal configuration is done via an SD Card (a). The fallback solution (b) uses a combination of QSPI memories and eMMC storage.

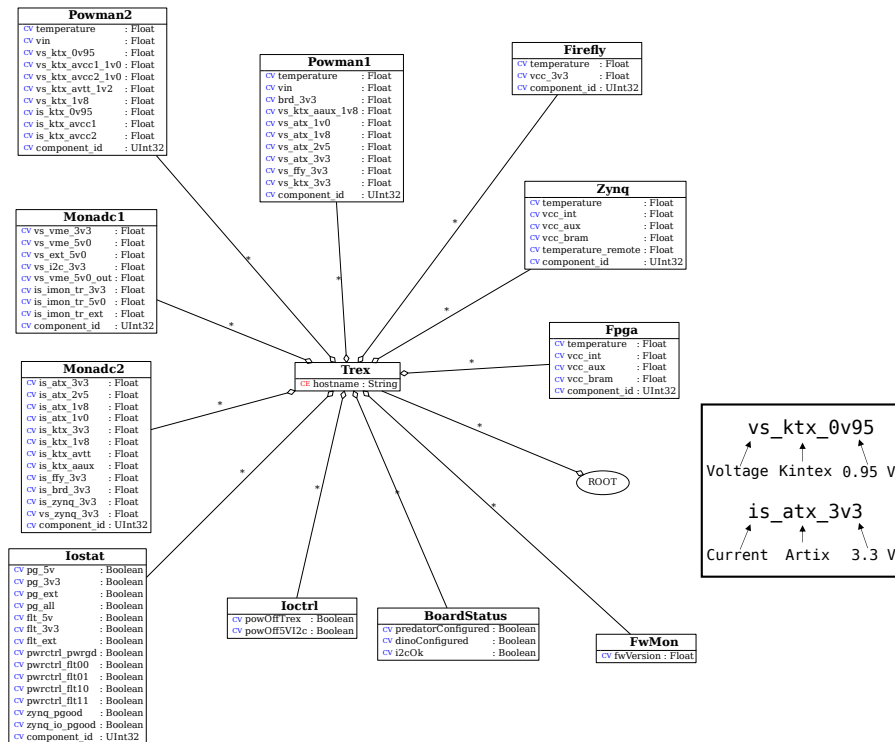
### 6.7.3 The Monitoring Framework

The core function of the Zynq SoC is the collection of health information from the onboard components and forwarding the information to the ATLAS Detector Control System (DCS). For this purpose, a monitoring framework is developed which includes the necessary driver interfaces to connect to the local TREX hardware, combined with an industry-standard machine-to-machine communication protocol called Open Platform Communications Unified Architecture (OPC UA) [71]. The ATLAS DCS has chosen the OPC Unified Architecture standard for transferring data between the sub-detectors and the control systems. The choice of the standard is driven by its performance, data model universality and modularity.

#### OPC UA Server

The OPC UA server is generated via the quasar framework [72]. The user designs a device-specific object-oriented information model, upon which the server code is generated. The

generated server code is then integrated into the TREX monitoring application that enables the OPC UA interface. Any OPC UA compatible client can subscribe and retrieve the published data. Fig. 6.29 depicts the TREX datapoint elements monitored by the framework.



**Figure 6.29.** The datapoints elements and their corresponding types in a hierarchical illustration, generated via the quasar framework and adapted. An example of the datapoint notation is shown on the right.

## Component drivers

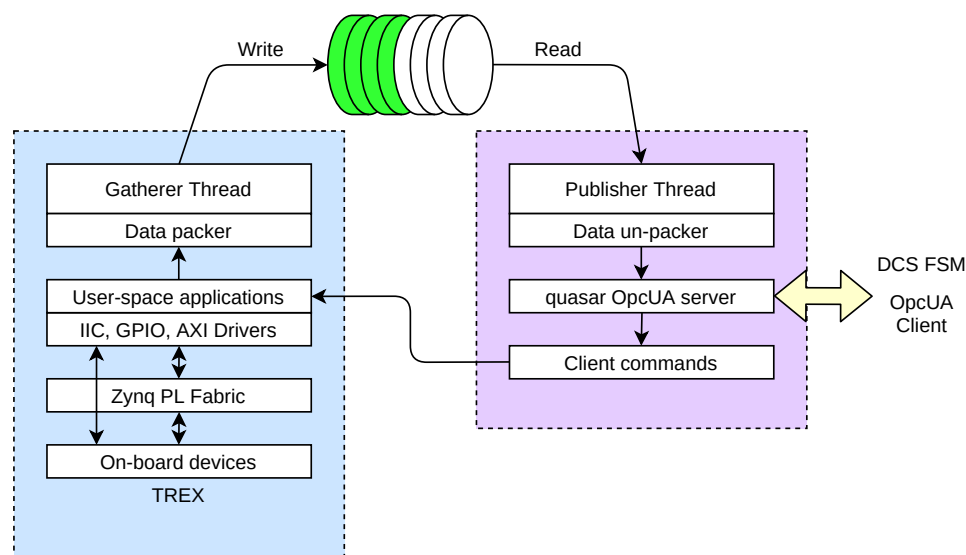
The next part of the monitoring framework is the individually prepared classes for communicating and retrieving the sensor information. The objects are designed to have a multi-functional nature and can be used in standalone applications outside of the monitoring environment. This can be defined as part of the "back-end" of the monitoring framework. The various classes and their functionality is listed below:

- **Control.** The Control segment declares memory-mapped IO signals which handle the data transfer handshaking flags, power control and the provide the current configuration state of the TREX FPGAs. The addresses for the memory-mapped IO signal registers are also defined within this segment, such as the base address and their corresponding offsets for driving the pins as either input or output and for reading its value.

- **Communication Protocol.** This segment is the backbone of the data transfer between the application and the hardware components. It defines the structure of the TREX I2C bus and provides the functions for reading and writing to and from the hardware, respectively. The set of functions creates a level of abstraction for the user applications, where a single call performs all the necessary initialisation routines. It is the parent class for the individual component instances and also declares the control segment for IO steering.
- **Monitoring ADC.** The set of functions provides access to the two onboard ADCs, where the FPGA currents and voltages are measured. During the object instantiation, the set-up routine is performed to prepare the ADC for data collection. The functions included within the class object return the monitoring results with all the necessary conversion factors applied.
- **Power Manager Monitor.** The object provides access to the two Power Manager units and their register map consisting of 8 pages. The format of the data is using an 11-bit signed mantissa and a 5-bit signed exponent. The conversion to floating-point values is included in the output results.
- **FireFly Monitor.** The FireFly class object is dedicated to all 5 optical transceiver modules. Retrieval of temperatures and voltages as well as channel power control is implemented. Generic access to any of the registers within the memory map is available to the user. The temperature data is formatted in signed bits, 2s complement with an accuracy of  $\pm 3^\circ C$ . The voltage value is a 16-bit integer given in units of  $100 \mu V$ , and the accuracy of the measurement is  $\pm 3\%$ .
- **Predator Monitor.** The Predator class retrieves the junction temperature and the core, IO voltage values from the main processing FPGA (PREDATOR). Here a conversion factor is applied for both the temperature and the voltage results.
- **Dino Monitor.** This class communicates with the four DINO FPGAs to fetch their monitoring data. For every function call, the communication to the four FPGAs is done in sequential order. Similar to the Predator class object, the conversion of the monitoring information to the appropriate units is built-in.
- **Zynq Monitor.** Last, but not least, the Zynq needs to monitor itself. Here, the monitoring data is gathered through a kernel call which retrieves the voltages and temperature from the internal system monitor.

## Monitoring Application

The main monitoring application acts as a hub, bringing together the OPC UA server, the component back-end and a multi-threaded gatherer-publisher front-end routine which manages the monitoring chain. The monitoring application starts upon boot as a daemon process using systemd. Within, the gatherer and publisher threads are started with shared memory space.



**Figure 6.30.** A schematic diagram illustrating the machinery of the TREX Monitoring framework. Two threads run in parallel, one is tasked with gathering the environmental data by communicating with the hardware. The data is stored in temporary buffers, which both threads share access to. The second thread reads the data from the buffer and prepares it for the consumption of an OPC UA client.

The Gatherer thread is launched first and initialises the communication with the hardware and instantiates all object component drivers. With the communication established with the on-board devices, the monitoring data is collected, packed and stored in a buffer (FIFO) for a given interval. The frequency of the data collection is dynamically changeable but for the purposes of collecting the data for the ATLAS DCS, it is set to every 2s.

The second thread is the Publisher which starts and initialises the OPC UA server, reads and sorts the datapoint elements based on the XML file. The thread polls on the status of the shared buffer containing the monitoring information and when there is no empty status, reads the data from the buffer. The data is unpacked and the datapoint elements are filled with the corresponding information. The frequency of reading from the buffer and updating the datapoint elements can be dynamically tuned, however, it is tied to the fetching frequency of the Gatherer thread. The Publisher thread can receive client commands from the DCS and passes them to the appropriate control segment. Such commands are for power control but can be used for other signalling purposes.

With each update to the value of the datapoint elements, a timestamp is made and attached along with the data that will be served to the client. The client, in this case, is the OPC UA client within the ATLAS DCS WinCC OA framework. However, any OPC UA-based client can connect and access the data. During the commissioning phase of the TREX and in the laboratory test benches an open-source OPC UA client is used to ensure the healthy operation of the modules.

Fig. 6.30 depicts the two threads which make up the monitoring framework. The Gatherer thread is shown on the left, with a data-flow direction from bottom to top. Data is extracted from the hardware, packed and placed in the buffer shown on top. The Publisher thread is depicted on the right with the interface to the ATLAS DCS.

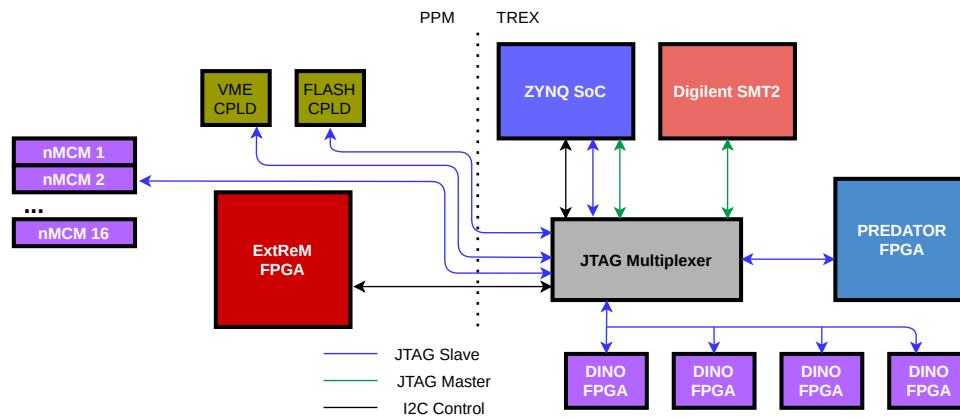
### The JTAG chain and the Zynq

Two onboard devices can act as the master of the JTAG chain. The Digilent SMT2 FTDI device [73] gives access to the bus via USB, while the Zynq implementation is network-based and the communication is realised via ethernet. This gives the Zynq the advantage of remote accessibility and is a crucial feature to have when there is no physical access to the modules.

Fig. 6.31 illustrates the JTAG bus for the PPM and the TREX. The JTAG bus consists of a multiplexer, dividing it into 6 separate chains. The multiplexer is connected to one of the IO expander devices which are controlled through the I2C bus. Either via VME or through the Zynq, the 2 ends of the chain can be selected. It does not only allow access to the FPGAs hosted on the TREX module, but also to the CPLDs and nMCMs via the VME backplane.

Since all four JTAG signals are directly connected to the Zynq PS portion of the device, they are declared as Memory-Mapped Input and Output (MMIO) lanes with direct memory-mapped access. A custom server application has been written which runs on the Zynq and receives the JTAG commands from the Xilinx Vivado framework [74] through a web socket. It translates and forwards the commands via a bit-banged JTAG protocol to the destination device. The signals returning from the destination device are transmitted through the web socket back to the Vivado framework in a similar fashion. To increase the performance of the JTAG server, rather than using a GPIO driver for controlling the bit transfer, the GPIO pins are directly accessed through the Zynq register memory map.





**Figure 6.31.** The JTAG chain of the PPM and TREX modules. A programmable multiplexer allows selecting between the Zynq and the Digilent devices as the master, and between the various FPGA chains.

In ATLAS, the Zynqs are connected within the ATLAS Technical and Control Network (ATCN) and further isolated via dedicated interdomain-sets that allow traffic only through a few gateway nodes. The second layer of isolation is added due to the custom kernel version running on the Zynq PS.

The JTAG access is established through multiple Secure Shell (SSH) tunnels, complying with all of the required computing security rules put in place by ATLAS.



# Chapter 7

## Functional Tests

Throughout the TREX development cycle, a set of test procedures have been developed and exercised in order to cover all aspects of the module functionality. That includes finding faults in the hardware and also mistakes in the firmware development. Key issues were identified by the test routines which drove the development of the pre-production and production versions of the hardware. A sub-set of the performed tests are presented in this chapter.

### 7.1 Test software suite

The combined PPM and TREX software suite, called PrePROcessor TEst SoftWare (PPROTEST) V3, is a collection of C++, Python and TCL applications that manage the module control, FPGA programming and provide test routines for verifying the board functionality. As the TREX is entirely a digital system, almost every aspect of its functionality requires companion test and control software. Each FPGA firmware-build workflow includes automated generation of the register maps which then can be used by the software.

### 7.2 Acceptance tests

During the production phase, each TREX module underwent a common set of power and configuration routines to ensure that the FPGAs and onboard components are operational and functioning as expected. Fig. 7.1 illustrates the step-by-step actions taken during the initial acceptance tests.

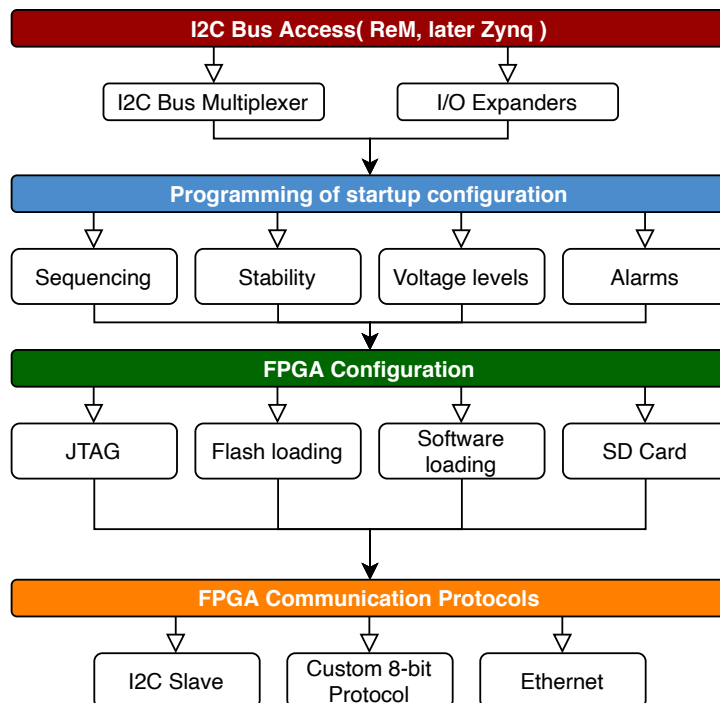
Upon the first cold-boot of a TREX module, only the 3.3 V I2C bus is powered up. The bus provides access to the power managers and the IO expander chips via the PPM-TREX I2C interface. At this stage, the ExtReM I2C bus master implementation, described in Sec. 6.3.3, is exercised for programming the onboard components. The

power sequencing scheme is loaded onto the power management unit which ramps up the voltages of the DC-DC converters and monitors its values. All voltage levels from the rectifiers are verified and in case of issues, alarms are tripped as digital flags to indicate faulty behaviour.

The next step in the validation chain is the FPGA configuration. The bitstream is transferred to the FPGAs via VME using the software loading mode, described in Sec. 6.3.4, and also via the JTAG interface. After the configuration is done, the QSPI flash memories can be accessed via the FPGA fabric and programmed over VME.

Once the FPGA firmware is loaded, a set of soak test applications are launched for the communication interfaces. The software is run for multiple hours, performing continuous read and write cycles using the available configuration protocols. An error counter is incremented when a transfer error is detected or when the data read-back does not match with the expected values. During the qualification period of the production batch, all modules passed the soak tests, reporting error-free communication.

The signal integrity verification of the electrical and optical paths is discussed at a later stage in the chapter.

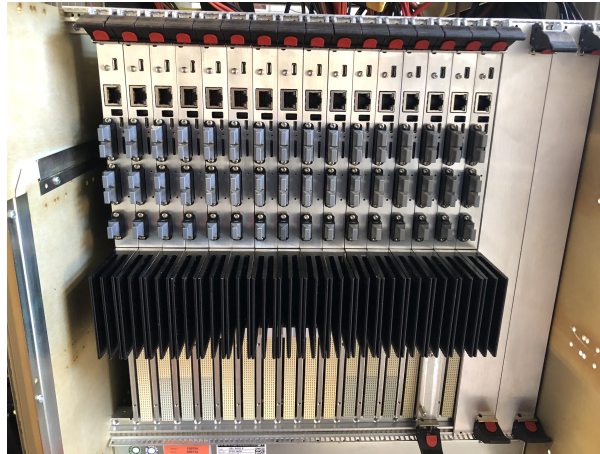


**Figure 7.1.** A flow chart diagram illustrating the acceptance test procedure for the production TREX modules after the visual hardware inspections and assembly.

### 7.3 Power and thermal measurements

Evaluating the power draw and thermal performance of the TREX modules is of key importance since they are powered by the same VME crate power supply as the PPMs. The available power budget is limited and the additional modules alter the overall cooling capabilities of the crate fan-trays.

To estimate the power consumption and the thermal performance of the individual boards, a setup consisting of a VME crate equipped fully with TREX modules is prepared and monitored in a laboratory environment. In this set of measurements, TREX modules in only four slots are actively powered up along with their respective PPMs. This constraint was due to the on-hand availability of spare and TREX-compatible PPMs. However, having all slots populated creates airflow restrictions similar to the final environment and allows for more accurate estimations.

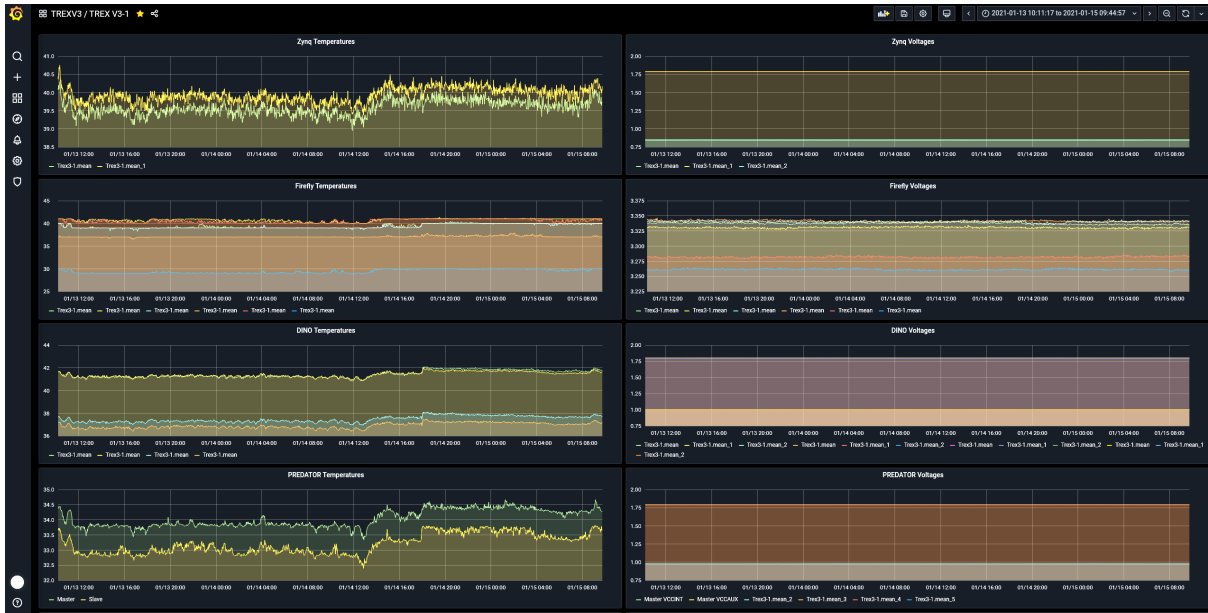


**Figure 7.2.** VME crate fully populated with TREX modules in the laboratory. The VME crate has undergone mechanical and electrical modifications for hosting the modules.

The FPGAs on the four active modules are configured with a full-load firmware that enables all 52 transceivers and utilises nearly all available resources, stressing the system to its maximum.

For collecting the environmental data, the monitoring application, described in Sec. 6.7.3, is launched on the Zynq SoCs. The application starts gathering the sensor parameters and provides them to an OPC UA client running on a local computing node. The data is retrieved every 30s, placed into a time-series database [75] and visualised by the Grafana platform [76].

Fig. 7.3 depicts the Grafana dashboard showcasing the TREX FPGA temperature-time evolution on the left and voltage values on the right column. The temperatures remained at a comfortable range of 40 – 50°C. The fluctuations in the temperature are directly correlated with the day-night cycle as the laboratory environment is not temperature controlled.



**Figure 7.3.** Grafana dashboard showing the time evolution of the TREN FPGA voltages and temperatures in the laboratory environment, monitored via the SoC.

The power consumption of the VME crate with four active PPM and TREN modules is measured to be **405 W**. Subtracting the contribution of the fan-tray and the SBC, the estimated power consumption for a PPM-TREN pair is **90W**. The results were encouraging as the values are within the available power budget of 170 W per slot and showcase stable temperatures with abundant cooling headroom.

## 7.4 Clock stability measurements

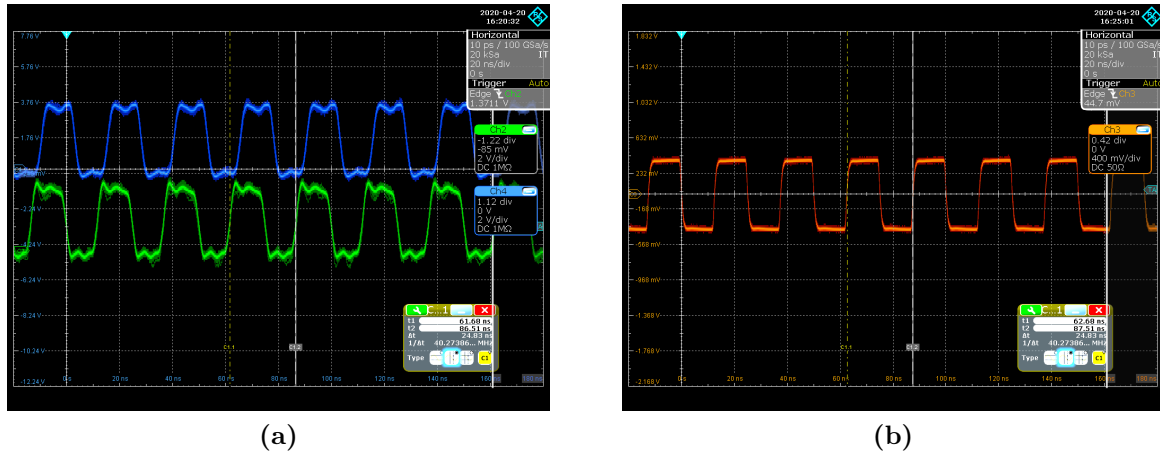
The 40.08 MHz LHC clock provided by the ExtReM FPGA on the PPM branches out directly to all FPGAs and to the Si5345 jitter attenuator PLL, as described in Sec. 6.4. Most of the PREDATOR logic blocks use the LHC clock or a clock derived from it through an internal Mixed-Mode Clock Manager (MMCM). The jitter attenuator PLL incorporates a feedback loop to correct for phase shifts and jitters which can be also measured with an oscilloscope using differential probes.

To verify the clock quality, the 40.08 MHz clock output from the PLL is used as an input for the PREDATOR FPGA. In the FPGA fabric, one branch of the clock tree is routed out of the FPGA to the onboard test pins, where the clock signal is sampled by an oscilloscope.

In Fig. 7.4 (a) the input and output of the LHC clock is depicted. The input, shown in green, is sampled at the connector between the PPM and the TREN, while the output, shown in blue, is the clock driven out of the PREDATOR FPGA to the onboard test pin. Both signals are single-ended, hence noise is added during the measurement, however,

the phase relation between the two is stable and does not introduce any shifts.

The jitter attenuator PLL output is directly measured via a differential probe. Fig. 7.4 (b) shows the clean output of the jitter attenuator. The clock pulses have sharp rising and falling edges with smooth flat tops.



**Figure 7.4.** The measurement of the input (green) and output (blue) 40.08 MHz clock quality of the PREDATOR FPGA (a). The output of the Si5345 PLL jitter attenuator (b), which provides the reference clocks for the MGTs, measured using a differential probe.

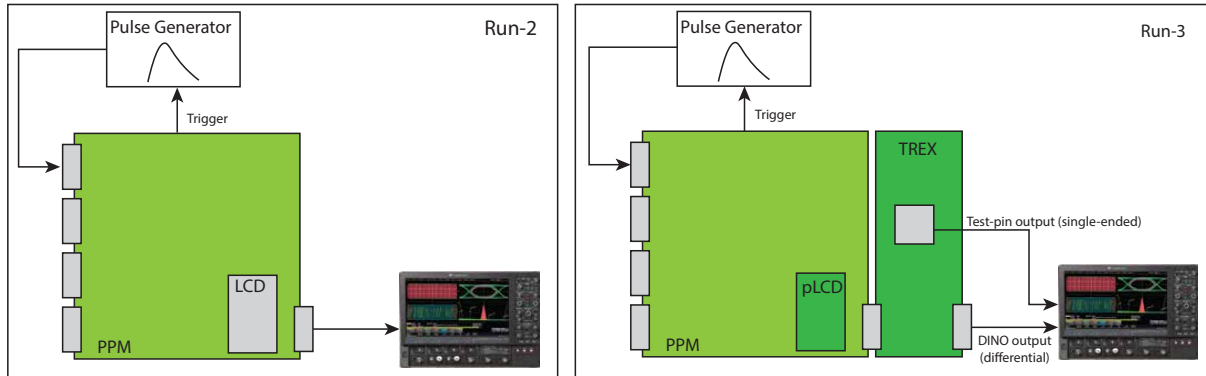
## 7.5 Latency measurements

The real-time path of the PPr is time sensitive and the latency of the system must be determined ahead of time to make sure it is kept as low as possible. From simulation, the overall latency of the system can be estimated, however, a clear and proper measurement is required to determine the real-world latency values. Multiple measurements have been developed at different points during the commissioning phase of the system to correctly determine how much time each part of the data processing takes.

The first leg of the latency measurement starts at the input of the PPM and ends at the output of the TREX module. It takes into account the digitisation of the calorimeter signals and processing time in the nMCMs. The propagation delay of the serial LVDS signals through the PPM is also included. With the replacement of the active LCD card by the pLCD, described in Sec. 6.2, the latency on the PPM is lowered by 5 ns.

On the TREX, this latency is consumed by the data duplication on the DINO FPGAs and the pre-emphasis circuit which drives the LVDS streams. Since there is additional physical routing that the signals have to travel, the TREX adds 2-3 ns of additional latency to the legacy trigger path. This small additional latency is acceptable and does not affect the trigger performance.

Fig. 7.5 visualises the testing setup in the laboratory environment and the methodology behind it. A comparison is shown between the PPM system in Run-2 and Run-3 with the TREX module.

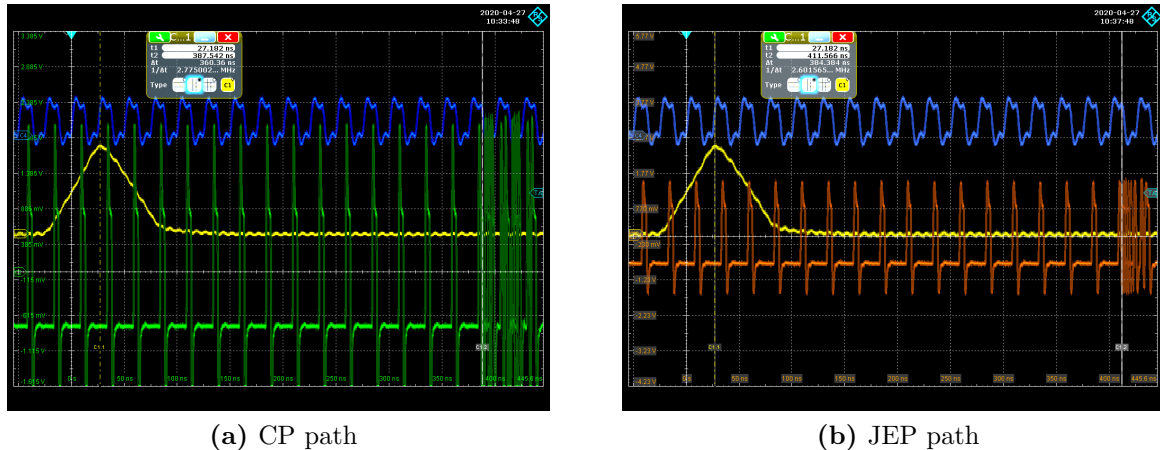


**Figure 7.5.** A schematic overview of the setup for measuring the latency at the TREX outputs for the legacy path. The left image shows the procedure for the Run-2 conditions, prior to the TREX. The right image illustrates the measurement where a PPM is equipped with a TREX module.

The setup consists of a function generator, programmed with a custom pulse shape that imitates an incoming signal from the calorimeter and a high-speed oscilloscope, which monitors the digital serial output stream at the back of the TREX. A programmable start signal, generated in the PPM, provides the trigger for the function generator. Prior to starting the measurement, the PPM is configured with appropriate FIR filters and the baseline is adjusted via the DAC in order for the signals to pass through the full processing chain on the nMCMs including the peak finding algorithm and the bunch-crossing identification logic, as mentioned in Sec. 4.1.2.

The measurement results are presented in Fig. 7.6. The generated pulse is shown in yellow, and the signal represented in blue shows the 40.08 MHz LHC clock. The serial LVDS stream for the CP path is shown on the left, depicted in green and on the right for the JEP path, in brown. The time interval between the peak of the analogue signal and the first bit of the non-zero LVDS data is the latency of the legacy path. It corresponds to **360 ns** for the CP path and **385 ns** for JEP. A difference of about 25 ns between the two is expected, as there is an additional summation step in the JEP path.





**Figure 7.6.** The latency measurement of the signal propagation time through the PPM and the TREX for the legacy trigger path. The yellow-coloured pulse is measured at the PPM input and the output LVDS stream is shown in green for CP (a) and brown for JEP(b). The 40.08 MHz LHC clock, shown in blue, is used as a reference.

With the latency for the legacy trigger path validated, the next set of measurements is dedicated to evaluating the latency through the PREDATOR FPGA and the optical-electrical FireFly transmitters.

Sampling of high-speed signals at the order of 10 GHz and above is very challenging and requires state-of-the-art equipment to do so. Rather than measuring the PREDATOR output directly, the latency measurement is done using logical flags to indicate the data position within the processing chain of the FPGA fabric. By taking the time difference of the logical signals the latency can be accurately determined without requiring special equipment.

The real-time path latency in the PREDATOR FPGA was initially extracted from simulation to be a maximum of **86.45 ns (3.5 BCs)**. However, measurements with real hardware are still necessary for the final result. The latency evaluation is split into two parts. The first part covers the signal travel time from the LVDS input stage up to the FEX Mapping & Fan-Out logic block shown in Fig. 6.19.

Using the same experimental setup, the generated pulse undergoes the data-duplication step in the DINO FPGAs and gets processed in the PREDATOR FPGA. Inside the PREDATOR FPGA, the data stream is monitored for non-zero values for every clock cycle at the FEX mapping stage. Upon receiving non-zero data after the start pulse, a dedicated signal is asserted and driven out to one of the available onboard test pins, where the oscilloscope is attached to. The oscilloscope captures the LVDS data from the TREX output, as done previously, but also receives the pulse, indicating that the non-zero real-time data has reached the mapping stage inside the PREDATOR. Using the time difference between the start of the LVDS data frame and the asserted signal, the latency is obtained to be **53.2 ns**, as shown in Fig. 6.19. The length of the traces routed

between the DINO and the PREDATOR is similar to the distance between the DINO FPGAs and the LVDS output connector where the oscilloscope is attached. Therefore, the travel time over the traces is negligible.

The measurement is shown in Fig. 7.7, where the LVDS streams are coloured in green and dark orange, while the asserted signal is depicted in blue. The latency is obtained by taking the time difference between the start of non-zero LVDS data (dark orange) and the blue signals. The results show good agreement with the values obtained from the simulation discussed in Sec. 6.5.5



**Figure 7.7.** The measurement results of the internal latency of the PREDATOR FPGA. As a starting reference, the LVDS stream is monitored for non-zero data. Once the incoming data is de-serialised and processed internally, a flag is raised (blue-coloured) to indicate the arrival of data at a specific logic block.

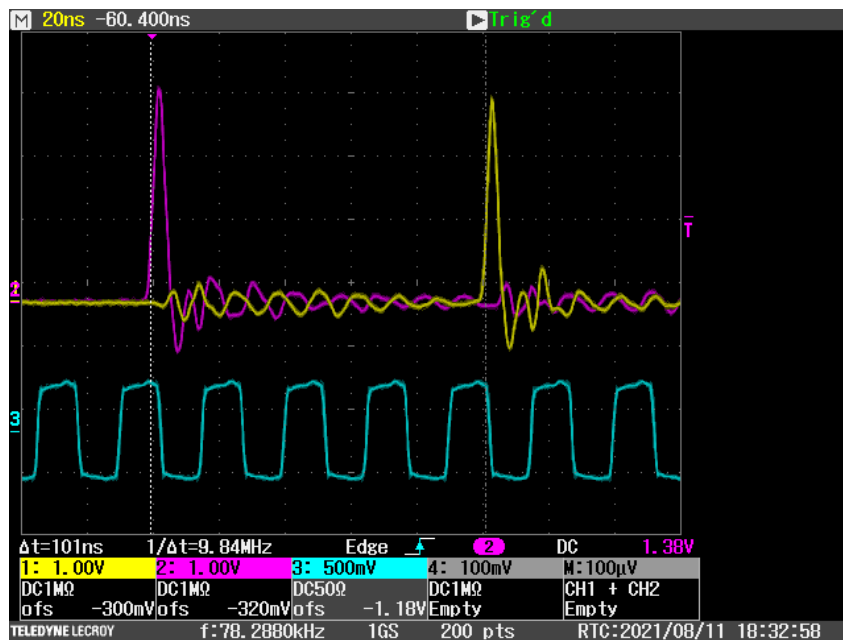
The last remaining measurement is the processing latency of the FEX formatting, 8b/10b encoding, high-speed serialisation steps and the latency of the FireFly transmitter module.

For this purpose, a special receiver-focused firmware is developed, which can receive the the high-speed data, de-serialise it and perform 8b/10b comma alignment and de-coding. The setup consists of a transmitter TREX, loaded with the main firmware and a receiver TREX mounted with a 12-way receiver module.

On the transmitter TREX side, two signals are driven out onto the onboard test pins.

The first signal indicates that the data reached the FEX formatting block, as done in the previous measurement, and the second start signal is asserted just before the K28.5 comma character is transmitted in the high-speed logic. Both signals are sampled on the oscilloscope. On the receiver end, the data is decoded and automatic comma detection and alignment is applied. Upon detecting the transmitted comma character, a stop signal is generated and routed to the test pin on the receiver TREX module.

The start and stop signals encompass the FPGA transmitter interface, the latency of the FireFly transmitter, the propagation of the light through 6m long fibres, the FireFly receiver and the receiver logic interface on the other TREX module. The latency of the FireFly modules is given by the manufacturer and the propagation time through the fibre can be estimated at 5 ns/m. The latency of the FPGA receiver logic is calculated for the unit interval  $UI = \frac{1}{11.2[GHz]}$  according to the specification provided by Xilinx [77].



**Figure 7.8.** The latency measurement of the FPGA logic and the high-speed transmission path of the TREX. The magenta-coloured signal is issued on the transmitter module acting as a start signal for the measurement. Upon capturing and decoding the data, a stop signal, coloured in yellow, is issued on the receiver module. The 40.08 MHz LHC clock (aqua) is used as a reference.

The result of the measurement is presented in Figure 7.8. The output flag driven out of the transmitter TREX is presented in magenta, while the comma detection flag on the receiver module is shown in yellow. The LHC clock is shown in blue as a reference. The obtained time difference between the two pulses includes the latency of the optoelectrical converters and the propagation over the fibre length and the receiver logic. After removing these contributions, the combined latency for the FEX Mapping & Fan-Out, FEX Data Formatting and the transmitter interface is determined to be **36.07 ns**.

The measured overall system latency results are presented in Table 7.1.

System	ns	BC
PPM	387	15.5
TREX Predator	89.27	3.5
<b>Overall</b>	<b>476.27</b>	<b>19.0 - 19.1</b>

**Table 7.1.** Latency estimations determined from physical measurements

A difference of 3 ns is observed between the measured value (**89.27 ns**) and the values obtained from the simulation (**86.45 ns**). However a total of **476.27 ns** for the full system is within the allocated latency budget of **480 ns** and no further optimisations are needed.

## 7.6 LVDS transmission measurements

With the removal of the LCD card on the PPM and the addition of the TREX module, the LVDS signal path carrying the trigger information to the CP and JEP processors is altered and requires validation. In order to qualify the data-duplication and the pre-emphasis circuit, the TREX output is connected to an oscilloscope that measures the bit transitions of the digital links.

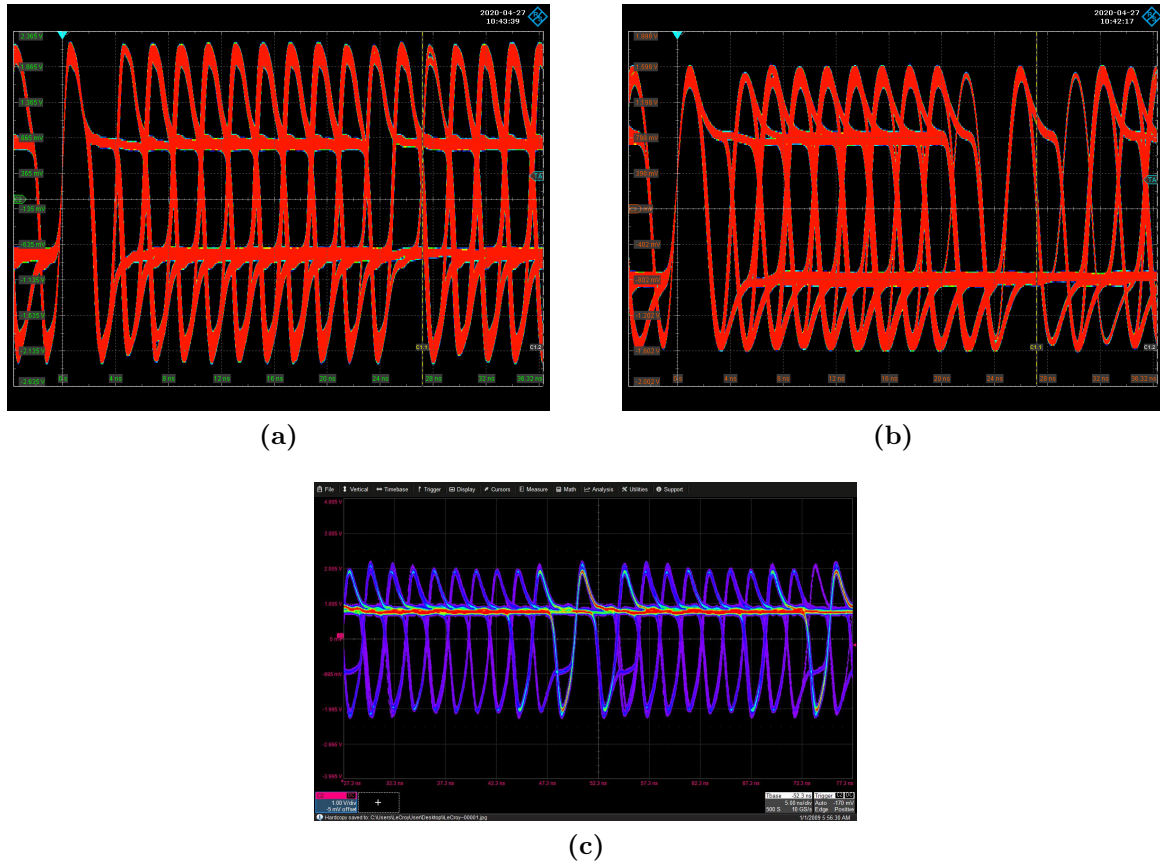
Fig. 7.9 shows the oscilloscope measurement for a link headed for the CP system on the left and to the JEP on the right. The images show the pre-emphasis and the 12-bit serialised data frames. The bottom measurement is set in persistence mode, which visualises the bit stream in a heat-map, integrated over a certain time interval.

The second method of validating the signal integrity is done through an automated test procedure using an assembly of 6U Common Mezzanine Cards<sup>1</sup> (CMCs) controllable via VME.

All 73 output LVDS signals are connected via 15m long cables to the CMCs. Data of various test patterns are loaded in the nMCM playback memories and transmitted via the LVDS links to the CMC. The received data is stored in memories onboard the CMC and accessed via the VMEbus. The data is read out via software and analysed for data-duplication, mapping, signal integrity and parity issues. Each production TREX module has been put through the test routine and qualified before installation.

Overall, excellent LVDS transmission quality has been demonstrated for all production TREX modules. During soak tests lasting multiple hours, the transmission remained stable and error-free for the output LVDS streams to the CP and JEP systems and also for the copies to the PREDATOR FPGA.

<sup>1</sup>Custom FPGA boards which receive the LVDS signals and transmit the data contents via VME to the user software



**Figure 7.9.** Oscilloscope measurements of the LVDS signals at the TREX output, designated for the CP (a) and JEP (b) systems. (c) depicts the measurement in the persistence mode through a heat-map of the transmission over an integrated period of time.

## 7.7 Signal integrity of the high-speed transmission

Stable and error-free transmission on the optical real-time data path forms an integral part of the TREX functionality. To qualify the transmission stability of the TREX optical outputs, bit-error ratio (BER) tests are performed to demonstrate the transceiver performance for all production TREX modules.

The testing methodology consists of a transmitter TREX that is under-test and four TREX receiver modules, each equipped with one 12-way FireFly module. All 48 optical links, dedicated for the real-time data are connected to the four 12-way receiver modules, such that all transmitters of a single TREX can be studied in parallel. A specialised flavour of the firmware is used on all processing FPGAs, which includes an Integrated Bit-Error Ratio Test (IBERT). The transmitting TREX generates a Pseudo-Random Binare Sequence (PRBS-31) data pattern that is differentially driven out of the FPGA and routed to the opto-electrical FireFly transmitters. The light travels via 6m long optical cables and is converted back to electrical on the receiver TREX modules. Inside the processing FPGA on the receiver end, the data pattern is analysed bit-wise and in



case of non-matching results, a bit error counter is incremented.

The test procedure is left running until a satisfying BER value is reached. The definition of the BER is described below:

$$BER = \begin{cases} \frac{N_{error}^{bits}}{N_{transf}^{bits}}, & N_{error}^{bits} > 0 \\ \frac{1}{N_{transf}^{bits}}, & N_{error}^{bits} = 0 \end{cases} \quad (7.1)$$

where  $N_{transf}^{bits}$  is the total number of bits transferred and  $N_{error}^{bits}$  is the total number of identified bit errors. The BER is the ratio between the number of non-matching bits and the number of bits transferred over the high-speed path. All optical links from the production batch have qualified and achieved the industry standard BER value of  $10^{-15}$ .

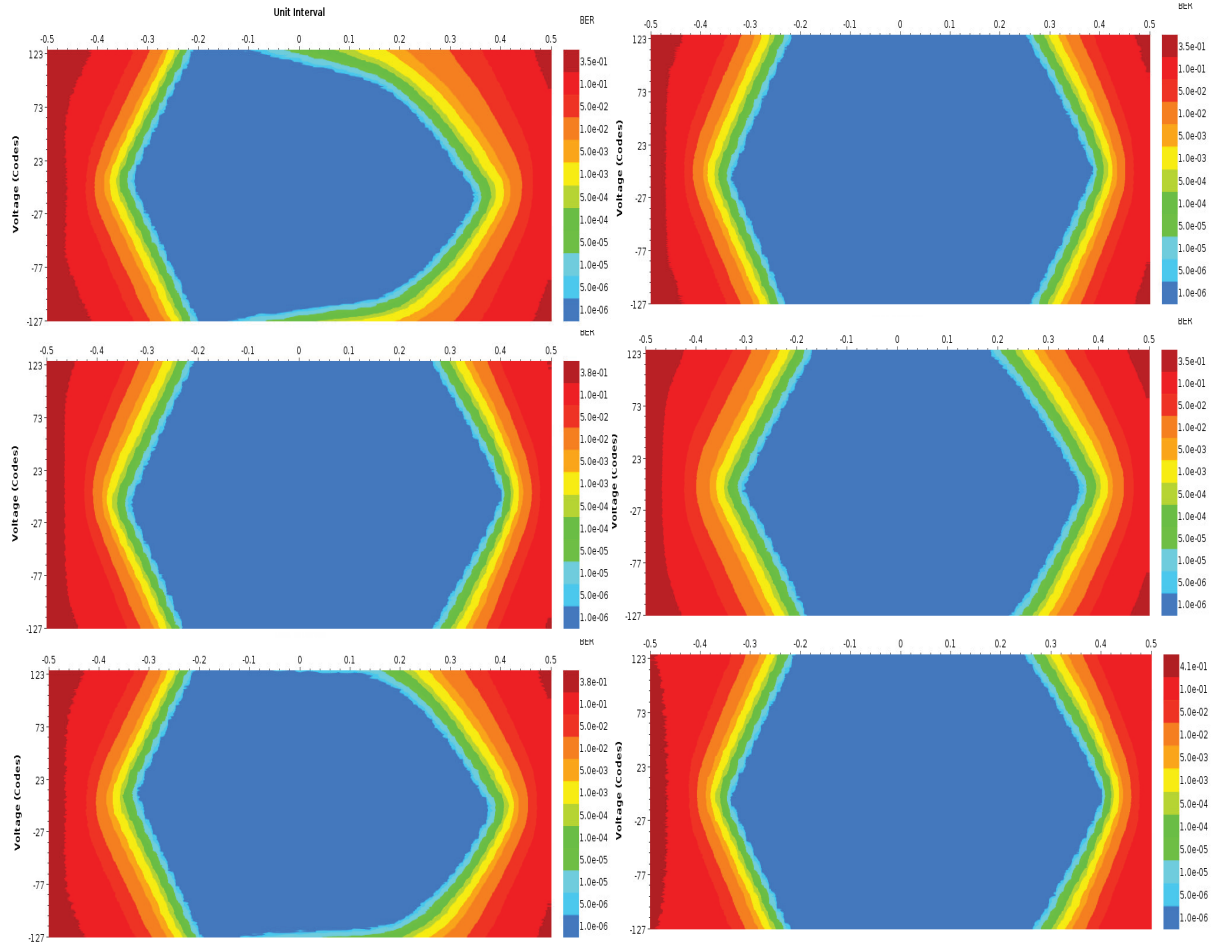
In parallel to the BER measurements, the individual optical links are verified through eye diagrams. The receiver FPGA repetitively samples the differential signals from the FireFly modules to create a heatmap showcasing the digital bit transitions for the high and low levels of the signal. It can be interpreted as a probability density function of the signal. The naming of the diagram is based on the shape of the bit transitions, which resembles that of the human eye. It is used to evaluate and expose the effects of noise, and interference over the entire transmission chain.



**Figure 7.10.** The test setup for measuring the optical link quality of the TREX modules. The leftmost module is the transmitter device under-test. It is connected with four fibre ribbons to the four modules on the right, which act as receivers.

Fig. 7.11 shows a set of eye-scan diagram results from the testing procedure. The vertical axis is the signal amplitude in units of voltage codes, typically corresponding to

$\sim 1.5$  mV per unit code value. The horizontal axis depicts time in unit intervals and depends on the transmission speeds, for the TREX, it is  $UI = \frac{1}{11.2[GHz]}$ . The colour palette indicates the probability of the signal having a possible voltage value within the unit interval. The open area of the eye pattern acts as a measure of the quality of the signal propagation. The larger the opening, the better the recovered amplitude and the rise and fall times.



**Figure 7.11.** A set of eye-scan diagrams illustrating the transmission quality of the optical links on the TREX. The data is captured on a TREX module with a receiver interface. The horizontal axis represents time in Unit Intervals for a link speed of 11.2 Gbps ( $UI = 1/11.2[GHz]$ ). The vertical axis represents the signal amplitude in voltage codes.

The measurement is similar to the oscilloscope measurement in persistence mode shown in Fig. 7.9 (c), but the focus is on a single-bit transition. Individual eye scans are taken for all 1664 optical links and each link is successfully characterised. The channel-by-channel variations on the receiver modules have been normalised and corrected for in each measurement. During the testing period, only two FireFly channels were identified as faulty and replaced. Overall, the high-speed transmission of the system of TREX modules is validated and showed excellent performance.





# Chapter 8

## Installation, Commissioning and First Data

The installation of the TREX modules took place in May 2021 and was completed within two weeks. The two L1Calo Tile PPr crates, located in the underground electronics cavern (USA15) of the ATLAS detector, were modified and equipped with 16 TREX modules each. The nature of the installation work was unique, as it required dismantling the two VME crates of the existing PPr system in order to add mechanical and electrical support to host the TREX modules. Once the installation came to a close, the integration into ATLAS began: first with standalone operation, then into combined running alongside the full detector. The TREX officially became part of standard operations in ATLAS after demonstrating stable and error-free data transmission on the legacy real-time path and achieving more than 100 kHz rate of the readout via the legacy ROD path.

This chapter describes the installation and integration process including consistency comparisons using collision data at the start of Run-3.

### 8.1 Installation in ATLAS

While the production modules were going through acceptance and stress testing, the infrastructure needed for the installation was planned and prepared in parallel.

The preparations started multiple months in advance, as the VME crates required new mechanical constructions to hold the TREX modules in place. Custom CNC-machined rails were placed in the back area of the crate on both the top and the bottom. The spacing of the slot indentations is done with high precision to align the modules with the backplane connectors. Multiple support bars were added to hold the added weight of the new modules. The new mechanical structures and backplane connectors are shown in Fig. 8.1; (a) presents the back view of the crate mid-installation of the TREX modules and (b) visualises the front view with the PPMs re-inserted.

The backplane of the crate is the second modification required. A new row of Compact PCI connectors was added which connects the LVDS signals travelling from the PPM to the TREX. The connectors are modular and grouped into four. In case of issues with one of the modules, the whole backplane does not need to be removed, but rather a single block of four connectors can be exchanged. The power distribution on the backplane was also changed by adding new adapter cards that supply the TREX modules with 3.3 V and 5 V of power directly from the VME crate power supply. The power adapter cards are hand-soldered and provide power to four TREX modules.

One of the most challenging aspects of the installation is the handling of the bulk LVDS cables that are attached to the back. During the dismantling, the cables were removed from the back of the PPM and re-attached to the back of the TREX. The cables are grouped into blocks of four and a specialised extractor tool was used to carefully remove the cable block without damaging the delicate pins. The insertion of the cable block is equally a delicate procedure, where a constant, uniform force is applied to make sure the pins make contact and sit flush with the mechanical structure.

Fig. 8.2 (a) shows the back of the VME crate, fully equipped with TREX modules and with two of the four cable blocks attached. The cable bundles are hoisted up to ease the mechanical tension on the connectors.



**Figure 8.1.** The back view (a) of the VME crate with TREX modules inserted. The frontal view (b) with PPMs re-inserted after the hardware adaptations.

With the additional mass of the TREX modules, bundles of optical fibres and ethernet cables, the available space inside the rack is reduced, hence affecting the airflow and cooling performance. Routing the optical fibres and the ethernet cables around the rack walls required optimisations in order to minimise the impact of the airflow restrictions. Fig. 8.2 (b) shows the modules with full connectivity in place, such as the optical fibres and ethernet cabling for the Zynq SoCs.

The required PPM modifications were done in-situ during the installation campaign,

such as the LCD card replacement and the necessary hardware patches.



**Figure 8.2.** Bundles of LVDS cable insertion after the TREX installation (a), which connect to the CP and JEP systems. The optical fibre connection to the FEX and FELIX systems (b).

### 8.1.1 Power consumption and thermal performance

With the installation complete, one of the first items to address was the hardware status and its operating conditions. While powering up the crates, one PPM showed power regulation issues and was replaced by a spare. The faulty PPM was diagnosed and repaired in the laboratory. For the TREX, all 32 newly installed modules power up correctly and the FPGAs can be configured via the crate controller SBC.

#### Power consumption

Measuring the power consumption of the two crates allows for estimating the exerted stress on the power supply and its longevity. To have the most realistic measurement, a stress pattern was loaded into the playback memories on the nMCMs, which creates a data flow through the entire system. Particularly, the power consumption increases when the LVDS streams have to drive non-zero data. Under these conditions, the power measurement was done via the VME crate power supply and the results are shown in Table 8.1. Both 3.3 V and 5 V rails of the VME power supply have a current budget of 345 A.

PPr Crate	Current (at 3.3 V)	Current (at 5.0 V)	Total Power
PP5	128.01 A	156.58 A	1205.33 W
PP6 (incl. TREX)	140.01 A	206.00 A	1492.03 W
PP7 (incl. TREX)	140.72 A	207.92 A	1503.97 W

**Table 8.1.** Typical current and power draw from the 3.3 V and 5 V sources during running in P1

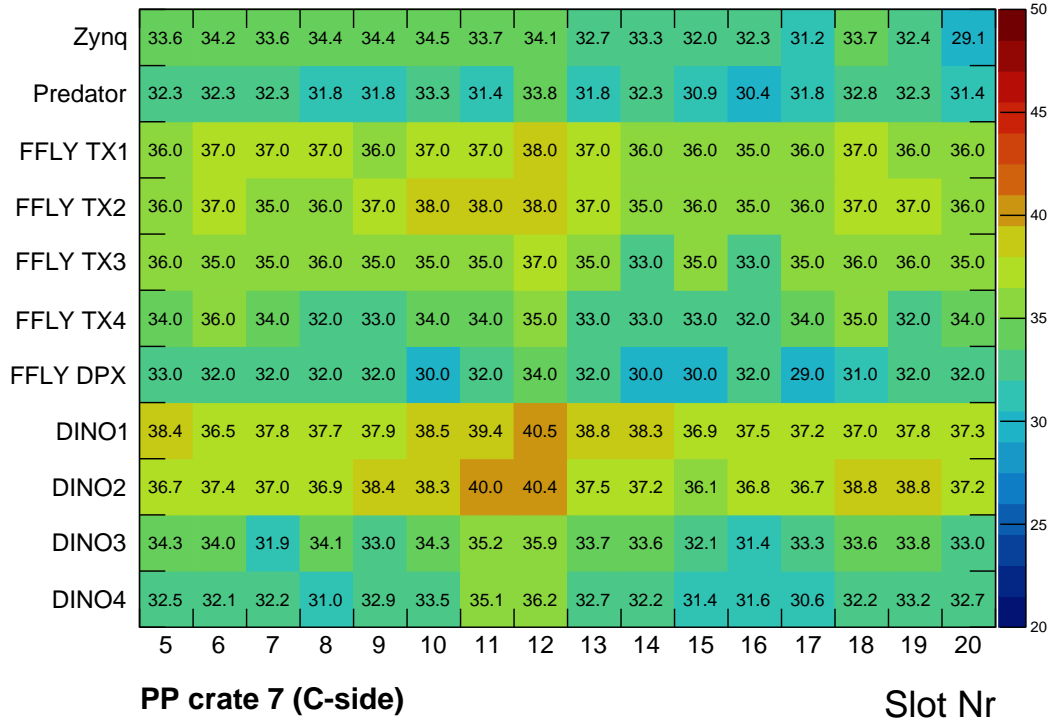
Table 8.1 presents the current draw values for both TREX crates and a non-TREX crate as a reference, the correctness of the reported values is verified also during collisions. A crate equipped only with PPMs consumes during data-taking a total of around 1205 W, while the TREX-equipped VME crates consume a total of around 1503 W. The total power consumption is in agreement with the values obtained in the laboratory environment, with a PPM-TREX pair consuming around 94 W of power.

### Thermal performance

Obtaining an accurate overall thermal status of the full system was not feasible in the laboratory environment, only the worst case was predictable by stressing the individual modules. With the modules running under load during the data-taking period, a snapshot of the temperature profile is made for both crates using the monitoring application running on the Zynq.

Fig. 8.3 shows the temperatures of the processing elements in each slot for a single VME crate. The profiles obtained for both crates show symmetrical behaviour. A slot dependence can be observed, with slots 11 and 12 having the highest temperatures. This is due to the distribution of the fans in the bottom tray of the crate. The fan placement creates a slight dead zone in-between the fans, where the airflow is much lower, causing elevated temperatures.

The highest temperatures lie in the region of 40°C, observed for the upper two DINO FPGAs, a value well within the specifications for operation. The processing FPGAs run at lower temperatures thanks to the protruding heatsink design for optimal heat dissipation. Both TREX crates demonstrate excellent thermal performance with ample headroom and stable operating temperatures. This result was achieved through stress-testing the modules and optimising the heatsink solution during the development phase.



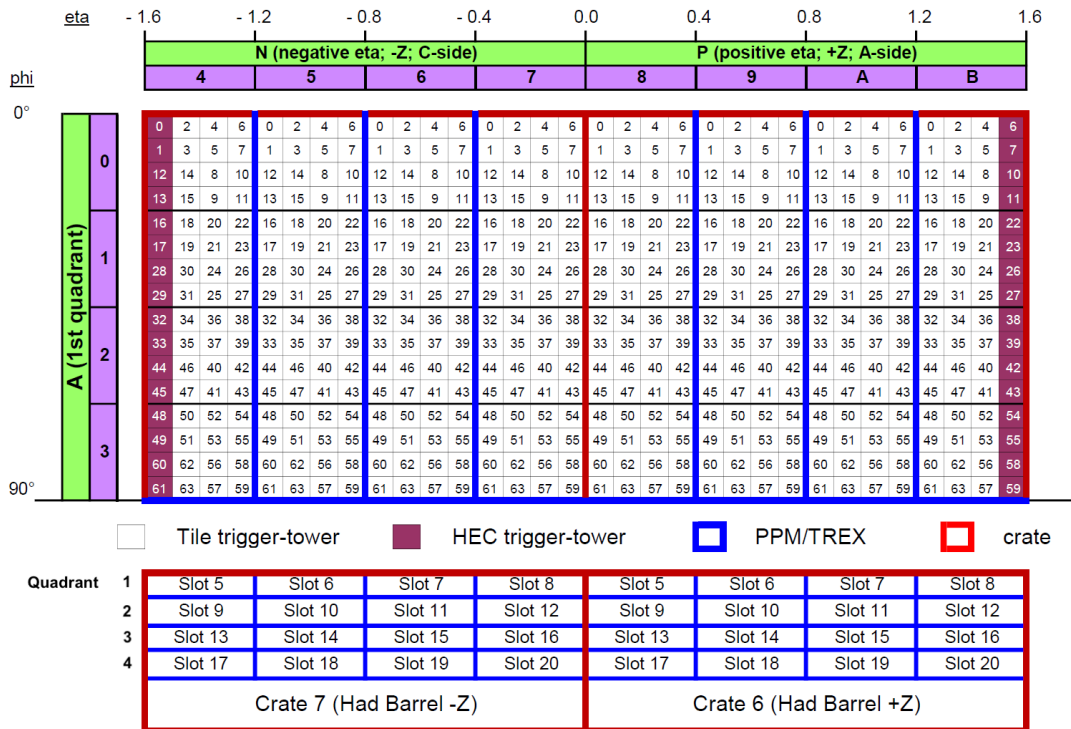
**Figure 8.3.** Temperature profile for the major TREX components in each slot within one VME crate. The environmental data is captured during a typical data-taking period, where all inputs and outputs are under continuous load.

## 8.2 Connectivity and Fibre Mapping

The optical outputs from the 12-way FireFly transmitters interface with the FOX optical plant through three Multi-fibre Termination Push-On (MTP) connectors. Each connector is dedicated to one FEX subsystem. The FOX receives the TREX inputs and distributes the fibres to the corresponding FEX modules. Based on the FOX inputs, the TREX firmware implements a channel mapping scheme that delivers the data to the expected destination modules.

Fig. 8.4 illustrates the complete coverage of the Tile Calorimeter by the PreProcessor system in Run-3, in terms of crates, modules and trigger channels. Each VME crate covers one side of the detector and each PPM module covers 64 trigger-towers from a  $\Delta\eta \times \Delta\phi = 0.4 \times 1.6$  region. The number attached to each trigger-tower indicates the corresponding PPM digital channel number.





**Figure 8.4.** The mapping of the PPM-TREX in the  $\eta - \phi$  space. Each crate processes half of the  $\eta$  coverage. Only one quadrant is depicted with the same mapping repeating for the other three quadrants.

## 8.2.1 Partitioning and mapping to the FEX processors

The TREX sends on each fibre link to the eFEX 16 trigger-tower  $E_T$  results from a  $0.4 \times 0.4$  ( $\eta \times \phi$ ) region. Depending on the calorimeter region, the TREX provides the eFEX subsystem with one, two or four copies of the data, due to the processing overlap between the eFEX modules.

As for the jFEX, the TREX sends on each fibre link 16 trigger-tower  $E_T$  results from a  $0.4 \times 0.4$  ( $\eta \times \phi$ ) region. However, the number of data copies required by jFEX is constantly three in all regions.

The gFEX requires  $0.2 \times 0.2$  ( $\eta \times \phi$ ) jet-sum data in only one copy. Since only 16 jet-sums are computed on each PPM, these can be all transferred to the gFEX via only one fibre link. Additionally, since the gFEX system consists of only one module that processes data from both the LAr and the Tile calorimeters, a second link to gFEX is allocated on each TREX, to serve as a spare connection.

The mapping of all FEX processors is validated with the use of the align frames sent by the TREX at a specified BCID within the LHC orbit. The align frame content includes the PPr crate and PPM-TREX slot number along with other metadata. The crate, module and nMCM numbers embedded into the frame translate directly to a region in the  $\eta - \phi$  space. This allows for end-to-end tracking of the fibre path from the source,

through the FOX to the destination modules. During dedicated mapping tests, swaps on the C-side of the coverage were discovered; this was mitigated by swapping the fibres on the TREX side. An example of the decoded align frame by the eFEX module is shown in Fig. 8.5.

00000152	da001cbc	←	TREX and FEX ID
00000153	3210740	←	nMCM and crate Number
00000154	beefcafe	←	Align frame payload indicator
00000155	0		
00000156	0		
00000157	0		
00000158	2c800dac	←	BCID and CRC value

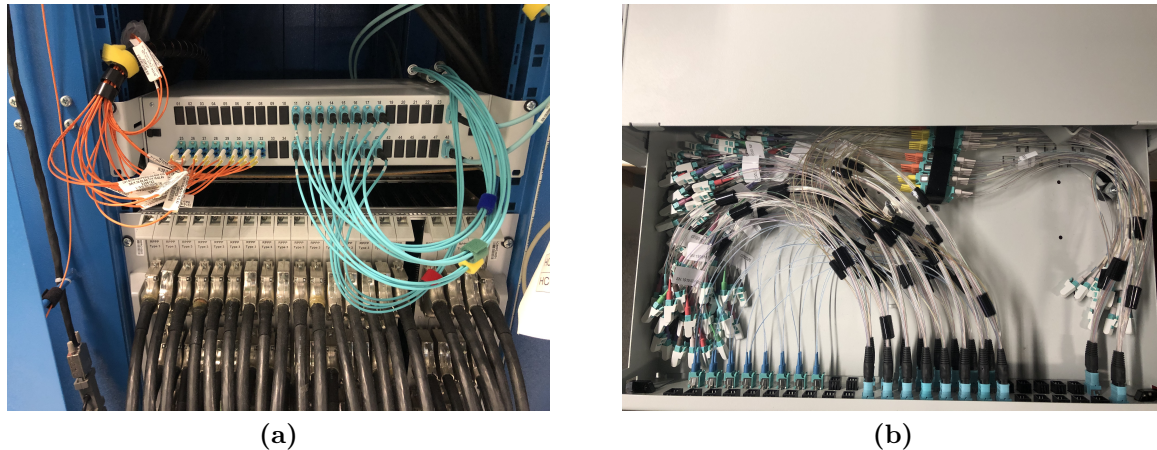
**Figure 8.5.** The TREX align frame sent on BCID 3500 and captured in eFEX memories. The payload includes the origin device identifier and the destination FEX type. In addition, further geographical information is added and a payload indicator makes finding the frame more human-friendly.

## 8.2.2 Readout fibre mapping

For connecting the TREX modules to both legacy ROD and FELIX interfaces, the fibres on the 4-way bi-directional FireFly transceiver need to be branched out and mapped to the corresponding systems.

To accomplish the mapping, two fibre exchange boxes are constructed, each servicing one PPr crate. A single fibre exchange box is equipped with 16 12-way MTP connectors that make the connection to the TREX modules, 16 single LC outputs to the legacy ROD modules, and two 24-way MTPs to the FELIX card. Internally, the mapping is done manually through MTP-to-LC break-out fibres. Each TREX ribbon maps one receiver and one transmitter fibre to the FELIX ribbon and a single transmitter fibre to the LC fibres going to the legacy ROD.

Fig. 8.6 (a) depicts the readout fibre exchange box situated in the rack neighbouring the PPr crate. The orange fibres leading to the legacy ROD are re-used and connected to the LC outputs. In the middle two rows of eight aqua-coloured MTP adapters connect the TREX with the fibre box. The rightmost column (aqua-coloured) is dedicated to the FELIX connection. Fig. 8.6 (b) presents the internal construction, the breakout fibres and the adapters used to perform the mapping.

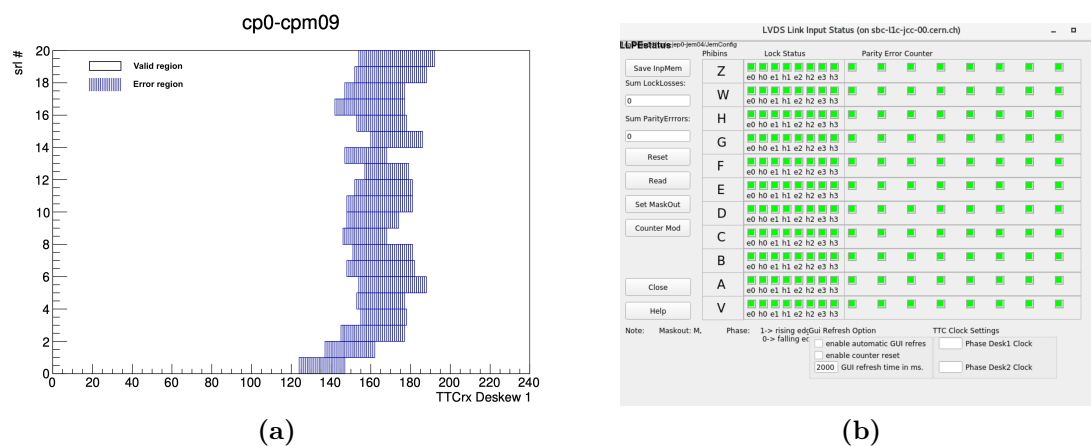


**Figure 8.6.** Frontal (a) and interior (b) view of the readout fibre exchange box showcasing the splitting between the legacy ROD fibres (orange) and the FELIX fibres (aqua).

### 8.3 Integration with the legacy trigger systems

The first priority in the integration is the verification of the legacy trigger path and the ability to provide the calorimeter transverse energies to the CP and JEP subsystems.

The TREX modules introduce a 2-3 ns delay to the LVDS data stream to the CP and JEP, therefore a re-validation is needed for the input timing parameters used for the downstream modules. A calibration routine is launched, that deskews the clock with respect to the incoming data stream and performs an alignment, such that the data bits are not sampled in a metastable region during a bit transition.



**Figure 8.7.** The output result of an input timing scan for the links of a single CP module. The deskew clock is shifted to align the incoming serial data with the de-serialisation clock.

The region in white indicates valid deskew values, while the blue region indicates de-serialisation errors (a). A graphical panel monitoring of the parity error counters of the input signals for a single JEP module (b).

The results of the input timing scan are presented in Fig. 8.7 (a) The CPM input timing scan shows the corresponding shift in the TTCrx Deskew1 clock [53]. The regions

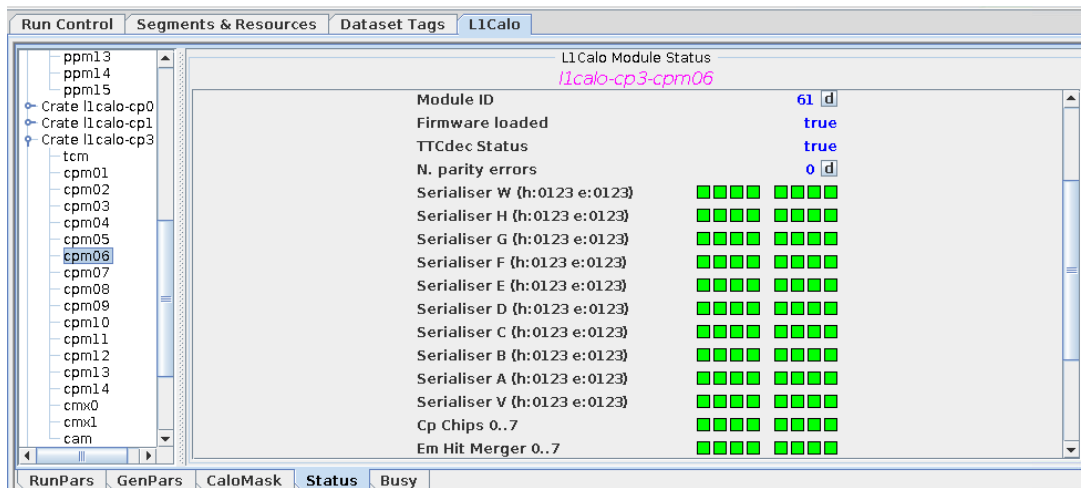


in white indicate error-free deserialisation and are valid deskew values. The blue-coloured regions show deskew values that cause de-serialisation errors.

However, as the timing window is large enough to accommodate the extra delay, the previously set values can be reused. With the timing in place, the CPM serialisers are able to lock onto all the LVDS inputs with no parity issues reported. The status of the serialiser links is shown in Fig. 8.8.

Validating the LVDS inputs for the data transmission to the JEP requires a manual approach. For this, the PPM playback memories are loaded with a special ramp pattern, designed specifically for verifying the inputs to the JEMs. On the JEM side, the LVDS inputs are monitored using a software tool for each module.

The LVDS input status for one of the JEP modules is shown in Fig. 8.7 (b), where an active parity counter is incremented, in case of a parity mismatch. After careful observation of the counters for all JEP modules with TREX inputs, the counters remain error-free.



**Figure 8.8.** The ATLAS TDAQ Run Control interface depicting the status of the serialisers on the CP modules with TREX inputs. The links are monitored every few seconds and report back in case of parity errors.

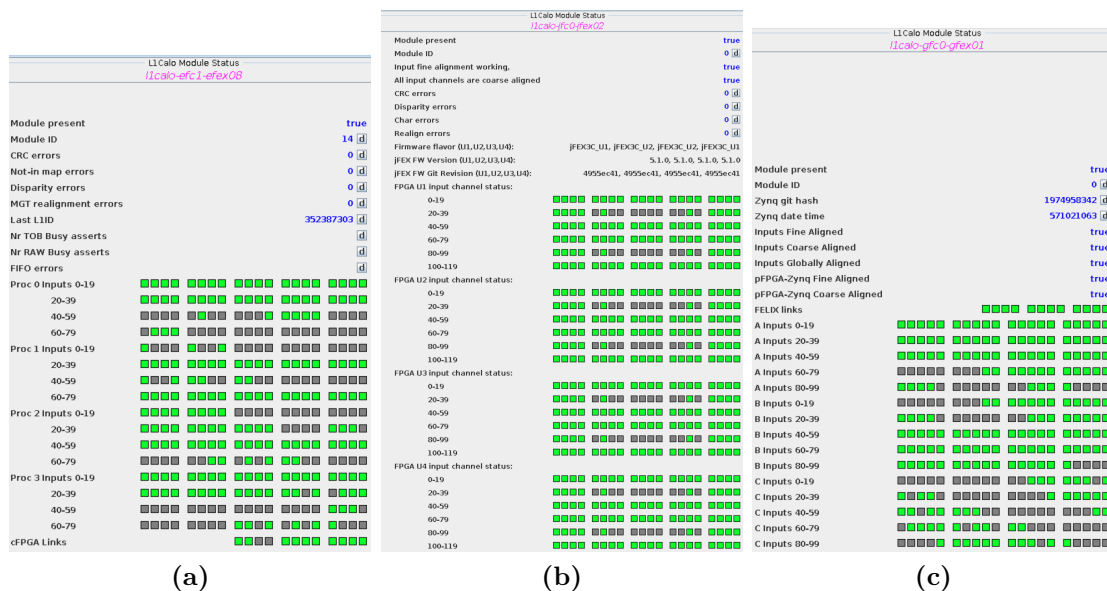
With long overnight data-taking runs and multiple months of stable running, the TREX demonstrates error-free data transmission to the trigger processors, achieving full and successful integration within the legacy trigger system.

## 8.4 Integration with the FEX systems

As the eFEX, jFEX and gFEX subsystems became available in ATLAS, integration into the upgraded trigger system could continue.

First, the optical links of the real-time path required validation. Implemented in each of the FEX modules is a receiver interface, where the incoming data is decoded and aligned. CRC calculation is also performed for the 40 MHz data frames and a status counter reports the results in the run control software.

The links connected up to the FEX systems are able to lock and recover the 280.56 MHz clock from the incoming data stream. The 8b/10b decoding is applied with a comma alignment to extract the TREX data frames. All links show stable performance, indicating that the transmitter logic implementation of the TREX is functioning correctly.



**Figure 8.9.** The ATLAS TDAQ Run Control interface indicating the optical input link status for single eFEX (a), jFEX (b) and gFEX (c) modules. The grey-coloured links are reused and disabled. In case of transmission issues, the link error status is propagated through colour changes and notifications.

Fig. 8.9 shows the optical input link status for single modules of the eFEX, jFEX and gFEX respectively during a data-taking run. The green colour indicates a healthy and error-free transmission status. In case of errors, the status colour changes and notifies the user to perform a reset procedure. The links indicated in grey are spare and unused channels, which remain disabled.

After basic validation of the data transmission, the fibre mapping required verification. The complex fibre network system going through the FOX and trunk cables with various polarity types can be verified by the use of the align-frames, described above. The analysis of such align-frame information led to discovering multiple swaps in the fibre trunk cables,

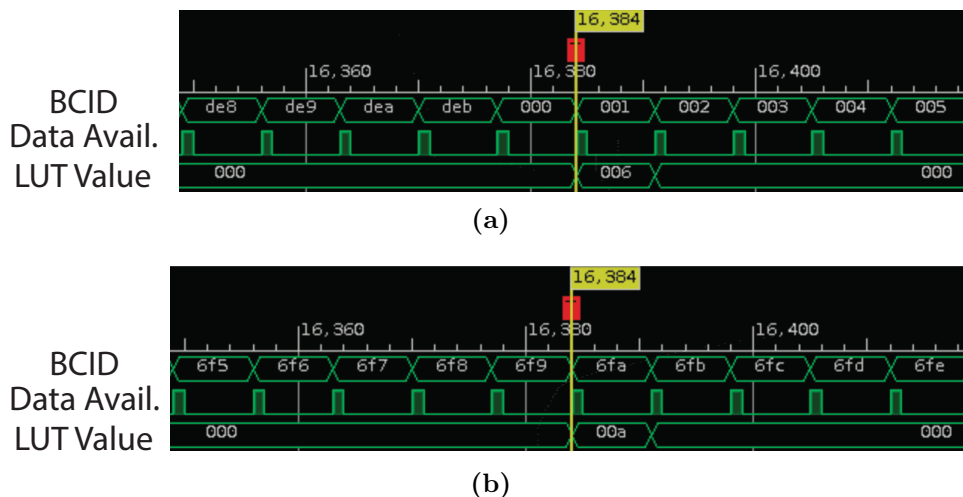
where the incoming data was expected from a neighbouring module. These were easily solved by changing the connection ordering on the TREX-end and did not need repairs in the FOX.

With the mapping verified, the next stage of validation is the analysis of the data contents transmitted to the FEX systems.

### BCID Assignment

For the FEX systems to align the energy deposits from LAr and Tile within the same Bunch-Crossing (BC), each FEX data frame is tagged with the Bunch-Crossing Identifier (BCID). This BCID must be calibrated, such that the energy deposits are associated with the correct BC at which the collision took place.

The initial calibration is done using the Tile Laser system, which creates detector activity at a specified BCID within the orbit. By sampling the data content for select links of the real-time data path on the processing FPGA of the TREX, the energy contents originating from the laser light are captured. As the BCID at which the laser light fires is fixed and deterministic, the FEX data frame BCID can be calibrated. An offset of 55 BCs was needed to align the BCID with the data. This offset is directly coupled to the latency of the trigger system and to the incoming Bunch-Counter Reset (BCR) signal, which is used as a reference. The BCR signal is issued at the beginning of every LHC turn to clear the BCID counters in all of the sub-detectors. To validate the offset value, collisions are needed. During the first 13.6 TeV p-p collisions, there were two isolated BCID values for which the beam did collide: 1 and 1786.



**Figure 8.10.** Data seen in a single data channel (noted as LUT Value) and its corresponding assigned BCID value, captured through a logic analyser. Two isolated collisions are present, at BCID 1 (a) and 1786 (b). The results show correct assignment between the LUT and the BCID values.

Fig. 8.10 depicts the results captured by a logic analyser of one data channel of the

real-time path and the BCID value transmitted with it. The data contents are zero in all neighbouring BCs, while non-zero  $E_T$  results are present in the expected BC. The results confirm that the TREX real-time path is properly aligned and provides correct BCID and energy values to the FEX processors.

## Input data comparison

The final validation stage before data-taking is the comparison between the input data received on the FEX modules and the transverse energy values transmitted by the TREX. To perform the verification, the nMCMs are loaded with a playback pattern that generates non-zero  $E_T$  results for every other BC. The data gets streamed to the DAQ system and written into a raw data file for offline analysis.

Fig. 8.11 depicts the event data packet, where the eFEX input data is compared with the TREX output data, read out via the FELIX-SWROD path. The values presented in the eFEX packet use hexadecimal notation, while the TREX values use decimal. The results are in agreement as the value 0x17 in the eFEX packet corresponds to 23.

```

eFEX input
readout
0000002f: [Tile] Errors=0, FPGA=0, Chan=47
81705cbc: KChr= bc, Ch00= 17, Ch01= 17, BC(6:5)=0, BC(4:3)=2
01705c17: Ch02= 17, Ch03= 17, Ch04= 17
01705c17: Ch05= 17, Ch06= 17, Ch07= 17
01705c17: Ch08= 17, Ch09= 17, Ch10= 17
01705c17: Ch11= 17, Ch12= 17, Ch13= 17
00005c17: Ch14= 17, Ch15= 0
40400014: BC(11:0)= 14, BC(2:0)=4, CRC= 80
00000030: [Tile] Errors=0, FPGA=0, Chan=48
81705cbc: KChr= bc, Ch00= 17, Ch01= 17, BC(6:5)=0, BC(4:3)=2
01705c17: Ch02= 17, Ch03= 17, Ch04= 17
01705c17: Ch05= 17, Ch06= 17, Ch07= 17
01705c17: Ch08= 17, Ch09= 17, Ch10= 17
01705c17: Ch11= 17, Ch12= 17, Ch13= 17
00005c17: Ch14= 17, Ch15= 0
40400014: BC(11:0)= 14, BC(2:0)=4, CRC= 80

f1811021: User header: Length=1, LiA Slice: PP FADC=2, LUT=0, CP=1, JEP=1, PP Lower Bound=24 (N.Worc
c4817429: SubBlock: PPM, Version=2, Compressed, CompVer=1, Crate=7, Module=4, nSlice2=5, nSlice1=1
TREX
readout
packet
Mcm00 Ch0A (00): {4} LUT-C 23 (PSE=100) LUT-J 23 (RHL=000) ADC 92, 0, 93, 0, 94
Mcm01 Ch0A (04): {4} LUT-C 23 (PSE=100) LUT-J 23 (RHL=000) ADC 92, 0, 93, 0, 94
Mcm02 Ch0A (08): {4} LUT-C 23 (PSE=100) LUT-J 23 (RHL=000) ADC 92, 0, 93, 0, 94
Mcm03 Ch0A (12): {4} LUT-C 23 (PSE=100) LUT-J 23 (RHL=000) ADC 92, 0, 93, 0, 94
Mcm04 Ch0A (16): {4} LUT-C 23 (PSE=100) LUT-J 23 (RHL=000) ADC 92, 0, 93, 0, 94
Mcm05 Ch0A (20): {4} LUT-C 23 (PSE=100) LUT-J 23 (RHL=000) ADC 92, 0, 93, 0, 94
Mcm06 Ch0A (24): {4} LUT-C 23 (PSE=100) LUT-J 23 (RHL=000) ADC 92, 0, 93, 0, 94
Mcm07 Ch0A (28): {4} LUT-C 23 (PSE=100) LUT-J 23 (RHL=000) ADC 92, 0, 93, 0, 94
Mcm08 Ch0A (32): {4} LUT-C 23 (PSE=100) LUT-J 23 (RHL=000) ADC 92, 0, 93, 0, 94
Mcm09 Ch0A (36): {4} LUT-C 23 (PSE=100) LUT-J 23 (RHL=000) ADC 92, 0, 93, 0, 94

```

**Figure 8.11.** Single event captured and read out by the eFEX and TREX for comparison of the data alignment consistency. Encircled in red are the values sent by the TREX and received on the eFEX. The results are in agreement, the eFEX uses a notation in hexadecimal, while the TREX values are in decimal.

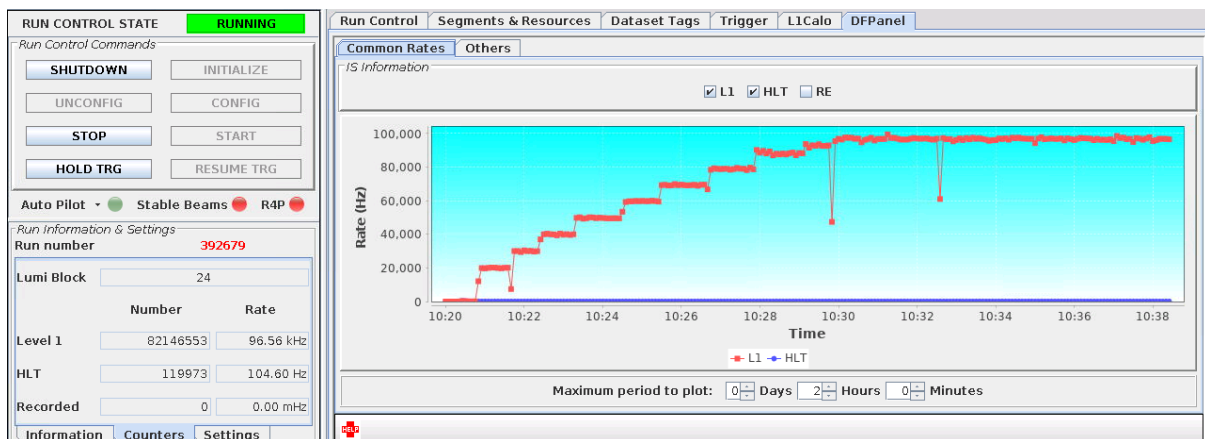
## 8.5 Interfaces to the DAQ

### 8.5.1 Legacy ROD readout

The integration of the TREX modules with the legacy L1Calo RODs is crucial for combined running in ATLAS and for performing calibrations with the Tile Calorimeter. It requires the TREX modules to be able to read out with a rate of up to 100 kHz and support configurable readout modes.

The readout functionality was tested successfully up to 60 kHz in the laboratory test-rig at CERN for a single module, prior to installation. After the installation of the 32 modules in P1, readout tests with the full L1Calo system began in a standalone environment. The modules ran error-free, however, limitations on the resources of the DAQ PCs in the standalone mode do not allow the rate to reach higher rates with the RODs reporting high backpressure and a large busy fraction.

To test the system in a high-rate configuration, the full ATLAS DAQ infrastructure was requested with true random triggers up to 100 kHz. During the initial tests in the ATLAS partition, BCID and Event mismatches between the TREX modules and the RODs were seen. The mismatches were solved by adjusting the de-randomization buffers of the incoming nMCM data streams within the GLink formatting block of the PREDATOR FPGA, and by decreasing the minimum gap needed for processing 2 consecutive GLink packets. This required specific changes to ensure the proper timing was maintained throughout the full event readout processing.



**Figure 8.12.** The ATLAS TDAQ Run Control graphical user interface depicting a gradual ramp-up of the L1 rate for testing the rate handling of the TREX system using random triggers.

Once the necessary changes were applied in the firmware, the readout was tested in a rate ramp configuration. Starting with 100 Hz the rate was ramped up in steps of 10 kHz. With the random triggers set to the highest available rate, the TREX modules are able to reach and sustain a rate of 100-105 kHz. The results of the test are presented in

Fig. 8.12, where the TDAQ panel and the rate of triggers are shown.

The high-rate readout functionality was validated in the ATLAS partition during allocated testing slots and multiple ATLAS Milestone Weeks. With stable overnight runs and sustained error-free high-rate data-taking, the TREX modules demonstrate full and complete readout functionality.

### 8.5.2 Integration with the FELIX and SW ROD

The integration with the Phase-I upgrade readout via the FELIX and SW ROD is done in parallel. Since it is the baseline for Run-3 and will act as the main readout path after the commissioning period, complete integration and validation is necessary.

Two commodity servers, each hosting a FELIX PCIe card are dedicated to the TREX system. Each card connects to 16 TREX modules, with one receiver fibre and one transmitter fibre per module. The receiver fibre carries the clock and the Timing, Trigger and Control (TTC) information to the TREX, while the transmitter fibre transfers the triggered event data to the FELIX. Once the event packets arrive, the FELIX forwards them to the SW ROD server via a 100 Gbps switching network using the NetIO protocol [78]. In addition, the FELIX also forwards the TTC information to the SW ROD for synchronisation purposes. On the SW ROD nodes, the data is synchronised with the TTC packets by comparing the L1ID and BCID quantities. After finding the corresponding data packet for each TTC packet, the custom processing plugin strips the TREX header and trailer, mentioned in 6.6.2, and passes the S-Link payload to the ATLAS event builder.

After tuning the available timeouts and buffer sizes, specifically tailored to the data rate that the TREX operates at, stable data-taking and event building has been achieved at rates up to 100 kHz in the ATLAS partition.

The data runs were recorded for further analysis and comparison with the data collected by the legacy ROD. So far, the data collected via the SW ROD matches with the event data from the legacy ROD system. By tuning multiple FELIX and SW ROD parameters, a continuous data flow is achieved with a rate of up to 108 kHz.

Overall, the event data readout of the TREX via the FELIX and SW ROD path demonstrates stable and great performance over many data-taking runs.

The TREX modules primarily receive the LHC clock and the TTC information through the TTCrx chip on the PPMs, as mentioned in 6.4, however, a full recovery of the LHC clock and TTC information decoding from FELIX is also implemented in the PREDATOR FPGA design. The ability to receive the clock from both the PPM and FELIX has been very beneficial during BCID synchronisation tests and for the migration to the new ALTI system.

The TTC commands such as the Bunch-Counter Reset (BCR), Level-1 Accept (L1A)

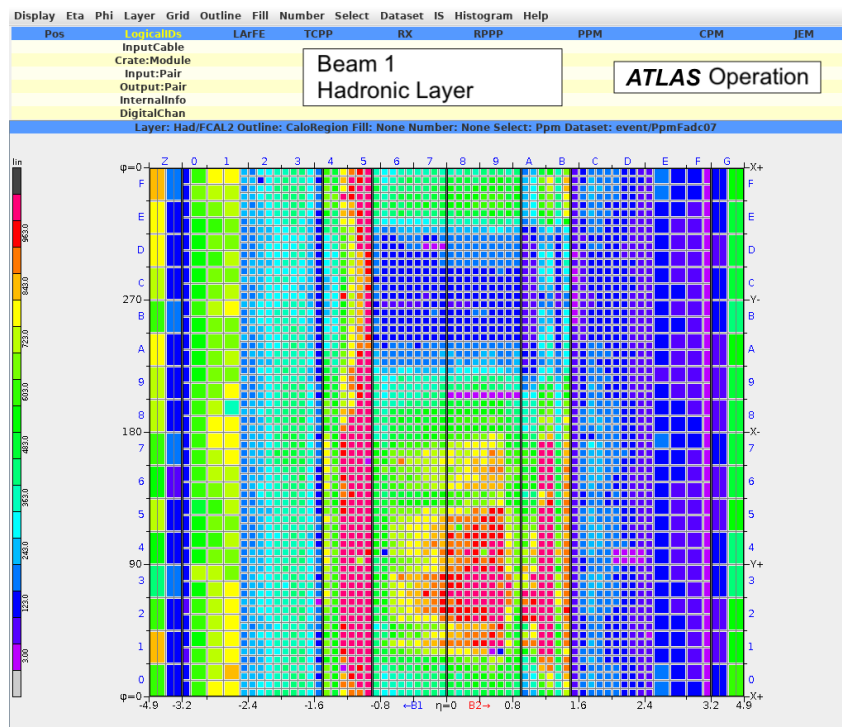


and Event Counter Reset (ECR), as described in Sec. 6.4, sent from the FELIX via Gigabit Transceiver (GBT) packets are decoded and counted on the processing FPGA. As an internal validation, the counters between the two TTC sources have been compared and show identical values. This means that the commands arrive intact, without errors during transmission and decoding.

## 8.6 Performance and first data

### 8.6.1 Commissioning and Data Monitoring

During the final stages of the commissioning period, the LHC provided dedicated runs with proton bunches that hit collimators placed 150 m away from the detector. The interactions with the collimators create large showers of secondary particles which cause simultaneous activity in all subsystems of the ATLAS detector.



**Figure 8.13.** Splash event displayed through the L1Calo mapping tool [39]

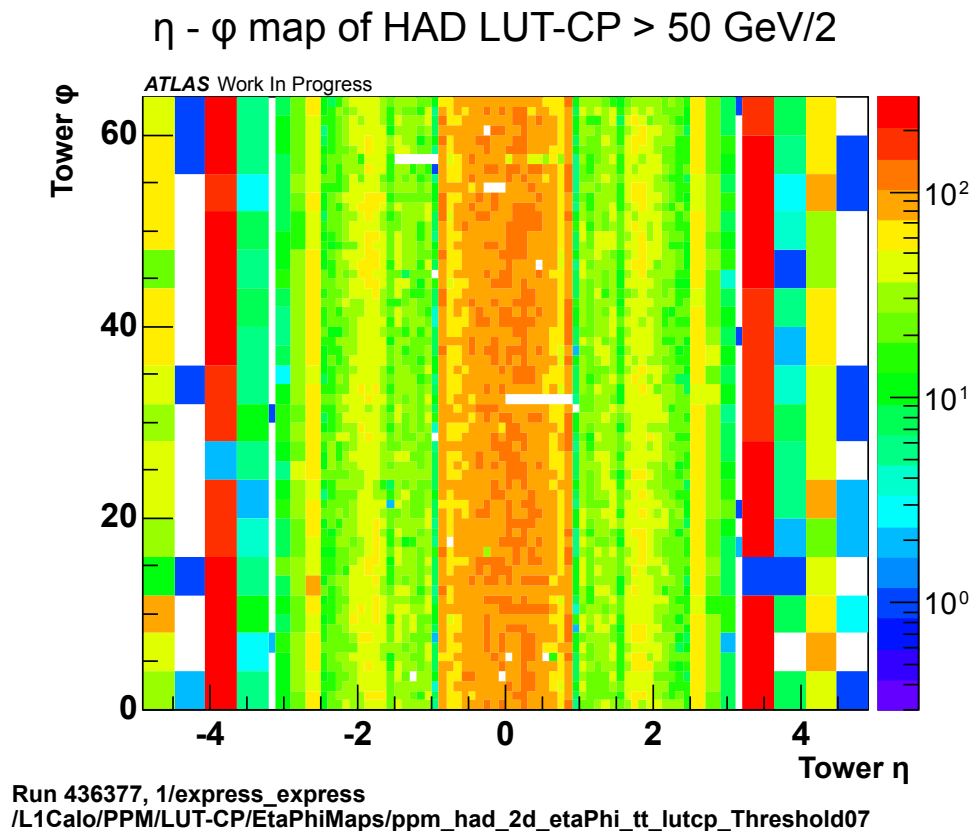
These events are referred to as *beam splashes* which allow verifying the readout of each subsystem and to synchronise with the LHC [79].

This applies also to the calorimeters and the PPr system. The calorimeter energy deposits, induced by the splashes, are used to readout out the full coverage of the TREX modules. For the first set of showers, the data from the legacy RODs were recorded, however, the FELIX-SW ROD path was enabled in parallel during the last runs of the beam splashes.

Fig. 8.13 presents an  $\eta - \phi$  map of the calorimeter activity in the hadronic layer, caused by the splashes moving from positive to negative  $\eta$  values. The displayed trigger-tower colours correspond to the central slice value of the pulse digitised on the nMCMs. For these runs, the readout is configured to store 15 ADC samples per event. The central barrel region  $|\eta| < 1.6$  is covered by the Tile calorimeter and read out by the TREX modules, demonstrating correct behaviour with the pulses following the path of the particles through the detector from right to left.

The trigger data provided by the TREX is also validated through the ATLAS Tier-0 data quality monitoring. After each run, a sub-set of recorded events are processed to give an overview of the detector status. Dedicated monitoring histograms are automatically generated which show the conditions of the PPr system.

As an example, a generated  $\eta - \phi$  map is shown in Fig. 8.14. Each entry corresponds to a trigger-tower with a Look-Up-Table (LUT) value exceeding a threshold of 50 GeV.



**Figure 8.14.**  $\eta - \phi$  map of trigger-towers with Look-Up-Table values greater than 50 GeV, taken from the ATLAS Tier0 data-quality histogramming [80].

## 8.6.2 First collision data

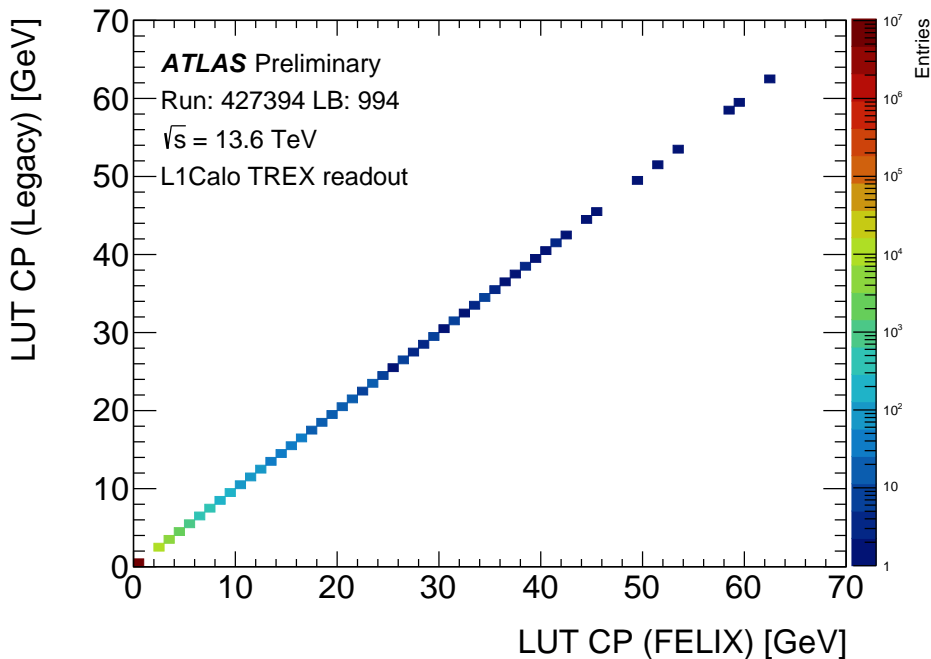
The final validation of the TREX functionality is done using collision data, where direct comparisons between the legacy and upgrade data paths are performed. Early samples



of 13.6 TeV proton-proton collision data, taken in July of 2022, are used to validate the performance of the new modules.

For verifying the consistency of the two parallel readout methods and their respective data paths, the recorded event data is analysed and compared between the two. One way to perform the comparison is to decode the raw bytestream data and correlate each parameter, such as the flash ADC and LUT values, extracted from the two readout streams.

An example is shown in Fig. 8.15, where the horizontal axis represents the LUT value extracted from the new FELIX packet, while on the vertical axis the same parameter is extracted from the legacy ROD path. The results show perfect agreement between the readout methods, indicating correct firmware implementation for the TREX processing FPGA. In addition, a second verification method that performs a bit-wise comparison of the entire payload packets is also used.

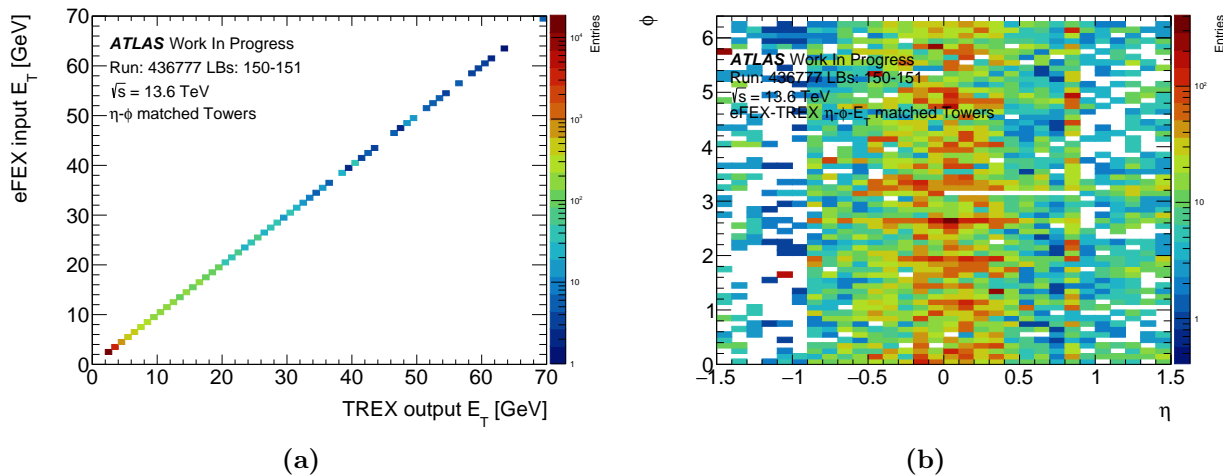


**Figure 8.15.** A comparison of Look-Up-Table values read out via FELIX and legacy ROD paths, the LUT CP notation implies a granularity of  $0.1 \eta \times 0.1 \phi$  transmitted to the legacy CP, eFEX and jFEX systems.

With the readout path validated, the real-time data path results that lead to the trigger decision can be monitored.

The  $\eta$ – $\phi$  coordinates and  $E_T$  results must match between the trigger towers processed by the PPM-TREX system and the FEX modules. By matching the  $\eta$  and  $\phi$  coordinates between the trigger towers of the PPM-TREX and the FEX modules, the transverse energy results can be compared. If any discrepancies are present in the matched results, it would indicate mapping inconsistencies between the modules or issues in the transmission.

The result is presented in Fig. 8.16, where the trigger towers sent by the TREX are matched with the input towers of the eFEX. The triggered event information contains both the TREX output data and eFEX inputs.



**Figure 8.16.** The transverse energy results read out on TREX and eFEX for triggered events. A matching is performed to correlate the eFEX input data with the TREX output (a). The  $\eta - \phi$  coordinates corresponding to the matched  $E_T$  values (b).

Fig. 8.16 (a) depicts a comparison of the  $\eta - \phi$  matched trigger tower  $E_T$  values between the TREX output and the eFEX input. The horizontal axis shows the  $E_T$  value sent to the eFEX module, read out via the TREX. The vertical axis represents the  $E_T$  values received by the eFEX at the input stage.

Fig. 8.16 (b) is the corresponding  $\eta - \phi$  map of the matched towers. The results are in agreement and illustrate the correct and expected behaviour of the real-time data processing.

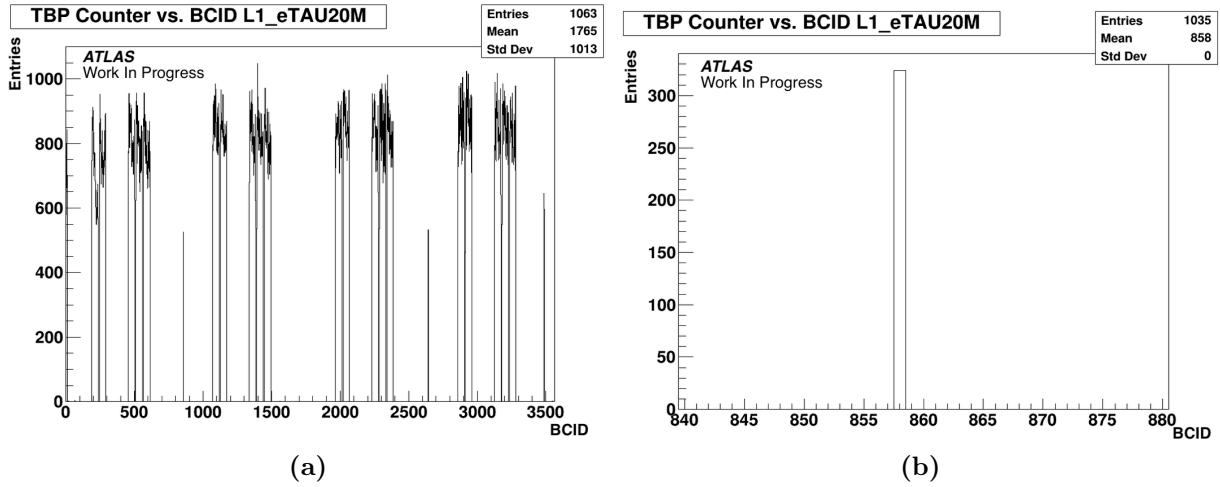
### Timing of the triggered items

To fully commission the system of TREX modules, triggers based only on the PPM-TREX  $E_T$  results are required to arrive in-time at the Central Trigger Processor. This means that the BCID sent along with the  $E_T$  values to the FEX processors must correspond to the BCID of the collision. In addition, the trigger objects formed by the FEX processors propagate this BCID to the CTP.

The validation is performed using pp collisions at 13.6 TeV, where during the data-taking run only the Tile calorimeter inputs are enabled. This allows for precise verification of the full trigger chain from the TREX to the FEX processors, followed by the L1Topo and ending at the CTP.

The  $\tau$ -based trigger algorithms in ATLAS rely on the hadronic calorimeter. This provides the opportunity to verify if TREX-only based triggers are assigned with the

correct BCID, within the BC where the collision occurred.



**Figure 8.17.** The trigger rate for tau candidates counted at the CTP for multiple orbits.

The triggers are based on the energy deposits provided only by the TREX and the tau algorithm running on the eFEX. The rate is a function of the BCID (a) and illustrates the LHC bunch structure within the 3564 possible values (a). A zoomed-in view of an isolated bunch-crossing (b).

Fig. 8.17 (a) presents a global overview of the timing of the eTAU20 triggers, which are  $\tau$ -candidates formed on the eFEX, based solely on energy deposits provided by the TREX. The trigger rates follow a visible pattern for every revolution of the LHC beam which corresponds to the bunch structure. (b) shows the same trigger item for an isolated bunch at BCID 858. The result showcases the very good timing stability of the system, where no leakage into the neighbouring BC is observed.

The PPM-TREX-based TAU triggers reaching all the way to the CTP illustrate excellent timing and correct BCID assignment of the real-time data frames.

Overall, the system of TREX modules is fully integrated into ALTAS and all of its functionality is verified. The modules run with high reliability and excellent performance has been demonstrated with the real-time path, providing well-timed trigger items to the CTP, and with the event data readout reaching sustainable high rates.



# Chapter 9

## Conclusions

This thesis focused on the research and development cycle of the Tile Rear Extension (TREX) modules which are installed and commissioned as part of the Phase-I upgrade of the ATLAS Level-1 Calorimeter (L1Calo) Trigger system.

The TREX is an extension of the L1Calo PreProcessor (PPr) system. Housing five Field-Programmable Gate Arrays (FPGAs), optical transceivers for fast data processing and a System-On-Chip (SoC) for monitoring and control.

The TREX modules are responsible for delivering the digitised Tile Calorimeter energy deposits to the new L1Calo Phase-I and legacy systems via optical and electrical outputs, respectively. The new trigger algorithms, running on the Feature Extractor processors, benefit greatly from the data provided by the TREX, such as improvements in the electron isolation, jet and tau energy resolution and more precise  $E_T^{miss}$  calculations, bringing overall higher trigger efficiencies.

Since its inception, the TREX hardware has undergone three iterations of design changes, each time improving and adding new functionality. Both the hardware and firmware designs focus on fast data processing and transmission, redundancy and optimised latency for the signals on the trigger path.

Most of the firmware development was performed and improved during the testing of the pre-production and production batches. A set of test routines has been developed to analyse each aspect of the board, such as the configuration of the logic devices, the readout and real-time paths, integration with the PreProcessor Modules (PPM) and the System-On-Chip functionality.

The latency of the system was determined through multiple sets of measurements and passed all requirements, staying within the envelope. Acceptance tests were carried out for the production boards, with every single module passing stringent performance verification. Special attention was given during the soak tests to the optical transmission path. The quality of the transmission was validated for over 1600 optical links prior to

installation.

After finalising the functional tests, 32 modules out of the production batch were selected to be installed in ATLAS. The installation in ATLAS brought unique challenges, as the existing PreProcessor crates were fully dismantled and modified to host the new modules. Every step of the installation and infrastructure availability required careful planning and management due to strict thermal, power and geometrical constraints within the PreProcessor racks.

After the successful installation in May of 2021, the hardware underwent low-level testing for any hardware faults. All available configuration protocols are utilised to establish communication with the ATLAS Trigger and Data Acquisition (TDAQ) run control software and the monitoring framework running on the System-On-Chip provides environmental data and the overall health status of the boards to the ATLAS Detector Control System.

The system integration began with the existing legacy interfaces to prepare for combined running with the other ATLAS subsystems. Here, strict time constraints were present, as the legacy L1Calo trigger system was required to be fully functional well ahead of time, before the LHC operation began, while the Phase-I path required more commissioning time, well into the start of Run-3. With multiple integration tests in ATLAS and new firmware iterations, the legacy real-time path and the readout were brought step-by-step to operating conditions at the highest possible data rates. During the commissioning phase, the firmware functionality of the TREX processing FPGA continued to develop and mature, achieving stable, high-rate readout at over more than 100 kHz.

With the legacy path fully validated, the focus shifted to the Phase-I interfaces. The optical link mapping to the new FEX processors was thoroughly understood and their stability was put under-test through various bit-error ratio schemes. The contents of the data frames sent by TREX were directly compared with the frames received by the FEX processors, indicating perfect agreement.

The timing of the data frames was also finely tuned to assign the energy deposits with the Bunch-Crossing Identifier (BCID) of their corresponding collision. With all parameters in place, TREX-based tau-triggers were monitored at the Central Trigger Processor (CTP), which showed excellent timing results with all triggers being in-time.

The new readout path via the Front-End Link Exchange (FELIX) and Software Readout Driver (SW ROD) was studied and integrated in parallel and the TREX became the first Phase-I module to show stable performance at an event-building rate of around 108 kHz, which was very helpful in overall FELIX commissioning.

At the start of Run-3 in July 2022, the PreProcessor system with the TREX modules was fully integrated in ATLAS and ready for data-taking, having demonstrated excellent performance in its functionality during prior pilot beams and combined running campaigns. So far, the TREX is operating successfully over many months of physics data-taking and pivotal in providing triggers from the pp collisions at  $\sqrt{s} = 13.6$  TeV.

### Looking Forward

After Run-3, the High Luminosity LHC upgrade will bring further challenging conditions for ATLAS. The luminosity increase and pileup of  $\langle\mu\rangle = 200$  will allow the experiments to record up to  $4000 \text{ fb}^{-1}$  of data. To cope with the harsher conditions, ATLAS is scheduled to undergo new and sophisticated upgrades. This includes the TDAQ and Calorimeter systems. The readout electronics of the Tile Calorimeter will undergo a complete redesign. A dedicated digital pre-processing back-end system is currently under development. Moving from analogue signals, the system will be entirely digital and will provide digitised data to the ATLAS trigger system with improved energy resolution. In designing the trigger interface (TDAQi) for the Tile PreProcessor, the experience gained in developing the TREX has been invaluable due to the similarities in high-speed data processing and transmission.





# Appendix A

## Glossary and conventions

- ATCN: ATLAS Technical and Control Network
- ATCA: Advanced Telecommunications Computing Architecture
- BC: Unit of time in Bunch Crossing (25 ns)
- BCID: Bunch Crossing Identifier
- BCR: Bunch Counter Reset
- BER: Bit-Error Ratio
- CP: Cluster Processor
- CPM: Cluster Processor Module
- CDC: Clock Domain Crossing
- CPLD: Complex Programmable Logic Device
- DCS: Detector Control System
- ECR: Event Counter Reset
- eFEX: Electron Feature EXtractor
- eMMC: embedded MultiMediaCard
- FireFly: Samtec Micro Flyover System
- FELIX: Front-End Link eXchange
- FEX: Feature EXtractor
- FPGA: Field Programmable Gate Array

- FSM: Finite-State-Machine
- gFEX: Global Feature EXtractor
- GTH: Xilinx Gigabit Transceiver type H
- GBT: GigaBit Transceiver protocol
- GPIO: General Purpose IO
- HDL: Hardware Description Language
- HLT: High-Level Trigger
- I2C: Two-wire serial protocol
- JEM: Jet Energy Processor Module
- JEP: Jet Energy Processor
- jFEX: Jet Feature EXtractor
- JTAG: IEEE 1149.1 protocol for FPGA programming and verification
- L1A: Level-1 Accept signal
- L1Calo: Level-1 Calorimeter trigger system
- LC: Lucent Connector fibre optic
- LUT: Look-Up Table
- LVDS: Low-Voltage Differential Signal
- MGT: Multi-Gigabit Transceiver
- MMCM: Mixed-Mode Clock Manager
- MMIO: Memory-Mapped Input and Output
- MPSoC: Multi-Processor System-on-Chip
- MTP: Multi-fibre Termination Push-on
- nMCM: new Multi-Chip Module
- PCB: Printed Circuit Board
- PL: Programmable Logic

- 
- PS: Processing System
  - PLL: Phase-Locked Loop
  - PPr: PreProcessor
  - PPM: PreProcessor Module
  - RAM: Random Access Memory
  - ROD: Read-Out Driver
  - RoI: Region of Interest
  - ROM: Read Only Memory
  - RTM: Rear Transition Module
  - SBC: Single-Board Computer
  - SMAP: SelectMAP programming mode
  - SoC: System-on-Chip
  - SPI: Serial Peripheral Interface
  - TDAQ: Trigger and Data Acquisition
  - TREX: Tile Rear EXtension
  - TileCal: Tile Calorimeter
  - TTC: Timing, Trigger and Control
  - TTCrx: TTC Receiver Chip
  - VIA: Electrical inter-layer connection
  - VME: Versa Module Europa bus
  - XML: Extensible Markup Language



# List of Figures

2.1	The Standard Model of Elementary Particles [1] . . . . .	6
2.2	The total production cross-sections predicted by the SM and measured by the ATLAS detector [5]. . . . .	7
3.1	The CERN accelerator complex [11]. . . . .	10
3.2	The ATLAS Detector [12]. . . . .	13
3.3	A cross-section of the ATLAS Inner Detector depicting the Pixel Detector, Semiconductor Tracker and the Transition Radiation Tracker [13]. . . . .	14
3.4	The ATLAS Magnet System. The solenoidal magnetic field lines are illustrated in green, the toroidal magnetic field lines in blue [14]. . . . .	15
3.5	The ATLAS Calorimeters. LAr is shown in yellow-gold and the Tile in grey colour [15]. . . . .	16
3.6	Segmentation of the Tile Calorimeter for the Long Barrel (a) and the Extended Barrel (b) [18]. . . . .	18
3.7	A cut-out of the ATLAS detector illustrating the Muon Spectrometers [19].	18
4.1	Overview of the ATLAS TDAQ system, taken and modified from [22]. . . . .	20
4.2	Overview of the L1Calo System in Run-2 [20]. . . . .	21
4.3	The PreProcessor Module in Run-2 (a) and its functionality (b), taken and modified from [27]. . . . .	22
4.4	The CP algorithm (left) depicting the core (local maximum) and the isolation and the definition of the RoI (right) [24]. . . . .	24
4.5	The three sizes of the sliding window of the JEP algorithm [24]. . . . .	24
5.1	Schematic of the L1Calo system in Run-3. The Phase-I upgrade modules are outlined in yellow. The existing legacy system is depicted alongside the new system in blue and green. Taken and adapted from [20]. . . . .	29
5.2	Single modules of the eFEX, jFEX and gFEX [34]. . . . .	30

5.3	Overview of the eFEX trigger algorithm [35] with the calorimeter layer segmentation depicted on the left and the $R_\eta$ condition on the right. The $R_\eta$ cluster in Layer-2 is illustrated in yellow and the environment in blue. The seed cell $A$ is the local maximum in $\eta$ , $B$ is the highest neighbour of $A$ in $\phi$ . . . . .	31
5.4	The diagram of the jFEX seeding algorithm (a) and the definition (b) of a jFEX jet candidate with a radius of $R = 0.4$ [36]. . . . .	32
5.5	Example event with large-R jet candidates spanning across multiple FPGAs of the gFEX module [37]. . . . .	33
5.6	Single-electron efficiency comparison between Run-2 and the Run-3 eFEX [39]. . . . .	34
5.7	Di-tau trigger efficiency comparison between Run-2 and the Run-3 eFEX and jFEX [39]. . . . .	35
5.8	$E_T^{miss}$ algorithm efficiency comparison between Run-2 and Run-3 jFEX and gFEX [39]. . . . .	36
6.1	A diagram illustrating the various data paths of the PPM and TREX. . .	38
6.2	The first TREX prototype (a) is depicted with the FPGA and optical transceivers exposed. The pre-production (b) is shown with the latest custom heatsinks. . . . .	40
6.3	A partially assembled production (V3) TREX without a heatsink and CompactPCI connectors and FireFly transceivers. It is used to showcase the various components of the TREX. . . . .	41
6.4	A schematic overview of the ExtReM firmware. The orange-coloured logic blocks illustrate the new or modified functionality for the TREX. . . . .	43
6.5	The VME address-space of the PPM and TREX. . . . .	45
6.6	The address-space and the divisions for the TREX components. . . . .	45
6.7	Control interface from VME to the TREX FPGAs. The data passes through decoding stages in the VME CPLD and the ExtReM before being forwarded to the TREX FPGAs via a custom 8-bit interface. . . . .	46
6.8	Timing diagram depicting a write cycle of the custom 8-bit protocol. . .	47
6.9	The schematic overview of the TREX I2C bus with all connected on-board devices. . . . .	49
6.10	A diagram depicting the available FPGA programming methods via the VMEbus. . . . .	50
6.11	The clock and TTC distribution network on the PPM and TREX. . . . .	52

---

6.12	The Bunch-Crossing Multiplexing (BCMUX) scheme illustrating the possible combinations of the data contents and the BCMUX flag, and the 12-bit data frame of the real-time path sent to the CP and JEP. Adapted from [55]. . . . .	54
6.13	Functional overview of the DINO FPGA design. The design is segmented into a monitoring, control block and LVDS data duplication. The LVDS streams are separated with copies transmitted to the PREDATOR FPGA and to the CP/JEP subsystems. Certain channels are duplicated for CP/-JEP due to overlap regions. . . . .	56
6.14	A behavioural simulation of the TREX deserialisation block of the incoming LVDS streams from the nMCMs. . . . .	57
6.15	The TREX data formats prepared for each type of FEX processor. The data fields are arranged according to FEX requirements and the tower granularity. . . . .	59
6.16	An example of the align frame issued by the TREX and its formatting. The second 32-bit word contains geographical information for identifying the TREX origin. . . . .	59
6.17	Bit and Byte ordering for the 8b10b Encoding [57]. The 32-bit data is encoded into 40-bit wide codes to achieve a DC-balanced output. . . . .	60
6.18	Simplified schematic overview of the high-speed transmitter interface implemented in the PREDATOR FPGA. The data prepared for transmission undergoes 8b/10b encoding and a clock domain crossing into the serialiser clock. At the end of the logic block, the data is serialised and driven out. . . . .	61
6.19	The real-time data flow in the PREDATOR FPGA and the estimated latency values. The values shown in green are obtained from the simulation of the logic. Values obtained from physical measurements are highlighted at two stages and show the latency starting from the LVDS input stage up to the measurement point. . . . .	62
6.20	A simplified schematic diagram of the TREX readout path implemented in the FPGA. The three processing clock domains are presented in different colours. Between each clock domain a buffer symbol indicates the crossing. . . . .	64

6.21	The formatting of the TREX data sent to the FELIX. It is composed of four 32-bit header words, which include metadata describing the event such as the Level-ID and the BCID of the triggered event, the number of transmitted read-out ADC and Look-Up Table (LUT) values and also geographical information. The S-Link event data starts with a header of its own, followed by the dynamically sized and compressed event data. The TREX trailer is attached at the end, containing further auxiliary information and status flags. . . . .	66
6.22	Finite-State-Machine for initiating the transmission to the FELIX. . . . .	67
6.23	Timing diagram illustrating the processing of events at a trigger rate of 100 kHz. The diagram shows the data flow through the GLink formatting, compression and packing into the FELIX format. The start and end of processing an event are indicated by the <i>EventStart</i> and <i>EventEnd</i> flags. The gap between two events is indicated when the <i>SlinkDAV</i> is driven low. During this gap period, the readout to FELIX is initiated and shown in <i>FlxData</i> . . . . .	69
6.24	A simplified diagram of the PL and PS sectors of the TREX Zynq. . . . .	70
6.25	Schematic overview of the logic blocks of the TREX Zynq PL and PS. . . . .	71
6.26	Finite-State-Machine for arbitrating the I2C bus master between the ExtReM and the Zynq. . . . .	73
6.27	The schematic overview of the Zynq PS [66]. The various processing and IO units are illustrated along with their internal interconnects. . . . .	74
6.28	The TREX Zynq booting methods for loading the PL bit stream and the operating system. The nominal configuration is done via an SD Card (a). The fallback solution (b) uses a combination of QSPI memories and eMMC storage. . . . .	78
6.29	The datapoints elements and their corresponding types in a hierarchical illustration, generated via the quasar framework and adapted. An example of the datapoint notation is shown on the right. . . . .	79
6.30	A schematic diagram illustrating the machinery of the TREX Monitoring framework. Two threads run in parallel, one is tasked with gathering the environmental data by communicating with the hardware. The data is stored in temporary buffers, which both threads share access to. The second thread reads the data from the buffer and prepares it for the consumption of an OPC UA client. . . . .	81
6.31	The JTAG chain of the PPM and TREX modules. A programmable multiplexer allows selecting between the Zynq and the Digilent devices as the master, and between the various FPGA chains. . . . .	83



- 
- 7.1 A flow chart diagram illustrating the acceptance test procedure for the production TREX modules after the visual hardware inspections and assembly. . . . . 86
- 7.2 VME crate fully populated with TREX modules in the laboratory. The VME crate has undergone mechanical and electrical modifications for hosting the modules. . . . . 87
- 7.3 Grafana dashboard showing the time evolution of the TREX FPGA voltages and temperatures in the laboratory environment, monitored via the SoC. . . . . 88
- 7.4 The measurement of the input (green) and output (blue) 40.08 MHz clock quality of the PREDATOR FPGA (a). The output of the Si5345 PLL jitter attenuator (b), which provides the reference clocks for the MGTs, measured using a differential probe. . . . . 89
- 7.5 A schematic overview of the setup for measuring the latency at the TREX outputs for the legacy path. The left image shows the procedure for the Run-2 conditions, prior to the TREX. The right image illustrates the measurement where a PPM is equipped with a TREX module. . . . . 90
- 7.6 The latency measurement of the signal propagation time through the PPM and the TREX for the legacy trigger path. The yellow-coloured pulse is measured at the PPM input and the output LVDS stream is shown in green for CP (a) and brown for JEP(b). The 40.08 MHz LHC clock, shown in blue, is used as a reference. . . . . 91
- 7.7 The measurement results of the internal latency of the PREDATOR FPGA. As a starting reference, the LVDS stream is monitored for non-zero data. Once the incoming data is de-serialised and processed internally, a flag is raised (blue-coloured) to indicate the arrival of data at a specific logic block. 92
- 7.8 The latency measurement of the FPGA logic and the high-speed transmission path of the TREX. The magenta-coloured signal is issued on the transmitter module acting as a start signal for the measurement. Upon capturing and decoding the data, a stop signal, coloured in yellow, is issued on the receiver module. The 40.08 MHz LHC clock (aqua) is used as a reference. . . . . 93
- 7.9 Oscilloscope measurements of the LVDS signals at the TREX output, designated for the CP (a) and JEP (b) systems. (c) depicts the measurement in the persistence mode through a heat-map of the transmission over an integrated period of time. . . . . 95

7.10	The test setup for measuring the optical link quality of the TREX modules. The leftmost module is the transmitter device under-test. It is connected with four fibre ribbons to the four modules on the right, which act as receivers. . . . .	96
7.11	A set of eye-scan diagrams illustrating the transmission quality of the optical links on the TREX. The data is captured on a TREX module with a receiver interface. The horizontal axis represents time in Unit Intervals for a link speed of 11.2 Gbps ( $UI = 1/11.2[GHz]$ ). The vertical axis represents the signal amplitude in voltage codes. . . . .	97
8.1	The back view (a) of the VME crate with TREX modules inserted. The frontal view (b) with PPMs re-inserted after the hardware adaptations.	100
8.2	Bundles of LVDS cable insertion after the TREX installation (a), which connect to the CP and JEP systems. The optical fibre connection to the FEX and FELIX systems (b). . . . .	101
8.3	Temperature profile for the major TREX components in each slot within one VME crate. The environmental data is captured during a typical data-taking period, where all inputs and outputs are under continuous load.	103
8.4	The mapping of the PPM-TREX in the $\eta - \phi$ space. Each crate processes half of the $\eta$ coverage. Only one quadrant is depicted with the same mapping repeating for the other three quadrants. . . . .	104
8.5	The TREX align frame sent on BCID 3500 and captured in eFEX memories. The payload includes the origin device identifier and the destination FEX type. In addition, further geographical information is added and a payload indicator makes finding the frame more human-friendly. . . . .	105
8.6	Frontal (a) and interior (b) view of the readout fibre exchange box showcasing the splitting between the legacy ROD fibres (orange) and the FELIX fibres (aqua). . . . .	106
8.7	The output result of an input timing scan for the links of a single CP module. The deskew clock is shifted to align the incoming serial data with the de-serialisation clock. The region in white indicates valid deskew values, while the blue region indicates de-serialisation errors (a). A graphical panel monitoring of the parity error counters of the input signals for a single JEP module(b). . . . .	106
8.8	The ATLAS TDAQ Run Control interface depicting the status of the serialisers on the CP modules with TREX inputs. The links are monitored every few seconds and report back in case of parity errors. . . . .	107

- 8.9 The ATLAS TDAQ Run Control interface indicating the optical input link status for single eFEX (a), jFEX (b) and gFEX (c) modules. The grey-coloured links are unused and disabled. In case of transmission issues, the link error status is propagated through colour changes and notifications. . . . . 108
- 8.10 Data seen in a single data channel (noted as LUT Value) and its corresponding assigned BCID value, captured through a logic analyser. Two isolated collisions are present, at BCID 1 (a) and 1786 (b). The results show correct assignment between the LUT and the BCID values. . . . . 109
- 8.11 Single event captured and read out by the eFEX and TREX for comparison of the data alignment consistency. Encircled in red are the values sent by the TREX and received on the eFEX. The results are in agreement, the eFEX uses a notation in hexadecimal, while the TREX values are in decimal. 110
- 8.12 The ATLAS TDAQ Run Control graphical user interface depicting a gradual ramp-up of the L1 rate for testing the rate handling of the TREX system using random triggers. . . . . 111
- 8.13 Splash event displayed through the L1Calo mapping tool [39] . . . . . 113
- 8.14  $\eta - \phi$  map of trigger-towers with Look-Up-Table values greater than 50 GeV, taken from the ATLAS Tier0 data-quality histogramming [80]. . . . . 114
- 8.15 A comparison of Look-Up-Table values read out via FELIX and legacy ROD paths, the LUT CP notation implies a granularity of  $0.1 \eta \times 0.1 \phi$  transmitted to the legacy CP, eFEX and jFEX systems. . . . . 115
- 8.16 The transverse energy results read out on TREX and eFEX for triggered events. A matching is performed to correlate the eFEX input data with the TREX output (a). The  $\eta - \phi$  coordinates corresponding to the matched  $E_T$  values (b). . . . . 116
- 8.17 The trigger rate for tau candidates counted at the CTP for multiple orbits. The triggers are based on the energy deposits provided only by the TREX and the tau algorithm running on the eFEX. The rate is a function of the BCID (a) and illustrates the LHC bunch structure within the 3564 possible values (a). A zoomed-in view of an isolated bunch-crossing (b). . . . . 117



# List of Tables

3.1	The naming conventions of LHC runs, the years of operation and the corresponding ATLAS upgrade phase. . . . .	11
6.1	Overview of the real-time output to the FEX processors . . . . .	55
7.1	Latency estimations determined from physical measurements . . . . .	94
8.1	Typical current and power draw from the 3.3 V and 5 V sources during running in P1 . . . . .	102



# Bibliography

- [1] Standard Model. Standard Model of Elementary Particles. [https://en.wikipedia.org/wiki/Standard\\_Model](https://en.wikipedia.org/wiki/Standard_Model).
- [2] Peter W. Higgs. Broken symmetries and the masses of gauge bosons. *Phys. Rev. Lett.*, 13:508–509, Oct 1964.
- [3] G. Aad et al. Observation of a new particle in the search for the standard model higgs boson with the ATLAS detector at the LHC. *Physics Letters B*, 716(1):1–29, sep 2012.
- [4] S. Chatrchyan et al. Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC. *Physics Letters B*, 716(1):30–61, sep 2012.
- [5] Standard Model Summary Plots February 2022. Technical report, CERN, Geneva, 2022. All figures including auxiliary figures are available at <https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2022-009>.
- [6] Lyndon Evans and Philip Bryant. LHC machine. *Journal of Instrumentation*, 3(08):S08001–S08001, aug 2008.
- [7] The ATLAS Collaboration. The ATLAS experiment at the CERN large hadron collider. *Journal of Instrumentation*, 3(08):S08003–S08003, aug 2008.
- [8] The CMS Collaboration. The CMS experiment at the CERN LHC. *Journal of Instrumentation*, 3(08):S08004–S08004, aug 2008.
- [9] The ALICE Collaboration. The ALICE experiment at the CERN LHC. *Journal of Instrumentation*, 3(08):S08002–S08002, aug 2008.
- [10] The LHCb Collaboration. The LHCb detector at the LHC. *Journal of Instrumentation*, 3(08):S08005–S08005, aug 2008.
- [11] Ewa Lopienska. The CERN accelerator complex, layout in 2022. Complexe des accélérateurs du CERN en janvier 2022. 2022. General Photo.

- 
- [12] ATLAS. The ATLAS detector, General Photo. <http://opendata.atlas.cern/release/2020/documentation/atlas/experiment.html>.
- [13] Joao Pequena. Computer generated image of the ATLAS inner detector. 2008.
- [14] Ana Maria Rodriguez Vera and Joao Antunes Pequena. ATLAS Detector Magnet System. General Photo, 2021.
- [15] Joao Pequena. Computer Generated image of the ATLAS calorimeter. 2008.
- [16] ATLAS liquid argon calorimeter: Technical design report. 12 1996.
- [17] *ATLAS Tile calorimeter: Technical Design Report*. Technical design report. ATLAS. CERN, Geneva, 1996.
- [18] Technical Design Report for the Phase-II Upgrade of the ATLAS Tile Calorimeter. Technical report, CERN, Geneva, 2017.
- [19] Joao Pequena. Computer generated image of the ATLAS Muons subsystem. 2008.
- [20] ATLAS Collaboration. Technical Design Report for the Phase-I Upgrade of the ATLAS TDAQ System. Technical Report CERN-LHCC-2013-018. ATLAS-TDR-023, Sep 2013. Final version presented to December 2013 LHCC.
- [21] H. Bertelsen, G. Carrillo Montoya, P.-O. Deviveiros, T. Eifert, G. Galster, J. Glatzer, S. Haas, A. Marzin, M.V. Silva Oliveira, T. Pauly, K. Schmieden, R. Spiwoks, and J. Stelzer. Operation of the upgraded ATLAS central trigger processor during the LHC run 2. *Journal of Instrumentation*, 11(02):C02020–C02020, feb 2016.
- [22] TDAQ. ATLAS DAQ Public Results. <https://twiki.cern.ch/twiki/bin/view/AtlasPublic/ApprovedPlotsDAQ>.
- [23] Ralf et al. Spiwoks. The ATLAS Muon-to-Central Trigger Processor Interface (MUCTPI) Upgrade. 2017.
- [24] R Achenbach et al. The ATLAS level-1 calorimeter trigger. *Journal of Instrumentation*, 3(03):P03001–P03001, mar 2008.
- [25] G. Aad et al. Performance of the upgraded PreProcessor of the ATLAS level-1 calorimeter trigger. *Journal of Instrumentation*, 15(11):P11016–P11016, nov 2020.
- [26] Xilinx. Spartan-6 FPGA Family. <https://www.xilinx.com/products/silicon-devices/fpga/spartan-6.html>.
- [27] Jan Jongmanns. *The Upgrade of the PreProcessor of the ATLAS Level-1 Calorimeter Trigger for LHC Run-2*. Phd thesis, Universität Heidelberg, 2017.



- [28] D. Husmann et al. Pre-Processor Asic User and Reference Manual, 2011.
- [29] Predrag Kuzmanovic. Development and Testing of the ALTI Module for Timing and Synchronization in the ATLAS experiment at CERN., Aug 2018. Presented 21 Sep 2018.
- [30] Carlo Alberto Gottardo. FELIX and SW ROD Commissioning of the New ATLAS Readout System. 2020.
- [31] Yasuyuki Okumura. Triggering in ATLAS in Run 2 and Run 3. 2022.
- [32] ATLAS. ATLAS Liquid Argon Calorimeter Phase-I Upgrade: Technical Design Report. Technical report, 2013. Final version presented to December 2013 LHCC.
- [33] Fibre Optic Exchange (FOX) Documentation (v1.2). Technical report, February 2018.
- [34] Emily Ann Smith. The phase-1 upgrade of the ATLAS level-1 calorimeter trigger. *PoS*, EPS-HEP2021:754, 2022.
- [35] Weiming Qian. Design and test performance of the ATLAS Feature Extractor trigger boards for the Phase-1 Upgrade. 2017.
- [36] ATLAS L1Calo. Documentation of the L1Calo algorithm specifications. <https://gitlab.cern.ch/l1calo-run3-simulation/documentation/Run3L1CaloOfflineSWReqs>.
- [37] ATLAS. gFEX overview. <https://gfex.cern.ch>.
- [38] Daniele Bertolini, Tucker Chan, and Jesse Thaler. Jet observables without jet algorithms. *Journal of High Energy Physics*, 2014(4), apr 2014.
- [39] L1Calo. Level-1 Calorimeter Trigger Public Results. <https://twiki.cern.ch/twiki/bin/view/AtlasPublic/L1CaloTriggerPublicResults>.
- [40]  $E_T^{\text{miss}}$  performance in the ATLAS detector using 2015-2016 LHC p-p collisions. 2018. All figures including auxiliary figures are available at <https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/CONFNOTES/ATLAS-CONF-2018-023>.
- [41] Xilinx Artix-7 FPGA Family. <http://www.xilinx.com/products/silicon-devices/fpga/artix-7.html>.
- [42] Xilinx Kintex UltraScale FPGA Family. <http://www.xilinx.com/products/silicon-devices/fpga/kintex-ultrascale.html>.

- 
- [43] Xilinx Zynq Ultrascale+ MPSoC Family. <https://www.xilinx.com/products/silicon-devices/soc/zynq-ultrascale-mpsoc.html>.
- [44] Samtec FireFly™ Active Optical Micro Flyover Cable Assembly. <https://www.samtec.com/products/ecuo>.
- [45] George Victor Andrei. The Data Path of the ATLAS Level-1 Calorimeter Trigger PreProcessor. [https://www.kip.uni-heidelberg.de/Veroeffentlichungen/download.php/4884/ps/vandrei\\_dissertation.pdf](https://www.kip.uni-heidelberg.de/Veroeffentlichungen/download.php/4884/ps/vandrei_dissertation.pdf).
- [46] Xilinx UltraScale Architecture Configuration. [https://www.xilinx.com/content/dam/xilinx/support/documents/user\\_guides/ug570-ultrascale-configuration.pdf](https://www.xilinx.com/content/dam/xilinx/support/documents/user_guides/ug570-ultrascale-configuration.pdf).
- [47] Maxim, DS4520, 9-bit Nonvolatile I/O Expander Plus Memory.
- [48] NXP Semiconductors, PCA9539A, Low-voltage 16-bit I2C-bus I/O port with interrupt and reset, Product Data Sheet.
- [49] Skyworks, Si5345/44/42, Rev D Data Sheet.
- [50] Skyworks, 10 MHz TO 1.4 GHz I2C PROGRAMMABLE XO/VCXO.
- [51] Linear Technology, LTC2977, 8-Channel PMBus Power System Manager Featuring Accurate Output Voltage Measurement.
- [52] Low-Power, 4-/8-/12-Channel, I2C, 12-Bit ADCs in Ultra-Small Packages. <https://datasheets.maximintegrated.com/en/ds/MAX11612-MAX11617.pdf>.
- [53] A Timing, Trigger and Control Receiver ASIC for LHC Detectors (v3.9). [http://ttc.web.cern.ch/TTC/TTCrx\\_manual3.9.pdf](http://ttc.web.cern.ch/TTC/TTCrx_manual3.9.pdf), 2004.
- [54] P Moreira, R Ballabriga, S Baron, S Bonacini, O Cobanoglu, F Faccio, T Fedorov, R Francisco, P Gui, P Hartin, K Kloukinas, X Llopert, A Marchioro, C Paillard, N Pinilla, K Wyllie, and B Yu. The GBT Project. 2009.
- [55] ATLAS TREX PRR Report. [https://edms.cern.ch/ui/file/1533071/0.2/TREX\\_PRR.pdf](https://edms.cern.ch/ui/file/1533071/0.2/TREX_PRR.pdf).
- [56] W. W. Peterson and D. T. Brown. Cyclic codes for error detection. *Proceedings of the IRE*, 49(1):228–235, 1961.
- [57] Xilinx. UltraScale Architecture GTH Transceivers. <https://docs.xilinx.com/v/u/en-US/ug576-ultrascale-gth-transceivers>.

- [58] Eduardo Mendes, Sophie Baron, Csaba Soos, Jan Troska, and Paolo Novellini. Achieving picosecond-level phase stability in timing distribution systems with xilinx ultrascale transceivers. *IEEE Transactions on Nuclear Science*, 67(3):473–481, 2020.
- [59] Agilent. Agilent HDMP 1032-1034 Transmitter- Receiver Chip-set Datasheet. <http://www.physics.ohio-state.edu/~cms/cfeb/datasheets/hdmp1032.pdf>.
- [60] Serguei Kolos, William P. Vazquez, and Gordon Crone. New Software-Based Readout Driver for the ATLAS Experiment. Technical report, CERN, Geneva, Jun 2021.
- [61] D.P.C. Sankey. ATLAS L1Calo Pre-processor S-Link data formats. [https://edms.cern.ch/ui/file/761046/5/PPM\\_S-Link\\_data\\_formats\\_v2.1.pdf](https://edms.cern.ch/ui/file/761046/5/PPM_S-Link_data_formats_v2.1.pdf).
- [62] ARM. Microprocessor Cores and Processor Technology. <https://www.arm.com>.
- [63] A Barriuso Poy, H Boterenbrood, H J Burckhart, J Cook, V Filimonov, S Franz, O Gutzwiller, B Hallgren, V Khomutnikov, S Schlenker, and F Varela. The detector control system of the ATLAS experiment. *Journal of Instrumentation*, 3(05):P05006–P05006, may 2008.
- [64] Trenc Electronic, TE0820-03-3AE21FA, MPSoC Module with Xilinx Zynq UltraScale+ ZU3CG-1E, e.g.
- [65] Wishbone I2C Master Core . <https://opencores.org/svnget/i2c?file=%2Ftrunk%2Fi2c>.
- [66] Xilinx. Zynq UltraScale+ MPSoC Processing System. <https://docs.xilinx.com/v/u/3.2-English/pg201-zynq-ultrascale-plus-processing-system>.
- [67] Denx Software Engineering. Das U-Boot – the Universal Boot Loader. <https://www.denx.de/wiki/U-Boot>.
- [68] Xilinx. PetaLinux Tools. <https://www.xilinx.com/products/design-tools/embedded-software/petalinux-sdk.html>.
- [69] Yocto. Yocto Project. <https://www.yoctoproject.org>.
- [70] CentOS Project. CentOS Linux distribution. <https://www.centos.org>.
- [71] Wolfgang Mahnke, Stefan-Helmut Leitner, and Matthias Damm. *OPC Unified Architecture*. Springer, Berlin, 2009.
- [72] Quick opcUA Server generAtion fRamework.
- [73] Digilent, JTAG-SMT2™ Programming Module for Xilinx® FPGAs.

- 
- [74] Xilinx. Vivado Framework. <https://www.xilinx.com/products/design-tools/vivado.html>.
- [75] InfluxData. InfluxDB: Open Source Time Series Database. <https://www.influxdata.com>.
- [76] Grafana Labs. Grafana: The open observability platform. <https://grafana.com>.
- [77] Xilinx. UltraScale GTH Transceiver: TX and RX latency values. <https://support.xilinx.com/s/article/64309>.
- [78] Jörn Schumacher, Christian Plessl, and Wainer Vandelli. High-throughput and low-latency network communication with netio. *Journal of Physics: Conference Series*, 898(8):082003, oct 2017.
- [79] ATLAS. Countdown to physics: Beams splash in the ATLAS experiment. <https://atlas.cern/Updates/News/Run3-beams-splash>.
- [80] ATLAS. ATLAS Data Quality Monitoring. <http://atlasdqm.web.cern.ch/atlasdqm/>.

## Acknowledgements

I am immensely grateful to my supervisor Prof. Dr. Hans-Christian Schultz-Coulon, who provided me with the opportunity to work on a subject I am deeply passionate about and taught me how to become an independent researcher. In putting his trust in me for such a big responsibility and supporting me throughout my journey.

I want to thank Prof. Dr. Ulrich Uwer for being my referee and dedicating his time to reading my thesis.

I am grateful to Klaus Schmitt, who took me under his wing and taught me how to view and solve problems from a lower-level hardware perspective. For the countless hours spent testing the hardware side-by-side, discussing Linux hardware, embedded devices and much more. In addition, my thanks go to Peter Stock and Victor Andrei.

I'm thankful to Martin Wessels for the many discussions about the structures of L1Calo and the online software, his knowledge of the fine details of the L1Calo system was essential. I want to thank Pavel Starovoitov and Rainer Stamen for making sure I was comfortable and for their constant readiness to answer any questions I would have.

I would like to thank Rainer Stamen, Vera Stankova, Varsiha Sothilingam, Thomas Junkermann, Anke Ackermann and Mathias Backes for taking the time to proofread my thesis.

I would like to thank the L1Calo community, especially Murrough Landon and Rhys Owen for the many hours of debugging sessions and their essential support during commissioning and operations. A warm thank you goes to the TileCal community and my friends at CERN, in particular, Seyedali Moayedi and Fernando Carrio Argos.

A heartfelt thank you to everyone in the KIP F8, F11 and S4 groups. Special thanks also go to Petra Pfeifer for putting up with me and all of my paperwork for the many CERN trips.

Last but not least, I want to thank my parents and my sister who show unconditional love and support in all of my pursuits.